

# A row analogue of Hecke column insertion

Daoji Huang, Mark Shimozono, and Tianyi Yu

July 2022

## Abstract

We introduce a new row insertion algorithm on decreasing tableaux and increasing tableaux, generalizing Edelman-Greene (EG) row insertion. Our row insertion algorithm is a nontrivial variation of Hecke column insertion which generalizes EG column insertion. Similar to Hecke column insertion, our row insertion is bijective and respects Hecke equivalence, and therefore recovers the expansions of Grothendieck symmetric functions into Grassmannian Grothendieck functions. In future work, we will use this row insertion to establish an expansion of products between Lascoux polynomials and certain Grothendieck polynomials, which cannot be done by Hecke column insertion.

## 1 Introduction

The EG insertion was introduced to give combinatorial expansions of Stanley symmetric functions into Schur functions [EG]. EG insertion comes in four flavors depending on the use of increasing versus decreasing tableaux and row versus column insertion.

The Hecke column insertion was introduced to give combinatorial expansions of Grothendieck symmetric functions into Grassmannian Grothendieck functions [BKSTY]. It can generalize the two flavors of column EG insertions. Even though it has a row version, it does not satisfy a suitable Pieri property that guarantees the recording tableaux to be set-valued tableaux. This property is required for the insertion algorithm to realize a combinatorial expansion of Grothendieck symmetric functions into Grassmannian Grothendieck symmetric functions. Our new insertion generalizes the row EG insertion and satisfies the desired Pieri property. Further applications of our algorithm to the nonsymmetric case of Grothendieck-to-Lascoux expansions, as well as the expansion of certain products of Grothendieck and Lascoux polynomials to Lascoux polynomials, are explored in [OY].

This novel insertion has the very unusual property that some values may be moved which are not part of the bumping path. It was originally inspired by, and can be computed by, certain row moves on marked bumpless pipedreams, which, like compatible pairs, are combinatorial objects whose generating function is a

Grothendieck polynomial. This connection will be developed in future work [HSY].

## 1.1 Stanley and Grothendieck symmetric functions

The main application of these various insertion algorithms, is to the expansions of Stanley (resp. Grothendieck) symmetric functions into Schur (resp. Grassmannian Grothendieck) functions.

A pair of words  $(a_1 \cdots a_n, i_1 \cdots i_n)$  is called *compatible*<sup>1</sup> if  $i_j \geq i_{j+1}$  and  $i_j = i_{j+1}$  implies  $a_j < a_{j+1}$  for all  $1 \leq j < n$ . Each word  $a$  has an associated permutation  $[a]_H$  (see §2.2). We say  $a$  is a *Hecke word* for  $w$  if  $[a]_H = w$ . Let  $\text{CP}_w$  be the set of compatible pairs  $(a, i)$  such that  $a$  is a Hecke word for  $w$ . Let  $\text{CP}_w^{\text{Red}}$  consists of  $(a, i) \in \text{CP}_w$  such that  $a$  is reduced (i.e.  $\text{len}(a) = \ell(w)$  where  $\ell(w)$  is the Coxeter length).

Then the Stanley symmetric function  $F_w$  and the Grothendieck symmetric function  $G_w$  can be defined as [BJS] [FK]

$$F_w = \sum_{(a,i) \in \text{CP}_w^{\text{Red}}} x^{\text{wt}(i)}$$

$$G_w = \sum_{(a,i) \in \text{CP}_w} (-1)^{|\text{wt}(i)| - \text{len}(w)} x^{\text{wt}(i)}.$$

Let  $\mathbb{Y}$  be the set of partitions. For  $\lambda = (\lambda_1 \geq \lambda_2 \geq \cdots) \in \mathbb{Y}$  let  $D(\lambda) = \{(i, j) \in \mathbb{Z}_{>0}^2 \mid 1 \leq j \leq \lambda_i\}$  be its diagram, with matrix-style indexing. A set-valued tableau  $T$  of shape  $\lambda \in \mathbb{Y}$  is a function which assigns to each  $s \in D(\lambda)$  a nonempty subset of positive integers, such that if  $s'$  is immediately to the right (resp. below)  $s$  in the same row (resp. column) then  $\max(T(s)) \leq \min(T(s'))$  (resp.  $\max(T(s)) < \min(T(s'))$ ). We denote by SVT (resp. RSVT) the set of set-valued tableaux (resp. reverse set-valued tableaux, meaning all inequalities are reversed). Let SSYT (resp. RSSYT) denote the set of semistandard (resp. reverse semistandard) Young tableaux, meaning set-valued (resp. reverse set-valued) tableaux in which each set is a singleton. The following is a reverse-set-valued tableau of shape  $(3, 2)$ .

5	5, 4	3, 2, 1
3, 2	2	

The Schur function  $s_\lambda$  and the Grassmannian Grothendieck symmetric func-

---

<sup>1</sup>These compatible sequences are the reverse words of those defined in [BJS].

tion  $G_\lambda$  each have two equivalent formulas ([Bu] for  $G_\lambda$ ):

$$s_\lambda = \sum_{\substack{Q \in \text{SSYT} \\ \text{shape}(Q) = \lambda}} x^{\text{wt}(Q)} = \sum_{\substack{Q \in \text{RSSYT} \\ \text{shape}(Q) = \lambda}} x^{\text{wt}(Q)}$$

$$G_\lambda = \sum_{\substack{Q \in \text{SVT} \\ \text{shape}(Q) = \lambda}} (-1)^{|\text{wt}(Q)| - |\lambda|} x^{\text{wt}(Q)} = \sum_{\substack{Q \in \text{RSVT} \\ \text{shape}(Q) = \lambda}} (-1)^{|\text{wt}(Q)| - |\lambda|} x^{\text{wt}(Q)}.$$

## 1.2 Expansions

The  $F_w$  (resp.  $G_w$ ) can be expanded into  $s_\lambda$  (resp.  $G_\lambda$ ). The expansion coefficients have geometric meaning; they contain all cohomological (resp.  $K$ -theoretic) equioriented type A quiver constants as special cases [BKSTY].

There are two ways to write down either of the two expansions, using either increasing tableaux or decreasing tableaux. For a permutation  $w$ , let  $\text{Inc}_w$  (resp.  $\text{Dec}_w$ ) be the set of increasing (resp. decreasing) tableaux  $P$  whose row word  $\text{row}(P)$  (resp. reverse row word  $\text{rev}(\text{row}(P))$ ); see §2.2) is a Hecke word for  $w$ . Let  $\text{Inc}_w^{\text{Red}}$  (resp.  $\text{Dec}_w^{\text{Red}}$ ) consists of tableaux in  $\text{Inc}_w$  (resp.  $\text{Dec}_w$ ) whose row word is reduced. Then we have ([EG] for  $F_w$  and [BKSTY] for  $G_w$ )

$$F_w = \sum_{P \in \text{Inc}_w^{\text{Red}}} s_{\text{shape}(P)} \quad (1)$$

$$= \sum_{P \in \text{Dec}_w^{\text{Red}}} s_{\text{shape}(P)}, \quad (2)$$

$$G_w = \sum_{P \in \text{Inc}_w} (-1)^{\ell(w) - |\text{shape}(P)|} G_{\text{shape}(P)} \quad (3)$$

$$= \sum_{P \in \text{Dec}_w} (-1)^{\ell(w) - |\text{shape}(P)|} G_{\text{shape}(P)}. \quad (4)$$

## 1.3 Insertion algorithms: General requirements

Let  $A$  and  $B$  be sets of tableaux of partition shape. We use the notation

$$A \times_{\mathbb{Y}} B = \{(P, Q) \in A \times B \mid \text{shape}(P) = \text{shape}(Q)\} \quad (5)$$

for the fiber product over the maps  $A \rightarrow \mathbb{Y}$  and  $B \rightarrow \mathbb{Y}$  given by taking the shape of a tableau.

To give a combinatorial proof of (3) it suffices to produce a bijection  $\Phi_{IS} : \text{CP}_w \rightarrow \text{Inc}_w \times_{\mathbb{Y}} \text{SVT}$  or  $\Phi_{IR} : \text{CP}_w \rightarrow \text{Inc}_w \times_{\mathbb{Y}} \text{RSVT}$  which is weight-preserving:

$$(a, i) \mapsto (P, Q) \quad (6)$$

$$\text{wt}(i) = \text{wt}(Q). \quad (7)$$

Similarly to prove (4) it suffices to supply a weight-preserving bijection  $\Phi_{DS} : \text{CP}_w \rightarrow \text{Dec}_w \times_{\mathbb{Y}} \text{SVT}$  or  $\Phi_{DR} : \text{CP}_w \rightarrow \text{Dec}_w \times_{\mathbb{Y}} \text{RSVT}$ .

