Quasisymmetric (k,l)-hook Schur functions

Sarah Mason, Elizabeth Niese

FPSAC 2014, Chicago, USA DMTCS proc. **AT**, 2014, 229–240

Quasisymmetric (k, l)-hook Schur functions

Sarah K. Mason[1]* and Elizabeth Niese[2]†

[1] *Department of Mathematics, Wake Forest University, Winston-Salem, NC, USA*
[2] *Department of Mathematics, Marshall University, Huntington, WV, USA*

Abstract. We introduce a quasisymmetric generalization of Berele and Regev's hook Schur functions and prove that these new quasisymmetric hook Schur functions decompose the hook Schur functions in a natural way. In this paper we examine the combinatorics of the quasisymmetric hook Schur functions, providing analogues of the Robinson-Schensted-Knuth algorithm and a generalized Cauchy Identity.

Résumé. Nous introduisons une généralisation quasisymetrique des fonctions "hook Schur" de Berele et Regev et nous prouvons ces nouvelle fonctions hook Schur quasisymetrique décomposent les fonctions hook Schur. Dans cet article, nous examinons la combinatoire des fonctions hook Schur quasisymetrique, fournissant des analogues de l'algorithme de Robinson-Schensted-Knuth et une généralisation d'identité Cauchy.

Keywords: quasisymmetric functions, Schur functions, tableaux, RSK

1 Introduction

Hook Young diagrams became of interest as the classical Schur-Weyl duality was extended to the general linear Lie superalgebra. Schur [Sch01] determined a one-to-one correspondence between irreducible representations of the general linear group $GL(V)$ and subsets of the irreducible representations of S_n. Weyl's *Strip Theorem* [Wey39] states that these irreducible representations of the general linear group $GL(V)$ are precisely those obtained from partitions whose Young diagrams lie inside a strip of height k, where k is the dimension of the vector space V. Schur's action of S_n on $V^{\otimes n}$ and that from Weyl's Strip Theorem are dual. Berele and Regev [BR87] generalize the two actions of S_n on $V^{\otimes n}$ into a single action by considering a decomposable vector space $V = T \oplus U$ such that $\dim(T) = k$ and $\dim(U) = l$. In this new setting, the indexing set is given by partitions which lie inside a hook shape of height k and width l, called a *hook Young diagram*, meaning there are at most k parts greater than l. Berele and Regev use certain fillings of these diagrams to generate polynomials known as (k, l)-hook Schur functions on two sets of variables, which appear naturally when examining characters of a certain S_n representation of $GL(k) \times GL(l)$, and generalize the classical Schur functions.

Remmel [Rem84] introduces an analogue of the Robinson-Schensted-Knuth (RSK) algorithm for (k, l)-semistandard tableaux, the objects used to generate (k, l)-hook Schur functions. The insertion algorithm

*Email: masonsk@wfu.edu
†Email:niese@marshall.edu

underlying this RSK analogue is an important component in the rule for multiplying two (k,l)-hook Schur functions that is similar to the Littlewood-Richardson rule used to multiply two Schur functions. Remmel [Rem87] further exploits the rich structure of these objects to prove a number of permutation statistic identities, including a generalization of the *Cauchy Identity* [Mac92] which provides a generating function for products of Schur functions.

The Schur functions (which form a basis for symmetric functions) can be obtained as specializations of Macdonald polynomials [Mac95]. Similarly, a basis for quasisymmetric functions is obtained through specializations of nonsymmetric Macdonald polynomials [HLMvW11]. This basis, called the *quasisymmetric Schur function basis*, can also be obtained by summing certain collections of Type A Demazure characters and is of interest due to its combinatorial similarities to the Schur functions as well as its algebraic significance in the noncommutative character theory of the symmetric group [vW13].

In this paper, we provide a quasisymmetric analogue of the (k,l)-hook Schur functions obtained by summing the weights of fillings of composition diagrams satisfying certain conditions and prove that this analogue decomposes the (k,l)-hook Schur functions in a natural way. In Section 2, we describe the hook composition tableaux used to generate the quasisymmetric hook Schur functions which are obtained from a combination of quasisymmetric Schur functions and row-strict quasisymmetric Schur functions. In Section 3 we discuss several properties of the quasisymmetric hook Schur functions. In Sections 4 and 5 we introduce an insertion algorithm and use it to provide an analogue of the Robinson-Schensted-Knuth algorithm as well as a generalized Cauchy identity. In Section 6 we discuss several avenues for future research, including a potential method for finding a multiplication rule for quasisymmetric hook Schur functions.

2 Background

The (k,l)-hook Schur functions introduced by Berele and Regev [BR87] are defined combinatorially using (k,l)-semistandard hook tableaux. (Frequently the k and l designations are dropped and these diagrams are referred to simply as *semistandard hook tableaux*.) Begin with the *Young diagram* of $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, which is given by placing λ_i boxes (or *cells*) in the i^{th} row from the bottom of the diagram, in French notation. A *semistandard hook tableau* of shape λ is a filling of the Young diagram of λ with letters from two different alphabets (one primed and one unprimed, where each primed entry is considered to be larger than all unprimed entries) such that the unprimed entries weakly increase from left to right along rows and strictly increase from bottom to top in columns while the primed entries strictly increase from left to right along rows and weakly increase from bottom to top in columns [BR87]. All rows and columns must be weakly increasing, so in any given column all of the primed entries appear in a higher row than the unprimed entries, and in a given row all of the primed entries appear to the right of all of the unprimed entries. (See Figure 2.1 for an example.)

