



ELSEVIER

Contents lists available at ScienceDirect

## Journal of Combinatorial Theory, Series A

[www.elsevier.com/locate/jcta](http://www.elsevier.com/locate/jcta)



# Variants of the RSK algorithm adapted to combinatorial Macdonald polynomials <sup>☆</sup>



Nicholas A. Loehr <sup>a,b</sup>

<sup>a</sup> *Dept. of Mathematics, Virginia Tech, Blacksburg, VA 24061-0123, United States*

<sup>b</sup> *Mathematics Department, United States Naval Academy, Annapolis, MD 21402-5002, United States*

### ARTICLE INFO

#### Article history:

Received 16 August 2015

Available online xxxx

#### Keywords:

RSK algorithm

Robinson–Schensted correspondence

Macdonald polynomials

Schur polynomials

Permutation statistics

### ABSTRACT

We introduce variations of the Robinson–Schensted correspondence parametrized by positive integers  $p$ . Each variation gives a bijection between permutations and pairs of standard tableaux of the same shape. In addition to sharing many of the properties of the classical Schensted algorithm, the new algorithms are designed to be compatible with certain permutation statistics introduced by Haglund in the study of Macdonald polynomials. In particular, these algorithms provide an elementary bijective proof converting Haglund’s combinatorial formula for Macdonald polynomials to an explicit combinatorial Schur expansion of Macdonald polynomials indexed by partitions  $\mu$  satisfying  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ . We challenge the research community to extend this RSK-based approach to more general classes of partitions.

© 2016 Elsevier Inc. All rights reserved.

This paper investigates and advocates an algorithmic approach to Macdonald’s famous conjecture [19] requesting a combinatorial interpretation for the  $q, t$ -Kostka matrix, which gives the Schur expansion of modified Macdonald polynomials. The main idea is to invent

<sup>☆</sup> This work was partially supported by a grant from the Simons Foundation (#244398 to Nicholas Loehr).  
E-mail address: [nloehr@vt.edu](mailto:nloehr@vt.edu).

variations of the Robinson–Schensted algorithm that can be used to translate Haglund’s combinatorial formula for Macdonald polynomials [10] into the required Schur expansion. We assume the reader is familiar with general background on permutations, partitions, tableaux, and symmetric polynomials (as found, for instance, in [8, Part I], [18, Chap. 10], [20, Chap. I], [22, Chap. 3], and [23, Chap. 7]). However, no specific prior knowledge of Macdonald polynomials will be required. For brevity, we refer to the Robinson–Schensted bijection (which maps permutations to pairs of standard tableaux) as the *RSK algorithm*, although we will not need the general version of this algorithm due to Knuth (which maps matrices to pairs of semistandard tableaux).

The paper is structured as follows. Section 1 motivates our invention of new RSK algorithms by describing Macdonald polynomials, Schur polynomials, Macdonald’s conjecture on the Schur expansion of Macdonald polynomials, prior progress on this conjecture, and our proposed algorithmic approach to this conjecture. Readers interested primarily in the combinatorics of RSK algorithms may safely omit this section. Section 2 reviews the ingredients of the classical Schensted algorithm on permutations—row insertion, tableau insertion, recording tableaux, and how to invert RSK. These elements are then generalized in Section 3 to define RSK algorithms parametrized by positive integers  $p$ . Section 4 proves that these new algorithms share many of the combinatorial properties of classical RSK; in particular, the recording tableaux for the new algorithms always agree with the classical recording tableau. Section 5 applies the parametrized RSK algorithms to give elementary bijective proofs of combinatorial formulas for the Schur expansions of Macdonald polynomials  $\tilde{H}_\mu$  indexed by partitions  $\mu$  satisfying  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ .

## 1. Motivating problem: the Schur expansion of Macdonald polynomials

### 1.1. Haglund’s combinatorial formula for Macdonald polynomials

Ian Macdonald introduced the symmetric polynomials now called *Macdonald polynomials* in 1988 [19]. There are several versions of Macdonald polynomials, each of which can be defined algebraically as the unique symmetric polynomials satisfying certain orthogonality or triangularity conditions. In this paper, we work with the *modified Macdonald polynomials*, denoted  $\tilde{H}_\mu(X; q, t)$ , indexed by integer partitions  $\mu$ . In 2004, Haglund [10] discovered a celebrated combinatorial formula for the fundamental quasisymmetric expansion of these polynomials in terms of new permutation statistics  $\text{inv}_\mu$  and  $\text{maj}_\mu$  depending on  $\mu$ . This formula was proved by Haglund, Haiman, and Loehr [11]. To minimize technical prerequisites for reading this paper, we will take Haglund’s combinatorial formula as our *definition* of the modified Macdonald polynomials. See [11] for the original algebraic definitions of Macdonald polynomials and their connection to Haglund’s formula.

Let  $\text{Par}(n)$  denote the set of all integer partitions of  $n$ , let  $[n]$  denote the set  $\{1, 2, \dots, n\}$ , and let  $S_n$  denote the set of permutations of  $[n]$ . Given a partition  $\mu \in \text{Par}(n)$ , Haglund’s formula has the form

$$\tilde{H}_\mu(X; q, t) = \sum_{w \in S_n} q^{\text{inv}_\mu(w)} t^{\text{maj}_\mu(w)} F_{n, \text{IDes}(w)}(X), \quad (1)$$

where the notation on the right side will be defined next.

Fix  $w \in S_n$ , and write  $w$  in one-line form as a word  $w = w_1 w_2 \cdots w_n$ . The *inverse descent set* of  $w$ , denoted  $\text{IDes}(w)$ , is the set of all  $i < n$  such that  $i + 1$  appears to the left of  $i$  in the word  $w$ . For any  $T \subseteq [n - 1]$ , Gessel's *fundamental quasisymmetric polynomial* in the set of variables  $X = \{x_1, x_2, \dots, x_n, \dots\}$  is

$$F_{n,T}(X) = \sum_{\substack{i_1 \leq i_2 \leq \cdots \leq i_n: \\ j \in T \Rightarrow i_j < i_{j+1}}} x_{i_1} x_{i_2} \cdots x_{i_n}.$$

The only fact about fundamental quasisymmetric polynomials needed here is that every symmetric polynomial can be written uniquely as a linear combination of the polynomials  $F_{n,T}$ .

To compute  $\text{inv}_\mu(w)$  and  $\text{maj}_\mu(w)$ , we first fill the Ferrers diagram of  $\mu$  (written in French notation—longest row at the bottom) with the letters  $w_1 \cdots w_n$ , filling each row from left to right starting with the top (shortest) row. We now obtain *column words* by reading each column from top to bottom. For any word  $v = v_1 v_2 \cdots v_s$ , let  $\text{maj}(v)$  be the sum of all  $i < s$  with  $v_i > v_{i+1}$ . Let  $\text{maj}_\mu(w)$  be the sum of  $\text{maj}(v)$  over all column words  $v$  for  $w$ .

The statistic  $\text{inv}_\mu(w)$  counts certain triples of cells in the filled diagram for  $w$  called inversion triples. Consider three cells in the Ferrers diagram of  $\mu$  (French notation) positioned as shown here:

$$\begin{array}{|c|} \hline a \\ \hline b \\ \hline \end{array} \quad \begin{array}{|c|} \hline c \\ \hline \end{array}$$

Thus, value  $b$  is in the cell directly below value  $a$ , and value  $c$  is somewhere to the right of  $a$  in the same row. We also allow  $a$  and  $c$  to be in the bottom row of the diagram of  $\mu$ , in which case we take  $b = \infty$ . The triple of cells shown here is an *inversion triple* iff  $a < b < c$  or  $b < c < a$  or  $c < a < b$ .

**Example 1.** Let  $n = 9$ ,  $w = 783695142 \in S_9$ , and  $\mu = (3, 3, 2, 1) \in \text{Par}(9)$ . The filled Ferrers diagram of  $\mu$  appears below.

$$\begin{array}{|c|c|c|} \hline 7 & & \\ \hline 8 & 3 & \\ \hline 6 & 9 & 5 \\ \hline 1 & 4 & 2 \\ \hline \end{array}$$

The column words are 7861, 394, and 52, so

$$\text{maj}_\mu(w) = \text{maj}(7861) + \text{maj}(394) + \text{maj}(52) = 5 + 2 + 1 = 8.$$

The following values  $(a, b, c)$  give inversion triples:  $(6, 1, 5)$ ,  $(9, 4, 5)$  and  $(4, \infty, 2)$ . So  $\text{inv}_\mu(w) = 3$ . Finally,  $\text{IDes}(w) = \{2, 4, 5, 6\}$ , so this permutation contributes the term  $q^3 t^8 F_{9, \{2, 4, 5, 6\}}$  to the expansion of  $\tilde{H}_{(3, 3, 2, 1)}$ .

**Example 2.** By computing a term for each  $w \in S_3$ , we find that

$$\tilde{H}_{(2, 1)} = 1F_{3, \emptyset} + (q + t)F_{3, \{1\}} + (q + t)F_{3, \{2\}} + qtF_{3, \{1, 2\}}.$$

### 1.2. Tableaux and Schur polynomials

Given a partition  $\lambda \in \text{Par}(n)$ , a *semistandard Young tableau* of shape  $\lambda$  is a filling of the cells in the Ferrers diagram of  $\lambda$  (French notation) with positive integers, so that values weakly increase reading from left to right in each row, and values strictly increase reading from bottom to top in each column. A *standard* tableau is a semistandard Young tableau in which the values  $1, 2, \dots, n$  appear once each. Let  $\text{SYT}(\lambda)$  be the set of all standard Young tableaux of shape  $\lambda$ .

For each  $\lambda \in \text{Par}(n)$ , the *Schur symmetric polynomial*  $s_\lambda(X)$  can be defined combinatorially [18, §10.3] as the sum of the content monomials of all semistandard Young tableaux of shape  $\lambda$ . We will find it more convenient to define Schur polynomials via their fundamental quasisymmetric expansions, leading to a sum indexed by standard tableaux. Given a standard tableau  $T$  with  $n$  cells, the *descent set of  $T$*  (denoted  $\text{Des}(T)$ ) is the set of all  $i < n$  such that the value  $i$  appears in a lower row of  $T$  than the value  $i + 1$  (French notation). We take as our definition of Schur polynomials the expansion [23, 7.19.7]

$$s_\lambda(X) = \sum_{T \in \text{SYT}(\lambda)} F_{n, \text{Des}(T)}(X). \quad (2)$$

**Example 3.** Given the partition  $\lambda = (3, 2) \in \text{Par}(5)$ , we compute

$$\text{SYT}(\lambda) = \left\{ \begin{array}{|c|c|c|} \hline 4 & 5 & \\ \hline 1 & 2 & 3 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 3 & 5 & \\ \hline 1 & 2 & 4 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 3 & 4 & \\ \hline 1 & 2 & 5 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 2 & 5 & \\ \hline 1 & 3 & 4 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array} \right\}.$$

From left to right, these tableaux have descent sets  $\{3\}$ ,  $\{2, 4\}$ ,  $\{2\}$ ,  $\{1, 4\}$ , and  $\{1, 3\}$ , so

$$s_{(3, 2)} = F_{5, \{3\}} + F_{5, \{2, 4\}} + F_{5, \{2\}} + F_{5, \{1, 4\}} + F_{5, \{1, 3\}}.$$

**Example 4.** For the partitions of  $n = 4$ , we find that  $s_{(4)} = F_{4, \emptyset}$ ,  $s_{(3, 1)} = F_{4, \{1\}} + F_{4, \{2\}} + F_{4, \{3\}}$ ,  $s_{(2, 2)} = F_{4, \{2\}} + F_{4, \{1, 3\}}$ ,  $s_{(2, 1, 1)} = F_{4, \{1, 2\}} + F_{4, \{1, 3\}} + F_{4, \{2, 3\}}$ , and  $s_{(1, 1, 1, 1)} = F_{4, \{1, 2, 3\}}$ .

Given a standard tableau  $T$ , the *reading word* of  $T$ , denoted  $\text{rw}(T)$ , is the permutation in  $S_n$  obtained by reading the entries in the rows of  $T$  from left to right, starting with

the top row (in French notation) and working down. For instance, the tableaux shown in [Example 3](#) above have reading words 45123, 35124, 34125, 25134, and 24135, respectively. We can recover the standard tableau  $T$  from its reading word, by writing the symbols in  $w = \text{rw}(T)$  from left to right, moving down to a new row every time we encounter a descent  $w_i > w_{i+1}$ . Moreover, it is routine to check that  $w \in S_n$  is the reading word of some  $T \in \text{SYT}(\lambda)$  iff  $w_1 w_2 \cdots w_n$  has ascending runs of lengths given by the parts of  $\lambda$  (from smallest to largest), and for all  $k$ , the subword of  $w$  consisting of the  $k$ th entry in each ascending run is a strictly decreasing sequence. In some discussions, it is convenient to *identify* a standard tableau  $T \in \text{SYT}(\lambda)$  with its reading word in  $S_n$ , writing  $w = \text{rw}(T) = T$ . Similar comments apply to *partial standard tableaux*, which are semistandard Young tableaux with no repeated values. The reading word of such a tableau is a *partial permutation*, which is a word with no repeated values.

### 1.3. The Schur expansion of Macdonald polynomials

Let  $\mu$  be a partition of  $n$ . Like any symmetric polynomial, the modified Macdonald polynomial  $\tilde{H}_\mu$  has a unique expansion as a linear combination of Schur polynomials, say

$$\tilde{H}_\mu(X; q, t) = \sum_{\lambda \in \text{Par}(n)} \tilde{K}_{\lambda, \mu}(q, t) s_\lambda, \quad (3)$$

where the scalars  $\tilde{K}_{\lambda, \mu}(q, t) \in \mathbb{Q}(q, t)$  are called *modified  $q, t$ -Kostka coefficients*. This Schur expansion contains a lot of important combinatorial and algebraic information; for instance, the coefficient of  $s_\lambda$  in  $\tilde{H}_\mu$  gives the bigraded multiplicity of the irreducible  $S_n$ -module indexed by  $\lambda$  in the bigraded Garsia–Haiman module  $M_\mu$  indexed by  $\mu$  [9,14].

In his original paper on Macdonald polynomials [19], Macdonald conjectured that for all  $\lambda, \mu \in \text{Par}(n)$ , there should exist two combinatorial statistics  $\tilde{a}_\mu$  and  $\tilde{b}_\mu$ , defined on standard tableaux of shape  $\lambda$ , such that

$$\tilde{K}_{\lambda, \mu}(q, t) = \sum_{T \in \text{SYT}(\lambda)} q^{\tilde{a}_\mu(T)} t^{\tilde{b}_\mu(T)}. \quad (4)$$

Macdonald’s famous problem, which is still unsolved in the general case, is to find and prove specific formulas for the statistics  $\tilde{a}_\mu$  and  $\tilde{b}_\mu$ . (In fact, Macdonald stated his conjecture for the related expansion  $H_\mu = \sum_\lambda K_{\lambda, \mu}(q, t) s_\lambda$ , where  $H_\mu(X; q, t) = q^{n(\mu)} \tilde{H}_\mu(X; q, 1/t)$  is a simple transformation of the modified Macdonald polynomials used here. One can readily incorporate this transformation into the combinatorics. For example, Haglund’s formula for  $\tilde{H}_\mu$  becomes a formula for  $H_\mu$  if one replaces  $\text{maj}_\mu$  by  $\text{comaj}_\mu$ , which is the sum of the positions of the *ascents* in all the column words. For this reason, we deal only with  $\tilde{H}_\mu$  in this paper.)

#### 1.4. Progress on Macdonald's conjecture

Various special cases of Macdonald's conjecture have been solved in the many years since its original formulation in 1988. Formulas for  $\tilde{K}_{\lambda,\mu}(q,t)$  are known when  $\lambda$  or  $\mu$  is a hook partition (i.e., at most one part of the partition exceeds one). In 1995, Susanna Fishel [7] gave combinatorial formulas for the Schur expansion of  $\tilde{H}_\mu$  when  $\mu$  is a two-column partition (i.e.,  $\mu_1 \leq 2$ ), using objects called *rigged configurations*. In 1999, Mike Zabrocki [25] used an approach based on vertex operators to obtain combinatorial formulas in the case where  $\mu_1 \leq 4$  and  $\mu_2 \leq 2$ . In 2003, Luc Lapointe and Jennifer Morse [16] found tableau-based statistics when  $\mu$  has two columns, using a mixture of creation operators and tableau combinatorics. Invoking the identity  $\tilde{H}_{\mu'}(X;q,t) = \tilde{H}_\mu(X;t,q)$ , these results lead to formulas for the  $q,t$ -Kostka coefficients when  $\mu$  is a two-row shape.

