Tableau Switching: Algorithms and Applications

Georgia Benkart
Department of Mathematics
University of Wisconsin
Madison, Wisconsin 53706

Frank Sottile
Department of Mathematics
University of Toronto
Toronto, Ontario
M5S 1A1 Canada

Jeffrey Stroomer
AT&T Bell Laboratories - ALC 2C-362
1247 South Cedar Crest Boulevard
Allentown, Pennsylvania 18103

Journal of Combinatorial Theory A, Vol. 76, No. 1, October 1996, 11–43.

Abstract

We define and characterize *switching*, an operation that takes two tableaux sharing a common border and "moves them through each other" giving another such pair. Several authors, including James and Kerber, Remmel, Haiman, and Shimozono, have defined switching operations; however, each of their operations is somewhat different from the rest and each imposes a particular order on the switches that can occur. Our goal is to study switching in a general context, thereby showing that the previously defined operations are actually special instances of a single algorithm. The key observation is that switches can be performed in virtually any order without affecting the final outcome. Many known proofs concerning the jeu de taquin, Schur functions, tableaux, characters of representations, branching rules, and the Littlewood-Richardson rule use essentially the same mechanism. Switching provides a common framework for interpreting these proofs. We relate Schützenberger's evacuation procedure to switching and in the process obtain further results concerning evacuation. We define *reversal*, an operation which extends evacuation to tableaux of arbitrary skew shape, and apply reversal and related mappings to give combinatorial proofs of various symmetries of Littlewood-Richardson coefficients.

Introduction

Schützenberger's *jeu de taquin* [Sc1] is a combinatorial algorithm that transforms a (column strict) tableau of skew shape into another tableau with the same content but different shape. This algorithm has become one of the fundamental tools for studying tableaux and their applications.

Bender and Knuth [BK] present a combinatorial procedure for showing Schur functions are symmetric. To prove the Littlewood-Richardson rule, James and Kerber [JK] modify the Bender-Knuth procedure, constructing an algorithm for moving two tableaux past one another. White [W] applies the methods of [JK] to generalize the Littlewood-Richardson rule.

In [H1] Haiman presents another approach to the problem of moving tableaux past one another, and Shimozono expands upon these ideas in [Sh]. Essentially, each author views two tableaux sharing a border as halves of a single larger tableau. The problem of moving the halves past one another then becomes one of rearranging the order of the alphabet of the union. The result is an algorithm that allows great freedom in the order in which steps are performed.

Addressing questions concerning superSchur functions, Remmel [R] also considers the problem of moving two tableaux past one another. However in Remmel's setting one tableau is column strict while the other is row strict.

The primary purpose of our paper is to define and study an algorithm called the *switching procedure* and the mapping it calculates. This mapping, which we call *switching*, operates on pairs of tableaux. If S and T are tableaux where T extends S (i.e., the outer border of S is the inner border of T), switching "moves S and T through each other" transforming S into S_T and T into ST. The map has the following properties which characterize it uniquely:

I The objects S_T and ST are tableaux such that S_T extends ST, and the shape of ${}^ST \cup S_T$ is the same as the shape of $S \cup T$. Moreover, the contents of S_T and S are the same, as are the contents of ST and T.

II If $S \cup T$ has multiple components, we can switch S and T by switching the components individually.

III When S or T contains more than one integer, we can switch S and T recursively, i.e., by decomposing each into subtableaux (in a way made precise in §2) and switching the pieces.

In §2 we argue there can be at most one such map and exhibit it by proving that the mapping the switching procedure calculates has these properties. We show, moreover, that the steps of the switching procedure can be performed in nearly any order without affecting the final outcome. This implies the algorithms of [H1], [Sh], [JK], and [R] are particular cases of the switching procedure.

In §3 we apply the results from §2 to deduce properties of switching. These properties quickly lead to a single approach by which a large number of combinatorial identities can be proven. To illustrate the technique we present identities involving Schur functions, superSchur functions, the Littlewood-Richardson coefficients, multisymmetric functions, and branching rules.

In [Sc1], Schützenberger introduces a procedure called evacuation that transforms a tableau of normal (partition) shape into another tableau of the same shape. Evacuation is related to the *jeu de taquin*, and like the *jeu de taquin* it provides a vehicle for studying tableaux. In §5 we show how the switching procedure of §2 suggests an algorithm that generalizes Schützenberger's. We prove that the evacuation of a tableau of normal shape is the normal form of the tableau's rotation. This leads to two properties that characterize the evacuation of a tableau of normal shape and motivates our definition of a mapping called *reversal* that operates upon tableaux of arbitrary skew shape. Schützenberger [Sc2] extends evacuation to tableaux of arbitrary skew shape. In general reversal and evacuation produce different results, but they agree when restricted to tableaux of normal shape. The techniques used to calculate reversal can be applied to other mappings such as the White-Hanlon-Sundaram map ([W], [HS]). Section 5 concludes with a discussion of these mappings and their relationship to the symmetries of the Littlewood-Richardson coefficients described by Berenstein and Zelevinsky [BZ].

1. Preliminaries

In this section we establish conventions, give definitions, and review results that we use in subsequent sections. More detailed treatments of this material can be found in [Sa1] and [F].

We work with $\mathbf{Z} \times \mathbf{Z}$, which we think of as consisting of boxes, and number the rows and columns of $\mathbf{Z} \times \mathbf{Z}$ "matrix style", so row numbers increase top to bottom and column numbers increase left to right. When b and b' are boxes in $\mathbf{Z} \times \mathbf{Z}$, b is said to be *north* of b' provided the row containing b is above or equal to the row containing b'. We define the other compass directions analogously and allow ourselves the freedom to combine directions; for example, b is *northwest* of b' if b is both north and west of b'. If b and b' are distinct but adjacent boxes, they are *neighbors*. The *neighbor to the north* of a box is the one directly above it. We often consider objects obtained by filling some of the boxes in $\mathbf{Z} \times \mathbf{Z}$ with integers. If in such an object the integer u fills b and b is a neighbor of b', then u is a *neighbor* of b'.

A *partition* (or *normal shape*) λ is a sequence of integers

$$(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0).$$

We ignore the distinction between two partitions that differ only in the number of trailing zeros. We write $|\lambda| = \lambda_1 + \cdots + \lambda_n$ for the number which λ partitions. The partition λ can be regarded as the set $\{(i, j) \in \mathbf{Z} \times \mathbf{Z} \mid 1 \leq j \leq \lambda_i\}$. Thinking of $\mathbf{Z} \times \mathbf{Z}$ as a collection of boxes, we can picture λ as containing n left-justified rows of boxes with λ_i boxes in the i^{th} row for each i. For our purposes the difference between a partition and its picture is unimportant. Throughout this paper, κ, λ, μ, and ν represent partitions.

When $\lambda_i \geq \mu_i$ for every i, we write $\lambda \supseteq \mu$. For such λ and μ, the *skew shape* (or simply *shape*) λ/μ is the collection of boxes inside of λ but not in μ, and $|\lambda/\mu| = |\lambda| - |\mu|$ counts the number of boxes in λ/μ. We consistently use γ and δ to denote arbitrary skew shapes. Two shapes are equal if one is a translate of the other. This means a choice of partitions $\lambda \supseteq \mu$ such that $\lambda/\mu = \gamma$ is a choice of coordinates for γ, establishing its position in the plane. The maximal connected subsets of γ are its *components*; they are themselves skew shapes. We let γ^{t} denote the image of γ under the transpose $(i, j) \mapsto (j, i)$, and γ^* the image under the rotation $(i, j) \mapsto (-i, -j)$ through 180°. The rotation λ^* of a normal shape λ is an *anti-normal shape*.

Whenever $\lambda \supseteq \mu \supseteq \nu$, then λ/μ *extends* μ/ν. If γ extends the single box b, then b is an *inside corner* of γ. When b extends γ, then b is an *outside corner*. The operation $*$ transforms inside corners into outside corners.

A *tableau with shape* γ is a filling of all the boxes in γ with integers. These integers may be positive, zero, or negative and need not be distinct. A tableau U is *column strict* provided it satisfies the following:

1. Whenever u and u' are integers in U and u is northwest of u', then $u \leq u'$.

2. Within each column of U the integers must be distinct.

If the transpose U^t of U is column strict, then U is *row strict*. A column or row strict tableau is *positive* if all of its integers are positive. Sometimes we write $\operatorname{sh} U$ for the shape of U. When U can be expressed as a disjoint union $U = V_1 \cup V_2 \cup \cdots \cup V_n$ of tableaux, each V_i is a *subtableau* of U. The subtableau V is a *component* of U provided $\operatorname{sh} V$ is a component of $\operatorname{sh} U$. Let U^* be the tableau obtained from U by rotating the shape 180° and replacing each integer u by $-u$. Note $U^{**} = U$. When S and T are tableaux and the shape of T extends the shape of S, we say T *extends* S. We write $S \cup T$ for the object formed by gluing S and T together. Except for a brief discussion in §2 of Remmel's work [R] and an example in §3 involving superSchur functions, every tableau in this paper is column strict. Accordingly we use "tableau" to mean "column strict tableau", "positive tableau" to mean "positive column strict tableau", and so forth.

The *content* of a tableau U is the sequence $(c_p, c_{p+1}, \ldots, c_q)$, where c_i is the number of occurrences of i in U. The *word* of U is the sequence of integers obtained by reading the rows of U west to east, starting with the southernmost row and working toward the north.

A tableau is *standard* if it has no repeated entries. Note the transpose of a standard tableau is again standard. It is sometimes necessary to start with a tableau U and derive a related standard tableau $\widehat{U}$ called the *standard renumbering* of U. If the content of U is $(c_p, c_{p+1}, \ldots, c_q)$ and n is some integer, we build $\widehat{U}$ by replacing the q's in U east to west by $n, n-1, \ldots, n-c_q+1$; the $(q-1)$'s east to west by $n-c_q, n-c_q-1, \ldots, n-c_q-c_{q-1}+1$; and so on. Most of the time the particular value we choose for n is of no consequence. Remembering the content of U allows us to recover U from $\widehat{U}$ in the obvious fashion. The following example shows a tableau U and its standard renumbering $\widehat{U}$ for $n = 15$:

$$U = \begin{array}{cccccc} & & & 1 & 1 & 3 \\ & & 2 & 2 & 4 & 5 \\ & 1 & 5 & 5 & 6 & 6 \\ & 3 & & & & \\ 1 & 5 & & & & \end{array} \qquad \widehat{U} = \begin{array}{cccccc} & & & 3 & 4 & 8 \\ & & 5 & 6 & 9 & 13 \\ & 2 & 11 & 12 & 14 & 15 \\ & 7 & & & & \\ 1 & 10 & & & & \end{array}$$

Let U be a tableau and b be an inner corner for U. A *contracting slide of U into the box b* is performed by moving the empty box at b through U, successively interchanging it with the neighboring integers to the south and east according to the following rules:

1. If the box has only one neighbor, interchange with that neighbor.

2. If it has two unequal neighbors, interchange with the smaller one.

3. If it has two equal neighbors, interchange with the one to the south.

The box moves in this fashion until it has no more neighbors to the south or east, i.e., until it has become an outer corner. We write $j^b(U)$ for the tableau produced (note the rules insure $j^b(U)$ is indeed column strict). When b is an outer corner there is an obvious analogous procedure called an *expanding slide*. We write $j_b(U)$ for its result.