The quasisymmetric Schur functions were introduced in [HLMvW11] as polynomials generated by *composition tableaux*, generalizations of semistandard Young tableaux whose underlying shapes are compositions instead of partitions. These polynomials form a basis for quasisymmetric functions and decompose the Schur functions in a natural way. A closely related set of polynomials (still given by fillings of Young diagrams [LMvW13]) is more natural for us to work with for the purposes of this paper, but note that this new form, denoted here by $\mathcal{CS}_\alpha$, is easily obtained from the original definition by a reversal of the entries in a filling. A slight modification of the definition produces a new basis for quasisymmetric functions that is generated using a row-strict analogue of the composition tableaux [MR]. (We will again

$$T = \begin{array}{|c|c|c|c|} \cline{1-1} 1' \\ \cline{1-2} 1' & 2' \\ \hline 2 & 2 & 3' & 4' \\ \hline 1 & 1 & 4 & 4' \\ \hline \end{array}$$

Fig. 2.1: T is a semistandard hook tableau of shape $(4,4,2,1)$ and weight $x_1^2x_2^2x_4y_1^2y_2y_3y_4^2$.

work with the variation, $\mathcal{RS}_\alpha$, of the row-strict quasisymmetric functions obtained by the same reversal procedure as is employed in [LMvW13]. In fact, we further extend this approach to include skew compositions as indexing compositions.) Combining these two approaches, we have the following definition for the composition analogue of a (k,l)-semistandard hook tableau.

Let $\mathcal{A}$ be the alphabet $1 < 2 < 3 < \ldots$ and let $\mathcal{A}'$ be the alphabet $1' < 2' < 3' < \ldots$ where each primed letter is greater than all unprimed letters to give a total ordering of $1 < 2 < 3 \ldots < 1' < 2' < 3' < \ldots$ on $\mathcal{A} \cup \mathcal{A}'$. Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_l)$ be a composition of n. The diagram associated to α consists of l rows of left-justified boxes, or *cells*, such that the i^{th} row from the bottom contains α_i cells, as in the French notation. Given a composition diagram $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_l)$ with largest part m, a *hook composition tableau* (HCT), F, is a filling of the cells of α with letters from the alphabets $\mathcal{A}$ and $\mathcal{A}'$ such that

1. the entries of F weakly increase in each row when read from left to right,
2. the unprimed (resp. primed) entries of F weakly (resp. strictly) increase in each row when read from left to right,
3. the unprimed (resp. primed) entries in the leftmost column of F strictly (resp. weakly) decrease when read from top to bottom,
4. and F satisfies the following *triple rule*:

Supplement F by adding enough cells with infinity-valued entries to the end of each row so that the resulting supplemented tableau, $\hat{F}$, is of rectangular shape $l \times m$. Then for $1 \le j < i \le l, \ \ 2 \le k \le m$, where $\hat{F}(i,j)$ denotes the entry of $\hat{F}$ that lies in the cell in the i-th row from the bottom and j-th column from the left,

(a) if $\hat{F}(j,k+1) \in \mathcal{A}$ and $\hat{F}(j,k+1) \ge \hat{F}(i,k)$, then $\hat{F}(j,k+1) > \hat{F}(i,k+1)$, and

(b) if $\hat{F}(j,k+1) \in \mathcal{A}'$ and $\hat{F}(j,k+1) > \hat{F}(i,k)$, then $\hat{F}(j,k+1) \ge \hat{F}(i,k+1)$.

Note that triple rule (a) is identical to the triple rule used to define a *standard Young composition tableau* in [LMvW13]. This is due to the fact that the unprimed portion of the filling behaves like a semistandard Young composition tableau while the primed portion behaves like a row-strict analogue of a skew semistandard Young composition tableau.

Definition 2.1 *The quasisymmetric (k,l)-hook Schur function $\mathcal{HQ}_\alpha(x_1,\ldots x_k; y_1,\ldots y_l)$ indexed by the composition α is given by*

$$\mathcal{HQ}_\alpha(x_1,\ldots x_k; y_1,\ldots y_l) = \sum_{F \in HCT(\alpha)} x_1^{v_1} x_2^{v_2} \cdots x_k^{v_k} y_1^{u_1} y_2^{u_2} \cdots y_l^{u_l},$$

where $HCT(\alpha)$ is the set of all hook composition tableaux of shape α, v_i is the number of times the letter i appears in F, and u_i is the number of times the letter i' appears in F.

See Figure 2.2 for an example of a quasisymetric (k,l)-hook Schur function and the fillings appearing in such a function.