We have adopted Haglund's 2003 combinatorial formula [10] as our definition of modified Macdonald polynomials. In fact, the discovery of this formula was a major breakthrough, enabling researchers to replace the complicated algebraic definition of Macdonald polynomials by an explicit combinatorial expansion. Different versions of Haglund's formula express  $\tilde{H}_\mu$  as linear combinations of either individual monomials, fundamental quasisymmetric polynomials, or versions of Lascoux–Leclerc–Thibon (LLT) polynomials indexed by tuples of ribbon shapes [11,12]. It turns out that Macdonald polynomials indexed by two-column partitions  $\mu$  are sums of LLT polynomials indexed by tuples of two ribbon shapes, which can be rewritten in terms of domino LLT polynomials [17]. A combinatorial rule for the Schur expansion of domino LLT polynomials was stated by Carré and Leclerc [6], with many details of the proof filled in later by van Leeuwen [24]. Combining this rule with Haglund's formula leads to another combinatorial formula for  $q,t$ -Kostka coefficients when  $\mu$  has at most two columns; see [11, Sect. 9].

Another recent advance came with Sami Assaf's theory of *dual equivalence graphs* [2, 3]. Assaf [1] and Austin Roberts [21] used this technique to find formulas for  $q,t$ -Kostka numbers in the case  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ . Assaf's more general theory of *D-graphs* offers a potential approach to finding similar formulas for all  $q,t$ -Kostka coefficients, although a new paper of Jonah Blasiak [5] has called attention to some unexpected technical barriers. In other recent work [4], Blasiak analyzed noncommutative Schur functions in Lam's algebra of ribbon Schur operators to find the Schur expansion for LLT polynomials indexed by tuples of three shapes. Using Haglund's formula, this result translates into a formula for the  $q,t$ -Kostka numbers  $\tilde{K}_{\lambda,\mu}(q,t)$  when  $\mu$  is a partition with at most three columns.

#### 1.5. $q,t$ -Kostka polynomials based on inverse reading words

In this paper, we will prove the following Schur expansion.

**Table 1**  
Computation of  $\tilde{H}_{(3,1)}$  and  $\tilde{H}_{(2,2)}$  via [Theorem 5](#).

shape $\lambda$	reading words for $T \in \text{SYT}(\lambda)$	$w = \text{rw}(T)^{-1}$	$q^{\text{inv}_\mu(w)} t^{\text{maj}_\mu(w)}$ $\mu = (3, 1)$	$q^{\text{inv}_\mu(w)} t^{\text{maj}_\mu(w)}$ $\mu = (2, 2)$
	1234	1234	$q^0 t^0$	$q^0 t^0$
	2134	2134	$q^0 t^1$	$q^1 t^0$
	3124	2314	$q^1 t^0$	$q^0 t^1$
	4123	2341	$q^2 t^0$	$q^1 t^1$
	3412	3412	$q^2 t^0$	$q^0 t^2$
	2413	3142	$q^1 t^1$	$q^2 t^0$
	4312	3421	$q^3 t^0$	$q^1 t^2$
	4213	3241	$q^2 t^1$	$q^2 t^1$
	3214	3214	$q^1 t^1$	$q^1 t^1$
	4321	4321	$q^3 t^1$	$q^2 t^2$

**Theorem 5.** For all  $n \geq 1$  and all  $\mu \in \text{Par}(n)$  such that  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ ,

$$\tilde{H}_\mu(X; q, t) = \sum_{\lambda \in \text{Par}(n)} \left( \sum_{T \in \text{SYT}(\lambda)} q^{\text{inv}_\mu(\text{rw}(T)^{-1})} t^{\text{maj}_\mu(\text{rw}(T)^{-1})} \right) s_\lambda(X).$$

Here  $\text{rw}(T)^{-1}$  denotes the inverse of the reading word of  $T$  in the group  $S_n$ .

This formula is similar to [\[11, Prop. 9.2\]](#) and [\[21, Cor. 4.4\]](#), both of which express  $\tilde{K}_{\lambda, \mu}(q, t)$  as a sum over Yamanouchi words of content  $\lambda$ . Those references obtain their results using the machinery of type A crystals and dual equivalence graphs (respectively). The main contribution here is our algorithmic approach based on new RSK variations (described below), which leads to a completely elementary and fully bijective proof of the theorem. Not surprisingly, these algorithms are closely connected to the dual equivalence graph theory studied by Assaf, Blasiak, Roberts, et al., as we shall see later.

**Example 6.** [Table 1](#) illustrates [Theorem 5](#) by computing the Schur expansions of  $\tilde{H}_{(3,1)}$  and  $\tilde{H}_{(2,2)}$ .

The table shows that

$$\begin{aligned} \tilde{H}_{(3,1)} &= 1s_{(4)} + (t + q + q^2)s_{(3,1)} + (q^2 + qt)s_{(2,2)} + (q^3 + q^2t + qt)s_{(2,1,1)} \\ &\quad + (q^3t)s_{(1,1,1,1)}; \\ \tilde{H}_{(2,2)} &= 1s_{(4)} + (q + t + qt)s_{(3,1)} + (t^2 + q^2)s_{(2,2)} + (qt^2 + q^2t + qt)s_{(2,1,1)} \\ &\quad + (q^2t^2)s_{(1,1,1,1)}. \end{aligned}$$

The formula in [Theorem 5](#) fails for the partition  $\mu = (4)$ : the coefficient of  $s_{(2,2)}$  is  $q^2 + q^4$  on the left side and  $q^3 + q^4$  on the right side.

### 1.6. RSK algorithms adapted to $\mu$

The famous Robinson–Schensted algorithm (see, e.g., [8, Part I], [18, Chap. 10], [22, Chap. 3]) is a bijection

$$\text{RSK} : S_n \rightarrow \bigcup_{\lambda \in \text{Par}(n)} \text{SYT}(\lambda) \times \text{SYT}(\lambda),$$

denoted  $\text{RSK}(w) = (P(w), Q(w))$  for  $w \in S_n$ , that sends a permutation  $w$  to a pair of standard tableaux  $P(w)$ ,  $Q(w)$  of the same shape  $\lambda \in \text{Par}(n)$ .  $P(w)$  is called the *insertion tableau* for  $w$ , and  $Q(w)$  is called the *recording tableau* for  $w$ . The details of this algorithm are recalled in greater depth in §2. As mentioned earlier, Knuth [15] generalized the Robinson–Schensted algorithm to a bijection from matrices to pairs of semistandard tableaux of the same shape. This generalization will not be discussed in this paper, but we still use the abbreviation RSK to honor Knuth’s contribution. In particular, the *Knuth relations* (discussed in §4.3) play a key role in implementing and analyzing the Schensted algorithm and its variations.

The RSK algorithm has many remarkable combinatorial properties. For example, given  $w \in S_n$ , define the *descent set* of  $w$  to be  $\text{Des}(w) = \{i < n : w_i > w_{i+1}\}$ . Note that the inverse descent set of  $w$  (defined in §1.1) can be written  $\text{IDes}(w) = \text{Des}(w^{-1})$ . It is known [18, Thm. 10.117] that for all  $w \in S_n$ ,  $\text{Des}(w) = \text{Des}(Q(w))$  (recall the descent set of a standard tableau was defined in §1.2). Another celebrated property [18, Thm. 10.112] is that if  $\text{RSK}(w) = (P, Q)$ , then  $\text{RSK}(w^{-1}) = (Q, P)$ . It follows that  $\text{IDes}(w) = \text{Des}(w^{-1}) = \text{Des}(Q(w^{-1})) = \text{Des}(P(w))$  for all  $w \in S_n$ .

Now we come to the key idea of this paper. Fix a partition  $\mu \in \text{Par}(n)$ . Substituting the expansion (2) and the conjecture (4) into the formula (3) gives the expansion

$$\tilde{H}_\mu(X; q, t) = \sum_{\lambda \in \text{Par}(n)} \sum_{P \in \text{SYT}(\lambda)} \sum_{Q \in \text{SYT}(\lambda)} q^{\tilde{a}_\mu(Q)} t^{\tilde{b}_\mu(Q)} F_{n, \text{Des}(P)}(X). \quad (5)$$

Compare this formula to Haglund’s expansion (1), which is a sum indexed by permutations in  $S_n$ . We could bijectively deduce (5) from (1) if we could find an *RSK-like map*  $\text{RSK}^\mu$  sending each  $w \in S_n$  to a pair of standard tableaux  $(P^\mu(w), Q^\mu(w))$  satisfying the properties stated in the theorem below. The desired tableau statistics  $\tilde{a}_\mu(Q)$  and  $\tilde{b}_\mu(Q)$  will emerge automatically by translating the statistics  $\text{inv}_\mu$  and  $\text{maj}_\mu$  through the bijection. In essence, rather than approaching conjecture (4) by looking for *statistics* on standard tableaux parametrized by  $\mu$ , we instead seek *RSK algorithms* parametrized by  $\mu$ , and then “pull back” to Haglund’s permutation statistics to find  $\tilde{a}_\mu$  and  $\tilde{b}_\mu$ . The next theorem gives the formal statement of this approach.

**Theorem 7.** Fix a partition  $\mu \in \text{Par}(n)$ . Assume we can construct an algorithm  $\text{RSK}^\mu = (P^\mu, Q^\mu)$  satisfying these properties:



- (i)  $\text{RSK}^\mu$  is a bijection from  $S_n$  to  $\bigcup_{\lambda \in \text{Par}(n)} \text{SYT}(\lambda) \times \text{SYT}(\lambda)$ .
- (ii) For all  $w \in S_n$ ,  $\text{IDes}(w) = \text{Des}(P^\mu(w))$ .
- (iii) For all  $w \in S_n$ , the values of  $\text{inv}_\mu(w)$  and  $\text{maj}_\mu(w)$  depend only on  $Q^\mu(w)$ , not on  $P^\mu(w)$ .

Then for every partition  $\lambda$ , (4) holds with

$$\begin{cases} \tilde{a}_\mu(T) = \text{inv}_\mu(w) & \text{for any } w \in S_n \text{ such that } Q^\mu(w) = T; \\ \tilde{b}_\mu(T) = \text{maj}_\mu(w) & \text{for any } w \in S_n \text{ such that } Q^\mu(w) = T. \end{cases} \quad (6)$$

Equivalently, given  $Q \in \text{SYT}(\lambda)$  and any fixed  $P \in \text{SYT}(\lambda)$ , we are defining

$$\begin{cases} \tilde{a}_\mu(Q) &= \text{inv}_\mu((\text{RSK}^\mu)^{-1}(P, Q)); \\ \tilde{b}_\mu(Q) &= \text{maj}_\mu((\text{RSK}^\mu)^{-1}(P, Q)), \end{cases} \quad (7)$$

which are independent of the choice of  $P$  by (iii).

**Proof.** For any standard tableau  $Q$ , define  $\tilde{a}_\mu(Q)$  and  $\tilde{b}_\mu(Q)$  as in the theorem statement; this is well-defined by properties (i) and (iii). Start with Haglund's known formula

$$\tilde{H}_\mu(X; q, t) = \sum_{w \in S_n} q^{\text{inv}_\mu(w)} t^{\text{maj}_\mu(w)} F_{n, \text{IDes}(w)}(X).$$

By property (i), we can use the bijection  $\text{RSK}^\mu$  to change the summation over  $w \in S_n$  to a summation over  $(P, Q) \in \bigcup_{\lambda \in \text{Par}(n)} \text{SYT}(\lambda) \times \text{SYT}(\lambda)$ . Temporarily denote the inverse of  $\text{RSK}^\mu$  by  $R^{-1}$ . By (7) and property (ii), we obtain:

$$\begin{aligned} \tilde{H}_\mu(X; q, t) &= \sum_{\lambda \in \text{Par}(n)} \sum_{P \in \text{SYT}(\lambda)} \sum_{Q \in \text{SYT}(\lambda)} q^{\text{inv}_\mu(R^{-1}(P, Q))} t^{\text{maj}_\mu(R^{-1}(P, Q))} \\ &\quad \times F_{n, \text{IDes}(R^{-1}(P, Q))}(X) \\ &= \sum_{\lambda \in \text{Par}(n)} \sum_{P \in \text{SYT}(\lambda)} F_{n, \text{Des}(P)}(X) \sum_{Q \in \text{SYT}(\lambda)} q^{\tilde{a}_\mu(Q)} t^{\tilde{b}_\mu(Q)} \\ &= \sum_{\lambda \in \text{Par}(n)} s_\lambda(X) \left( \sum_{Q \in \text{SYT}(\lambda)} q^{\tilde{a}_\mu(Q)} t^{\tilde{b}_\mu(Q)} \right), \end{aligned}$$

where the last step used (2).  $\square$

**Example 8.** As an initial example of this theorem, consider the partition  $\mu = (1^n)$ . We claim that we may take  $\text{RSK}^\mu$  to be the classical RSK algorithm in this case. On one hand, (i) and (ii) are known properties of RSK. On the other hand, when  $\mu = (1^n)$ , one readily

checks that for  $w \in S_n$ ,  $\text{inv}_\mu(w) = 0$  and  $\text{maj}_\mu(w) = \sum_{i \in \text{Des}(w)} i = \sum_{i \in \text{Des}(Q(w))} i$ . This expression depends only on  $Q(w)$ , so (iii) holds.

In fact, we claim that for  $\mu = (2, 1^{n-1})$  we may also take  $\text{RSK}^\mu$  to be the classical RSK algorithm. To check condition (iii) for this  $\mu$ , note that  $\text{inv}_\mu(w) = 1$  if  $w_{n-1} > w_n$  and 0 otherwise, so that  $\text{inv}_\mu$  can be computed knowing only  $\text{Des}(w) = \text{Des}(Q(w))$ . Similarly,  $\text{maj}_\mu(w)$  is the sum of all  $i < n - 2$  such that  $i \in \text{Des}(w) = \text{Des}(Q(w))$ , so that (iii) holds.

For other choices of the partition  $\mu$ , the classical RSK algorithm does not satisfy condition (iii). In the rest of this paper, we show how to execute our proposed approach to Macdonald's conjecture by using [Theorem 7](#) to prove [Theorem 5](#). Specifically, we will construct algorithms  $\text{RSK}^\mu$ , for each partition  $\mu$  satisfying  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ , having properties (i), (ii), and (iii). It turns out to be more convenient initially to index our new RSK algorithms by positive integers  $p$ , rather than by partitions. We define these algorithms in [§3](#) after reviewing the relevant ingredients of the classical RSK algorithm in [§2](#). We expect these RSK variants to have substantial combinatorial interest in their own right, in addition to their relationship to Macdonald polynomials given in [Theorem 7](#).

## 2. Review of the RSK algorithm

### 2.1. Row insertion

We can think of the Robinson–Schensted algorithm, mapping permutations to pairs of standard tableaux, as being built up from simpler component subroutines. The first such subroutine, called *row insertion*, takes as input a strictly increasing sequence  $S$  and an integer  $a$  not appearing in  $S$ . The row insertion subroutine inserts  $a$  into  $S$ , producing as output a new increasing sequence  $S'$  and (optionally) a new value  $b$ . In more detail, let  $S$  be the increasing sequence  $s_1 < s_2 < \cdots < s_k$ . If  $S$  is empty or  $s_k < a$ , then the output  $S'$  is  $s_1 < s_2 < \cdots < s_k < a$ , and no value of  $b$  is produced; we call this the **terminating case**. Otherwise, let  $b$  be the smallest  $s_j$  such that  $a < s_j$ , and let  $S'$  be  $S$  with  $b$  replaced by  $a$ . Identifying the strictly increasing sequence  $s_1 < s_2 < \cdots < s_k$  with the set  $\{s_1, s_2, \dots, s_k\}$ , and writing  $S + a = S \cup \{a\}$ ,  $S - b = S \setminus \{b\}$  for brevity, we can describe row insertion by the following formula:

$$\text{RI}(S, a) = \begin{cases} S + a & \text{if } S = \emptyset \text{ or } \max(S) < a; \\ (b, S + a - b) & \text{if } b \in S \text{ is minimal such that } a < b. \end{cases}$$

When the second case occurs, we say that  $a$  *bumps*  $b$  out of row  $S$  to produce the new row  $S'$ .