More generally, when S and T are tableaux and T extends S, we can use S as a set of instructions telling where contracting slides should start in T: the first slide begins at the box containing the largest integer in $\widehat{S}$, the second at the box containing the next largest, and so on. We write $j^S(T)$ for the resulting tableau. Similarly, T tells where expanding slides can be applied to S; in this case we write $j_T(S)$ for the result.

Suppose a sequence of contracting slides reduces the tableau U to a tableau U^{n} of normal shape. Thomas [T] shows U^{n} is independent of the particular sequence of slides used, and so we refer to U^{n} as the *normal form* of U. Similarly, there is exactly one tableau of anti-normal shape that can be produced by expanding U with slides, the so-called *anti-normal form* U^{a} of U. Two tableaux are *Knuth equivalent* if one can be transformed into the other with a sequence of expanding and contracting slides. When U and V are Knuth equivalent we write $U \overset{\mathrm{k}}{\cong} V$.

A word $\omega = \omega_1, \ldots, \omega_n$ of positive integers is a *reverse lattice permutation* if each final segment $\omega_k, \ldots, \omega_n$ of ω contains at least as many i's as $(i+1)$'s for each $i > 0$. A *Littlewood-Richardson* or *LR* tableau is a tableau whose word is a reverse lattice permutation. Given any partition λ, define $Y(\lambda)$ to be the tableau obtained by filling the first row of λ with 1's, the second with 2's, and so on. It follows that the LR tableaux of partition shape are precisely the $Y(\lambda)$. The number of LR tableaux of content μ and shape λ/ν is the *Littlewood-Richardson coefficient* $c^{\lambda}_{\nu\mu}$. Often it is convenient to write this number as $c^{\lambda/\nu}_{\mu}$. Note that $c^{\lambda/\nu}_{\mu} = 0$ if $\lambda \not\supseteq \nu$.

The definition of LR tableaux presented above is conventional, but there is a second characterization which from our viewpoint is more useful: a tableau is LR if and only if it is Knuth equivalent to some $Y(\lambda)$. (One way to prove this is to show the tableau resulting from a slide on an LR tableau is again LR. This is the approach used in [Sa1], Lemma 4.9.5.) From this perspective, $c^{\lambda/\nu}_{\mu}$ counts the number of tableaux of shape λ/ν that are Knuth equivalent to $Y(\mu)$.

There are many ways to define the Schur functions, but the following is the most suitable for our purposes. If $\mathbf{x}$ stands for the infinitely many variables $x_1, x_2, \ldots$, the *Schur function* $s_{\lambda/\nu} = s_{\lambda/\nu}(\mathbf{x})$ is the symmetric function given by

$$s_{\lambda/\nu} = \sum_U \mathbf{x}^U .$$

Here the sum is over all positive tableaux U of shape λ/ν. The monomial $\mathbf{x}^U$ is the product $x_1^{c_1} \cdot x_2^{c_2} \cdots x_q^{c_q}$, where $(c_1, c_2, \ldots, c_q)$ is the content of U.

2. The Switching Procedure

In this section we describe an algorithm which we call the *switching procedure* and characterize *switching*, the mapping it calculates. We prove that the algorithms of Haiman [H1], Shimozono [Sh], James and Kerber [JK], and with a slight adjustment that of Remmel [R] are special cases of the switching procedure. We conclude by showing the switching procedure behaves as claimed.

Before we can present the algorithm we must make a few definitions. Our aim is to describe the intermediate objects produced as the algorithm moves tableaux S and T through each other.

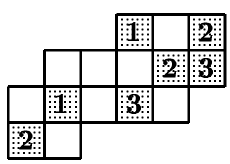
Let γ be a skew shape. A *perforated tableau U of shape γ* is a filling of some of the boxes in γ with integers. These integers may be positive, zero, or negative, and need not be distinct, but they must satisfy the following restrictions:

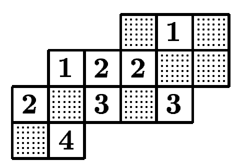
1. Whenever u and u' are integers in U and u is northwest of u', then $u \leq u'$.

2. Within each column of U the integers must be distinct.

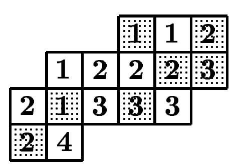
We write $\operatorname{sh} U$ for the shape of U, and extend the definitions of content and standard renumbering to perforated tableaux in the natural ways. Suppose S and T are perforated tableaux of shape γ and together they completely fill γ, i.e., every box of γ is filled with an integer from S or T and no box is filled twice. Then $S \cup T$ is a *perforated pair of shape γ*.

Displayed below are perforated tableaux S and T of shape γ.

$S =$ 

$T =$ 

Since together these tableaux fill all of γ and no box is numbered twice, they constitute a perforated pair $S \cup T$ of shape γ.

$S \cup T =$ 

We are interested in moving the integers in a perforated tableau in such a way that the result is again perforated. Let U be perforated of shape γ, and suppose the integer u in U is the neighbor to the north or west of an empty box of γ. If interchanging the positions of u and the empty box produces a perforated tableau, we say the interchange *expands* U. Similarly, when u is immediately south or east of the empty box and interchanging produces a perforated tableau, it *contracts* U. A perforated tableau that cannot be expanded (contracted) is *fully expanded* (*fully contracted*). Let $S \cup T$ be a perforated pair and suppose s and t are adjacent integers from S and T respectively. Interchanging s and t is a *switch* provided it simultaneously expands S and contracts T. We write $s \leftrightarrow t$ to represent the switch. If no s and t in $S \cup T$ can be switched, $S \cup T$ is *fully switched*.

Our algorithm is the following:

Algorithm 2.1 (The switching procedure).

1. *Start with tableaux S and T such that T extends S.*

2. *Switch integers from S with integers from T until it is no longer possible to do so.*

Of course, all we can say at this point is that the end result is a perforated pair whose shape is that of $S \cup T$. In fact, considerably more is true:

Theorem 2.2. *Assume the switching procedure transforms S into S_T and T into ${}^{S}T$. Then*

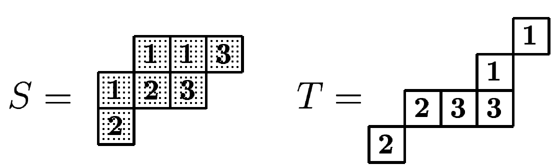
1. *S_T and ${}^{S}T$ are tableaux, and S_T extends ${}^{S}T$.*

2. *${}^{S}T \cup S_T$ has the same shape as $S \cup T$.*

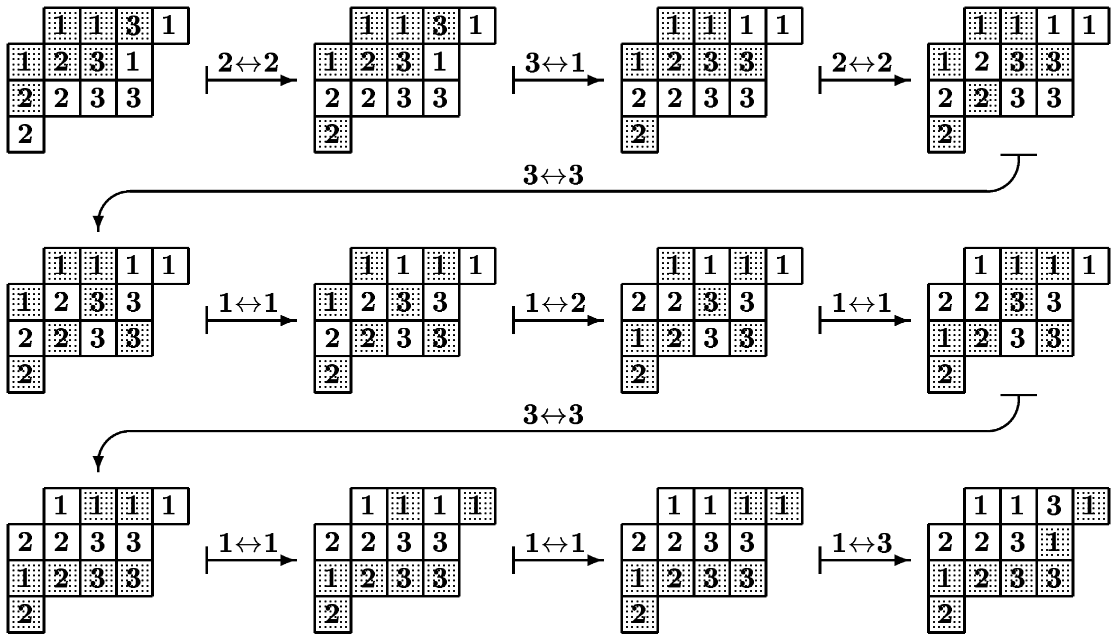
3. *S and S_T have the same content, as do T and ${}^{S}T$.*

4. *S_T and ${}^{S}T$ are independent of the particular sequence of switches used to produce them.*

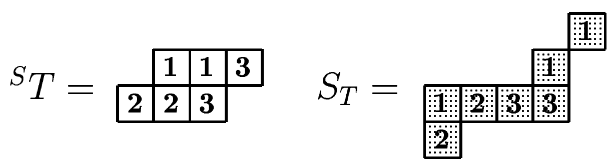
Parts 2 and 3 are obviously true. We defer the proofs of 1 and 4 until the end of the section and proceed to give an example. Let S and T be the following tableaux:



If we apply the algorithm to $S \cup T$, one possible sequence of switches is the following:



At the end the algorithm has produced the following tableaux:



Now we define the switching map, prove it has the properties we described in the introduction, and show these properties characterize the map uniquely.

Suppose S and T are tableaux and T extends S. Throughout the rest of this paper we write S_T and ST for the tableaux that S and T respectively become when the switching procedure is applied to $S \cup T$. Define the *switching map* (or more briefly, *switching*) to be the mapping $S \cup T \longmapsto {}^ST \cup S_T$ the procedure calculates.

To characterize switching we need a definition. When U is a tableau, let us say subtableaux U_1 and U_2 *decompose* U provided $U = U_1 \cup U_2$ and U_2 extends U_1. We require, moreover, that whenever u_1 and u_2 are integers in U_1 and U_2, respectively, then either $u_1 < u_2$, or $u_1 = u_2$ and u_1 is west of u_2.