3 Properties of the quasisymmetric (k,l)-hook Schur functions

Every Schur function decomposes into a positive sum of quasisymmetric Schur functions. Similarly, every Schur function also decomposes into a positive sum of row-strict quasisymmetric Schur functions. In particular,

$$s_\lambda = \sum_{\tilde{\alpha}=\lambda} \mathcal{CS}_\alpha = \sum_{\tilde{\alpha}=\lambda'} \mathcal{RS}_\alpha$$

where $\mathcal{CS}_\alpha = \mathcal{HQ}_\alpha(\overline{x}, \emptyset)$ and $\mathcal{RS}_\alpha = \mathcal{HQ}_\alpha(\emptyset, \overline{x})$; that is, $\mathcal{CS}_\alpha$ is the quasisymmetric Schur function generated by column-strict composition tableaux of shape α satisfying the first triple condition while $\mathcal{RS}_\alpha$ is the row-strict quasisymmetric Schur function generated by row-strict composition tableaux of shape α satisfying the second triple condition. The following theorem demonstrates the fact that this behavior continues as expected in the case of quasisymmetric (k,l)-hook Schur functions.

Theorem 3.1 *The (k,l)-hook Schur functions decompose into a positive sum of quasisymmetric (k,l)-hook Schur functions in the following way:*

$$HS_\lambda(x_1,\dots x_k; y_1,\dots y_l) = \sum_{\tilde{\alpha}=\lambda} \mathcal{HQ}_\alpha(x_1,\dots x_k; y_1,\dots y_l),$$

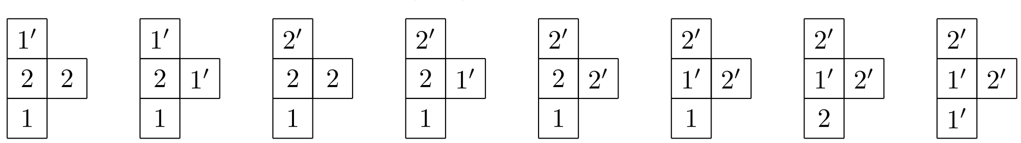
where $\tilde{a} = \lambda$ indicates that the parts of α rearrange to λ when placed in weakly decreasing order.

Proof: We exhibit a weight-preserving bijection, f, between the set of all semistandard hook tableaux of shape λ and the set of all hook composition tableaux whose shape rearranges to λ. This map is a generalization of the map given in [HLMvW11] between semistandard tableaux and composition tableaux.

Given a semistandard hook tableau T of shape λ, map the entries in the leftmost column of T to the leftmost column of $f(T)$ by placing them in weakly increasing order from bottom to top. Map each remaining set of column entries from T into the corresponding column of $f(T)$ by the following process:

1. Assume that the entries in the first $j-1$ columns have been inserted into $f(T)$ and begin with the smallest entry, a_1, in the set of entries in the j^{th} column of T.

$$\mathcal{HQ}_{(1,2,1)}(x_1,x_2;y_1,y_2) =$$



$$x_1x_2^2y_1 \ + \ x_1x_2y_1^2 \ + \ x_1x_2^2y_2 + x_1x_2y_1y_2 + x_1x_2y_2^2 \ + \ x_1y_1y_2^2 \ + \ x_2y_1y_2^2 \ + \ y_1^2y_2^2$$

Fig. 2.2: The quasisymmetric $(2,2)$-hook Schur function $\mathcal{HQ}_{(1,2,1)}(x_1,x_2;y_1,y_2)$.

2. If a_1 is unprimed, map a_1 to the highest available cell that is immediately to the right of an entry weakly smaller than a_1. If a_1 is primed, map a_1 to the highest available cell that is immediately to the right of an entry strictly smaller than a_1.

3. Repeat Step 2 with the next smallest entry, noting that a cell is *available* if no entry has already been placed in this cell.

4. Continue until all entries from this column have been placed, and then repeat with each of the remaining columns.

We must show that this process produces a hook composition tableau. The first two conditions are satisfied by construction, so we must check the third condition. Consider two cells $\hat{F}(j,k+1)$ and $\hat{F}(i,k)$ such that $\hat{F}(j,k+1) \in \mathcal{A}$ and $\hat{F}(j,k+1) \geq \hat{F}(i,k)$. We must show that $\hat{F}(j,k+1) > \hat{F}(i,k+1)$. Let $\hat{F}(i,k) = b$, $\hat{F}(j,k+1) = a$, and $\hat{F}(i,k+1) = c$. Then the cells are situated as shown, where $a \geq b$:

$$\begin{array}{|c|c|}\hline b & c \\ \hline \end{array}\,.$$

$$\begin{array}{c|c|}\cline{2-2} & a \\ \cline{2-2} \end{array}$$

We must prove that $\hat{F}(i,k+1) < \hat{F}(j,k+1)$, or in other words, that $c < a$. Assume, to get a contradiction, that $c > a$. (We know $c \neq a$ since $a \in \mathcal{A}$ and there are no repeated column entries from $\mathcal{A}$. Then a would be inserted into its column before c. But then the cell immediately to the right of b would be available during the insertion of a, and therefore a would be placed in that cell since $a \geq b$ and $a \in \mathcal{A}$. Therefore this configuration would not occur and thus $c < a$. The case where $\hat{F}(j,k+1) \in \mathcal{A}'$ is similar.