## 2.2. Tableau insertion

The next component of the RSK algorithm is the *tableau insertion* subroutine. The input here is a partial standard tableau  $P$  and an integer  $a$  not appearing in  $P$ . The output is a partial standard tableau  $P'$  containing  $a$  and the entries in  $P$  (possibly rearranged), such that the shape of  $P'$  is obtained from the shape of  $P$  by adding one new corner box. To describe this algorithm, let  $S_j$  be the increasing sequence of entries in row  $j$  of  $P$ , where row 1 is the lowest (longest) row of  $P$ . Apply the row insertion subroutine to insert  $a$  into  $S_1$ . Replace the first row of  $P$  with the sequence  $S'_1$  produced by the row insertion. If no value of  $b$  is produced, the tableau insertion is complete. Otherwise, apply row insertion again to insert  $b$  into  $S_2$ , producing a new second row  $S'_2$  and possibly a value  $c$ . If no value of  $c$  is produced, stop here. Otherwise, use row insertion again to insert  $c$  into  $S_3$ , and so on. Identifying  $P$  with its reading word  $S_\ell \cdots S_3 S_2 S_1$ , we can describe tableau insertion via this recursive formula:

$$\text{TI}(S_\ell \cdots S_2 S_1, a) = \begin{cases} S_\ell \cdots S_2 S'_1 & \text{if } \text{RI}(S_1, a) = S'_1; \\ \text{TI}(S_\ell \cdots S_2, b) S'_1 & \text{if } \text{RI}(S_1, a) = (b, S'_1). \end{cases}$$

Letting  $\ell = 1$  and  $S_1 = \emptyset$ , we see that when  $P$  is empty,  $\text{TI}(P, a) = a$ . One can check that the output of tableau insertion always is a partial standard tableau whose shape is a partition obtained from the shape of  $P$  by adding one new box (see [8, Chap. 1]; we prove a more general result in Lemma 12 below).

## 2.3. RSK algorithm for permutations

Now we are ready to describe the *RSK algorithm for permutations*. Let  $w = w_1 w_2 \cdots w_n$  be a permutation in  $S_n$  written in one-line form, so  $w_1, w_2, \dots, w_n$  is a rearrangement of  $1, 2, \dots, n$ . We build a sequence of partial standard tableaux  $P^{(0)}, P^{(1)}, \dots, P^{(n)}$  and  $Q^{(0)}, Q^{(1)}, \dots, Q^{(n)}$  by processing the letters of  $w$  from left to right. To start, let  $P^{(0)}$  and  $Q^{(0)}$  be empty tableaux. Once we have built  $P^{(i-1)}$  and  $Q^{(i-1)}$  (where  $1 \leq i \leq n$ ), we obtain  $P^{(i)}$  by inserting  $w_i$  into the tableau  $P^{(i-1)}$ . The shape of  $P^{(i)}$  has a new corner box not present in  $P^{(i-1)}$ . We obtain  $Q^{(i)}$  from  $Q^{(i-1)}$  by placing the value  $i$  in this same location (so  $P^{(i)}$  and  $Q^{(i)}$  have the same shape for all  $i$ ). The final output is the pair  $\text{RSK}(w) = (P^{(n)}, Q^{(n)})$ . We write  $P(w) = P^{(n)}$  and call this the *insertion tableau* of  $w$ ; we write  $Q(w) = Q^{(n)}$  and call this the *recording tableau* of  $w$ . We can describe the insertion function  $P$  recursively as follows: for an empty word  $w$ ,  $P(w)$  and  $Q(w)$  are both empty; while for  $n > 0$ ,

$P(w_1 \cdots w_n) = \text{TI}(P(w_1 \cdots w_{n-1}), w_n)$ , creating a new box at the end of some row  $j$ ;  
 $Q(w_1 \cdots w_n) = Q(w_1 \cdots w_{n-1})$  with  $n$  placed in a new box at the end  
of the same row  $j$ .

Note that RSK has now been defined for all partial permutations  $v$ ; the insertion tableau  $P(v)$  is a partial standard tableau in which all symbols in  $v$  appear, whereas  $Q(v)$  is a standard tableau of the same partition shape as  $P(v)$ .

#### 2.4. Inversion of the RSK algorithm

It is known that RSK is a bijection from  $S_n$  to the set of pairs of  $n$ -celled standard tableaux of the same shape. One can prove this by explicitly describing the inverse algorithm. We invert the row insertion subroutine as follows. If  $\text{RI}(S, a) = S'$  with no new value  $b$  produced, then  $a$  is the largest value in  $S'$ , and  $S$  is  $S'$  with  $a$  deleted. Otherwise, if  $\text{RI}(S, a) = (b, S')$ , then we obtain  $(S, a) = \text{RI}^{-1}(b, S')$  by letting  $a$  be the largest value in  $S'$  less than  $b$  and letting  $S = S' + b - a$ . Writing this as a formula:

$$\text{RI}^{-1}(b, S') = \begin{cases} (S' - \max(S'), \max(S')) & \text{if } b \text{ does not exist;} \\ (S' + b - a, a) & \text{if } a \in S \text{ is maximal such that } a < b. \end{cases}$$

The tableau insertion subroutine  $\text{TI}$ , as defined above, cannot be inverted without additional information. We need to know how many rows in the tableau  $S_\ell \cdots S_2 S_1$  (viewed as a reading word) were affected by the initial insertion of value  $a$ . Suppose we are told that

$$\text{TI}(S_\ell \cdots S_2 S_1, a) = S_\ell \cdots S_{j+1} z S'_j \cdots S'_1$$

where  $0 \leq j \leq \ell$  and the output word has increasing runs  $S_\ell, \dots, S_{j+1} z, S'_j, \dots, S'_1$ . (If  $j = \ell$ ,  $S_{j+1}$  is empty.) We can then apply  $\text{RI}^{-1}$  to  $z$  and  $S'_j$  to recover  $S_j$  and a value  $y$ , then apply  $\text{RI}^{-1}$  to  $y$  and  $S'_{j-1}$  to recover  $S_{j-1}$  and a value  $x$ , and so on. Stating this recursively, we have:

$$\begin{aligned} \text{TI}^{-1}(j, S_\ell \cdots S_{j+1} z S'_j \cdots S'_1) = \\ \begin{cases} (S_\ell \cdots S_2 S_1, z) & \text{if } j = 0; \\ \text{TI}^{-1}(j-1, S_\ell \cdots S_{j+1} S_j y S'_{j-1} \cdots S'_1) & \text{if } j > 0 \text{ and } (S_j, y) = \text{RI}^{-1}(z, S'_j). \end{cases} \end{aligned} \quad (8)$$

Finally, we invert the full RSK algorithm (acting on a permutation or partial permutation) as follows. Given  $(P, Q) = \text{RSK}(w_1 \cdots w_n)$ , we use  $P$  and  $Q$  to recover the symbols

$w_n, w_{n-1}, \dots, w_1$  in this order. To get  $w_n$ , suppose  $n$  appears in  $Q$  at the end of row  $j+1$ , where  $0 \leq j < n$ . Compute  $\text{TI}^{-1}(j, P) = (P^{(n-1)}, w_n)$ , and let  $Q^{(n-1)}$  be  $Q$  with  $n$  erased. Then use the position of  $n-1$  in  $Q^{(n-1)}$  to find  $w_{n-1}$  and  $P^{(n-2)}$ , and let  $Q^{(n-2)}$  be  $Q^{(n-1)}$  with  $n-1$  erased. We proceed similarly to recover the remaining letters in  $w$ .

### 3. New RSK algorithms

Fix a positive integer  $p$ . This section defines variants of the Robinson–Schensted algorithm using subroutines denoted  $\text{RI}_p$ ,  $\text{TI}_p$ ,  $\text{RSK}_p$ ,  $P_p$ , and  $Q_p$ . The key changes occur in the row insertion subroutine  $\text{RI}_p$ .

#### 3.1. $p$ -row insertion

As in §2.1, the input to the new row insertion subroutine is a strictly increasing sequence (or set)  $S = \{s_1 < s_2 < \dots < s_k\}$  and an integer  $a \notin S$ . The output is a new increasing sequence  $S'$  and an optional value  $b$  satisfying  $S + a = S' + b$  (or  $S + a = S'$  if there is no  $b$ ). If  $S$  is empty or  $s_k < a$ , we define  $S' = \{s_1 < \dots < s_k < a\}$ , and no  $b$  is produced. Otherwise, we compute  $b \in S + a$  using the rules described below, and let  $S' = S + a - b$ . To state the rules for finding  $b$ , it is convenient to let  $[u, v]$  denote the interval of integers  $\{u, u+1, u+2, \dots, v\}$  for any integers  $u \leq v$ .

- **Special Rule 1.** Suppose  $p \leq a$  and  $a+1 \in S$  and  $a+2 \in S$ . Then  $b = a+2$ .
- **Special Rule 2.** Suppose  $p < a$ ,  $a+1 \in S$ ,  $a+2 \notin S$ , and  $a-1 \in S$ . Let  $t \geq 1$  be maximal such that  $[a-t, a-1] \subseteq S$ . Then  $b = \max(a-t+1, p+1)$ .
- **Default Rule.** If special rules 1 and 2 do not apply, let  $b$  be the minimum element in  $S$  such that  $b > a$  (as in the classical case).

Stated as a formula, the new row insertion rule is:

$$\text{RI}_p(S, a) = \begin{cases} S + a & \text{if } S = \emptyset \text{ or } \max(S) < a; \\ (b, S + a - b) & \text{otherwise, using the rules above to find } b. \end{cases}$$

One novel feature of the new rule is that  $b = a$  can occur, in which case  $S + a - b = S$  and  $a$  “passes through” the row without changing it, “bumping itself” into the next row.

**Example 9.** Taking  $a = 8$ , we compute  $\text{RI}_p(S, a)$  for various choices of  $p$  and  $S$ :

$$\begin{aligned}
\text{RI}_3(\boxed{1\,2\,4\,5\,6\,7}, 8) &= \boxed{1\,2\,4\,5\,6\,7\,8} && \text{by the base case;} \\
\text{RI}_1(\boxed{1\,2\,4\,5\,6\,7\,9\,10\,12}, 8) &= (10, \boxed{1\,2\,4\,5\,6\,7\,8\,9\,12}) && \text{by special rule 1;} \\
\text{RI}_9(\boxed{1\,2\,4\,5\,6\,7\,9\,10\,12}, 8) &= (9, \boxed{1\,2\,4\,5\,6\,7\,8\,10\,12}) && \text{by the default rule;} \\
\text{RI}_1(\boxed{1\,2\,4\,5\,6\,7\,9\,11}, 8) &= (5, \boxed{1\,2\,4\,6\,7\,8\,9\,11}) && \text{by special rule 2;} \\
\text{RI}_6(\boxed{1\,2\,4\,5\,6\,7\,9\,11}, 8) &= (7, \boxed{1\,2\,4\,5\,6\,8\,9\,11}) && \text{by special rule 2;} \\
\text{RI}_1(\boxed{1\,2\,3\,4\,6\,7\,9\,11}, 8) &= (7, \boxed{1\,2\,3\,4\,6\,8\,9\,11}) && \text{by special rule 2;} \\
\text{RI}_1(\boxed{1\,2\,3\,4\,5\,7\,9\,11}, 8) &= (8, \boxed{1\,2\,3\,4\,5\,7\,9\,11}) && \text{by special rule 2;} \\
\text{RI}_3(\boxed{1\,2\,3\,4\,5\,6\,9\,11}, 8) &= (9, \boxed{1\,2\,3\,4\,5\,6\,8\,11}) && \text{by the default rule.}
\end{aligned}$$

Note how 8 bumps itself in the next-to-last example.

A key fact about  $p$ -row insertion is that it can be inverted. For an optional value  $b$  and increasing sequence (or set)  $S'$  with  $b \notin S'$ , define

$$\text{RI}'_p(b, S') = \begin{cases} (S' - \max(S'), \max(S')) & \text{if } b \text{ does not exist;} \\ (S' + b - a, a) & \text{otherwise, using the rules below to find } a. \end{cases}$$

- **Special inverse rule 1.** Suppose  $p \leq b - 2$  and  $b - 1 \in S'$  and  $b - 2 \in S'$ . Then  $a = b - 2$ .
- **Special inverse rule 2.** Suppose  $b - 1 \in S'$ ,  $b + 1 \in S'$ , and either  $p = b - 1$  or ( $p \leq b - 2$  and  $b - 2 \notin S'$ ). Let  $u \geq 1$  be maximal with  $[b + 1, b + u] \subseteq S'$ . Then  $a = b + u - 1$ .
- **Default inverse rule.** If special inverse rules 1 and 2 do not apply, let  $a$  be the maximum element in  $S'$  such that  $a < b$  (as in the classical case).

One can check that these rules correctly invert the  $p$ -row insertions in the example above. More generally:

**Lemma 10.** For all  $p \geq 1$ ,  $\text{RI}'_p$  is the two-sided inverse of  $\text{RI}_p$ .

The proof of this lemma is elementary but somewhat tedious, so we defer it until §3.4.

### 3.2. $p$ -tableau insertion

We define  $p$ -tableau insertion exactly as in the classical case, replacing row insertion with  $p$ -row insertion in the definition. Specifically, if  $P$  is a partial standard tableau with reading word factored into ascending runs  $S_\ell \cdots S_2 S_1$ , and if  $a$  is a symbol not appearing in  $P$ , then

$$\mathrm{TI}_p(S_\ell \cdots S_2 S_1, a) = \begin{cases} S_\ell \cdots S_2 S'_1 & \text{if } \mathrm{RI}_p(S_1, a) = S'_1; \\ \mathrm{TI}_p(S_\ell \cdots S_2, b) S'_1 & \text{if } \mathrm{RI}_p(S_1, a) = (b, S'_1). \end{cases}$$

Informally, we insert  $a$  into row 1 of  $P$  and either terminate or bump out a value  $b$  (possibly  $b = a$ , in which case we say  $a$  “passes through” row 1). If  $b$  exists, we continue recursively by inserting  $b$  into the partial tableau consisting of rows 2 through  $\ell$  of  $P$ .

