Chapter 2
Primer on k-Schur Functions

JENNIFER MORSE,[1] ANNE SCHILLING[2] AND MIKE ZABROCKI[3]
morsej@math.drexel.edu, anne@math.ucdavis.edu, zabrocki@mathstat.yorku.ca

based on lectures by Luc Lapointe and Jennifer Morse
lapointe@inst-mat.utalca.cl and morsej@math.drexel.edu

The purpose of this chapter is to outline some of the results and open problems related to k-Schur functions, mostly in the setting of symmetric function theory. This chapter roughly follows the outline of several talks given by Luc Lapointe and Jennifer Morse at a conference titled "Affine Schubert Calculus" held in July of 2010 at the Fields Institute in Toronto.[4]

In addition it presents many examples based on code written in SAGE [140, 151] by Jason Bandlow, Nicolas M. Thiéry, the last two authors, and many other SAGE developers. The following presentation is intended to give both an idea of the origins of the k-Schur functions as well as the current ideas and computational tools which have been most productive for demonstrating their properties.

We will present almost no proofs in this chapter, but rather refer to the original articles for detailed arguments. Instead the concepts are illustrated with many SAGE examples to highlight how to discover and experiment with many of the still open conjectures related to k-Schur functions. The purpose behind most of the SAGE examples is to demonstrate the formulas with examples and to give the commands that would allow a first time user of SAGE to be able to use the functions to generate data that they might need for their own research.

---

[1]The author was supported by NSF gant DMS–0652641, DMS–0652668, DMS–1001898.

[2]The author was supported by NSF grants DMS–0652641, DMS–0652652, DMS–1001256, and OCI–1147247.

[3]The author was supported by NSERC.

[4]see http://www.fields.utoronto.ca/programs/scientific/10-11/schubert/

T. Lam et al., *k-Schur Functions and Affine Schubert Calculus*, Fields Institute Monographs 33, DOI 10.1007/978-1-4939-0682-6_2,

Section 1 reviews much of the combinatorial background of k-Schur theory including partitions, cores, (partial) orders on the affine symmetric group, and some symmetric function theory. This section also sets up the combinatorial backdrop needed to give the Pieri rules for k-Schur functions and their duals. In Sect. 2, we define a parameterless ($t = 1$) family of k-Schur functions using an analogue of the Pieri rule for Schur functions [96]. This definition is used to relate k-Schur functions to the geometry and to Stanley symmetric functions discussed in Chap. 3. We also give the dual Pieri rule [81] which gives rise to a monomial expansion of the k-Schur functions. The Pieri and dual Pieri rule motivate the definition of weak and strong order tableaux.

In Sect. 3, we present four conjecturally equivalent definitions of the k-Schur functions for generic t. Some are known to be equivalent when $t = 1$. The first definition of k-Schur functions appeared in a paper by Lapointe, Lascoux and Morse [91] and is purely combinatorial in nature; defined as a sum over certain classes of tableaux called atoms. Lapointe and Morse [94] followed this paper by defining symmetric functions which were defined by algebraic operations instead of a sum over combinatorial objects. The last two definitions of the k-Schur functions with a generic parameter t are defined along lines similar to the parameterless k-Schur functions, but now a t-statistic is introduced on weak (resp. strong) order tableaux.

In Sect. 4 we present many of the properties of k-Schur functions and outline what is known about which property for each of the definitions. This is followed by Sect. 5 which contains further research directions and many conjectures that remain to be resolved (and hence the content is likely to change in the future)! Section 6 explains the duality between strong and weak order in terms of a k-analogue of the Robinson–Schensted–Knuth algorithm, which gives rise to an affine insertion algorithm. We present part of this algorithm by giving a bijection between permutations and pairs of tableaux. Finally in Sect. 7 some details about the branching from k to $(k + 1)$-Schur functions are given.

1 Background and Notation

1.1 Partitions and Cores

A *partition* $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_{\ell(\lambda)})$ of m is a sequence of weakly decreasing positive integers which sum to $m = \lambda_1 + \lambda_2 + \cdots + \lambda_{\ell(\lambda)}$. The value of m is called the *size* of the partition and this will be denoted by $|\lambda|$. The entries of the partition are called the parts and the number of parts of the partition is denoted by $\ell(\lambda)$. As a general convention, if $i > \ell(\lambda)$ then $\lambda_i = 0$ and the definition of symmetric functions (which turn out to be indexed by partitions) given later in this section respects this convention. The statistic $n(\lambda) = \sum_{i=1}^{\ell(\lambda)}(i-1)\lambda_i$ on partitions has a value between 0 and $m(m-1)/2$ for partitions of m and this will arise in the definitions of symmetric functions. A partition λ is called *k-bounded* if $\lambda_1 \leq k$. The notation $\lambda \vdash m$ indicates that λ is a partition of m and generally we reserve the symbols λ, μ, ν to denote partitions.

A partition will be identified with its *Young (or Ferrers) diagram*. This is a diagram consisting of square cells arranged in left justified rows stacked on top of each other with the largest row with λ_1 cells on the bottom. (This convention is also called the French notation; when stacking the rows with the largest row at the top is called the English convention). Alternatively, a Young diagram is a collection of cells in the first quadrant of the (x, y)-plane with $dg(\lambda) = \{(i, j) : 1 \leq i \leq \ell(\lambda) \text{ and } 1 \leq j \leq \lambda_i\}$ represented as boxes in the Cartesian plane so that the upper right hand corner of a cell has coordinate which is in this collection. For consistency with other references we have chosen that the first coordinate represents the row and the second coordinate represents the column (each beginning at 1 for the first row and column). For an example the Young diagram for the partition $\lambda = (4, 3, 3, 3, 2, 2, 1)$ is drawn in Example 1.1.

There is a partial order on partitions that arises naturally in symmetric functions when ordering basis elements. For two partitions λ, μ such that $|\lambda| = |\mu|$, we say that $\lambda \leq \mu$ if $\sum_{i=1}^{r} \lambda_i \leq \sum_{i=1}^{r} \mu_i$ for all $r \geq 1$. This is usually referred to as the *dominance order* on partitions.

The *conjugate* of a partition λ is the sequence $\lambda' = (\lambda'_1, \lambda'_2, \ldots, \lambda'_{\lambda_1})$ where $\lambda'_r = \#\{i : \lambda_i \geq r\}$. Alternatively, this can be seen on Young diagrams by reflecting the diagram in the $x = y$ line of the coordinate plane so that $dg(\lambda') = \{(j, i) : (i, j) \in dg(\lambda)\}$. For example in Example 1.1 below, $\lambda' = (7, 6, 4, 1)$ for the partition $\lambda = (4, 3, 3, 3, 2, 2, 1)$.

For many uses we will need to refer to the number of parts of a partition of a given size i and this will be denoted by $m_i(\lambda) = \#\{j : \lambda_j = i\}$. The quantity

$$z_\lambda = \prod_{i \geq 1} m_i(\lambda)! \; i^{m_i(\lambda)} \tag{1.1}$$

is the size of the stabilizer of a permutation $\sigma \in S_m$, the symmetric group on $m = |\lambda|$ letters, whose cycle type is λ under the conjugation action of S_m. That is, if σ has cycle type λ, then $z_\lambda = \#\{\tau \in S_m : \tau\sigma\tau^{-1} = \sigma\}$. Since we know that all permutations with the same cycle type are conjugate, the number of permutations with cycle type λ is equal to $m!/z_\lambda$.

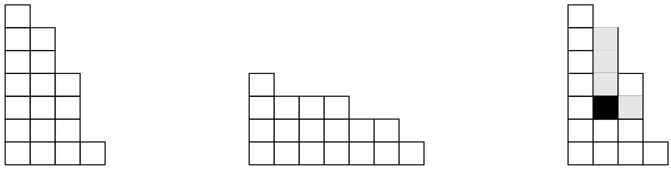
Each cell in a partition λ has a *hook length* which consists of the number of cells in the column above and in the row to the right (including the cell itself). Namely, for a cell $(i, j) \in dg(\lambda)$, the hook length of the cell is $\text{hook}_\lambda(i, j) = \lambda_i + \lambda'_j - i - j + 1$. In Example 1.1 below $\text{hook}_{(4,3,3,3,2,2,1)}(3, 2) = 5 = \lambda_3 + \lambda'_2 - 3 - 2 + 1$.

For a partition λ with $\lambda_1 \leq k$, define the *k-split* of λ as a sequence of partitions (which will be denoted by $\lambda^{\to k}$) recursively. If $\lambda_1 + \ell(\lambda) - 1 \leq k$, then $\lambda^{\to k} = (\lambda)$. Otherwise,

$$\lambda^{\to k} = ((\lambda_1, \lambda_2, \ldots, \lambda_{k-\lambda_1+1}), (\lambda_{k-\lambda_1+2}, \lambda_{k-\lambda_1+3}, \ldots, \lambda_{\ell(\lambda)})^{\to k}) \,. \tag{1.2}$$

In other words, the k-split of a partition is found by successively splitting off parts of the partition with hook k, starting with the first part, until that is no longer possible.

Example 1.1. The Young diagram for the partition $\lambda = (4, 3, 3, 3, 2, 2, 1)$ is the diagram on the left and its conjugate partition $\lambda' = (7, 6, 4, 1)$ is the diagram in the center.



The diagram on the right is the Young diagram for the partition $\lambda=(4, 3, 3, 3, 2, 2, 1)$ with the cells that are in the hook of the cell $(3, 2)$ shaded in. In this case $\text{hook}_\lambda(3, 2) = 5$. The 4-split of λ is $\lambda^{\to 4} = ((4), (3, 3), (3, 2), (2, 1))$ and the 5-split is $\lambda^{\to 5} = ((4, 3), (3, 3, 2), (2, 1))$.

We will use the realization of the Young diagram as the set of cells in our notation and define $\lambda \subseteq \mu$ if $\text{dg}(\lambda) \subseteq \text{dg}(\mu)$. This forms a lattice, also known as the *Young lattice*, on set of partitions and the cover relation is given by $\lambda \to \mu$ if $\lambda \subseteq \mu$ and $|\lambda| + 1 = |\mu|$. The lattice is graded by the size of the partition and the first six levels of the infinite Hasse diagram are shown in Fig. 2.1.

There are several special types of containments of partitions that will arise in this discussion. If $\lambda \subseteq \mu$, then μ/λ is called a *skew partition* and it will represent the cells which are in $\text{dg}(\mu)/\text{dg}(\lambda)$, with the / here representing the difference of sets.

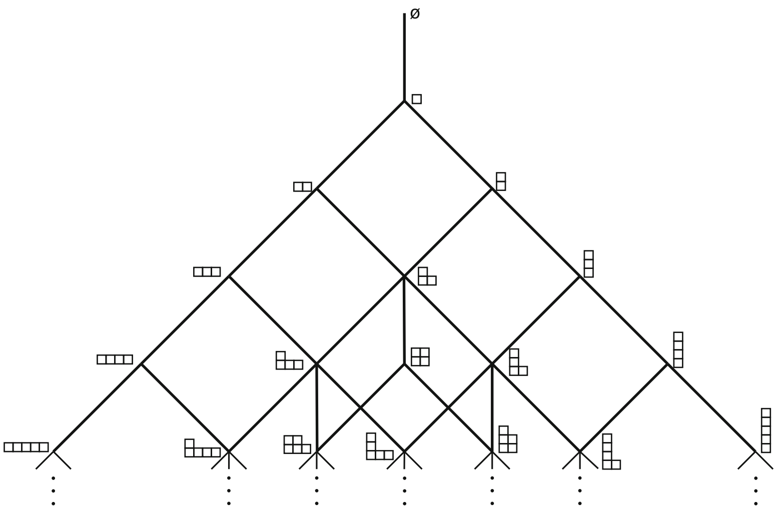


Fig. 2.1 The Young lattice of partitions (up to those of size 5) ordered by inclusion

We call μ/λ *connected* if for any two cells there is a sequence of cells in μ/λ from one to the other where consecutive cells share an edge. We say that μ/λ is a *horizontal* (*vertical*) strip if there is at most one cell in each column (row) of μ/λ. The skew partition μ/λ is called a *ribbon* if it does not contain any 2×2 subset of cells.

Sage Example 1.2. We now demonstrate how to access partitions and their properties in the open source computer algebra system SAGE (see section "Appendix: SAGE" in Chap. 1). We begin by listing all partitions of 4:

```
sage: P = Partitions(4); P
Partitions of the integer 4
sage: P.list()
[[4], [3, 1], [2, 2], [2, 1, 1], [1, 1, 1, 1]]
```

SAGE has list comprehension so that the last line could have also been written as

```
sage: [p for p in P]
[[4], [3, 1], [2, 2], [2, 1, 1], [1, 1, 1, 1]]
```

We can check how two partitions λ and μ relate in the dominance order

```
sage: la=Partition([2,2]); mu=Partition([3,1])
sage: mu.dominates(la)
True
```

and draw the entire Hasse diagram

```
sage: ord = lambda x,y: y.dominates(x)
sage: P = Poset([Partitions(6), ord], facade=True)
sage: H = P.hasse_diagram()
sage: view(H)        #optional
```

which outputs the graph. The `view(H)` command may not work properly unless dot2tex and Graphviz are installed on your version of Sage. Here we used the python syntax for a function, which is `lambda x : f(x)` for a function that maps x to $f(x)$. We can also compute the conjugate of a partition, its k-split

```
sage: la=Partition([4,3,3,3,2,2,1])
sage: la.conjugate()
[7, 6, 4, 1]
sage: la.k_split(4)
[[4], [3, 3], [3, 2], [2, 1]]
```

and create skew partitions

```
sage: p = SkewPartition([[2,1],[1]])
sage: p.is_connected()
False
```

1.2 Bounded Partitions, Cores, and Affine Grassmannian Elements

We will see that k-Schur functions are symmetric functions indexed by k-bounded partitions and consequently, the underlying combinatorial framework we need often comes out of a refinement of classical ideas in the theory of partitions. As it happens, the set of k-bounded partitions is in bijection with several different sets of natural combinatorial objects and often the k-Schur function setting is better expressed in those terms. To this end, we begin with a discussion of several other examples of possible indexing sets.

As with the k-bounded partitions, we are interested in another special subset of partitions. In particular, an *r-core* is a shape where none of its cells have a hook-length equal to r. We denote the set of all r-cores by $\mathcal{C}_r$. When we consider a partition as a core, the notion of size differs from the usual notion (where size counts the number of cells in the shape). In contrast, the relevant notion of size on a $(k+1)$-core is to count only the number of cells which have a hook-length smaller than $k+1$. We call this the *length* of the core. For a $(k+1)$-core κ, its length will be denoted by $|\kappa|_{k+1}$ or simply $|\kappa|$ if it is clear from the context that κ is viewed as a $(k+1)$-core. As $k \to \infty$, this becomes the usual size of the partition. Later in this section, we will see that the length is related to the length of elements in the affine symmetric group. Now, we give the connection between cores and bounded partitions.

Proposition 1.3 ([96, Theorem 7]). *There is a bijection between the set of $(k+1)$-cores κ with $|\kappa|_{k+1} = m$ and partitions $\lambda \vdash m$ with $\lambda_1 \leq k$.*

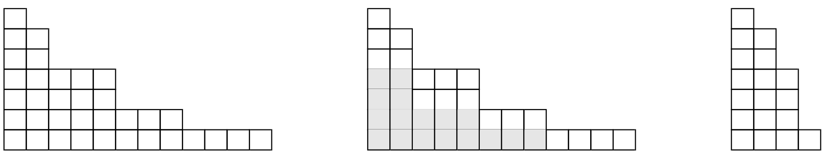
The bijection from $(k+1)$-cores to k-bounded partitions is

$$\mathfrak{p} : \kappa \mapsto \lambda \,,$$

defined by setting

$$\lambda_i = \#\{(i,j) \in \kappa : \mathrm{hook}_\kappa(i,j) \leq k\} \,. \tag{1.3}$$

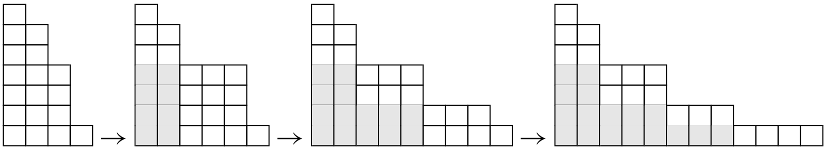
Example 1.4. The partition $(12, 8, 5, 5, 2, 2, 1)$ on the left is a 5-core since there are no cells in its Ferrers diagram with hook-length equal to 5. Equation (1.3) tells us how to applying $\mathfrak{p}$ to this core to obtain a 4-bounded partition; delete each cell in the diagram for the 5-core whose hook-length exceeds 5 and then slide all remaining cells to the left.



The first part of the resulting partition is at most 4.

The other direction of the bijection is also not difficult. Consider a k-bounded partition and work from the smallest part of the partition to the largest and slide the cells to the right until it is a $(k+1)$-core. Here is a description of the procedure which can be followed with Example 1.5. Start with the top row $\lambda_{\ell(\lambda)}$ of the k-bounded partition λ and successively move down a row. For a given row, calculate the hook lengths of its cells; if there is a cell with hook length greater than k, slide this row to the right until all cells have hook length less than or equal to k. Continue this process until all rows have been adjusted. The end result will be a $(k+1)$-core which we shall denote by $\mathfrak{c}_k(\lambda)$ or just $\mathfrak{c}(\lambda)$ if k is clear from the context.

Example 1.5. The partition $(4, 3, 3, 3, 2, 2, 1)$ is a 4-bounded partition. Here we draw the successive slides of the rows until we reach a 5-core:



Sage Example 1.6. Here is the way to compute the map $\mathfrak{c}$ in SAGE:

```
sage: la = Partition([4,3,3,3,2,2,1])
sage: kappa = la.k_skew(4); kappa
[12, 8, 5, 5, 2, 2, 1] / [8, 5, 2, 2]
```

For the inverse $\mathfrak{p}$ we write

```
sage: kappa.row_lengths()
[4, 3, 3, 3, 2, 2, 1]
```

If one is only handed the 5-core $(12, 8, 5, 5, 2, 2, 1)$ instead of the skew partition, one can do the following:

```
sage: tau = Core([12,8,5,5,2,2,1],5)
sage: mu = tau.to_bounded_partition(); mu
[4, 3, 3, 3, 2, 2, 1]
sage: mu.to_core(4)
[12, 8, 5, 5, 2, 2, 1]
```

All 3-cores of length 6 can be listed as:

```
sage: Cores(3,6).list()
[[6, 4, 2], [5, 3, 1, 1], [4, 2, 2, 1, 1], [3, 3, 2, 2, 1, 1]]
```

We now turn our attention to the third set of objects that is in bijection with the set of k-bounded partitions (and the set of $(k+1)$-cores). These come out of studying the type A affine Weyl group and its realization as the *affine symmetric group* $\tilde{S}_n$ given by generators $\{s_0, s_1, \ldots, s_{n-1}\}$ satisfying the relations

$$
\begin{aligned}
&s_i^2 = 1,\\
&s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1},\\
&s_i s_j = s_j s_i \quad \text{for } i - j \not\equiv 0, 1, n-1 \pmod{n}
\end{aligned}
\tag{1.4}
$$

with all indices related $\pmod{n}$. Hereafter, we shall reserve parameters n and k and we will set $n = k + 1$ throughout.

There is a subset of the elements in $\tilde{S}_n$ that is particularly conducive to combinatorics in large part because it is in bijection with the set of k-bounded partitions and of $(k+1)$-cores. Note that the symmetric group S_n generated by $\{s_1, s_2, \ldots, s_{n-1}\}$ is a subgroup, where the element s_i represents the permutation which interchanges i and $i+1$. We will refer to the left cosets of $\tilde{S}_n/S_n$ as *affine Grassmannian elements* and they will be identified with their minimal length coset representatives, that is, the elements of $w \in \tilde{S}_n$ such that either $w = id$ or s_0 is the only elementary transposition such that $\ell(ws_0) < \ell(w)$.

Remark 1.7. The definition of affine Grassmannian elements are the special case of a more general definition. The l-Grassmannian elements are the minimal length coset representatives of $\tilde{S}_n/S_n^l$ where S_n^l is the group generated by $\{s_0, s_1, s_2, \ldots, s_{n-1}\} \backslash \{s_l\}$ and the affine Grassmannian elements are the 0-Grassmannian elements. Due to the cyclic symmetry of the affine type A Dynkin diagram, these constructions are of course all equivalent.

Sage Example 1.8. We can create the affine symmetric group and its generators in SAGE as

```
sage: W = WeylGroup(["A",4,1])
sage: S = W.simple_reflections()
sage: [s.reduced_word() for s in S]
[[0], [1], [2], [3], [4]]
```

For a given element, we can ask for its reduced word or create it from a word in the generators and ask whether it is Grassmannian:

```
sage: w = W.an_element(); w
[ 2  0  0  1 -2]
[ 2  0  0  0 -1]
[ 1  1  0  0 -1]
[ 1  0  1  0 -1]
[ 1  0  0  1 -1]
sage: w.reduced_word()
[0, 1, 2, 3, 4]
sage: w = W.from_reduced_word([2,1,0])
sage: w.is_affine_grassmannian()
True
```

Proposition 1.9 ([103] [96, Proposition 40]). *There is a bijection between the collection of cosets $\tilde{S}_{k+1}/S_{k+1}$ whose minimal length representative has length m and $(k+1)$-cores of length m.*

The bijection of Proposition 1.9 is defined by an action of the affine symmetric group on cores. It suffices to define the left action of the generators s_i of the affine symmetric group on $(k+1)$-cores. The diagonal index or *content* of a cell $c = (i, j)$ in the diagram for a core is $j - i$. We will often instead be concerned with the *residue* of c, denoted by $\mathrm{res}(c)$, which is the diagonal index mod $k + 1$. We call a cell c an addable corner of a partition μ if $\mathrm{dg}(\mu) \cup \{c\}$ is the diagram for a partition and a cell c is a removable corner of μ if $\mathrm{dg}(\mu) \backslash \{c\}$ is diagram for a partition.

Definition 1.10 ([103] [96, Definition 18]). For κ a $(k + 1)$-core, let $s_i \cdot \kappa$ be the partition with

1. If there is at least one addable corner of residue i, then the result is κ with all addable corners of κ of residue i added,
2. If there is at least one removable corner of residue i, then the result is κ with all removable corners of κ of residue i removed,
3. Otherwise, the result is κ.

Example 1.11. Consider the 5-core, $(7, 3, 1)$. If we draw its Ferrers diagrams and label each of the cells with the content modulo 5 we have the following diagram

$$\begin{array}{|c|c|c|c|c|c|c|}\cline{1-1} 3 \\ \cline{1-3} 4 & 0 & 1 \\ \hline 0 & 1 & 2 & 3 & 4 & 0 & 1 \\ \hline\end{array}\;.$$

This diagram has addable corners with residue 2 and 4 and removable corners with residue 1 and 3 and all cells of residue 0 are neither addable nor removable. Therefore, $s_2 \cdot (7, 3, 1) = (8, 4, 1, 1)$, $s_4 \cdot (7, 3, 1) = (7, 3, 2)$, $s_1 \cdot (7, 3, 1) = (6, 2, 1)$, $s_3 \cdot (7, 3, 1) = (7, 3)$, $s_0 \cdot (7, 3, 1) = (7, 3, 1)$.

Sage Example 1.12. In SAGE we can get the affine symmetric group action on cores as follows:

```
sage: c = Core([7,3,1],5)
sage: c.affine_symmetric_group_simple_action(2)
[8, 4, 1, 1]
sage: c.affine_symmetric_group_simple_action(0)
[7, 3, 1]
```

We can also check directly that the set of affine Grassmannian elements of given length are in bijection with the corresponding cores:

```
sage: k=4; length=3
sage: W = WeylGroup(["A",k,1])
sage: G=W.affine_grassmannian_elements_of_given_length(length)
sage: [w.reduced_word() for w in G]
[[2, 1, 0], [4, 1, 0], [3, 4, 0]]
```

```
sage: C = Cores(k+1,length)
sage: [c.to_grassmannian().reduced_word() for c in C]
[[2, 1, 0], [4, 1, 0], [3, 4, 0]]
```

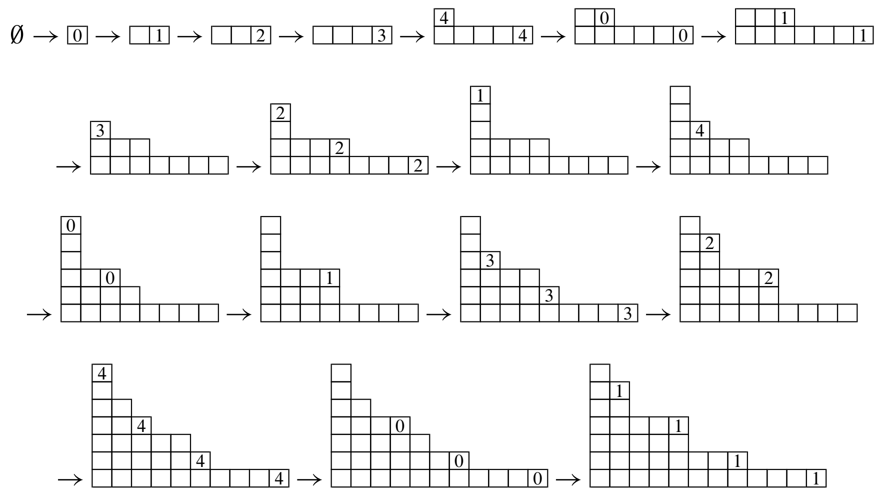
The bijection of Proposition 1.9 is realized by taking a reduced word for an affine Grassmannian element and acting on the empty $(k+1)$-core. The resulting $(k+1)$-core is the image of the bijection. Lapointe and Morse [96, Corollary 48] then states that the reverse bijection can be found by taking $\lambda = \mathfrak{p}(\kappa)$ and forming the element in the affine symmetric group $s_{\mathrm{res}(c_1)} s_{\mathrm{res}(c_2)} \cdots s_{\mathrm{res}(c_m)}$, where $c_1, c_2, \ldots, c_m$ are the cells of $\mathrm{dg}(\lambda)$ read from the smallest row to the largest with each read from right to left.

We denote the map which sends $(k+1)$-core κ to the corresponding affine Grassmannian element by $\mathfrak{a}(\kappa) = w_\kappa$. Since $(k+1)$-cores and k-bounded partitions are in bijection, we will also use the notation $\mathfrak{a}(\lambda) = w_\lambda$ to represent the map from a k-bounded partition λ to an affine Grassmannian element.

Example 1.13. Consider the reduced word,

$$w = s_1 s_0 s_4 s_2 s_3 s_1 s_0 s_4 s_1 s_2 s_3 s_1 s_0 s_4 s_3 s_2 s_1 s_0 \ .$$

We apply this word on the left on an empty 5-core to build up the result. The sequence of applications builds the core as follows:



and hence the resulting 5-core is $\mathfrak{a}^{-1}(w) = (12, 8, 5, 5, 2, 2, 1)$.

The reverse bijection comes from reading the residues of the corresponding 4-bounded partition $(4, 3, 3, 3, 2, 2, 1)$ from the smallest row to the largest row, and from right to left within the rows. For example:

$$\begin{array}{|c|c|c|c|}
\cline{1-1} 4 \\
\cline{1-2} 0 & 1 \\
\cline{1-2} 1 & 2 \\
\cline{1-3} 2 & 3 & 4 \\
\cline{1-3} 3 & 4 & 0 \\
\cline{1-3} 4 & 0 & 1 \\
\hline 0 & 1 & 2 & 3 \\
\hline
\end{array}$$

is sent under $\mathfrak{a}$ to the word

$$w' = s_4 s_1 s_0 s_2 s_1 s_4 s_3 s_2 s_0 s_4 s_3 s_1 s_0 s_4 s_3 s_2 s_1 s_0 \,.$$

It is not difficult to show that w is equivalent to w'.

Sage Example 1.14. We can verify the previous example in SAGE.

```
sage: la = Partition([4,3,3,3,2,2,1])
sage: c = la.to_core(4); c
[12, 8, 5, 5, 2, 2, 1]
sage: W = WeylGroup(["A",4,1])
sage: w = W.from_reduced_word([4,1,0,2,1,4,3,2,0,4,3,1,0,
....:     4,3,2,1,0])
sage: c.to_grassmannian() == w
True
```

The affine symmetric group $\tilde{S}_n$ can also be thought of as the group of permutations of $\mathbb{Z}$ with the property that for $w \in \tilde{S}_n$ we have $w(i + rn) = w(i) + rn$ for all $r \in \mathbb{Z}$ with the additional property that $\sum_{i=1}^{n} w(i) - i = 0$. We can choose the convention that the elements $s_i \in \tilde{S}_n$ for $0 \le i \le n-1$ act on $\mathbb{Z}$ by $s_i(i+rn) = i+1+rn$, $s_i(i+1+rn) = i+rn$, and $s_i(j) = j$ for $j \not\equiv i, i+1 \pmod{n}$. While the elements s_i generate the group, there is also the notion of a general transposition t_{ij} which generalizes this notion by interchanging i and $j \pmod{n}$. Take integers $i < j$ with $i \not\equiv j \pmod{n}$ and $v = \lfloor (j-i)/n \rfloor$, then $t_{i,i+1} = s_i$ and for $j - i > 1$,

$$t_{ij} = s_i s_{i+1} s_{i+2} \cdots s_{j-v-2} s_{j-v-1} s_{j-v-2} s_{j-v-3} \cdots s_{i+1} s_i$$

where all of the indices of the s_m are taken $\pmod{n}$. For $j > i$, we set $t_{ij} = t_{ji}$. The t_{ij} generalize the elements s_i by their action $t_{ij}(i+rn) = j+rn$, $t_{ij}(j+rn) = i + rn$ and $t_{ij}(\ell) = \ell$ for $\ell \not\equiv i, j \pmod{n}$. It is not hard to show that

$$t_{ij} w = w t_{w^{-1}(i) w^{-1}(j)},$$

which allows us to define a left as well as a right action on affine permutations. Here we state the results in terms of the left action.

The elements $w \in \tilde{S}_n$ are determined by the action of w on the values 1 through n since this determines the action on all of $\mathbb{Z}$ by $w(i + rn) = w(i) + rn$. If w is represented in two line notation,

$$\begin{array}{ccccccc} \cdots & -2 & -1 & 0 & 1 & 2 & \cdots \\ \cdots & w(-2) & w(-1) & w(0) & w(1) & w(2) & \cdots \end{array}$$

then $t_{ij}w$ is obtained from w by exchanging $i + rn$ and $j + rn$ in the lower row of the two line notation for w. We also have that wt_{ij} is obtained from w by exchanging $w(i + rn)$ and $w(j + rn)$. An element is affine Grassmannian if $w(1) < w(2) < \cdots < w(n)$. The tuple of values $[w(1), w(2), \ldots, w(n)]$ is referred to as the *window notation* for w.

There is a close relationship between the action of w^{-1} on integers and on cores. This relationship is best demonstrated by an example before we give the precise statement.

Example 1.15. Start with $n = k + 1 = 3$ and the affine Grassmannian element $w = s_1s_0s_2s_1s_0$ that has window notation $[w(1), w(2), w(3)] = [-2, 0, 8]$. We can use this to determine that w^{-1} is given in two line notation as

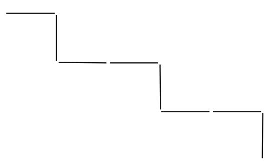
$$\begin{array}{rcccccccccccccccccc} & \cdots & -7 & -6 & -5 & -4 & -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & \cdots \\ w^{-1} = & \cdots & -12 & -4 & -2 & -9 & -1 & 1 & -6 & 2 & 4 & -3 & 5 & 7 & 0 & 8 & 10 & \cdots \end{array}$$

Now we notice in this notation that $w^{-1}(d) \leq 0$ for all integers $d \leq -3$ (and for any affine permutation there will always be an integer d' such that $w^{-1}(d) \leq 0$ for all $d \leq d'$). We also see that for all integers $d \geq 6$ we have $w^{-1}(d) > 0$ (and for any affine permutation there will always be a value D' such that $w^{-1}(d) > 0$ for all $d \geq D'$).

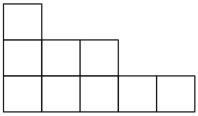
Now consider the integers $-2 \leq d \leq 5$ (which are the integers strictly between these two values $d' = -3$ and $D' = 6$). Reading from left to right in the two line notation, we construct a path consisting of East and South steps where for each d such that $w^{-1}(d) \leq 0$ we place a South step, and for each d such that $w^{-1}(d) > 0$ we place an East step. In this example we are looking at the sequence

$$\begin{array}{cccccccc} -2 & -1 & 0 & 1 & 2 & 3 & 4 & 5 \\ 1 & -6 & 2 & 4 & -3 & 5 & 7 & 0 \end{array}$$

in order to create this path. The way we have chosen our d' and D' the first step of this path will always be East and the last step will always be South. In this example we have the path:



and it is the outline for the diagram of a 3-core:



It is the case that $\mathfrak{c}(2,2,1) = (5,3,1)$ and corresponds to the affine permutations $w = s_1 s_0 s_2 s_1 s_0 = \mathfrak{a}(2,2,1)$ by the other bijections.