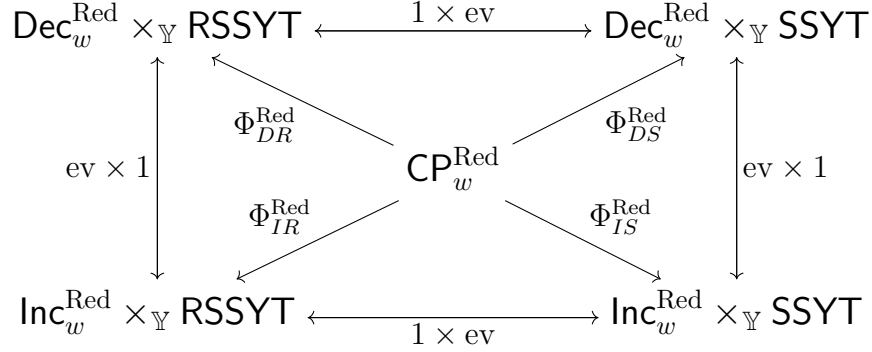


Figure 1: The four reduced bijections

#### 1.4 Edelman-Greene insertion: solution for reduced case

Historically first to be discovered were “reduced” restrictions of the above bijections. The expansions (1) and (2) are obtained via four weight-preserving bijections. These bijections are given by four variations of the Edelman-Greene insertion (EG insertion) [EG]:

- $\Phi_{IS}^{\text{Red}} : \text{CP}_w^{\text{Red}} \rightarrow \text{Inc}_w^{\text{Red}} \times_{\mathbb{Y}} \text{SSYT}$ : EG column insertion into increasing tableaux, starting from the right end of the compatible pairs.
- $\Phi_{IR}^{\text{Red}} : \text{CP}_w^{\text{Red}} \rightarrow \text{Inc}_w^{\text{Red}} \times_{\mathbb{Y}} \text{RSSYT}$ : EG row insertion into increasing tableaux, starting from the left end of the compatible pairs.
- $\Phi_{DS}^{\text{Red}} : \text{CP}_w^{\text{Red}} \rightarrow \text{Dec}_w^{\text{Red}} \times_{\mathbb{Y}} \text{SSYT}$ : EG row insertion into decreasing tableaux, starting from the right end of the compatible pairs.
- $\Phi_{DR}^{\text{Red}} : \text{CP}_w^{\text{Red}} \rightarrow \text{Dec}_w^{\text{Red}} \times_{\mathbb{Y}} \text{RSSYT}$ : EG column insertion into decreasing tableaux, starting from the left end of the compatible pairs.

The four “reduced” bijections are essentially equivalent: EG row insertion and EG column insertion are merely transposes of each other. The relationships are summarized in the commutative diagram in Figure 1. There are two different analogues of Schützenberger’s evacuation involution being used here. The first is a weight-preserving bijection  $\text{SSYT} \xleftrightarrow{\text{ev}} \text{RSSYT}$ . It may be defined by jeu-de-taquin which repeatedly slide initial horizontal strips to the outside without relabeling values. The second is a map  $\text{Dec}_w \xleftrightarrow{\text{ev}} \text{Inc}_w$  that appears in [SY]. For the restriction  $\text{Dec}_w^{\text{Red}} \rightarrow \text{Inc}_w^{\text{Red}}$  it is the same as Thomas and Yong’s evacuation map based on their jeu-de-taquin for increasing tableaux, which they call Kevac [TY, §4].

**Example 1.1.** Consider the following element  $(a, i) \in \text{CP}^{\text{Red}}$ .

$i$	3	3	3	2	2	2	1	1
$a$	1	2	4	1	3	5	2	4

We have

$$\begin{aligned} \Phi_{IS}(a, i) &= \left( \begin{array}{|c|c|c|c|} \hline 1 & 2 & 3 & 4 \\ \hline 2 & 3 & 5 & \\ \hline 4 & & & \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 1 & 1 & 2 & 3 \\ \hline 2 & 2 & 3 & \\ \hline 3 & & & \\ \hline \end{array} \right) \\ \Phi_{IR}(a, i) &= \left( \begin{array}{|c|c|c|c|} \hline 1 & 2 & 3 & 4 \\ \hline 2 & 3 & 5 & \\ \hline 4 & & & \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 3 & 3 & 3 & 2 \\ \hline 2 & 2 & 1 & \\ \hline 1 & & & \\ \hline \end{array} \right) \\ \Phi_{DS}(a, i) &= \left( \begin{array}{|c|c|c|c|} \hline 5 & 4 & 2 & 1 \\ \hline 4 & 3 & 1 & \\ \hline 2 & & & \\ \hline \end{array}, \begin{array}{|c|c|c|c|} \hline 1 & 1 & 2 & 3 \\ \hline 2 & 2 & 3 & \\ \hline 3 & & & \\ \hline \end{array} \right) \end{aligned}$$

The map  $\text{ev} : \text{SSYT} \rightarrow \text{RSSYT}$  is defined as follows. For a  $T \in \text{SSYT}$ , there is a unique  $T' \in \text{RSSYT}$  such that  $\text{row}(T)$  and  $\text{rev}(\text{row}(T'))$  are Knuth equivalent, where  $\text{rev}(\cdot)$  is the operator that reverse a word. Then  $\text{ev}(T) := T'$ . The computation  $\text{ev} : \text{SSYT} \rightarrow \text{RSSYT}$  can be done by jeu-de-taquin as follows. Sliding out the 1's using the usual jeu-de-taquin we obtain

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

Then the 2's are slid out but not past the 1s.

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

The 3's need no moving. The result is

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

The following Proposition asserts that the lower triangle in Figure 1 commutes.

**Proposition 1.2.** [EG, Cor. 7.22] Let  $(a, i) \in \text{CP}_w^{\text{Red}}$  and  $\Phi_{IS}^{\text{Red}}(a, i) = (P, Q)$  and  $\Phi_{IR}^{\text{Red}}(a, i) = (P', Q')$ . Then  $P = P'$  and  $Q' = \text{ev}(Q)$  where  $\text{ev} : \text{SSYT} \rightarrow \text{RSSYT}$  is Schützenberger's evacuation involution (usual evacuation but without relabeling).

The upper triangle also commutes: it is the same statement but with the total order on values reversed.

The map  $\text{ev} : \text{Inc}_w^{\text{Red}} \rightarrow \text{Dec}_w^{\text{Red}}$  can be defined similarly. For a  $T \in \text{Inc}_w^{\text{Red}}$ , there is a unique  $T' \in \text{Dec}_w^{\text{Red}}$  such that  $\text{row}(T)$  and  $\text{rev}(\text{row}(T'))$  are Coxeter-Knuth equivalent. Then  $\text{ev}(T) := T'$ . Its computation is more involved.<sup>2</sup>

There is also a beautiful connection between the insertions into increasing versus decreasing tableaux. The following result says that the triangle on the right of Figure 1 is commutative.

**Proposition 1.3.** *Let  $(a, i) \in \text{CP}_w^{\text{Red}}$ ,  $\Phi_{IS}^{\text{Red}}(a, i) = (P, Q)$  and  $\Phi_{DS}^{\text{Red}}(a, i) = (P', Q')$ . Then  $Q = Q'$  and  $P' = \text{ev}(P)$  where  $\text{ev} : \text{Inc}_w^{\text{Red}} \rightarrow \text{Dec}_w^{\text{Red}}$  is the map  $P \mapsto P^b$ .*

*Proof.* The statement for  $Q$  tableaux is proved in [EG, Cor. 7.21]. By [EG, Thm. 6.24],  $\text{row}(P)$  and  $a$  are Coxeter-Knuth equivalent. On the other hand,  $\text{row}(P')$  and  $\text{rev}(a)$  are Coxeter-Knuth equivalent. Thus,  $\text{rev}(\text{row}(P'))$  and  $\text{row}(P)$  are Coxeter-Knuth equivalent, so  $P' = \text{ev}(P)$ .  $\square$

An analogous relationship exists between  $\Phi_{DS}$  and  $\Phi_{IR}$ .

**Remark 1.4.** In particular, for any fixed  $(a, i) \in \text{CP}_w^{\text{Red}}$ , upon applying any of the four EG bijections, the tableau pair has the same shape.

## 1.5 Solutions for general case

Hecke column insertion [BKSTY] defines a bijection  $\Phi_{IS} : \text{CP}_w \rightarrow \text{Inc}_w \times_{\mathbb{Y}} \text{SVT}_w$  whose restriction to  $\text{CP}_w^{\text{Red}}$  is EG column insertion. By merely reversing the total order on entries in tableaux, the resulting variant of Hecke column insertion gives a bijection  $\Phi_{DR} : \text{CP}_w \rightarrow \text{Dec}_w \times_{\mathbb{Y}} \text{RSVT}_w$ .

However, there are no known easy variations of the Hecke insertion which achieve the bijections  $\Phi_{IR}$  or  $\Phi_{DS}$ . Hecke row insertion is the variant of Hecke column insertion in which the roles of rows and columns are exchanged. Unfortunately, it cannot achieve this goal since it does not satisfy the relevant Pieri property.