**Example 11.** We claim that

$$\mathrm{TI}_3 \left( \begin{array}{|c|c|c|c|} \hline 3 & & & \\ \hline 2 & 5 & 9 & \\ \hline 1 & 4 & 7 & 8 \\ \hline \end{array}, 6 \right) = \begin{array}{|c|c|c|c|} \hline 3 & 9 & & \\ \hline 2 & 5 & 8 & \\ \hline 1 & 4 & 6 & 7 \\ \hline \end{array}.$$

In detail, the 3-insertion of 6 into row 1 bumps  $b = 8$  into row 2 by special rule 1. Inserting 8 into row 2 bumps the 9 into row 3 by the default rule. Inserting 9 at the end of row 3 terminates the algorithm. Some additional examples of  $p$ -tableau insertion follow.

$$\begin{aligned} \mathrm{TI}_1 \left( \begin{array}{|c|c|c|c|} \hline 8 & & & \\ \hline 2 & 6 & 7 & \\ \hline 1 & 4 & 5 & \\ \hline \end{array}, 3 \right) &= \begin{array}{|c|c|c|c|} \hline 8 & & & \\ \hline 7 & & & \\ \hline 2 & 5 & 6 & \\ \hline 1 & 3 & 4 & \\ \hline \end{array} & \mathrm{TI}_1 \left( \begin{array}{|c|c|c|c|} \hline 4 & 5 & 7 & \\ \hline 1 & 2 & 6 & 8 \\ \hline \end{array}, 3 \right) &= \begin{array}{|c|c|c|c|} \hline 5 & & & \\ \hline 4 & 6 & 7 & \\ \hline 1 & 2 & 3 & 8 \\ \hline \end{array} \\ \mathrm{TI}_1 \left( \begin{array}{|c|c|c|c|} \hline 6 & & & \\ \hline 1 & 2 & 3 & 5 \\ \hline \end{array}, 4 \right) &= \begin{array}{|c|c|c|c|c|} \hline 6 & & & & \\ \hline 2 & & & & \\ \hline 1 & 3 & 4 & 5 & \\ \hline \end{array} & \mathrm{TI}_1 \left( \begin{array}{|c|c|c|} \hline 2 & 6 & 8 \\ \hline 1 & 3 & 5 \\ \hline \end{array}, 4 \right) &= \begin{array}{|c|c|c|c|} \hline 6 & & & \\ \hline 2 & 4 & 8 & \\ \hline 1 & 3 & 5 & \\ \hline \end{array} \\ \mathrm{TI}_1 \left( \begin{array}{|c|c|c|c|} \hline 3 & & & \\ \hline 1 & 5 & 6 & \\ \hline \end{array}, 4 \right) &= \begin{array}{|c|c|c|c|} \hline 3 & 6 & & \\ \hline 1 & 4 & 5 & \\ \hline \end{array} & \mathrm{TI}_1 \left( \begin{array}{|c|c|c|c|} \hline 2 & 4 & & \\ \hline 1 & 3 & 5 & 7 \\ \hline \end{array}, 6 \right) &= \begin{array}{|c|c|c|c|c|} \hline 2 & 4 & 6 & & \\ \hline 1 & 3 & 5 & 7 & \\ \hline \end{array} \end{aligned}$$

In the preceding examples, the output of  $p$ -tableau insertion was always another partial standard tableau. This illustrates the general fact in the next lemma.

**Lemma 12.** For all  $p \geq 1$ , all partial standard tableaux  $P$ , and any symbol  $a$  not appearing in  $P$ ,  $P' = \mathrm{TI}_p(P, a)$  is a partial standard tableau of partition shape.

The proof is given in §3.5.

### 3.3. $\mathrm{RSK}_p$ for permutations

The variant Robinson–Schensted algorithm

$$\mathrm{RSK}_p : S_n \rightarrow \bigcup_{\lambda \in \mathrm{Par}(n)} \mathrm{SYT}(\lambda) \times \mathrm{SYT}(\lambda)$$

is defined exactly as in the classical case, with  $\mathrm{TI}_p$  replacing  $\mathrm{TI}$  as the tableau insertion subroutine. Specifically, for  $w \in S_n$ , let  $\mathrm{RSK}_p(w) = (P_p(w), Q_p(w))$ , where

$P_p(w_1 \cdots w_n) = \text{TI}_p(P_p(w_1 \cdots w_{n-1}), w_n)$ , creating a new box at the end  
of some row  $j$ ;  
 $Q_p(w_1 \cdots w_n) = Q_p(w_1 \cdots w_{n-1})$  with  $n$  placed in a new box at the end  
of the same row  $j$ .

By repeated application of [Lemma 12](#), we see that  $P_p(w) \in \text{SYT}(\lambda)$  for some  $\lambda \in \text{Par}(n)$ . By induction on  $n$ , one sees that  $Q_p(w) \in \text{SYT}(\lambda)$  as well. Thus,  $\text{RSK}_p$  does map  $S_n$  into the stated codomain. Similarly,  $\text{RSK}_p$  maps a partial permutation  $v$  to a pair  $(P_p(v), Q_p(v))$  of tableaux of the same shape, where  $P_p(v)$  is a partial standard tableau using the symbols in  $v$ , and  $Q_p(v)$  is a standard tableau.

**Example 13.** Let  $n = 6$  and  $w = 516243$ . We compute

$$\begin{aligned} \text{RSK}_1(w) &= \left( \begin{array}{|c|c|c|} \hline 5 & & \\ \hline 2 & 6 & \\ \hline 1 & 3 & 4 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array} \right), \\ \text{RSK}_2(w) &= \left( \begin{array}{|c|c|c|} \hline 5 & & \\ \hline 3 & 6 & \\ \hline 1 & 2 & 4 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array} \right), \\ \text{RSK}_3(w) &= \left( \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 4 & 5 & \\ \hline 1 & 2 & 3 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array} \right) = \text{RSK}_4(w), \\ \text{RSK}_5(w) &= \left( \begin{array}{|c|c|c|} \hline 5 & & \\ \hline 4 & 6 & \\ \hline 1 & 2 & 3 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 2 & 4 & \\ \hline 1 & 3 & 5 \\ \hline \end{array} \right) = \text{RSK}_6(w) = \text{RSK}(w). \end{aligned}$$

For  $w = 123465$ , we compute

$$\begin{aligned} \text{RSK}_1(w) &= \left( \begin{array}{|c|c|c|c|c|c|} \hline 2 & & & & & \\ \hline 1 & 3 & 4 & 5 & 6 & \\ \hline \end{array}, \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array} \right), \\ \text{RSK}_2(w) &= \left( \begin{array}{|c|c|c|c|c|c|} \hline 3 & & & & & \\ \hline 1 & 2 & 4 & 5 & 6 & \\ \hline \end{array}, \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array} \right), \\ \text{RSK}_3(w) &= \left( \begin{array}{|c|c|c|c|c|c|} \hline 4 & & & & & \\ \hline 1 & 2 & 3 & 5 & 6 & \\ \hline \end{array}, \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array} \right), \\ \text{RSK}_4(w) &= \left( \begin{array}{|c|c|c|c|c|c|} \hline 5 & & & & & \\ \hline 1 & 2 & 3 & 4 & 6 & \\ \hline \end{array}, \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array} \right), \\ \text{RSK}_5(w) &= \left( \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array}, \begin{array}{|c|c|c|c|c|c|} \hline 6 & & & & & \\ \hline 1 & 2 & 3 & 4 & 5 & \\ \hline \end{array} \right) = \text{RSK}_6(w) = \text{RSK}(w). \end{aligned}$$

In these examples,  $Q_p(w)$  is independent of  $p$ , but  $P_p(w)$  varies with  $p$ . We will prove that the  $p$ -recording tableau  $Q_p(w)$  always equals the classical recording tableau  $Q(w)$  in [Theorem 24](#).



**Theorem 14.** For all  $n, p \geq 1$ ,  $\text{RSK}_p$  is a bijection from  $S_n$  to the set of pairs of  $n$ -celled standard tableaux of the same partition shape.

**Proof.** Given a pair  $(P, Q)$  known to be of the form  $(P_p(w), Q_p(w))$  for some  $w \in S_n$ , we can recover  $w = w_1 w_2 \cdots w_n$  uniquely from  $P$  and  $Q$  by the same process used to invert the classical RSK algorithm. Suppose  $n$  appears in  $Q$  at the end of row  $j + 1$ . We can then compute  $\text{TI}_p^{-1}(j, P) = (P^{(n-1)}, w_n)$  using formula (8) with  $\text{TI}^{-1}$  and  $\text{RI}^{-1}$  replaced by  $\text{TI}_p^{-1}$  and  $\text{RI}_p^{-1}$  (recall from Lemma 10 that  $\text{RI}_p$  is invertible with inverse  $\text{RI}'_p$ ). Let  $Q^{(n-1)}$  be  $Q$  with  $n$  erased. We continue recursively, processing  $P^{(n-1)}$  and  $Q^{(n-1)}$  in the same way to recover the symbols  $w_{n-1}, \dots, w_2, w_1$  in this order.

This argument shows  $\text{RSK}_p$  is an injective map into the stated codomain. Since this codomain is known to have the same size as  $S_n$  (because the classical RSK algorithm is bijective), we deduce that each  $\text{RSK}_p$  is a bijection. To check this without invoking bijectivity of classical RSK, one needs to prove that the output of  $\text{TI}_p^{-1}(j, P)$  always is another partial standard tableau  $P^{(n-1)}$ , requiring a tedious argument analogous to the proof of Lemma 12. We omit this.  $\square$

**Example 15.** Given

$$(P, Q) = \left( \begin{array}{|c|c|} \hline 6 & \\ \hline 4 & \\ \hline 3 & 5 \\ \hline 1 & 2 \\ \hline \end{array}, \begin{array}{|c|c|} \hline 6 & \\ \hline 4 & \\ \hline 3 & 5 \\ \hline 1 & 2 \\ \hline \end{array} \right),$$

we compute  $\text{RSK}_p^{-1}(P, Q) = 365142$  for  $p = 1, 2, 3$ . On the other hand,  $\text{RSK}_4^{-1}(P, Q) = 564132$ , whereas  $\text{RSK}_5^{-1}(P, Q) = \text{RSK}_6^{-1}(P, Q) = \text{RSK}^{-1}(P, Q) = 465132$ . Given

$$(P, Q) = \left( \begin{array}{|c|c|c|} \hline 3 & & \\ \hline 2 & 6 & \\ \hline 1 & 4 & 5 \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 6 & & \\ \hline 4 & 5 & \\ \hline 1 & 2 & 3 \\ \hline \end{array} \right),$$

we compute  $\text{RSK}_p^{-1}(P, Q) = 356241$  for  $1 \leq p \leq 4$ , and  $\text{RSK}_p^{-1}(P, Q) = 346251$  for  $p = 5, 6$ .

The preceding examples suggest that when acting on  $S_n$ ,  $\text{RSK}_p$  agrees with the classical RSK algorithm if  $p \geq n - 1$ . We now prove this.

**Theorem 16.** Suppose  $w$  is a permutation or partial permutation using symbols in  $\{1, 2, \dots, n\}$ . For all  $p \geq n - 1$ ,  $\text{RSK}_p(w) = \text{RSK}(w)$ .

**Proof.** It suffices to show that whenever we calculate  $\text{RI}_p(S, a)$  in the process of finding  $\text{RSK}_p(w)$ , special rules 1 and 2 can never apply. Note that  $a$  and the symbols in  $S$  are all symbols in  $w$ . To use special rule 1, we need (among other things)  $p \leq a$  and  $a + 2 \in S$ , but then  $a + 2 \geq p + 2 \geq n + 1$ , so  $a + 2$  does not appear in  $w$ . To use special rule 2, we

need (among other things)  $p < a$  and  $a + 1 \in S$ , but then  $a + 1 > p + 1 \geq n$ , so  $a + 1$  does not appear in  $w$ .  $\square$

### 3.4. Proof of Lemma 10

Fix  $p \geq 1$ ; we must verify that the maps  $\text{RI}_p$  and  $\text{RI}'_p$  defined in §3.1 are two-sided inverses. Given an input  $(S, a)$ , we prove that  $\text{RI}'_p(\text{RI}_p(S, a)) = (S, a)$  by showing that if a given rule is used to compute  $(b, S') = \text{RI}_p(S, a)$ , then the corresponding inverse rule will apply and give the answer  $(S, a) = \text{RI}'_p(b, S')$ . This result is evident in the terminating case, where  $b$  does not exist.

Case 1: Assume special rule 1 applies to  $(S, a)$ . This means that  $p \leq a$ ,  $a + 1 \in S$ ,  $a + 2 \in S$ ,  $b = a + 2$ , and the set  $S'$  is obtained from  $S$  by removing  $a + 2$  and adding  $a$ . Since  $a = b - 2$ , our assumptions become  $p \leq b - 2$ ,  $b - 2 \in S'$ , and  $b - 1 \in S'$ . Therefore special inverse rule 1 is used to find  $\text{RI}'_p(b, S')$ . Since  $b - 2 = a$  here, we see that  $\text{RI}'_p(b, S') = (S, a)$  in this case.

Case 2: Assume special rule 2 applies to  $(S, a)$ . This means that  $p < a$ ,  $a + 1 \in S$ ,  $a + 2 \notin S$ , and  $a - 1 \in S$ . Let  $t \geq 1$  satisfy  $[a - t, a - 1] \subseteq S$  and  $a - t - 1 \notin S$ . Consider two subcases depending on whether  $p < a - t$  or  $p \geq a - t$ .

*Subcase 1:* If  $p < a - t$ , then  $b = a - t + 1 \leq a$  and  $S' = S + a - b$ . Since  $p < a - t = b - 1$ ,  $p \leq b - 2$  follows. Since  $b \neq a - t \in S$  and  $a \neq a - t - 1 \notin S$ , it follows that  $b - 1 \in S'$  and  $b - 2 \notin S'$ . We show  $b + 1 \in S'$  by considering three possibilities.

- If  $t = 1$ , then  $b = a$  and so  $b + 1 = a + 1 \in S'$ .
- If  $t = 2$ , then  $b = a - 1 \neq a$  and so  $b + 1 = a \in S'$ .
- If  $t > 2$ , then  $b + 1 = a - t + 2 \in [a - t, a - 1] \subseteq S$ , so  $b + 1 \in S'$ .

Therefore special inverse rule 2 is used to compute  $(S^*, a') = \text{RI}'_p(b, S')$ . To show  $(S^*, a') = (S, a)$ , it suffices to show  $a' = a$ . To find  $a'$ , let  $u \geq 1$  satisfy  $[b + 1, b + u] \subseteq S'$  and  $b + u + 1 \notin S'$ . We know  $a' = b + u - 1$  and  $a = b + t - 1$ , so it suffices to show that  $u = t$ . Again consider three possibilities.

- If  $t = 1$ , then  $u = 1$  because  $b = a$ ,  $b + 1 \in S'$  and  $b + 2 = a + 2 \notin S'$ .
- If  $t = 2$ , then  $u = 2$  because  $b = a - 1$ ,  $a \in S'$ ,  $a + 1 \in S'$ , and  $a + 2 \notin S'$ .
- If  $t > 2$ , then  $u = t$  because  $[b + 1, b + t] = [a - t + 2, a + 1] \subseteq S'$  and  $b + t + 1 = a + 2 \notin S'$ .

*Subcase 2:* If  $p \geq a - t$ , then  $b = p + 1 \leq a$  and  $S' = S + a - b$ . Since  $b = p + 1$ ,  $p = b - 1$  follows. Since  $a - t + 1 \leq b \leq a$  and  $[a - t, a - 1] \subseteq S$ ,  $b - 1 \in S'$  follows. We show  $b + 1 \in S'$  by considering three possibilities.

- If  $b = a$ , then  $b + 1 = a + 1 \in S'$ .
- If  $b = a - 1$ , then  $b + 1 = a \neq b$ , so  $b + 1 \in S + a - b = S'$ .
- If  $b \leq a - 2$ , then  $b + 1 \in [a - t + 2, a - 1] \subseteq S$  and  $b + 1 \neq b$ , so  $b + 1 \in S + a - b = S'$ .

Therefore special inverse rule 2 is used to find  $(S^*, a') = \text{RI}'_p(b, S')$ . Let  $u \geq 1$  satisfy  $[b+1, b+u] \subseteq S'$  and  $b+u+1 \notin S'$ . We show  $a' = a$  (hence  $S^* = S$ ) by considering three possibilities.

- If  $b = a$ , then  $u = 1$  because  $b+1 = a+1 \in S'$  and  $b+2 = a+2 \notin S'$ . So  $a' = b+u-1 = b = a$ .
- If  $b = a-1$ , then  $u = 2$  because  $b+1 = a \in S'$ ,  $b+2 = a+1 \in S'$ , and  $b+3 = a+2 \notin S'$ . So  $a' = b+u-1 = b+1 = a$ .
- If  $b \leq a-2$ , then  $[b+1, a+1] \subseteq S'$  since  $a-t < b$ ,  $[a-t, a-1] \subseteq S$ ,  $a \in S'$ , and  $a+1 \in S$ . But  $a+2 \notin S'$ . Thus  $b+u = a+1$ , and  $a' = b+u-1 = a$ .

**Case 3:** Assume the default rule applies to  $(S, a)$ . In this case,  $b$  is the least member of  $S$  with  $b > a$ , and  $S' = S + a - b$ . Moreover,  $a$  is the greatest member of  $S'$  with  $a < b$ , so it suffices to show that the default inverse rule will apply when computing  $\text{RI}'_p(b, S')$ .

*Subcase 1:* Assume  $b \geq a+2$ . Since  $a < b-1 < b$ , it follows that  $b-1 \notin S'$ . So the default inverse rule applies here.