Theorem 2.3. *Switching $S \cup T \longmapsto {}^ST \cup S_T$ is the unique map with the following properties:*

I *S_T and ST are tableaux, S_T extends ST, and ${}^ST \cup S_T$ has the same shape as $S \cup T$. Moreover, S_T and S share the same content, as do T and ST.*

II *If $S \cup T$ has multiple components, we can calculate S_T and ST by switching the components of $S \cup T$ independently.*

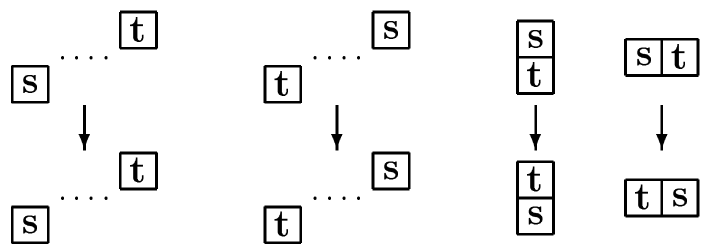
III *Suppose T_1 and T_2 decompose T. Then we can switch S with T in stages as follows. Writing S' for S_{T_1}, we have*

$$S_T = (S')_{T_2} \qquad \text{and} \qquad {}^ST = {}^ST_1 \cup {}^{S'}T_2.$$

Similarly if subtableaux decompose S, we can switch T with S in stages.

Proof. Switching has Property I by parts 1, 2, and 3 of Theorem 2.2, and it is clear from the definition of the procedure that switching has Property II. Note that switching in stages is simply a certain choice of order in the procedure; the map therefore has Property III by 4 of Theorem 2.2.

To see switching is the unique map with these properties, suppose $S \cup T \longmapsto \overline{T} \cup \overline{S}$ transforms S into $\overline{S}$, T into $\overline{T}$, and has Properties I, II, and III. When S and T are the single-box tableaux $\boxed{s}$ and $\boxed{t}$ respectively, then $\overline{T} \cup \overline{S} = {}^ST \cup S_T$ since Properties I and II force S and T to transform in the way shown by the picture below:



In words, if S and T are not adjacent, then $S = S_T = \overline{S}$ and $T = {}^ST = \overline{T}$; if they are, then $S_T = \overline{S}$ is the tableau whose position is that of T, but whose content is that of S, and similarly for ${}^ST = \overline{T}$. But then inducting on the number of boxes in $S \cup T$ and using Property III gives $\overline{T} \cup \overline{S} = {}^ST \cup S_T$ for every $S \cup T$. ◼

Next we show the algorithms of Haiman [H1] and Shimozono [Sh] are special cases of the switching procedure. Let S and T be tableaux with T extending S. To avoid hiding the essentials behind unnecessary details, assume S and T are standard, say with integers $1_S, 2_S, \ldots, p_S$ and $1_T, 2_T, \ldots, q_T$ respectively. Note that if we assign the ordering $O : 1_S < \cdots < p_S < 1_T < \cdots < q_T$, we can think of $S \cup T$ as a standard tableau. In an ordering when $i_S < j_T$ and $i_S \leq i'_S < j'_T \leq j_T$ forces $i_S = i'_S$ and $j_T = j'_T$, let us say j_T *covers* i_S. To move S and T through one another, Haiman and Shimozono use the following procedure, which we call *shuffling*:

Algorithm 2.4 (Shuffling).

1. *Start with S, T, and O as above.*

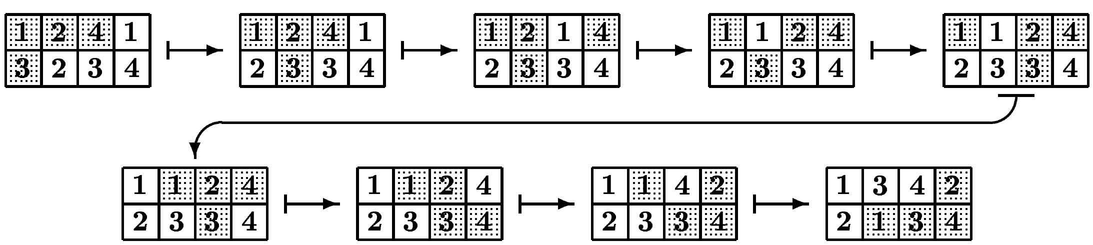
2. *Suppose after a (possibly empty) sequence of steps we have obtained the perforated pair $S' \cup T'$ and an ordering O' with respect to which $S' \cup T'$ is a standard tableau. Choose some i_S and j_T such that j_T covers i_S and interchange their order in O' so that $j_T < i_S$. Simultaneously, adjust $S' \cup T'$ as follows: if i_S and j_T are adjacent, interchange their positions; otherwise, do nothing.*

3. *Repeat 2 until there are no i_S and j_T with j_T covering i_S.*

An easy induction on the number of steps shows that every time step 2 is performed, the result is a tableau which is standard with respect to the updated order. Moreover, regardless of how the order is updated, we always have $1_S < \cdots < p_S$ and $1_T < \cdots < q_T$. It follows that the $S' \cup T'$ produced at each step is a perforated pair. When shuffling ends, $1_T < \cdots < q_T < 1_S < \cdots < p_S$, and thus

Proposition 2.5. *Shuffling is a particular case of the switching procedure of Algorithm 2.1.*

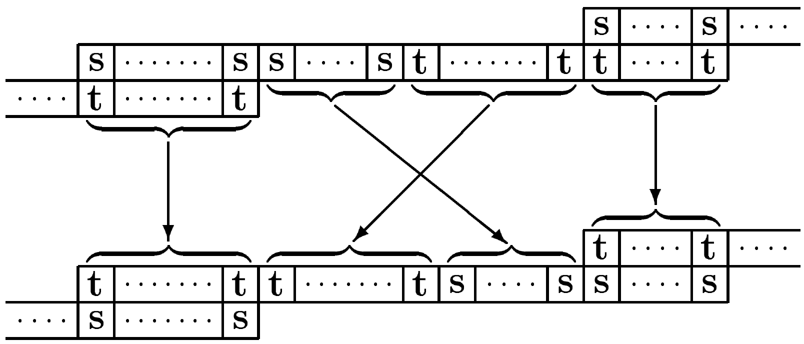
The reverse is not true: some sequences of switches allowed by the switching procedure cannot be obtained with shuffling. To see this, consider the following example:



This is an acceptable sequence of switches for Algorithm 2.1. However, the first switch in shuffling must interchange the shaded 4 with the unshaded 1, and therefore the above cannot occur.

In [H2] Haiman defines two algorithms for moving S and T through one another when S and T are standard tableaux and T extends S. Essentially, these algorithms are extreme cases of shuffling. Haiman's first algorithm amounts to consistently switching the greatest possible integer from S and his second to switching the least possible integer in T. Therefore, Proposition 2.5 implies the algorithms of Haiman [H2] are also particular cases of the switching procedure.

For the algorithm of James and Kerber [JK], assume S and T are tableaux, T extends S, every integer in S equals s, and every integer in T equals t. Clearly, each column of $S \cup T$ has one or two boxes, each two-box column has an s above a t, and each one-box column contains an s or a t. Moreover, if we discard the two-box columns, each row of what remains must consist of a (possibly empty) sequence of s's followed by a (possibly empty) sequence of t's. Their procedure for moving S through T is described by the picture below:



To state this explicitly,

1. In each two-box column, interchange the positions of s and t.

2. In each row, reorder the s's and t's not moved in step 1. If the row contains k such s's followed by ℓ such t's, replace them by a sequence of ℓ t's followed by k s's.

It is easy to calculate the effect the switching procedure has on S and T (for example, by consistently switching the easternmost possible s); the result is the same as that produced by the algorithm of James and Kerber.

More generally, suppose S and T are tableaux with T extending S. To move S and T through one another using the algorithm of James and Kerber, we break S and T into subtableaux $S_p, S_{p+1}, \ldots, S_q$ and $T_p, T_{p+1}, \ldots, T_q$, where, for each i and j, S_i contains the i's of S, and T_j contains the j's of T. We then iterate the above to move T_p through all of S, then T_{p+1} through what S has become, and so on. But this is nothing but a particular choice of switches in the switching procedure. Consequently, by 4 of Theorem 2.2 we have

Proposition 2.6. *The algorithm of James and Kerber is a special case of the switching procedure of Algorithm 2.1.*

Let S and T be tableaux, one row strict and the other column strict, such that T extends S. Remmel [R] describes a method for moving S and T through one another. The following simple adjustment to the switching procedure yields an algorithm that generalizes Remmel's.

A *perforated* $\mathfrak{t}$*-tableau of skew shape* γ is the transpose of a perforated tableau of shape $\gamma^{\mathfrak{t}}$. Assume one of S or T is a perforated tableau and the other is a perforated $\mathfrak{t}$-tableau, both of shape γ. Suppose together they completely fill γ, i.e., every box of γ is filled with an integer from S or T, and no box is filled twice. Then $S \cup T$ is a *perforated* $\mathfrak{t}$*-pair of shape* γ. The notions of expanding and contracting extend to perforated $\mathfrak{t}$-tableaux in the natural way. Let $S \cup T$ be a perforated $\mathfrak{t}$-pair, and suppose s and t are adjacent integers from S and T respectively. Interchanging s and t is a $\mathfrak{t}$*-switch* if it simultaneously expands S and contracts T. The modified version of the switching procedure is the following:

1. Start with tableaux S and T such that one is column strict, the other is row strict, and T extends S.

2. Perform $\mathfrak{t}$-switches of integers from S with integers from T until it is no longer possible to do so.

First assume S is column strict and T is row strict. Theorem 2.2, translated in the obvious way, remains true in this setting. Thus, if the new procedure transforms S into S_T and T into ST, then S_T and ST are, respectively, column strict and row strict tableaux, S_T extends ST, and both are independent of the sequence of $\mathfrak{t}$-switches used. Remmel's algorithm is the extreme case of the modified procedure in which the $\mathfrak{t}$-switch always involves the largest possible integer from S. (When there is more than possibility we choose the easternmost one.) The case where S is row strict and T is column strict is analogous.

We conclude this section by proving Theorem 2.2. Our first step in this direction is

Lemma 2.7. *If a perforated tableau U cannot be expanded (respectively, contracted), then the nonempty boxes in U form a tableau of skew shape.*

Proof. First note that whenever some integer u in U is the north or west neighbor of an empty box, U can be expanded. To see this, start by assuming u is the north neighbor of an empty box. (If there is more than one such u, take the westernmost one.) If u cannot be slid south into the box, there must be some integer $u' > u$ of U to the west of the box, and by choosing the easternmost such u' we can assume u' is the west neighbor of some empty box. The choice of u precludes the possibility that u' is the north neighbor of an empty box. This and the fact that U is perforated imply that u' can be slid east, and thus U can be expanded. The case where u is the west neighbor of the empty box is similar.

Next let U be a fully expanded perforated tableau, and suppose u and u' are integers in U such that u is to the northwest of u'. If the rectangle whose northwest and southeast corners are u and u' respectively contains any empty boxes, the paragraph above shows U can be expanded. Therefore, the nonempty boxes in U occupy a skew shape. This implies the nonempty boxes in U form a tableau of skew shape.

The arguments when U is fully contracted are completely analogous. ◼

This lemma suggests that if we start with a perforated pair $S' \cup T'$ and switch until it is no longer possible to do so, the result will be a pair of tableaux. Unfortunately, as the following simple example illustrates, this need not be the case.

$\mathfrak{t}$	1
2	$\mathfrak{t}$

As we shall see below, this problem does not arise in practice. Rather than starting with an arbitrary $S' \cup T'$, we begin with tableaux S and T where T extends S, and then perform a (possibly empty) sequence of switches to produce $S' \cup T'$. We prove that whenever S' is not fully expanded (or equivalently, T' is not fully contracted), there is a switch $s \leftrightarrow t$ for $S' \cup T'$. Our method is to show that $S' \cup T'$ must have a form that precludes configurations like the one above.