Sage Example 1.16. We can demonstrate the previous example in SAGE. Using the class AFFINEPERMUTATIONGROUP allows to input an affine permutation in window notation:

```
sage: A = AffinePermutationGroup(["A",2,1])
sage: w = A([-2,0,8])
sage: w.reduced_word()
[1, 0, 2, 1, 0]
sage: w.to_core()
[5, 3, 1]
```

The action of t_{ij} on the two line notation for w can be translated into the action for t_{ij} on the two line notation for w^{-1}. Since we have $(t_{ij}w)^{-1} = w^{-1}t_{ij}$, then the same action of left multiplication by t_{ij} on w has the effect of exchanging $w^{-1}(i + rn)$ and $w^{-1}(j + rn)$ in the two line notation for w^{-1}. Similarly, right multiplication by t_{ij} on w has the effect of exchanging the values of $i + rn$ and $j + rn$ in the two line notation for w^{-1}.

As in our example above, the two line notation for w^{-1} keeps track of the outline of the $(k+1)$-core representing the affine permutation. The two line notation for w^{-1} represents an infinite path where the 0 and negative values represent South steps and the positive values represent East steps. There is some point d' for which all steps before are South and another point D' where all steps after are East. Between d' and D' there is a path which traces the outline of a $(k + 1)$-core. Notice that when $w = w^{-1}$ is the identity, then $d' = 0$ and $D' = 1$ and the path between these two points represents the empty core.

We can work out precisely what the action of the transpositions s_i and t_{ij} are on this path and we will see that left multiplication by s_i on w has the same effect on this path as the action that s_i has on the $(k + 1)$-core given in Definition 1.10. If left multiplication on w by an s_i increases the length by 1, then on the sequence of values of $w^{-1}(i)$ this has the effect of interchanging a negative value which lies to the left of a positive value. On the path consisting of South steps for negative values and East steps for positive values, interchanging a South step that comes just before an East step has the effect of adding a cell on the path representing the core.

Although we have limited ourselves to affine permutations, k-bounded partitions and $k + 1$-cores, we note that there are other useful ways to describe the set such as with abaci or bit sequences that we do not discuss.

1.3 Weak Order and Horizontal Chains

Because our indexing set comes from a quotient of $\tilde{S}_{k+1}$, and every Coxeter system is naturally equipped with the weak and the strong (Bruhat) orders, the close study of the weak and strong order posets on $\tilde{S}_{k+1}$ is called for. Here we examine posets that are isomorphic to the weak subposet on affine Grassmannian elements and whose vertices are given by the set of k-bounded partitions and by the set of $(k+1)$-cores.

The (left) weak order is defined by saying that w is less than or equal to v in $\tilde{S}_{k+1}$ if and only if there is some $u \in \tilde{S}_{k+1}$ such that $uw = v$ and $\ell(u) + \ell(w) = \ell(v)$. We denote a cover in (left) weak order by $\to_k$, that is, $w \to_k v$ if and only if there is some $s_i \in \tilde{S}_{k+1}$ such that $s_i w = v$ and $\ell(w) + 1 = \ell(v)$. The affine Weyl group of type A forms a lattice under inclusion in the weak order (this is known due to results of Waugh [155]).

Now given the bijection between $\tilde{S}_{k+1}/S_{k+1}$ and k-bounded partitions or $(k+1)$-cores, it is natural to question how weak order $\to_k$ is characterized on these other sets. On the set of $(k+1)$-cores, the weak order relation can be framed in terms of the action of the elements s_i. This result can be found in [103, 112] and is restated in a similar form in [81, Lemma 8.6].

Proposition 1.17. *If κ and τ are $(k+1)$-cores with $|\kappa|_{k+1} = |\tau|_{k+1} + 1$, then $\tau \to_k \kappa$ if and only if there exists an i such that $\kappa = s_i \cdot \tau$.*

Remark 1.18. It follows that $\to_k$ can also be characterized on $(k+1)$-cores by $\tau \to_k \kappa$ if and only if all cells in κ/τ have the same $k+1$-residue.

The characterization of the weak order poset on $\tilde{S}_n/S_n$ on the level of k-bounded partitions is inspired by viewing the Young covering relation as $\lambda \to \mu$ if $\lambda \subseteq \mu$ and $\lambda' \subseteq \mu'$ and $|\lambda| + 1 = |\mu|$. Of course, $\lambda' \subseteq \mu'$ if and only if $\lambda \subseteq \mu$. However, working on the subset of k-bounded partitions, there is a generalization of conjugation under which this is not a superfluous condition.

Definition 1.19. Let λ be a k-bounded partition. Then the *k-conjugate* of λ is defined as

$$\lambda^{\omega_k} := \mathfrak{p}(\mathfrak{c}(\lambda)').$$

Sage Example 1.20. We can obtain the 4-conjugate of the partition $(4,3,3,3,2,2,1)$ from the last partition in Example 1.5 by reading off the column lengths of the unshaded boxes in each column. In SAGE:

```
sage: la = Partition([4,3,3,3,2,2,1])
sage: la.k_conjugate(4)
[3, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1]
```

Proposition 1.21 ([96, Corollary 25]). *For k-bounded partitions λ and μ, $\lambda \to_k \mu$ if and only if $\lambda \subseteq \mu$, $\lambda^{\omega_k} \subseteq \mu^{\omega_k}$, and $|\lambda| + 1 = |\mu|$.*

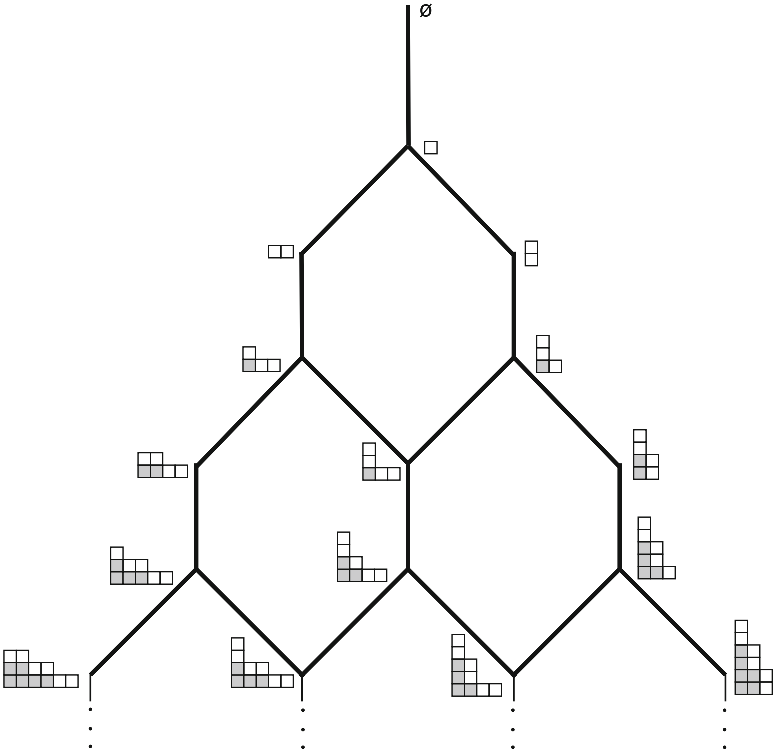


Fig. 2.2 The lattice of 3-cores (up to those of length 6), which correspond to the 2-bounded partitions, ordered by the weak order

We now turn our attention to a distinguished set of saturated chains. Recall that standard tableaux can be viewed as saturated chains in the Young lattice. This notion can be generalized to semi-standard tableaux. A semi-standard tableau is an increasing sequence of partitions in the Young lattice such that two adjacent partitions in this sequence differ by a horizontal strip. Horizontal strips are skew shapes with at most one cell in any column. As we will see in Sect. 2.1 horizontal strips are fundamental in the formulation of the Pieri rule for Schur functions. The analogue of horizontal strips in the affine setting was introduced in [96]. As we will see in Sect. 2.2 these will play a central role in the combinatorics of k-Schur functions.

Crudely, we define a *weak horizontal strip* of size $r \le k$ to be a horizontal strip κ/τ of $(k+1)$-cores κ and τ such that there exists a saturated chain

$$\tau \to_k \tau^{(1)} \to_k \tau^{(2)} \to_k \cdots \to_k \tau^{(r)} = \kappa\,. \tag{1.5}$$

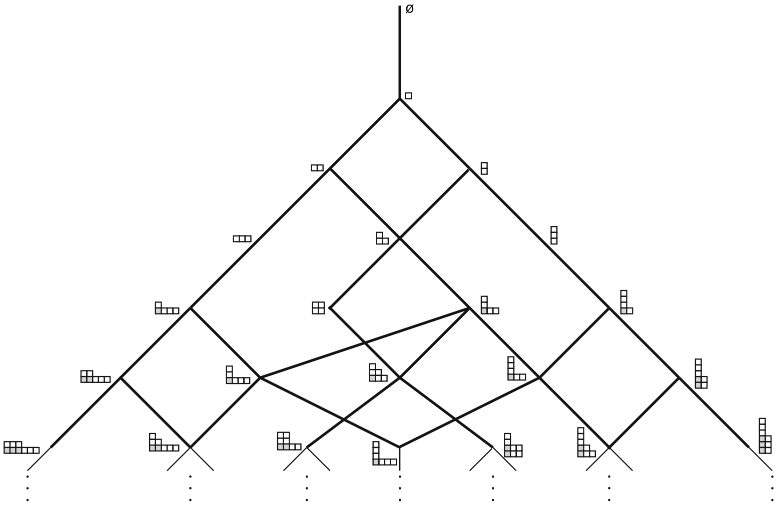


Fig. 2.3 The lattice of 4-cores (up to those of length 6), which correspond to the 3-bounded partitions, ordered by the weak order

It is helpful to instead think of these strips as the skew κ/τ of $(k+1)$-cores κ and τ, where

$$\kappa/\tau \text{ is a horizontal strip} \tag{1.6}$$

$$|\kappa|_{k+1} = |\tau|_{k+1} + r \tag{1.7}$$

$$\text{there are exactly } r \text{ residues in the set of cells of } \kappa/\tau. \tag{1.8}$$

We discussed how the weak order is naturally realized on k-bounded partitions and affine Grassmannian elements and it is also worthwhile to rephrase the notion of strips in these different contexts.

In the k-bounded partition framework (for example, useful in [7, 91, 92, 94, 95]), the notion of weak horizontal strip is defined to be a horizontal strip μ/λ where $\mu^{\omega_k}/\lambda^{\omega_k}$ is a vertical strip. This characterization is motivated by the following result about weak horizontal strips.

Proposition 1.22 ([96, Sect. 9]). *Let $\tau \subseteq \kappa$ be $(k+1)$-cores. Then κ/τ forms a weak horizontal strip if and only if $\mathfrak{p}(\kappa)/\mathfrak{p}(\tau)$ is a horizontal strip and $\mathfrak{p}(\kappa')/\mathfrak{p}(\tau')$ is a vertical strip.*

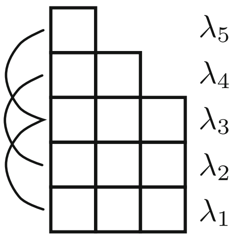
It is important to note that it is not sufficient to characterize κ/τ being a weak horizontal strip by assuming that $\mathfrak{p}(\kappa)/\mathfrak{p}(\tau)$ is a horizontal strip. A good example of this can be observed in Fig. 2.3. Consider the 4-cores $\kappa = (4, 1)$ and $\tau = (2, 1)$. We note that $\mathfrak{p}(\kappa) = (3, 1)$ and $\mathfrak{p}(\tau) = (2, 1)$ and so even though we see that $\mathfrak{p}(\kappa)/\mathfrak{p}(\tau)$ is a horizontal strip, there does not exist a path from the 4-core $(2, 1)$ to $(4, 1)$ in the lattice. If we consider the conjugate partitions, we see that $\mathfrak{p}(\kappa') = (1, 1, 1, 1)$ and $\mathfrak{p}(\tau') = (2, 1)$ and so $\mathfrak{p}(\kappa')/\mathfrak{p}(\tau')$ is not a vertical strip.

Before we discuss the notion of weak horizontal strip in the framework of affine permutations, we give an alternative description of the k-conjugate and the map $\mathfrak{c}$ from k-bounded partitions to $(k+1)$-cores that was communicated to us by Karola Mészáros (October 2011, private communication). Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_\ell)$ be a k-bounded partition. Start from the longest part of λ, namely λ_1, and successively connect a row of length i to the $(k+1-i)$th row above it (in other words, skip $k-i$ rows). Call this a string. Repeat this with the next longest part that is not yet part of a string until all parts of λ are part of a string. Denote by $\{\ell_1^{(j)}, \ell_2^{(j)}, \ldots\}$ the parts of λ in the jth string where $1 \leq j \leq r$. Then

$$(\lambda^{\omega_k})' = (\ell_1^{(1)} + \ell_2^{(1)} + \cdots, \ell_1^{(2)} + \ell_2^{(2)} + \cdots, \ldots, \ell_1^{(r)} + \ell_2^{(r)} + \cdots). \tag{1.9}$$

Similarly, the i^{th} part of $\mathfrak{c}(\lambda)$ is obtained by adding all elements in the string containing λ_i that are weakly above λ_i.

Example 1.23. Let $\lambda = (3, 3, 3, 2, 1)$ with $k = 4$. Then the strings are as follows:



and

$$(\lambda^{\omega_4})' = (\lambda_1 + \lambda_3 + \lambda_5, \lambda_2 + \lambda_4) = (7, 5),$$
$$\mathfrak{c}(\lambda) = (\lambda_1 + \lambda_3 + \lambda_5, \lambda_2 + \lambda_4, \lambda_3 + \lambda_5, \lambda_4, \lambda_5) = (7, 5, 4, 2, 1).$$

Lastly, we interpret horizontal chains in the framework of affine permutations. In particular, weak horizontal strips are the *cyclically decreasing* elements of $\tilde{S}_{k+1}/S_{k+1}$. These are affine Grassmannian elements w where $w = s_{i_1} \cdots s_{i_\ell}$ for a sequence $i_1 \cdots i_\ell$ such that no number is repeated and j precedes $j-1$ (taken modulo $k+1$) when both $j, j-1 \in \{i_1, \ldots, i_\ell\}$. Then, based on the following proposition, a weak horizontal strip can be thought of as a pair of affine Grassmannian elements v and w where $uw = v$ and $\ell(u) + \ell(w) = \ell(v)$ for some cyclically decreasing u.

Proposition 1.24. *Let $\tau \subseteq \kappa$ be $(k+1)$-cores. Then κ/τ forms a weak horizontal strip if and only if $\kappa = s_{i_1} \cdots s_{i_\ell} \tau$ for some cyclically decreasing element $w = s_{i_1} \cdots s_{i_\ell}$.*

1.4 Cores and the Strong Order of the Affine Symmetric Group

We are now ready to discuss the strong (Bruhat) order, starting from the core viewpoint [103, 122]. A *strong cover* is defined on $k+1$ cores by

$$\tau \Rightarrow_k \kappa \iff |\mathfrak{p}(\tau)| + 1 = |\mathfrak{p}(\kappa)| \quad \text{and} \quad \tau \subseteq \kappa\,.$$

When a pair of cores satisfies $\tau \Rightarrow_i \kappa$, their skew diagram has is made up of ribbons. To be precise, the *head* of a connected ribbon is the southeast most cell of the ribbon and the *tail* is the northwest most corner. Then (see [81, Proposition 9.5]):

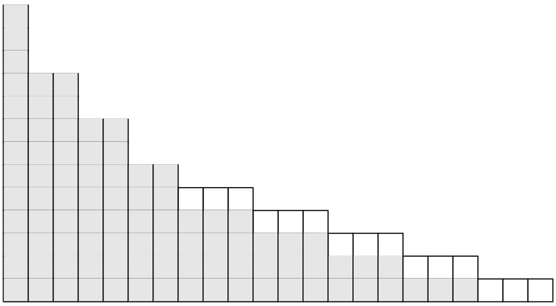
- Each connected component of κ/τ is a ribbon and they are all identical translates each other;
- The residues of the heads of the connected components must all be the same and must lie on consecutive positions of those residues (the term 'consecutive' here means that if two heads are separated by a multiple of $k+1$ cells by a taxicab distance then there is one which is exactly $k+1$ distance that is in-between).

A *strong marked cover* is a strong cover along with a value c which indicates the content of the head of one of the copies of the ribbons. More precisely, we define a *marking* as a triple (κ, τ, c) where κ, τ are $(k+1)$-cores such that $\tau \Rightarrow_k \kappa$ and c is a number which is $j-i$ for the cell (the diagonal index of the cell) at position (i, j) of the south-east most cell of the connected component of κ/τ which is marked.

Example 1.25. Consider the 4-cores

$$\tau = (19, 16, 13, 10, 7, 7, 5, 5, 5, 3, 3, 1, 1, 1) \Rightarrow_3 (22, 19, 16, 13, 10, 7, 5, 5, 3, 3, 1, 1, 1) = \kappa$$

which correspond to the skew diagram

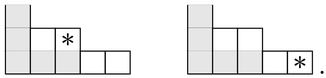


This is a relatively large example, where κ/τ contains five different copies of three cells in a row. The marking, c, can be any one of the five values representing the content of the rightmost cell in the connected component, $c \in \{21, 17, 13, 9, 5\}$.

Example 1.26. The diagram for the poset of the strong order at $k = 2$ is given in Fig. 2.4 and it can be read off the diagram that the strong covers of

$(3, 1, 1)$ with respect to this order are $(5, 3, 1)$, $(4, 2, 1, 1)$ and $(3, 2, 2, 1, 1)$ because $(3, 1, 1)$ is contained in each of these 3-cores. Note that it is not true that there is containment of the corresponding 2-bounded partitions since $\mathfrak{p}(3, 1, 1) = (2, 1, 1)$ and $\mathfrak{p}(3, 2, 2, 1, 1) = (1, 1, 1, 1, 1)$.

In particular, if $\tau = (3, 1, 1)$ and $\kappa = (5, 3, 1)$, then $(3, 1, 1) \Rightarrow_2 (5, 3, 1)$ because $(5, 3, 1)/(3, 1, 1)$ consists of two copies of a connected horizontal strip. This means that $((5, 3, 1)/(3, 1, 1), 1)$ and $((5, 3, 1)/(3, 1, 1), 4)$ are strong marked covers since the cells $(2, 3)$ and $(1, 5)$ are the coordinates of the heads of the horizontal strips and they have content 1 and 4 respectively. These are represented by the skew tableaux



Although we started with a discussion of strong order in the setting of $k + 1$-cores, it comes from ordering elements u, w of the Coxeter system $\tilde{S}_{k+1}$ by

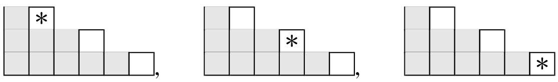
$$w \Rightarrow_k u \iff t_{ij} w = u \quad \text{and} \quad \ell(w) + 1 = \ell(u)\,.$$

The notion of marked covers can be interpreted in this framework as well.

Proposition 1.27 ([81, Sect. 2.3]). *Let τ, κ be two $(k + 1)$-cores such that $\tau \Rightarrow_k \kappa$ and assume there is a marking of κ/τ at diagonal $j - 1$ and i is the diagonal index of the tail of the marked ribbon. Let w be the affine Grassmannian permutation corresponding to τ and u the affine Grassmannian permutation corresponding to κ. Then we have*

- $w^{-1}(i) \leq 0 < w^{-1}(j)$.
- $t_{ij} w = u$.
- *The number of connected ribbons which are below the marked one is $(-w^{-1}(i) - a)/n$ where $a = -w^{-1}(i) \pmod n$ (the representative between 0 and n).*
- *The number of connected ribbons which are above the marked one is $(w^{-1}(j) - b)/n$ where $b = w^{-1}(j) \pmod n$ and the total number of connected components in κ/τ is $1 + (-w^{-1}(i) + w^{-1}(j) - a - b)/n$.*
- *The number of cells in the ribbon is $j - i$.*
- *The height of the ribbon is the number of d such that $i \leq d < j$ such that $w^{-1}(d) \leq 0$.*

Example 1.28. Consider the 3-core $\tau = (5, 3, 1)$ which corresponds to the reduced word $w = s_1 s_0 s_2 s_1 s_0$ and which is also given by its action on the integers by $[w(1), w(2), w(3)] = [-2, 0, 8]$. Then $t_{-10} = t_{23} = t_{56} = s_2$ and $\kappa = t_{-10} \cdot (5, 3, 1) = (6, 4, 2)$.



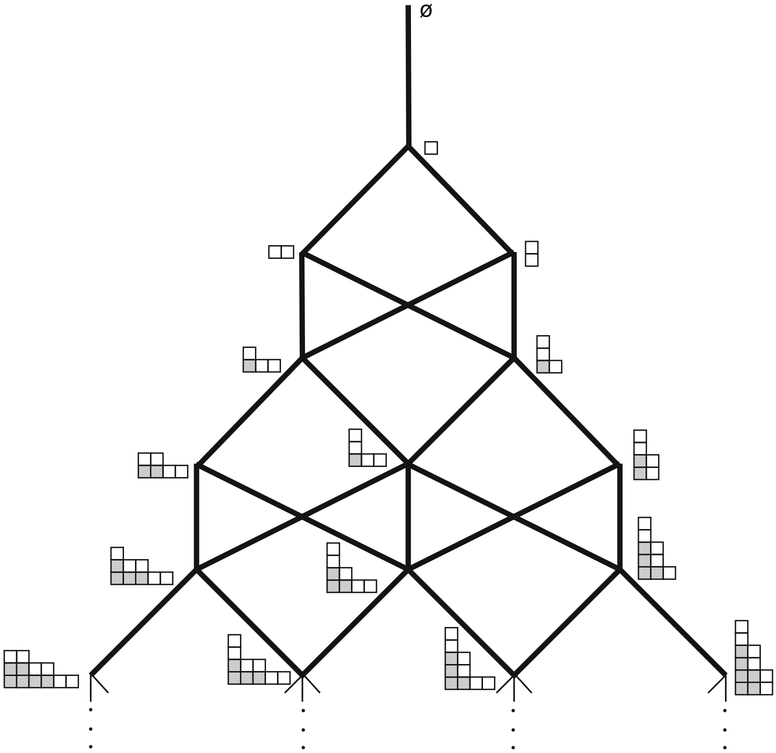


Fig. 2.4 The poset of 3-cores (up to those of length 6), ordered by the strong order (no markings)

The three markings are on diagonal -1 and 2 and 5. Moreover since $w^{-1}(-1) = -6 \le 0 < w^{-1}(0) = 2$, $w^{-1}(2) = -3 \le 0 < w^{-1}(3) = 5$ and $w^{-1}(5) = 0 \le 0 < w^{-1}(6) = 8$, these three transpositions satisfy the conditions of the proposition for each of the three marked strong covers. The right action is expressed as the element $wt_{-62} = wt_{-35} = wt_{08}$.

The number of connected components is equal to $1 + (-w^{-1}(i) + w^{-1}(j) - a - b)/n$ which for $j = 0$ and $i = -1$ amounts to $1 + (6 + 2 - 0 - 2)/3 = 3$.

Example 1.29. The 4-core τ in Example 1.25 corresponds to the reduced word

$$w = s_1 s_2 s_0 s_3 s_1 s_2 s_0 s_3 s_1 s_2 s_1 s_0 s_3 s_1 s_2 s_1 s_0 s_3 s_0 s_1 s_2 s_1 s_0 s_3 s_2 s_1 s_0$$

and $t_{19,22}w = t_{15,18}w = t_{11,14}w = t_{7,10}w = t_{3,6}w$ represent the five strong covers with the left action and $wt_{0,19} = wt_{-4,15} = wt_{-8,11} = wt_{-12,7} = wt_{-16,3}$ represent the five strong covers with the right action.

Remark 1.30. Recall that for two elements $w, w' \in S_n$ with $\ell(w') = \ell(w) + 1$, w' is a cover of w in the (left) weak order if

$$s_i w = w'$$

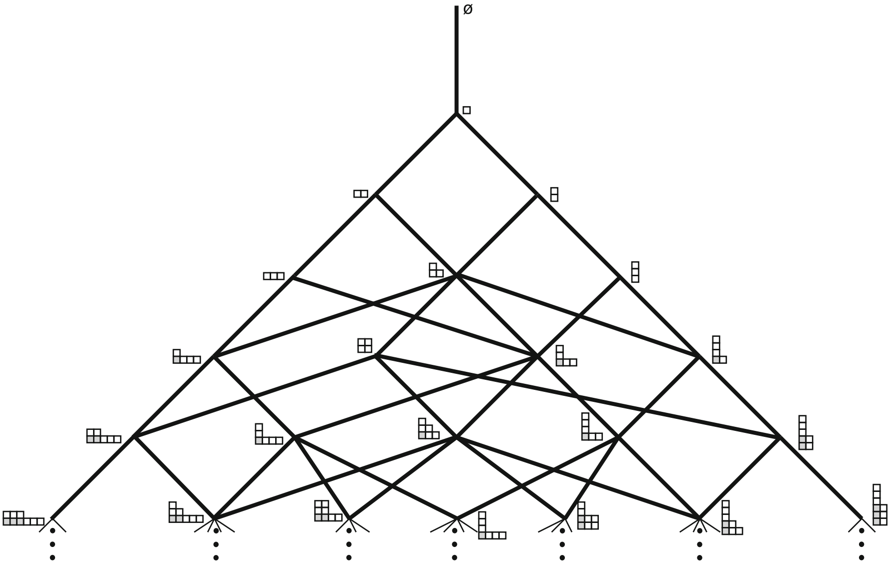


Fig. 2.5 The poset of 4-cores (up to those of length 6), ordered by the strong order (no markings)

for some simple transposition s_i and w' is a cover of w in the strong (or Bruhat) order if

$$t_{ij} w = w'$$

for some transposition t_{ij}. By analogy, since $\Rightarrow_k$ is a left multiplication by an affine transposition (Proposition 1.27) and $\rightarrow_k$ is left multiplication by a simple affine transposition on cores (from Definition 1.10), the cover relations are called strong and weak covers, respectively.

We have included the Hasse diagrams for the weak and strong orders for the poset of 3 and 4-cores up to those of length 6 in Figs. 2.2 through 2.5. Note that the strong order on cores does not form a lattice.

Sage Example 1.31. We can produce the weak and strong covers of a given core

```
sage: c = Core([3,1,1],3)
sage: c.weak_covers()
[[4, 2, 1, 1]]
sage: c.strong_covers()
[[5, 3, 1], [4, 2, 1, 1], [3, 2, 2, 1, 1]]
```

as well as compare two $(k+1)$-cores with respect to weak and strong order

```
sage: kappa = Core([4,1],4)
sage: tau = Core([2,1],4)
```

```
sage: tau.weak_le(kappa)
False
sage: tau.strong_le(kappa)
True
```

Figure 2.3 can be reproduced in SAGE via:

```
sage: C = sum(([c for c in Cores(4,m)] for m in range(7)),[])
sage: ord = lambda x,y: x.weak_le(y)
sage: P = Poset([C, ord], cover_relations = False)
sage: H = P.hasse_diagram()
sage: view(H)        #optional
```

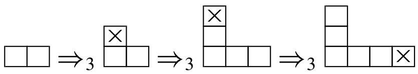
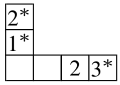
The `view(H)` command is optional because dot2tex and Graphviz need to be installed for it to display properly. If they are not installed, a diagram can be displayed by inserting the line `sage: Partitions.global_options (latex="list")`.

As with the weak order poset, we are also concerned with certain chains in the strong order poset. A *strong marked horizontal strip* of size r is a succession of strong marked covers,

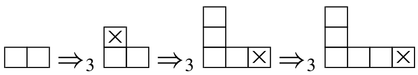
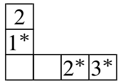
$$\kappa^{(0)} \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(r)}, \tag{1.10}$$

where the markings c_i associated to $\kappa^{(i-1)} \Rightarrow_k \kappa^{(i)}$ satisfy $c_1 < c_2 < \cdots < c_r$.

Example 1.32. Consider the following sequence of 4-cores with the markings indicated on the diagram.


, or more compactly denoted

.

This is not a strong marked horizontal strip because the markings $c_1 = -1, c_2 = -2, c_3 = 3$ are not increasing. However, the same set of 4-cores with markings


, or more compactly denoted

.

as shown have $c_1 = -1 < c_2 = 2 < c_3 = 3$ and hence this is a strong marked horizontal strip.

Remark 1.33. It is worth pointing out that when k is large, a succession of strong marked covers,

$$\kappa^{(0)} \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(r)}, \tag{1.11}$$

reduces to a saturated chain in Young's lattice where the condition that $c_1 < c_2 < \cdots < c_r$ implies that $\kappa^{(r)}/\kappa^{(0)}$ is a horizontal strip.

1.5 Symmetric Functions

The ring of symmetric functions shall be defined as

$$\Lambda = \mathbb{Q}[h_1, h_2, h_3, \ldots], \tag{1.12}$$

the ring of polynomials in the generators h_r. Here we are considering the ring Λ without reference to 'variables' for which there is a symmetric group action but we will now make explicit the connection with symmetric polynomials and symmetric series.

Let σ be a permutation that acts on the variables $\{x_1, x_2, x_3, \ldots, x_m\}$ by $\sigma(x_i) = x_{\sigma_i}$ and this action extends to polynomials. We call a polynomial S_m-invariant (or symmetric) if $\sigma f(x_1, x_2, \ldots, x_m) = f(x_1, x_2, \ldots, x_m)$ for all $\sigma \in S_m$. The ring Λ is identified with functions which are symmetric series in an infinite set of variables by setting

$$h_r[X] = \sum_{1 \le i_1 \le i_2 \le \cdots \le i_r} x_{i_1} x_{i_2} \cdots x_{i_r} \tag{1.13}$$

and symmetric polynomials are then just a specialization of these series with a finite number of variables, $h_r[X_m] = \sum_{1 \le i_1 \le i_2 \le \cdots \le i_r \le m} x_{i_1} x_{i_2} \cdots x_{i_r}$.

For each partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_{\ell(\lambda)})$, we set

$$h_\lambda[X] := h_{\lambda_1}[X] h_{\lambda_2}[X] \cdots h_{\lambda_{\ell(\lambda)}}[X] \,. \tag{1.14}$$

The set of these symmetric series forms a linear basis for an algebra isomorphic to Λ. We will consider various bases for Λ. One such basis is the *monomial basis*

$$m_\lambda[X] = \sum_{\mathrm{sort}(\alpha) = \lambda} x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_{\ell(\alpha)}^{\alpha_{\ell(\alpha)}} \,, \tag{1.15}$$

where the sum is over all sequences α such that if the parts are arranged in weakly decreasing order the resulting sequence is the partition λ. It is not hard to see from the definitions that the generators of Λ are related to the monomial symmetric functions by

$$h_r[X] = \sum_{\lambda \vdash r} m_\lambda[X] \,. \tag{1.16}$$

The h_r are known as the *complete homogeneous generators* and the their products will be referred to as the complete homogeneous or simply homogeneous basis.

The *power sum* generators are the elements

$$p_r[X] = m_{(r)}[X] = \sum_{i \ge 1} x_i^r \tag{1.17}$$

and the *elementary generators* are defined as

$$e_r[X] = m_{(1^r)}[X] = \sum_{1 \le i_1 < i_2 < \cdots < i_r} x_{i_1} x_{i_2} \cdots x_{i_r} \ . \tag{1.18}$$

The monomials in these sets of generators, $p_\lambda[X] := p_{\lambda_1}[X] p_{\lambda_2}[X] \cdots p_{\lambda_{\ell(\lambda)}}[X]$ and $e_\lambda[X] := e_{\lambda_1}[X] e_{\lambda_2}[X] \cdots e_{\lambda_{\ell(\lambda)}}[X]$ also form bases for the space Λ indexed by partitions.

The variable X in these symmetric functions is, for the moment, superfluous notation and to make certain formulas more compact we will drop $[X]$ when it is implicit that it is there. However, we will later consider transformations on the ring of symmetric functions by adding notation to $[X]$ and there will be times that the $[X]$ will be used to indicate that the expression it is attached to is a symmetric function.