This paper introduces a new insertion algorithm  $\Phi$  which gives an explicit weight-preserving bijection  $\Phi_{DS} : \text{CP}_w \rightarrow \text{Dec}_w \times_{\mathbb{Y}} \text{SVT}_w$ . Our algorithm is a row insertion which, like Hecke insertion, respects the Hecke equivalence relation  $\equiv_H$ . Our insertion possesses a different Pieri property than the one satisfied by Hecke row insertion; this is necessary to achieve set-valued recording tableaux. Moreover, when restricted to  $\text{CP}_w^{\text{Red}}$ , our algorithm recovers EG row insertion. A simple variation of our algorithm (reversing the total order on entries) gives a bijection  $\Phi_{IR} : \text{CP}_w \rightarrow \text{Inc}_w \times_{\mathbb{Y}} \text{RSVT}_w$ . Together with the variants of Hecke insertion, our insertion completes the picture in §1.3: we have produced the generalization of the four diagonal maps in Figure 1. Now the picture looks like Figure 2.

<sup>2</sup>We believe the shape-preserving map  $P \mapsto P^b$  of [SY, Def. 2.9] agrees with  $\text{ev}(\cdot)$  on  $\text{Inc}_w^{\text{Red}}$ . To prove this, one just needs to check that performing K-jeu-de-taquin on one box will preserve the Coxeter-Knuth class.

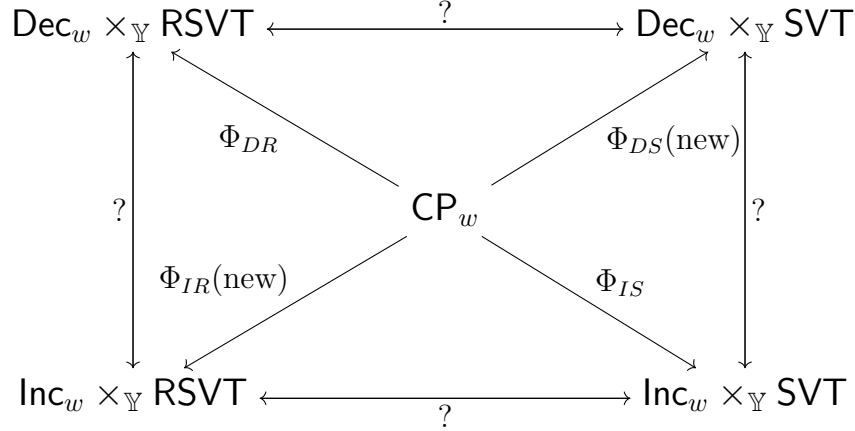


Figure 2: The four bijections

**Remark 1.5.** None of the coherence properties of Propositions 1.2 or 1.3 generalize to any of the bijections in the nonreduced setting. In general the four bijections produce 4 different groupings of compatible pairs for the various expansions of  $G_w$ . In addition we are not aware of any well-behaved map that can play the role of  $\text{ev} : \text{SVT} \rightarrow \text{RSVT}$ .

## 2 New reverse row insertion

### 2.1 Ejectable values in decreasing tableaux

To define the new reverse insertion algorithm on decreasing tableaux, we require the notion of an *ejectable* value in a decreasing tableau. This is defined recursively.

In this article English notation is used for partitions and tableaux. A tableau is *decreasing* if its entries strictly decrease from left to right along each row and strictly decrease from top to bottom in each column. For a decreasing tableau  $P$  let  $P_{>r}$  denote the decreasing tableau obtained by removing the first  $r$  rows of  $P$ . Let  $P_{\geq r} = P_{>(r-1)}$ .

**Definition 2.1.** Let  $P$  be a decreasing tableau. A value  $x$  is  *$P$ -ejectable* if  $x$  occurs in the first row of  $P$  and either  $x - 1$  is not in the first row of  $P$ , or  $x - 1$  is in the first row of  $P$  and  $x - 1$  is  $P_{>1}$ -ejectable.

**Example 2.2.** The value 3 is  $P$ -ejectable for the tableau  $P$  depicted below.

$$P = \begin{array}{|c|c|c|c|} \hline 7 & 6 & 3 & 2 \\ \hline 5 & 2 & 1 & \\ \hline 3 & 1 & & \\ \hline \end{array}$$

Since 3 and 2 both occur in the first row, 3 is  $P$ -ejectable if and only if 2 is  $P_{>1}$ -ejectable.

$$P_{>1} = \begin{array}{|c|c|c|} \hline 5 & 2 & 1 \\ \hline 3 & 1 & \\ \hline \end{array}$$

Since 2 and 1 occur in the first row of  $P_{>1}$ , 2 is  $P_{>1}$ -ejectable if and only if 1 is  $P_{>2}$ -ejectable.

$$P_{>2} = \begin{array}{|c|c|} \hline 3 & 1 \\ \hline \end{array}$$

Since the first row of  $P_{>2}$  has a 1 but no 0, 1 is  $P_{>2}$ -ejectable. Hence 3 is  $P$ -ejectable.

The value 7 is not  $P$ -ejectable because there is a 6 in the first row but not in the second.

## 2.2 Ejectable values and Hecke equivalence

The  $0$ -Hecke monoid is the quotient of the free monoid of words on the alphabet  $\mathbb{Z}_{>0}$  by the relations

$$\begin{aligned} ii &\equiv_H i \\ i(i+1)i &\equiv_H (i+1)i(i+1) \\ ij &\equiv_H ji \quad \text{for } |i-j| \geq 2. \end{aligned}$$

The minimum-length elements of each  $\equiv_H$  class are the reduced words of some permutation  $w$ , giving a canonical bijection between the  $\equiv_H$  classes and permutations of  $\mathbb{Z}_{>0}$  moving finitely many elements. We denote by  $[a]_H$  the permutation associated with the  $\equiv_H$  class of the word  $a$ .

The row-reading word  $\text{row}(P)$  of a tableau  $P$  is the word  $\dots u^{(2)}u^{(1)}$  where  $u^{(i)}$  is the word given by reading the  $i$ -th row of  $P$  from left to right.

**Lemma 2.3.** *Let  $P$  be a decreasing tableau. If  $x$  is an ejectable entry of  $P$  then  $\text{row}(P) \equiv_H \text{row}(P)x$ .*

*Proof.* This is proved by induction on the number of rows in  $P$ . Let  $w$  be the decreasing word given by the first row of  $P$  and let  $R$  be the set of letters in  $w$ . By definition  $\text{row}(P) = \text{row}(P_{>1})w$ . It suffices to show that

$$\text{row}(P_{>1})w \equiv_H \text{row}(P_{>1})wx. \tag{8}$$

If  $x \in R$  and  $x-1 \notin R$  then  $w \equiv_H wx$  and hence (8) holds. Otherwise  $x, x-1 \in R$  and the  $x-1$  is ejectable in  $P_{>1}$ . By the inductive hypothesis,  $\text{row}(P_{>1}) \equiv_H \text{row}(P_{>1})(x-1)$ . In this case  $(x-1)w \equiv_H wx$  and

$$\text{row}(P_{>1})wx \equiv_H \text{row}(P_{>1})(x-1)w \equiv_H \text{row}(P_{>1})w$$

and again (8) holds as required.  $\square$



### 2.3 Bumping paths

Let  $D(\lambda) = \{(i, j) \in \mathbb{Z}_{>0}^2 \mid i \geq 1, j \leq \lambda_i\}$  be the diagram of the partition  $\lambda$ . The elements of  $D(\lambda)$  are called the *cells* of  $\lambda$  and have a matrix-style indexing: the cell  $(i, j)$  is depicted as a box in the  $i$ -th row and  $j$ -th column. For a partition  $\lambda$ , a  $\lambda$ -removable cell is one that is at the end of its row and bottom of its column. For a tableau  $P$ , a  $P$ -removable cell is a  $\lambda$ -removable cell where  $\lambda$  is the shape of  $P$ .

**Definition 2.4.** Let  $(r, c)$  be a removable cell for the decreasing tableau  $P$ . The (reverse) *bumping path* of  $(r, c)$  in  $P$  is the following sequence of numbers  $m_r < m_{r-1} < \dots < m_1$  together with their positions in  $P$ . Let  $m_r$  be the value of  $P$  in  $(r, c)$ . With the entry  $m_{i+1}$  in row  $i + 1$  defined, let  $m_i$  be the smallest number in row  $i$  such that  $m_{i+1} < m_i$ .

**Example 2.5.** A decreasing tableau and the bumping path for its removable cell  $(3, 2)$  are pictured below.

8	7	6
5	4	2
3	2	
1		

In the example above, notice that the column index is weakly increasing, as you go up in bumping path. This is true in general.

**Lemma 2.6.** *Let  $m_r < m_{r-1} < \dots < m_1$  be a bumping path in  $P$ . For  $r \geq j > i \geq 1$ , the  $m_i$  in row  $i$  of  $P$  is weakly right of the  $m_j$  in row  $j$  of  $P$ .*

*Proof.* Only need to prove this claim for  $j = i + 1$ . Let  $y$  be the number immediately above the  $m_{i+1}$  in row  $i + 1$  of  $P$ . We have  $y > m_{i+1}$ . Thus,  $y \geq m_i$ . The  $m_i$  in row  $i$  is weakly right of the  $y$  in this row, which implies our claim.  $\square$

The element in the first row of any bumping path is ejectable.