Let $S' \cup T'$ be a perforated pair and suppose there are two occurrences of t in T', one to the northwest of the other. The two t's define a rectangle which we call a $t \cdots t$ *rectangle* provided all other boxes in the rectangle are filled with integers from S'. If T' contains letters and S' integers, then the following is a $t \cdots t$ rectangle:

t	2	2	2	3	4	6
1	3	3	4	5	5	7
4	4	5	5	7	8	8
5	6	6	6	9	9	t

A pair (s, s') of integers from S' is a *step* provided $s \leq s'$ and s is immediately to the southwest of s'. Assume the rows in a $t \cdots t$ rectangle are numbered $1, 2, \dots, p$ north to south, and columns $1, 2, \dots, q$ west to east. Then a sequence $(s_1, s_1'), (s_2, s_2'), \dots, (s_{p-1}, s_{p-1}')$ of steps is a *staircase* for the rectangle provided each s_i' is in row i and column j_i, where $2 \leq j_1 \leq j_2 \leq \cdots \leq j_{p-1}$. Note any $t \cdots t$ rectangle with only a single row vacuously contains a staircase. The following displays the staircase $(1, 2), (5, 5), (6, 7)$ within our previous example:

t	2	2	2	3	4	6
1	3	3	4	5	5	7
4	4	5	5	7	8	8
5	6	6	6	9	9	t

Interchanging the roles of S' and T', we define $s \cdots s$ rectangles and steps and staircases for $s \cdots s$ rectangles analogously. A perforated pair $S' \cup T'$ is said to *have staircases* if every $s \cdots s$ rectangle and every $t \cdots t$ rectangle contains a staircase.

Lemma 2.8. *Suppose $S' \cup T'$ is a perforated pair having staircases and $S'' \cup T''$ is obtained from $S' \cup T'$ by performing a switch $s \leftrightarrow t$. Then $S'' \cup T''$ has staircases.*

Proof. To see $S'' \cup T''$ has staircases we first examine its $t \cdots t$ rectangles. Each such rectangle either is inherited unchanged from $S' \cup T'$, is newly created by the switch, or is produced by modifying some previously existing rectangle. Inherited rectangles obviously contain staircases. A new rectangle either contains one row, or contains every row but the last of some rectangle in $S' \cup T'$; in either case the new rectangle contains a staircase. If $s \leftrightarrow t$ alters an existing rectangle, then t is the rectangle's northwest or southeast corner. There are four possibilities:

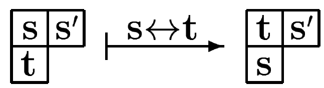
1. The integer t is a southeast corner and $s \leftrightarrow t$ moves t west, deleting a column;

2. The integer t is a southeast corner and $s \leftrightarrow t$ moves t north, deleting a row;

3. The integer t is a northwest corner and $s \leftrightarrow t$ moves t west, adding a column;

4. The integer t is a northwest corner and $s \leftrightarrow t$ moves t north, adding a row.

If $s \leftrightarrow t$ deletes a column, s cannot be the southwest part of a step, so no steps are deleted and the resulting rectangle contains the same staircase as its precursor. If a row is deleted, the resulting rectangle is missing the last step from its precursor, but is also missing the bottom row; hence, it contains a staircase. If a column is added, the resulting rectangle contains the staircase its precursor had. Finally, if a row is added, s and t switch as in the following picture, creating a new step:



It follows the new $t \cdots t$ rectangle contains a staircase.

Similar arguments show the $s \cdots s$ rectangles in $S'' \cup T''$ have staircases. ◾

Lemma 2.9. *Let $S' \cup T'$ be a perforated pair which has staircases. Either there is a switch $s \leftrightarrow t$ for $S' \cup T'$, or S' and T' are tableaux with S' extending T'.*

Proof. Suppose S' and T' are not tableaux with S' extending T'. By Lemma 2.7, T' is not fully contracted (equivalently, S' is not fully expanded), and therefore there is an integer t in T' to the southeast of some integer in S'. Let t be minimal among such integers; if there is more than one such t take the westernmost one. Choose s so it is the greatest integer in S' to the northwest of t. If there is more than one possibility, take the easternmost one. We claim s and t can be switched. If s is to the north of t, the choices of s and t guarantee they can be switched, so suppose s is to the west. The only circumstances under which it might be impossible to switch the two would be the following:

1. If there were a second copy of t, this one in the same column as s and to the north of s;

2. If there were a second copy of s, this one in the same column as t and to the south of t.

Consider the first case. Necessarily the two copies of t delimit a two-column $t \cdots t$ rectangle, and therefore s has to be the southwest piece of a step. This implies s', the northeast piece of the step, is immediately to the north of the rectangle's southeast

t. But $s' \geq s$, contradicting our choice of s. An analogous argument using $s \cdots s$ rectangles applies in the second case, and therefore s and t can be switched. ■

Proof of 1 of Theorem 2.2. Let S and T be tableaux where T extends S. Then $S \cup T$ is a perforated pair having staircases since every $s \cdots s$ rectangle and every $t \cdots t$ rectangle contains one row. By Lemma 2.8, any $S' \cup T'$ obtained from $S \cup T$ by a sequence of switches has staircases. But then by Lemma 2.9, either $S' \cup T'$ has a switch, or S' and T' are tableaux with S' extending T'. ■

Besides proving 1 of Theorem 2.2 we have shown that at every intermediate step in the switching procedure there is a method we can employ to locate an s and t to switch. What remains is to prove part 4 of Theorem 2.2, i.e., that the results of the algorithm do not depend on the sequence of switches used. Our strategy is to reduce to the case where the tableaux are standard.

Extending our notation, let us write $\widehat{U}$ for the standard renumbering of the perforated tableau U. We require that when we renumber several perforated tableaux we do so in a way that guarantees the largest integer assigned to each is the same. Remembering the content of a perforated tableau U allows us to recover U from $\widehat{U}$ in the obvious way.

Suppose $S' \cup T'$ is a perforated pair which the switch $s \leftrightarrow t$ transforms into $S'' \cup T''$. Moreover, suppose when S' and T' are renumbered to produce $\widehat{S'}$ and $\widehat{T'}$, s and t become $\widehat{s}$ and $\widehat{t}$. It is not hard to see that $\widehat{s} \leftrightarrow \widehat{t}$ is a switch for $\widehat{S'} \cup \widehat{T'}$, and the following diagram commutes:

$$\begin{array}{ccc} S' \cup T' & \xrightarrow{\text{renumber}} & \widehat{S'} \cup \widehat{T'} \\ {\scriptstyle s \leftrightarrow t}\downarrow & & \downarrow{\scriptstyle \widehat{s} \leftrightarrow \widehat{t}} \\ S'' \cup T'' & \xrightarrow{\text{renumber}} & \widehat{S''} \cup \widehat{T''} \end{array}$$

Proof of 4 of Theorem 2.2. As in the definition of the switching procedure, let S and T be tableaux such that T extends S. Consider the effect of applying the switching procedure to $S \cup T$ twice, each time with a different sequence of switches.

Suppose we know the assertion to be true for standard tableaux. If we perform two different sequences of switches on S and T, then using our commutative diagram, we get two sequences of switches that move $\widehat{S}$ and $\widehat{T}$ through one another, and the end result of those sequences must be the same. Since every tableau can be recovered from its standard renumbering, the final result of the two original sequences must also be the same. We can, therefore, assume S and T are standard, say with integers $s = 1, 2, \ldots, m$ and $t = 1, 2, \ldots, n$ respectively.

We induct on $t - s$ to show the following:

1. If $s \leftrightarrow t$ is a switch that occurs in one sequence, then it occurs in the other.

2. When $s \leftrightarrow t$ occurs in both, the boxes that s and t occupy immediately before their switch (and hence immediately after) are the same for both sequences.

First note that since s's move to the south and east and t's to the north and west, and every switch produces a perforated pair, it must be the case that each s switches with an increasing sequence of t's. Similarly, each t switches with a decreasing sequence of s's.

We begin our induction. For $t-s < 1-m$ there are no switches, so the hypotheses hold vacuously.

Suppose that 1 and 2 hold for $t-s < k$, and let $t-s = k$. Assume $s \leftrightarrow t$ occurs in the first sequence, and just prior to their switch s and t occupy boxes b and c respectively. We show this is also the case for the second sequence. First let us establish that at some point in the second sequence s occupies b. If s occupies b in $S \cup T$ this is clearly the case, so suppose the first sequence contains a switch $s \leftrightarrow t'$ that moves s into b. Necessarily $t' < t$, so by the first induction hypothesis $s \leftrightarrow t'$ occurs in the second sequence as well. Similarly, at some point in the second sequence t occupies c. Some switch in the second sequence must move s from b or t from c; otherwise, as b is northwest of c, the end result of the second sequence would not be fully switched. Suppose that t is the first to move, switching with s''. If $s'' > s$, then the first induction hypothesis implies $s'' \leftrightarrow t$ occurs in the first sequence; this is a contradiction since $s \leftrightarrow t$ is the switch in the first sequence that moves t from c. Thus $s'' \leq s$. On the other hand, the switch $s'' \leftrightarrow t$ leaves s'' in c, and c is southeast of s. Since switches produce perforated pairs, $s'' \geq s$. This forces $s'' = s$. ∎

3. Properties and Applications of Switching

In this section we list some properties of switching and show they afford a unified approach to proving a large number of combinatorial identities.

Theorem 3.1. *Suppose S and T are tableaux, T extends S, and switching S with T transforms S into ${}^{T}S$ and T into T_S. Then*

1. *S and S_T are Knuth equivalent.*

2. *T and ${}^{S}T$ are Knuth equivalent.*

3. *If S is standard (respectively, column strict, positive, LR), so is S_T.*

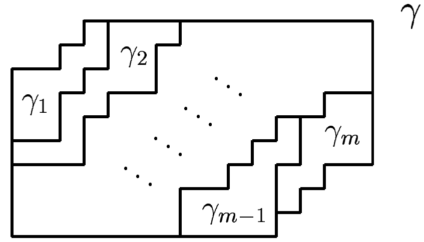
4. *If T is standard (respectively, column strict, positive, LR), so is ${}^{S}T$.*

5. *Switching ${}^{S}T$ with S_T transforms ${}^{S}T$ into T and S_T into S.*

Proof. Items 2 and 4 are analogous to 1 and 3, and we leave their proofs to the reader. From 4 of Theorem 2.2 and the definition of $j_T(S)$ it follows that $S_T = j_T(S)$. This and the definition of Knuth equivalence imply 1. Part 3 follows from 1. For 5, assume we obtain ${}^S T$ and S_T by applying the switching algorithm to S and T with some sequence of switches. When reversed, this sequence moves ${}^S T$ and S_T through one another, transforming ${}^S T$ into T and S_T into S. Item 5 then follows from 4 of Theorem 2.2. ◾

When one of S and T is row strict the other is column strict and t-switching is used instead of switching, then results similar to those in Theorem 3.1 can be shown to hold.