The inverse map, f^{-1}, is given by arranging the entries from each column of a hook composition tableau U so that the unprimed entries are strictly increasing from bottom to top, and above them the primed entries are weakly increasing from bottom to top. We must prove that if two entries, x and y, are in the same row with x immediately to the left of y, then either $x < y$ or $\{x = y$ and $x = y$ is unprimed$\}$. Argue by contradiction. Assume first that there exists a row in which x is immediately to the left of y but $x > y$. Choose the leftmost column c in which such an x exists, and the lowest row r containing this situation with x in column c. Then column c contains only $r-1$ entries which are less than or equal to y while column $c+1$ contains r entries less than or equal to y. Since the column entries in the hook composition tableau U are the same as the column entries in $f^{-1}(U)$, this implies that one of the entries less than or equal to y in column $c+1$ of the hook composition tableau must lie immediately to the right of an entry that is greater than y, which contradicts the definition of a hook composition tableau.

Next assume there exists a row in which x is immediately to the left of y and $x = y$ but $x = y$ is primed. Again, select the leftmost column c containing such an x, and the lowest row r containing this situation. Again, the column c contains only $r-1$ entries which are less than y while column $c+1$ contains r entries less than or equal to y. Since the column entries in the hook composition tableau U are the same as the column entries in $f^{-1}(U)$, this implies that one of the entries less than or equal to y in column $c+1$ of the hook composition tableau must lie immediately to the right of an entry that is greater than or equal to y, which contradicts the definition of a hook composition tableau. $\square$

Notice that each hook composition tableau appearing in a given quasisymmetric hook Schur function can be broken into its row-strict portion and its column-strict portion. We may therefore decompose each

$$\begin{array}{|c|c|c|c|}\hline 1' \\ \cline{1-3} 1' & 2' & 4' \\ \hline 2 & 2 & 3' & 4' \\ \hline 1 & 1 & 3 & 1' \\ \hline\end{array} \quad \longrightarrow \quad \begin{array}{|c|c|c|c|}\hline 1' & 2' & 3' & 4' \\ \hline 1' \\ \hline 2 & 2 & 3 & 1' \\ \hline 1 & 1 & 4' \\ \cline{1-3}\end{array}$$

Fig. 3.1: The bijection f maps a semistandard hook tableau of shape $(4, 4, 3, 1)$ to a hook composition tableau of shape $(3, 4, 1, 4)$.

quasisymmetric hook Schur function into a sum of products of quasisymmetric Schur functions and skew row-strict quasisymmetric Schur functions as follows:

$$\mathcal{HQ}_\alpha(\overline{x}, \overline{y}) = \sum_{\beta \subseteq \alpha} \mathcal{CS}_\beta(\overline{x}) \mathcal{RS}_{\alpha//\beta}(\overline{y}).$$

This is analogous to the similar decomposition of the hook Schur functions into sums of products of Schur functions and skew Schur functions given by

$$HS_\lambda(\overline{x}, \overline{y}) = \sum_{\mu \subseteq \alpha} s_\mu(\overline{x}) s_{\lambda'/\mu'}(\overline{y}).$$

However, some other quasisymmetric analogies of straightforward results about hook Schur functions do not carry through as easily. For example, one can see that

$$HS_\lambda(\overline{x}, \overline{y}) = HS_{\lambda'}(\overline{y}, \overline{x}) \tag{3.1}$$

by taking the transpose of each generating semistandard hook tableau. However, taking the transpose of a composition (which rearranges to a partition λ) using the standard method (writing it as a ribbon and then transposing the ribbon and recording the underlying composition) does not produce a composition which rearranges the transpose of the partition λ. Other possible choices for the transpose of a composition yield shapes whose fillings do not contain the appropriate weights. Thus the obvious analogs to (3.1) in the quasisymmetric setting fail.

4 An insertion algorithm for hook composition tableaux

We give an analogue of the composition tableau insertion algorithm [Mas08] for hook composition tableaux. Note that this algorithm also gives insertion algorithms for column- and row-strict composition tableaux if restricted to just one alphabet.

Given a hook composition tableau F and $x \in \mathcal{A} \cup \mathcal{A}'$, we insert x into F, denoted $F \leftarrow x$, in the following way:

1. Read down each column of $\hat{F}$, starting from the rightmost column and moving left. This is the *reading order* for $\hat{F}$.

   (a) If $x \in \mathcal{A}$, bump the first entry $\hat{F}(i, j)$ such that $\hat{F}(i, j) > x$ and $\hat{F}(i, j-1) \leq x$ and $j \neq 1$. If there is no such entry, then insert x into the first column, in between the unique pair $\hat{F}(i, 1)$ and $\hat{F}(i+1, 1)$ such that $\hat{F}(i, 1) < x < \hat{F}(i+1, 1)$. If $x < \hat{F}(1, 1)$, insert x at the bottom of the first column.

(b) If $x \in \mathcal{A}'$, bump the first entry $\hat{F}(i,j)$ such that $\hat{F}(i,j) \geq x$ and $\hat{F}(i,j-1) < x$ and $j \neq 1$. If there is no such entry, then insert x into the first column, in between the unique pair $\hat{F}(i,1)$ and $\hat{F}(i+1,1)$ such that $\hat{F}(i,1) < x \leq \hat{F}(i+1,1)$. If $x \leq \hat{F}(i,1)$ for all i, insert x at the bottom of the first column.