One place where it will be necessary to keep the reference to the variables explicit is in the use of a few 'plethystic' expressions involving parameters q and t. We may extend the notation defined above to include all rational expressions in variables $q, t, x_1, x_2, x_3, \ldots$, $E = E(x_1, x_2, \ldots; q, t)$, then

$$p_r[E] = E(x_1^r, x_2^r, \ldots; q^r, t^r) \ . \tag{1.19}$$

Note that by using the expression E as the infinite sum $X = x_1 + x_2 + x_3 + \cdots$, the notation in Eq. (1.19) is consistent with the expression in Eq. (1.17).

In particular we will frequently use the notation $f\left[\frac{X}{1-t}\right]$, $f[X(1-t)]$ and $f\left[X\frac{1-q}{1-t}\right]$ to represent the symmetric function $f[X]$ with $p_r[X]$ replaced with $p_r[X]/(1-t^r)$, $p_r[X](1-t^r)$ and $p_r[X]\frac{1-q^r}{1-t^r}$ respectively. This transformation is sometimes also denoted by

$$\theta_{qt} f[X] = f\left[X\frac{1-q}{1-t}\right] . \tag{1.20}$$

When we need to use a finite number of variables, the expression $X_m = x_1 + x_2 + x_3 + \cdots + x_m$ is used to indicate that $p_r[X_m] = \sum_{i=1}^m x_i^r$. Normally we consider symmetric functions in an arbitrary alphabet which can be specialized appropriately and we assume that there is an implicit $[X]$ following all symmetric function expressions where no variables are specified.

The three types of generators are related by

$$\sum_{i=0}^{r} (-1)^{r-i} h_i e_{r-i} = 0 \qquad\qquad r h_r = \sum_{i=1}^{r} h_{r-i} p_i \tag{1.21}$$

$$\sum_{i=0}^{r} (-1)^{r-i} i h_i e_{r-i} = p_r \qquad\qquad r e_r = \sum_{i=1}^{r} (-1)^{i-1} e_{r-i} p_i \ . \tag{1.22}$$

These relations are sufficient to express any one of the generators $\{e_r, h_r, p_r\}$ in terms of another.

There is a scalar product on the ring of symmetric functions for which the monomial and homogeneous symmetric functions are orthonormal. We have also,

$$\langle h_\lambda, m_\mu \rangle = \langle p_\lambda, p_\mu / z_\mu \rangle = \delta_{\lambda\mu} := \begin{cases} 1 & \text{if } \lambda = \mu, \\ 0 & \text{otherwise,} \end{cases} \tag{1.23}$$

where z_μ is defined in Eq. (1.1).

One powerful use of this scalar product is that it allows us to compute a single coefficient in the expansion of a symmetric function in terms of these bases. If $f \in \Lambda$, then $\langle f, h_\mu \rangle$ is the coefficient of m_μ in f. That is, if $f = \sum_\gamma c_\gamma m_\gamma$, then

$$\langle f, h_\mu \rangle = \sum_\gamma c_\gamma \langle m_\gamma, h_\mu \rangle = c_\mu \,. \tag{1.24}$$

Similarly $\langle f, m_\mu \rangle$ is the coefficient of h_μ in f and $\langle f, p_\mu / z_\mu \rangle$ is equal to the coefficient of p_μ in f.

Sage Example 1.34. We now show how to create various bases in SAGE and how to obtain the coefficients of a given symmetric function using the computer. We begin by defining the homogeneous and monomial bases:

```
sage: Sym = SymmetricFunctions(QQ)
sage: h = Sym.homogeneous()
sage: m = Sym.monomial()
```

Then we define a symmetric function f and expand it in terms of the monomial basis:

```
sage: f = h[3,1]+h[2,2]
sage: m(f)
10*m[1, 1, 1, 1] + 7*m[2, 1, 1] + 5*m[2, 2]
 + 4*m[3, 1] + 2*m[4]
```

There are several ways to obtain the coefficients of a given term. Both of the following yield the coefficient of m_{211} in f:

```
sage: f.scalar(h[2,1,1])
7
sage: m(f).coefficient([2,1,1])
7
```

The order in which bases are multiplied and added determines which is the output basis. For instance to demonstrate (1.22) we consider the following two equivalent expressions:

```
sage: p = Sym.power()
sage: e = Sym.elementary()
sage: sum((-1)**(i-1)*e[4-i]*p[i] for i in range(1,4)) - p[4]
4*e[4]
```

```
sage: sum((-1)**(i-1)*p[i]*e[4-i] for i in range(1,4)) - p[4]
1/6*p[1, 1, 1, 1] - p[2, 1, 1] + 1/2*p[2, 2]
 + 4/3*p[3, 1] - p[4]
```

1.6 Schur Functions

A combinatorial definition of the Schur functions is given in Sect. 2 which we shall generalize to the k-Schur functions in Sect. 2.2 and the dual k-Schur functions in Sect. 2.4. For now we start with two (equivalent) algebraic definitions.

Definition 1.35. The Schur functions s_λ are the unique basis of Λ for which

1. $\langle s_\lambda, s_\mu \rangle = \delta_{\lambda\mu}$ for any partitions λ, μ;
2. $s_\lambda = m_\lambda +$ terms of the form $r_{\lambda\mu} m_\mu$ for partitions μ of $|\lambda|$ with $\mu < \lambda$ in dominance order.

We have chosen this as the definition of the Schur functions because it naturally generalizes to the Hall–Littlewood and Macdonald symmetric functions (which we shall introduce in the next few pages). There are many formulas known for the Schur functions which can either be taken as a defining relation or as a consequence. In the next section we shall shift perspectives and consider the Schur functions as the family of symmetric functions which satisfy the Pieri rule (see Eq. (2.2)).

Example 1.36. In the following we abbreviate $s_{(2,1)}$ by s_{21} if there is no confusion about the parts. By definition we have that

$$s_{111} = m_{111}$$

and we may proceed by calculating Gram-Schmidt orthonormalization. For instance, since $s_{111} = m_{111} = e_3 = h_{111} - 2h_{21} + h_3$ we have

$$s_{21} = m_{21} - \langle m_{21}, s_{111} \rangle\, s_{111} = m_{21} + 2m_{111}\ .$$

If then s_{21} is expanded in the homogeneous basis, we see that it is $s_{21} = h_{21} - h_3$. Finally, to calculate s_3 we note that

$$s_3 = m_3 - \langle m_3, s_{21} \rangle\, s_{21} - \langle m_3, s_{111} \rangle\, s_{111} = m_3 + m_{21} + m_{111}\ .$$

Sage Example 1.37. If we wanted to check Example 1.36 using SAGE, we could define h and m as in Sage Example 1.34 and then run

```
sage: Sym = SymmetricFunctions(QQ)
sage: s = Sym.schur()
sage: m = Sym.monomial()
```

```
sage: h = Sym.homogeneous()
sage: m(s[1,1,1])
m[1, 1, 1]
sage: h(s[1,1,1])
h[1, 1, 1] - 2*h[2, 1] + h[3]
```

etc. We can also obtain the expansion of the Schur functions into power sum symmetric functions:

```
sage: p = Sym.power()
sage: s = Sym.schur()
sage: p(s[1,1,1])
1/6*p[1, 1, 1] - 1/2*p[2, 1] + 1/3*p[3]
sage: p(s[2,1])
1/3*p[1, 1, 1] - 1/3*p[3]
sage: p(s[3])
1/6*p[1, 1, 1] + 1/2*p[2, 1] + 1/3*p[3]
```

and the following calculation shows that the Schur functions are orthogonal:

```
sage: s[2,1].scalar(s[1,1,1])
0
sage: s[2,1].scalar(s[2,1])
1
```

Since the Schur functions form a basis of the ring of symmetric functions Λ, a product of two Schur functions can again be expanded in terms of Schur functions:

$$s_\lambda s_\mu = \sum_\nu c_{\lambda\mu}^\nu s_\nu . \tag{1.25}$$

It turns out that the coefficients $c_{\lambda\mu}^\nu$, called *Littlewood–Richardson coefficients*, are nonnegative integer coefficients. The famous *Littlewood–Richardson* rule [118] provides a combinatorial expression for the coefficients $c_{\lambda\mu}^\nu$. It says that $c_{\lambda\mu}^\nu$ is equal to the number of semi-standard tableaux of skew shape ν/λ and weight μ whose column reading word is Yamanouchi. Here λ, μ, ν are partitions and a semi-standard tableau of shape ν/λ is a filling of the skew shape which is weakly increasing across rows and strictly increasing up columns. The weight of a tableau or word is $\mu = (\mu_1, \mu_2, \ldots)$, where μ_i counts the number of i in the tableau or word. Furthermore, a word is Yamanouchi if all right subwords have partition weight.

Remarkably, Schur functions and their Littlewood–Richardson coefficients tie into the study of the geometry of the Grassmannian $\mathrm{Gr}_{\ell n}$ (the manifold of ℓ-dimensional subspaces of $\mathbb{C}^n$). The cohomology ring of $\mathrm{Gr}_{\ell n}$ has a basis of Schubert classes σ_λ, indexed by shapes $\lambda \in \mathcal{P}^{\ell n}$ contained in an $\ell \times (n-\ell)$ rectangle. The intersection numbers are encoded by the structure constants of $H^*(\mathrm{Gr}_{\ell n})$ in the Schubert basis:

$$\sigma_\lambda \cup \sigma_\mu = \sum_{\nu \in \mathcal{P}^{\ell n}} c_{\lambda\mu}^\nu \sigma_\nu . \tag{1.26}$$

The explicit understanding of $H^*(\mathrm{Gr}_{\ell n})$ and of these intersections is gained by Schur functions. Letting $I = \langle e_{n-\ell+1}, \ldots, e_n\rangle$ and $\Lambda_{(\ell)} = \mathbb{Q}[h_1, h_2, h_3, \ldots, h_\ell]$, there is an isomorphism,

$$H^*(\mathrm{Gr}_{\ell n}) \cong \Lambda_{(\ell)}/I \ , \tag{1.27}$$

under which σ_λ corresponds to s_λ. Importantly, $s_\nu \in I$ when $\nu \notin \mathcal{P}^{\ell n}$, and thus the structure constants of $H^*(\mathrm{Gr}_{\ell n})$ are none other than the Littlewood–Richardson coefficients (1.25) for Schur function products.

The ring of symmetric functions is also endowed with a *Hopf algebra structure*. A systematic study of Λ from the Hopf algebra perspective is given in [159], see also [120]. In particular, the coproduct on the Schur basis is given in terms of the Littlewood–Richardson coefficients

$$\Delta(s_\nu) = \sum_{\lambda,\mu} c^{\nu}_{\lambda\mu} s_\lambda \otimes s_\mu. \tag{1.28}$$

There is also an algebraic involution on Λ for which $\omega(h_\lambda) = e_\lambda$ and $\omega(p_\lambda) = (-1)^{|\lambda|-\ell(\lambda)} p_\lambda$ and $\omega(s_\lambda) = s_{\lambda'}$.

With the scalar product it is natural to introduce the operation which is dual to multiplication. That is, for a homogeneous symmetric function f of degree k, multiplication by f is an operation which will raise the degree of a symmetric function by k, and the notation $f^\perp$ will represent an operator that will lower a symmetric function by degree k and its action is defined as

$$f^\perp(g) = \sum_\lambda \langle g, f s_\lambda\rangle\, s_\lambda = \sum_\lambda \langle g, f h_\lambda\rangle\, m_\lambda \ . \tag{1.29}$$

The reason why we introduce these operators is that we find that we can define 'creation operators' for the Schur functions which allow us to show that the Schur functions satisfy the Pieri rule. Set $\mathbf{S}_m = \sum_{r\geq 0}(-1)^r h_{m+r} e_r^\perp$. The sum is apparently infinite but we need only calculate up to r equal to the degree of the symmetric function it is acting on. These operators have the property that for a partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_\ell)$ and $m \geq \lambda_1$,

$$\mathbf{S}_m s_\lambda = s_{(m,\lambda_1,\lambda_2,\ldots,\lambda_\ell)} \ . \tag{1.30}$$

One reason these operators are particularly useful is that commutation rules such as $p_r\mathbf{S}_m = \mathbf{S}_m p_r + \mathbf{S}_{m+r}$ and $e_r\mathbf{S}_m = \mathbf{S}_{m+1}e_{r-1} + \mathbf{S}_m e_r$ and $h_r\mathbf{S}_m = \sum_{i=0}^r \mathbf{S}_{m+i} h_{r-i}$ can be used to show the Murnaghan–Nakayama rule and the Pieri rules (respectively). We will also use the operators $\mathbf{S}_m$ to express creation operators for the Hall–Littlewood symmetric functions and a means for computing them. In particular,

$$s_\lambda = \mathbf{S}_{\lambda_1}\mathbf{S}_{\lambda_2}\cdots\mathbf{S}_{\lambda_{\ell(\lambda)}}(1) \ .$$

1.7 Hall–Littlewood Symmetric Functions

The Hall–Littlewood symmetric functions have a definition which is similar to that of the Schur functions. These functions form a basis of the ring of symmetric functions over a field containing a parameter t. We work in the fraction field over the polynomials in the parameter t, and set

$$\Lambda_t = \mathbb{Q}(t)[h_1, h_2, h_3, \ldots] . \tag{1.31}$$

The functions $Q'_\lambda[X;t]$ are defined as the family of symmetric functions satisfying

$$Q'_\lambda[X;t] = s_\lambda + \text{ terms of the form } r_{\lambda\mu}(t)s_\mu \text{ for } \mu > \lambda$$

and

$$\left\langle Q'_\lambda[X;t], Q'_\mu[X;t]\right\rangle_t = 0 \text{ if } \lambda \neq \mu$$

where the scalar product $\langle \,\cdot\, , \,\cdot\, \rangle_t$ is defined so that

$$\left\langle p_\lambda, p_\mu\right\rangle_t = z_\lambda \delta_{\lambda\mu} \prod_i (1 - t^{\lambda_i}) . \tag{1.32}$$

By this definition of the t-scalar product, we see that $\left\langle p_\lambda, p_\mu\right\rangle_t = \langle p_\lambda[X], p_\mu[X(1-t)]\rangle$, where on the right hand side we have used the usual scalar product from Eq. (1.23).

We will mainly be using the scalar product from Eq. (1.23), so define $\{P_\lambda[X;t]\}_{\lambda\vdash n}$ to be the dual basis to the basis $\{Q'_\lambda[X;t]\}_{\lambda\vdash n}$. Since we know that

$$\left\langle Q'_\lambda[X;t], Q'_\mu[X(1-t);t]\right\rangle = 0 \quad \text{if } \lambda \neq \mu,$$

we must have $P_\lambda[X;t] = c_\lambda Q'_\lambda[X(1-t);t]$ for some coefficients c_λ. Explicit formulas for the coefficient c_λ are known, but since

$$1 = \left\langle Q'_\lambda[X;t], P_\lambda[X;t]\right\rangle = \left\langle Q'_\lambda[X;t], c_\lambda Q'_\lambda[X(1-t);t]\right\rangle ,$$

it follows that $c_\lambda = \left\langle Q'_\lambda[X;t], Q'_\lambda[X(1-t);t]\right\rangle^{-1}$.

Example 1.38. We compute the Hall–Littlewood symmetric functions for partitions of 3 using this method to demonstrate how they might be implemented in a computer program.

The triangularity relation shows that $Q'_3 = s_3$. Then Q'_{21} is defined as

$$Q'_{21} = s_{21} - \frac{\langle s_{21}, Q'_3\rangle_t}{\langle Q'_3, Q'_3\rangle_t} Q'_3 \,.$$

By calculating from the expansion of the Schur functions in the power sums as in Example 1.37, we have that $\langle s_{21}, s_3\rangle_t = t^2 - t$ and $\langle s_3, s_3\rangle_t = 1 - t$. We conclude that $Q'_{21} = s_{21} + ts_3$.

Similarly we can compute the last Hall–Littlewood symmetric function of size 3 by the computation

$$Q'_{111} = s_{111} - \frac{\langle s_{111}, Q'_{21}\rangle_t}{\langle Q'_{21}, Q'_{21}\rangle_t} Q'_{21} - \frac{\langle s_{111}, Q'_3\rangle_t}{\langle Q'_3, Q'_3\rangle_t} Q'_3 \,.$$

Using the previous examples it is not difficult to compute the scalar products $\langle s_{111}, s_{21}\rangle_t = t^2 - t$, $\langle s_{111}, s_3\rangle_t = t^2 - t^3$, $\langle s_{21}, s_{21}\rangle_t = (1-t)(1-t+t^2)$. From these computations we compute that

$$Q'_{111} = s_{111} + (t + t^2)s_{21} + t^3 s_3 \,.$$

There are other ways of computing the Hall–Littlewood symmetric functions. They can also be defined by means of 'creation' operators that generalize the creation operators for the Schur functions. Define

$$\mathbf{B}_m = \sum_{i,j\geq 0} (-1)^i t^j h_{m+i+j} e_i^\perp h_j^\perp = \sum_{j\geq 0} t^j \mathbf{S}_{m+j} h_j^\perp \,. \tag{1.33}$$

This family of operators [66] has the property that if $m \geq \lambda_1$,

$$\mathbf{B}_m(Q'_\lambda[X;t]) = Q'_{(m,\lambda_1,\lambda_2,\ldots,\lambda_\ell)}[X;t] \,. \tag{1.34}$$

These operators will play an important role in one of the definitions of the k-Schur functions in Sect. 3.

Sage Example 1.39. In SAGE, Example 1.38 can be checked as follows. Note that now we need to define the Schur functions over the base ring of the Hall–Littlewood functions:

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: Qp = Sym.hall_littlewood().Qp()
sage: Qp.base_ring()
Fraction Field of Univariate Polynomial Ring in t
  over Rational Field
sage: s = Sym.schur()
sage: s(Qp[1,1,1])
s[1, 1, 1] + (t^2+t)*s[2, 1] + t^3*s[3]
```

Recall the map $\theta_{qt}(q,t)$ of Eq. (1.20) which sends $p_k \mapsto p_k(1-q^k)/(1-t^k)$. Then we can transform the usual scalar product $\langle\cdot,\cdot\rangle$ to the scalar product for the Hall–Littlewood polynomials $\langle\ \cdot\ ,\ \cdot\ \rangle_t$ of (1.32) by setting $t=0$ and replacing q by t in $\theta_{qt}(q,t)$, that is $\theta_{qt}(t,0)$. We can now check our previous computation for $\langle s_{21}, s_3\rangle_t$ in SAGE as follows:

```
sage: t = Qp.t
sage: s[2,1].scalar(s[3].theta_qt(t,0))
t^2 - t
```

We can also check (1.34):

```
sage: s(Qp([1,1])).hl_creation_operator([3])
s[3, 1, 1] + t*s[3, 2] + (t^2+t)*s[4, 1] + t^3*s[5]
sage: s(Qp([3,1,1]))
s[3, 1, 1] + t*s[3, 2] + (t^2+t)*s[4, 1] + t^3*s[5]
```

1.8 Macdonald Symmetric Functions

Finally we introduce the Macdonald symmetric functions which form a basis of the space of the symmetric functions with two parameters

$$\Lambda_{q,t} = \mathbb{Q}(q,t)[h_1,h_2,h_3,\ldots]\,. \tag{1.35}$$

We provide here a definition of the Macdonald symmetric functions that generalizes those of the Hall–Littlewood and Schur functions introduced in the previous sections. The Macdonald symmetric functions $H_\lambda[X;q,t]$ are defined so that they are the unique basis which has the property that

$$H_\lambda[X;q,t] = r_\lambda(q,t)s_\lambda[X/(1-q)] + \text{ terms of the form } r_{\lambda\mu}(q,t)s_\mu[X/(1-q)]$$

with $\mu > \lambda$ and

$$\left\langle H_\lambda[X;q,t], H_\mu[X;q,t]\right\rangle_{qt} = 0 \text{ if } \lambda \neq \mu$$

where the scalar product $\langle\ \cdot\ ,\ \cdot\ \rangle_{qt}$ is defined so that

$$\left\langle p_\lambda, p_\mu\right\rangle_{qt} = z_\lambda \delta_{\lambda\mu} \prod_i (1-q^{\lambda_i})(1-t^{\lambda_i})\,.$$

In addition, the condition that $\left\langle H_\lambda[X;q,t], s_{(n)}[X]\right\rangle = t^{n(\lambda)}$ determines the correct scalar multiple of the elements.

Sage Example 1.40. Here we show how to expand the Macdonald symmetric functions in terms of Schur functions for all partitions of 3:

```
sage: Sym = SymmetricFunctions(FractionField(QQ["q,t"]))
sage: Mac = Sym.macdonald()
sage: H = Mac.H()
sage: s = Sym.schur()
sage: for la in Partitions(3):
....:     print "H", la, "=", s(H(la))
H [3] = q^3*s[1, 1, 1] + (q^2+q)*s[2, 1] + s[3]
H [2, 1] = q*s[1, 1, 1] + (q*t+1)*s[2, 1] + t*s[3]
H [1, 1, 1] = s[1, 1, 1] + (t^2+t)*s[2, 1] + t^3*s[3]
```

When $q = 0$, the expansion is upper triangular with respect to dominance order. In particular, $H_\lambda[X;0,t] = Q'_\lambda[X;t]$.

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: Mac = Sym.macdonald(q=0)
sage: H = Mac.H()
sage: s = Sym.schur()
sage: for la in Partitions(3):
....:     print "H",la, "=", s(H(la))
H [3] = s[3]
H [2, 1] = s[2, 1] + t*s[3]
H [1, 1, 1] = s[1, 1, 1] + (t^2+t)*s[2, 1] + t^3*s[3]
sage: Qp = Sym.hall_littlewood().Qp()
sage: s(Qp[1, 1, 1])
s[1, 1, 1] + (t^2+t)*s[2, 1] + t^3*s[3]
```

and when $t = 0$ it is lower triangular

```
sage: Sym = SymmetricFunctions(FractionField(QQ["q"]))
sage: Mac = Sym.macdonald(t=0)
sage: H = Mac.H()
sage: s = Sym.schur()
sage: for la in Partitions(3):
....:     print "H",la, "=", s(H(la))
H [3] = q^3*s[1, 1, 1] + (q^2+q)*s[2, 1] + s[3]
H [2, 1] = q*s[1, 1, 1] + s[2, 1]
H [1, 1, 1] = s[1, 1, 1]
```

Macdonald [119] introduced the symmetric functions that bear his name in 1988 by expanding on the work of Kevin Kadell (see notes in [120, p. 387]). Attraction to researching Macdonald polynomials grew from conjectures giving combinatorial, geometric, and representation theoretic meaning to the Macdonald/Schur coefficients

$$K_{\lambda\mu}(q,t) := \left\langle H_\mu[X;q,t], s_\lambda \right\rangle , \tag{1.36}$$

usually referred to as the Macdonald–Kostka or q, t-Kostka coefficients. From the definition that we have presented here, it is not even clear that these coefficients are polynomials in q and t. Nevertheless, Macdonald conjectured that they are in fact positive sums of monomials in q and t; that is, $K_{\lambda\mu}(q,t) \in \mathbb{N}[q,t]$. These coefficients have since been a matter of great interest.

When $q = 0$, the *Kostka-Foulkes polynomials* $K_{\lambda\mu}(0,t)$ are the Hall-Littlewood/Schur transition coefficients since $H_\lambda[X;0,t] = Q'_\lambda[X;t]$. Kostka–Foulkes polynomials appear in other contexts including affine Kazhdan–Lusztig theory [111] and affine tensor product multiplicities [129, 143]. Moreover, these polynomials encode the dimensions of certain bigraded S_n-modules [51]. Lascoux and Schützenberger [107] give an intrinsically positive formula for $K_{\lambda\mu}(0,t)$ using tableaux which we now describe.

Recall that a semi-standard tableau is a nested sequence of partitions such that consecutive partitions form a horizontal strip, also identified by a filling of a partition with the integers so that the label of the integer i indicates the cells which are added between the $(i-1)$st and ith partition in the sequence. The horizontal strip condition ensures that the entries increase strictly (resp. weakly) along columns (resp. rows). A semi-standard tableau has *weight* μ when there are μ_i labels of i. When μ forms a partition, the tableau is said to have partition weight and when the weight is $(1, 1, \ldots, 1)$, the tableau is called *standard*. A statistic (non-negative integer) called *charge*, defined in [107], can be associated to each semi-standard tableau. Here, we describe charge for any semi-standard tableaux with partition weight.

First consider the definition of charge on a standard tableau T. Define the *index* I of T, starting from $I_1 = 0$, by

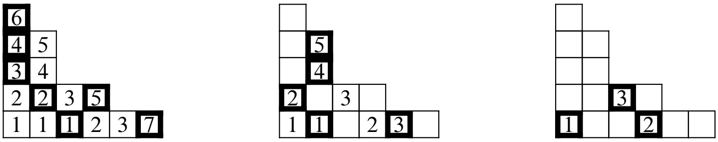
$$I_r = \begin{cases} I_{r-1} + 1 & \text{if } r \text{ is east of } r-1 \\ I_{r-1} & \text{otherwise}, \end{cases} \tag{1.37}$$

for $r = 2, \ldots, n$. The charge of T is the sum of entries in $I(T)$. The notion of charge is easily extended to a generic semi-standard tableau by successively computing the index of an appropriate choice of i cells containing the letters $1, 2, \ldots, i$.

Definition 1.41. From a specific x in cell c of a tableau T, the desired choice of $x + 1$ is the south-easternmost one lying above c. If there are none above c, the choice is the south-easternmost $x + 1$ in all of T.

Consider now any semi-standard tableau T with partition weight. Starting from the rightmost 1 in T, use Definition 1.41 to distinguish a standard sequence of i cells containing $1, 2, \ldots, i$. Compute the index and then delete all cells in this sequence. Repeat the process on the remaining cells. The total *charge* is defined to be the sum of all the index vectors.

Example 1.42.



$I = [0, 0, 0, 0, 1, 1, 2]$ $\quad$ $I = [0, 0, 1, 1, 1]$ $\quad$ $I = [0, 1, 1]$

so that the charge is 9.

We can check this in SAGE:

```
sage: t = Tableau([[1,1,1,2,3,7],[2,2,3,5],[3,4],[4,5],[6]])
sage: t.charge()
9
```

This given, with the *shape* of a semi-standard tableau T denoted by shape(T) and the charge denoted charge(T), it is proven in [107] that

$$K_{\lambda\mu}(0,t) = \sum_{\substack{\text{weight}(T)=\mu \\ \text{shape}(T)=\lambda}} t^{\text{charge}(T)}\,. \tag{1.38}$$

Despite having such concrete results for the $q = 0$ case, it was a big effort just to establish polynomiality for general $K_{\lambda\mu}(q,t)$ [53, 70, 71, 100, 141]. The geometry of Hilbert schemes was finally needed to prove positivity [61, 62], where Haiman completed his proof by showing that there is a representation theoretical model (often referred to as the "$n!$ conjecture" [50]) for which these coefficients are formulas for graded multiplicities of occurrences of irreducible representations. A formula in the spirit of (1.38) still remains a mystery.

There are other ways to express the charge. For example in [115] a different formulation of charge is used, which comes from the Ram–Yip formula for Macdonald polynomials [136], and is related to the quantum Bruhat graph, which first arose in connection with the quantum cohomology of the flag variety [24, 49]. This point of view is dual to the description above in the sense that the positions of the entries in a tableau T are recorded by columns $b_1 \cdots b_n$, where n is the value of the largest letter in T. Column b_i records the column positions of the letters i in T, where the columns of T are labelled from right to left.

We attach to $b_1 \cdots b_n$ a reordered filling $c := c_1 \cdots c_n$ according to the following algorithm, which is based on the circular order $\prec_i$ on $[m]$ starting at i, namely $i \prec_i i+1 \prec_i \cdots \prec_i m \prec_i 1 \prec_i \cdots \prec_i i-1$. Here m is the width of T.

Algorithm 1.43.
let $c_1 := b_1$;
for j from 2 to n do
  for i from 1 to height of b_j do

    let $c_j(i) := \min\left(b_j \setminus \{c_j(1), \ldots, c_j(i-1)\}, \prec_{c_{j-1}(i)}\right)$
  end do;
end do;
return $c := c_1 \ldots c_n$.

Then the charge of T is

$$\mathrm{charge}(T) = \sum_{\gamma \in \mathrm{Des}(c)} \mathrm{arm}(\gamma),$$

where $\mathrm{Des}(c)$ are all boxes in c which contain a descent with the box directly to the right and $\mathrm{arm}(\gamma)$ is the arm length of the box γ, that is the number of boxes to the right of γ.

Example 1.44. Let us consider the tableau T of Example 1.42. Then

$$\mathrm{b} = \begin{array}{|c|c|c|c|c|c|c|}\cline{1-3} 6 & 6 & 6 \\ \cline{1-5} 5 & 5 & 4 & 6 & 5 \\ \hline 4 & 3 & 2 & 5 & 3 & 6 & 1 \\ \hline \end{array} \qquad \text{and} \qquad \mathrm{c} = \begin{array}{|c|c|c|c|c|c|c|}\cline{1-3} \mathbf{6} & 3 & 4 \\ \cline{1-5} 5 & \mathbf{6} & 2 & 5 & 5 \\ \hline 4 & 5 & 6 & \mathbf{6} & 3 & \mathbf{6} & 1 \\ \hline \end{array}$$

where the boxes with descents are in bold, so that $\mathrm{charge}(T) = 2 + 3 + 3 + 1 = 9$.

1.9 *Empirical Approach to k-Schur Functions*

The k-Schur functions came out of a study of the q,t-Kostka coefficients and this study [91] led to a refinement of Macdonald's positivity conjecture; for any fixed integer $k > 0$ and each $\lambda \in \mathcal{P}^k$ (a partition where $\lambda_1 \le k$),

$$H_\lambda[X;q,t] = \sum_{\mu \in \mathcal{P}^k} K_{\mu\lambda}^{(k)}(q,t)\, A_\mu^{(k)}[X;t] \quad \text{where} \quad K_{\mu\lambda}^{(k)}(q,t) \in \mathbb{N}[q,t]\,, \tag{1.39}$$

for some family of polynomials defined by certain sets of tableaux $\mathcal{A}_\mu^k$ as:

$$A_\mu^{(k)}[X;t] = \sum_{T \in \mathcal{A}_\mu^k} t^{\mathrm{charge}(T)}\, s_{\mathrm{shape}(T)}\ . \tag{1.40}$$

In [91], Lapointe, Lascoux, and Morse conjectured that such $\{A_\mu^{(k)}[X;t]\}_{\mu_1 \le k}$ exists and forms a basis for

$$\Lambda_{(k)}^t = \mathrm{span}\{H_\lambda[X;q,t]\}_{\lambda_1 \le k}\ .$$

They also conjectured that more remarkably, for any $k' > k$,

$$A_\lambda^{(k)}[X;t] = \sum_\mu B_{\lambda,\mu}^{(k,k')}(t)\, A_\mu^{(k')}[X;t] \quad \text{where } B_{\lambda,\mu}^{(k,k')}(t) \in \mathbb{N}[t]. \tag{1.41}$$

Given that $A_\mu^{(k)}[X;t] = s_\mu$ for $k \geq |\mu|$, the decomposition (1.39) strengthens Macdonald's conjecture. Although these new bases arose in the context of Macdonald polynomials, pursuant work led to unexpected connections with geometry, physics, and representation theory. At the root, $\{A_\mu^{(k)}[X;t]\}$ generalizes the very aspects of the Schur basis that make it so fundamental and wide-reaching. As such, the functions are called *k-Schur functions*.

Before we give any formal definition for k-Schur functions, we begin with some computational examples that demonstrate how this all came about.

Example 1.45. Consider the Hall–Littlewood symmetric functions that are spanned by partitions of 3 and 4 with $\lambda_1 \leq 2$:

$$\begin{aligned} Q'_{111}[X;t] &= s_{111} + (t+t^2)s_{21} + t^3 s_3 \\ Q'_{21}[X;t] &= s_{21} + t s_3 \end{aligned} \tag{1.42}$$

and

$$\begin{aligned} Q'_{1111}[X;t] &= s_{1111} + (t+t^2+t^3)s_{211} + (t^2+t^4)s_{22} + (t^3+t^4+t^5)s_{31} + t^6 s_4 \\ Q'_{211}[X;t] &= s_{211} + t s_{22} + (t+t^2)s_{31} + t^3 s_4 \\ Q'_{22}[X;t] &= s_{22} + t s_{31} + t^2 s_4\,. \end{aligned}$$

The first clue of the existence of 2-Schur functions is to notice that there is a set of symmetric functions $A_\lambda^{(2)}[X;t]$ for $\lambda \vdash 3, 4$ such that $\lambda_1 \leq 2$ and

- They form a basis for this subset of Hall–Littlewood symmetric functions that have coefficients that are non-negative polynomials in t when expanded in the Schur basis;
- When expanded in the Hall–Littlewood basis have a term $Q'_\lambda[X;t]$ and all other terms are larger in dominance order;
- Have a leading term in the Schur basis which is s_λ and are in the linear span of Schur functions indexed by partitions which are larger than λ in dominance order;
- If the involution ω is applied and t is replaced by $1/t$, then the 2-Schur function is equal to another 2-Schur function up to a power of t;
- The Schur function indexed by the partition which is largest in dominance order is a power of t times $s_{(\lambda^{\omega_k})'}$ (indexed by the conjugate of the k-conjugate of λ);

- For which the Macdonald symmetric functions are positive when expressed in terms of these elements (have coefficients which are polynomials in q and t with non-negative integer coefficients).