These properties can be exploited to prove a wide variety of identities. The method is based on the following observation. Suppose we start with a skew shape γ and break it into skew subshapes $\gamma_1, \ldots, \gamma_m$ such that γ_{i+1} extends γ_i for each i.



Assume U_i is a tableau obtained by filling γ_i with integers. We allow U_i to have different "flavors", i.e., it can be standard, column strict, positive, or LR. Switching U_i with U_{i+1} moves the two through one another in a way that preserves Knuth equivalence and the shape of their union. By 5 of Theorem 3.1, switching is an involution, and therefore a second application restores U_i and U_{i+1} to their original states. Since every permutation σ is a product $\tau_{j_1} \cdots \tau_{j_k}$ of adjacent transpositions $\tau_i = (i\ i+1)$, applying τ_{j_k}, then $\tau_{j_{k-1}}, \ldots$, then τ_{j_1} successively transforms $(U_1, \ldots, U_m)$ into $(U_1^\sigma, \ldots, U_m^\sigma)$, where for each i, U_i and $U_{\sigma(i)}^\sigma$ are Knuth equivalent and therefore share the same content and flavor.

To formalize the above let us say $\Gamma = (\gamma; \gamma_1, \ldots, \gamma_m)$ is an *m-fold multishape with outer shape* γ, and $\mathbf{U} = (U_1, \ldots, U_m)$ is an *m-fold multitableau of shape* Γ *and outer shape* γ. Then we have

Lemma 3.2. *Let σ be a permutation of $\{1, \ldots, m\}$. Then there is a bijection*

$$\mathbf{U} = (U_1, \ldots, U_m) \quad \longmapsto \quad \mathbf{U}^\sigma = (U_1^\sigma, \ldots, U_m^\sigma)$$

mapping the set of m-fold multitableaux onto itself. Under this bijection, $\mathbf{U}$ *and* $\mathbf{U}^\sigma$ *have the same outer shape, and the tableaux U_i and $U_{\sigma(i)}^\sigma$ are Knuth equivalent for each i.*

Our strategy should now be clear: by choosing m, γ, and different flavors for the U_i we obtain a combinatorial identity; Lemma 3.2 supplies the proof. The list of possible identities is extensive, and we content ourselves with some examples.

Example 3.3 (The skew Littlewood-Richardson rule). Suppose γ is the partition λ, $m = 2$, γ_1 is the partition ν, and hence γ_2 is λ/ν. Consider the multitableaux $\mathbf{U} = (U_1, U_2)$ of shape $\Gamma = (\gamma; \gamma_1, \gamma_2) = (\lambda; \nu, \lambda/\nu)$ where U_1 is LR and U_2 is positive. There is a unique LR tableau of shape ν, namely $Y(\nu)$. The mapping of Lemma 3.2 transforms $\mathbf{U}$ into $\mathbf{U}^\sigma = (U_1^\sigma, U_2^\sigma)$, where U_1^σ is a tableau of shape μ (for some partition μ), and U_2^σ is LR of shape λ/μ and content ν. This gives a bijection that proves the Littlewood-Richardson rule for Schur functions,

$$s_{\lambda/\nu} = \sum_{\mu} c_\nu^{\lambda/\mu}\, s_\mu.$$

(Here $c_\nu^{\lambda/\mu}$ is the usual Littlewood-Richardson coefficient, i.e., the number of LR tableaux of skew shape λ/μ and content ν.) This is essentially the proof in [JK].

Example 3.4 (The generalized skew Littlewood-Richardson rule). Let us broaden Example 3.3 slightly. This time we again take $m = 2$ but allow γ to be an arbitrary skew shape. Fix a partition ν and consider the multitableaux $\mathbf{U} = (U_1, U_2)$ of shape $\Gamma = (\gamma; \gamma_1, \gamma_2)$ where U_1 is LR of content ν and U_2 is positive. Basically the same argument as in Example 3.3 shows

$$\sum_{\Gamma=(\gamma;\gamma_1,\gamma_2)} c_\nu^{\gamma_1} s_{\gamma_2} = \sum_{\Gamma'=(\gamma;\delta_1,\delta_2)} c_\nu^{\delta_2} s_{\delta_1},$$

which generalizes the skew Littlewood-Richardson rule. This identity can be found in [W].

Example 3.5 (SuperSchur functions). If $\mathbf{x}$ stands for the infinitely many variables $x_1, x_2, \ldots$, and $\mathbf{y}$ for the infinitely many variables $y_1, y_2, \ldots$, the superSchur function (hook Schur function) corresponding to the partition λ is given by

$$HS_\lambda(\mathbf{x}, \mathbf{y}) = \sum_{U_1, U_2} \mathbf{x}^{U_1} \mathbf{y}^{U_2}.$$

The sum ranges over all partitions $\kappa \subseteq \lambda$, all positive tableaux U_1 of shape κ, and all positive row strict tableaux U_2 of shape λ/κ. Consider the multitableaux $U = (U_1, U_2)$ of shape $\Gamma = (\lambda; \kappa, \lambda/\kappa)$, where κ, U_1, and U_2 are as above. The mapping of Lemma 3.2 (adjusted suitably for a mixture of row and column strict tableaux) transforms U to $U^\sigma = (U_1^\sigma, U_2^\sigma)$ where U_1^σ is a row strict tableau of shape (say) ν^t and U_2^σ is a tableau of shape λ/ν^t. Suppose $V = (V_1, V_2)$ where V_1 is $Y(\nu^t)$ and $V_2 = U_2^\sigma$. Applying the mapping of Example 3.3 to V gives (V_1^σ, V_2^σ), where V_1^σ is a tableau of shape μ and V_2^σ is an LR tableau of shape λ/μ and content ν^t. The map $U = (U_1, U_2) \longrightarrow (T, V_1^\sigma, V_2^\sigma)$ where $T = (U_1^\sigma)^t$ is a bijection that establishes the identity

$$HS_\lambda(\mathbf{x}, \mathbf{y}) = \sum_{\mu, \nu^t \subseteq \lambda} c_{\nu^t}^{\lambda/\mu} s_\mu(\mathbf{x}) s_\nu(\mathbf{y}).$$

(Compare [R], Eqn. 1.3 and [BL], §3.)

Example 3.6 (Symmetries of Littlewood-Richardson coefficients). As in Example 3.3 let us again take $m = 2$, $\gamma = \lambda$, $\gamma_1 = \nu$ and $\gamma_2 = \lambda/\nu$. As before let U_1 be LR, but this time assume U_2 is LR as well, say of content μ. Our mapping transforms $\mathbf{U} = (U_1, U_2)$ into $\mathbf{U}^\sigma = (U_1^\sigma, U_2^\sigma)$, where U_1^σ and U_2^σ are LR, and U_1^σ has partition shape. But this forces U_1^σ to have shape μ, and therefore U_2^σ is LR of shape λ/μ and content ν. Not only have we proven $c_\nu^{\lambda/\mu} = c_\mu^{\lambda/\nu}$; we have displayed an explicit involution that interchanges the inner shape and content of an LR tableau. A brief description of this mapping based upon the algorithm of James and Kerber can be found in [W].

Example 3.7 (Generalized Littlewood-Richardson coefficients). Let us expand on the ideas in Example 3.6. Given a skew shape γ and partitions $\nu_1, \ldots, \nu_m$, let $c_{\nu_1 \cdots \nu_m}^\gamma$ be the number of m-fold multitableaux for which the outer shape is γ and the i^{th} tableau is LR of content ν_i. These "generalized" Littlewood-Richardson coefficients are related to the ordinary ones by the formula

$$c_{\nu_1 \cdots \nu_m}^\gamma = \sum_{\Gamma = (\gamma; \gamma_1, \ldots, \gamma_m)} c_{\nu_1}^{\gamma_1} \cdots c_{\nu_m}^{\gamma_m}.$$

Suppose $\mathbf{U} = (U_1, \ldots, U_m)$ has outer shape γ, each U_i is LR of content ν_i, and σ is a permutation of $\{1, \ldots, m\}$. Then by Lemma 3.2 $\mathbf{U}^\sigma = (U_1^\sigma, \ldots, U_m^\sigma)$ also has outer shape γ, and each $U_{\sigma(i)}^\sigma$ is LR of content ν_i. This proves

$$c_{\nu_1 \cdots \nu_m}^\gamma = c_{\nu_{\sigma(1)} \cdots \nu_{\sigma(m)}}^\gamma.$$

Since $c_\nu^{\lambda/\mu} = c_{\mu\nu}^\lambda$, this generalizes Example 3.6.

It is interesting to note that the identities arising in Examples 3.3, 3.6, and 3.7 make it possible to deduce representation theoretic results such as branching rules. We illustrate this with the following example.

Example 3.8 (Branching rule). Assume $\underline{\mathbf{x}} = \{x_1, \ldots, x_m\}$, and consider the evaluation of the Schur function s_γ given by

$$s_\gamma(\underline{\mathbf{x}}) = s_\gamma \mid_{x_{m+1}=x_{m+2}=\cdots=0} .$$

Thus, $s_\gamma(\underline{\mathbf{x}}) = \sum \mathbf{x}^U$ where the sum is over all tableaux U of shape γ whose content is contained in $\{1, \ldots, m\}$. Suppose $m_1, \ldots, m_k$ are positive integers such that $m_1 + \cdots + m_k = m$, and let $\underline{\mathbf{x}}^{(i)} = \{x_{M_{i-1}+1}, \ldots, x_{M_i}\}$ for $i = 1, \ldots, k$, where $M_0 = 0$ and $M_i = m_1 + \cdots + m_i$. In each tableau of partition shape λ with entries in $\{1, \ldots, m\}$ there is a subtableau of some skew shape, say γ_i, which contains the entries in $\{M_{i-1}+1, \ldots, M_i\}$, and γ_{i+1} extends γ_i for each $i = 1, \ldots, k$. As a result, it follows that

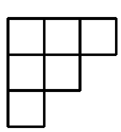
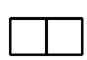
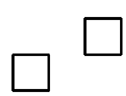
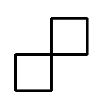
$$s_\lambda(\underline{\mathbf{x}}) = \sum_{\Gamma=(\lambda;\gamma_1,\ldots,\gamma_k)} s_{\gamma_1}(\underline{\mathbf{x}}^{(1)}) \cdots s_{\gamma_k}(\underline{\mathbf{x}}^{(k)}).$$

By Examples 3.3 and 3.6 $s_\gamma = \sum_\nu c_\nu^\gamma s_\nu$, which can be combined with Example 3.7 to give

$$\begin{aligned} s_\lambda(\underline{\mathbf{x}}) &= \sum_{\substack{\Gamma=(\lambda;\gamma_1,\ldots,\gamma_k)\\ \nu_1,\ldots,\nu_k}} c_{\nu_1}^{\gamma_1} c_{\nu_2}^{\gamma_2} \cdots c_{\nu_k}^{\gamma_k} s_{\nu_1}(\underline{\mathbf{x}}^{(1)}) \cdots s_{\nu_k}(\underline{\mathbf{x}}^{(k)}) \\ &= \sum_{\nu_1,\ldots,\nu_k} c_{\nu_1,\ldots,\nu_k}^{\lambda} s_{\nu_1}(\underline{\mathbf{x}}^{(1)}) \cdots s_{\nu_k}(\underline{\mathbf{x}}^{(k)}). \end{aligned}$$

The irreducible polynomial representations for the general linear group $GL(m)$ are in one-to-one correspondence with the partitions λ having length $\leq m$, and $s_\lambda(\underline{\mathbf{x}})$ is the character of the irreducible polynomial $GL(m)$-representation labeled by λ. Thus, the identity derived above is just the branching rule for $GL(m)$ to the subgroup $GL(m_1) \times \cdots \times GL(m_k)$ (or equivalently, for the general linear Lie algebra $gl(m)$ to the subalgebra $gl(m_1) \times \cdots \times gl(m_k)$). This identity holds equally well with λ replaced by any skew shape γ.