2. If $\hat{F}(i,j) = \infty$, the insertion terminates. If $\hat{F}(i,j) \neq \infty$, set $x = \hat{F}(i,j)$ and continue to scan cell entries in reading order, starting at cell (i,j).
3. Continue likewise until the insertion terminates.

In Figure 4.1 we show the insertion algorithm for several valus of $x \in \mathcal{A} \cup \mathcal{A}'$.

$$\begin{array}{cccc} 1' & 2' & 3' & 4' \\ 1' & & & \\ 2 & 2 & 3 & 1' \\ 1 & 1 & 1' & \end{array} \xrightarrow{\text{insert } 2} \begin{array}{cccc} 1' & 2' & 3' & 4' \\ 1' & & & \\ \mathbf{1'} & & & \\ 2 & 2 & \mathbf{2} & 1' \\ 1 & 1 & \mathbf{3} & \end{array} \qquad \begin{array}{ccccc} 1' & 3' & & & \\ 1' & & & & \\ 3 & 3 & 3 & 1' & 2' \\ 2 & 1' & 2' & & \\ 1 & 1 & 2 & 2' & 3' \end{array} \xrightarrow{\text{insert } 2'} \begin{array}{ccccc} 1' & \mathbf{2'} & & & \\ 1' & \mathbf{3'} & & & \\ 3 & 3 & 3 & 1' & \mathbf{2'} \\ 2 & 1' & \mathbf{2'} & & \\ 1 & 1 & 2 & \mathbf{2'} & 3' \end{array}$$

Fig. 4.1: Row-insertion into a hook composition tableau with bumping paths in bold.

Lemma 4.1 *Let F be a hook composition tableau and let $c_1 = (i_1, j_1)$ and $c_2 = (i_2, j_2)$ be cells in F such that (i_1, j_1) appears before (i_2, j_2) in reading order, $F(c_1) = F(c_2) = a$, and no cell between c_1 and c_2 in reading order has label a. In $F \leftarrow k$, let $(\overline{i_1}, \overline{j_1})$ and $(\overline{i_2}, \overline{j_2})$ be the cells that the entries $F(c_1)$ and $F(c_2)$ (respectively) are bumped to. Then $(\overline{i_1}, \overline{j_1})$ appears before $(\overline{i_2}, \overline{j_2})$ in reading order and if $a \in \mathcal{A}$ then $\overline{j_1} > \overline{j_2}$.*

The proof of Lemma 4.1 is a straightforward case-by-case argument left to the reader.

Lemma 4.2 *The result of the insertion algorithm is a hook composition tableau.*

Proof: Suppose the sequence of labels bumped during the insertion $F \leftarrow x$ is $x_0 = x, x_1, \ldots, x_m$ with each x_i bumped from cell c_i for $i \geq 1$. We prove by induction that the result of each bump is a hook composition tableau. Any triple of cells not involving the bumped entry will be unaffected by the insertion, so it is sufficient to consider only those triples involving the bumped entry. Assuming that $x_0, x_1, \ldots, x_{j-1}$ have been placed by the insertion algorithm, consider what occurs when x_j bumps x_{j+1} from cell c_{j+1}. We suppose that $x_j, x_{j+1} \in \mathcal{A}$, noting that the cases where $x_j \in \mathcal{A}$ with $x_{j+1} \in \mathcal{A}'$ and $x_j, x_{j+1} \in \mathcal{A}'$ are similar. When x_j is inserted, bumping x_{j+1} from cell c_{j+1}, we consider three possible locations of the cell c_{j+1} in the triple

$$\begin{array}{cc} \boxed{b} & \boxed{c} \\ \vdots & \\ & \boxed{a} \end{array}.$$

First we consider when c_{j+1} is in position a. Note that it is impossible for $F(b) \leq x_j < F(c)$ since if this were the case, x_j would have been bumped from a cell above c by induction and hence would end up bumping $F(c)$ instead of x_{j+1}. Thus, the triple condition is satisfied.

Next we consider when x_j bumps x_{j+1} from position b in the triple. We must show that if $x_j \leq F(a)$, then $F(a) > F(c)$, so assume $x_j \leq F(a)$. Note that by Lemma 4.1, x_j cannot be equal to a so in fact we may assume that $x_j < F(a)$. For any cell d, let $\overline{d}$ indicate the cell directly to the left of d. Given cells arranged as in the diagram