*Subcase 2:* Assume  $b = a+1$  and  $a+2 \in S$ . Because we assumed special rule 1 did not apply, we must have  $p > a$ . Then  $p > b-1 > b-2$ , so the default inverse rule applies here.

*Subcase 3:* Assume  $b = a+1$  and  $a+2 \notin S$  and  $a-1 \notin S$ . Then  $b-2 \notin S'$ , so special inverse rule 1 does not apply. Also  $b+1 \notin S'$ , so special inverse rule 2 does not apply. So the default inverse rule applies here.

*Subcase 4:* Assume  $b = a+1$  and  $a+2 \notin S$  and  $a-1 \in S$ . Because we assumed special rule 2 did not apply, we must have  $p \geq a$ . Then  $p \geq b-1 > b-2$ , so special inverse rule 1 does not apply. Also  $b+1 \notin S'$ , so special inverse rule 2 does not apply. So the default inverse rule applies here.

We have now finished proving that  $\text{RI}'_p \circ \text{RI}_p$  is an identity map. One can similarly verify that  $\text{RI}_p \circ \text{RI}'_p$  is an identity map. Alternatively, this automatically follows from the first verification, since (for any finite alphabet of symbols) the domain and codomain of the injective map  $\text{RI}_p$  are finite sets of the same size. This concludes the proof of [Lemma 10](#).

### 3.5. Proof of [Lemma 12](#)

Fix  $p \geq 1$ , a partial standard tableau  $P$ , and a symbol  $a$  not appearing in  $P$ . We must prove that  $P' = \text{TI}_p(P, a)$  is a partial standard tableau of partition shape.

The proof will use the following observation about  $\text{RI}_p$ . Whenever we apply  $p$ -row insertion to  $(S, a)$  to obtain a new row  $S'$  and possibly a new value  $b$ , each entry in row  $S$  either stays the same, decreases, or increments by 1 to become the corresponding entry in  $S'$ . This follows immediately from the definition of  $\text{RI}_p$ . In particular, certain entries in  $S$  will be incremented iff special rule 2 is used with  $b < a$ , in which case each entry in  $S$  in the range  $[b, a-1]$  increases by 1 when we pass to  $S'$ . (See [Example 9](#) in §3.1.)

Let the reading word of  $P$  (factored into ascending runs) be  $S_\ell \cdots S_2 S_1$ , so the bottom two rows of  $P$  are  $S_1$  and  $S_2$  (which may be empty). If  $\max(S_1) < a$ , we obtain  $P'$  by adding  $a$  to the end of row 1, and the result of the lemma is evident. From now on, suppose  $a < \max(S_1)$  and  $(b, S'_1) = \text{RI}_p(S_1, a)$ . By induction on the number of rows of  $P$ , we can assume that  $P^* = \text{TI}_p(S_\ell \cdots S_2, b)$  is a partial standard tableau of partition shape. By definition of  $\text{RI}_p$ , the new bottom row  $S'_1$  is strictly increasing. To complete the proof of the lemma, we must show that each entry in  $S'_1$  is less than the entry directly above it (if any) in  $P^*$ , and that the lowest row of  $P^*$  is not longer than  $S'_1$ . Denote the output of  $\text{RI}_p(S_2, b)$  by  $S'_2$  (if  $\max(S_2) < b$ ) or  $(c, S'_2)$  (if  $b < \max(S_2)$ ). For any increasing row  $S$ , write  $S(k)$  for the  $k$ th entry in row  $S$ .

Let us say that  $P'$  has a *column violation* iff for some index  $k \leq |S'_2|$ ,  $S'_1(k) \geq S'_2(k)$ . Note that the entries of  $P'$  are distinct (being the entries of the partial standard tableau  $P$  together with  $a$ , which does not occur in  $P$ ). So any column violation must actually be a strict violation  $S'_1(k) > S'_2(k)$ . We know the original partial standard tableau  $P$  has no column violations. To show that  $P'$  has no column violations, consider cases based on what happens to row 2 when we compute  $\text{RI}_p(S_2, b)$ . (Compare this discussion to the examples in §3.2.)

- (1) Suppose special rule 1 applies to  $(S_2, b)$ . Then we obtain  $S'_2$  by decrementing two entries of  $S_2$  (namely  $b+1$  and  $b+2$ ) by one. On the other hand, we got  $S'_1$  from  $S_1$  either by decreasing certain entries, doing nothing, or incrementing certain entries by one. Since a tie  $S'_1(k) = S'_2(k)$  is impossible, we see that the only way a column violation might occur here is if special rule 2 applied when computing  $(b, S'_1) = \text{RI}_p(S_1, a)$  and  $b < a$ . However, in this event,  $b+1 \in S_1 \cup \{a\}$ , so  $b+1$  cannot appear in  $S_2$ , so special rule 1 could not have applied in row 2 after all. Thus no column violation occurs in this case.
- (2) Suppose special rule 2 applies to  $(S_2, b)$ . Then we obtain  $S'_2$  by incrementing zero or more entries of  $S_2$  by one. Since entries in row 1 either stay the same, decrease, or increment by one, and since ties are impossible, we see that no column violation can occur in this case.
- (3) Suppose the default rule applies to  $(S_2, b)$ , replacing the entry  $c$  in column  $k$  of  $S_2$  by  $b < c$ . Since entries in row 1 increase by at most one and ties are impossible, the only possible column violation that might occur is in column  $k$ .

*Subcase 1:* Suppose  $b$  appears somewhere in the original row  $S_1$ , say in column  $j$ , so  $S_1(j) = b$ . We consider the three possibilities  $j < k$ ,  $j > k$ , and  $j = k$ .

- Suppose  $j < k$ . Let  $d = S_2(j)$ , so the first two rows of  $P$  look like this:

		$d$			$c$														
		$b$																	

Since  $P$  is a partial standard tableau,  $b < d < c$ . But then the default rule would cause  $b$  to bump  $d$  or some earlier element in  $S_2$ , not  $c$ . So this possibility cannot occur, and we must have  $k \leq j$ .

- Suppose  $j > k$ . Let  $e = S'_1(k)$  and  $f = S'_1(j)$ , so the first two rows of  $P'$  look like this:

		$b$							
		$e$					$f$		

Now  $e < f$  since  $S'_1$  is an increasing row, and  $f = S'_1(j) \leq S_1(j) + 1 = b + 1$  by the initial observation about  $\text{RI}_p$ . So  $e < b + 1$ , hence  $e \leq b$ , hence  $e < b$  since ties are impossible. Thus there is no column violation in column  $k$ .

- Suppose  $k = j$ . Let  $e = S'_1(k) = S'_1(j)$ . As before,  $e = S'_1(j) \leq S_1(j) + 1 = b + 1$ , and we want to show  $e < b$ . Since ties are impossible, we need only rule out the case  $e = b + 1$ . This could only happen if special rule 2 had been used in row 1 to bump  $b < a$  into row 2. By definition of special rule 2,  $j - 1 > 0$  and  $S_1(j - 1) = b - 1$ . Let  $d = S_2(j - 1)$ , so the first two rows of  $P$  look like this (with  $b' = b - 1$ ):

		$d$	$c$						
		$b'$	$b$						

We know  $b - 1 = b' < d$ , and  $d$  cannot equal  $b$  since otherwise  $b$  would appear twice in the partial standard tableau  $P$ . So in fact  $b < d$ . But then the default rule would cause  $b$  to bump  $d$  or some earlier entry in row 2, not  $c$ . Thus  $e = b + 1$  is impossible, so  $e < b$  and there is no column violation in column  $k$ .

*Subcase 2:* Suppose  $b = a$ , so  $S'_1 = S_1$ . In this case,  $a - 1$  must appear somewhere in  $S_1$ , say in column  $j$ . Let  $d = S_2(j)$  if this entry exists, or  $d = \infty$  if  $|S_2| < j$ . The first two rows of  $P$  look like this (writing  $a' = a - 1 = S_1(j)$ ):

		$d$							
		$a'$							

or

							$a'$		

In both cases,  $a - 1 = a' < d$ , so  $a < d$  since  $a$  does not appear in  $P$ . When we use the default rule to insert  $b = a$  into column  $k$  of row 2,  $a$  bumps  $d$  or some earlier value. This means that  $k \leq j$ , and hence  $S'_1(k) = S_1(k) \leq S_1(j) = a - 1 < a = b = S'_2(k)$ . So no column violation occurs in column  $k$ .

- (4) Suppose  $\max(S_2) < b$ , so that  $S'_2$  consists of  $S_2$  followed by a new cell containing  $b$ . We argue similarly to the previous case to see that no column violation can occur. In fact, the proof in (3) applies verbatim here if we let  $k = |S_2| + 1$  and use the convention that  $c = S_2(k) = \infty$ . In particular, the deduction that  $k \leq j$  in subcase 1 and subcase 2 shows that the new row 2 cannot be longer than the new row 1, so that the shape of  $P'$  is still a partition diagram.

This concludes the proof of [Lemma 12](#).

#### 4. Properties of the $\text{RSK}_p$ algorithms

In this section, we show that the  $\text{RSK}_p$  algorithms share many of the familiar properties of the classical RSK algorithm.

##### 4.1. Descents and the $Q_p$ -recording tableau

Our first goal is to show that the descent set of a permutation maps under  $\text{RSK}_p$  to the descent set of the permutation's  $Q_p$ -recording tableau. In [Theorem 24](#), we will prove the sharper result that  $Q_p(w) = Q(w)$  for all  $w \in S_n$  and all  $p \geq 1$ . The proof of that theorem does not depend on the results of this subsection, so readers may omit this subsection if they wish. We begin with a lemma describing where new boxes appear in two successive  $p$ -tableau insertions.

**Lemma 17.** *Let  $P$  be a partial standard tableau and let  $a, c$  be values not appearing in  $P$ . Suppose  $\text{TI}_p(P, a) = P'$ , where  $P'$  has a new box in row  $r$ , and  $\text{TI}_p(P', c) = P''$ , where  $P''$  has a new box in row  $s$ . (a) If  $a < c$ , then  $s \leq r$ . (b) If  $a > c$ , then  $s > r$ .*

**Proof.** Let  $S_1$  be the first row of  $P$ . We may assume  $S_1 \neq \emptyset$ . To prove (a), assume  $a < c$ . If  $s = 1$ , the conclusion  $s \leq r$  holds. If  $r = 1$ , we must have  $\max(S_1) < a < c$ , so  $s = 1$ , and  $s \leq r$  holds. Now assume  $r > 1$  and  $s > 1$ , which means both  $a$  and  $c$  bump values (possibly themselves) from row 1 into row 2. Define  $(b, S'_1) = \text{RI}_p(S_1, a)$  and  $(d, S''_1) = \text{RI}_p(S'_1, c)$ . It suffices to show that  $b < d$ , by induction on the number of rows of  $P$ .

*Case 1.* Assume special rule 1 is used to compute  $\text{RI}_p(S_1, a)$ . Since  $a + 1 \in S_1$  and  $a + 2 \in S_1$ , we must have  $c > a + 2$ . Also  $b = a + 2 \notin S'_1$ . When  $c$  is inserted into  $S'_1$ , note that  $d > c > b$  if special rule 1 or the default rule applies. On the other hand, if special rule 2 applies, the fact that  $c > a + 2 \notin S'_1$  forces  $d \geq a + 4 > a + 2 = b$ . In more detail, if  $t$  is the maximum integer with  $[c - t, c - 1] \subseteq S'_1$ , then  $a + 2 < c - t$ , and so the bumped value  $d$  satisfies  $d \geq c - t + 1 \geq a + 4$ .

*Case 2.* Assume special rule 2 is used to compute  $\text{RI}_p(S_1, a)$ . Here  $a + 1 \in S_1$ ,  $a + 2 \notin S_1$ ,  $a + 2 \notin S'_1$ ,  $b \leq a$ , and  $c \geq a + 2$ . If  $c > a + 2$ , then  $a + 2 \notin S'_1$  forces  $d \geq a + 4 > a \geq b$  as in Case 1. If  $c = a + 2$ , then either  $d > c > b$  (if special rule 1 or the default rule applies when inserting  $c$ ) or  $d \geq b + 2 > b$  (if special rule 2 applies, since  $c > b \notin S'_1$ ).

*Case 3.* Assume the default rule is used to compute  $\text{RI}_p(S_1, a)$ . Then  $b$  is the smallest entry in  $S_1$  greater than  $a$ . Also,  $b$  must be followed in  $S_1$  by some larger entry  $e \geq a + 2$  (if  $e$  did not exist, then  $\max(S'_1) = a < c$  contradicts our assumption that  $s > 1$ ). We obtain  $S'_1$  from  $S_1$  by replacing  $b$  by  $a$ , so  $a \in S'_1$ ,  $e \in S'_1$ , but none of the values in  $[a + 1, e - 1]$  appears in  $S'_1$ . If  $c > e$ , then (as in previous cases)  $d > c > b$  or  $d \geq e + 1 > b$ . If  $a < c < e$ , consider subcases based on which rule is used to insert  $c$  into  $S'_1$ . Special rule 1 can apply only if  $c = e - 1$  and  $e + 1 \in S'_1$ ; then  $d = e + 1 > b$ . Special rule 2 can apply only if  $c = a + 1$ ,  $e = a + 2$ , and  $a + 3 \notin S'_1$ . But then  $a < b < e$  forces  $b = a + 1 = c$ ,

contradicting the fact that  $c$  does not appear in  $P$ . Finally, if the default rule is used to insert  $c$ , then  $c$  bumps  $e$ , so  $d = e > b$ .

To prove (b), assume  $a > c$ . If  $\max(S_1) < a$ , then  $r = 1$  and  $a$  is the last entry in  $S'_1 = \text{RI}_p(S_1, a)$ . Since  $c < a$ , the insertion of  $c$  into  $S'_1$  must bump some value into row 2, so that  $s \geq 2 > r$  in this case. Henceforth assume  $a < \max(S_1)$  and write  $(b, S'_1) = \text{RI}_p(S_1, a)$ . No matter what rule is used to insert  $a$  into row 1, we must have  $c < a \leq \max(S'_1)$ . Thus the insertion of  $c$  into  $S'_1$  must bump some value into row 2; write  $(d, S''_1) = \text{RI}_p(S'_1, c)$ . By induction on the number of rows of  $P$ , it suffices to show that  $b > d$ . We certainly have  $b \neq d$ , by hypothesis on  $P$ ,  $a$ , and  $c$ . Observe that if a special rule applies to compute  $\text{RI}_p(S'_1, c)$ , then  $d \leq c + 2$ ; otherwise,  $d$  is the smallest value in  $S'_1$  larger than  $c$ .

*Case 1.* Assume special rule 1 is used to compute  $\text{RI}_p(S_1, a)$ . Then  $a + 1 \in S_1$ ,  $a + 2 \in S_1$ , and  $b = a + 2$ . Now  $c \leq a - 1$  and  $a + 1 \in S'_1$ , so the largest possible value of  $d$  is  $a + 1 < b$ .

*Case 2.* Assume special rule 2 is used to compute  $\text{RI}_p(S_1, a)$ . Then all values in the interval  $[b - 1, a - 1]$  must appear in  $S_1$ , so that  $c \leq b - 2$ . As  $b - 1 \in S'_1$ , we see that the largest possible value of  $d$  is  $c + 2 = b$ . Since  $d$  cannot equal  $b$ , we conclude  $d < b$ .