We conclude this section by taking a closer look at the bijection of Lemma 3.2. We defined the mapping $\mathbf{U} = (U_1, \ldots, U_m) \longmapsto \mathbf{U}^\sigma = (U_1^\sigma, \ldots, U_m^\sigma)$ by factoring σ into a product of adjacent transpositions, and it is natural to ask whether the bijection depends on the factorization. Unfortunately, as the next example shows, the answer is yes. Let $\Gamma = (\gamma; \gamma_1, \gamma_2, \gamma_3)$ and $\mathbf{U} = (U_1, U_2, U_3)$ be defined by the pictures below.

$\gamma =$  $\gamma_1 =$  $\gamma_2 =$  $\gamma_3 =$ 

$$\mathbf{U} = \begin{array}{|c|c|c|}\hline 1 & 1 & 2 \\ \hline 2 & 3 \\ \cline{1-2} 3 \\ \cline{1-1}\end{array} \qquad U_1 = \begin{array}{ccc} 1 & 1 & \\ & & \\ & & \end{array} \qquad U_2 = \begin{array}{ccc} & & 2 \\ 2 & & \\ & & \end{array} \qquad U_3 = \begin{array}{ccc} & & \\ & 3 & \\ 3 & & \end{array}$$

Now $\sigma = (13)$ factors as both $(12)(23)(12)$ and $(23)(12)(23)$, and applying the corresponding mappings to $\mathbf{U}$ yields

$$\begin{array}{ccc} 1 & 1 & 2 \\ 2 & 3 & \\ 3 & & \end{array} \xmapsto{(12)} \begin{array}{ccc} 2 & 2 & 1 \\ 1 & 3 & \\ 3 & & \end{array} \xmapsto{(23)} \begin{array}{ccc} 2 & 2 & 1 \\ 3 & 3 & \\ 1 & & \end{array} \xmapsto{(12)} \begin{array}{ccc} 3 & 3 & 1 \\ 2 & 2 & \\ 1 & & \end{array}$$

$$\begin{array}{ccc} 1 & 1 & 2 \\ 2 & 3 & \\ 3 & & \end{array} \xmapsto{(23)} \begin{array}{ccc} 1 & 1 & 2 \\ 3 & 3 & \\ 2 & & \end{array} \xmapsto{(12)} \begin{array}{ccc} 3 & 3 & 2 \\ 1 & 1 & \\ 2 & & \end{array} \xmapsto{(23)} \begin{array}{ccc} 3 & 3 & 2 \\ 2 & 1 & \\ 1 & & \end{array}$$

As the results are different, the mapping in Lemma 3.2 depends on the factorization.

4. Dual Equivalence

In [H2] Haiman introduces the notion of *dual equivalence* for standard tableaux and notes that most of his results extend to column strict tableaux. Here, to lay the groundwork for the next section, we describe these extensions explicitly.

Boxes $b_1, b_2, \ldots, b_k$ define a *sequence of slides* for a tableau U if it meaningful to form $U = U_0, U_1, \ldots, U_k$, where b_i is a corner of U_{i-1}, and U_i is the tableau that results when we perform a slide starting at b_i on U_{i-1}. Following Haiman [H2], we define tableaux U and V to be *dual equivalent* if every sequence of slides for U is a sequence of slides for V, and vice-versa. We write $U \overset{\mathrm{d}}{\cong} V$ to indicate U and V are dual equivalent. Dual equivalent tableaux have the same corners and therefore have the same shape.

In large measure Haiman's results concerning dual equivalence carry over into the column strict world without change, and for the most part the same proofs work. The fact that bridges the gap is the following.

Lemma 4.1. *Any tableau U is dual equivalent to its standard renumbering $\widehat{U}$.*

Proof. We intend to induct on the number of boxes in U; however, to do so we need some machinery. Recall that to say subtableaux V and W decompose U means that W extends V, and whenever v and w are integers from V and W respectively, either $v < w$, or $v = w$ and v is west of w. Let U and U' be tableaux of the same shape, and suppose γ_1 and γ_2 are shapes such that γ_2 extends γ_1 and $\operatorname{sh} U = \operatorname{sh} U' = \gamma_1 \cup \gamma_2$. Assume U and U' have decompositions $U = V \cup W$ and $U' = V' \cup W'$, where $\operatorname{sh} V = \operatorname{sh} V' = \gamma_1$ and $\operatorname{sh} W = \operatorname{sh} W' = \gamma_2$. Then the following is a simple consequence of ([H2], Lemma 2.1): if $V \overset{\mathrm{d}}{\cong} V'$ and $W \overset{\mathrm{d}}{\cong} W'$ then $U \overset{\mathrm{d}}{\cong} U'$.

Now we start the induction. When U contains fewer than two boxes the assertion is obviously true, so suppose U has at least two boxes and let V and W be nontrivial tableaux that decompose U. Then $\widehat{U} = \widehat{V} \cup \widehat{W}$, provided when we renumber we choose maximum values appropriately. By the induction hypothesis, $V \overset{\mathrm{d}}{\cong} \widehat{V}$ and $W \overset{\mathrm{d}}{\cong} \widehat{W}$, so $U \overset{\mathrm{d}}{\cong} \widehat{U}$. ∎

Lemma 4.1 shows the following result of Haiman remains true when the tableaux are column strict:

Proposition 4.2 ([H2] Corollary 2.5). *Let U and V be tableaux of the same normal shape. Then U and V are dual equivalent.*

With Proposition 4.2 in hand, the proofs in [H2] can be applied to give

Theorem 4.3 ([H2] Corollaries 2.8 and 2.9). *Let U and V be dual equivalent tableaux.*

1. *If W is any tableau that extends U (or equivalently, extends V), then ${}^{U}W = {}^{V}W$ and $U_W \overset{\mathrm{d}}{\cong} V_W$.*
2. *If W is any tableau that U extends (or equivalently, V extends), then $W_U = W_V$ and ${}^{W}U \overset{\mathrm{d}}{\cong} {}^{W}V$.*

Perhaps the deepest of Haiman's results on dual equivalence is that tableaux are uniquely characterized by dual and Knuth equivalence.

By Theorem 4.3, dual equivalence shares with Knuth equivalence the following property: Any two equivalent tableaux have normal forms with the same shape. Thus, to each dual or Knuth equivalence class there corresponds a unique normal shape.

Theorem 4.4 ([H2] Theorem 2.13). *Let $\mathcal{D}$ be a dual equivalence class and $\mathcal{K}$ be a Knuth equivalence class, both corresponding to the same normal shape. Then there is a unique tableau in $\mathcal{D} \cap \mathcal{K}$.*

Proof. Haiman's arguments work without change, but we present his proof of the existence of the tableau since it involves a construction to be used in §5. Let $U \in \mathcal{D}$ be any representative, and $V \in \mathcal{K}$ be the unique representative with normal shape. We construct the unique tableau in $\mathcal{D} \cap \mathcal{K}$ as follows:

1. Choose any tableau W such that U extends W and $W \cup U$ has normal shape.

2. Switch W with U to produce $U^{\mathrm{n}} = {}^{W}U$ and W_U.

3. Since $\mathrm{sh}\, V = \mathrm{sh}\, U^{\mathrm{n}}$ we can switch V with W_U to obtain the tableau V_{W_U}.

By Theorem 4.3, V_{W_U} is dual equivalent to U, and by Theorem 3.1 it is Knuth equivalent to V. ∎

In light of this theorem the usefulness of the Littlewood-Richardson tableaux is apparent: they form a complete transversal for the set of dual equivalence classes. Moreover, the Littlewood-Richardson coefficient c^{γ}_{μ} counts the number of dual equivalence classes of tableaux of shape γ whose normal shape is μ.

Haiman establishes a number of other dual equivalence results that can also be transferred to column strict tableaux; however, the above suffice for our purposes.

5. Evacuation, Reversal, and Related Mappings

In this section we describe Schützenberger's algorithm [Sc1] for evacuating a tableau of normal shape and show using switching how this algorithm can be generalized. For tableaux of normal shape we prove the evacuation is the normal form of the rotation. This leads to two properties that characterize the evacuation of a tableau of normal shape and motivates the definition of a mapping called *reversal* that operates upon tableaux of arbitrary shape. In [Sc2] Schützenberger extends his evacuation algorithm so it can be applied to tableaux of arbitrary shape. For tableaux of normal shape, reversal and evacuation agree, though for general tableaux they yield different results. The mapping $U \longmapsto U^{\circ}$ is defined and shown to be closely related to both reversal and $*$. We discuss the White-Hanlon-Sundaram map $U \longmapsto U^{\mathrm{WHS}}$ ([W], [HS]), which transforms LR tableaux of shape γ and content μ into LR tableaux of shape γ^{t} and content μ^{t}. The section closes with a proof that the symmetries of Littlewood-Richardson coefficients observed by Berenstein and Zelevinsky [BZ] follow from §3 and ([W], [HS]).

We start by recalling some notation. Whenever U is a tableau, U^{n} (respectively, U^{a}) is the unique tableau of normal (respectively, anti-normal) shape Knuth equivalent to U. The rotation U^* is the tableau obtained by rotating the shape of U by 180° and replacing each integer u by $-u$. If γ is a skew shape, then γ^{t} is the transpose of γ;

similarly, when U is a tableau, U^{t} is its transpose. We write $Y(\lambda)$ for the LR tableau obtained by filling the first row of λ with 1's, the second with 2's, and so on.