It turns out that these conditions are enough to characterize the 2-Schur functions. The triangularity conditions with respect to the Hall–Littlewood polynomials and the Schur functions require that $A^{(2)}_{22}[X;t] = s_{22} + ts_{31} + t^2s_4$. Then again by triangularity and positivity we have $A^{(2)}_{211}[X;t] = s_{211} + ts_{31}$, and finally $A^{(2)}_{1111}[X;t] = s_{1111} + ts_{211} + t^2s_{22}$.

Sage Example 1.46. The conditions involving the k-conjugate of the partition can easily be checked

```
sage: la = Partition([2,2])
sage: la.k_conjugate(2).conjugate()
[4]
sage: la = Partition([2,1,1])
sage: la.k_conjugate(2).conjugate()
[3, 1]
sage: la = Partition([1,1,1,1])
sage: la.k_conjugate(2).conjugate()
[2, 2]
```

as well as the statement about the application of ω composed with $t \mapsto 1/t$:

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks = Sym.kschur(2)
sage: ks[2,2].omega_t_inverse()
1/t^2*ks2[1, 1, 1, 1]
sage: ks[2,1,1].omega_t_inverse()
1/t*ks2[2, 1, 1]
sage: ks[1,1,1,1].omega_t_inverse()
1/t^2*ks2[2, 2]
```

We can also check the positive expansion of the Macdonald polynomials:

```
sage: Sym = SymmetricFunctions(FractionField(QQ["q,t"]))
sage: H = Sym.macdonald().H()
sage: ks = Sym.kschur(2)
sage: ks(H[2,2])
q^2*ks2[1, 1, 1, 1] + (q*t+q)*ks2[2, 1, 1] + ks2[2, 2]
sage: ks(H[2,1,1])
q*ks2[1, 1, 1, 1] + (q*t^2+1)*ks2[2, 1, 1] + t*ks2[2, 2]
sage: ks(H[1,1,1,1])
ks2[1, 1, 1, 1] + (t^3+t^2)*ks2[2, 1, 1] + t^4*ks2[2, 2]
```

At $k = 3$, the above conditions are no longer a complete characterization of the k-Schur functions and it is difficult to guess what $A^{(3)}_\lambda$ is for some of the partitions at $\lambda \vdash 7$. We add as a last condition that the k-Schur functions should be the 'smallest' basis with the above properties. The existence of such a basis alone is enough to recognize that there is something remarkable about the k-Schur functions, because it is unusual to see a basis for which the Macdonald symmetric functions expand positively.

Example 1.47. At $k = 3$, the triangularity condition for the Hall–Littlewood basis implies that,

$$A^{(3)}_{331}[X;t] = Q'_{331}[X;t] = s_{331} + ts_{421} + (t+t^2)s_{43} + t^2 s_{511} + (t^2+t^3)s_{52} + (t^3+t^4)s_{61} + t^5 s_7$$

and since

$$Q'_{322}[X;t] = s_{322} + ts_{331} + (t+t^2)s_{421} + (t^2+t^3)s_{43} + t^3 s_{511} + (t^2+t^3+t^4)s_{52} + (t^4+t^5)s_{61} + t^6 s_7,$$

we can use the triangularity and positivity properties from above that imply $A^{(3)}_{322}[X;t]$ must be given by

$$A^{(3)}_{322}[X;t] = Q'_{322}[X;t] - tQ'_{331} = s_{322} + ts_{421} + t^2 s_{52}\,.$$

However, to try to determine $A^{(3)}_{3211}[X;t]$, we calculate

$$Q'_{3211}[X;t] = s_{3211} + ts_{322} + (t+t^2)s_{331} + ts_{4111} + (t+t^2+t^3)s_{421} + (t^2+t^3+t^4)s_{43} + (t^2+t^3+t^4)s_{511} + (2t^3+t^4+t^5)s_{52} + (t^4+t^5+t^6)s_{61} + t^7 s_7$$

and it is difficult to tell from the conditions above whether $A^{(3)}_{3211}$ should be $Q'_{3211}[X;t] - t^2 Q'_{331}[X;t]$ or $Q'_{3211}[X;t] - tQ'_{322}[X;t]$ or some other linear combination of terms.

We leave it as an exercise for the reader to calculate $A^{(2)}_{111}[X;t]$, $A^{(2)}_{21}[X;t]$ from (1.42), using the properties in Example 1.45 as a characterization. With the additional symmetric function $Q'_{31}[X;t] = s_{31} + ts_4$, it is a worthwhile exercise to determine the symmetric functions $A^{(3)}_{31}[X;t]$, $A^{(3)}_{22}[X;t]$, $A^{(3)}_{211}[X;t]$ and $A^{(3)}_{1111}[X;t]$ to see how it might be possible to define the k-Schur functions for small values by experimentation.

Once it is clear that these symmetric functions exist, it is a matter of determining an algorithm or a formula for computing them. It was along these lines that Lapointe, Lascoux, and Morse discovered k-Schur functions and their first formula appeared in [94] where a method is given for constructing the sets $\mathcal{A}_\lambda$ in (1.40). Subsequently, various conjecturally equivalent definitions have arisen, each having different benefits and detriments. In the following section we present a construction for k-Schur functions when the parameter t is set to one and the parameter case is then addressed in Sect. 3.

1.10 Notes on References

The combinatorics of affine permutations can be expressed in terms of combinatorial models other than k-bounded partitions and $(k+1)$-cores such as abaci, windows, codes, and k-castles [12, 21, 38, 41, 103]. Certain combinatorial aspects of k-Schur functions are best expressed in terms of k-bounded partitions whereas others are better suited to $(k+1)$-cores or affine permutations. It could be that other formulations for the index set are well suited for expressing properties which have not yet been discovered.

For symmetric function notation we generally follow the notation of Macdonald [120] with the addition of the use of plethystic notation (see Eq. (1.19)) to encode certain transformations of alphabets. The scalar product $\langle \cdot , \cdot \rangle_t$ defined in Sect. 1.7 and $\langle \cdot , \cdot \rangle_{qt}$ defined in Sect. 1.8 are not the scalar products that are used in [120], but they are the scalar products needed in order to define the Hall–Littlewood Q'-basis and the Macdonald H-basis that we will use in subsequent sections. Macdonald does not use the H-basis in [120], however it is a transformation of the basis referred to as the integral basis (the J-basis) and they are related by the transformation $J_\lambda[X; q, t] = H_\lambda[X(1-t); q, t] = \theta_{t0}(H_\lambda[X; q, t])$ (see, for instance, [50, Eq. I.16]).

The operators $\mathbf{S}_m$ defined in Eq. (1.30) are usually referred to as Bernstein operators. They first appear in [159, p. 69] (see also [120, Example 29 Sect. I.5]). Their generalizations to creation operators for Hall–Littlewood symmetric functions is due to Jing [66] (see also [120, Example 8 Sect. III.5]).

2 From Pieri Rules to k-Schur Functions at $t = 1$

We first present the definition of the Schur functions as a generating function for semi-standard tableaux. This presentation provides the context to show how the k-Schur functions and the dual k-Schur functions both generalize this definition of the Schur functions. We carefully explain how the algebraic Pieri rule naturally gives rise to distinguished sequences of partitions and show how the combinatorics of 'weak' and 'strong' tableaux captures these relations.

2.1 Semi-standard Tableaux and a Monomial Expansion of Schur Functions

Recall that the Schur functions satisfy the *Pieri rule* for multiplication of a Schur function by a homogeneous symmetric function

$$h_1 s_\lambda = \sum_{\mu:\lambda\to\mu} s_\mu \tag{2.1}$$

and more generally,

$$h_r s_\lambda = \sum_{\mu} s_\mu \ , \tag{2.2}$$

where the sum is over all partitions μ where $\lambda \subseteq \mu$, $|\mu| = |\lambda| + r$ and μ/λ is a horizontal strip.

Consider how we can use these rules now to expand a homogeneous symmetric function in terms of the Schur basis. A given $h_\lambda = h_{\lambda_1} h_{\lambda_2} \cdots h_{\lambda_\ell}$ may be seen as a sequence of operators that act on $1 = s_\emptyset$ which we act on, first by h_{λ_1}, then h_{λ_2}, and so on. We demonstrate this with the following example.

Example 2.1. We expand the expression h_{431} in terms of Schur functions by successive applications of the Pieri rule:

$$\begin{aligned}
h_{431} &= s_{(4)} h_{31} = \left(s_{(4,3)} + s_{(5,2)} + s_{(6,1)} + s_{(7)}\right) h_1 \\
&= \Big(s_{(4,3,1)} + s_{(4,4)} + s_{(5,3)} + s_{(5,2,1)} + s_{(5,3)} + s_{(6,2)} \\
&\qquad + s_{(6,1,1)} + s_{(6,2)} + s_{(7,1)} + s_{(7,1)} + s_{(8)}\Big) \\
&= s_{(4,3,1)} + s_{(4,4)} + s_{(5,2,1)} + 2s_{(5,3)} + s_{(6,1,1)} + 2s_{(6,2)} + 2s_{(7,1)} \\
&\qquad + s_{(8)} \ .
\end{aligned}$$

Notice in the example that every term in the expansion of the product of h's is represented by a sequence of partitions which records how that term appears in the final expression. For instance there are two terms representing $(5,3)$, one that arose from the sequence $\emptyset \subseteq (4) \subseteq (5,2) \subseteq (5,3)$ and the other which arose from the sequence $\emptyset \subseteq (4) \subseteq (4,3) \subseteq (5,3)$. A sequence

$$\lambda^{(0)} \subseteq \lambda^{(1)} \subseteq \cdots \subseteq \lambda^{(d)} \ , \tag{2.3}$$

where each $\lambda^{(i+1)}/\lambda^{(i)}$ for $0 \le i < d$ is a horizontal strip, is called a *semi-standard skew tableau*. When $\lambda^{(0)}$ is empty, the sequence is called a *semi-standard tableau* (see also Sect. 1.8). We say that the *shape* of the tableau is $\lambda^{(d)}/\lambda^{(0)}$ (or $\lambda^{(d)}$ if $\lambda^{(0)} = \emptyset$) and the weight of the tableau is the sequence

$$(|\lambda^{(1)}/\lambda^{(0)}|, |\lambda^{(2)}/\lambda^{(1)}|, \ldots, |\lambda^{(d)}/\lambda^{(d-1)}|) . \tag{2.4}$$

Note that for any partition there is precisely one semi-standard tableau that has both shape and weight equal to λ. If λ and μ are both partitions of the same size, then there is at least one partition of shape λ and weight μ if and only if $\lambda \geq \mu$.

A semi-standard tableau of shape λ is usually thought of as a filling of the partition diagram for λ by placing a 1 in each of the cells of $\lambda^{(1)}/\lambda^{(0)}$, a 2 in each of the cells of $\lambda^{(2)}/\lambda^{(1)}$, and more generally each of the cells of $\lambda^{(i)}/\lambda^{(i-1)}$ is labelled with an i. Note, iteration of the special case (2.1) ensures that each $\lambda^{(i)}/\lambda^{(i-1)}$ contains exactly one cell. These are the tableaux of weight $(1, 1, \ldots, 1)$ and they are called *standard* tableaux.

Example 2.2. The coefficient of s_{52} in h_{421} is equal to 2. The reason for this is that one term comes from

$$\begin{array}{l}\\ \square\square\square\square\end{array} \subseteq \begin{array}{l}\square\square\\ \square\square\square\square\end{array} \subseteq \begin{array}{l}\square\square\\ \square\square\square\square\square\end{array} \text{ which is represented by the diagram } \begin{array}{|c|c|c|c|c|}\cline{1-2} 2 & 2 \\ \hline 1 & 1 & 1 & 1 & 3 \\ \hline\end{array}$$

and the other comes from

$$\begin{array}{l}\\ \square\square\square\square\end{array} \subseteq \begin{array}{l}\square\\ \square\square\square\square\square\end{array} \subseteq \begin{array}{l}\square\square\\ \square\square\square\square\square\end{array} \text{ which is represented by the diagram } \begin{array}{|c|c|c|c|c|}\cline{1-2} 2 & 3 \\ \hline 1 & 1 & 1 & 1 & 2 \\ \hline\end{array} .$$

Remark 2.3. Another useful alternative to the conventional definition of tableaux is to instead define a semi-standard tableau to be a standard tableau with certain conditions on its reading word. The reading word of a tableau is obtained by taking the entries of T from top to bottom and left to right. A tableau of weight α is then a standard tableau having increasing reading words in the alphabets

$$\mathcal{A}_{\alpha,x} = [1 + \Sigma^{x-1}\alpha, \Sigma^{x}\alpha] \quad \text{where} \quad \Sigma^{x}\alpha = \sum_{i \leq x} \alpha_i , \tag{2.5}$$

for each $x = 1, \ldots, \ell(\alpha)$. For example, this observation is central in the study of quasisymmetric functions.

It is typical to represent the number of semi-standard tableaux of shape λ and weight μ by the symbol $K_{\lambda\mu}$. These numbers are often referred as the *Kostka* coefficients. Based on the Pieri rule and a relatively straightforward proof by induction, we can generalize roughly what we see in the example, namely that for $\mu \vdash m$,

$$h_{\mu} = \sum_{\lambda \vdash m} K_{\lambda\mu} s_{\lambda} . \tag{2.6}$$

From this, the monomial expansion of a Schur function can be derived. In particular, because $\langle s_\lambda, s_\mu \rangle = \delta_{\lambda\mu}$, we can conclude that $\langle h_\mu, s_\lambda \rangle = K_{\lambda\mu}$. The coefficient in the monomial expansion of a Schur function then pops out by the duality $\langle m_\lambda, h_\mu \rangle = \delta_{\lambda\mu}$:

$$s_\lambda = \sum_\mu d_{\mu\lambda}\, m_\mu \quad \text{where} \quad d_{\mu\lambda} = \langle h_\mu, \sum_\alpha d_{\alpha\lambda} m_\alpha \rangle = \langle h_\mu, s_\lambda \rangle = K_{\lambda\mu}\,.$$

This formula provides us with a combinatorial expansion of the Schur functions in the monomial basis. We can also derive a formula in terms of variables. Note that the construction of tableaux with weight α applies for any composition α. Let $\mathrm{SSYT}(\lambda, \alpha)$ be the set of tableaux with weight α and shape λ. There is an involution [10, 109] on this set that maps a tableau of weight α to a tableau whose weight is a permutation of α. Thus, given that $K_{\lambda\alpha} = K_{\lambda\sigma(\alpha)}$,

$$s_\lambda[X_m] = \sum_T \mathbf{x}^T \,, \tag{2.7}$$

where the sum is over all possible semi-standard tableaux of shape λ with entries in $\{1, 2, \ldots, m\}$. Here $\mathbf{x}^T$ denotes $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_m^{\alpha_m}$, where α is the weight of T.

Sage Example 2.4. We now demonstrate on how to produce all semi-standard tableaux of a given shape and weight

```
sage: SemistandardTableaux([5,2],[4,2,1]).list()
[[[1, 1, 1, 1, 2], [2, 3]], [[1, 1, 1, 1, 3], [2, 2]]]
```

The Kostka matrix can be computed as follows:

```
sage: P = Partitions(4)
sage: P.list()
[[4], [3, 1], [2, 2], [2, 1, 1], [1, 1, 1, 1]]
sage: n = P.cardinality(); n
5
sage: K = matrix(QQ,n,n,
....:     [[SemistandardTableaux(la,mu).cardinality()
....:     for mu in P] for la in P])
sage: K
[1 1 1 1 1]
[0 1 1 2 3]
[0 0 1 1 2]
[0 0 0 1 3]
[0 0 0 0 1]
```

The self duality of the Schur functions is a very remarkable property. In fact, the self duality and a condition on triangularity can be taken as either the defining property for Schur functions or as property which easily follows from the definition.

It is this duality that implies that $\langle h_\mu, s_\lambda \rangle$ will be the coefficient of s_λ in the expansion of h_μ. In the next two sections we shall introduce two bases of a subalgebra/quotient algebra which follows from ideas in this construction.

2.2 Weak Tableaux and a Monomial Expansion of Dual k-Schur Functions

We will present the k-Schur functions as a basis of the space

$$\Lambda_{(k)} = \mathbb{Q}[h_1, h_2, h_3, \ldots, h_k]\,. \tag{2.8}$$

It will develop that this basis has a role in $\Lambda_{(k)}$ that in many ways is analogous to the role of the Schur basis in the study of the symmetric function space Λ. The algebra of $\Lambda_{(k)}$ is no longer a self dual Hopf algebra. The algebra is dual however to a quotient of the ring of symmetric functions. We define

$$\Lambda^{(k)} = \Lambda / \langle m_\lambda : \lambda_1 > k \rangle\,. \tag{2.9}$$

A basis of $\Lambda_{(k)}$ are the elements h_λ for partitions λ where $\lambda_1 \le k$. The elements m_λ with $\lambda_1 \le k$ may be chosen as representatives of the dual algebra. We will present two bases: the *k-Schur functions* $s^{(k)}_\lambda$ which form a basis for $\Lambda_{(k)}$ and the *dual k-Schur functions* $\tilde{F}^{(k)}_\lambda$ which are representative elements of the basis for the dual algebra $\Lambda^{(k)}$. We proceed by defining the basis of k-Schur functions and the other will be determined by duality.

Recall by Proposition 1.3 that the k-bounded partitions of size m are in bijection with the $(k+1)$-cores of length m (or equivalently cores with m cells with hook length of size less than or equal to k). At this point, we keep in mind that the k-bounded partition λ that serves as an index for either a k-Schur function or a dual k-Schur function represents the shape of a $k+1$-core, $\mathfrak{c}(\lambda)$. Sometimes we will use the properties of the partition λ and other times we will use the corresponding core, $\mathfrak{c}(\lambda)$. Later, we will interpret things in the language of k-bounded partitions and affine Grassmannian elements.

We will define the k-Schur functions at $t = 1$ based on a *weak Pieri rule* for k-Schur functions (which at this point exists only as data on the computer because we deduced what it must be from the atom definition in the last chapter). This is the relation, for a k-bounded partition μ and $r \le k$,

$$h_r\, s^{(k)}_\mu = \sum_\lambda s^{(k)}_\lambda\,, \tag{2.10}$$

summing over k-bounded partitions λ where $\mathfrak{c}(\lambda)/\mathfrak{c}(\mu)$ is a weak horizontal strip of size r (see Eq. (1.5)).

Example 2.5. With $k = 3$, to compute $h_1 s_{31}^{(3)}$, we find all 4-cores that cover $\mathfrak{c}(3,1) = (4,1)$ in the weak order poset Fig. 2.3 and there are two, one of shape $\mathfrak{c}(3,2) = (5,2)$ and one of shape $\mathfrak{c}(3,1,1) = (4,1,1)$:

$$h_1 s_{31}^{(3)} = s_{32}^{(3)} + s_{311}^{(3)} .$$

To compute $h_2 s_{31}^{(3)}$, we follow the weak horizontal chains of length 2 from $(4,1)$ in Fig. 2.3 and notice that there are two of shape $\mathfrak{c}(3,3) = (6,3)$ and $\mathfrak{c}(3,2,1) = (5,2,1)$:

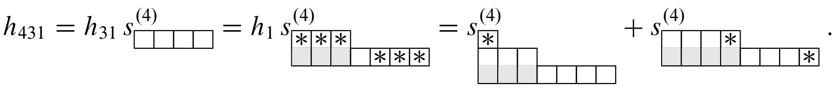
$$h_2 s_{31}^{(3)} = s_{33}^{(3)} + s_{321}^{(3)} .$$

Note that there is a length 2 chain from $\mathfrak{c}(3,1) = (4,1)$ to $\mathfrak{c}(3,1,1,1) = (4,1,1,1)$, but because $(4,1,1,1)/(4,1)$ is not a weak horizontal strip, this term is omitted. There is only one horizontal chain of length 3 from $\mathfrak{c}(3,1) = (4,1)$ implying that

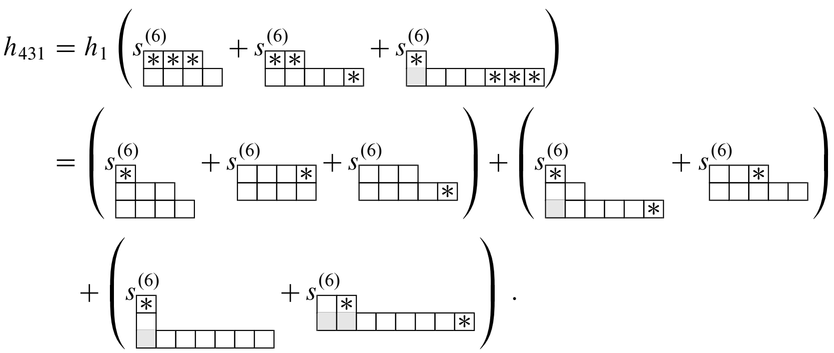
$$h_3 s_{31}^{(3)} = s_{331}^{(3)} .$$

As we have shown for usual Schur functions, the iteration of the Pieri rule can be used to inspire a family of tableaux where, in this case, their enumeration gives the coefficient of $s_\lambda^{(k)}$ in the expansion of $h_\mu s_\emptyset^{(k)}$. Consider the following example.

Example 2.6. For $k = 4$, we compute h_{431} in terms of k-Schur functions. We will index the k-Schur function by a diagram for a 5-core with the added cells indicated by an $*$. The indexing 4-bounded partition can be read off of these diagrams by counting only the non-grey cells in each row.



Let $k = 6$ and compute an example with more terms:



The iteration imposes the conditions needed to characterize a *weak tableau*; it must be a sequence of $(k+1)$-cores,

$$\emptyset = \kappa^{(0)} \subseteq \kappa^{(1)} \subseteq \cdots \subseteq \kappa^{(d)} = \mathfrak{c}(\lambda) \tag{2.11}$$

such that $\kappa^{(i)}/\kappa^{(i-1)}$ is a weak horizontal strip. We say that its *shape* is $\mathfrak{c}(\lambda)$ and define its weight α as we did in Eq. (2.4), but now using the size function on cores;

$$\alpha_i = |\kappa^{(i)}/\kappa^{(i-1)}|_{k+1}\,, \quad \text{for } i = 1, \ldots, d\,. \tag{2.12}$$

Let us emphasize that an entry α_i of the weight does *not* record the number of times letter i appears in the tableau. In fact, using (1.8), we see that instead, α_i records the number of distinct residues used to label the cells of $\kappa^{(i)}/\kappa^{(i-1)}$. We can more concisely describe weak tableaux in the following terms:

Proposition 2.7 ([96]). *Let κ be a $(k+1)$-core and let $\alpha = (\alpha_1, \ldots, \alpha_d)$ be a composition of $|\kappa|_{k+1}$ with no part larger than k. A weak tableau of weight α is a semi-standard filling of shape κ with letters $1, \ldots, d$ such that the collection of cells filled with letter i is labeled by exactly α_i distinct $(k+1)$-residues.*

Example 2.8. For $k = 6$, the weak tableaux of weight $(4, 3, 1)$ are

$$\begin{array}{|c|}\hline 3\\\hline\end{array}\!\!\begin{array}{l} \end{array}
\begin{array}{llll} 3 & & & \\ 2 & 2 & 2 & \\ 1 & 1 & 1 & 1 \end{array} \qquad
\begin{array}{llll} 2 & 2 & 2 & 3 \\ 1 & 1 & 1 & 1 \end{array} \qquad
\begin{array}{lllll} 2 & 2 & 2 & & \\ 1 & 1 & 1 & 1 & 3 \end{array} \qquad
\begin{array}{lllll} 2 & 2 & 3 & & \\ 1 & 1 & 1 & 1 & 2 \end{array}$$

$$\begin{array}{llllll} 3 & & & & & \\ 2 & 2 & & & & \\ 1 & 1 & 1 & 1 & 2 & 3 \end{array} \qquad
\begin{array}{lllllll} 3 & & & & & & \\ 2 & & & & & & \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 \end{array} \qquad
\begin{array}{llllllll} 2 & 3 & & & & & & \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 3 \end{array}\,.$$

For $k = 3$, we list all weak tableaux of shape $(5, 2, 1)$ which can be extracted by looking at all successions of horizontal chains (with non-increasing sizes) in Fig. 2.3.

$$\begin{array}{lllll} 6 & & & & \\ 4 & 5 & & & \\ 1 & 2 & 3 & 4 & 5 \end{array} \qquad
\begin{array}{lllll} 5 & & & & \\ 4 & 6 & & & \\ 1 & 2 & 3 & 4 & 6 \end{array} \qquad
\begin{array}{lllll} 4 & & & & \\ 3 & 6 & & & \\ 1 & 2 & 4 & 5 & 6 \end{array} \qquad
\begin{array}{lllll} 4 & & & & \\ 2 & 6 & & & \\ 1 & 3 & 4 & 5 & 6 \end{array}$$

$$\begin{array}{lllll} 5 & & & & \\ 3 & 4 & & & \\ 1 & 1 & 2 & 3 & 4 \end{array} \qquad
\begin{array}{lllll} 4 & & & & \\ 3 & 5 & & & \\ 1 & 1 & 2 & 3 & 5 \end{array} \qquad
\begin{array}{lllll} 3 & & & & \\ 2 & 5 & & & \\ 1 & 1 & 3 & 4 & 5 \end{array} \qquad
\begin{array}{lllll} 4 & & & & \\ 2 & 3 & & & \\ 1 & 1 & 2 & 2 & 3 \end{array} \qquad
\begin{array}{lllll} 3 & & & & \\ 2 & 4 & & & \\ 1 & 1 & 2 & 2 & 4 \end{array}$$

$$\begin{array}{lllll} 3 & & & & \\ 2 & 3 & & & \\ 1 & 1 & 2 & 2 & 3 \end{array} \qquad
\begin{array}{lllll} 4 & & & & \\ 2 & 3 & & & \\ 1 & 1 & 1 & 2 & 3 \end{array} \qquad
\begin{array}{lllll} 3 & & & & \\ 2 & 4 & & & \\ 1 & 1 & 1 & 2 & 4 \end{array} \qquad
\begin{array}{lllll} 3 & & & & \\ 2 & 2 & & & \\ 1 & 1 & 1 & 2 & 2 \end{array}$$

Sage Example 2.9. In SAGE we can list all weak k-tableaux for a given shape and weight. For example, the two weak 6-tableaux of weight $(4, 3, 1)$ and shape $(5, 3)$ can be obtained as follows:

```
sage: T = WeakTableaux(6, [5,3], [4,3,1])
sage: T.list()
[[[1, 1, 1, 1, 3], [2, 2, 2]], [[1, 1, 1, 1, 2], [2, 2, 3]]]
```

The 13 weak 3-tableaux of shape $(5, 2, 1)$ in Example 2.8 can be obtained as

```
sage: k = 3
sage: c = Core([5,2,1], k+1)
sage: la = c.to_bounded_partition(); la
[3, 2, 1]
sage: for mu in Partitions(la.size(), max_part = 3):
....:     T = WeakTableaux(k, c, mu)
....:     print "weight", mu
....:     print T.list()
....:
weight [3, 3]
[]
weight [3, 2, 1]
[[[1, 1, 1, 2, 2], [2, 2], [3]]]
weight [3, 1, 1, 1]
[[[1, 1, 1, 2, 4], [2, 4], [3]],
 [[1, 1, 1, 2, 3], [2, 3], [4]]]
weight [2, 2, 2]
[[[1, 1, 2, 2, 3], [2, 3], [3]]]
weight [2, 2, 1, 1]
[[[1, 1, 2, 2, 4], [2, 4], [3]],
 [[1, 1, 2, 2, 3], [2, 3], [4]]]
weight [2, 1, 1, 1, 1]
[[[1, 1, 3, 4, 5], [2, 5], [3]],
 [[1, 1, 2, 3, 5], [3, 5], [4]],
 [[1, 1, 2, 3, 4], [3, 4], [5]]]
weight [1, 1, 1, 1, 1, 1]
[[[1, 3, 4, 5, 6], [2, 6], [4]],
 [[1, 2, 4, 5, 6], [3, 6], [4]],
 [[1, 2, 3, 4, 6], [4, 6], [5]],
 [[1, 2, 3, 4, 5], [4, 5], [6]]]
```

For any k-bounded partition λ and k-bounded partition μ, we denote the number of weak tableaux of shape $\mathfrak{c}(\lambda)$ and weight μ by $K^{(k)}_{\lambda\mu}$. These numbers, called *(weak) k-Kostka coefficients*, satisfy an important property

$$K^{(k)}_{\lambda\mu} = \begin{cases} 1 & \text{if} \quad \lambda = \mu \\ 0 & \text{if} \quad \lambda \not\geq \mu\,, \end{cases} \tag{2.13}$$

with respect to dominance order on partitions. Thus, the matrix of coefficients $||K^{(k)}_{\lambda\mu}||$ over all k-bounded partitions λ and μ of the same size is unitriangular and thus invertible. It is with this in hand that we arrive at a family of functions that satisfy the weak Pieri rule. To be precise, k-Schur functions were characterized in [92] by the system obtained by taking

$$h_\mu = \sum_{\lambda\,:\,\lambda_1 \le k} K^{(k)}_{\lambda\mu} s^{(k)}_\lambda \,, \tag{2.14}$$

for all k-bounded partitions μ. In fact, this system defines the k-Schur basis $\{s^{(k)}_\lambda\}_{\lambda_1 \le k}$ because the elements h_λ for $\lambda_1 \le k$ form a basis for the space $\Lambda_{(k)}$ and the transition matrix is invertible over the integers.

Example 2.10. For $k = 6$, the weak tableaux in Example 2.8 tell us that

$$h_{431} = s^{(6)}_{431} + s^{(6)}_{44} + 2s^{(6)}_{53} + s^{(6)}_{521} + s^{(6)}_{611} + s^{(6)}_{62} \,.$$

In Sect. 3, various different notions of k-Schur functions will be given. We will work with a running example and do a computation to give an idea of how each of the notions can be implemented and to demonstrate their relative difficulty.

Example 2.11. Let us calculate $s^{(3)}_{3211}$ in terms of homogeneous symmetric functions. The k-Schur function $s^{(k)}_\lambda$ can be computed recursively using the weak Pieri rule. The product $h_{\lambda_1} s^{(k)}_{(\lambda_2,\lambda_3,\ldots,\lambda_\ell)} = s^{(k)}_\lambda +$ other terms which are indexed by k bounded partitions γ where λ is smaller than γ in dominance order.

We begin by noting that $h_3 s^{(3)}_{311} = s^{(3)}_{3211}$, and we thus must expand $s^{(3)}_{211}$. This time, noting that $h_2 s^{(3)}_{11} = s^{(3)}_{211}$ we turn to the computation of $s^{(3)}_{11}$. Since $h_1 s^{(3)}_1 = s^{(3)}_{11} + s^{(3)}_2$ and $s^{(3)}_2 = h_2$, we have computed $s^{(3)}_{11} = h_{11} - h_2$ which can be substituted back to obtain $s^{(3)}_{211} = h_{211} - h_{22}$. We conclude that $s^{(3)}_{3211} = h_{3211} - h_{322}$.

We leave it to the reader as an exercise to compute at least a few other of the 3-Schur functions of size 7 by hand to get a feel for the difficulty of these computations. Fortunately, the functions can be verified against a calculation using SAGE.

Sage Example 2.12. Here we give the expansion of the k-Schur functions for $k = 3$ in terms of the homogeneous basis as we did in the previous example. Our current setting is the $t = 1$ case.