*Case 3.* Assume the default rule is used to compute  $\text{RI}_p(S_1, a)$ . Then  $b$  is the smallest entry in  $S_1$  greater than  $a$ . Note  $c < a \in S'_1$ . So if the default rule is used to compute  $\text{RI}_p(S'_1, c)$ , then  $d \leq a < b$ , as needed. If a special rule is used to compute  $\text{RI}_p(S'_1, c)$ , then  $d \leq c + 2 \leq a + 1 \leq b$ . Since  $d$  cannot equal  $b$ , we conclude  $d < b$ .  $\square$

**Theorem 18.** For all  $n, p \geq 1$  and all  $w \in S_n$ ,  $\text{Des}(w) = \text{Des}(Q_p(w))$ .

**Proof.** Fix  $n, p \geq 1$  and  $w = w_1 w_2 \cdots w_n \in S_n$ . For any  $i < n$ , we can apply the lemma with  $P = P_p(w_1 \cdots w_{i-1})$ ,  $a = w_i$ , and  $c = w_{i+1}$ . The conclusion is that  $w_i > w_{i+1}$  iff the new box created by  $w_{i+1}$  is in a strictly higher row than the new box created by  $w_i$  iff  $i + 1$  appears in  $Q_p(w)$  in a higher row than  $i$ . In other words,  $i \in \text{Des}(w)$  iff  $i \in \text{Des}(Q_p(w))$ . So  $\text{Des}(w) = \text{Des}(Q_p(w))$ .  $\square$

#### 4.2. Effect of $\text{RSK}_p$ on reading words of standard tableaux

If  $w \in S_n$  is the reading word of a standard tableau, it is known that  $P(w) = w$  (identifying the standard tableau with its reading word). We now show the same property holds for  $P_p$ . In fact, we prove the stronger result that  $\text{RSK}_p(w) = \text{RSK}(w)$  whenever  $w$  is the reading word of a standard tableau.

**Theorem 19.** Suppose  $p \geq 1$  and  $w$  is the reading word of a partial standard tableau. Then  $P_p(w) = P(w) = w$  and  $Q_p(w) = Q(w)$ .

**Proof.** We give the proof in conjunction with a running example where  $w$  is the reading word of the partial standard tableau

$$U = \begin{array}{|c|c|c|c|c|} \hline v_1 & v_2 & v_3 & v_4 & v_5 \\ \hline u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \\ \hline t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 \\ \hline s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 & s_8 \\ \hline \end{array} .$$

In general,  $w$  consists of ascending runs  $S_\ell \cdots S_2 S_1$ , where we write  $S_1 = s_1 < s_2 < \cdots < s_{k_1}$ ,  $S_2 = t_1 < t_2 < \cdots < t_{k_2}$ , etc. If  $\ell = 1$ , then  $w$  itself is an increasing sequence, and the conclusions of the theorem are evident. Otherwise, note that  $w' = S_\ell \cdots S_2$  is also the reading word of a partial standard tableau  $U'$ , obtained by ignoring the lowest row of the tableau encoded by  $w$ . By induction on  $\ell$ , we can assume that  $P_p(w') = P(w') = w'$  and  $Q_p(w') = Q(w')$ .

We now analyze what happens when we use  $\text{TI}_p$  to insert the entries of  $S_1$  into  $P(w')$ , one at a time. By induction on  $j \in \{0, 1, 2, \dots, k_1\}$ , we may assume that after  $s_1, \dots, s_j$  have been inserted, the current partial  $P_p$ -insertion tableau  $U^*$  is obtained from  $U'$  by moving all cells in the first  $j$  columns up one row, and placing  $s_1, \dots, s_j$  in the vacated cells in the bottom row. The base case  $j = 0$  vacuously holds; consider the induction step from some fixed  $j$  to  $j + 1$ . In our example, when  $j = 3$ , we are assuming that the current partial  $P_p$ -tableau looks like

$$U^* = \begin{array}{|c|c|c|} \hline v_1 & v_2 & v_3 \\ \hline u_1 & u_2 & u_3 & u_4 & u_5 \\ \hline t_1 & t_2 & t_3 & u_4 & u_5 & u_6 \\ \hline s_1 & s_2 & s_3 & t_4 & t_5 & t_6 & t_7 \\ \hline \end{array} ,$$

and we want to show that

$$\text{TI}_p(U^*, s_4) = \begin{array}{|c|c|c|c|} \hline v_1 & v_2 & v_3 & v_4 \\ \hline u_1 & u_2 & u_3 & u_4 & v_5 \\ \hline t_1 & t_2 & t_3 & t_4 & u_5 & u_6 \\ \hline s_1 & s_2 & s_3 & s_4 & t_5 & t_6 & t_7 \\ \hline \end{array} .$$

More specifically, we shall show that when computing this new tableau, only the default rule (or the terminating case) is used in the  $p$ -row insertion subroutine. On one hand, this will mean that  $P_p$  and  $P$  act in exactly the same way on the prefix of  $w$  ending at  $s_{j+1}$ . On the other hand, since  $p$ -tableau insertion of  $s_{j+1}$  and classical tableau insertion of  $s_{j+1}$  produce the same new box, it also follows that  $Q_p$  and  $Q$  agree up to this point. When the induction reaches  $j + 1 = k_1$ , the entire algorithm is complete, and we will have  $P_p(w) = P(w) = w$  and  $Q_p(w) = Q(w)$  as needed.

In our example, consider the insertion of  $s_4$  into the row  $s_1, s_2, s_3, t_4, t_5, t_6, t_7$ . We know  $s_3 < s_4 < t_4$  since  $U$  is a partial standard tableau, so the classical insertion algorithm would replace  $t_4$  by  $s_4$  and bump  $t_4$  into row 2. Since this action is the default rule in  $p$ -row insertion, it suffices to show that special rules 1 and 2 cannot apply in this situation. Special rule 1 could only apply if  $t_4$  and  $t_5$  exist,  $t_4 = s_4 + 1$ , and  $t_5 = s_4 + 2$ . But then  $s_5$  must exist (since  $U$  has partition shape) and  $s_4 < s_5 < t_5 = s_4 + 2$  (since  $U$  is a partial standard tableau). But this forces  $s_5 = s_4 + 1 = t_4$ , which cannot happen since  $U$  has no repeated values. So special rule 1 cannot apply in our example, and the



same argument holds in general. Similarly, special rule 2 could only apply if  $s_3$  and  $t_4$  exist,  $s_3 = s_4 - 1$ , and  $t_4 = s_4 + 1$ . But then  $t_3$  must exist and  $s_3 < t_3 < t_4$ , leading to the impossible conclusion  $t_3 = s_4$ .

So far we know that both  $p$ -insertion and classical insertion bump  $t_4$  into row 2. We now repeat the argument used in row 1 to see that both algorithms must bump  $u_4$  into row 3. Then both algorithms bump  $v_4$  into row 4, and (in our example) both algorithms stop with  $v_4$  residing in a new box at the end of row 4. To see this holds in general, we use another induction on  $\ell$  to see that the argument in the previous paragraph always applies in each successive row, until we reach the lowest row with no value in column  $j + 1$ . The earlier induction hypothesis (describing how  $U^*$  is related to  $U'$ ) assures us that the row we have reached does have a value in column  $j$ , since otherwise the row below it could not have had a value in column  $j + 1$ . We have now shown that  $\text{TI}_p(U^*, s_{j+1})$  has the form described in the inductive hypothesis, which completes the proof.  $\square$

#### 4.3. Knuth relations and Assaf relations

Knuth [15] introduced the famous *Knuth relations* on words to implement the row insertion operation used in the classical RSK algorithm. It can be shown that for  $v, w \in S_n$ ,  $P(v) = P(w)$  iff  $w$  can be transformed into  $v$  by applying a sequence of Knuth relations (see, e.g., [8, Chaps. 2–3]). Haiman [13] studied the *dual Knuth relations*, which have the property that  $Q(v) = Q(w)$  iff  $w$  can be transformed into  $v$  by applying a sequence of dual Knuth relations. In her study of Macdonald polynomials and LLT polynomials via dual equivalence graphs, Assaf [3] introduced modifications of the dual Knuth relations that we will call *dual Assaf relations*. Our purpose here is to define *Assaf relations* and show how an appropriate mixture of Knuth relations and Assaf relations can be used to implement  $p$ -row insertion.

Fix a positive integer  $p$ . We define the *elementary  $p$ -Knuth–Assaf relations* as follows. Suppose  $w, w'$  are partial permutations,  $y, z$  are words (possibly empty), and  $a, b, c$  are positive integers such that  $a < b < c$ . The ordered pair  $(w, w')$  is an *elementary  $p$ -Knuth–Assaf relation* iff one of the following four rules applies.

- **A1<sub>p</sub>**:  $w = ybcaz$  and  $w' = ycabz$  where  $p \leq a$  and  $a, b, c$  are consecutive integers.
- **A2<sub>p</sub>**:  $w = yacbz$  and  $w' = ybacz$  where  $p \leq a$  and  $a, b, c$  are consecutive integers.
- **K1<sub>p</sub>**:  $w = ybcaz$  and  $w' = ybacz$  where  $p > a$  or  $a, b, c$  are not consecutive integers.
- **K2<sub>p</sub>**:  $w = yacbz$  and  $w' = ycabz$  where  $p > a$  or  $a, b, c$  are not consecutive integers.

We omit the subscript  $p$  when it is clear from context. Let  $\sim_p$  denote the equivalence relation on the set of partial permutations generated by all the ordered pairs  $(w, w')$ , so that  $u \sim_p v$  iff  $v$  can be obtained from  $u$  by applying a finite sequence of transformations  $w \mapsto w'$  or  $w' \mapsto w$ , where  $(w, w')$  is an elementary  $p$ -Knuth–Assaf relation. Note that

the  $p$ -Assaf relations **A1** and **A2** cause a cyclic rotation of consecutive values  $a, b = a + 1, c = a + 2$  in adjacent positions of  $w$ , when  $p \leq a$  and the middle value is not in the middle position. The  $p$ -Knuth relations **K1** and **K2** cause the extreme values  $a, b, c$  in adjacent positions of  $w$  to swap, when the middle value is not in the middle position and when  $a, b, c$  are not three consecutive integers or  $p > a$ . More general versions of the Knuth–Assaf relations (indexed by partitions  $\mu$  instead of integers  $p$ ) will be discussed in §5.3.

#### 4.4. Implementing $p$ -Row insertion via Knuth–Assaf relations

Our next goal is to show how the Knuth–Assaf relations can be used to implement the  $p$ -row insertion subroutine.

**Lemma 20.** *Let  $S = s_1 < \cdots < s_k$  be an increasing sequence,  $a \notin S$ , and  $p \geq 1$ . If  $\text{RI}_p(S, a) = (z, S')$ , then the word  $Sa$  can be transformed into the word  $zS'$  by repeated moves of the form  $w \mapsto w'$ , where  $(w, w')$  is one of the elementary  $p$ -Knuth–Assaf relations.*

**Proof.** If  $S = \emptyset$  or  $\max(S) < a$ , then  $z$  does not exist,  $zS' = Sa$ , and the lemma holds. Now assume  $a < \max(S)$  and  $s_j$  is the smallest symbol in  $S$  greater than  $a$ . We can implement the classical row insertion algorithm  $\text{RI}(S, a)$  in two stages. In stage 1, use rule **K1** (ignoring the conditions on  $p, a, b, c$ ) to move  $a$  to the left past all letters in  $S$  larger than  $s_j$ :

$$\begin{aligned} Sa &= s_1 \cdots s_j \cdots s_{k-1} s_k a \\ &\mapsto s_1 \cdots s_j \cdots s_{k-2} s_{k-1} a s_k \\ &\mapsto s_1 \cdots s_j \cdots s_{k-3} s_{k-2} a s_{k-1} s_k \\ &\mapsto \cdots \\ &\mapsto s_1 \cdots s_j s_{j+1} a \cdots s_k \\ &\mapsto s_1 \cdots s_j a s_{j+1} \cdots s_k. \end{aligned}$$

Stage 1 is now complete. If  $j = 1$ , then  $z = s_j = s_1$  and we have reached the word  $zS'$ . Otherwise, we have  $s_{j-1} < a < s_j$ , and we proceed to stage 2. In stage 2, use rule **K2** (ignoring the conditions on  $p, a, b, c$ ) to move the bumped value  $z = s_j$  to the left past all letters in  $S$  smaller than  $s_j$ :

$$\begin{aligned} s_1 \cdots s_{j-3} s_{j-2} s_{j-1} s_j a s_{j+1} \cdots s_k \\ &\mapsto s_1 \cdots s_{j-3} s_{j-2} s_j s_{j-1} a s_{j+1} \cdots s_k \\ &\mapsto s_1 \cdots s_{j-4} s_{j-3} s_j s_{j-2} s_{j-1} \cdots s_k \end{aligned}$$

$$\begin{aligned}
 &\mapsto s_1 \cdots s_{j-4} s_j s_{j-3} s_{j-2} s_{j-1} \cdots s_k \\
 &\mapsto \cdots \\
 &\mapsto s_1 s_2 s_j s_3 \cdots s_k \\
 &\mapsto s_1 s_j s_2 s_3 \cdots s_k \\
 &\mapsto s_j s_1 s_2 s_3 \cdots s_k = zS'.
 \end{aligned}$$

(See the first example following the proof.)

Now consider how the two stages are affected by the new rules **A1** and **A2**, which replace the corresponding rules **K1** and **K2** when the three letters involved are consecutive integers  $\geq p$ . Stage 1 proceeds exactly as before, except when  $p \leq a$  and  $a+1 \in S$  and  $a+2 \in S$  (the condition under which special rule 1 applies in the computation of  $\text{RI}_p(S, a)$ ). In this case, the last step of stage 1 uses **A1** instead of **K1**, as shown here:

$$Sa \mapsto \cdots \mapsto s_1 \cdots s_{j-1}, a+1, a+2, a \cdots s_k \mapsto s_1 \cdots s_{j-1}, a+2, a, a+1 \cdots s_k.$$

At this point, stage 2 proceeds as before to move  $z = a+2$  (instead of  $z = a+1$ ) all the way to the left via repeated use of **K2**; since all symbols  $s_1, \dots, s_{j-1}$  are less than  $a$ , the new rule **A2** will never be used here. Thus the  $p$ -Knuth–Assaf relations have implemented special rule 1 correctly. (See the second example below.)

Now suppose special rule 1 did not apply, so that stage 1 converts the word  $Sa$  to the word  $s_1 \cdots s_{j-1}, s_j, a, s_{j+1} \cdots s_k$ , where  $j = 1$  or  $s_{j-1} < a < s_j$ . Stage 2 will begin as in the classical case, switching  $s_{j-1}$  and  $s_j$  via rule **K2**, except when  $s_{j-1} = a-1$  and  $s_j = a+1$  and  $p \leq a-1$ . Since special rule 1 did not apply, we must also have  $a+2 \notin S$ , so that the exception occurs exactly when the hypothesis of special rule 2 is satisfied. If this exception does not occur at the beginning of stage 2, then we can move  $s_j$  all the way to the left using rule **K2**. In particular, rule **A2** cannot apply to the triple  $s_{j-2}, s_j, s_{j-1}$ , because these three integers are not consecutive (as  $s_{j-1} < a < s_j$ ). Similarly, rule **A2** cannot apply to  $s_{j-3}, s_j, s_{j-2}$  because  $s_{j-2} < s_{j-1} < s_j$ , and so on. Thus the  $p$ -Knuth–Assaf relations will implement the default insertion rule in this situation.

Now return to the case where the exception does occur at the start of stage 2. With notation as in special rule 2, the current word (after applying **A2** to change the subword  $a-1, a+1, a$  to  $a, a-1, a+1$ ) looks like

$$s_1 \cdots s_i, a-t, a-t+1 \cdots a-3, a-2, a, a-1, a+1, s_{j+1} \cdots s_k. \quad (9)$$

Assuming  $t \geq 2$  and  $p \leq a-2$ , we can apply **A2** to the triple  $a-2, a, a-1$  to get the word

$$s_1 \cdots s_i, a-t, a-t+1 \cdots a-3, a-1, a-2, a, a+1, s_{j+1} \cdots s_k.$$

Assuming  $t \geq 3$  and  $p \leq a - 3$ , we can apply **A2** to the triple  $a - 3, a - 1, a - 2$  to get the word

$$s_1 \cdots s_i, a - t, a - t + 1 \cdots a - 2, a - 3, a - 1, a, a + 1, s_{j+1} \cdots s_k.$$

We can continue applying rule **A2** until one of two things happens. On one hand, if  $p \leq a - t$ , we eventually reach

$$s_1 \cdots s_i, a - t + 1, a - t, a - t + 2, a - t + 3 \cdots a, a + 1, s_{j+1} \cdots s_k$$

where  $s_i < a - t - 1$  or  $s_i$  does not exist. If  $s_i$  exists, we now apply **K2** (instead of **A2**) to the triple  $s_i, a - t + 1, a - t$  to produce  $a - t + 1, s_i, a - t$ . Now rule **K2** moves  $z = a - t + 1$  all the way to the left as before, giving the word  $zs'$ . Since  $a - t$  now appears at least two positions right of  $a - t + 1$ , rule **A2** can never apply again. (See the third and fourth examples below.)