**Lemma 2.7.** *Let  $m_r < \dots < m_1$  be the bumping path of a removable cell of  $P$ . Then  $m_1$  is ejectable in  $P$ .*

*Proof.* The proof proceeds by induction on the number of rows in  $P$ . Let  $R$  be first row of  $P$ . If  $m_1 - 1 \notin R$  then  $m_1$  is ejectable in  $P$ . Otherwise  $m_1, m_1 - 1 \in R$ . Since  $m_1$  is the smallest in  $R$  such that  $m_1 > m_2$ , it follows that  $m_2 = m_1 - 1$ . It suffices to show that  $m_1 - 1 = m_2$  is ejectable in  $P_{>1}$ . This follows from the inductive hypothesis since  $m_r < \dots < m_2$  is the bumping path of a removable cell in  $P_{>1}$ .  $\square$

### 2.4 New reverse insertion

The reverse insertion algorithm is a map  $\Psi$

$$(P, s, \alpha) \mapsto (P', m)$$

where the input triple consists of a decreasing tableau  $P$ , a  $P$ -removable cell  $s = (r, c)$ , and  $\alpha \in \{0, 1\}$ . The output pair consists of a decreasing tableau  $P'$  and  $m \in \mathbb{Z}_{>0}$  such that

$$\text{shape}(P') = \begin{cases} \text{shape}(P) & \text{if } \alpha = 0 \\ \text{shape}(P) - \{s\} & \text{if } \alpha = 1. \end{cases} \quad (9)$$

For conceptual clarity we precompute the bumping path in  $P$  starting at  $(r, c)$ . For  $1 \leq i \leq r$  let  $m_i$  denote the entry in the  $i$ -th row of the bumping path. The output value  $m$  is by definition the value  $m_1$  in the first row of the bumping path.

The output tableau  $P'$  will only differ from  $P$  along the bumping path. It is only necessary to specify whether each  $m_i$  on the bumping path gets replaced, and if so, by what value.<sup>3</sup> This decision is determined iteratively by decreasing  $i$  based on the values  $m_i$  and  $m_{i+1}$ , the  $i$ -th row of  $P$ , the subtableau  $P'_{>i}$ , and a status indicator  $\alpha_{i+1} \in \{0, 1\}$ . The  $i$ -th iteration updates the  $i$ -th row of  $P$  (which becomes the  $i$ -th row of  $P'$ ) and produces  $\alpha_i \in \{0, 1\}$ .

Let  $P'$  be a working tableau which is initialized to  $P$ . In the initialization step, if  $\alpha = 1$ , remove from  $P'$  the removable cell in row  $r$  and its contents  $m_r$  and set  $\alpha_r = 1$  and  $i = r - 1$ . If  $\alpha = 0$  set  $m_{r+1} = 0$ ,  $\alpha_{r+1} = 0$  and  $i = r$ .

The algorithm enters a loop. If  $i = 0$  the algorithm terminates and the current tableau  $P'$  is the output tableau. Now assume  $i \geq 1$ . Let  $R$  be the set consisting of numbers in row  $i$  of the current tableau  $P'$  (or equivalently  $P$ , since  $P$  and  $P'$  differ only in rows of index greater than  $i$ ). By definition  $m_i \in R$ .

There are several cases. We give each a nickname and mnemonic.

- **Dummy (D)**: If  $m_i - 1 \in R$  (which implies  $m_{i+1} = m_i - 1$ ) do not change the  $i$ -th row and set  $\alpha_i = \alpha_{i+1}$ .
- **Direct Replacement (DR)**: Otherwise if  $\alpha_{i+1} = 1$  and  $m_{i+1} \notin R$ , replace  $m_i$  by  $m_{i+1}$  in row  $i$  of  $P'$  and set  $\alpha_i = 1$ .

Suppose neither of the two above cases hold. Find the smallest ejectable entry  $x$  in  $P'_{>i}$  such that  $m_i > x > m_{i+1}$ .

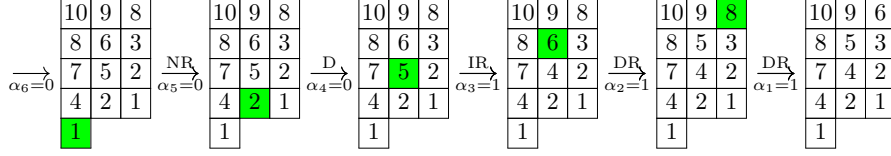
- **Indirect Replacement (IR)**: Suppose  $x$  exists. Replace  $m_i$  by  $x$  in row  $i$  of  $P'$  and set  $\alpha_i = 1$ .
- **No Replacement (NR)**: Suppose  $x$  does not exist. Do not change the  $i$ -th row and set  $\alpha_i = 0$ .

Now decrement  $i$  and go to the top of the loop.

**Example 2.8.** In the following example, the input parameters are  $s = (5, 1)$  and  $\alpha = 0$ . To initialize, set  $(m_6, m_5, m_4, m_3, m_2, m_1) = (0, 1, 2, 5, 6, 8)$ ,  $i = 5$ ,

<sup>3</sup>If replacement occurs the replacement value comes from the row below. However, unlike most insertion algorithms, the replacement value need not come from the bumping path.

and  $\alpha_6 = 0$ . The shaded box in the  $i$ -th row indicates the value  $m_i$ . The label on the arrow leaving this tableau is the mnemonic for the case of  $\Psi$ .



### 3 Properties of the reverse insertion

In this section the reverse insertion map  $\Psi$  is shown to be well-defined and some of its properties are established.

**Lemma 3.1.** *For  $r \geq i \geq 1$ ,  $\alpha_i = 0$  if and only if  $m_i$  is ejectable in  $P'_{\geq i}$ .*

*Proof.* Note that after the  $i$ -th row is processed, the subtableau  $P'_{\geq i}$  remains the same thereafter: only bumping path entries in rows above may be changed.

The proof proceeds by descending induction on  $i$ . For the initial step, if  $\alpha = 1$ , the algorithm sets  $\alpha_r = 1$  and  $m_r$  gets removed and is therefore absent from the  $r$ -th row of  $P'$ . Thus,  $m_r$  is not ejectable in  $P'_{\geq r}$ . If  $\alpha = 0$ , during the first iteration,  $m_r$  is replaced in the  $r$ -th row of  $P'$  if and only if  $\alpha_r = 1$ . Hence our claim holds for  $i = r$ .

Now suppose the claim holds for row  $i + 1$ . In the Dummy case,  $m_{i+1} = m_i - 1$  and  $m_i$  is not replaced. Thus  $m_i$  is ejectable in  $P'_{\geq i}$  if and only if the entry  $m_i - 1 = m_{i+1}$  is ejectable in  $P'_{> i}$  (by definition of ejectable) if and only if  $\alpha_{i+1} = 0$  (by induction) if and only if  $\alpha_i = 0$  (since in the Dummy case  $\alpha_i = \alpha_{i+1}$ ). Otherwise suppose the Dummy case does not hold. Then  $m_i$  and  $m_i - 1$  cannot both live in row  $i$  of  $P'$ . Thus  $m_i$  is ejectable in  $P'_{\geq i}$  if and only if it is not removed from row  $i$ . This happens only in the No Replacement case and  $\alpha_i = 0$  only in that case. Thus our claim holds for row  $i$  as required.  $\square$

**Theorem 3.2.** *The reverse insertion is a well-defined map.*

*Proof.* It must be shown that the output tableau  $P'$  is a decreasing tableau. We show  $P'$  is a decreasing tableau after each iteration of the algorithm.

During initialization, if  $\alpha = 0$ , the iteration for  $i = r$  either leaves  $P'$  unchanged or replaces  $m_r$  in the removable cell  $(r, c)$  by a smaller number. If  $\alpha = 1$ , after the initialization step  $P'$  is a decreasing tableau, since it is obtained from a decreasing tableau by removing a corner entry. In either case  $P'$  is a decreasing tableau before the  $i = r - 1$  iteration.

Suppose  $P'$  is a decreasing tableau after the iteration for row  $i + 1$ . It is enough to check that  $P'$  is still a decreasing tableau after the iteration for row  $i$ . In the Dummy or No Replacement cases there is nothing to check. In the two remaining cases, the number  $m_i$  is replaced by a smaller number. We need to make sure this smaller number is larger than all numbers on its right and under it.

Consider  $P'$  before this iteration. By the definition of a bumping path, numbers on the right of the  $m_i$  in row  $i$  are at most  $m_{i+1}$ . By Lemma 2.6, numbers below this  $m_i$  are also at most  $m_{i+1}$ . Next, we consider the two cases.

- Direct Replacement:  $m_i$  is replaced by  $m_{i+1}$ . We need to make sure  $m_{i+1}$  is not on the right or under this  $m_i$  in  $P'$  before this iteration. First,  $m_{i+1}$  cannot be in row  $i$  by the condition of this case. Now assume toward contradiction that  $m_{i+1}$  is immediately below  $m_i$ . This part of  $P'$  looks like

$m_i$	$a$
$m_{i+1}$	$b$

Since  $\alpha_{i+1} = 1$ ,  $m_{i+1}$  is not ejectable in  $P'_{\geq i+1}$ . Thus,  $b = m_{i+1} - 1$ . Then  $a > m_{i+1} - 1$  and  $m_{i+1} \geq a$ , so  $a = m_{i+1}$ . Since  $m_{i+1}$  is not in row  $i$  a contradiction is reached. Thus, after replacing the  $m_i$  by  $m_{i+1}$ ,  $P'$  is still a decreasing tableau.