$$\begin{array}{|c|c|c|}\hline \overline{b} & b & c \\ \hline \end{array},$$

$$\begin{array}{|c|c|}\hline \overline{a} & a \\ \hline \end{array}$$

recall that x_j bumps the entry x_{j+1} from position b. Since the triple condition was satisfied prior to inserting x_j either $x_{j+1} \leq F(a)$ and $F(a) > F(c)$ or $x_{j+1} > F(a)$. If $x_{j+1} \leq F(a)$ and $F(a) > F(c)$, then we are done, so assume $x_{j+1} > F(a)$. Then $x_{j+1} > F(\overline{a})$, since $F(a) \geq F(\overline{a})$. Therefore, by the triple condition, $F(\overline{b}) > F(\overline{a})$. Then, by the insertion rules, since $x_j \geq F(\overline{b})$, we have $x_j > F(\overline{a})$. Since $F(\overline{a}) < x_j < F(a)$, if x_j was bumped from a cell earlier in the reading order than a, then x_j would bump $F(a)$ rather than x_{j+1}. Thus x_j must be bumped from a cell between a and b in the reading order. Since $F(\overline{a}) < x_j < F(a)$, x_j could not have been bumped from a cell in the same column and below a or else the triple condition would not have been satisfied. Thus, x_j must have been bumped from the same column as b. However, in this case, since $x_j > F(\overline{a})$, we know that $F(\overline{c_j}) > F(\overline{a})$ since the triple condition was satisfied prior to x_j being bumped from cell c_j, so $x_{j-1} > F(\overline{a})$, since the triple condition is satisfied after x_j is bumped from cell c_j. Similarly, $x_i > F(\overline{a})$ for all $0 \leq i \leq j$ and each of $x_1, x_2, \ldots, x_j$ is bumped from the same column. But, since $x_0 > F(\overline{a})$ and $x_0 < F(a)$, x_0 must have bumped $F(a)$ instead of x_1. Therefore $x_j < F(c) < F(a)$ and the triple condition is satisfied.

Finally, if x_j bumps x_{j+1} from position c in the triple, then $F(b) \leq x_j < x_{j+1}$. Thus, since the triple condition was satisfied in F, we know that either $F(a) < F(b)$ or $x_{j+1} < F(a)$. In either case, the triple condition is satisfied after the insertion of x_j. □

We note that this insertion procedure commutes with the bijection f used in the proof of Theorem 3.1. That is, given a semistandard hook tableau T and $x \in \mathcal{A} \cup \mathcal{A}'$, $f(T \leftarrow x) = f(T) \leftarrow x$ where $T \leftarrow x$ is computed using the row-insertion algorithm in [Rem84].

5 An analogue of the Robinson-Schensted-Knuth Algorithm

The Robinson-Schensted-Knuth (RSK) Algorithm is a bijection between matrices with non-negative integer entries and pairs of semi-standard Young tableaux. This correspondence utilizes an intermediary step which sends the matrix to a two-line array (a bi-word) satisfying certain properties and can be used to obtain information about the bottom line of this array, such as the length of the longest increasing subsequence in this word. When restricted to permutation matrices, the Robinson-Schensted-Knuth algorithm provides an elegant proof that the number of pairs of standard Young tableaux with n cells is equal to $n!$.

Berele and Remmel [BR85] extend this algorithm to a bijection between members of a certain class of matrices and pairs of semistandard hook tableaux. They use this to prove several important identities for Hook Schur functions including the following analogue of the Cauchy identity:

$$\sum_{\lambda} HS_\lambda(\overline{x};\overline{s})HS_\lambda(\overline{y};\overline{t}) = \prod_{i,j}(\frac{1}{1-x_iy_j})\prod_{i,j}(\frac{1}{1-s_it_j})\prod_{i,j}(1+x_it_j)\prod_{i,j}(1+y_is_j). \tag{5.1}$$

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 2 \\ 1 & 0 & 0 & 2 & 0 \end{bmatrix} \longrightarrow \begin{pmatrix} 1 & 1 & 2 & 1' & 1' & 1' & 2' & 2' & 2' \\ 2 & 1' & 3 & 2' & 2' & 3 & 1' & 1' & 1 \end{pmatrix} \longrightarrow \begin{array}{llll} \boxed{2'} & & & \\ \boxed{1'} & \boxed{2'} & & \\ \boxed{1'} & & & \\ \boxed{2} & \boxed{3} & \boxed{3} & \boxed{1'} \\ \boxed{1} & & & \end{array} \quad , \quad \begin{array}{llll} \boxed{2'} & & & \\ \boxed{2'} & & & \\ \boxed{1'} & & & \\ \boxed{2} & \boxed{1'} & & \\ \boxed{1} & \boxed{1} & \boxed{1'} & \boxed{2'} \end{array}$$

Fig. 5.1: Example of the RSK analogue for $M \in \mathcal{M}(k_1, l_1, k_2, l_2)$

We use the insertion procedure described in Section 4 to introduce an analogous bijection between members of a certain class of matrices and pairs of hook composition tableaux. This algorithm can be used to prove a generating function identity for quasisymmetric hook Schur functions.

Definition 5.1 *Let M be a $(k_2 + l_2) \times (k_1 + l_1)$ matrix. Then $M \in \mathcal{M}(k_1, l_1, k_2, l_2) \iff M$ satisfies the conditions given in the following diagram:*

$k_2 \times k_1$ *nonnegative integers*	$k_2 \times l_1$ 0 *or* 1
$l_2 \times k_1$ 0 *or* 1	$l_2 \times l_1$ *nonnegative integers*

This definition is identical to the definition given by Berele and Remmel [BR85] and the Theorem below provides a bijection from this same set of matrices to a different collection of tableau diagrams; namely to pairs of hook composition tableaux instead of pairs of semistandard hook tableaux. The process is similar to that of Berele and Remmel; the hook composition tableaux diagrams appear because we use the insertion process described in Section 4 rather than the insertion given by Berele and Remmel.