Algorithm 5.1 (Schützenberger's evacuation algorithm). *Let U be a tableau of normal shape. The following algorithm transforms U into U^{E}, the* evacuation *of U:*

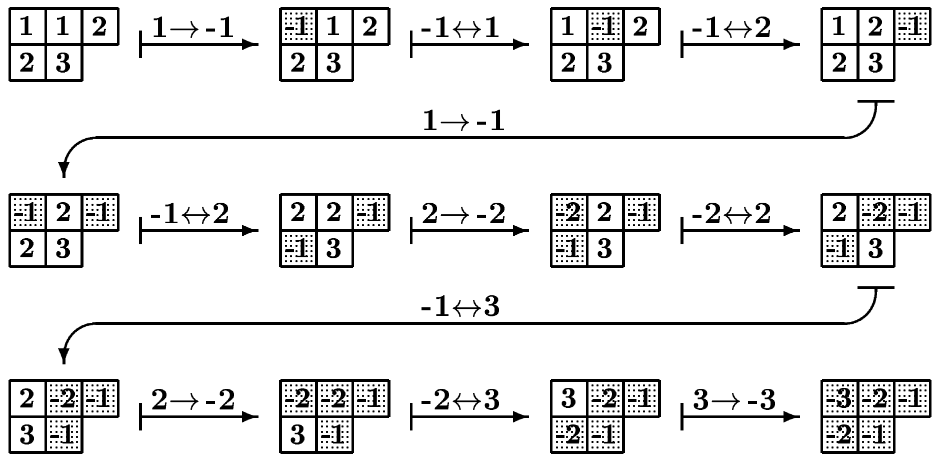
1. *Replace the integer u at the northwest corner of U with $-u$ and mark the new integer.*
2. *Use a contracting slide to move the marked integer through the tableau formed by the unmarked integers.*
3. *Repeat steps 1 and 2 until every integer has been marked.*

In §2 we generalized the algorithms of Haiman [H1], Shimozono [Sh], James and Kerber [JK], and Remmel [R] with the switching procedure, and it is natural to do something similar here. Roughly speaking, the idea is that at each step the marked and unmarked integers form a perforated pair. The algorithm converts unmarked integers to marked ones and switches marked with unmarked integers, stopping when no more conversions or switches are possible.

Algorithm 5.2 (Generalized evacuation). *Let U be a tableau of normal shape λ.*

1. *Start with every integer in U unmarked.*
2. *Do one of the following:*
   (a) *If the integer u at the northwest corner of λ is unmarked, replace it with $-u$ and mark the new integer, or*
   (b) *Switch some marked integer with an unmarked one.*
3. *Repeat 2 until no more switches or conversions are possible.*

For brevity let us say that every time we perform step 2 of the algorithm we have made a *move*. We write $t \to s$ for a move that converts an unmarked t into a marked s, and $s \leftrightarrow t$ for a move that switches s with t. The following shows one possible sequence of moves the algorithm could use to transform a tableau:



There are several facts we must prove about generalized evacuation. We define staircases as in the discussion preceding the proof of Lemma 2.8.

Theorem 5.3. *Suppose that after performing a (possibly empty) sequence of moves in Algorithm 5.2 we obtain S' and T' where S' consists of the marked integers and T' of the unmarked ones. Then $S' \cup T'$ is a perforated pair with staircases.*

Proof. We induct on the number of moves. The initial case is trivial, and the case in which the last move is a switch follows from Lemma 2.8. Consider what happens if the last move converts an unmarked t into a marked $s = -t$.

Start by supposing S' contains exactly one copy of s. The smallest integer in S' is s, so $S' \cup T'$ is a perforated pair. The conversion of t into $s = -t$ might have destroyed a $t \cdots t$ rectangle, but could not have created one. Since S' contains only one s, the conversion could not have created any $s \cdots s$ rectangles. Putting these facts together we see that $S' \cup T'$ is a perforated pair with staircases.

Now suppose S' contains $k \geq 2$ copies of s. Then S' and T' were obtained by applying a sequence of moves to a perforated pair $S'' \cup T''$ where S'' contained $k-2$ copies of s. We can assume the first of these moves converted a t to a marked $s = -t$. Note this conversion must have destroyed a $t \cdots t$ rectangle. By the induction hypothesis the conversion produced a perforated pair with staircases. Next came a nonempty sequence of switches. Among these were ones which moved the t originally in the southeast corner of the rectangle to the northwest, eventually switching t with the s produced by the conversion mentioned above. Just before they switched, the two must have been side-by-side, so the switch slid t west and s east. (We can be sure of this because the staircase from the destroyed rectangle prevented t from moving into the same column as s.) The final move converted t, now in the northwest corner, to an s. It follows that S' contains exactly one s in its westernmost column. Since no integer in S' is smaller than s, $S' \cup T'$ is a perforated pair. Reasoning as in the proof of Lemma 2.8, we see that the newly created s and the first-formed one are the northwest and southeast corners respectively of an $s \cdots s$ rectangle with a staircase. The second conversion of t to $s = -t$ might have destroyed a $t \cdots t$ rectangle, but it could not have created one. This proves $S' \cup T'$ is a perforated pair with staircases. ∎

Theorem 5.4.

1. *When Algorithm 5.2 stops, every integer is marked.*

2. *The end result of the algorithm is independent of the sequence of moves used.*

3. *Schützenberger's evacuation procedure is a special case of Algorithm 5.2.*

Proof. Whenever W is a perforated tableau, let us write $\widehat{W}$ for the standard perforated tableau obtained by renumbering W as in §2. Again we require that when several perforated tableaux are renumbered, the same largest integer must be assigned to each.

For part 1 assume that after performing a (possibly empty) sequence of moves we have obtained perforated tableaux S' and T' of of marked and unmarked integers respectively. By Theorem 5.3, $S' \cup T'$ is a perforated pair with staircases. It is enough to show that if S' is not fully expanded (or equivalently, T' is not fully contracted), there is a switch $s \leftrightarrow t$ for $S' \cup T'$. This follows directly from Lemma 2.9.

To prove 2, assume at some point the algorithm has produced the perforated pair $S' \cup T'$ of marked and unmarked integers, and the move m turns $S' \cup T'$ into $S'' \cup T''$. There is a corresponding move $\widehat{m}$ that transforms $\widehat{S'} \cup \widehat{T'}$ into $\widehat{S''} \cup \widehat{T''}$, and it is not hard to see the following diagram commutes:

$$\begin{array}{ccc} S' \cup T' & \xrightarrow{\text{renumber}} & \widehat{S'} \cup \widehat{T'} \\ m \downarrow & & \downarrow \widehat{m} \\ S'' \cup T'' & \xrightarrow{\text{renumber}} & \widehat{S''} \cup \widehat{T''} \end{array}$$

Suppose we know 2 to be true for standard tableaux, and let U be column strict. Assume we apply the algorithm to U twice, each time with a different sequence of moves. Using our commutative diagram, we obtain two corresponding sequences for $\widehat{U}$, and the end result of each sequence must be the same. But every tableau can be recovered from its standard renumbering, so the final result of the two original sequences must also be the same. We are therefore free to assume U is standard, say with integers $1, 2, \ldots, p$. Let us write t's for the unmarked integers the algorithm consumes and s's for marked ones it creates. Obviously each of the two sequences contains a move that converts each t into the corresponding $s = -t$, and every other move in either sequence is a switch. To complete the proof of 2, use induction on $t + s$ to show

1. If $s \leftrightarrow t$ is a switch that occurs in one sequence, then it occurs in the other.
2. When $s \leftrightarrow t$ occurs in both, the boxes s and t occupy immediately before their switch (and hence immediately after) are the same for both sequences.

The induction is virtually identical to that used to prove 4 of Theorem 2.2, and we leave the details to the reader.

Finally, to see 3, observe Schützenberger's evacuation algorithm is the special case of the generalized evacuation algorithm obtained by consistently deferring conversions of unmarked integers into marked ones as long as possible. ▪

The next result shows how evacuation and rotation are related.

Theorem 5.5. *Let U be a tableau of normal shape. Then $U^{\mathrm{E}} = U^{*\mathrm{n}}$.*

Proof. We induct on the number of boxes in U. When U is empty or consists of a single box, the assertion is obvious, so suppose it contains more than one. Let u be the integer at the northwest corner of U and let V be the tableau that results when this u is deleted.

We can transform U into U^{E} by converting the unmarked u at the northwest corner to a marked $-u$, switching to move this $-u$ to the outer edge, and then applying the algorithm to what remains of U, i.e., to the normal form W of V. It follows U^{E} can be obtained by adjoining a $-u$ at an outer corner of W^{E}.

There is a $-u$ at the southeast corner of U^*, and if we delete this $-u$, the tableau we obtain is V^*. It follows that $U^{*\mathrm{n}}$ consists of $V^{*\mathrm{n}}$ extended by $-u$. But $V^{*\mathrm{n}} \overset{\mathrm{k}}{\cong} W^*$, and by the induction hypothesis $W^{*\mathrm{n}} = W^{\mathrm{E}}$, so $V^{*\mathrm{n}} = W^{\mathrm{E}}$. We know $\operatorname{sh} U^{\mathrm{E}} = \operatorname{sh} U$, and in Lemma 5.6 below we prove $\operatorname{sh} U^{*\mathrm{n}} = \operatorname{sh} U$. Then the $-u$ that extends W^{E} in U^{E} must occur at the same position as the $-u$ that extends W^{E} in $U^{*\mathrm{n}}$, and therefore $U^{\mathrm{E}} = U^{*\mathrm{n}}$. ▪

Lemma 5.6. *Let U be a tableau of normal shape. Then $\operatorname{sh} U^{*\mathrm{n}} = \operatorname{sh} U$.*

Proof. Before inducting on the number of boxes in U we need to show the following: if V is a subtableau of U derived by discarding one box b from the southeast edge of U, then $\operatorname{sh} U^{*\mathrm{n}} \supset \operatorname{sh} V^{*\mathrm{n}}$. First consider the case where U is standard and the discarded box b contains the largest integer u in U. Then $-u$ is the integer in the northwest corner of $U^{*\mathrm{n}}$, and $V^{*\mathrm{n}}$ is produced if we erase $-u$ and use a contracting slide to move the empty box through what remains of $U^{*\mathrm{n}}$. It follows $\mathrm{sh}\, U^{*\mathrm{n}} \supset \mathrm{sh}\, V^{*\mathrm{n}}$. For the general case let U' be a standard tableau of shape $\mathrm{sh}\, U$ whose largest integer is in the box b, and let V' be the tableau obtained by discarding b from U'. Then $U \overset{\mathrm{d}}{\cong} U'$ and $V \overset{\mathrm{d}}{\cong} V'$, so $U^* \overset{\mathrm{d}}{\cong} (U')^*$ and $V^* \overset{\mathrm{d}}{\cong} (V')^*$. This implies $\mathrm{sh}\, U^{*\mathrm{n}} = \mathrm{sh}\,(U')^{*\mathrm{n}} \supset \mathrm{sh}\,(V')^{*\mathrm{n}} = \mathrm{sh}\, V^{*\mathrm{n}}$, as claimed.

Now we begin the induction. Observe that when U is empty or rectangular the result is clear, so we may assume U contains at least two boxes and is not a rectangle. Then there are two distinct subtableaux V and W of U, each obtained by discarding one box from the southeast edge of U. By the induction hypothesis, $\mathrm{sh}\, V^{*\mathrm{n}} = \mathrm{sh}\, V$ and $\mathrm{sh}\, W^{*\mathrm{n}} = \mathrm{sh}\, W$. Then $\mathrm{sh}\, U^{*\mathrm{n}} \supseteq \mathrm{sh}\, V^{*\mathrm{n}} \cup \mathrm{sh}\, W^{*\mathrm{n}} = \mathrm{sh}\, V \cup \mathrm{sh}\, W = \mathrm{sh}\, U$. Since $|\mathrm{sh}\, U^{*\mathrm{n}}| = |\mathrm{sh}\, U|$ this forces $\mathrm{sh}\, U^{*\mathrm{n}} = \mathrm{sh}\, U$. ▪

It follows easily from [Sc1] that $U^* \overset{\mathrm{k}}{\cong} U^{\mathrm{E}}$, and Fulton [F] proves $U^{*\mathrm{n}} = U^{\mathrm{E}}$. Both arguments are based on Schensted insertion using the words of the tableaux rather than the approach we have presented here. Much of the importance of evacuation stems from these results. After developing our proof of Theorem 5.5 we learned Haiman has also related U^{E} to U^{a} ([Sa2])

The above results allow us to characterize U^{E} as follows.