- Indirect Replacement:  $m_i$  is replaced by  $x$ . We know  $x > m_{i+1}$ . After replacing  $m_i$  by  $x$ ,  $P'$  is still a decreasing tableau, since numbers on its right and under it are at most  $m_{i+1}$ .

□

The reverse insertion respects Hecke equivalence.

**Lemma 3.3.** *Let  $\Psi(P, (r, c), \alpha) = (P', m)$ . Then  $\text{row}(P) \equiv_H \text{row}(P')m$ .*

*Proof.* Let  $w$  be the decreasing word given by the first row of  $P$  and let  $R$  be the set of letters in  $w$ . Define  $R'$  and  $w'$  similarly for  $P'$ . Notice that  $\text{row}(P) = \text{row}(P_{>1})w$  and  $\text{row}(P') = \text{row}(P'_{>1})w'$ . It suffices to show that

$$\text{row}(P_{>1})w \equiv_H \text{row}(P'_{>1})w'm. \quad (10)$$

The proof proceeds by induction on  $r$ , the row index of the entry in the input of  $\Psi$ . The base case is  $r = 1$ . In this case  $\text{row}(P_{>1}) = \text{row}(P'_{>1})$ . If  $\alpha = 1$ , then  $w = w'm$ . Otherwise, in the first iteration, the algorithm searches for the smallest ejectable  $x < m$  in  $P_{>1}$ . If  $x$  does not exist then  $w \equiv_H wm = w'm$ . Otherwise  $w'$  is obtained by changing  $m$  in  $w$  into  $x$ . We see that (10) holds:

$$\text{row}(P_{>1})w \equiv_H \text{row}(P_{>1})xw \equiv_H \text{row}(P_{>1})w'm.$$

For the inductive step let  $r > 1$ . Before the last iteration the algorithm behaves as if doing  $\Psi$  on  $(P_{>1}, (r-1, c), \alpha)$ . By the definition of  $\Psi$  the result is  $(P'_{>1}, m_2)$ . By the inductive hypothesis,  $\text{row}(P_{>1}) \equiv_H \text{row}(P'_{>1})m_2$ . It is enough to check

$$\text{row}(P'_{>1})m_2w \equiv_H \text{row}(P'_{>1})w'm.$$

Consider the first two cases of the last iteration.

- **Dummy:** In this case,  $m, m-1 \in R$  and  $m_2 = m-1$ . We have  $m_2 w \equiv_H w m = w' m$ .
- **Direct Replacement:** In this case,  $m-1, m_2 \notin R$ . We know  $w$  is obtained from  $w'$  by changing  $m$  into  $m_2$ . We have  $m_2 w \equiv_H w' m$ .

We may assume the above two cases do not hold. Then either  $m_2 \in R$  or  $\alpha_2 = 0$  ( $m_2$  is ejectable in  $P'_{>1}$ ). In either case we claim  $\text{row}(P'_{>1})m_2 w \equiv_H \text{row}(P'_{>1})w$ : If  $m_2 \in R$ , then  $m_2 w \equiv_H w$  since  $m_2 + 1$  is not in  $w$  and  $m_2$  is in  $w$ . If the  $m_2$  is ejectable in  $P'_{>1}$  then  $\text{row}(P'_{>1})m_2 \equiv_H \text{row}(P'_{>1})$  by Lemma 2.3.

With this claim, it must be shown that

$$\text{row}(P'_{>1})w \equiv_H \text{row}(P'_{>1})w' m.$$

It must be verified that this holds in the remaining two cases:

- **Indirect Replacement:** Since  $x$  is ejectable in  $P'_{>1}$ ,  $\text{row}(P'_{>1})x \equiv_H \text{row}(P'_{>1})w'$ .  $w'$  is obtained by changing  $m$  to  $x$  in  $w$ . Thus  $xw \equiv_H w' m$ .
- **No Replacement:** We have  $w \equiv_H w m = w' m$ .

□

Our reverse row insertion satisfies the following Pieri condition, which is not satisfied by Hecke reverse row insertion.

**Lemma 3.4.** *Let  $\alpha, \alpha' \in \{0, 1\}$ ,  $P$  a decreasing tableau with removable corner  $(r_1, c_1)$ ,  $\Psi(P, (r_1, c_1), \alpha) = (P', m)$ , and  $\Psi(P', (r_2, c_2), \alpha') = (P'', m')$  with  $(r_2, c_2)$  a removable corner of  $P'$  with  $c_2 < c_1$ . Then  $m' > m$ .*

*Proof.* Let  $m_{r_1} < \dots < m_1$  be the bumping path for  $\Psi$  on  $(P, (r_1, c_1), \alpha)$  and  $n_{r_2} < \dots < n_1$  the bumping path for  $\Psi$  on  $(P', (r_2, c_2), \alpha')$ . By definition  $m_1 = m$  and  $n_1 = m'$  so it is enough to show that  $n_1 > m_1$ .

When  $m_i$  is ejected from  $P'_{\geq i}$ ,  $n_i$  is a number in the top row of  $P'_{\geq i}$ . Moreover, since  $n_i$  is the last number in a bumping path in  $P'_{\geq i}$ ,  $n_i$  is ejectable in  $P'_{\geq i}$ . We check  $n_i > m_i$  for all  $1 \leq i \leq r_1$  by descending induction on  $i$ ; in the case  $\alpha = 0$  the initial index is  $i = r_1 + 1$ .

For the base case consider the value of  $\alpha$ . If  $\alpha = 1$ ,  $m_{r_1}$  is removed from row  $r_1$  and ejected. Clearly  $n_{r_1} > m_{r_1}$ . If  $\alpha = 0$ ,  $m_{r_1+1} = 0 < n_{r_1+1}$ .

By induction we assume that  $m_{i+1} < n_{i+1}$ . We consider the cases of the two reverse insertions when they process row  $i$ .

- **(Dummy case):** In this case,  $m_i = m_{i+1} + 1$ . We have  $n_i > n_{i+1} \geq m_i$ .
- **(Direct Replacement case):** In this case, we replace  $m_i$  by  $m_{i+1}$  in row  $i$ . Then  $n_i$  is a number in row  $i$  of  $P'$ , and since  $n_i > n_{i+1} > m_{i+1}$  it must be to the left of  $m_{i+1}$ . Thus  $n_i$  is to the left of  $m_i$  in row  $i$  of  $P$ . We conclude that  $n_i > m_i$ .

- (Indirect Replacement case): In this case, we replace  $m_i$  by  $x$  on row  $i$ . Since  $n_{i+1} > m_{i+1}$  and  $n_{i+1}$  is ejectable in  $P'_{>i}$ ,  $n_{i+1} \geq x$  by the choice of  $x$ . Thus  $n_i$  is a number in row  $i$  of  $P'$ , and since  $n_i > n_{i+1} \geq x$ , it must be to the left of  $x$ . Similar to the previous case,  $n_i > m_i$ .
- (No Replacement case): In this case, there is no  $x$  that is ejectable in  $P'_{>i+1}$  and  $m_{i+1} < x < m_i$ . Since  $n_{i+1}$  is ejectable in  $P'_{\geq i+1}$ ,  $n_{i+1} \geq m_i$ . Thus  $n_i > m_i$ .

□

The reverse insertion algorithm is a generalization of EG reverse insertion.

**Lemma 3.5.** *Let  $P$  be a decreasing tableau such that  $\text{row}(P)$  is reduced. Let  $\Psi(P, (r, c), 1) = (P', m)$ . Then we also get  $(P', m)$  if we apply EG reverse row insertion at  $(r, c)$  in  $P$ .*

*Proof.* Since the Dummy and Direct Replacement cases agree with EG reverse insertion, it is enough to show that during each iteration, one of these cases must apply.

For  $\alpha = 1$  the initial step agrees with reverse EG insertion. By induction we assume  $\alpha_{i+1} = 1$ . We will assume the iteration for row  $i$  is not in the Dummy nor the Direct Replacement cases and reach a contradiction. Let  $R$  be the  $i$ -th row of  $P$ . We assume  $m_i - 1 \notin R$  and  $m_{i+1} \in R$ . By the minimality of  $m_i$ ,  $m_{i+1} + 1 \notin R$ . Let  $w$  be the row word of the first  $i$  rows of  $P$ . We have  $m_{i+1}w \equiv_H w$ . Then notice that

$$\text{row}(P) = \text{row}(P_{>i})w \equiv_H \text{row}(P'_{>i})m_{i+1}w \equiv_H \text{row}(P'_{>i})w.$$

Then  $\text{row}(P)$  is not reduced and we obtain the required contradiction. □

## 4 The insertion

This section gives a direct description of the inverse of  $\Psi$ , an insertion algorithm  $\Phi$  which “inserts  $m$  into  $P$ ”:

$$(P, m) \mapsto (P', s, \alpha)$$

where the input pair consists of a decreasing tableau  $P$  and  $m \in \mathbb{Z}_{>0}$ , and the output triple consists of a decreasing tableau  $P'$ , a removable cell  $s = (r, c)$  of  $P'$ , and  $\alpha \in \{0, 1\}$  such that the following holds:

$$\text{shape}(P') = \begin{cases} \text{shape}(P) & \text{if } \alpha = 0 \\ \text{shape}(P) \cup \{s\} & \text{if } \alpha = 1. \end{cases} \quad (11)$$

The working tableau  $P'$  has initial value  $P$ . The  $i$ -th iteration consists of an insertion of a number  $N \in \mathbb{Z}_{>0}$  into  $P'_{\geq i}$ . At this point  $P'_{\geq i} = P_{\geq i}$ ; only values in rows before the  $i$ -th have been changed. Let  $R$  be the set consisting of numbers in row  $i$  of  $P$ . Find the largest  $n_1 \in R$  such that  $n_1 \leq N$ .