Theorem 5.2 *There exists a weight preserving bijection between matrices in the collection $\mathcal{M}(k_1, l_1, k_2, l_2)$ and pairs of hook composition tableaux with the same underlying partition.*

The proof of Theorem 5.2 involves sending a matrix from the collection $\mathcal{M}(k_1, l_1, k_2, l_2)$ to a biword using the Berele-Remmel procedure and then mapping this biword to a pair of composition tableaux using the insertion procedure described in Section 4. In particular, the bottom line of the biword is inserted into an empty diagram to form a hook composition tableau while the top line records the locations of the inserted letters, where entries inserted into the leftmost column possibly appear in a different row than in the insertion tableau to maintain the decreasing condition on the leftmost column.

The following analogue of the Cauchy identity is a corollary to Theorem 5.2 but we prove it using Equation 5.1 and Theorem 3.1.

Corollary 5.3

$$\sum_{\lambda}\sum_{\tilde{\alpha}=\tilde{\beta}=\lambda}\big(HQ_{\alpha}(\overline{x};\overline{s})HQ_{\beta}(\overline{y};\overline{t})\big)=\prod_{i,j}(\frac{1}{1-x_iy_j})\prod_{i,j}(\frac{1}{1-s_it_j})\prod_{i,j}(1+x_it_j)\prod_{i,j}(1+y_is_j),$$

where α and β are compositions and $\tilde{\alpha}=\lambda$ means that when the parts of α are arranged in weakly decreasing order, the resulting partition is λ.

Proof: Recall Equation 5.1:

$$\sum_{\lambda}HS_{\lambda}(\overline{x};\overline{s})HS_{\lambda}(\overline{y};\overline{t})=\prod_{i,j}(\frac{1}{1-x_iy_j})\prod_{i,j}(\frac{1}{1-s_it_j})\prod_{i,j}(1+x_it_j)\prod_{i,j}(1+y_is_j),$$

which provides a generating function for products of hook Schur functions. Theorem 3.1 states that:

$$HS_{\lambda}(\overline{x};\overline{y})=\sum_{\tilde{\alpha}=\lambda}HQ_{\alpha}(\overline{x};\overline{y});$$

substitute this refinement of the hook Schurs into the generating function identity to obtain

$$\sum_{\lambda}\sum_{\tilde{\alpha}=\lambda}HQ_{\alpha}(\overline{x};\overline{y})\sum_{\tilde{\beta}=\lambda}HQ_{\beta}(\overline{x};\overline{y})=\prod_{i,j}(\frac{1}{1-x_iy_j})\prod_{i,j}(\frac{1}{1-s_it_j})\prod_{i,j}(1+x_it_j)\prod_{i,j}(1+y_is_j),$$

which reduces to

$$\sum_{\lambda}\sum_{\tilde{\alpha}=\tilde{\beta}=\lambda}\big(HQ_{\alpha}(\overline{x};\overline{s})HQ_{\beta}(\overline{y};\overline{t})\big)=\prod_{i,j}(\frac{1}{1-x_iy_j})\prod_{i,j}(\frac{1}{1-s_it_j})\prod_{i,j}(1+x_it_j)\prod_{i,j}(1+y_is_j)$$

when the summations are combined. □

6 Future directions

The multiplication of hook Schur functions behaves exactly the same as the multiplication of Schur functions in the sense that the structure constants are the same.

Theorem 6.1 (Remmel) *If*

$$s_{\nu}(\overline{x})s_{\mu}(\overline{x})=\sum_{\lambda}g^{\lambda}_{\nu,\mu}s_{\lambda}(\overline{x}),$$

then

$$HS_{\nu}(\overline{x};\overline{y})HS_{\mu}(\overline{x};\overline{y})=\sum_{\lambda}g^{\lambda}_{\nu,\mu}HS_{\lambda}(\overline{x};\overline{y})$$

We conjecture that the multiplication of quasisymmetric hook Schur functions similarly mimics the multiplication of quasisymmetric Schur functions, in so far as such multiplication rules are known. Currently, the known multiplication rules give a method for writing the product of a quasisymmetric Schur function and a Schur function as a positive sum of quasisymmetric Schur functions. One method that could potentially be used to prove that quasisymmetric hook Schur functions behave similarly is to expand the product using known rules (and some variants of known rules) and then collapse it back into a sum of quasisymmetric hook Schur functions as shown below, where the ultimate goal is to prove that the functions $\mathcal{CS}_\gamma(\overline{x})\mathcal{RS}_\rho(\overline{y})$ are in fact quasisymmetric hook Schur functions.