```
sage: Sym = SymmetricFunctions(QQ)
sage: ks = Sym.kschur(3,t=1)
sage: h = Sym.homogeneous()
sage: for mu in Partitions(7, max_part =3):
....:     print h(ks(mu))
....:
h[3, 3, 1]
h[3, 2, 2] - h[3, 3, 1]
h[3, 2, 1, 1] - h[3, 2, 2]
h[3, 1, 1, 1, 1] - 2*h[3, 2, 1, 1] + h[3, 3, 1]
h[2, 2, 2, 1] - h[3, 2, 1, 1] - h[3, 2, 2] + h[3, 3, 1]
h[2, 2, 1, 1, 1] - 2*h[2, 2, 2, 1] - h[3, 1, 1, 1, 1]
 + 2*h[3, 2, 1, 1] + h[3, 2, 2] - h[3, 3, 1]
```

```
h[2, 1, 1, 1, 1, 1] - 3*h[2, 2, 1, 1, 1] + 2*h[2, 2, 2, 1]
 + h[3, 2, 1, 1] - h[3, 2, 2]
h[1, 1, 1, 1, 1, 1, 1] - 4*h[2, 1, 1, 1, 1, 1]
 + 4*h[2, 2, 1, 1, 1] + 2*h[3, 1, 1, 1, 1] - 4*h[3, 2, 1, 1]
 + h[3, 3, 1]
```

We are now in the position to use duality to produce a second basis, this time for the algebra $\Lambda^{(k)}$. Although we do not have a scalar product on the spaces $\Lambda_{(k)}$ and $\Lambda^{(k)}$ separately, we appeal to a pairing between the two spaces

$$\langle\ \cdot\ ,\ \cdot\ \rangle\ :\ \Lambda_{(k)} \times \Lambda^{(k)} \to \mathbb{Q}\,, \tag{2.15}$$

where $h_\mu \in \Lambda_{(k)}$ and $m_\lambda \in \Lambda^{(k)}$ are dual elements

$$\left\langle h_\mu, m_\lambda \right\rangle = \delta_{\lambda\mu}\ . \tag{2.16}$$

This equation is precisely Eq. (1.23) for the scalar product on symmetric functions.

The dual k-Schur functions $\tilde{F}_\lambda^{(k)}$ were introduced in [93] as the unique basis of the degree m subspace of $\Lambda^{(k)}$ (with $m \geq 1$) that is dual to the basis of $\{s_\lambda^{(k)}\}_{\lambda \vdash m, \lambda_1 \leq k}$ under the pairing (2.15). It follows from (2.14) that the enumeration of weak tableaux gives their monomial expansion:

$$\begin{aligned}
\tilde{F}_\lambda^{(k)} &= \sum_{\mu:\mu_1 \leq k} \left\langle h_\mu, \tilde{F}_\lambda^{(k)} \right\rangle m_\mu \\
&= \sum_{\mu:\mu_1 \leq k}\ \sum_{\gamma:\gamma_1 \leq k} K_{\gamma\mu}^{(k)} \left\langle s_\gamma^{(k)}, \tilde{F}_\lambda^{(k)} \right\rangle m_\mu \\
&= \sum_{\mu:\mu_1 \leq k} K_{\lambda\mu}^{(k)} m_\mu\ .
\end{aligned} \tag{2.17}$$

There is an involution on the set of weak tableaux of fixed shape $\mathfrak{c}(\lambda)$ and weight α that sends these tableaux to the set of weak tableaux of shape $\mathfrak{c}(\lambda)$ and weight which is a permutation of α. Thus, the dual k-Schur functions are the weight generating functions for weak tableaux. That is, for $(k+1)$-core λ,

$$\tilde{F}_\lambda^{(k)} = \sum_{\substack{T=\text{weak tab} \\ \text{shape}(T)=\mathfrak{c}(\lambda)}} x^{\text{weight}(T)}\,. \tag{2.18}$$

Note that what we call the dual k-Schur functions here are also equal to affine Stanley symmetric functions indexed by affine Grassmannian elements. This connection is discussed with more detail in Chap. 3, Sect. 8.2 and Chap. 2, Sect. 2.5. Note that our notation differs slightly from the notation in Chap. 3 by adding a superscript indicating k.

Example 2.13. The calculation of the dual k-Schur function $\tilde{F}^{(3)}_{321}$ follows immediately by extracting the weights of each weak tableau of shape $(5, 2, 1)$, listed in Example 2.8. We conclude that

$$\tilde{F}^{(3)}_{321} = m_{321} + 2m_{3111} + m_{222} + 2m_{2211} + 3m_{21111} + 4m_{111111} \,.$$

2.3 Other Realizations

We have now seen how the weak Pieri rule – given in terms of weak horizontal chains in the $(k + 1)$-core realization of the weak poset – leads to the family of dual k-Schur functions in terms of weak tableaux. Equivalently, we could have invoked the definition of weak horizontal chains on the level of k-bounded partitions or on affine Grassmannian elements and this would easily give rise to characterizations for dual k-Schur functions in these other settings. Before we move on to draw a weight generating function characterization for k-Schur functions starting instead from a "strong" Pieri rule, some exposition on these other interpretations is warranted.

Let us start by retracing our steps that led from the weak Pieri rule to the generating function for dual k-Schur functions, this time in the setting of k-bounded partitions. In these terms, the weak Pieri relation is given for any partition λ with $\lambda_1 \leq k$ to be

$$h_r s^{(k)}_\lambda = \sum_{\mu:\mu_1 \leq k} s^{(k)}_\mu \,, \tag{2.19}$$

where the sum is over partitions μ such that μ/λ is a horizontal strip and $\mu^{\omega_k}/\lambda^{\omega_k}$ is a vertical strip of size r. Again, we use the iteration of this relation to impose conditions on a family of weak tableaux in the k-bounded setting.

Example 2.14. Iteratively, we calculate h_{431} in terms of k-Schur functions for $k = 6$:

$$\begin{aligned}
h_{431} &= h_1 \left(s^{(6)}_{\begin{ytableau} \ast & \ast & \ast \\ {} & {} & {} & {} \end{ytableau}} + s^{(6)}_{\begin{ytableau} \ast & \ast \\ {} & {} & {} & {} & \ast \end{ytableau}} + s^{(6)}_{\begin{ytableau} \ast \\ {} & {} & {} & {} & \ast & \ast \end{ytableau}} \right) \\
&= \left(s^{(6)}_{\begin{ytableau} \ast \\ {} & {} & {} \\ {} & {} & {} & {} \end{ytableau}} + s^{(6)}_{\begin{ytableau} {} & {} & {} & \ast \\ {} & {} & {} & {} \end{ytableau}} + s^{(6)}_{\begin{ytableau} {} & {} & {} \\ {} & {} & {} & {} & \ast \end{ytableau}} \right) \\
&\quad + \left(s^{(6)}_{\begin{ytableau} \ast \\ {} & {} \\ {} & {} & {} & {} & {} \end{ytableau}} + s^{(6)}_{\begin{ytableau} {} & {} & \ast \\ {} & {} & {} & {} & {} \end{ytableau}} \right) + \left(s^{(6)}_{\begin{ytableau} \ast \\ {} \\ {} & {} & {} & {} & {} & {} \end{ytableau}} + s^{(6)}_{\begin{ytableau} {} & \ast \\ {} & {} & {} & {} & {} & {} \end{ytableau}} \right) .
\end{aligned}$$

Sage Example 2.15. SAGE can be used to verify Example 2.14:

```
sage: ks6 = Sym.kschur(6,t=1)
sage: ks6(h[4,3,1])
ks6[4, 3, 1] + ks6[4, 4] + ks6[5, 2, 1] + 2*ks6[5, 3]
 + ks6[6, 1, 1] + ks6[6, 2]
```

SAGE also knows that the k-Schur functions live in the subring $\Lambda_{(k)}$ of the ring of symmetric functions:

```
sage: Sym = SymmetricFunctions(QQ)
sage: ks = Sym.kschur(3,t=1)
sage: ks.realization_of()
3-bounded Symmetric Functions over Rational Field with t=1
sage: s = Sym.schur()
sage: s.realization_of()
Symmetric Functions over Rational Field
```

When λ and μ are k-bounded partitions, the weak Kostka coefficients $K^{(k)}_{\lambda\mu}$ are interpreted to be the number of sequences of k-bounded partitions,

$$\emptyset = \lambda^{(1)} \subseteq \lambda^{(2)} \subseteq \cdots \subseteq \lambda^{(d)} = \lambda \tag{2.20}$$

where $\lambda^{(i)}/\lambda^{(i-1)}$ is a horizontal strip of size μ_i and $(\lambda^{(i)})^{\omega_k}/(\lambda^{(i-1)})^{\omega_k}$ is a vertical strip. We then follow the line of reasoning from earlier to yield (2.18), where we can instead think of weak tableaux as these sequences of k-bounded shapes.

Example 2.16. The seven tableaux that make up the terms in the expansion of h_{431} in terms of k-Schur functions with $k = 6$ are

$$\begin{ytableau} 3 \\ 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{ytableau} \quad \begin{ytableau} 2 & 2 & 2 & 3 \\ 1 & 1 & 1 & 1 \end{ytableau} \quad \begin{ytableau} 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 3 \end{ytableau} \quad \begin{ytableau} 3 \\ 2 & 2 \\ 1 & 1 & 1 & 1 & 2 \end{ytableau}$$

$$\begin{ytableau} 2 & 2 & 3 \\ 1 & 1 & 1 & 1 & 2 \end{ytableau} \quad \begin{ytableau} 3 \\ 2 \\ 1 & 1 & 1 & 1 & 2 & 2 \end{ytableau} \quad \begin{ytableau} 2 & 3 \\ 1 & 1 & 1 & 1 & 2 & 2 \end{ytableau}\,.$$

Sage Example 2.17. We can reproduce these tableaux in SAGE using the bounded representation of weak k-tableaux:

```
sage: k = 6
sage: weight = Partition([4,3,1])
sage: for la in Partitions(weight.size(), max_part = k):
....:     if la.dominates(weight):
....:         print la
....:         T = WeakTableaux(k, la, weight,
....:             representation = 'bounded')
....:         print T.list()
```

```
....:
[6, 2]
[[[1, 1, 1, 1, 2, 2], [2, 3]]]
[6, 1, 1]
[[[1, 1, 1, 1, 2, 2], [2], [3]]]
[5, 3]
[[[1, 1, 1, 1, 3], [2, 2, 2]], [[1, 1, 1, 1, 2], [2, 2, 3]]]
[5, 2, 1]
[[[1, 1, 1, 1, 2], [2, 2], [3]]]
[4, 4]
[[[1, 1, 1, 1], [2, 2, 2, 3]]]
[4, 3, 1]
[[[1, 1, 1, 1], [2, 2, 2], [3]]]
```

Lastly, let us turn to the language of the affine symmetric group and retrace the steps leading to the dual k-Schur functions. Here, the weak Pieri rule is, for a k-bounded partition λ the corresponding affine Grassmannian element (described in Sect. 1.2) will be $\mathfrak{a}(\mathfrak{c}(\lambda))$ which we will shorten to $\mathfrak{a}(\lambda)$. The weak Pieri rule on affine Grassmannian elements can be stated as

$$h_r\, s_\lambda^{(k)} = \sum_{\substack{u=\text{cyclically decreasing}\\ \ell(u)=r, u\mathfrak{a}(\lambda)=\mathfrak{a}(\mu)}} s_\mu^{(k)}\,, \tag{2.21}$$

where the sum is over cyclically decreasing reduced words such that there is a k-bounded partition μ such that $u\mathfrak{a}(\lambda) = \mathfrak{a}(\mu)$.

The iteration of this relation produces another interpretation of the weak Kostka numbers. First, for any k-bounded partition μ, define a *μ-factorization* of w to be a decomposition of the form $w = w^{\ell(\mu)} \cdots w^1$ where each w^i is a cyclically decreasing element of length μ_i. Then, for k-bounded partitions λ and μ, $K_{\lambda,\mu}^{(k)}$ is the number of μ-factorizations of $w = \mathfrak{a}(\lambda)$. In particular, if we consider the case that $\mu = (1, 1, \ldots, 1)$, then the weak Kostka number $K_{\lambda,(1^n)}^{(k)}$ is precisely the number of reduced words for $w = \mathfrak{a}(\lambda)$.

Example 2.18. Note from the previous example that the coefficient of m_{111111} in $\tilde{F}_{521}^{(k)}$ is 4 indicating that there are 4 reduced words in $\tilde{S}_3$ for the affine Grassmannian element corresponding to the core $(5, 2, 1)$; they are $s_2s_0s_3s_2s_1s_0 = s_0s_2s_3s_2s_1s_0 = s_0s_3s_2s_3s_1s_0 = s_0s_3s_2s_1s_3s_0$.

From here, we again use the line of reasoning and duality to arrive at the interpretation for dual k-Schur functions as

$$\tilde{F}_w^{(k)} = \sum_{\mu} K_{\mathfrak{a}^{-1}(w)\mu}^{(k)}\, m_\mu \tag{2.22}$$

where we have indexed the dual k-Schur function by an affine Grassmannian permutation to emphasize that the coefficients $K_{\mathfrak{a}^{-1}(w)\mu}^{(k)}$ represent μ-factorizations of w. From the definition and symmetry of the weak Kostka numbers, the dual k-Schur functions can be suggestively written as the weight generating function:

$$\tilde{F}_w^{(k)} = \sum_{w=w^1 w^2 \ldots w^r} x^{\ell(w^1)} x^{\ell(w^2)} \cdots x^{\ell(w^r)}, \tag{2.23}$$

over all factorizations of w into products of cyclically decreasing w^i.

This interpretation for dual k-Schur functions is the starting point in [79] to a family of symmetric functions called *affine Stanley symmetric functions* where the condition that w is affine Grassmannian is relaxed and we allow the function to be indexed by arbitrary affine permutations (not just an affine Grassmannian element). For more information see Chap. 3, Definition 3.2.

Programs to compute affine Stanley functions are in SAGE and we can do computations with dual k-Schur functions by taking affine Grassmannian elements.

Sage Example 2.19. We verify Example 2.13 in SAGE by first converting the indexing partition $(3, 2, 1)$ to an affine Grassmannian element:

```
sage: mu = Partition([3,2,1])
sage: c = mu.to_core(3)
sage: w = c.to_grassmannian()
sage: w.stanley_symmetric_function()
4*m[1, 1, 1, 1, 1, 1] + 3*m[2, 1, 1, 1, 1] + 2*m[2, 2, 1, 1]
 + m[2, 2, 2] + 2*m[3, 1, 1, 1] + m[3, 2, 1]
sage: w.reduced_words()
[[2, 0, 3, 2, 1, 0], [0, 2, 3, 2, 1, 0],
 [0, 3, 2, 3, 1, 0], [0, 3, 2, 1, 3, 0]]
```

Alternatively, we can access the dual k-Schur functions from the quotient space:

```
sage: Sym = SymmetricFunctions(QQ)
sage: Q3 = Sym.kBoundedQuotient(3,t=1)
sage: F3 = Q3.affineSchur()
sage: m = Q3.kmonomial()
sage: m(F3([3,2,1]))
4*m3[1, 1, 1, 1, 1, 1] + 3*m3[2, 1, 1, 1, 1]
 + 2*m3[2, 2, 1, 1] + m3[2, 2, 2] + 2*m3[3, 1, 1, 1]
 + m3[3, 2, 1]
```

2.4 Strong Marked Tableaux and a Monomial Expansion of *k*-Schur Functions

The Pieri rule for the dual k-Schur functions $\tilde{F}_\lambda^{(k)}$ is probably less intuitive than the one for the k-Schur functions because it does have coefficients in the expansion which are not simply 1 or 0. In the last section we gave an explicit definition of the dual k-Schur functions in (2.17), so it is possible to experiment with these elements to see how they behave under multiplication by an element $h_r \in \Lambda^{(k)}$. Through the following computations we hope to demonstrate how it might be possible to experiment with data for the Pieri rule for the dual k-Schur functions and then later explain what that Pieri rule is.

Example 2.20. In Example 2.13 we computed the dual k-Schur function indexed by $(3, 2, 1)$ for $k = 3$. Using the tableau definition of the dual k-Schur functions from Eq. (2.17) it is possible to expand $\tilde{F}^{(3)}_{\lambda}$ for all 3-bounded partitions λ of size 7 and use this to find the expansion of $h_1 \tilde{F}^{(3)}_{321}$. If the k-Schur functions up to size 7 are already known, then this also can be computed using the duality. In particular we obtain

$$h_1 \tilde{F}^{(3)}_{321} = 2\tilde{F}^{(3)}_{331} + \tilde{F}^{(3)}_{322} + 3\tilde{F}^{(3)}_{3211} + \tilde{F}^{(3)}_{31111} \,. \tag{2.24}$$

This example demonstrates that if a dual k-Schur function indexed by a partition λ is multiplied by h_1, the resulting partitions indexing the functions in the expansion do not necessarily contain λ (notice that $(3, 2, 1)$ is not contained in $(3, 1, 1, 1, 1)$).

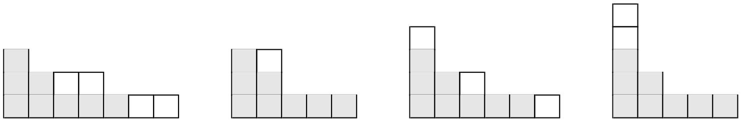
Sage Example 2.21. We now demonstrate how the computations for Example 2.20 can be carried out in SAGE. The dual k-Schur functions can be accessed through the k-bounded quotient space:

```
sage: Sym = SymmetricFunctions(QQ)
sage: Q3 = Sym.kBoundedQuotient(3,t=1)
sage: F3 = Q3.affineSchur()
sage: h = Sym.homogeneous()
sage: f = F3[3,2,1]*h[1]; f
F3[3, 1, 1, 1, 1] + 3*F3[3, 2, 1, 1] + F3[3, 2, 2]
 + 2*F3[3, 3, 1]
```

In terms of 3-bounded partitions, it is not so clear how to interpret the coefficients in the expansion in Example 2.20. However, in terms of 4-cores it becomes more obvious. The 4-core of $(3, 2, 1)$ is contained in the cores of the partitions in the expansion:

```
sage: c = Partition([3,2,1]).to_core(3).to_partition()
sage: for p in f.support():
....:     print p, [p.to_core(3).to_partition(),c]
....:
[3, 1, 1, 1, 1] [[5, 2, 1, 1, 1], [5, 2, 1]]
[3, 2, 1, 1] [[6, 3, 1, 1], [5, 2, 1]]
[3, 2, 2] [[5, 2, 2], [5, 2, 1]]
[3, 3, 1] [[7, 4, 1], [5, 2, 1]]
```

The corresponding skew diagrams are

,

respectively. Comparing with (2.24), one is led to conjecture that the coefficient of $\tilde{F}_\mu^{(k)}$ in the expansion of $h_1\tilde{F}_\lambda^{(k)}$ equals the number of connected components of $\mathfrak{c}(\mu)/\mathfrak{c}(\lambda)$. Alternatively, we may mark the lowest rightmost cell of one of these connected components and the number of ways of marking is equal to this coefficient.

Recall that we defined a strong marked horizontal strip of size r in Eq. (1.10) as a sequence cores

$$\kappa^{(0)} \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(r)}$$

and integers $c_1 < c_2 < \cdots < c_r$ such that $(\kappa^{(i)}, \kappa^{(i-1)}, c_i)$ is a marking for each $1 \leq i \leq r$. It is in these terms that we state the Pieri rule for dual k-Schur functions.

Theorem 2.22 ([81, Theorem 4.9]). *For $r \geq 1$,*

$$h_r\tilde{F}_\lambda^{(k)} = \sum_{(\kappa^{(*)}, c_*)} \tilde{F}^{(k)}_{\mathfrak{p}(\kappa^{(r)})}\,, \tag{2.25}$$

where the sum is over all strong marked horizontal strips

$$\kappa^{(*)} = \left(\mathfrak{c}(\lambda) = \kappa^{(0)} \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(r)}\right) \tag{2.26}$$

with markings $c_* = (c_1 < c_2 < \cdots < c_r)$.

Iterating the Pieri rule on the dual k-Schur functions defines a different notion of 'semi-standard tableaux'. We say that a *strong marked tableau* of shape $\lambda \vdash m$ (or shape $\mathfrak{c}(\lambda)$ if the shape is more properly given as a $(k+1)$-core) and content $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ with $\alpha_1 + \alpha_2 + \cdots + \alpha_d = m$ and $\alpha_i \geq 1$ is a sequence of $(k+1)$-cores

$$\kappa^{(0)} = \emptyset \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(m)} = \mathfrak{c}(\lambda) \tag{2.27}$$

and markings $c_* = (c_1, c_2, \ldots, c_m)$ such that $(\kappa^{(v)}, \kappa^{(v+1)}, \ldots, \kappa^{(v+\alpha_r)})$ and $(c_{v+1}, c_{v+2}, \ldots, c_{v+\alpha_r})$ with $v = \alpha_1 + \cdots + \alpha_{r-1}$ forms a strong marked horizontal strip for each $1 \leq r \leq d$.

Recall from Sect. 1.4 that a strong marked cover is also an application of a transposition t_{ij} in the affine symmetric group to a core (either by the left or right action, see Proposition 1.27). Therefore, it is possible to also view a tableau as an element of the affine Grassmannian written as a sequence of these transpositions.

The condition that a sequence of transpositions $t_{i_{a+b}j_{a+b}} \cdots t_{i_{a+2}j_{a+2}} t_{i_{a+1}j_{a+1}} t_{i_a j_a}$ forms a strong marked horizontal strip (with the left action) implies that $j_a < j_{a+1} < \cdots < j_{a+b}$.

Example 2.23. Consider the path

$$\begin{aligned} \emptyset \Rightarrow_3 (1) \Rightarrow_3 (2) \Rightarrow_3 (2,1) \Rightarrow_3 (3,1,1) \Rightarrow_3 (3,1,1,1) \Rightarrow_3 (3,3,1,1) \\ \Rightarrow_3 (4,3,2,1) \Rightarrow_3 (4,4,2,2). \end{aligned}$$

We choose $\alpha = (2,2,3,1)$ and a sequence $c = (0,1,-1,2,-3,1,3,-2)$ as a sequence of markings which form strong marked horizontal strips of the lengths given by the entries of α to demonstrate the concept of a strong marked tableau. The strong marked tableau of shape $(4,4,2,2)$ records all cells representing strong marked horizontal strips labeled with the same main label but subscripted by which set of ribbons they belong to. The cell which is marked (the head of one of the ribbons) will be indicated by including an $*$ as a superscript. Hence the example above is represented by the diagram:

$$\begin{array}{|c|c|cc} \cline{1-2} 3_1^* & 4_1^* \\ \cline{1-2} 2_2 & 3_3 \\ \hline 2_1^* & 3_2 & \multicolumn{1}{c|}{3_2^*} & \multicolumn{1}{c|}{4_1} \\ \hline 1_1^* & 1_2^* & \multicolumn{1}{c|}{2_2^*} & \multicolumn{1}{c|}{3_3^*} \\ \hline \end{array} \; .$$

This example is also represented by the following sequence of transpositions with the left action

$$t_{-2-1} t_{34} t_{02} t_{-3-2} t_{23} t_{-10} t_{12} t_{01}$$

which act on the empty 4-core. Each transposition adds the strong marking in a horizontal strip according to the tableau.
This same example is also represented using the right action of the transpositions by

$$t_{01} t_{02} t_{-11} t_{05} t_{-21} t_{-12} t_{06} t_{-52} \; .$$

Just as we did for semi-standard and weak tableaux, we set $\mathbf{K}_{\lambda\mu}^{(k)}$ to be the number of strong marked tableaux of shape λ and weight μ. By iterating the Pieri rule on dual k-Schur functions and using the notion of strong marked tableaux to record the terms which appear in the expansion of products of elements of the homogeneous symmetric functions, we have that for a partition $\mu \vdash m$ (not necessarily k-bounded)

$$h_\mu = \sum_{\lambda : \lambda_1 \leq k} \mathbf{K}_{\lambda\mu}^{(k)} \tilde{F}_\lambda^{(k)} \; . \tag{2.28}$$

Then for a partition $\lambda \vdash m$ with $\lambda_1 \leq k$,

$$s_\lambda^{(k)} = \sum_\mu \langle s_\lambda^{(k)}, h_\mu \rangle m_\mu = \sum_\mu \mathbf{K}_{\lambda\mu}^{(k)} m_\mu \,. \tag{2.29}$$

We have hidden all of the work that is needed to show the monomial expansions of k-Schur functions and their duals are correct by this presentation. In fact, the proof [81] follows a nearly reverse path of reasoning to demonstrate the results we have presented here. Equation (2.29) is taken as the definition of the k-Schur functions and (2.17) is the definition of the dual k-Schur functions, and then the argument uses combinatorics and algebraic expressions to show that these elements are dual. The combinatorial part of that argument is developed in Sect. 6.

Example 2.24. Consider the coefficient of m_{421} in $s_{3211}^{(3)}$. It is calculated by finding all strong marked tableaux that begin with a horizontal strip of length 4 and hence contains the subtableau $\begin{array}{llll} 1_4 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* \end{array}$. This is followed by a horizontal strip of length 2 and hence contains one of the five subtableaux:

$$\begin{array}{llllll} 1_4 & 2_1^* & 2_2^* \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1 & 2_2 \end{array}, \quad \begin{array}{llllll} 1_4 & 2_1^* & 2_2 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1 & 2_2^* \end{array}, \quad \begin{array}{llllll} 1_4 & 2_1 & 2_2 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1^* & 2_2^* \end{array}, \quad \begin{array}{lllll} 2_1^* \\ 1_4 & 2_2^* \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2 \end{array}, \quad \begin{array}{lllll} 2_1^* \\ 1_4 & 2_2 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2^* \end{array}.$$

There are 9 strong marked tableaux of shape $\mathfrak{c}(3, 2, 1, 1) = (6, 3, 1, 1)$ of weight $(4, 2, 1)$ which are given by the following

$$\begin{array}{llllll} 3_1 \\ 3_1^* \\ 1_4 & 2_1^* & 2_2^* \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1 & 2_2 \end{array}, \quad \begin{array}{llllll} 3_1 \\ 3_1^* \\ 1_4 & 2_1^* & 2_2 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1 & 2_2^* \end{array}, \quad \begin{array}{llllll} 3_1 \\ 3_1^* \\ 1_4 & 2_1 & 2_2 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_1^* & 2_2^* \end{array}, \quad \begin{array}{llllll} 3_1^* \\ 2_1^* \\ 1_4 & 2_2^* & 3_1 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2 & 3_1 \end{array}, \quad \begin{array}{llllll} 3_1 \\ 2_1^* \\ 1_4 & 2_2^* & 3_1^* \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2 & 3_1 \end{array},$$

$$\begin{array}{llllll} 3_1 \\ 2_1^* \\ 1_4 & 2_2^* & 3_1 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2 & 3_1^* \end{array}, \quad \begin{array}{llllll} 3_1^* \\ 2_1^* \\ 1_4 & 2_2 & 3_1 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2^* & 3_1 \end{array}, \quad \begin{array}{llllll} 3_1 \\ 2_1^* \\ 1_4 & 2_2 & 3_1^* \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2^* & 3_1 \end{array}, \quad \begin{array}{llllll} 3_1 \\ 2_1^* \\ 1_4 & 2_2 & 3_1 \\ 1_1^* & 1_2^* & 1_3^* & 1_4^* & 2_2^* & 3_1^* \end{array}$$

A similar argument can be used to find any coefficient of m_μ for any $\mu \vdash 7$, however there are 210 strong marked tableaux of shape $\mathfrak{c}(3, 2, 1, 1) = (6, 3, 1, 1)$ and weight $(1, 1, 1, 1, 1, 1, 1)$ and hence we will not show the complete computation of $s_{3211}^{(3)}$ using this method.

In later sections we will roughly outline how this formula is proven by generalizing the Robinson–Schensted–Knuth algorithm.

Sage Example 2.25. In this example we show how SAGE can be used to complete calculations from Examples 2.23 and 2.24. The strong tableaux can be entered as a list of entries with markings indicated by a negative number.

```
sage: T = StrongTableau([[-1,-1,-2,-3],[-2,3,-3,4],
....:       [2,3],[-3,-4]], 3)
sage: T.to_transposition_sequence()
[[-2, -1], [3, 4], [0, 2], [-3, -2], [2, 3],
 [-1, 0], [1, 2], [0, 1]]
sage: T.intermediate_shapes()
[[], [2], [3, 1, 1], [4, 3, 2, 1], [4, 4, 2, 2]]
sage: [T.content_of_marked_head(v+1) for v in range(8)]
[0, 1, -1, 2, -3, 1, 3, -2]
sage: T.left_action([0,1])
[[-1, -1, -2, -3, 5], [-2, 3, -3, 4], [2, 3, -5],
 [-3, -4], [5]]
```

The strong tableaux can be listed and by Eq. (2.29) the number of strong tableaux of shape $\mathfrak{c}(\lambda)$ and content μ can be calculated by determining the coefficient of m_μ in $s_\lambda^{(k)}$.