- **Terminating case 1 (T1):** If  $n_1$  does not exist, put  $N$  at the end of row  $i$  in  $P'$  and terminate the algorithm. The output  $P'$  is the current tableau. The output  $(r, c)$  is the coordinate of this newly added  $N$ . Set  $\alpha = 1$ .

Otherwise  $n_1$  exists. Change the  $n_1$  in row  $i$  of  $P'$  into  $N$ .

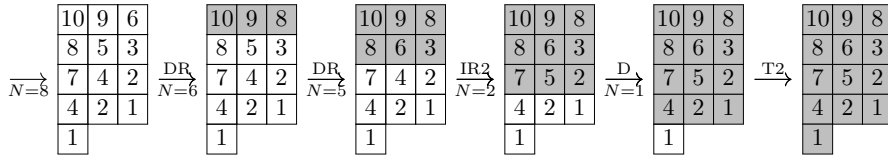
- **Dummy case (D):** If  $n_1 = N$  and  $N - 1 \in R$ : insert  $N - 1$  into  $P'_{>i}$ .
- **Direct Replacement case (DR):** If  $n_1 < N$  and  $n_1$  is not ejectable in  $P'_{>i}$ : insert  $n_1$  into  $P'_{>i}$ .

Otherwise assume none of the above cases hold. Let  $n_2$  be the number to the right of  $n_1$  in row  $i$  of  $P$ , or  $n_2 = 0$  if  $n_1$  is the rightmost number in this row. Find the largest ejectable  $y$  in  $P'_{>i}$  such that  $n_1 > y > n_2$ .

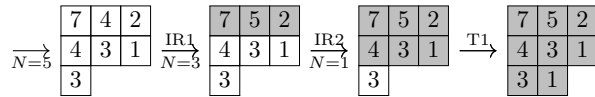
- **Indirect Replacement case 1 (IR1):** If  $y$  exists: insert  $y$  into  $P'_{>i}$ .
- **Indirect Replacement case 2 (IR2):** If  $y$  does not exist and  $n_2 > 0$ : insert  $n_2$  into  $P'_{>i}$ .
- **Terminating case 2 (T2):** If  $y$  does not exist and  $n_2 = 0$ : terminate the algorithm. The output  $P'$  is the current tableau. The output  $(r, c)$  is the coordinate of this  $N$  in row  $i$  of  $P'$ .

Set  $\alpha = 0$ .

**Example 4.1.** In the following example, we let  $P$  be the leftmost tableau and insert  $m = 8$  into  $P$ . The output is the rightmost tableau  $P'$ ,  $s = (5, 1)$ , and  $\alpha = 0$ . The unshaded part of each tableau is the part being considered by the insertion in each step.



**Example 4.2.** In the following example, we insert  $m = 5$  into the leftmost tableau  $P$ . The output is the rightmost tableau  $P'$ ,  $s = (3, 2)$ , and  $\alpha = 1$ .



## 5 Properties of the insertion

In this section the well-definedness of the insertion algorithm  $\Phi$  is established and some of its properties are studied.

**Lemma 5.1.** *Consider an iteration of  $\Phi$  in which  $N$  is being inserted into  $P'_{\geq i}$  in Indirect Replacement case 2. Consider the value  $n_1$  in row  $i$  of  $P$ . If there is a number below this  $n_1$ , it must be at most  $n_2$ .*

*Proof.* Let  $t_1$  be the number below this  $n_1$ . This part of  $P$  looks like

$$\begin{array}{|c|c|} \hline n_1 & n_2 \\ \hline t_1 & t_2 \\ \hline \end{array}$$

Now assume toward contradiction that  $t_1 > n_2$ . The number  $t_2$  either does not exist or we have  $t_2 < n_2 \leq t_1 - 1$ . In either case,  $t_1$  is ejectable in  $P'_{> i}$ . By  $n_1 > t_1 > n_2$ , we should go to Indirect Replacement case 1. Contradiction.  $\square$

**Lemma 5.2.** *The insertion algorithm is well-defined.*

*Proof.* The algorithm initializes the working tableau to equal  $P$  which is decreasing. To show the output tableau is decreasing it suffices to assume that before any particular iteration the working tableau is decreasing and show that after that iteration, the resulting tableau is decreasing.

Let  $P'$  be the working tableau at the beginning of the current iteration, in which  $N$  is being inserted into  $P'_{\geq i}$ . Let  $P''$  be the working tableau after this iteration. During the iteration, in row  $i$  the number  $n_1$  is replaced by  $N$  or  $N$  is appended at the end; let  $(i, j')$  be the position of this  $N$ . After this iteration, the row will clearly be strictly decreasing. We may assume  $i > 1$  and must show that there is a number  $M$  in position  $(i - 1, j')$  of  $P''$  and it satisfies  $M > N$ .

If  $n_1 = N$ , then we are done since this iteration does not change the working tableau at all. We assume  $n_1 < N$ , so the previous iteration is not in the Indirect Replacement case 1. Consequently,  $N$  is in row  $i - 1$  of  $P$ , say at  $(i - 1, j)$ . We have  $j' \leq j$  by the choice of  $n_1$ . In particular there is a number  $M$  in position  $(i - 1, j')$  of  $P''$ . It remains to show that  $M > N$ .

If  $j' < j$  then we obtain the required inequality  $M > N$  since the  $(i - 1)$ -th row was strictly decreasing before the previous iteration. So we may assume  $j' = j$ .

We consider the cases of the previous iteration:

- Dummy case.  $N + 1$  and  $N$  are in row  $i - 1$  of  $P$ . Below this  $N + 1$ , we have a number at most  $N$ , so  $j' \leq j - 1$ , contradiction.
- Direct Replacement case. During the previous iteration, the  $N$  in cell  $(i - 1, j)$  is replaced by a larger number. Thus, there is an  $M > N$  at  $(i - 1, j)$  of  $P''$ .
- Indirect Replacement case 2. By the lemma above, the  $n_1$  is in the first  $j - 1$  columns, a contradiction.



Since the row  $i - 1$  iteration was not terminal and we ruled out Indirect Replacement case 1, all cases are covered.  $\square$

**Theorem 5.3.**  $\Phi$  and  $\Psi$  are mutually inverse functions.

This theorem is implied by the following two lemmata.

**Lemma 5.4.** Let  $\Psi(P, (r, c), \alpha) = (P', m)$ . Then  $\Phi(P', m) = (P, (r, c), \alpha)$ .

*Proof.* Let  $R$  (resp.  $R'$ ) consist of the numbers in row 1 of  $P$  (resp.  $P'$ ). The proof proceeds by induction on  $r$ .

For the base case, assume  $r = 1$ . If  $\alpha = 1$ ,  $m = \min(R)$  and  $R' = R - \{m\}$ . When we insert  $m$  into  $P'$ , the first iteration is in Terminating case 1. We will just append  $m$  at the end of row 1 and terminate at this cell. If  $\alpha = 0$ , we study the cases of the only iteration in the reverse insertion:

- Indirect Replacement case: In this case,  $R' = R - \{m\} \sqcup \{x\}$  where  $x < m$  and  $x$  is the smallest number in row 2 of  $P$ . When we insert  $m$  into  $P'$ , it sets  $n_1 = x$ . Since  $n_1$  is ejectable in  $P'_{>1}$ , it does not go to the first 3 cases. Then we have  $n_2 = 0$ . There are no ejectable numbers in  $P'_{>1}$  between  $n_2$  and  $n_1$ . Thus, it goes to the Terminating case 2. It replaces  $x$  by  $m$  and ends at this cell with  $\alpha = 0$ .
- No Replacement case: In this case,  $R = R'$  and there are no ejectable numbers in  $P'_{>1}$  that are less than  $m$ . When we insert  $m$  into  $P'$ , it sets  $n_1 = m$  and  $n_2 = 0$ . Thus, it goes to Terminating case 2. It replaces  $m$  by  $m$  and ends at this cell with  $\alpha = 0$ .