On the other hand, if  $p > a - t$ , repeatedly applying rule **A2** to (9) eventually leads to

$$\cdots p - 1, p, p + 2, p + 1 \cdots s_k \mapsto p - 1, p + 1, p, p + 2 \cdots s_k,$$

where **A2** can no longer be applied to the triple  $p - 1, p + 1, p$  because  $p - 1 < p$ . In this case, rule **K2** applies instead, and  $z = p + 1$  moves all the way to the left as before. Thus the Knuth–Assaf relations have implemented special rule 2 correctly. (See the fifth example below.)

As remarked earlier, if neither special rule applies when computing  $\text{RI}_p(S, a)$ , we can always use **K1** in stage 1 and **K2** in stage 2 to implement the default rule, as in the classical case.  $\square$

**Example 21.** The following row-insertion examples illustrate the cases in the previous proof. In each case, we show which three symbols are about to change by underlining them. First, we compute  $\text{RI}_3(123689, 5) = (6, 123589)$  by the default rule:

$$1236\textbf{\underline{89}}5 \mapsto 1236\textbf{\underline{85}}9 \mapsto 123\textbf{\underline{65}}89 \mapsto 1\textbf{\underline{26}}3589 \mapsto \textbf{\underline{16}}23589 \mapsto 6123589.$$

Second, we compute  $\text{RI}_5(123679, 5) = (7, 123569)$  by special rule 1:

$$1236\textbf{\underline{79}}5 \mapsto 1236\textbf{\underline{75}}9 \mapsto 123\textbf{\underline{75}}69 \mapsto 1\textbf{\underline{27}}3569 \mapsto \textbf{\underline{17}}23569 \mapsto 7123569.$$

Third, we compute  $\text{RI}_1(13579, 6) = (6, 13579)$  by special rule 2:

$$135\textbf{\underline{79}}6 \mapsto 135\textbf{\underline{76}}9 \mapsto 1\textbf{\underline{36}}579 \mapsto \textbf{\underline{16}}3579 \mapsto 613579.$$

Fourth, we compute  $\text{RI}_1(134568, 7) = (4, 135678)$  by special rule 2:

$$1345\overline{687} \mapsto 1345\overline{768} \mapsto 134\overline{6578} \mapsto 1\overline{354678} \mapsto \overline{1435678} \mapsto 4135678.$$

Fifth, we compute  $\text{RI}_5(134568, 7) = (6, 134578)$  by special rule 2:

$$1345\overline{687} \mapsto 1345\overline{768} \mapsto 134\overline{6578} \mapsto 1\overline{364578} \mapsto \overline{1634578} \mapsto 6134578.$$

#### 4.5. Implementing $P_p$ via Knuth–Assaf relations

Since  $p$ -row insertion can be implemented by a sequence of elementary Knuth–Assaf relations, it readily follows that  $p$ -tableau insertion and  $P_p$  can be implemented by these relations as well. Specifically, consider the computation of  $\text{TI}_p(S_\ell \cdots S_2 S_1, a)$ , with notation as in §3.2. In the base case  $\text{RI}_p(S_1, a) = S'_1$ , the word  $S_\ell \cdots S_2 S_1 a$  is the same as the word  $S_\ell \cdots S_2 S'_1$ . Otherwise, when  $\text{RI}_p(S_1, a) = (b, S'_1)$ , Lemma 20 shows that we can transform  $S_\ell \cdots S_2 S_1 a$  to  $S_\ell \cdots S_2 b S'_1$  by the  $p$ -Knuth–Assaf relations. It follows by induction on  $\ell$  that  $S_\ell \cdots S_2 b S'_1$  can be transformed to  $\text{TI}_p(S_\ell \cdots S_2, b) S'_1$ , so that  $S_\ell \cdots S_2 S_1 a$  can be transformed to  $\text{TI}_p(S_\ell \cdots S_2 S_1, a)$ , as needed. We can now prove the following fundamental result.

**Theorem 22.** *For all  $p \geq 1$  and all partial permutations  $w$ ,  $w \sim_p P_p(w)$  (identifying  $P_p(w)$  with its reading word).*

**Proof.** Write  $w = w_1 w_2 \cdots w_n$ . We can implement the  $P_p$  algorithm on input  $w$  via the following loop: for  $i = 2, 3, \dots, n$ , replace the current subword  $w'_1 \cdots w'_i$  by  $\text{TI}_p(w'_1 \cdots w'_{i-1}, w'_i)$ . We have just seen that each loop iteration transforms the subword  $w'_1 \cdots w'_i$  via a sequence of  $p$ -Knuth–Assaf relations. Thus the original word  $w$  is equivalent under  $\sim_p$  to the final output word, which is the reading word of the tableau  $P_p(w)$ .  $\square$

#### 4.6. Increasing subsequences

We want to show that for all  $p \geq 1$  and all partial permutations  $w$ ,  $Q_p(w) = Q(w)$ , so that the  $p$ -recording tableau always matches the classical recording tableau. We know that  $P_p(w)$  and  $Q_p(w)$  are partial standard tableaux of the same partition shape, which we denote  $\text{shape}_p(w)$ . As in the classical case,  $\text{shape}_p(w)$  can be determined by analysis of disjoint increasing subsequences in  $w$ .

**Theorem 23.** *Fix  $p \geq 1$  and a partial permutation  $w$ ; let  $\lambda = \text{shape}_p(w)$  have parts  $\lambda_1 \geq \lambda_2 \geq \cdots$ . For all  $k \geq 0$ , let  $L(w, k)$  be the maximum number of symbols appearing in any set of  $k$  pairwise disjoint increasing subsequences of  $w$ . Then  $L(w, k) = \lambda_1 + \lambda_2 + \cdots + \lambda_k$ .*

**Proof.** First suppose  $w$  happens to be the reading word of a partial standard tableau of partition shape  $\mu$ . By Theorem 19, we know  $P_p(w) = w$  (identifying the tableau

with its reading word), so  $\lambda = \text{shape}_p(w) = \mu$ . It is shown in [8, Lemma 1, p. 32] that  $L(w, k) = \mu_1 + \cdots + \mu_k = \lambda_1 + \cdots + \lambda_k$  in this case.

For general  $w$ , note  $P_p(w)$  is the reading word of a partial standard tableau of shape  $\lambda$ . By the previous paragraph, it suffices to show  $L(w, k) = L(P_p(w), k)$  for all  $k \geq 0$ . From §4.5, we know that  $P_p(w)$  can be obtained from  $w$  by a finite sequence of elementary  $p$ -Knuth–Assaf relations. So it suffices to prove that if  $w'$  is obtained from  $w$  by applying one of the relations **A1**, **A2**, **K1**, **K2**, then  $L(w, k) = L(w', k)$  for all  $k \geq 0$ . The case of the classical Knuth relations **K1** and **K2** is treated in [8, Lemma 2, p. 32].

To analyze **A1**, assume

$$w = ybc az, \quad w' = ycabz \text{ where } p \leq a \text{ and } a < b < c \text{ are consecutive integers.}$$

Consider any collection  $S_1, \dots, S_k$  of  $k$  disjoint increasing subsequences of  $w$ . We construct a collection  $S'_1, \dots, S'_k$  of  $k$  disjoint increasing subsequences of  $w'$  using the same number of symbols, as follows. We can let  $S'_i = S_i$  for all  $i$ , except when  $b$  and  $c$  both belong to the same subsequence (say  $S_1$ ). In this case, form  $S'_1$  by replacing  $b$  and  $c$  in  $S_1$  by  $a$  and  $b$ , respectively. If  $a$  appears in some  $S_j$  (where necessarily  $j \neq 1$ ), form  $S'_j$  by replacing  $a$  by  $c$ . Let  $S'_i = S_i$  for all  $i \neq 1, j$ . Since  $a, b, c$  are consecutive integers and  $w$  has no repeated symbols, the new subsequences are all still increasing. Likewise, we can pass from a collection  $S'_1, \dots, S'_k$  of  $k$  disjoint increasing subsequences of  $w'$  to a similar collection  $S_1, \dots, S_k$  for  $w$  of the same total size. No change is needed unless  $a, b$  both appear in the same sequence, say  $S'_1$ . In this case, replace  $a, b$  by  $b, c$  to get  $S_1$ , and also replace  $c$  (if it appears in some other  $S'_j$ ) by  $a$  to get  $S_j$ . These constructions show that  $L(w, k) = L(w', k)$  in this case.

The same proof applies verbatim when  $w$  and  $w'$  are related by **A2**, if we interchange all primed and unprimed variables. We need only observe that no change is needed in the case where  $a, c$  appear in the same increasing sequence  $S_1$  (or  $S'_1$ ), since  $b$  cannot also appear in that sequence.  $\square$

#### 4.7. Comparison of $Q_p$ and $Q$

We can use the increasing subsequence characterization of the shape of  $Q_p(w)$  to deduce the following fundamental fact.

**Theorem 24.** For all  $p \geq 1$  and all partial permutations  $w$ ,  $Q_p(w) = Q(w)$ .

**Proof.** Let  $w = w_1 w_2 \cdots w_n$ . We show that for  $1 \leq i \leq n$ ,  $i$  appears in the same box in  $Q_p(w)$  and  $Q(w)$ . Write  $\alpha, \beta, \lambda, \mu$  for the partitions giving the shapes of the tableaux  $Q_p(w_1 \cdots w_{i-1})$ ,  $Q_p(w_1 \cdots w_i)$ ,  $Q(w_1 \cdots w_{i-1})$ , and  $Q(w_1 \cdots w_i)$ , respectively. The location of  $i$  in  $Q_p(w)$  is the unique box in the Ferrers diagram of  $\beta$  not present in the diagram of  $\alpha$ , whereas the location of  $i$  in  $Q(w)$  is the unique box in the Ferrers diagram of  $\mu$  not present in the diagram of  $\lambda$ . Now, for all  $k \geq 0$ , the subsequence theorem tells us that

$$\alpha_1 + \cdots + \alpha_k = L(w_1 \cdots w_{i-1}, k) = \lambda_1 + \cdots + \lambda_k,$$

so that  $\alpha = \lambda$ . Similarly  $\beta = \mu$ , so  $i$  appears in the same box in  $Q_p(w)$  and  $Q(w)$ .  $\square$

## 5. Application of RSK algorithms to $q, t$ -Kostka coefficients

### 5.1. The algorithms $\text{RSK}^\mu$

We are finally ready to implement the strategy outlined at the end of §1.6 for finding certain  $q, t$ -Kostka coefficients. Let  $\text{Par}^*(n)$  denote the set of partitions  $\mu \in \text{Par}(n)$  with  $\mu_1 \leq 3$  and  $\mu_2 \leq 2$ . For  $\mu \in \text{Par}^*(n)$  with  $m_1(\mu)$  parts equal to 1, define  $p = p(\mu) = m_1(\mu) + 1$ . For  $w \in S_n$ , define

$$\text{RSK}^\mu(w) = (P^\mu(w), Q^\mu(w)) = (Q_p(w^{-1}), P_p(w^{-1})).$$

We must check that  $\text{RSK}^\mu$  has properties (i), (ii), and (iii) from Theorem 7. Property (i) follows from Theorem 14, since  $\text{RSK}^\mu$  is the composition of the bijection  $w \mapsto w^{-1}$  on  $S_n$ , followed by the bijection  $\text{RSK}_p$ , followed by the bijection  $(U, V) \mapsto (V, U)$  on pairs of standard tableaux of the same shape. Property (ii) follows from Theorem 18 (or Theorem 24), since for any  $w \in S_n$ ,

$$\text{Des}(P^\mu(w)) = \text{Des}(Q_p(w^{-1})) = \text{Des}(w^{-1}) = \text{IDes}(w).$$

Finally, property (iii) follows from the theorem below, which will be proved in this section.

**Theorem 25.** *For all  $\mu \in \text{Par}^*(n)$  and all  $w \in S_n$ ,*

$$\text{inv}_\mu(w) = \text{inv}_\mu(\text{rw}(Q^\mu(w))^{-1}) \text{ and } \text{maj}_\mu(w) = \text{maj}_\mu(\text{rw}(Q^\mu(w))^{-1}).$$

Granting this theorem for the moment, we deduce Theorem 5 as follows. Fix  $\mu \in \text{Par}^*(n)$  and let  $p = p(\mu)$ . Given  $T \in \text{SYT}(\lambda)$ , define  $w = \text{rw}(T)^{-1} \in S_n$ . Identifying  $T$  with its reading word, we have  $w^{-1} = T$ . Now, Theorem 19 tells us that  $Q^\mu(w) = P_p(w^{-1}) = w^{-1} = T$ . By (6) in Theorem 7, (4) holds with  $\tilde{a}_\mu(T) = \text{inv}_\mu(w) = \text{inv}_\mu(\text{rw}(T)^{-1})$  and  $\tilde{b}_\mu(T) = \text{maj}_\mu(w) = \text{maj}_\mu(\text{rw}(T)^{-1})$ . This completes the proof of Theorem 5.

### 5.2. Alternate formulation of $\text{inv}_\mu$ and $\text{maj}_\mu$

To prove Theorem 25, it is convenient to use an alternate formulation of Haglund's permutation statistics  $\text{inv}_\mu$  and  $\text{maj}_\mu$ . Given any partition  $\mu \in \text{Par}(n)$ , we identify cells in the Ferrers diagram of  $\mu$  with the positions  $\{1, 2, \dots, n\}$  in a permutation by numbering the cells row by row from the top down, as shown here.

$$\mu = \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 3 & 4 & 5 \\ \hline 6 & 7 & 8 & 9 \\ \hline \end{array}$$

For each position  $k \in [n]$ , let  $\text{arm}(k)$  be the number of cells to the right of the cell numbered  $k$  in its row, and let  $\text{leg}(k)$  be the number of cells above the cell numbered  $k$  in its column. In our example,  $\text{arm}(6) = 3$  and  $\text{leg}(6) = 2$ , whereas  $\text{arm}(5) = \text{leg}(5) = 0$ . For a position  $k$  above the bottom row, the *pistol starting at  $k$*  is the interval of integers  $\text{pstl}(k) = [k, m]$ , where  $m$  is the number of the cell directly below  $k$ . For a position  $k$  in the bottom row, the pistol starting at  $k$  is  $\text{pstl}(k) = [k, n]$ . In our example,  $\text{pstl}(2) = [2, 4]$ ,  $\text{pstl}(4) = [4, 7]$ , and  $\text{pstl}(7) = [7, 9]$ . Finally, let  $d(k)$  be the number of the cell directly below  $k$ , if  $k$  is above the bottom row, and let  $d(k) = \infty$  otherwise. In our example,  $d(4) = 7$  and  $d(7) = \infty$ .

Now given  $w \in S_n$ , we fill the diagram of  $\mu$  with the entries of  $w$  as described in §1.1. For example, if  $\mu = (4, 3, 2)$  and  $w = 361958274$ , the filled diagram looks like

$$\begin{array}{|c|c|} \hline 3 & 6 \\ \hline 1 & 9 & 5 \\ \hline 8 & 2 & 7 & 4 \\ \hline \end{array}.$$

Define  $\text{Des}_\mu(w)$  to be the set of all  $k \in [n]$  such that  $d(k) < \infty$  and  $w_k > w_{d(k)}$ . One readily shows that

$$\text{maj}_\mu(w) = \sum_{k \in \text{Des}_\mu(w)} (1 + \text{leg}(k)).$$

In our example,  $\text{Des}_\mu(w) = \{1, 4\}$  and  $\text{maj}_\mu(w) = 3$ .