```
sage: ST = StrongTableaux(3, [6,3,1,1], [4,2,1]); ST
Set of strong 3-tableaux of shape [6, 3, 1, 1] and
  of weight (4, 2, 1)
sage: ST.list()
[[[-1, -1, -1, -1, 2, 2], [1, -2, -2], [-3], [3]],
 [[-1, -1, -1, -1, 2, -2], [1, -2, 2], [-3], [3]],
 [[-1, -1, -1, -1, -2, -2], [1, 2, 2], [-3], [3]],
 [[-1, -1, -1, -1, 2, 3], [1, -2, 3], [-2], [-3]],
 [[-1, -1, -1, -1, 2, 3], [1, -2, -3], [-2], [3]],
 [[-1, -1, -1, -1, 2, -3], [1, -2, 3], [-2], [3]],
 [[-1, -1, -1, -1, -2, 3], [1, 2, 3], [-2], [-3]],
 [[-1, -1, -1, -1, -2, 3], [1, 2, -3], [-2], [3]],
 [[-1, -1, -1, -1, -2, -3], [1, 2, 3], [-2], [3]]]
sage: ks = SymmetricFunctions(QQ).kschur(3,1)
sage: m = SymmetricFunctions(QQ).m()
sage: m(ks[3,2,1,1]).coefficient([4,2,1])
9
```

Remark 2.26. We have been purposely lax in our notation to make some of the concepts slightly easier to follow, but the duality creates a few issues with the element which represents $\tilde{F}_\lambda^{(k)}$ in $\Lambda^{(k)}$. The equalities in Eqs. (2.25) and (2.28) represent equality in the realization of the dual algebra $\Lambda^{(k)} = \Lambda/\langle m_\lambda : \lambda_1 > k\rangle$, so the equality really means equivalence in the quotient algebra. For computational purposes, we would typically take a representative element from the linear span of $\{m_\lambda\}_{\lambda_1 \le k}$, but for certain purposes a multiplicative basis might be more desirable. In those cases, the basis $\{p_\lambda\}_{\lambda_1 \le k}$ works well since $p_r \in \langle m_\lambda : \lambda_1 > k\rangle$ for $r > k$. A combinatorial formula for $\tilde{F}_\lambda^{(k)}$ in terms of the power sum basis is also known by Ref. [7].

2.5 *k*-Littlewood–Richardson Coefficients

Although the original definition [91] of k-Schur functions was inspired to explain the positivity of the expansion of Macdonald symmetric functions in terms of Schur functions, it has since been established that the theory of k-Schur functions and their duals can be naturally applied to study problems in geometry, physics and representation theory.

The application of k-Schur functions to geometric problems began when Lapointe and Morse discovered that their structure constants could be identified with certain geometric invariants in a way that mimics the identification of Littlewood–Richardson coefficients with Schubert structure constants in the Grassmannian variety (recall from Sect. 1.6); computation in the quantum cohomology of Grassmannians reduces to k-Schur calculations. The (small) quantum cohomology ring $QH^*(\mathrm{Gr}_{\ell n})$ is a deformation of the classical cohomology ring that is motivated by ideas in string theory (e.g. [1, 156]). As abelian groups, $QH^*(\mathrm{Gr}_{\ell n}) = H^*(\mathrm{Gr}_{\ell n}) \otimes \mathbb{Z}[q]$ and the Schubert classes σ_λ with $\lambda \in \mathcal{P}^{\ell n}$ form a $\mathbb{Z}[q]$-linear basis, where recall that $\mathcal{P}^{\ell n}$ is the set of all partitions in an $\ell \times (n-\ell)$ rectangle. The appeal lies in the multiplicative structure which is defined by

$$\sigma_\lambda * \sigma_\mu = \sum_{\substack{\nu \in \mathcal{P}^{\ell n} \\ |\nu| = |\lambda| + |\mu| - dn}} q^d C_{\lambda\mu}^{\nu,d} \sigma_\nu \,,$$

where $C_{\lambda\mu}^{\nu,d}$ are the *3-point Gromov–Witten invariants*, counting the number of rational curves of degree d in $\mathrm{Gr}_{\ell n}$ that meet generic translates of certain Schubert varieties.

As with the usual cohomology, the quantum cohomology ring can be connected to symmetric functions. In particular,

$$QH^*(\mathrm{Gr}_{\ell n}) \cong \left(\Lambda_{(\ell)} \otimes \mathbb{Z}[q]\right) / J_q^{\ell n} \,,$$

where $J_q^{\ell n} = \langle e_{n-\ell+1}, \ldots, e_{n-1}, e_n + (-1)^\ell q \rangle$. When $\lambda \in \mathcal{P}^{\ell n}$, the Schubert class σ_λ still maps to the Schur function s_λ implying that for $\lambda, \mu \in \mathcal{P}^{\ell n}$,

$$\sum_\nu c_{\lambda\mu}^\nu s_\nu \mod J_q^{\ell n} = \sum_{\substack{\nu \in \mathcal{P}^{\ell n} \\ |\nu| = |\lambda| + |\mu| - dn}} q^d C_{\lambda\mu}^{\nu,d} s_\nu \,. \tag{2.30}$$

Unfortunately, there are $\nu \notin \mathcal{P}^{\ell n}$ where s_ν does not lie in $J_q^{\ell n}$. Instead, reduction modulo this ideal requires a complicated algorithm [16, 34, 67, 154] involving negatives. Therefore, the Schur functions cannot be used to directly obtain the quantum structure constants.

It was proven in [93] that the k-Schur basis can be used to circumvent this problem; the appropriate k-Schur functions lie in $J_q^{\ell n}$ (as usual, we set $n = k + 1$).

To be precise, the *k-Littlewood–Richardson coefficients* are the structure coefficients $c_{\lambda\mu}^{\nu(k)}$ of the algebra of k-Schur functions

$$s_\lambda^{(k)} s_\mu^{(k)} = \sum_\nu c_{\lambda\mu}^{\nu(k)} s_\nu^{(k)} . \tag{2.31}$$

Let $\Pi^{\ell,k+1}$ be the set of partitions with no part larger than ℓ and no more than $k+1-\ell$ rows of length smaller than ℓ. It is proven that $s_\nu^{(k)}$ modulo $J_q^{\ell n}$ reduces to a power of q times a positive $s_{\mathfrak{r}(\nu)}$ when $\nu \in \Pi^{\ell n}$ and is otherwise zero (where $\mathfrak{r}(\nu) \in \mathcal{P}^{\ell n}$ is the n-core of ν). Thus, considering

$$s_\lambda^{(k)} s_\mu^{(k)} = \sum_{\nu \in \Pi^{\ell,k+1}} a_{\lambda\mu}^{\nu(k)} s_\nu^{(k)} + \sum_{\nu \notin \Pi^{\ell,k+1}} c_{\lambda\mu}^{\nu,k} s_\nu^{(k)} , \tag{2.32}$$

the 3-point Gromov–Witten invariants are none other than a special case of k-Littlewood–Richardson coefficients; for $\lambda, \mu, \nu \in \mathcal{P}^{\ell n}$,

$$C_{\lambda\mu}^{\nu,d} = a_{\lambda\mu}^{\hat{\nu}(n-1)} , \tag{2.33}$$

where the value of d associates a certain unique element $\hat{\nu} \in \Pi^{\ell n}$ to each ν. It also follows that the k-Littlewood–Richardson coefficients include the fusion rules for the Wess–Zumino–Witten conformal field theories associated to $\widehat{su}(\ell)$ at level $k+1-\ell$ and certain Hecke algebra structure constants studied by Goodman and Wenzl in [56].

Note that the quantum structure constants (2.30) are only a subset of the complete set of k-Littlewood–Richardson coefficients (2.32). To understand the bigger picture, recall that the k-Schur functions are a basis for $\Lambda_{(k)}$ and that $\Lambda_{(k)} \cong H_*(\mathrm{Gr})$ where Gr is the *affine Grassmannian* quotient $\mathrm{Gr} = SL_{k+1}(\mathbb{C}((t)))/SL_{k+1}(\mathbb{C}[[t]])$ [22]. Bott also showed that $H^*(\mathrm{Gr}) \cong \Lambda^{(k)}$, the space equipped with the dual k-Schur basis. Morse and Shimozono conjectured that the k-Schur functions and their duals are isomorphic to the Schubert classes of the homology and cohomology of the affine Grassmannian. Chapter 3 explains how Lam proved this conjecture.

Many attempts have been made to understand the coefficients $c_{\lambda\mu}^{\nu(k)}$ but the complete combinatorial picture has yet to be drawn. Knutson formulated a conjecture for the subset of quantum Littlewood–Richardson coefficients as presented in [29] in terms of puzzles [74]. Coskun [33] gave a positive geometric rule to compute the structure constants of the cohomology ring of two-step flag varieties in terms of Mondrian tableaux. As will be discussed in Sect. 4.6, the case when either λ or μ is a rectangle with a hook of size k is special. Lapointe and Morse [92] showed that if R is a rectangular partition with maximal hook k, then

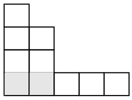
$$s_R^{(k)} s_\lambda^{(k)} = s_{R\cup\lambda}^{(k)} .$$

There is a result due to Denton [38] which partially explains why the product of a rectangle with maximal hook has one term for a more general reason. He noticed that if the cells with a k-bounded hook in the $k+1$-core $\mathfrak{c}(\lambda)$ are not connected with a taxicab path between the rth and $r+1$st row, then

$$s^{(k)}_{\lambda} = s^{(k)}_{(\lambda_1,\ldots,\lambda_r)} \cdot s^{(k)}_{(\lambda_{r+1},\ldots,\lambda_\ell)} \, .$$

If the bounded partition properly contains a maximal rectangle R_i then $\mathfrak{c}(\lambda)$ will have one of these disconnected rows, but there are examples where the k bounded cells of $\mathfrak{c}(\lambda)$ may be disconnected but the partition does not contain a maximal rectangle.

Example 2.27. To illustrate this we may draw the 5-core corresponding to the 4-bounded partition $(3, 2, 2, 1)$ and shade the cells which are not 4-bounded.



Denton's result then explains why $s^{(4)}_{(3)} \cdot s^{(4)}_{(2,2,1)} = s^{(4)}_{(3,2,2,1)}$.

In [124, 125], Morse and Schilling define crystal operators on α-factorizations (or equivalently weak k-tableaux) to determine some structure coefficients of the Schur function times k-Schur function expansion using a sign-reversing involution. This includes the case of fusion coefficients.

Recently there has been some progress in the understanding of these coefficients by viewing them in terms of the nil-Coxeter algebra. Slightly change the setting from the affine symmetric group to the affine nil-Coxeter algebra $\mathbb{A}_n$ of type A_{n-1} defined by the generators $A_0, A_1, A_2, \ldots, A_{n-1}$ satisfying the same relations as the affine symmetric group Eq. (1.4) except that the quadratic relation is altered to be

$$A_i^2 = 0 \, ,$$

for any $i \in I := \{0, 1, 2, \ldots, n-1\}$.

As before, $n = k + 1$. For $1 \leq r \leq k$, we set

$$\tilde{\mathbf{h}}_r = \sum_{\substack{w \text{ cyclically decreasing} \\ \ell(w)=r}} A_w \, .$$

Note that the coefficient of x^α in (2.23) is the coefficient of A_w in $\tilde{\mathbf{h}}_{\alpha_1}\tilde{\mathbf{h}}_{\alpha_2}\cdots\tilde{\mathbf{h}}_{\alpha_\ell}$ and thus, the affine Stanley symmetric function is

$$\tilde{F}_w = \sum_{\alpha} (\text{coefficient of } A_w \in \tilde{\mathbf{h}}_{\alpha_1}\tilde{\mathbf{h}}_{\alpha_2}\cdots\tilde{\mathbf{h}}_{\alpha_\ell}) \; x^\alpha \, .$$

As is stated in Theorem 8.6 of Chap. 3, the elements $\tilde{\mathbf{h}}_r$ mutually commute and there is a subalgebra $\mathbb{B}_{\mathrm{af}} \subseteq \mathbb{A}_n$ which is generated by the elements $\tilde{\mathbf{h}}_r$ for $1 \leq r \leq k$. Moreover, $\mathbb{B}_{\mathrm{af}}$ is isomorphic to $\Lambda_{(k)}$ (see Proposition 8.8 of Chap. 3) and there is a coalgebra structure which encodes the structure of the dual algebra. The *noncommutative k-Schur functions* can be realized as elements $\mathbf{s}_\lambda^{(k)} \in \mathbb{B}_{\mathrm{af}}$ in this algebra by expanding $s_\lambda^{(k)}$ in the homogeneous generators h_i and replacing h_i with $\tilde{\mathbf{h}}_i$.

Let $a_{\lambda w}$ be the coefficients in the expansion of $\mathbf{s}_\lambda^{(k)}$ in terms of expressions in the generators A_i in the affine nil-Coxeter algebra

$$\mathbf{s}_\lambda^{(k)} = \sum_w a_{\lambda w} A_w \,. \tag{2.34}$$

The expansion of $\mathbf{s}_\lambda^{(k)}$ has a single term indexed by an affine Grassmannian permutation, it is precisely the affine Grassmannian permutation which corresponds to the partition λ and the coefficient $a_{\lambda w_\lambda} = 1$ where $\mathfrak{a}(\lambda) = w_\lambda$.

By [79, Proposition 6.7], the coefficient $a_{\lambda w}$ is the coefficient of $\tilde{F}_\lambda$ in the element $\tilde{F}_w$. In [80], these coefficients were shown to be positive. Furthermore, by the arguments in [79, Sect. 17] it follows that

$$c_{\lambda\mu}^{\nu(k)} = a_{\lambda v w^{-1}}, \tag{2.35}$$

where $w = \mathfrak{a}(\mu)$ and $v = \mathfrak{a}(\nu)$, see Proposition 1.9.

Berg, Bergeron, Thomas, and Zabrocki [11] gave a combinatorial expansion of k-Schur functions indexed by a rectangle with maximal hook k in the nil-Coxeter algebra of the form of Eq. (2.34). This work was extended by Berg, Saliola, and Serrano [14] to give expansions of k-Schur functions when the indexing partition is a 'maximal rectangle' with a smaller hook removed. In addition they proved some conjectures of [81] in [13], in particular that "skew shaped" strong Schur functions are symmetric.

Sage Example 2.28. We now show how to compute the noncommutative Schur functions in the affine nil-Coxeter algebra $\mathbb{A}_n$. We can construct all cyclically decreasing words from the reduced words of the Pieri factors for the affine type A_{n-1} where $n = k + 1$.

```
sage: W = WeylGroup(["A",3,1])
sage: [w.reduced_word() for w in W.pieri_factors()]
[[], [0], [1], [2], [3], [1, 0], [2, 0], [0, 3], [2, 1],
 [3, 1], [3, 2], [2, 1, 0], [1, 0, 3], [0, 3, 2], [3, 2, 1]]
```

Then the noncommutative homogeneous symmetric functions are given by summing over all cyclically decreasing words of specified length:

```
sage: A = NilCoxeterAlgebra(WeylGroup(["A",3,1]),prefix = 'A')
sage: A.homogeneous_noncommutative_variables([2])
A[1,0] + A[2,0] + A[0,3] + A[3,2] + A[3,1] + A[2,1]
```

The noncommutative k-Schur functions are obtained by expanding the usual k-Schur functions in terms of the homogeneous symmetric function:

```
sage: A.k_schur_noncommutative_variables([2,2])
A[0,3,1,0] + A[3,1,2,0] + A[1,2,0,1] + A[3,2,0,3] + A[2,0,3,1]
 + A[2,3,1,2]
```

Now let us test that $a_{\lambda w}$ is indeed related to $c_{\lambda\mu}^{\nu(k)}$. The structure coefficient $c_{21,21}^{321(5)} = 2$ as we can see from the following computation:

```
sage: Sym = SymmetricFunctions(ZZ)
sage: ks = Sym.kschur(5,t=1)
sage: ks[2,1]*ks[2,1]
ks5[2, 2, 1, 1] + ks5[2, 2, 2] + ks5[3, 1, 1, 1]
 + 2*ks5[3, 2, 1] + ks5[3, 3] + ks5[4, 2]
```

Let v (resp. w) be the affine Grassmannian element corresponding to the 5-bounded partition $(3,2,1)$ (resp. $(2,1)$). Then by (2.35) the coefficient of $A_{vw^{-1}}$ in $\mathbf{s}_{21}^{(5)}$ should also be 2:

```
sage: mu = Partition([2,1])
sage: nu = Partition([3,2,1])
sage: w = mu.from_kbounded_to_grassmannian(5)
sage: v = nu.from_kbounded_to_grassmannian(5)
sage: A = NilCoxeterAlgebra(WeylGroup(["A",5,1]),prefix = 'A')
sage: ks = A.k_schur_noncommutative_variables([2,1])
sage: ks.coefficient(v*w^(-1))
2
```

2.6 Notes on References

Note that in certain references (e.g. [92, 94, 95]) the notation for $\Lambda_{(k)}$ and $\Lambda^{(k)}$ are switched. We chose our convention to be consistent with [80], where the notation is motivated from the relation between these spaces and the homology/cohomology of the affine Grassmannian.

In Ref. [94, Conjecture 21] it is stated that the k-Schur functions defined using an algebraic definition satisfy the Pieri rule of Eq. (2.19). Later Refs. [92, 96] do use Eq. (2.19) as the definition and prove properties from this point of view. The definition used in [94, 95] will be presented in the next section and is conjecturally equivalent to Eq. (2.19) at $t = 1$.

The proof of Theorem 2.22 is cited here as due to the affine insertion algorithm that was studied by Lam, Lapointe, Morse and Shimozono [81]. This algorithm will be discussed further in Sect. 6.3. A bijective algorithm is used to show that there is a duality between strong and weak orders on affine Grassmannians that is manifest in the Pieri rules for k-Schur functions and their duals.

When the details of affine Stanley symmetric functions are carried out in Chap. 3, there is a minor difference of notation; the affine nil-Coxeter algebra is denoted $\mathbb{A}_n$ in Sect. 2.5 and as $\mathbb{A}_{\text{af}}$ in Chap. 3. In [132], Postnikov used the affine nil-Temperley–Lieb algebra, which is a quotient of the nil-Coxeter algebra, to provide a model for the quantum cohomology.

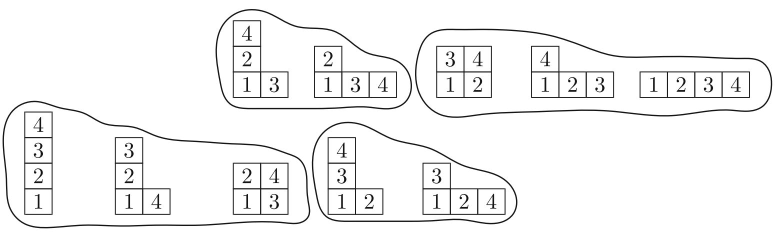
3 Definitions of k-Schur Functions

In the previous section we defined k-Schur functions and their dual basis using the notions of strong and weak tableaux, but these are parameterless symmetric functions. In this section we will provide the original definition of k-Schur functions as well as several others that are conjecturally equivalent.

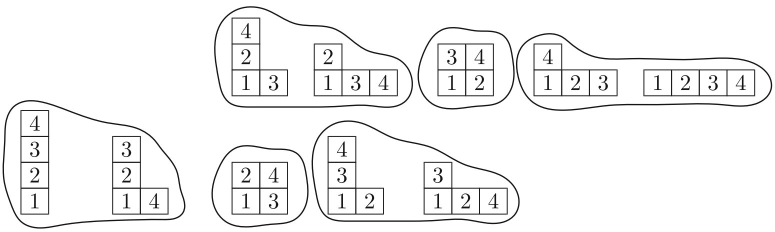
3.1 Atoms as Tableaux

The origin of the tableaux definition (1.40) comes from identifying the k-Schur functions on the poset of standard tableaux ranked by the charge statistic. It is known that there is one Schur function in the expansion of the Macdonald symmetric functions for each standard tableau, so given that the k-Schur functions are these irreducible components of the Macdonald symmetric functions, it is natural to try to connect them with standard tableaux. These conjecturally symmetric functions were first given the name 'atoms' because they are the indecomposable pieces that come together to give Macdonald symmetric functions. They can be determined experimentally and from that point it is possible to conjecture beautiful properties that they possess.

Example 3.1. From Example 1.45 we 'know' that $A_{22}^{(2)}[X;t] = s_{22} + ts_{31} + t^2 s_4$, $A_{211}^{(2)}[X;t] = s_{211} + ts_{31}$, $A_{1111}^{(2)}[X;t] = s_{1111} + ts_{211} + t^2 s_{22}$. The following picture contains the standard tableaux of size 4 graded by charge. The tableau on the left has charge 0, the one on the right has charge 6 and otherwise the left to right positions depends on the value of the charge. On this diagram we have circled the groups of these tableaux which represent the atoms:



Moreover using the same method as in Example 1.45 to compute the 3-Schur functions, we obtain $s^{(3)}_{1111}[X;t] = s_{1111} + ts_{211}$, $s^{(3)}_{211} = s_{211} + ts_{31}$, $s^{(3)}_{22} = s_{22}$, $s^{(3)}_{31} = s_{31} + ts_4$. We then picture these symmetric functions as groups of tableaux on the tableaux poset ranked by charge by placing a circle around the 'copies' of the 3-Schur functions:



Notice in this example that the atoms of level $k = 3$ are contained in those of level 2. We will see later that it is a property of k-Schur functions that the ones of level k always expand positively in terms of k-Schur functions of level $k + 1$.

At $k = 4$ the k-Schur function $s^{(4)}_{\lambda} = s_{\lambda}$ for $\lambda \vdash 4$. Hence if we were to redraw the picture and circle the pieces representing the k-Schur functions, then each tableau would be in its own circle. This observation shows that moving from level 2 to 3 to 4 is that the circles representing the k-Schur functions break into smaller and smaller pieces.

The atoms can be computed by means of a recursive combinatorial procedure. We will describe each of the steps as operations on tableaux. The operators will act on a single tableau or a set of tableaux (depending on what is appropriate) and return a set of tableaux.

First we define σ_i as the operator which takes a tableau with a cells labeled with i and b cells labeled with $i + 1$ and changes it into a tableau with b cells labelled by i and a labelled with $i + 1$. This is done by considering the reading word and placing a closed parenthesis ")" under each letter in the word labelled with an i and an open parenthesis "(" under each letter in the word labelled by an $i + 1$. The word naturally has cells which have matching open and closed parentheses and the remainder of the parentheses are 'free.' Change the free open or closed parentheses and their corresponding labels so that in the end there are b labels with i and a labels of $i + 1$. The result of this operation is the tableau of the same shape, but with this new reading word. This operation is also known as the 'reflection along an i-string' in a crystal graph, see for example [104].

Example 3.2. Consider the action of σ_3 on the semi-standard tableau

$$\begin{array}{ccccccccc} 4 & 5 & 5 \\ 3 & 3 & 4 & 6 & 6 \\ 2 & 2 & 3 & 3 & 3 & 5 & 6 \\ 1 & 1 & 1 & 1 & 2 & 2 & 3 & 3 & 4 \end{array}.$$

Because there are four more 3s in the word than there are 4s, we should change four of the label 3s to the label 4. Reading the entries in this tableau that are 3 or 4 only, the word is 4334333334 which, replacing 4 with (and 3 with), corresponds to the word of parentheses ())()))))(. If we ignore all the parentheses which match up and change the rightmost four unmatched closed parentheses to open parentheses, this corresponds to the word ())()(((((→ 4334344444. Now this sequence of labels is placed back in the tableau in the positions of the labels of 3 and 4, and the resulting tableau is

$$\begin{array}{ccccccccc} 4 & 5 & 5 \\ 3 & 3 & 4 & 6 & 6 \\ 2 & 2 & 3 & 4 & 4 & 5 & 6 \\ 1 & 1 & 1 & 1 & 2 & 2 & 4 & 4 & 4 \end{array}.$$

The next operation is an analogue of the Pieri rule on tableaux. Define an operation $\mathbb{B}_r$ for $r \geq 1$ which adds a horizontal strip of size r to the tableau in all possible ways and then acts by operators σ_i to change the weight so that there are r labels of 1. That is, if we take a tableau T of shape $\lambda \vdash m$,

$$\mathbb{B}_r(T) = \{\sigma_1\sigma_2\cdots\sigma_{\ell(\mu)}(T \subseteq \mu) : \mu/\lambda \text{ is a horizontal strip of size } r\}$$

where $T \subseteq \mu$ represents the semi-standard tableau T as a sequence of partitions with an additional partition μ attached.

Example 3.3. Consider the action of $\mathbb{B}_3$ on the tableau $\begin{array}{cc} 3 \\ 2 \\ 1 & 1 \end{array}$. This is equal to

$$\mathbb{B}_3\left(\begin{array}{cc} 3 \\ 2 \\ 1 & 1 \end{array}\right) = \sigma_1\sigma_2\sigma_3\left\{\begin{array}{ccc} 4 \\ 3 \\ 2 & 4 \\ 1 & 1 & 4 \end{array}, \begin{array}{cccc} 4 \\ 3 \\ 2 \\ 1 & 1 & 4 & 4 \end{array}, \begin{array}{cccc} 3 \\ 2 & 4 \\ 1 & 1 & 4 & 4 \end{array}, \begin{array}{ccccc} 3 \\ 2 \\ 1 & 1 & 4 & 4 & 4 \end{array}\right\}$$

$$= \sigma_1\sigma_2\left\{\begin{array}{ccc} 4 \\ 3 \\ 2 & 3 \\ 1 & 1 & 3 \end{array}, \begin{array}{cccc} 4 \\ 3 \\ 2 \\ 1 & 1 & 3 & 3 \end{array}, \begin{array}{cccc} 3 \\ 2 & 3 \\ 1 & 1 & 3 & 4 \end{array}, \begin{array}{ccccc} 3 \\ 2 \\ 1 & 1 & 3 & 3 & 4 \end{array}\right\}$$

$$= \sigma_1 \left\{ \begin{array}{lll} 4 \\ 3 \\ 2 & 2 \\ 1 & 1 & 2 \end{array}, \begin{array}{llll} 4 \\ 3 \\ 2 \\ 1 & 1 & 2 & 2 \end{array}, \begin{array}{llll} 3 \\ 2 & 2 \\ 1 & 1 & 2 & 4 \end{array}, \begin{array}{lllll} 3 \\ 2 \\ 1 & 1 & 2 & 2 & 4 \end{array} \right\}$$

$$= \left\{ \begin{array}{lll} 4 \\ 3 \\ 2 & 2 \\ 1 & 1 & 1 \end{array}, \begin{array}{llll} 4 \\ 3 \\ 2 \\ 1 & 1 & 1 & 2 \end{array}, \begin{array}{llll} 3 \\ 2 & 2 \\ 1 & 1 & 1 & 4 \end{array}, \begin{array}{lllll} 3 \\ 2 \\ 1 & 1 & 1 & 2 & 4 \end{array} \right\}.$$

The last notion that we need is that of a *katabolizable tableau*. To introduce this definition we need the notion of jeu de taquin, Knuth equivalence or the Robinson–Schensted–Knuth algorithm. We introduce a generalization of the Robinson–Schensted–Knuth algorithm in Sect. 6. We assume here that the reader is familiar with this notion (and if not then one can skip ahead a few sections or consult [139] for example).

Let T be a tableau with reading word w. The definition that T is katabolizable with respect to a sequence of partitions $\lambda^{(*)} = (\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(r)})$ is recursive. It is required that $r = \ell(\lambda^{(1)})$. Then let $w = uv$, where u is the largest subword of w that does not contain an r. Let v' be v with all letters 1 through r deleted. We say that T is *katabolizable* with respect to $\lambda^{(*)}$ if T contains as a subtableau the semi-standard tableau of shape $\lambda^{(1)}$ and weight $\lambda^{(1)}$ and if $\mathrm{RSK}(v'u) = (P, Q)$ where P is a tableau which is $(\lambda^{(2)}, \ldots, \lambda^{(r)})$ katabolizable with the labels shifted by r.

Example 3.4. Consider the tableau

$$T = \begin{array}{llll} 3 \\ 2 & 2 & 3 \\ 1 & 1 & 1 & 4 \end{array}$$

which is katabolizable with respect to the sequence of partitions $((3, 2), (2, 1))$. This is because $r = \ell(\lambda^{(1)}) = 2$. Furthermore, the reading word of T is $w = 32231114 = uv$, where $u = 3$ and $v = 2231114$. Then $v' = 34$ and the P tableau of $\mathrm{RSK}(v'u)$ is $\begin{array}{ll} 4 \\ 3 & 3 \end{array}$.

Now T is not katabolizable with respect to the sequence $((3), (2, 2), (1))$. The reason is that in this case $u = 3223$ and $v = 1114$, so that $v' = 4$. Then the P tableau of $\mathrm{RSK}(v'u)$ is $\begin{array}{lll} 4 \\ 3 \\ 2 & 2 & 3 \end{array}$ and this tableau does not contain $\begin{array}{ll} 3 & 3 \\ 2 & 2 \end{array}$ and hence it is not katabolizable with respect to the sequence $((2, 2), (1))$.

Recall that we introduced the notion of the k-split of a partition in (1.2). Now define an operator on tableaux $\mathbb{K}^{\to k}$ that acts on a set of tableaux with partition weight λ such that $\mathbb{K}^{\to k}$ kills all tableaux that are not katabolizable with respect to the k-split of λ and keeps the ones that are katabolizable.

Example 3.5. Consider the action of $\mathbb{K}^{\to 3}$ on the following set of semi-standard tableaux with weight $(2,1,1,1,1)$. Since $(2,1,1,1,1)^{\to 3} = ((2,1),(1,1,1))$, a tableau will only survive if it contains $\begin{ytableau} 2 \\ 1 & 1 \end{ytableau}$ as a subtableau.

$$\mathbb{K}^{\to k}\left\{ \begin{ytableau} 3 \\ 2 & 4 \\ 1 & 1 & 5 \end{ytableau}, \begin{ytableau} 3 \\ 2 & 5 \\ 1 & 1 & 4 \end{ytableau}, \begin{ytableau} 4 \\ 2 & 3 \\ 1 & 1 & 5 \end{ytableau}, \begin{ytableau} 4 \\ 2 & 5 \\ 1 & 1 & 3 \end{ytableau}, \begin{ytableau} 5 \\ 2 & 3 \\ 1 & 1 & 4 \end{ytableau}, \begin{ytableau} 5 \\ 2 & 4 \\ 1 & 1 & 3 \end{ytableau} \right\} = \left\{ \begin{ytableau} 3 \\ 2 & 5 \\ 1 & 1 & 4 \end{ytableau} \right\}.$$

Given a k-bounded partition λ, we define the k-atom $\mathbb{A}_\lambda^{(k)}$ as a set of tableaux which are computed recursively as

$$\mathbb{A}_\lambda^{(k)} = \mathbb{K}^{\to k}\mathbb{B}_{\lambda_1}\mathbb{A}^{(k)}_{(\lambda_2,\lambda_3,\ldots,\lambda_{\ell(\lambda)})}\,. \tag{3.1}$$

Example 3.6. We can compute the atom $\mathbb{A}_{11}^{(3)} = \left\{ \begin{ytableau} 2 \\ 1 \end{ytableau} \right\}$ and $\mathbb{B}_2(\mathbb{A}_{11}^{(3)}) = \left\{ \begin{ytableau} 3 \\ 2 \\ 1 & 1 \end{ytableau}, \begin{ytableau} 2 \\ 1 & 1 & 3 \end{ytableau} \right\}$. The atom $\mathbb{A}_{211}^{(3)} = \left\{ \begin{ytableau} 3 \\ 2 \\ 1 & 1 \end{ytableau}, \begin{ytableau} 2 \\ 1 & 1 & 3 \end{ytableau} \right\}$ because each of these tableaux survives the operator $\mathbb{K}^{\to 3}$. As we have seen in Example 3.3, we know what the action of $\mathbb{B}_3$ on the first tableau is, and there are six additional tableaux when $\mathbb{B}_3$ acts on $\begin{ytableau} 2 \\ 1 & 1 & 3 \end{ytableau}$ so that

$$\mathbb{B}_3(\mathbb{A}_{211}^{(3)}) = \left\{ \begin{ytableau} 4 \\ 3 \\ 2 & 2 \\ 1 & 1 & 1 \end{ytableau}, \begin{ytableau} 4 \\ 3 \\ 2 \\ 1 & 1 & 1 & 2 \end{ytableau}, \begin{ytableau} 3 \\ 2 & 2 \\ 1 & 1 & 1 & 4 \end{ytableau}, \begin{ytableau} 3 \\ 2 \\ 1 & 1 & 1 & 2 & 4 \end{ytableau}, \begin{ytableau} 3 \\ 2 & 2 & 4 \\ 1 & 1 & 1 \end{ytableau}, \begin{ytableau} 3 \\ 2 & 4 \\ 1 & 1 & 1 & 2 \end{ytableau}, \begin{ytableau} 4 \\ 2 \\ 1 & 1 & 1 & 2 & 3 \end{ytableau}, \begin{ytableau} 2 & 2 & 4 \\ 1 & 1 & 1 & 3 \end{ytableau}, \begin{ytableau} 2 & 4 \\ 1 & 1 & 1 & 2 & 3 \end{ytableau}, \begin{ytableau} 2 \\ 1 & 1 & 1 & 2 & 3 & 4 \end{ytableau} \right\} \tag{3.2}$$

It is an unusual situation, but all 10 of these tableaux are katabolizable with respect to the sequence $(3,2,1,1)^{\to 3} = ((3),(2,1),(1))$ and hence survive the operator $\mathbb{K}^{\to 3}$.

If on the other hand, we do the computation with $k = 4$, we find

$$\mathbb{A}_{221}^{(4)} = \left\{ \begin{ytableau} 3 \\ 2 \\ 1 & 1 \end{ytableau} \right\} \quad \text{and} \quad \mathbb{A}_{3211}^{(4)} = \left\{ \begin{ytableau} 4 \\ 3 \\ 2 & 2 \\ 1 & 1 & 1 \end{ytableau}\,, \quad \begin{ytableau} 3 \\ 2 & 2 \\ 1 & 1 & 1 & 4 \end{ytableau} \right\} . \tag{3.3}$$

We can check (3.3) in SAGE via

```
sage: la = Partition([3,2,1,1])
sage: la.k_atom(4)
[[[1, 1, 1], [2, 2], [3], [4]], [[1, 1, 1, 4], [2, 2], [3]]]
```

Now we define the symmetric function $A_\lambda^{(k)}[X;t]$ in terms of the set of tableaux $\mathbb{A}_\lambda^{(k)}$ as

$$A_\lambda^{(k)}[X;t] = \sum_{T \in \mathbb{A}_\lambda^{(k)}} t^{\mathrm{charge}(T)} s_{\mathrm{shape}(T)} . \tag{3.4}$$

Example 3.7. The charge of the tableau of shape $(3,2,1,1)$ in (3.3) is 0, whereas the charge of the tableau of shape $(4,2,1)$ is 1. Hence we obtain

$$A_{3211}^{(4)}[X;t] = s_{3211} + t s_{421} .$$

The element $A_{3211}^{(3)}[X;t]$ is a generating function for the tableaux in (3.2) with a term of the form t raised to the charge times the Schur function indexed by the shape. By computing the charge of each of these tableaux we determine

$$A_{3211}^{(3)}[X;t] = s_{3211} + t s_{4111} + (t+t^2) s_{421} + t s_{331} + (t^2+t^3) s_{511} + t^2 s_{43} + t^3 s_{52} + t^4 s_{61} .$$

The atoms by this definition are conjectured to be a basis of the space

$$\begin{aligned} \Lambda_{(k)}^t &= \mathcal{L}_{\mathbb{Q}(q,t)}\{H_\lambda[X;q,t] : \lambda_1 \le k\} \\ &= \mathcal{L}_{\mathbb{Q}(q,t)}\{s_\lambda\left[\frac{X}{1-t}\right] : \lambda_1 \le k\} \\ &= \mathcal{L}_{\mathbb{Q}(q,t)}\{Q'_\lambda[X;t] : \lambda_1 \le k\} . \end{aligned} \tag{3.5}$$

However, no proof that these functions are even elements of this space currently exists.

3.2 A Symmetric Function Operator Definition

In this section we present a conjecturally equivalent definition for k-Schur functions that is in similar spirit to the atom description given in the previous section, but now with an algebraic flavor.

In Sect. 1.7 we defined the operator $\mathbf{B}_m$ which has the property that $\mathbf{B}_m(Q'_\lambda[X;t]) = Q'_{(m,\lambda)}[X;t]$. We now define a new operator in terms of the $\mathbf{B}_m$ as

$$\mathbf{B}_\lambda := \prod_{1 \le i < j \le \ell(\lambda)} (1 - tR_{ij})\mathbf{B}_{\lambda_1}\mathbf{B}_{\lambda_2}\cdots\mathbf{B}_{\lambda_{\ell(\lambda)}} , \tag{3.6}$$

where

$$R_{ij}(\mathbf{B}_{\mu_1}\mathbf{B}_{\mu_2}\cdots\mathbf{B}_{\mu_{\ell(\mu)}}) = \mathbf{B}_{\mu_1}\mathbf{B}_{\mu_2}\cdots\mathbf{B}_{\mu_i+1}\cdots\mathbf{B}_{\mu_j-1}\cdots\mathbf{B}_{\mu_{\ell(\mu)}} .$$

Alternatively $\mathbf{B}_\lambda$ is also given by the equation

$$\mathbf{B}_\lambda = \sum_{\nu,\mu} c^\nu_{\mu\lambda} s_\nu[X] s_\mu[X(t-1)]^\perp , \tag{3.7}$$

where the sum is over partitions ν and μ such that $\ell(\nu) \le \ell(\lambda)$ and $\ell(\mu) \le \ell(\lambda)$ and $c^\nu_{\mu\lambda} = \langle s_\mu s_\lambda, s_\nu \rangle$ is the Littlewood–Richardson coefficient from (1.25).

The 'parabolic' Hall–Littlewood symmetric functions that we will need are defined in terms of these operators. For a sequence of partitions $\lambda^{(*)} = (\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(d)})$ define the symmetric function

$$H_{\lambda^{(*)}}[X;t] = \mathbf{B}_{\lambda^{(1)}}\mathbf{B}_{\lambda^{(2)}}\cdots\mathbf{B}_{\lambda^{(d)}}(1) .$$

It is easy to see from (3.7) that when $t = 1$, we have $H_{\lambda^{(*)}}[X;1] = s_{\lambda^{(1)}} s_{\lambda^{(2)}} \cdots s_{\lambda^{(d)}}$. The index set of these symmetric functions is much larger than the set of partitions, so that these elements are not linearly independent. In fact, these symmetric function interpolate between the Hall–Littlewood symmetric functions and the Schur functions since $H_{(\lambda)}[X;t] = s_\lambda$, that is, the element indexed by a list containing exactly one partition, and $H_{((\lambda_1),(\lambda_2),\ldots,(\lambda_{\ell(\lambda)}))}[X;t] = Q'_\lambda[X;t]$, that is, the element indexed by a list where each partition is a single part of λ.

There is a conjectured combinatorial interpretation for the expansion of $H_{\lambda^{(*)}}[X;t]$ (see [147]) if $\lambda^{(*)}$ is a sequence of partitions such that the concatenation of the partitions in $\lambda^{(*)}$ is a partition (i.e. if for each of the adjacent partitions in $\lambda^{(*)}$ we have $\lambda^{(i)}_{\ell(\lambda^{(i)})} \ge \lambda^{(i+1)}_1$). In this case

$$H_{\lambda^{(*)}}[X;t] = \sum_T t^{\mathrm{charge}(T)} s_{\mathrm{shape}(T)} , \tag{3.8}$$

where the sum is over all tableaux T which are katabolizable with respect to $\lambda^{(*)}$.

The first algebraic definition of k-Schur functions requires an intermediary basis; the k-split basis for $\Lambda^t_{(k)}$. This basis is made up of, for each k-bounded partition λ, $G^{(k)}_\lambda[X;t] = H_{\lambda\to k}[X;t]$.

Sage Example 3.8. The 3-split basis element $G^{(3)}_{3211}$ is a t-analogue of the product of $s_3 s_{21} s_1$. The operators $\mathbf{B}_\lambda$ are programmed in SAGE and we may calculate $\mathbf{B}_{(3)}\mathbf{B}_{(2,1)}\mathbf{B}_{(1)}(1)$ in the following steps:

```
sage: s = SymmetricFunctions(QQ["t"]).schur()
sage: G1 = s[1]
sage: G211 = G1.hl_creation_operator([2,1]); G211
s[2, 1, 1] + t*s[2, 2] + t*s[3, 1]
sage: G3211 = G211.hl_creation_operator([3]); G3211
s[3, 2, 1, 1] + t*s[3, 2, 2] + t*s[3, 3, 1] + t*s[4, 1, 1, 1]
 + (2*t^2+t)*s[4, 2, 1] + t^2*s[4, 3] + (t^3+t^2)*s[5, 1, 1]
 + 2*t^3*s[5, 2] + t^4*s[6, 1]
```

This calculation shows that $\mathbf{B}_{(2,1)}(s_1) = s_{(2,1,1)} + t s_{(2,2)} + t s_{(3,1)}$ and

$$\begin{aligned}\mathbf{B}_3\left(s_{(2,1,1)} + t s_{(2,2)} + t s_{(3,1)}\right) = {} & s_{(3,2,1,1)} + t s_{(3,2,2)} + t s_{(3,3,1)} + t s_{(4,1,1,1)} + (t+2t^2) s_{(4,2,1)} \\ & + t^2 s_{(4,3)} + (t^2+t^3) s_{(5,1,1)} + 2t^3 s_{(5,2)} \\ & + t^4 s_{(6,1)}\end{aligned}$$

and this expression is equal to $G^{(3)}_{3211}$.

Then, to arrive at the k-Schur functions, a second operator is needed. Let $T^{(k)}_i$ be an operator on symmetric functions defined so that

$$T^{(k)}_i(G^{(k)}_\lambda[X;t]) = \begin{cases} G^{(k)}_\lambda[X;t] & \text{if } \lambda_1 = i, \\ 0 & \text{otherwise.}\end{cases} \tag{3.9}$$

For k-bounded partition λ, the elements $\tilde{A}_\lambda[X;t]$ were defined in [94] by a recursive algorithm; if $\ell(\lambda) = 1$ and $r \le k$, then $\tilde{A}_{(r)}[X;t] = s_{(r)}$. Otherwise we set for $\lambda_1 \le m \le k$,

$$\tilde{A}^{(k)}_{(m,\lambda_1,\lambda_2,\ldots,\lambda_{\ell(\lambda)})}[X;t] = T^{(k)}_m \mathbf{B}_m \tilde{A}_{\lambda^{(k)}}[X;t]\,. \tag{3.10}$$

Example 3.9. The 3-split of $(2,1,1)$ is $(2,1,1)^{\to 3} = ((2,1),(1))$, hence

$$\tilde{A}^{(3)}_{211} = T^{(3)}_2 \mathbf{B}_2(s_{11}) = T^{(3)}_2(s_{211} + t s_{31}) = T^{(3)}_2(G^{(3)}_{211} - G^{(3)}_{22}) = s_{211} + t s_{31}.$$

Moreover, we may calculate using Example 3.8 that

$$\begin{aligned}\mathbf{B}_3(\tilde{A}^{(3)}_{211}) &= s_{3211} + ts_{331} + ts_{4111} + (t+t^2)s_{421} + t^2 s_{43} + (t^2+t^3)s_{51} \\ &\quad + t^3 s_{52} + t^4 s_{61} \\ &= G^{(3)}_{3211} - ts_{322} - t^2 s_{421} - t^3 s_{52}\,.\end{aligned}$$

We can also determine that $G^{(3)}_{322} = s_{322} + ts_{421} + t^2 s_{52}$ (using the same techniques as in Example 3.8), so that

$$\tilde{A}^{(3)}_{3211} = T_3^{(3)}\mathbf{B}_3(\tilde{A}^{(3)}_{211}) = G^{(3)}_{3211} - tG^{(3)}_{322}\,.$$

This (not coincidently) is equal to $A^{(3)}_{3211}[X;t]$ as was calculated in Example 3.7.

The algebraic operations mimic those of the combinatorial definition defined in the previous section and so it is important to emphasize that the combinatorial and algebraic definitions are equivalent.

Conjecture 3.10. *For $k > 0$ and a k-bounded partition λ,*

$$A^{(k)}_\lambda[X;t] = \tilde{A}^{(k)}_\lambda[X;t]\,. \tag{3.11}$$

3.3 Weak Tableaux II

We have seen that a fruitful characterization for k-Schur functions (without parameter t) is given by inverting (2.14); for $\mu_1 \le k$,

$$h_\mu = \sum_{\lambda:\lambda_1\le k} K^{(k)}_{\lambda\mu}\, s^{(k)}_\lambda\,, \tag{3.12}$$

where the weak Kostka numbers $K^{(k)}_{\lambda\mu}$ count weak k-tableaux. Here we present k-Schur functions that reduce to these parameterless k-Schur functions when $t = 1$. The method is to introduce *weak Kostka-Foulkes polynomials* as polynomials in $\mathbb{N}[t]$ defined by refining the charge statistic to a statistic that associates a non-negative integer called the k-charge to each k-tableau. Then setting

$$K^{(k)}_{\lambda\mu}(t) = \sum_{\substack{\mathrm{shape}(T)=\mathfrak{c}(\lambda)\\ \mathrm{weight}(T)=\mu}} t^{\mathrm{kcharge}(T)}\,,$$

it happens that $K^{(k)}_{\lambda\lambda}(t) = 1$ and since there are no k-tableaux of shape $\mathfrak{c}(\lambda)$ and weight μ when $\mu > \lambda$, the k-charge matrix $K^{(k)}_{\lambda\mu}(t)$ is unitriangular. So, in the spirit of (3.12),

$$Q'_\mu[X;t] = \sum_\lambda K^{(k)}_{\lambda\mu}(t)\, \tilde{s}^{(k)}_\lambda[X;t] \tag{3.13}$$

characterizes the functions $\{\tilde{s}^{(k)}_\lambda[X;t]\}$.

There are several different characterizations for k-charge. We give here two distinct formulations defined directly on k-tableaux, discovered by Lapointe-Pinto and Morse [36, 99]. There are other formulations including one on α-factorizations, one on an object called affine Bruhat countertableau [35, 36], and in relation with the energy function on Kirillov–Reshetikhin crystals [125].

The k-charge statistic on k-tableaux is first described in the standard case since it is in these terms that we define it for semi-standard k-tableaux. Important to the definition is a number $\mathrm{diag}(c_1, c_2)$, associated to cells c_1 and c_2 in a $(k+1)$-core, defined to be the number of diagonals of residue x that are strictly between c_1 anc c_2 where x is the residue of the lower cell. When it is well-defined to do so, functions defined with a cell as input can instead take a letter as input. For example, in a standard k-tableau it is natural to discuss the residue of a specific letter (since any cell containing that letter has the same residue) instead of the residue of a specific cell.

Definition 3.11. Given a standard k-tableau T on m letters, put a bar on the topmost occurrence of letter r, for each $r = 1, \ldots, m$. Define the *index* of T, starting from $I_1 = 0$, by

$$I_r = \begin{cases} I_{r-1} + 1 + \mathrm{diag}(\bar{r}, \overline{r-1}) & \text{if } \bar{r} \text{ is east of } \overline{r-1} \\ I_{r-1} - \mathrm{diag}(\bar{r}, \overline{r-1}) & \text{otherwise}\,, \end{cases} \tag{3.14}$$

for $r = 2, \ldots, m$. The *k-charge* of T is the sum of entries in $I(T)$, denoted by $\mathrm{kcharge}(T)$.

Example 3.12. For $k = 3$,

$$T = \begin{array}{|c|c|c|c|c|} \hline 4_2 \\ \cline{1-1} 2_3 & 6_0 \\ \cline{1-2} 1_0 & 3_1 & 4_2 & 5_3 & 6_0 \\ \hline \end{array} \quad \Longrightarrow \quad I(T) = [0, 0, 1, 1, 3, 3] \quad \Longrightarrow \quad \mathrm{kcharge}(T) = 8\,.$$

It is not immediately clear that the k-charge is a non-negative integer and it is sometimes helpful to use a different formulation of k-charge. Let $T_{\le x}$ denote the subtableau obtained by deleting all letters larger than x from T.

Definition 3.13. Given a k-tableau T, the T-residue order of $\{0, \ldots, k\}$ is defined by

$$x > x-1 > \cdots > 0 > k > \cdots > x+1\,,$$

where x is the residue of the highest addable corner of T. Note that $x = 1 - \ell(\lambda)$ (mod $k+1$), for λ the shape of T.

Example 3.14. With $k = 3$, consider

$$T = \begin{array}{|c|cccc}\cline{1-1} 4_3 \\ \hline 1_0 & \multicolumn{1}{|c|}{2_1} & \multicolumn{1}{c|}{3_2} & \multicolumn{1}{c|}{4_3} \\ \hline\end{array} \qquad T_{\leq 3} = \begin{array}{|c|c|c|}\hline 1_0 & 2_1 & 3_2 \\ \hline\end{array}\,.$$

The $T_{\leq 3}$-residue order is $3 > 2 > 1 > 0$ and the T-residue order is $2 > 1 > 0 > 3$.

Given a standard k-tableau T on m letters, define the *index* $J(T) = [J_1, \ldots, J_m]$, starting from $J_1 = 0$, by setting for $r = 2, \ldots, m$,

$$J_r = \begin{cases} J_{r-1} + 1 & \text{if } \operatorname{res}(r) > \operatorname{res}(r-1) \\ J_{r-1} & \text{otherwise}\,, \end{cases} \tag{3.15}$$

under $T_{\leq r}$-residue order (see Example 3.16).

Proposition 3.15. *For a standard k-tableau T of shape λ,*

$$\operatorname{kcharge}(T) = \sum_r \left(J_r(T) + \operatorname{diag}(c_r, c^{(r)})\right)\,,$$

where c_r is the highest cell containing an r and $c^{(r)} = (\ell(\operatorname{shape}(T_{\leq r})) + 1, 1)$.

Example 3.16. For $k = 3$,

$$T = \begin{array}{|c|c|ccc}\cline{1-1} 4_2 \\ \cline{1-2} 2_3 & 6_0 \\ \hline 1_0 & 3_1 & \multicolumn{1}{c|}{4_2} & \multicolumn{1}{c|}{5_3} & \multicolumn{1}{c|}{6_0} \\ \hline\end{array} \implies J(T) = [0, 0, 1, 1, 2, 3]\,, \operatorname{diag}(c_5, (4, 1)) = 1$$
$$\implies \operatorname{kcharge}(T) = 8\,.$$

Remark 3.17. Given a standard k-tableau of shape λ where $k \geq h(\lambda)$, the index conditions (3.14) and (3.15) both reduce to (1.37). Thus, since a diagonal of residue x occurs at most once in λ for any x, k-charge reduces to charge.

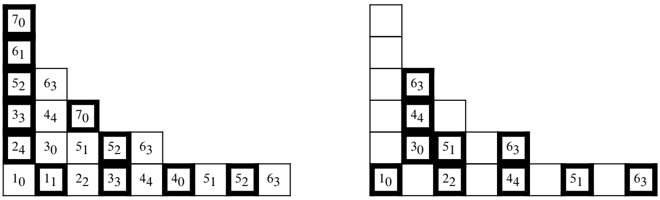
As with the charge of Lascoux and Schützenberger, we extend the definition of k-charge to semi-standard k-tableaux by successively computing on an appropriate choice of standard sequences. The trick is to introduce a method for making this choice in k-tableaux.

Definition 3.18. From an x (of some residue i) in a semi-standard k-tableau T, the appropriate choice of $x+1$ will be determined by choosing its residue from the set A of all $(k+1)$-residues labelling $x+1$'s. Reading counter-clockwise from i, this choice is the closest $j \in A$ on a circle labelled clockwise with $0, 1, \ldots, k$.

Remark 3.19. Definition 3.18 reduces to Definition 1.41 when T is a semi-standard tableau of shape λ and $k \geq h(\lambda)$. In particular, consider x of $(k+1)$-residue i in T. Note that $h(\lambda) \leq k$ implies there is a unique cell c of residue i that contains x. Let j be the first entry on the circle reading counter-clockwise from i that is a residue of a cell containing $x+1$. If there is an $x+1$ above c, then the south-easternmost cell containing an $x+1$ that is above c has residue j since there are no $x+1$'s of a residue counter-clockwise between i and j. If there are none above c, then for the same reason, the south-easternmost cell containing an $x+1$ has residue j.

In the semistandard case, we need to be more specific about the residue order used in (3.15). The letters r that occur in the tableau are ordered with respect to the standard subsequence they belong to under Definition 3.18. So r in the first chosen standard subword is bigger than the one from the second standard subword etc. Each letter r has its own distinct residue i. The J-index of (3.15) should be computed with respect to the $T_{\leq r_i}$-residue order when dealing with letter r_i.

Example 3.20. With $k=4$ and weight $(2,2,2,2,2,2,1)$:



Note that for example

$$T_{\leq 6_3} = \begin{array}{|c|c|c|c|c|c|c|c|c|}\cline{1-2} 5_2 & 6_3 \\ \cline{1-2} 3_3 & 4_4 \\ \cline{1-5} 2_4 & 3_0 & 5_1 & 5_2 & 6_3 \\ \hline 1_0 & 1_1 & 2_2 & 3_3 & 4_4 & 4_0 & 5_1 & 5_2 & 6_3 \\ \hline \end{array}$$

so that the highest addable cell in $T_{\leq 6_3}$ has residue 1. We have

$J = [0,0,0,1,1,1,1]$ and $\mathrm{diag}(c_4, c^{(4)}) = 1$ $\qquad$ $J = [0,1,1,1,2,2]$

$I = [0,0,0,2,1,1,1]$ since $\mathrm{diag}(\bar{3},\bar{4}) = 1, \mathrm{diag}(\bar{4},\bar{5}) = 1$ $\qquad$ $I = [0,1,1,1,2,2]$

Using either index I or J, we find that the k-charge of T is 12.

Sage Example 3.21. We verify the above example in SAGE:

```
sage: T = WeakTableau([[1,1,2,3,4,4,5,5,6],[2,3,5,5,6],
....:      [3,4,7],[5,6],[6],[7]],4)
sage: T.k_charge()
12
```

We can also demonstrate an example of Eq. (3.13) using SAGE.

```
sage: Sym=SymmetricFunctions(QQ["t"].fraction_field())
sage: Qp = Sym.hall_littlewood().Qp()
sage: ks = Sym.kBoundedSubspace(3).kschur()
sage: t = ks.base_ring().gen()
sage: ks(Qp[3,2,2,1])
ks3[3, 2, 2, 1] + t*ks3[3, 3, 1, 1] + t^2* ks3[3, 3, 2]
sage: sum(t^T.k_charge()*ks(la) for la in
....:     Partitions(8, max_part=3)
....:     for T in WeakTableaux(3,la,[3,2,2,1],
....:     representation = 'bounded'))
ks3[3, 2, 2, 1] + t*ks3[3, 3, 1, 1] + t^2*ks3 [3, 3, 2]
```

3.4 Strong Tableaux II

The definition of the k-Schur functions in terms of strong marked tableaux as in (2.29) also has a version with a parameter t. For this definition we need to define the spin of a strong marked ribbon. Recall that if τ and κ are $(k+1)$-cores such that $\tau \Rightarrow_k \kappa$, then κ/τ is a skew partition which consists of several copies of connected components which are ribbons of the same size and shape. Let h represent the *height* of one of these ribbons (that is that it occupies h rows). Now a strong marked cover consists of the skew partition κ/τ and a marking c of one of the connected components. If there are r connected components in κ/τ, then the *spin* of a marked cover is equal to $(h-1)\times r$ plus the number of ribbons which are above the marked one.

The *spin of a strong marked tableau*, $\kappa^{(0)} = \emptyset \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(m)}$ with markings $c_1, c_2, \ldots, c_m$ is the sum of the spins of the strong marked ribbons $\kappa^{(i)}/\kappa^{(i-1)}$ with marking c_i.

Example 3.22. Recall from Example 2.23 the marked semi-standard tableau with $k=3$:

$$\begin{array}{|c|c|cc}
\cline{1-2}
3_1^* & 4_1^* & & \\
\cline{1-2}
2_2 & 3_3 & & \\
\hline
2_1^* & 3_2 & 3_2^* & \multicolumn{1}{c|}{4_1} \\
\hline
1_1^* & 1_2^* & 2_2^* & \multicolumn{1}{c|}{3_3^*} \\
\hline
\end{array}\;.$$

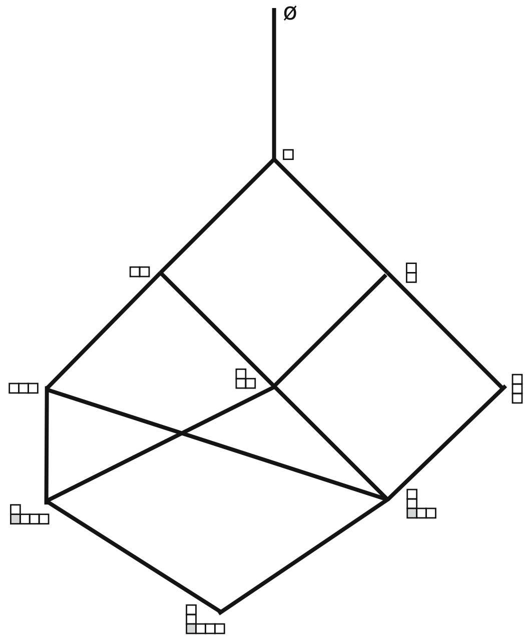
There is a contribution of 1 to the spin for the ribbon of cells labelled by 2_2 and there is a contribution of 1 due to the labeling of the lower occurrence of 3_3. Therefore the total spin of this tableau is 2.

The k-Schur function (this time with a t) in terms of strong tableaux are defined as (see also [81, Conjecture 9.11])

$$s_\lambda^{(k)}[X;t] = \sum_{\mu \vdash |\lambda|} \sum_{(\kappa^{(*)}, c_*)} t^{\mathrm{spin}(\kappa^{(*)}, c_*)} m_\mu \ , \tag{3.16}$$

where the sum is over all strong marked tableaux $(\kappa^{(*)}, c_*)$ of shape λ and weight μ.

Example 3.23. For this definition, there is a reason to choose a smaller example for computation. We have thus far used as our running example the 3-Schur function indexed by the partition $(3,2,1,1)$. For the coefficient of the monomial $m_{1111111}$ there are 210 strong marked tableaux. To choose a smaller example we take as an example the 3-bounded partition $(3,1,1)$ which corresponds to the 4-core $(4,1,1)$ which has 10 strong marked tableaux in total. The strong partial order on 4-cores contains the following interval (see Fig. 2.5):



These correspond to the following 10 marked strong standard tableaux:

$$\begin{array}{|c|c|c|c|}\cline{1-1} 5^* \\ \cline{1-1} 4 \\ \hline 1^* & 2^* & 3^* & 4^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 4 \\ \cline{1-1} 4^* \\ \hline 1^* & 2^* & 3^* & 5^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 4 \\ \cline{1-1} 3^* \\ \hline 1^* & 2^* & 4^* & 5^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 4 \\ \cline{1-1} 2^* \\ \hline 1^* & 3^* & 4^* & 5^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 5^* \\ \cline{1-1} 3^* \\ \hline 1^* & 2^* & 4 & 4^* \\ \hline \end{array},$$

$$\begin{array}{|c|c|c|c|}\cline{1-1} 4^* \\ \cline{1-1} 3^* \\ \hline 1^* & 2^* & 4 & 5^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 4^* \\ \cline{1-1} 2^* \\ \hline 1^* & 3^* & 4 & 5^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 5^* \\ \cline{1-1} 2^* \\ \hline 1^* & 3^* & 4 & 4^* \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 5^* \\ \cline{1-1} 4^* \\ \hline 1^* & 2^* & 3^* & 4 \\ \hline \end{array}, \quad \begin{array}{|c|c|c|c|}\cline{1-1} 3^* \\ \cline{1-1} 2^* \\ \hline 1^* & 4 & 4^* & 5^* \\ \hline \end{array}.$$

The first four of these strong marked standard tableaux have spin equal to 1 and the remaining 6 have spin equal to 0. There is a semi-standard tableau of weight μ which corresponds to the standard tableau if the cells labeled $1, 2, \ldots, \mu_1$ form a strong marked horizontal strip, $\mu_1 + 1, \mu_1 + 2, \ldots, \mu_1 + \mu_2$ form another strong marked horizontal strip, etc. From these 10 strong marked standard tableaux it is possible to read off that

$$s^{(3)}_{311}[X;t] = tm_{41} + tm_{32} + (1+2t)m_{311} + (1+2t)m_{221} + (3+3t)m_{2111} + (6+4t)m_{11111}.$$

Sage Example 3.24. In this example we will show how SAGE can be used to compute the monomial expansion of the k-Schur function by computing a statistic for each strong k-tableau.

```
sage: t = var("t")
sage: for mu in Partitions(5):
....:     print mu, sum(t^T.spin() for T in
....:         StrongTableaux(3,[4,1,1],mu))
....:
[5] 0
[4, 1] t
[3, 2] t
[3, 1, 1] 2*t + 1
[2, 2, 1] 2*t + 1
[2, 1, 1, 1] 3*t + 3
[1, 1, 1, 1, 1] 4*t + 6
sage: StrongTableaux( 3, [4,1,1], (1,)*5 ).cardinality()
10
sage: StrongTableaux( 3, [4,1,1], (1,)*5 ).list()
[[[-1, -2, -3, 4], [-4], [-5]],
 [[-1, -2, -3, -4], [4], [-5]],
 [[-1, -2, -3, -5], [-4], [4]],
 [[-1, -2, 4, -4], [-3], [-5]],
 [[-1, -2, 4, -5], [-3], [-4]],
 [[-1, -2, -4, -5], [-3], [4]],
 [[-1, -3, 4, -4], [-2], [-5]],
 [[-1, -3, 4, -5], [-2], [-4]],
 [[-1, -3, -4, -5], [-2], [4]],
 [[-1, 4, -4, -5], [-2], [-3]]]
```

The definitions $A^{(k)}_\lambda[X;t]$, $\tilde{A}^{(k)}_\lambda[X;t]$, $s^{(k)}_\lambda[X;t]$, and $\tilde{s}^{(k)}_\lambda[X;t]$ of what are generically known as k-Schur functions have very different characters. Each connects to a different area of algebraic combinatorics and has its own benefits and detriments. The conjectured properties of k-Schur functions are sometimes clear from one definition, but difficult to prove for another. In the next section we will examine these properties and see why it would be very beneficial to resolve the following conjecture.

Conjecture 3.25. *For $k > 0$ and a k-bounded partition λ,*

$$s^{(k)}_\lambda[X;t] = A^{(k)}_\lambda[X;t] = \tilde{A}^{(k)}_\lambda[X;t] = \tilde{s}^{(k)}_\lambda[X;t].$$

3.5 Notes on References

The atom definition of the k-Schur functions was the first description and they were referred to by Alain Lascoux as potatoes ('les patates') since when copies of the atoms were identified on photocopies of the cyclage poset they could be circled to form tuber-like shapes (which we have endeavored to recreate in Example 3.1). The combinatorial definition uses the concept of katabolizable tableaux [145, 146] which generalizes the notion of cyclage which comes from [107]. The non-recursive definition of the charge statistic on tableaux also comes from [107].

It turns out that the atom definition is still roughly the easiest and fastest definition to implement on a computer. The implementation of the k-Schur functions in SAGE uses the definition of $A_\lambda^{(k)}[X;t]$. For this reason, it is important to note that while Conjecture 3.10 has been checked up to degree $m = 19$, Conjecture 3.25 has only been checked up to bounded partitions of $m = 11$, but for $m > 11$ there is currently no proof that these definitions are equivalent in general, even for $t = 1$.

The operators $\mathbf{B}_\lambda$ from Eq. (3.6) were defined by Shimozono and Zabrocki in [148] as a tool for understanding the generalized Kostka polynomials (also known as parabolic Kostka coefficients) that were studied by Shimozono and Weyman [147]. The combinatorial interpretation for the Schur expansion of a composition of these operators when the indexing partitions concatenate to a partition in terms of katabolizable tableaux is still an open problem. Certain cases of this are known (e.g. [143]) and its resolution would be helpful in attacking Conjecture 3.10. In [23] the Bernstein operators at $t = 1$ are used to derive a recursion relation for the k-Schur functions which allow an easy expansion in the complete homogeneous basis.

In Refs. [92, 93] the definition of the k-Schur functions at $t = 1$ is taken to be the symmetric functions which satisfy the k-Pieri rule of Eq. (2.19). Later Lam, Lapointe, Morse and Shimozono [81] showed that the k-Schur functions which satisfy the k-Pieri rule (2.19) are equivalent to Eq. (3.16) at $t = 1$, and Eq. (3.16) is [81, Conjecture 9.11]. Assaf and Billey [6, Definition 3.2] have a slightly different statement of Eq. (3.16) as a quasi-symmetric function expansion in the fundamental basis.

Recently, Dalal and Morse [35] have proven a second characterization of the k-Pieri at $t = 1$ which leads to a very different sort of tableaux that enumerate $K_{\lambda\mu}^{(k)}$. They use these tableaux to provide an alternative combinatorial interpretation to Eq. (1.38).

We note that the line between what is called a 'conjectured property' and what is called a 'definition' is sometimes a little blurry because we have provided several definitions of k-Schur functions which are conjectured to be equivalent. There are reasons to do this instead of taking one as definition and the rest as conjectured formulas; historically the k-Schur functions were presented in the literature this way. However, we note that there may be more definitions of the k-Schur functions than presented in this section. For instance, the k-shape poset presented in Sect. 7 and the representation theoretical definition which is briefly discussed in Sect. 5.5

are two other properties which may be taken as definitions, but are currently only conjectured to be equivalent to the definitions presented here.

4 Properties of k-Schur Functions and Their Duals

In this section we list many of the properties of the k-Schur functions, both conjectured and proven. This section is mainly meant to be a statement of the 'current state of affairs' and is likely to change. There may not be a lot to say about each of these properties and instead we provide a proper reference for the statement of the conjecture or property, but we shall endeavor to add a few words about what needs to be proven in order to say why the property is true. The properties are each marked in the discussion with one of four headers $A^{(k)}$, $\tilde{A}^{(k)}$, $\tilde{s}^{(k)}$, $s^{(k)}$ to indicate comments on which of the four definitions (the k-atom definition of Sect. 3.1, the operator definition of Sect. 3.2, the weak tableaux definition of Sect. 3.3, and the strong tableaux definition of Sect. 3.4). Since $s_\lambda^{(k)} = \tilde{s}_\lambda^{(k)}$ at $t = 1$ by [81, Theorem 4.11], we often group these two definitions together.

4.1 *k-Schur Functions Are Schur Functions When $k \geq |\lambda|$ and When $t = 0$*

Note that the stated property for $k \geq |\lambda|$ (resp. $t = 0$) are two different statements. Nevertheless, it has been proven for all definitions of the k-Schur functions. It is known that $s_\lambda^{(k)}[X;t] = s_\lambda$ plus other terms indexed by partitions μ which are strictly larger in dominance order, each with a positive power of t as a coefficient.

$A^{(k)}$: (proven, [91, Property 6]) When k is larger than the largest hook of λ, then the only katabolizable tableau is the semi-standard tableau of shape λ and weight λ. The charge of this tableau is 0 so it is clear that $A_\lambda^{(k)}[X;t] = s_\lambda$. For $t = 0$, we know that $A_\lambda^{(k)}[X;0] = s_\lambda$ because only the tableau of shape λ and weight λ has charge 0 in the atom tableaux.

$\tilde{A}^{(k)}$: (proven, [94, Property 8]) This is slightly more complicated than the tableau definition but notice that $G_\lambda^{(k)}[X;t] = s_\lambda$ if $k \geq |\lambda|$. The action of the operator $\mathbf{B}_m(s_\lambda) = s_{(m,\lambda_1,\ldots,\lambda_{\ell(\lambda)})}$ + other terms with first part which is larger than m (and hence will be killed by the operator $T_m^{(k)}$). Also at $t = 0$, the operator $\mathbf{B}_m\Big|_{t=0} = \mathbf{S}_m$ and $G_\lambda^{(k)}[X;0] = s_\lambda$.

$s^{(k)}$: (proven, [92, Property 39]) One would need to trace through the definition of the strong order, but for $k \geq |\lambda|$, all marked strong tableaux will be standard tableaux and the markings of the marked strong tableaux are exactly the condition that these tableaux should be isomorphic to semi-standard tableaux.

Because the heights of the ribbons are all 1, the spins for all tableaux are 0. Therefore, $s_\lambda^{(k)}[X;t] = \sum_\mu K_{\lambda\mu} m_\mu = s_\lambda$.

$\tilde{s}^{(k)}$: (proven) When k is large, weak k-tableaux are usual semi-standard tableaux and (3.13) turns into the definition of Schur functions. Similarly, $Q'_\lambda[X;0] = s_\lambda$ which shows the second statement.

4.2 The *k*-Schur Function Is Schur Positive

In fact, a more refined conjecture to the statement that k-Schur functions are Schur positive, is that the k-Schur functions expand positively in the $(k+1)$-Schur functions. This property is usually referred to as 'k-branching' and is discussed in slightly more detail in Sect. 4.10 and again in Sect. 7. Repeated applications of the k-branching property yields a positive expansion of the k-Schur functions in the Schur functions. An explicit rule for the k-branching formula with a t is conjectured in [82, Conjecture 1].

$A^{(k)}$: (proven, [91, Property 7]) This property follows directly from the definition. Since $A_\lambda^{(k)}[X;t]$ is a sum of Schur functions, one for each tableau in the set, this property is true by definition.

$\tilde{A}^{(k)}$: (conjecture, [94, Eq. (1.6)]) Equation (3.8) is an outstanding conjecture due to Shimozono and Weyman [147] that $H_{\lambda^{(*)}}[X;t]$ is Schur positive whenever the concatenation of all of the partitions of $\lambda^{(*)}$ form a partition (which happens for all k-splits of partitions). Conjecture 3.10 is likely to follow from this conjecture and hence so would the property that $\tilde{A}_\lambda^{(k)}[X;t]$ is Schur positive.

$s^{(k)}$: (for arbitrary t it follows from [82, Conjecture 1], for $t = 1$ follows from [82, Theorem 2]) In Sect. 7 we will give a precise conjecture of how $s_\lambda^{(k)}[X;t]$ expands positively in $s_\lambda^{(k+1)}[X;t]$. If we iteratively apply that rule, it must be that $s_\lambda^{(k)}[X;t]$ expands positively in the limit as k increases, and when $k \geq \lambda_1 + \ell(\lambda) - 1$ then $s_\lambda^{(k)}[X;t] = s_\lambda[X]$. In addition, some conjectures of [81] were proven in [13], in particular that "skew shaped" strong Schur functions are symmetric.

Assaf and Billey [6] convert the monomial expansion of k-Schur functions in terms of strong marked tableaux into a quasi-symmetric function expansion. They conjectured that a dual equivalence graph structure can be placed on strong tableaux and using Assaf's earlier work [4, 5] this structure can in theory be used to show that the $s_\lambda^{(k)}[X;t]$ are Schur positive as long as a computer check on a finite number of elements is completed. Billey informs us that this calculation has a current estimated running time which is quite long and has not yet been completed.

$\tilde{s}^{(k)}$: (conjecture, proven for $t = 1$) Since $s_\lambda^{(k)} = \tilde{s}_\lambda^{(k)}$ for $t = 1$, this property is true by [82, Theorem 2]. A promising approach for generic t is underway; by duality, the positivity of $\tilde{s}_\lambda^{(k)}[X;t]$ in terms of $\tilde{s}_\mu^{(k+1)}[X;t]$'s follows from the positivity of dual $k+1$-Schur functions into dual k-Schur functions. This, in turn, would follow by showing that there is a compatibility between k-charge and the weak bijection introduced in [82]. A partial solution has been given in [99], where the compatibility is shown for standard k-tableaux.

4.3 At $t = 1$, the k-Schur Functions Satisfy the k-Pieri Rule

In Sects. 2.2 and 2.4 we discussed the k-Pieri rule at $t = 1$. It states that for $1 \leq r \leq k$, $h_r s_\lambda^{(k)}[X;1] = \sum_\mu s_\mu^{(k)}[X;1]$ where the sum is over all partitions μ such that $\mu \vdash |\lambda| + r$, $\mathfrak{c}(\lambda) \subseteq \mathfrak{c}(\mu)$ and $\mathfrak{c}(\mu)/\mathfrak{c}(\lambda)$ is a weak horizontal strip.

$A^{(k)}$: (conjecture [91, Conjecture 41]) The tableaux operations for the definition of the k-atoms are somewhat unusual and more work needs to be done to understand many of their properties. Combinatorially, the k-Pieri rule is most clearly stated in terms of $(k+1)$-cores, the connection between k-atoms and $(k+1)$-cores is not clear in this definition.

$\tilde{A}^{(k)}$: (conjecture [94, Conjecture 21]) In order to prove this property it seems as though it would be necessary to understand the commutation relationship between multiplication by a symmetric function h_r and the operators T_i of Eq. (3.9) when $t = 1$.

$\tilde{s}^{(k)}$: (proven [92, Theorem 29]) This property follows by showing that the number of weak k-tableaux of weight α equals the number of k-tableaux of weight β, for β any rearrangement of the parts of α.

$s^{(k)}$: (proven [81, Theorem 4.11]) It was proven in [81] that $s_\lambda^{(k)} = \tilde{s}_\lambda^{(k)}$ when $t = 1$. The result then follows from the Pieri rule on $\tilde{s}_\lambda^{(k)}$ from [92].

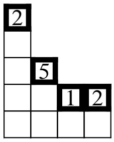
It is conjectured that the k-Schur functions also satisfy a t-analogue of the k-Pieri rule Eq. (2.19) where the operator $\mathbf{B}_m$ takes the role of multiplication by h_m (and reduces as such when $t = 1$). Recall that Zabrocki [158] determined the action of $\mathbf{B}_m$ on Schur functions in his thesis and he gave a new proof for the charge formulation of Hall–Littlewood polynomials $Q_\lambda[X;t]$.

Given a $(k+1)$-core λ with $\lambda_1 = m$, let ρ be the unique partition whose first part is m and where ρ/λ is a horizontal strip of size m (that is, ρ is obtained by adding a cell to the top of each column of λ). For $\mathfrak{p}(\lambda)_1 \leq r \leq k$, Maria-Elena Pinto conjectured that

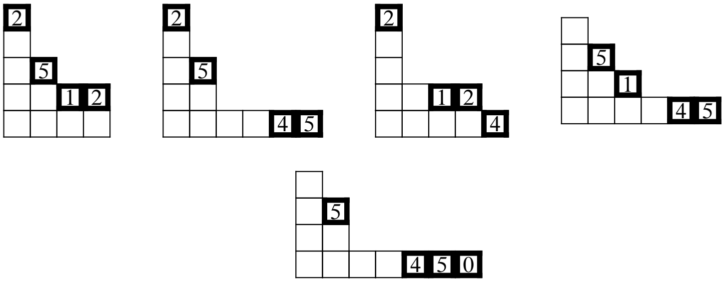
$$\mathbf{B}_r\, s_\lambda^{(k)}[X;t] = \sum_{\mu/\lambda \text{ weak } r\text{-strip}} t^{a(\mu)} s_\mu^{(k)}[X;t]\,, \tag{4.1}$$

where $a(\mu)$ is the number of cells of ρ/λ whose residue does not label a cell of μ/λ (see also [36] for a different conjectured formula using strong order chains in $\tilde{A}^k$).

Example 4.1. Let $k = 5$ and consider $\lambda = (4, 2, 1, 1)$, so that $\mathfrak{p}(\lambda) = (3, 2, 1, 1)$.

$\rho =$  .

Possible horizontal weak 3-strips are



and their respective powers are t^0, t^1 (the cell of residue 1 in ρ), t^1 (the cell of residue 5 in ρ), t^2 (the two cells of residue 2 in ρ) and t^3 (the cell of residue 1 and the 2 cells of residue 2 in ρ). This gives

$$\mathbf{B}_3\, s^{(5)}_{4211}[X;t] = s^{(5)}_{44211}[X;t] + t\, s^{(5)}_{62211}[X;t] + t\, s^{(5)}_{54111}[X;t] + t^2\, s^{(5)}_{6321}[X;t] + t^3\, s^{(5)}_{7221}[X;t]\,.$$

As with the action of $\mathbf{B}_m$ on a Schur functions, if $m < \lambda_1$, the k-Schur expansion of $\mathbf{B}_m(s^{(k)}_\lambda[X;t])$ has negative terms. Currently, there is no conjecture describing these terms that cancel when $t = 1$.

Sage Example 4.2. Here we demonstrate the action of the $\mathbf{B}_m$ operator on the k-Schur function basis. For $k \geq m \geq \lambda_1 - 1$ this is conjectured to expand positively in the k-Schur basis.

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks4 = Sym.kschur(4)
sage: ks4([3, 1, 1]).hl_creation_operator([1])
(t-1)*ks4[2, 2, 1, 1] + t^2*ks4[3, 1, 1, 1] + t^3*ks4[3, 2, 1]
 + (t^3-t^2)*ks4[3, 3] + t^4*ks4[4, 1, 1]
sage: ks4([3, 1, 1]).hl_creation_operator([2])
t*ks4[3, 2, 1, 1] + t^2*ks4[3, 3, 1] + t^2*ks4[4, 1, 1, 1]
 + t^3*ks4[4, 2, 1]
sage: ks4([3, 1, 1]).hl_creation_operator([3])
ks4[3, 3, 1, 1] + t*ks4[4, 2, 1, 1] + t^2*ks4[4, 3, 1]
sage: ks4([3, 1, 1]).hl_creation_operator([4])
ks4[4, 3, 1, 1]
```