Now assume  $r > 1$ . Consider the reverse insertion. Before the last iteration, a number  $m_2 > 0$  is ejected from  $P'_{>1}$ . During the last iteration, it changes at most one number in  $R$  and get  $R'$ . Then it ejects  $m$ . By induction it suffices to show that when  $m$  is inserted into  $P'$ , the first iteration of insertion changes  $R'$  back to  $R$  and inserts  $m_2$  into  $P'_{>1}$ . Let us do a case study on the last iteration of the reverse insertion.

- Dummy case:  $m_2 = m - 1$  and  $m_2, m \in R$ . The algorithm fixes row 1 of  $P$  so  $R = R'$ . The first iteration of insertion goes to the Dummy case. It fixes row 1 and inserts  $m - 1 = m_2$  into  $P'_{>1}$ .
- Direct Replacement case:  $m_2$  was ejected with  $\alpha_2 = 1$ . Thus  $m_2$  is not ejectable in  $P'_{>1}$ . The algorithm replaces  $m$  by  $m_2$ . The first iteration of insertion sets  $n_1 = m_2$ . It goes to the Direct Replacement case:  $m_2$  is replaced by  $m$  and  $m_2$  is inserted.
- Indirect Replacement case:  $m$  is replaced by  $x$  which is ejectable in  $P'_{>1}$ . By the choice of  $x$  there are no ejectable numbers in  $P'_{>1}$  between  $m_2$  and  $x$ . The first iteration of insertion sets  $n_1 = x$  and replaces it by  $m$ . It will not go to the first 3 cases. Since  $m$  is the smallest number in  $R$  that is larger than  $m_2$ ,  $m_2 \geq n_2$ . If  $m_2 > n_2$  then  $\alpha_2 = 0$ . Thus  $m_2$  is ejectable

in  $P'_{>1}$ . During the subsequent insertion, the algorithm sets  $y = m_2$  and inserts  $m_2$ . Now assume  $m_2 = n_2$ . Then the first iteration of the insertion cannot find such a  $y$ . It inserts  $n_2 = m_2$ .

- No Replacement case:  $R = R'$ . There are no ejectable numbers in  $P'_{>1}$  between  $m_2$  and  $m$ . During the insertion, the algorithm sets  $n_1 = m$  and will not go to the first three cases. The proof proceeds as in the Indirect Replacement case.

□

**Lemma 5.5.** *Let  $\Phi(P, m) = (P', (r, c), \alpha)$ . Then  $\Psi(P', (r, c), \alpha) = (P, m)$ .*

*Proof.* Let  $R$  (resp.  $R'$ ) consist of the numbers in row 1 of  $P$  (resp.  $P'$ ). The proof proceeds by induction on  $r$ .

In the base case  $r = 1$ , the insertion has only one iteration. If  $\alpha = 1$ , this iteration is in Terminating case 1. It appends  $m$  at the end of row 1. During the subsequent reverse insertion,  $m$  will be removed from row 1 and ejected. Now assume  $\alpha = 0$ . If  $n_1 = m$ , then the insertion leaves row 1 unchanged. There are no ejectable numbers in  $P_{>1}$  that are less than  $m$ . During the reverse insertion, the sole iteration goes to the No Replacement case: row 1 is unchanged and  $m$  is ejected. If  $n_1 < m$ , then the insertion replaces  $n_1$  by  $m$ . Since it is in the Terminating case 2,  $n_1$  is smallest ejectable number in  $P_{>1}$ . During the reverse insertion, the only iteration goes to the Indirect Replacement case: the  $m$  is changed to  $n_1$  and  $m$  is ejected.

Now assume  $r > 1$ . Consider the insertion. During the first iteration, it changes  $n_1$  into  $m$  in row 1. Then it inserts a number into  $P'_{>1}$ . Let  $z$  be that number. Now consider the reverse insertion. By our inductive hypothesis, before the last iteration,  $z$  is ejected under row 1 of the tableau. Moreover, currently the tableau below row 1 is identical to  $P_{>1}$ . We need to make sure the last iteration changes  $R'$  back to  $R$  and ejects  $m$ . Let us do a case study on the first iteration of the insertion.

- Dummy case:  $m, m - 1 \in R$ ,  $R = R'$ , and  $z = m - 1$ . The last iteration of the reverse insertion goes to the Dummy case: it fixes the first row and ejects  $m$ .
- Direct replacement case:  $n_1$  is changed to  $m$  and  $z = n_1 < m$ . Moreover  $z$  is not ejectable in  $P_{>1}$ . When  $z$  is ejected from  $P_{>1}$ ,  $\alpha_2 = 1$ . The last iteration of the reverse insertion goes to the Direct Replacement case: It changes  $m$  into  $n_1$  and ejects  $m$ .
- Indirect Replacement case 1:  $n_1$  is changed to  $m$  and  $z = y$ .  $y$  is the largest ejectable number in  $P_{>1}$  less than  $n_1$ . Moreover  $y > n_2$ . Consider the last iteration of the reverse insertion. Before this iteration, by induction and Lemma 3.1  $y$  is ejected from  $P_{>1}$  with  $\alpha_2 = 0$ . Then it sets  $m_1 = m$ . It looks for  $x$ , which is the smallest ejectable number in  $P_{>1}$  between  $y$  and  $m$ . If  $n_1 = m$ , then it goes to the No Replacement case: Row 1 is fixed

and  $m$  is ejected. If  $n_1 < m$  then  $n_1$  must be ejectable in  $P_{>1}$  and  $x = n_1$ . It goes to the Indirect Replacement case:  $m$  is replaced by  $n_1$  and  $m$  is ejected.

- Indirect Replacement case 2:  $n_1$  is replaced by  $m$  and  $z = n_2 > 0$ . There are no ejectable numbers between  $n_2$  and  $n_1$  in  $P_{>1}$ . Consider the last iteration of the reverse insertion. It sets  $m_1 = m$ . Since  $n_2$  is already in row 1, it must go to the last two cases. If  $n_1 = m$ , then it goes to the No Replacement case: The first row is fixed and  $m$  is ejected. If  $n_1 < m$ , then  $n_1$  must be ejectable in  $P_{>1}$ . It goes to the Indirect Replacement case:  $m$  is replaced by  $n_1$  and  $m$  is ejected.

□

Our insertion satisfies a Pieri property.

**Lemma 5.6.** *Let  $\Phi(P, m) = (P', (r_1, c_1), \alpha)$  and  $\Phi(P', m') = (P'', (r_2, c_2), \alpha')$ . If  $m' < m$ , then  $c_1 < c_2$ .*

*Proof.* Let  $m_{r_1} < \dots < m_1$  be the bumping path of  $(r_1, c_1)$  in  $P'$ . By the definition of  $\Psi$  on  $(P', (r_1, c_1), \alpha)$ , the output value is  $m_1$ . Since  $\Psi$  is inverse to  $\Phi$ ,  $\Psi(P', (r_1, c_1), \alpha) = (P, m)$ . Thus  $m_1 = m > m'$ . If  $\Phi$  on  $(P', m')$  ends in the first iteration, we are done. Otherwise, after this iteration, another number is inserted into  $P'_{>1}$ . It is enough to ensure that this number is smaller than  $m_2$ . During this iteration,  $\Phi$  finds a number  $n_1$  in row 1 of  $P'$ . We have  $n_1 \leq m' < m_1$ . Since  $m_{r_1} < \dots < m_1$  is a bumping path,  $m_2 \geq n_1$ . Now consider the case of the first iteration.

- Dummy case: The number  $m' - 1$  is inserted into  $P'_{>1}$ . We have  $m_2 \geq n_1 = m' > m' - 1$ .
- Direct Replacement case: The number  $n_1$  is inserted into  $P'_{>1}$ . Notice that  $m_2$  is ejectable in  $P'_{>1}$  since it is the end of a bumping path in  $P'_{>1}$ . However,  $n_1$  is not ejectable in  $P'_{>1}$  by the condition of this case. Thus  $m_2 \neq n_1$ . Since  $m_2 \geq n_1$  we deduce that  $m_2 > n_1$ .
- Indirect Replacement case 1: The number  $y$  is inserted into  $P'_{>1}$ . We have  $m_2 \geq n_1 > y$ .
- Indirect Replacement case 2: The number  $n_2$  is inserted into  $P'_{>1}$ . We have  $m_2 \geq n_1 > n_2$ .

□

To summarize, our new reverse insertion satisfies the following Pieri property.

**Theorem 5.7.** *Let  $P$  be a decreasing tableau. Apply successive reverse insertions*

$$\begin{aligned}\Psi(P, (r_1, c_1), \alpha) &= (P', m) \\ \Psi(P', (r_2, c_2), \alpha') &= (P'', m')\end{aligned}$$

*Then  $c_2 < c_1$  if and only if  $m' > m$ .*

*Proof.* Follows from Lemma 3.4, Lemma 5.6, and Theorem 5.3.  $\square$

The following is an equivalent restatement for insertion.

**Corollary 5.8.** *Let  $P$  be a decreasing tableau and  $m, m' \in \mathbb{Z}_{>0}$ . Applying successive insertions  $\Phi(P, m) = (P', (r_1, c_1), \alpha)$  and  $\Phi(P', m') = (P'', (r_2, c_2), \alpha')$ , we have  $m > m'$  if and only if  $c_1 < c_2$ .*

Given a compatible pair  $(a, i)$  and starting with the empty tableau pair, use  $\Phi$  to insert  $a_1$ , then  $a_2$ , and so on, recording the insertion of  $a_k$  by  $i_k$ , producing a tableau pair  $(P, Q)$  where  $P$  is decreasing. Denote this map by  $\Phi_{RSK}(a, i) = (P, Q)$ .