Next, define a  $\mu$ -inversion pair of  $w$  to be a pair of positions  $(j, k)$  with  $1 \leq j < k \leq n$ ,  $k \in \text{pstl}(j)$ ,  $k \neq d(j)$ , and  $w_j > w_k$ . It can be shown that

$$\text{inv}_\mu(w) = (\text{the number of } \mu\text{-inversion pairs of } w) - \sum_{s \in \text{Des}_\mu(w)} \text{arm}(s). \tag{10}$$

A proof may be found, for instance, in [11, §2]. In our example, the  $\mu$ -inversion pairs of  $w$  are  $(2, 3)$ ,  $(4, 5)$ ,  $(4, 6)$ ,  $(5, 7)$ ,  $(6, 7)$ ,  $(6, 8)$ ,  $(6, 9)$ , and  $(8, 9)$ , corresponding to the inverted values  $6 > 1$ ,  $9 > 5$ ,  $9 > 8$ ,  $5 > 2$ ,  $8 > 2$ ,  $8 > 7$ ,  $8 > 4$ , and  $7 > 4$ , respectively. The formula above states that  $\text{inv}_\mu(w) = 8 - (1 + 1) = 6$ , which agrees with the value obtained by counting inversion triples.

5.3. Knuth–Assaf relations for partitions

We now introduce a more general form of the Knuth–Assaf relations. Fix a partition  $\mu \in \text{Par}(n)$ . Assume  $w$  and  $w'$  are permutations in  $S_n$ ,  $y, z$  are words (possibly empty), and  $a, b, c$  are values in  $[n]$  such that  $a < b < c$ . The *elementary Knuth–Assaf relations for  $\mu$*  are ordered pairs  $(w, w')$  satisfying one of the following four rules.



- **A1 $^\mu$** :  $w = ybc az$  and  $w' = ycabz$  where  $c \in \text{pstl}(a)$ .
- **A2 $^\mu$** :  $w = yacb z$  and  $w' = ybac z$  where  $c \in \text{pstl}(a)$ .
- **K1 $^\mu$** :  $w = ybc az$  and  $w' = ybac z$  where  $c \notin \text{pstl}(a)$ .
- **K2 $^\mu$** :  $w = yacb z$  and  $w' = ycabz$  where  $c \notin \text{pstl}(a)$ .

We let  $\sim_\mu$  denote the equivalence relation on  $S_n$  generated by all the ordered pairs  $(w, w')$  with  $w, w' \in S_n$ . Thus, for all  $u, v \in S_n$ ,  $u \sim_\mu v$  iff  $v$  can be obtained from  $u$  by applying a finite sequence of transformations  $w \mapsto w'$  or  $w' \mapsto w$ , where  $(w, w')$  is an elementary Knuth–Assaf relation for  $\mu$ .

**Example 26.** Suppose  $\mu \in \text{Par}^*(n)$ , and let  $p = p(\mu)$ . One readily checks that  $\text{pstl}(a) = [a, a + 1]$  for  $1 \leq a < p$ ,  $\text{pstl}(a) = [a, a + 2]$  for  $p \leq a < n - 1$ , and  $\text{pstl}(a) = [a, n]$  for  $n - 1 \leq a \leq n$ . These formulas are illustrated below for  $\mu = (3, 2, 2, 1, 1)$  and  $\mu = (2, 2, 2, 2, 1)$ , which have  $p = 3$  and  $p = 2$ , respectively.

Comparison of the definitions shows that for  $\mu \in \text{Par}^*(n)$ , each elementary Knuth–Assaf relation for  $\mu$  coincides with the corresponding  $p$ -Knuth–Assaf relation, so that  $\sim_\mu$  and  $\sim_p$  give the same equivalence relation on  $S_n$ . In particular, for all  $w \in S_n$ , we know  $w^{-1} \sim_p \text{rw}(P_p(w^{-1}))$  by [Theorem 22](#), and so  $w^{-1} \sim_\mu \text{rw}(Q^\mu(w))$ .

#### 5.4. Preservation of Haglund's statistics

Assaf [3] introduced what we call the dual Knuth–Assaf relations for  $\mu$  because they preserve Haglund’s permutation statistics  $\text{inv}_\mu$  and  $\text{maj}_\mu$ . The following lemma proves this fact in our notation (Assaf’s version does not need the inverse, because she uses the dual relations to the ones we defined).

**Lemma 27.** *For all  $\mu \in \text{Par}(n)$  and all  $w, v \in S_n$ , if  $w^{-1} \sim_\mu v^{-1}$ , then  $\text{maj}_\mu(v) = \text{maj}_\mu(w)$  and  $\text{inv}_\mu(v) = \text{inv}_\mu(w)$ .*

**Proof.** It suffices to prove the lemma when  $(w^{-1}, v^{-1})$  is one of the four elementary Knuth–Assaf relations defined above. Consider the four possible cases.

For relation  $\mathbf{A1}^\mu$ ,  $w^{-1} = ybcaz$  and  $v^{-1} = ycabz$  where  $a < b < c$  and  $c \in \text{pstl}(a)$ . Let  $i - 2$  be the length of  $y$ . Writing  $w$  and  $v$  in two-line form, we see

$$w = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i+1 & \cdots & i-1 & \cdots & i & \cdots \end{bmatrix},$$

$$v = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i & \cdots & i+1 & \cdots & i-1 & \cdots \end{bmatrix},$$

where the symbols not shown are the same in  $v$  and  $w$ . On one hand, consider two positions  $j < k$  with  $w_j \notin \{i-1, i, i+1\}$  or  $w_k \notin \{i-1, i, i+1\}$ . Since the three symbols  $i-1, i, i+1$  have the same relative order compared to all symbols not in this set, we see that  $(j, k)$  is a  $\mu$ -inversion pair for  $w$  iff  $(j, k)$  is a  $\mu$ -inversion pair for  $v$ . On the other hand, consider  $\mu$ -inversion pairs  $(j, k)$  where  $j, k \in \{a, b, c\}$ . Since  $a < b < c$  and  $c \in \text{pstl}(a)$ , the positions  $a, b, c$  all belong to the pistol  $\text{pstl}(a)$ . In the case  $c = d(a)$ , these positions could appear in the diagram of  $\mu$  as shown here (where  $\text{pstl}(a)$  is the set of cells labeled with  $a, b, c$ , or a star):

			$a$	*	*	$b$	*
*	*	$c$					

or

			$a$	*	*	*	*
$b$	*	$c$					

(11)

In this case, both  $w$  and  $v$  have exactly one  $\mu$ -inversion pair  $(j, k)$  with  $j, k \in \{a, b, c\}$ . Specifically,  $(a, b)$  is a  $\mu$ -inversion pair of  $w$  since  $i+1 > i-1$ , whereas  $(b, c)$  is a  $\mu$ -inversion pair of  $v$  since  $i+1 > i-1$ .

Now consider the case  $c \neq d(a)$ , as illustrated in these diagrams:

			$a$	$b$	*	*	$c$
*	*	*					

or

			$a$	*	*	*	$b$
*	$c$	*					

In this case,  $w$  has two  $\mu$ -inversion pairs  $(a, b)$  and  $(a, c)$ , whereas  $v$  also has two  $\mu$ -inversion pairs  $(a, c)$  and  $(b, c)$ . Thus,  $v$  and  $w$  have the same total number of  $\mu$ -inversion pairs.

Similarly, for a position  $k$  not in row 1 with  $w_k \notin \{i-1, i, i+1\}$  or  $w_{d(k)} \notin \{i-1, i, i+1\}$ , we see  $w_k > w_{d(k)}$  iff  $v_k > v_{d(k)}$ . Since  $a, b, c$  all belong to the pistol  $\text{pstl}(a)$ , the only way we could have  $w_k$  and  $w_{d(k)}$  both in  $\{i-1, i, i+1\}$  is if  $k = a$  and  $d(k) = c$ , as pictured in (11). In this case,  $k \in \text{Des}_\mu(w)$  since  $i+1 > i$ , and  $k \in \text{Des}_\mu(v)$  since  $i > i-1$ . We now see that  $\text{Des}_\mu(v) = \text{Des}_\mu(w)$ , hence  $\text{maj}_\mu(v) = \text{maj}_\mu(w)$  and (by (10))  $\text{inv}_\mu(v) = \text{inv}_\mu(w)$ .

Relation **A2** $^\mu$  is handled by a similar argument, comparing

$$w = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i-1 & \cdots & i+1 & \cdots & i & \cdots \end{bmatrix} \text{ to } v = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i & \cdots & i-1 & \cdots & i+1 & \cdots \end{bmatrix}.$$

For relation **K1** $^\mu$ ,  $w^{-1} = ybc az$  and  $v^{-1} = ybac z$  where  $a < b < c$  and  $c \notin \text{pstl}(a)$ , so

$$w = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i+1 & \cdots & i-1 & \cdots & i & \cdots \end{bmatrix},$$

$$v = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i & \cdots & i-1 & \cdots & i+1 & \cdots \end{bmatrix}.$$

Note that  $v$  is obtained from  $w$  by switching the positions of  $i$  and  $i+1$ . As before,  $\mu$ -descents and  $\mu$ -inversion pairs involving at least one symbol  $w_k$  not in  $\{i, i+1\}$  will be the same in  $w$  and  $v$ . Furthermore, no  $\mu$ -descent in  $w$  or  $v$  can involve both symbols  $i$  and  $i+1$ , since otherwise  $c = d(a)$ , implying  $c \in \text{pstl}(a)$ , contrary to hypothesis. Thus,  $\text{Des}_\mu(v) = \text{Des}_\mu(w)$  and so  $\text{maj}_\mu(v) = \text{maj}_\mu(w)$ . Similarly, since  $c \notin \text{pstl}(a)$ , no inversion pair in  $w$  or  $v$  can involve both symbols  $i$  and  $i+1$ , so (by (10))  $\text{inv}_\mu(v) = \text{inv}_\mu(w)$ .

Relation  $\mathbf{K2}^\mu$  is handled by a similar argument, comparing

$$w = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i-1 & \cdots & i+1 & \cdots & i & \cdots \end{bmatrix} \text{ to}$$

$$v = \begin{bmatrix} \cdots & a & \cdots & b & \cdots & c & \cdots \\ \cdots & i & \cdots & i+1 & \cdots & i-1 & \cdots \end{bmatrix}. \quad \square$$

We can now quickly prove [Theorem 25](#). Given  $\mu \in \text{Par}^*(n)$  and  $w \in S_n$ , let  $p = p(\mu)$ . By [Example 26](#),  $w^{-1} \sim_\mu \text{rw}(Q^\mu(w))$ . Applying [Lemma 27](#) with  $v = \text{rw}(Q^\mu(w))^{-1}$ , we conclude

$$\text{inv}_\mu(w) = \text{inv}_\mu(\text{rw}(Q^\mu(w))^{-1}) \text{ and } \text{maj}_\mu(w) = \text{maj}_\mu(\text{rw}(Q^\mu(w))^{-1}).$$

## 6. Conclusion

The present work suggests two natural directions for further research. First, the new  $\text{RSK}_p$  algorithms introduced here surely have many additional combinatorial properties similar to those of the classical  $\text{RSK}$  algorithm. Are there  $p$ -versions of jeu de taquin (the sliding game) and evacuation? If  $\text{RSK}_p(w) = (P, Q)$  and  $\text{RSK}_p(w^{-1}) = (P', Q')$ , how are  $P'$  and  $Q'$  related to  $P$  and  $Q$ ? In the classical case,  $Q' = P$  and  $P' = Q$ , but this is not universally true for general  $p$ . Is there a nice direct description for the “ $p$ -inverse map” defined by sending  $w \in S_n$  to  $\text{RSK}_p^{-1}(Q_p(w), P_p(w))$ ? Similarly, what involution on  $S_n$  arises by sending  $w$  to  $\text{RSK}_p^{-1}(P_p(w)', Q_p(w)')$ , where the prime denotes conjugation of standard tableaux?

The second avenue to explore involves using [Theorem 7](#) to find  $q, t$ -Kostka coefficients for partitions  $\mu$  outside the restricted class  $\text{Par}^*(n)$ . The trouble is figuring out how to define  $\text{RSK}^\mu$  for general partitions  $\mu$ . I believe that a deeper study of the methods of Assaf [\[2,3\]](#), Blasiak [\[4\]](#), Roberts [\[21\]](#), et al. may lead to more progress on this front. Just as the  $p$ -Knuth–Assaf relations (which arise from Knuth–Assaf relations for partitions  $\mu \in \text{Par}^*(n)$ ) lead to an implementation of  $p$ -row insertion, it may be that Assaf’s proposed process for transforming  $D$ -graphs into dual equivalence graphs contains clues

as to how the general  $\text{RSK}^\mu$  algorithms should proceed. I plan to explore both of these research directions in future papers.

## References

- [1] Sami Assaf, A generalized major index statistic, *Sém. Lothar. Combin.* 60 (2008) B60c.
- [2] Sami Assaf, Dual equivalence graphs I: a new paradigm for Schur positivity, *Forum Math. Sigma* 3 (2015) e12 (electronic).
- [3] Sami Assaf, Dual equivalence graphs, ribbon tableaux, and Macdonald polynomials, Ph.D. thesis, University of California at Berkeley, 2007.
- [4] Jonah Blasiak, Haglund's conjecture on 3-column Macdonald polynomials, preprint, available online at arXiv:1411.3646, 2015, accessed August 2016.
- [5] Jonah Blasiak, What makes a  $D_0$  graph Schur positive?, preprint, available online at arXiv:1411.3624, 2014, accessed August 2016.
- [6] Christophe Carré, Bernard Leclerc, Splitting the square of a Schur function into its symmetric and antisymmetric parts, *J. Algebraic Combin.* 4 (1995) 201–231.
- [7] Susanna Fishel, Statistics for special  $q, t$ -Kostka polynomials, *Proc. Amer. Math. Soc.* 123 (1995) 2961–2969.
- [8] William Fulton, *Young Tableaux*, Cambridge University Press, 1997.
- [9] Adriano Garsia, Mark Haiman, A graded representation model for Macdonald's polynomials, *Proc. Natl. Acad. Sci. USA* 90 (1993) 3607–3610.
- [10] James Haglund, A combinatorial model for the Macdonald polynomials, *Proc. Natl. Acad. Sci. USA* 101 (2004) 16127–16131.
- [11] James Haglund, Mark Haiman, Nicholas Loehr, A combinatorial formula for Macdonald polynomials, *J. Amer. Math. Soc.* 18 (2005) 735–761.
- [12] James Haglund, Mark Haiman, Nicholas Loehr, Jeffrey Remmel, Alexander Ulyanov, A combinatorial formula for the character of the diagonal coinvariants, *Duke Math. J.* 126 (2005) 195–232.
- [13] Mark Haiman, Dual equivalence with applications, including a conjecture of Proctor, *Discrete Math.* 99 (1992) 79–113.
- [14] Mark Haiman, Hilbert schemes, polygraphs, and the Macdonald positivity conjecture, *J. Amer. Math. Soc.* 14 (2001) 941–1006.
- [15] Donald E. Knuth, Permutations, matrices, and generalized Young tableaux, *Pacific J. Math.* 34 (1970) 709–727.
- [16] Luc Lapointe, Jennifer Morse, Tableau statistics for two part Macdonald polynomials, in: *Algebraic Combinatorics and Quantum Groups*, World Scientific Publishing, 2003, pp. 61–84.
- [17] Alain Lascoux, Bernard Leclerc, Jean-Yves Thibon, Ribbon tableaux, Hall–Littlewood functions, quantum affine algebras, and unipotent varieties, *J. Math. Phys.* 38 (1997) 1041–1068.
- [18] Nicholas A. Loehr, *Bijjective Combinatorics*, Chapman and Hall/CRC Press, 2011.
- [19] Ian Macdonald, A new class of symmetric functions, *Actes du 20e Séminaire Lotharingien* 372/S-20 (1988) 131–171.
- [20] Ian Macdonald, *Symmetric Functions and Hall Polynomials*, second ed., Oxford University Press, 1995.
- [21] Austin Roberts, Dual equivalence graphs revisited and the explicit Schur expansion of a family of LLT polynomials, *J. Algebraic Combin.* 39 (2014) 389–428.
- [22] Bruce Sagan, *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*, Springer, New York, 2001.
- [23] Richard Stanley, *Enumerative Combinatorics*, vol. 2, Cambridge University Press, 1999.
- [24] Marc van Leeuwen, Some bijective correspondences involving domino tableaux, *Electron. J. Combin.* 7 (2000) R35 (electronic).
- [25] Mike Zabrocki, Positivity for special cases of  $(q, t)$ -Kostka coefficients and standard tableaux statistics, *Electron. J. Combin.* 6 (1999) R41 (electronic).