Theorem 5.7. *Let U be a tableau of normal shape. Then U^{E} is the unique tableau Knuth equivalent to U^* and dual equivalent to U.*

Proof. Whenever V and W are Knuth (respectively, dual) equivalent, V^* and W^* are as well. Since $U^{\mathrm{E}} = U^{*\mathrm{n}}$, U^{E} is Knuth equivalent to U^*. Also, U^{E} and U are tableaux of the same normal shape, so are dual equivalent by Proposition 4.2. Theorem 4.4 says there can be only one tableau with these properties. ▪

There is a simple way to extend evacuation to tableaux of arbitrary skew shape: in Algorithm 5.1, rather than saying

"replace the integer u at the northwest corner of U with $-u$ and mark the new integer"

use instead

"replace the smallest integer u in U with $-u$ and mark the new integer (if there is more than one such u, take the easternmost one)".

Schützenberger [Sc2] and Haiman [H2] study evacuation in this broader context. The mapping $U \longmapsto U^{\mathrm{E}}$ of the new algorithm is the same as the original mapping when restricted to tableaux U having normal shape, and the new mapping $U \longmapsto U^{\mathrm{E}}$ is an involution. However, when this mapping is applied to a skew tableau the result does not in general enjoy the properties indicated in Theorem 5.7.

In operating upon tableaux of arbitrary shape, we follow a different path. Our idea is to use Theorem 5.7 to motivate the definition.

Definition 5.8. *Let U be a tableau of arbitrary skew shape. Define U^{e}, the* reverse *of U, to be the unique tableau Knuth equivalent to U^* and dual equivalent to U. The mapping $U \longmapsto U^{\mathrm{e}}$ is called* reversal.

The construction in the proof of Theorem 4.4 gives an algorithm for computing U^{e}. Let W be any tableau such that U extends W and $W \cup U$ has normal shape. Then

$$U^{\mathrm{e}} = (U^{*\mathrm{n}})_{W_U}.$$

Next we introduce a tableau U° which is closely related to U^* and U^{e}. For an arbitrary tableau U let U° be the unique tableau Knuth equivalent to U and dual equivalent to U^*. Note that since $U^\circ = U^{\mathrm{e}*}$, we have an explicit algorithm for calculating U°.

Proposition 5.9. *The mappings*

$$\{U \longmapsto U, U \longmapsto U^*, U \longmapsto U^{\mathrm{e}}, U \longmapsto U^\circ\}$$

determine an action of the Klein four group $\mathbf{Z}_2 \oplus \mathbf{Z}_2$ on the set of tableaux.

Proof. Since each mapping is its own inverse it suffices to show the set is closed under composition. Applying one or more of the mappings to U yields a tableau dual equivalent to U or U^* and Knuth equivalent to U or U^*, and hence is one of U, U^*, U^{e}, or U°. ■

It is instructive to consider an orbit of this group:

$U =$

			1	1	3
		2	2	4	5
	1	5	5	6	6
	3				
1	5				

$= U^o$

				1	5
				3	
1	1	1	2	6	
2	3	4	6		
5	5	5			

$U^e =$

			-5	-5	-5
		-6	-4	-3	-2
	-6	-2	-1	-1	-1
	-3				
-5	-1				

$= U^*$

				-5	-1
				-3	
-6	-6	-5	-5	-1	
-5	-4	-2	-2		
-3	-1	-1			

As an application of the mapping $U \longmapsto U^\circ$ let us fix a skew shape γ and consider the problem of showing the skew Schur functions s_γ and s_{γ^*} are equal. The mapping $U \longmapsto U^*$ defines a bijection between tableaux of shape γ and those of shape γ^* which can be used to show $s_\gamma = s_{\gamma^*}$. However, corresponding tableaux do not have the same content, and therefore cannot be Knuth equivalent. From a combinatorial viewpoint this is unsatisfactory; if $s_\gamma = s_{\gamma^*}$, it ought to be possible to use slides to transform tableaux of shape γ into tableaux of shape γ^*. The difficulty disappears when we use the mapping $U \longmapsto U^\circ$ in place of $U \longmapsto U^*$; corresponding tableaux are Knuth equivalent. In particular, $U \longmapsto U^\circ$ sends LR tableaux of shape γ and content μ to LR tableaux of shape γ^* and content μ, so we have an explicit combinatorial involution that proves

Proposition 5.10. *The Littlewood-Richardson coefficients c^γ_μ and $c^{\gamma^*}_\mu$ are equal.*

White [W] describes a mapping that transforms LR tableaux of shape γ and content μ into LR tableaux of shape γ^{t} and and content μ^{t}. Using an algorithm based on Schensted insertion, Hanlon and Sundaram [HS] introduce an analogous map and use it to give a bijective proof that the LR coefficients c^γ_μ and $c^{\gamma^{\mathrm{t}}}_{\mu^{\mathrm{t}}}$ are equal. As Fulton shows in [F], the White map and the Hanlon-Sundaram map produce the same result, and so we denote the map by $U \longrightarrow U^{\mathrm{WHS}}$. It is interesting to note the technique used to construct U^{e} and U° can be applied to build U^{WHS}. From the proof of Theorem 4.4 there is exactly one tableau Knuth equivalent to $Y((\operatorname{sh} U)^{\mathrm{t}})$ and dual equivalent to $(\widehat{U})^{\mathrm{t}}$. That tableau is U^{WHS}, and so the construction affords an explicit way of calculating this tableau. In essence this is the approach adopted in [W].

Identifying the Littlewood-Richardson coefficients with the number of lattice points in certain polytopes, Berenstein and Zelevinsky [BZ] prove the coefficients are symmetric under an action of the group $\mathbf{Z}_2 \times \mathbf{S}_3$. We conclude by showing the same result can be derived from the usual definition of the coefficients.

Let us fix notation. Whenever ρ is a rectangular shape, κ is a partition, and $\rho \supseteq \kappa$, we write $\kappa^{\mathrm{c}} = \kappa^{\mathrm{c}}(\rho)$ for the partition $(\rho/\kappa)^*$, and $(\kappa^{\mathrm{t}})^{\mathrm{c}}$ for $(\rho^{\mathrm{t}}/\kappa^{\mathrm{t}})^*$. Recall in Example 3.7 we defined the generalized Littlewood-Richardson coefficient $c^\gamma_{\nu_1 \cdots \nu_m}$ to be the number of m-fold multitableaux for which the outer shape is γ and the i^{th} tableau is LR of content ν_i. We proved $c^\gamma_{\nu_1 \cdots \nu_m}$ is symmetric in $\nu_1, \nu_2, \ldots, \nu_m$. Let us restrict to the case where γ is a rectangular shape ρ, $m = 3$, and ν_1, ν_2, and ν_3 are the partitions ν, μ, and λ respectively. Consider a typical multitableau obtained in this special case. Necessarily the inner tableau is $Y(\nu)$ and the outer tableau is $Y(\lambda)^\circ$. Therefore $c^\rho_{\nu\mu\lambda} = c^{\lambda^{\mathrm{c}}/\nu}_\mu = c^{\lambda^{\mathrm{c}}}_{\nu\mu}$ counts the number of middle tableaux; this is the number of LR tableaux of shape λ^{c}/ν and content μ. In light of Example 3.7, $c^{\lambda^{\mathrm{c}}}_{\nu\mu}$ is symmetric in ν, μ, and λ. Putting this together with the symmetries given by the White-Hanlon-Sundaram map, we have proven

Proposition 5.11 (Berenstein-Zelevinsky). *The Littlewood-Richardson coefficient* $c_{\nu\mu}^{\lambda^c}$ *is symmetric under the following action of the group* $\mathbf{Z}_2 \times \mathbf{S}_3$*: the nonidentity element of* $\mathbf{Z}_2$ *simultaneously transposes each of* ν*,* μ*, and* λ*, and* $\mathbf{S}_3$ *permutes* ν*,* μ*, and* λ*.*

Note this gives another way to derive Proposition 5.10. Fix a skew shape γ and choose ρ, λ, and ν such that $\gamma = \lambda^c/\nu$. Then $\gamma^* = (\nu^c/\lambda)$, so $c_\mu^\gamma = c_\mu^{\gamma^*}$ by Proposition 5.11.

Acknowledgments. The authors would like to thank William Fulton, Bruce Sagan, and Sheila Sundaram for their helpful advice and encouragement.

References

[BK] E. Bender and D. Knuth, Enumeration of plane partitions, J. Combin. Theory Ser. A **13** (1972), 40-54.

[BL] G. Benkart and C. Lee, Stability in modules for general linear Lie superalgebras, Nova J. Algebra and Geometry **2** (1993), 383-409.

[BZ] A.D. Berenstein, A.V. Zelevinsky, Triple multiplicities for $\mathfrak{sl}(\mathfrak{r}+1, \mathbf{C})$ and the spectrum of the exterior algebra of the adjoint representation, J. Algebraic Combin. **1** (1992) 7-22.

[F] W. Fulton, *Young Tableaux, with Applications to Representation Theory and Geometry*, Draft of a book (1992).

[H1] M. Haiman, On mixed insertion, symmetry, and shifted Young tableaux, J. Combin. Theory Ser. A **50** (1989) 196-225.

[H2] M. Haiman, Dual equivalence with applications, including a conjecture of Proctor, Discrete Math. **99** (1992) 79-113.

[HS] P. Hanlon and S. Sundaram, On a bijection between Littlewood-Richardson fillings of conjugate shape, J. Combin. Theory Ser. A **60** (1992) 1-18.

[JK] G. James and A. Kerber, *The Representation Theory of the Symmetric Group*, Encyclopedia of Mathematics and its Applications **16** Addison-Wesley, Reading, Massachusetts (1981).

[R] J. Remmel, The combinatorial (k, l)-hook Schur functions, Contemp. Math. **34** (1984) 253-287.

[Sa1] B.E. Sagan, *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions*, Wadsworth and Brooks/Cole, Pacific Grove, California (1991).

[Sa2] B.E. Sagan, conversation.

[Sc1] M.-P. Schützenberger, Quelques remarques sur une construction de Schensted, Math Scand. **12** (1963) 117-128.

[Sc2] M.-P. Schützenberger, Promotion des morphisms d'ensembles ordonnés, Discrete Math. **2** (1972) 73-94.

[Sh] M. Shimozono, Littlewood-Richardson rules for ordinary and projective representations of symmetric groups, Ph.D. Thesis, University of California, San Diego (1991).

[T] G.P. Thomas, On a construction of Schützenberger, Discrete Math. **17** (1977) 107-118.

[W] D.E. White, Hybrid tableaux and the Littlewood-Richardson rule, Discrete Math. **80** (1990) 183-206.