**Corollary 5.9.**  $\Phi_{RSK}$  is a weight-preserving bijection  $\text{CP}_w \rightarrow \text{Dec}_w \times_{\mathbb{Y}} \text{SVT}$ .

*Proof.* Corollary 5.8 implies that  $Q$  is set-valued. Moreover it also implies that the process can be inverted: if  $(P, Q) \in \text{Dec}_w \times_{\mathbb{Y}} \text{SVT}$ , then using  $\Psi$  at the sequence of removable boxes given by the entries of  $Q$ , one recovers  $(a, i) \in \text{CP}_w$ .  $\square$

**Example 5.10.** Consider the following element  $(a, i) \in \text{CP}_{25143}$ .

$i$	5	5	5	4	3	2	2	1
$a$	1	3	4	4	3	1	4	2

We have

$$\Phi_{RSK}(a, i) = \left( \begin{array}{|c|c|c|} \hline 4 & 3 & 1 \\ \hline 3 & 1 & \\ \hline 2 & & \\ \hline \end{array}, \begin{array}{|c|c|c|} \hline 1 & 2 & 5 \\ \hline 2 & 3, 5 & \\ \hline 4, 5 & & \\ \hline \end{array} \right)$$

**Remark 5.11.** Reversing comparison of values, one obtains a weight-preserving bijection  $\Phi_{RSK}^{\text{Inc}} : \text{CP}_w \cong \text{Inc}_w \times_{\mathbb{Y}} \text{RSVT}$ .

Finally and crucially, we have the following property.

**Theorem 5.12.** *[OY] Let  $(a, i) \in \text{CP}_w$  and  $\Phi_{RSK}^{\text{Inc}}(a, i) = (P, Q) \in \text{Inc}_w \times_{\mathbb{Y}} \text{RSVT}$ . Then  $(a, i)$  is bounded (that is  $i_k \leq a_k$  for all  $k$ ) if and only if  $K_-(P) \geq K_-(Q)$  where  $K_-$  is the left key and the comparison is entry-wise.*

In [SY] Hecke column insertion was used to prove a bijection of bounded compatible pairs with pairs  $(P, Q)$  with  $P$  decreasing and  $Q$  reverse set-valued of the same shape, such that  $K_+(P) \geq K_-(Q)$ , leading to a bijective proof of the expansion of a Grothendieck polynomial into Lascoux polynomials. Theorem 5.12 leads to a new bijective proof [OY] of the expansion of a Grothendieck polynomial into Lascoux polynomials.

## 6 Comparison with Hecke row insertion

In this section, we compare our new insertion  $\Phi$  with Hecke row insertion.

Hecke row insertion is a map  $\Phi^{\text{Hecke}}(P, m) \mapsto (P', s, \alpha)$ , where the inputs and outputs have the same types as our insertion algorithm. It is merely the transpose of the Hecke column insertion of [BKSTY], in which the roles of rows and columns have been exchanged. The inverse of  $\Phi^{\text{Hecke}}$  is called Hecke reverse row insertion and denoted  $\Psi^{\text{Hecke}}$ .

Let  $P$  be a decreasing tableau. Apply  $\Psi^{\text{Hecke}}$  sending  $(P, s, \alpha) \mapsto (P', m)$ . Let  $m_r < \dots < m_1$  be the bumping path of  $s$  in  $P$ . Then  $m_1 = m$ .

**Remark 6.1.** On any input  $(P, s, \alpha)$ ,  $\Psi^{\text{Hecke}}$  and  $\Psi$  yield the same output number.

Besides yielding the same output number,  $\Psi$  and  $\Psi^{\text{Hecke}}$  share many properties:  $\Psi^{\text{Hecke}}$  also satisfies Equation (9), Lemma 3.3, and Lemma 3.5. However  $\Psi^{\text{Hecke}}$  does not satisfy Theorem 5.7. Consider the following counterexample:

**Example 6.2.** The green cell indicates the starting position of the reverse insertion.

$$\begin{aligned} \Psi^{\text{Hecke}} \left( \begin{array}{|c|c|} \hline 3 & 2 \\ \hline 1 & \\ \hline \end{array}, \alpha = 0 \right) &= \left( \begin{array}{|c|c|} \hline 3 & 2 \\ \hline 1 & \\ \hline \end{array}, 2 \right) \\ \Psi^{\text{Hecke}} \left( \begin{array}{|c|c|} \hline 3 & 2 \\ \hline 1 & \\ \hline \end{array}, \alpha = 0 \right) &= \left( \begin{array}{|c|c|} \hline 3 & 1 \\ \hline 1 & \\ \hline \end{array}, 2 \right) \end{aligned}$$

**Remark 6.3.**  $\Psi^{\text{Hecke}}$  satisfies a variation of Theorem 5.7:

**Theorem 6.4.** *Let  $P$  be a decreasing tableau. Applying successive reverse Hecke row insertions  $\Psi^{\text{Hecke}}(r_1, c_1), \alpha = (P', m)$  and  $\Psi^{\text{Hecke}}(P', (r_2, c_2), \alpha') = (P'', m')$ , we have  $r_1 > r_2$  if and only if  $m' < m$ .*

This is not satisfied by  $\Psi$ , as shown in the following example.

**Example 6.5.** We have

$$\begin{aligned} \Psi \left( \begin{array}{|c|c|} \hline 3 & 2 \\ \hline 1 & \\ \hline \end{array}, \alpha = 0 \right) &= \left( \begin{array}{|c|c|} \hline 3 & 1 \\ \hline 1 & \\ \hline \end{array}, 2 \right) \\ \Psi \left( \begin{array}{|c|c|} \hline 3 & 1 \\ \hline 1 & \\ \hline \end{array}, \alpha = 1 \right) &= \left( \begin{array}{|c|} \hline 3 \\ \hline 1 \\ \hline \end{array}, 1 \right) \end{aligned}$$

So  $r_2 = r_1$  and  $m' < m$ .

**Remark 6.6.** We have introduced two types of Pieri rules for reverse insertions on decreasing tableaux: the rule in Theorem 5.7 for  $\Psi$  and the rule in Theorem 6.4 for  $\Psi^{\text{Hecke}}$ . The reverse EG insertion satisfies both versions of Pieri rules. In other words, our insertion and Hecke row insertion are two different generalizations of EG insertion, aiming for different Pieri rules and yielding different bijections.

## References

- [BJS] S. Billey, W. Jockusch, and R. Stanley. Some combinatorial properties of Schubert polynomials. *J. Algebraic Combin.* 2 (1993), no. 4, 345–374.
- [BKSTY] A. S. Buch, A. Kresch, M. Shimozono, H. Tamvakis, and A. Yong. Stable Grothendieck polynomials and K-theoretic factor sequences. *Math. Ann.* 340 (2008), no. 2, 359–382.
- [Bu] A. Buch. A Littlewood-Richardson rule for the K-theory of Grassmannians. *Acta Math.* 189 (2002), no. 1, 37–78.
- [BS] A.S. Buch and M.J. Samuel, 2016. K-theory of minuscule varieties. *Journal für die reine und angewandte Mathematik*, 2016 (719), pp. 133–171.
- [EG] P. Edelman and C. Greene. Balanced tableaux. *Adv. in Math.* 63 (1987), no. 1, 42–99.
- [FK] S. Fomin and A. N. Kirillov. Grothendieck polynomials and the Yang-Baxter equation. Formal power series and algebraic combinatorics/Séries formelles et combinatoire algébrique, DIMACS, 1994, 183–189, Piscataway, NJ.
- [GMP+] C. Gaetz, M. Mastrianni, R. Patrias, H. Peck, C. Robichaux, D. Schwein, K. Y. Tam. K-Knuth equivalence for increasing tableaux. *Electron. J. Combin.* 23 (2016), no. 1, Paper 1.40, 37 pp.
- [HSY] D. Huang, M. Shimozono, and T. Yu. Marked Bumpless Pipedreams. In preparation.
- [LS] A. Lascoux and M. P. Schützenberger. Tableaux and non-commutative Schubert polynomials. *Funct. Anal. Appl.* 23 (1989), 63–64.
- [OY] G. Orelowitz and T. Yu. K-theoretic non-symmetric refinement of the Littlewood-Richardson rule. In preparation.
- [RS] V. Reiner and M. Shimozono. Key polynomials and a flagged Littlewood-Richardson rule. *J. Combin. Theory Ser. A* 70 (1995), no. 1, 107–143.

- [SY] M. Shimozono and T. Yu. Grothendieck-to-Lascoux expansions. arXiv:2106.13922.
- [TY] H. Thomas and A. Yong. A jeu de taquin theory for increasing tableaux, with applications to K-theoretic Schubert calculus. *Algebra Number Theory* 3 (2009), no. 2, 121–148.