$$\begin{aligned}
\mathcal{HQ}_\alpha(\overline{x};\overline{y})HS_\lambda(\overline{x};\overline{y}) &= \left[\sum_{\beta\subseteq\alpha}\mathcal{CS}_\beta(\overline{x})\mathcal{RS}_{\alpha//\beta}(\overline{y})\right]\left[\sum_{\mu\le\lambda}s_\mu(\overline{x})s_{\lambda'/\mu'}(\overline{y})\right] \\
&= \sum_{\beta\subseteq\alpha}\sum_{\mu\le\lambda}\mathcal{CS}_\beta(\overline{x})s_\mu(\overline{x})\mathcal{RS}_{\alpha//\beta}(\overline{y})s_{\lambda'/\mu'}(\overline{y}) \\
&= \sum_{\beta\subseteq\alpha}\sum_{\mu\le\lambda}\left(\sum_\gamma A^\gamma_{\beta,\mu}\mathcal{CS}_\gamma(\overline{x})\right)\left(\sum_\delta B^\alpha_{\delta,\beta}\mathcal{RS}_\delta(\overline{y})\right)\left(\sum_\nu D^\lambda_{\mu,\nu}s_\nu(\overline{y})\right) \\
&= \sum_{\substack{\beta\subseteq\alpha,\ \mu\le\lambda\\ \gamma,\delta,\nu}} A^\gamma_{\beta,\mu}B^\alpha_{\delta,\beta}D^\lambda_{\mu,\nu}\mathcal{CS}_\gamma(\overline{x})\big[\mathcal{RS}_\delta(\overline{y})s_\nu(\overline{y})\big] \\
&= \sum_{\substack{\beta\subseteq\alpha,\ \mu\le\lambda\\ \gamma,\delta,\nu,\rho}} A^\gamma_{\beta,\mu}B^\alpha_{\delta,\beta}D^\lambda_{\mu,\nu}E^\rho_{\delta,\nu}\mathcal{CS}_\gamma(\overline{x})\mathcal{RS}_\rho(\overline{y})
\end{aligned}$$

One long-term goal is to extend these multiplication results to include a formula for an arbitrary product of two quasisymmetric hook Schur functions as a sum of quasisymmetric hook Schur functions. Such an expansion will have both positive and negative coefficients in general, and will involve removal as well as addition of cells to the diagrams appearing in the factors.

Several other interesting questions about the quasisymmetric hook Schur functions remain unanswered. In particular, we would like to understand the relationship between these functions and the superization of Gessel's Fundamental quasisymmetric functions given by Kwon [Kwo09]. We also seek branching rules for the enumeration of hook composition tableaux and generalizations of identities such as the Jacobi-Trudi identity. Finally, it would be valuable to understand the algebraic (representation theoretic) interpretation of the quasisymmetric hook Schur functions. One important avenue for studying the representation theoretic significance of the quasisymmetric hook Schur functions is to work in the dual, as was done in the case of quasisymmetric Schur functions [vW13].

References

[BR85] A. Berele and J. B. Remmel. Hook flag characters and their combinatorics. *J. Pure Appl. Algebra*, 35(3):225–245, 1985.

[BR87] A. Berele and A. Regev. Hook Young diagrams with applications to combinatorics and to representations of Lie superalgebras. *Adv. in Math.*, 64(2):118–175, 1987.

[HLMvW11] J. Haglund, K. Luoto, S. Mason, and S. van Willigenburg. Quasisymmetric Schur functions. *J. Combin. Theory Ser. A*, 118(2):463–490, 2011.

[Kwo09] Jae-Hoon Kwon. Crystal graphs for general linear Lie superalgebras and quasi-symmetric functions. *J. Combin. Theory Ser. A*, 116(7):1199–1218, 2009.

[LMvW13] Kurt Luoto, Stefan Mykytiuk, and Stephanie van Willigenburg. *An introduction to quasisymmetric Schur functions*. Springer Briefs in Mathematics. Springer, New York, 2013. Hopf algebras, quasisymmetric functions, and Young composition tableaux.

[Mac92] IG Macdonald. Schur functions: theme and variations. *Séminaire Lotharingien de Combinatoire (Saint-Nabor, 1992), Publ. Inst. Rech. Math. Av*, 498:5–39, 1992.

[Mac95] I. G. Macdonald. *Symmetric functions and Hall polynomials*. Oxford Mathematical Monographs. The Clarendon Press Oxford University Press, New York, second edition, 1995. With contributions by A. Zelevinsky, Oxford Science Publications.

[Mas08] S. Mason. A decomposition of Schur functions and an analogue of the Robinson-Schensted-Knuth algorithm. *Séminaire Lotharingien de Combinatoire*, 57(B57e), 2008.

[MR] S. Mason and J. Remmel. Row-strict quasisymmetric schur functions. *Annals of Combinatorics, to appear*.

[Rem84] J. B. Remmel. The combinatorics of (k,l)-hook Schur functions. In *Combinatorics and algebra (Boulder, Colo., 1983)*, volume 34 of *Contemp. Math.*, pages 253–287. Amer. Math. Soc., Providence, RI, 1984.

[Rem87] JB Remmel. Permutation statistics and (k, l)-hook schur functions. *Discrete Mathematics*, 67(3):298, 1987.

[Sch01] I Schur. *Über eine Klasse von Matrizen, die sich einer gegebenen Matrix zuordnen lassen*. PhD thesis, Berlin, 1901.

[vW13] S. van Willigenburg. Noncommutative irreducible characters of the symmetric group and noncommutative schur functions. *J. Comb.*, 4:403–418, 2013.

[Wey39] Hermann Weyl. *The Classical Groups. Their Invariants and Representations*. Princeton University Press, Princeton, N.J., 1939.