4.4 k-Conjugation

The ω-involution sending s_λ to $s_{\lambda'}$ acts simply on k-Schur functions as well. It was conjectured in [91, 94] that, for some non-negative power of t,

$$\omega(s_\lambda^{(k)}[X;t]) = t^d s_{\lambda^{\tilde{\omega}_k}}^{(k)}[X;1/t], \tag{4.2}$$

where $\lambda^{\tilde{\omega}_k}$ is a k-bounded partition depending on λ and k. Later, it was shown in [96] that $\lambda^{\tilde{\omega}_k} = \lambda^{\omega_k}$ corresponds to usual conjugation in the $k+1$-core framework (see Definition 1.19). One consequence of k-branching [82] is that the power of d counts cells of $\mathfrak{c}(\lambda)$ with hook greater than k (see also [31]).

$A^{(k)}$: (conjecture [91, Conjecture 36]) Using the definition of k-Schur functions in terms of tableaux atoms, there is definitely an orientation in the definition of katabolism that is not compatible with the notion of conjugation of the tableau. This is because the first step of the definition involves a split of the reading word of the tableau which involves reading the rows, and it really is not clear how this would carry to the columns.
One possible approach to the tableaux definition would be to show that there is a bijection between the tableaux in the atoms $\mathbb{A}_\lambda^{(k)}$ and $\mathbb{A}_{\lambda^{\omega_k}}^{(k)}$. Since the k-conjugation is most easily expressed in terms of $(k+1)$-cores, it seems that some connection between atoms and $(k+1)$-cores will need to be found.

$\tilde{A}^{(k)}$: (conjecture [94, Conjecture 40], at $t = 1$ this is [94, Conjecture 19]) Since the algebraic definition depends on the operation of k-split, it is not clear how the action of ω interacts with the individual operators. There is some hope that some algebraic tools will be developed to resolve this conjecture by looking at the expansion in terms of $s_\lambda[X/(1-t)]$ since for an element $f[X;t] \in \Lambda_{(n)}^t$ it is at least known that $\omega(f[X;1/t]) \in \Lambda_{(n)}^t$.

$s^{(k)}, \tilde{s}^{(k)}$: (conjecture, for $t = 1$ this is [92, Theorem 38]) At $t = 1$ this property follows because of [92, Theorem 33], where a formula for the product of e_r and a k-Schur function is given as

$$e_r s_\lambda^{(k)} = \sum_\mu s_\mu^{(k)}, \tag{4.3}$$

where the sum is over all k-bounded partitions μ of size $|\lambda| + r$ such that μ/λ is a vertical strip and $\mu^{\omega_k}/\lambda^{\omega_k}$ a horizontal strip.

Sage Example 4.3. Let us check an example of Eq. (4.2) by a calculation in SAGE. In order to invert the parameter t, we must first expand the k-Schur function in a basis which is independent of the parameter t. In fact, if we apply the involution ω alone, the function no longer lies in the space spanned by the k-Schur functions.

However, if we apply ω and invert the parameter, then it does belong to the right space.

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks3 = Sym.kschur(3)
sage: ks3([3,2]).omega()
Traceback (most recent call last):
...
ValueError: t^2*s[1, 1, 1, 1, 1] + t*s[2, 1, 1, 1]
 + s[2, 2, 1] is not in the image

sage: s = Sym.schur()
sage: s(ks3[3,2])
s[3, 2] + t*s[4, 1] + t^2*s[5]
sage: t = s.base_ring().gen()
sage: invert = lambda x: s.base_ring()(x.subs(t=1/t))
sage: ks3(s(ks3([3,2])).omega().map_coefficients(invert))
1/t^2*ks3[1, 1, 1, 1, 1]
```

In fact, there is a short-cut for the last computation in SAGE by simply asking

```
sage: ks3[3,2].omega_t_inverse()
1/t^2*ks3[1, 1, 1, 1, 1]
```

4.5 The k-Schur Functions Form a Basis for $\Lambda^t_{(k)}$

Recall from Eq. (3.5) that the definition of $\Lambda^t_{(k)}$ is the linear span over $\mathbb{Q}(q,t)$ of the symmetric functions $H_\lambda[X;q,t]$ (or $s_\lambda[X/(1-t)]$ or $Q'_\lambda[X;t]$) over all partitions λ with $\lambda_1 \leq k$. For each of our definitions, it is not necessarily clear that the k-Schur functions even lie in $\Lambda^t_{(k)}$. However, if they do and if they are linearly independent, they will form a basis since they are also indexed by k-bounded partitions.

$A^{(k)}$: (conjecture [91, Conjecture 8]) The atoms are known to be linearly independent [91, Property 7] since they are triangular with respect to the Schur functions. Nevertheless, it remains a conjecture that they are elements of $\Lambda^t_{(k)}$; their combinatorial definition does not give a direct connection with the known bases of $\Lambda^t_{(k)}$. It seems as though the most likely means of proving this conjecture is to show Conjecture 3.10, otherwise there is no obvious connection with the spanning elements which define $\Lambda^t_{(k)}$.

$\tilde{A}^{(k)}$: (proven [94, Theorem 33]) This result is non-trivial because it is not easy to demonstrate that the elements $G^{(k)}_\lambda[X;t]$ form a basis of $\Lambda^t_{(k)}$.

$s^{(k)}$: (conjecture, discussion of this definition is in [81, Sect. 9.3] but this particular property is not directly addressed; at $t = 1$ this is [92, Property 27]) In Ref. [92] the k-Schur functions $s^{(k)}_\lambda$ are defined as the basis which satisfies the k-Pieri rule of (2.19) and from that definition it is clear that $s^{(k)}_\lambda \in \Lambda_{(k)}$.

The fact that they form a basis follows from a unitriangularity relation with the basis $\{h_\lambda : \lambda_1 \leq k\}$ that follows from the k-Pieri rule.

$\tilde{s}^{(k)}$: (proven [35]) Since $Q'_\lambda[X;t]$ forms a basis of $\Lambda^t_{(k)}$ and the matrix $K^{(k)}_{\lambda\mu}(t)$ is invertible, $\tilde{s}^{(k)}_\lambda[X;t]$ also forms a basis of $\Lambda^t_{(k)}$.

4.6 The k-Rectangle Property

A remarkable property of the k-Schur functions is that it is trivial to multiply any $s^{(k)}_\lambda$ by a k-Schur function indexed by a k-rectangle – any partition of the form $(\ell^{k-\ell+1})$. Precisely, for any k-bounded partition λ and any integer $1 \leq \ell \leq k$,

$$s_{\ell^{k+1-\ell}}\, s^{(k)}_\lambda = s^{(k)}_{\lambda\cup\ell^{k+1-\ell}}\,, \tag{4.4}$$

where $\lambda \cup \nu$ depicts the partition obtained by putting the parts of λ and ν into non-increasing order. In fact, this property has a generic t analog in which the Schur function $s_{\ell^{k+1-\ell}}$ is replaced by the operator $\mathbf{B}_{\ell^{k+1-\ell}}$ defined in Eq. (3.6). Then, given λ and an integer $1 \leq \ell \leq k$,

$$\mathbf{B}_{(\ell^{k-\ell+1})} s^{(k)}_\lambda[X;t] = t^{|\mu|-\ell(\mu)\ell} s^{(k)}_{(\ell^{k-\ell+1})\cup\lambda}[X;t]\,, \tag{4.5}$$

where $\lambda = (\mu,\nu)$ with $\mu_{\ell(\mu)} > \ell \geq \nu_1$. A by-product of this result is that any k-Schur function can be obtained by k-rectangle translation of elements in a distinguished set of $k!$ k-Schur functions. These $k!$ elements are those indexed by irreducible partitions – partitions with at most $k-r$ parts of size r, for $1 \leq r \leq k$. For any k-bounded partition ν, up to a t-factor,

$$s^{(k)}_\nu[X;t] = \mathbf{B}_{R_1}\mathbf{B}_{R_2}\cdots\mathbf{B}_{R_d} s^{(k)}_\lambda[X;t]\,, \tag{4.6}$$

where λ is the irreducible partition obtained by removing k-rectangles $R_1,\ldots,R_d$ from ν.

$A^{(k)}$: (conjecture [91, Conjecture 21]) This is not known and there are no obvious techniques to be tried. One caveat of the atom definition is that the $t=1$ case does not simplify things; the result is also unknown when $t=1$.

$\tilde{A}^{(k)}$: (proven [95, Theorem 26]) This is shown by developing properties of the $\mathbf{B}_\lambda$ operators and the commutation relations with the operator $T^{(k)}_m$.

$s^{(k)}, \tilde{s}^{(k)}$: (conjecture, but proven for $t=1$ in [92, Theorem 40]) When $t=1$, the operator $\mathbf{B}_R$ reduces to multiplication by s_R and it was shown that the linear operation of adding a k-rectangle to the index of a k-Schur function commutes with the Pieri rule from Eq. (2.19).

4.7 When $t = 1$, the Product of k-Schur Functions Is k-Schur Positive

Note that $\Lambda^t_{(k)}$ is not an algebra and the product of two arbitrary k-Schur functions does not remain in the space. However, when $t = 1$ we have $\Lambda^{t=1}_{(k)} = \Lambda_{(k)}$ and as discussed in Sect. 2.2, the space defined in Eq. (2.8) is closed under multiplication. The structure coefficients $c^{\nu(k)}_{\lambda\mu}$ defined by

$$s^{(k)}_\lambda s^{(k)}_\mu = \sum_\nu c^{\nu(k)}_{\lambda\mu} s^{(k)}_\nu \tag{4.7}$$

are non-negative integer coefficients. Recall from Sect. 2.5 that these are now called k-Littlewood-Richardson coefficients and they are a family of constants that includes Gromov-Witten invariants for complete flag varieties, WZW-fusion coefficients, and the structure constants of Schubert polynomials.

$A^{(k)}$: (conjecture [91, Conjecture 39]) Note that the k-Pieri rule is a special case of computing the product of two k-Schur functions. As discussed in Sect. 4.3, the techniques to work with atoms have yet to be developed and even this 'simple' case remains unproven. However, the atom definition is likely to be useful in gaining insight into the combinatorial nature of the structure coefficients.

$\tilde{A}^{(k)}$: (conjecture [94, Conjecture 20]) In this case we have again that the k-Pieri rule at $t = 1$ is a conjecture. It would be sufficient to show that $s^{(k)}_\lambda[X; 1] = \tilde{A}^{(k)}_\lambda[X; 1]$ since the k-Pieri rule characterizes $s^{(k)}_\lambda[X; 1]$ and hence the result is known in this case. It seems that the algebraic definition using operators might be helpful finding a t-analogue of the coefficients $c^{\nu(k)}_{\lambda\mu}$.

$s^{(k)}, \tilde{s}^{(k)}$: (proven [80, Corollary 8.2]) Lam [80] proved that the $s^{(k)}_\lambda[X; 1]$ are isomorphic to the Schubert basis for the homology of the affine Grassmannian. From geometric considerations, it follows that the structure coefficients $c^{\nu(k)}_{\lambda\mu}$ enumerate certain curves in a finite flag variety.

4.8 Positively Closed Under Coproduct

The space $\Lambda^t_{(k)}$ is not an algebra, but since it is linearly spanned by the elements $s_\lambda[X/(1-t)]$ for $\lambda_1 \leq k$, it is a coalgebra under the coproduct defined by

$$\Delta(s_\lambda[X/(1-t)]) = \sum_{\mu,\nu} c^\lambda_{\mu\nu} s_\mu[X/(1-t)] s_\nu[Y/(1-t)] \tag{4.8}$$

(where the $c^{\lambda}_{\mu\nu}$ are the Littlewood–Richardson coefficients). Compare this also with the coproduct (1.28) on Λ. Since if λ is k-bounded then all of the terms μ, ν which appear in this expansion will also be k-bounded, it is conjectured that if the coefficients $C^{\lambda}_{\mu\nu}(t)$ are defined as the coefficients in the expansion

$$\Delta(s^{(k)}_{\lambda}[X;t]) = \sum_{\mu,\nu} C^{\lambda(k)}_{\mu\nu}(t) s^{(k)}_{\mu}[X;t] s^{(k)}_{\nu}[Y;t] \,, \tag{4.9}$$

then the $C^{\lambda}_{\mu\nu}(t)$ are polynomials in t with non-negative integer coefficients.

$A^{(k)}$: (conjecture [91, Conjecture 17])
$\tilde{A}^{(k)}$: (conjecture [94, Conjecture 41])
$s^{(k)}$: (conjecture, proven for $t = 1$ in [80, Corollary 8.1]) Because of the duality of the elements $\tilde{F}^{(k)}_{\mu}$ to the elements $s^{(k)}_{\lambda}$, it follows that at $t = 1$, the basis $\tilde{F}^{(k)}_{\lambda}$ of the space $\Lambda^{(k)}$ multiplies as

$$\tilde{F}^{(k)}_{\mu} \tilde{F}^{(k)}_{\nu} = \sum_{\lambda} C^{\lambda(k)}_{\mu\nu} \tilde{F}^{(k)}_{\lambda} \,,$$

where $C^{\lambda(k)}_{\mu\nu} = C^{\lambda(k)}_{\mu\nu}(1)$.

Although we can say little about the $A^{(k)}$ and $\tilde{A}^{(k)}$ cases, we can determine from the definition of $s^{(k)}_{\lambda}[X;t]$ that

$$s^{(k)}_{\lambda}[X+z;t] = \sum_{r\geq 0} z^r h^{\perp}_r s^{(k)}_{\lambda}[X;t] = \sum_{r\geq 0} z^r C^{\lambda(k)}_{\mu(r)}(t) s^{(k)}_{\mu}[X;t] \,.$$

In this special case the coefficients are

$$C^{\lambda(k)}_{\mu(r)}(t) = \sum_{\kappa^{(*)},c_*} t^{\mathrm{spin}(\kappa^{(*)},c_*)}$$

with the sum is over all strong marked horizontal strips from $\mathfrak{c}(\mu)$ to $\mathfrak{c}(\lambda)$. That is, the sum runs over $\kappa^{(*)}, c_*$ which are $(k+1)$-core tableaux of the form

$$\kappa^{(0)} = \mathfrak{c}(\lambda) \Rightarrow_k \kappa^{(1)} \Rightarrow_k \kappa^{(2)} \Rightarrow_k \cdots \Rightarrow_k \kappa^{(r)} = \mathfrak{c}(\mu) \tag{4.10}$$

and markings $c_1 < c_2 < \cdots < c_r$ where c_i is the content of the lower right hand cell of one of the ribbons of $\kappa^{(i)}/\kappa^{(i-1)}$ and where $\mathrm{spin}(\kappa^{(*)}, c_*)$ is defined, as before, as the sum of the spins of the strong marked ribbons $(\kappa^{(i)}/\kappa^{(i-1)}, c_i)$.

$\tilde{s}^{(k)}$: (conjecture) For $t = 1$, $s^{(k)}_{\lambda} = \tilde{s}^{(k)}_{\lambda}$ and hence also follows from [80, Corollary 8.1].

Sage Example 4.4. Here is a calculation in SAGE where we observe that the coefficients that appear in an example of (4.9) are polynomials in $\mathbb{N}[t]$.

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks3 = Sym.kschur(3)
sage: ks3[3,1].coproduct()
ks3[] # ks3[3, 1] + ks3[1] # ks3[2, 1] + ks3[1, 1] # ks3[2]
 + ks3[2, 1] # ks3[1] + ks3[2] # ks3[1, 1] + ks3[3, 1] # ks3[]
 + (t+1)*ks3[2] # ks3[2] + (t+1)*ks3[1] # ks3[3]
 + (t+1)*ks3[3] # ks3[1]
```

4.9 The Product of a k-Schur and ℓ-Schur Function Is $(k+\ell)$-Schur Positive

Recall that one characterization of $\Lambda^t_{(k)}$ is that it is the linear span of $\{s_\lambda[X/(1-t)]\}_{\lambda_1 \leq k}$. Now if we know that $f \in \Lambda^t_{(k)}$ and $g \in \Lambda^t_{(\ell)}$ then we know by the Littlewood–Richardson rule that fg will be in the linear span of $\{s_\lambda[X/(1-t)]\}_{\lambda_1 \leq k+\ell}$. By the discussion in Sect. 4.5, it is not even clear that the products $s^{(k)}_\lambda[X;t]s^{(\ell)}_\mu[X;t]$ or $A^{(k)}_\lambda[X;t]A^{(\ell)}_\mu[X;t]$ will be in the space $\Lambda^t_{(k+\ell)}$, but it has been proven that $\tilde{A}^{(k)}_\lambda[X;t]\tilde{A}^{(\ell)}_\mu[X;t]$ and $\tilde{s}^{(k)}_\lambda[X;t]\tilde{s}^{(\ell)}_\mu[X;t]$ is an element of $\Lambda^t_{(k+\ell)}$.

Given that the product of a k-Schur function and an ℓ-Schur function is in the linear span of $\Lambda^t_{(k+\ell)}$, it is natural to conjecture that the resulting product will be $(k+\ell)$-Schur positive. We do not know of an attribution for this conjecture but it seems to have been passed around in discussions and talks on the subject.

Sage Example 4.5. We demonstrate an example of this conjecture in SAGE by showing that the product of a 3-Schur function and a 2-Schur function expands positively in terms of 5-Schur functions.

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks2 = Sym.kschur(2)
sage: ks3 = Sym.kschur(3)
sage: ks5 = Sym.kschur(5)
sage: ks5(ks3[2])*ks5(ks2[1])
ks5[2, 1] + ks5[3]
sage: ks5(ks3[2])*ks5(ks2[2,1])
ks5[2, 2, 1] + ks5[3, 1, 1] + (t+1)*ks5[3, 2]
 + (t+1)*ks5[4, 1] + t*ks5[5]
```

4.10 Branching Property from k to $k+1$

One of the properties that is easy to observe when conjecturing the existence of atoms is that the atoms seem to split into smaller pieces as k increases. In the limit (when $k \geq |\lambda|$), we know that $s^{(k)}_\lambda[X;t] = s_\lambda$ (see Sect. 4.1). Although it is clear that $\Lambda^t_{(k)} \subseteq \Lambda^t_{(k+1)}$, it is not easy to prove this branching property.

One reason in particular that this is a difficult property to understand is that both the definition of $A_\lambda^{(k)}$ and $\tilde{A}_\lambda^{(k)}$ involve the operation of the k-split, one in the katabolism procedure, and the other in the k-split basis $G_\lambda^{(k)}[X;t]$. In theory the k-split of a partition λ can be very different than the $(k+1)$-split of the same partition (e.g. consider the 4 and 5 split of $(4,4,4,3,3,3,2,2,1,1)$ which are $((4),(4),(4),(3,3),(3,2),(2,1,1))$ and $((4,4),(4,3),(3,3,2),(2,1,1))$ respectively). A priori we would not expect to see that $A_\lambda^{(k)}[X;t]$ expands positively in $A^{(k+1)}[X;t]$ or $\tilde{A}^{(k)}[X;t]$ expands positively in $\tilde{A}_\lambda^{(k+1)}[X;t]$. However this property was one that was used to conjecture/compute the k-atoms before there was a first formal definition.

$s^{(k)}$: (proven for $t = 1$ in [82], [82, Conjecture 3] is combinatorial formula) At $t = 1$, the proof of the $k \to k+1$ branching for $s_\lambda^{(k)}[X;t]$ follows from the study [82] of a poset on particular partitions called k-shapes. In Sect. 7 we will give some details on the combinatorics behind these results and state the explicit combinatorial formula for this rule in Theorem 7.7. In short, [82] proves that $s_\lambda^{(k)}$ expands positively in the elements $s_\lambda^{(k+1)}$ and gives a conjecture for the expansion of $s_\lambda^{(k)}[X;t]$ in terms of elements of the form $s_\lambda^{(k+1)}[X;t]$. For generic t, the same paper gives a conjecture formula Conjecture 7.8 and discusses the additional properties needed for the result to hold in general. Some progress has been made in this direction in [99].

$\tilde{s}^{(k)}$: (conjecture) At $t = 1$, $s_\lambda^{(k)} = \tilde{s}_\lambda^{(k)}$, hence the result in this case also follows from Ref. [82].

Sage Example 4.6. Here are some examples confirming the branching conjecture (for the implementation in SAGE):

```
sage: Sym = SymmetricFunctions(FractionField(QQ["t"]))
sage: ks3 = Sym.kschur(3)
sage: ks4 = Sym.kschur(4)
sage: ks5 = Sym.kschur(5)
sage: ks4(ks3[3,2,1,1])
ks4[3, 2, 1, 1] + t*ks4[3, 3, 1] + t*ks4[4, 1, 1, 1]
 + t^2*ks4[4, 2, 1]
sage: ks5(ks3[3,2,1,1])
ks5[3, 2, 1, 1] + t*ks5[3, 3, 1] + t*ks5[4, 1, 1, 1]
 + t^2*ks5[4, 2, 1] + t^2*ks5[4, 3] + t^3*ks5[5, 1, 1]
sage: ks5(ks4[3,2,1,1])
ks5[3, 2, 1, 1]
sage: ks5(ks4[4,3,3,2,1,1])
ks5[4, 3, 3, 2, 1, 1] + t*ks5[4, 4, 3, 1, 1, 1]
 + t^2*ks5[5, 3, 3, 1, 1, 1]
sage: ks5(ks4[4,3,3,2,1,1,1])
ks5[4, 3, 3, 2, 1, 1, 1] + t*ks5[4, 3, 3, 3, 1, 1]
 + t*ks5[4, 4, 3, 1, 1, 1, 1] + t^2*ks5[4, 4, 3, 2, 1, 1]
 + t^2*ks5[5, 3, 3, 1, 1, 1, 1] + t^3*ks5[5, 3, 3, 2, 1, 1]
 + t^4*ks5[5, 4, 3, 1, 1, 1]
```

4.11 k-Schur Positivity of Macdonald Symmetric Functions

Even equipped with all these definitions, it has yet to be understood why the Macdonald polynomials expand positively in terms of k-Schur functions. Recall from (1.39) that, for any k-bounded partition μ, the coefficients in

$$H_\mu[X;q,t] = \sum_{\lambda:\lambda_1\le k} K_{\lambda\mu}^{(k)}(q,t)s_\lambda^{(k)}[X;t]$$

are conjectured to be polynomials in q and t with non-negative integer coefficients. An ideal solution to this problem would be to find statistics $a_\mu^{(k)}$ and $b_\mu^{(k)}$ on weak tableaux such that

$$K_{\lambda\mu}^{(k)}(q,t) = \sum_T q^{a_\mu^{(k)}(T)} t^{b_\mu^{(k)}(T)}$$

where the sum is over all standard weak tableaux of shape λ. Section 3.3 discusses partial progress in this direction where such a solution is given for the cases $K_{\lambda\mu}^{(k)}(1,1) = K_{\lambda 1^{|\mu|}}^{(k)}$ and $K_{\lambda\mu}^{(k)}(0,1) = K_{\lambda\mu}^{(k)}$.

$A^{(k)}$: (conjecture [91, Conjecture 8]) This conjecture was the original motivation for studying k-Schur functions; it was a promising attack on a combinatorial interpretation of the Macdonald–Kostka coefficients especially when coupled with the conjecture [91, Eq. (1.15)] that $K_{\lambda\mu}(q,t) - K_{\lambda\mu}^{(k)}(q,t)$ is in $\mathbb{N}[q,t]$. However, a clear combinatorial interpretation of $K_{\lambda\mu}(q,t)$ remains elusive as does even the positivity of the polynomials $K_{\lambda\mu}^{(k)}(q,t)$.

$\tilde{A}^{(k)}$: (conjecture [94, Eq. (1.7)]) A preliminary attack of the $k = 2$ case of this conjecture was considered in [97] and [157] although the complete formulation of the conjecture had not been yet made. Lapointe and Morse together with Lascoux developed the ideas further into k-atoms. Even without knowledge of this conjecture, the latter reference also refers to collections of tableaux as 'atoms' and some of the symmetric functions defined there were actually the 2-atoms.

$s^{(k)}$: (conjecture [96, Eq. (11.6)]) It was because of the characterization of the k-Schur functions as the basis that satisfies the k-Pieri rule of Eq. (2.19) that there is a combinatorial interpretation for $K_{\lambda\mu}^{(k)}(1,1)$ in terms of weak tableaux.

$\tilde{s}^{(k)}$: (conjecture) This is a conjecture, but for $q = 0$, $H_\mu[X;0,t] = Q'_\mu[X;t]$ and the definition [36] of $\tilde{s}_\lambda^{(k)}$ yields that $K_{\lambda\mu}^{(k)}(0,t) = K_{\lambda\mu}^{(k)}(t)$.

Sage Example 4.7. Here are some of the k-analogues of the (q,t)-Macdonald–Kostka coefficients computed in SAGE:

```
sage: Sym = SymmetricFunctions(FractionField(QQ["q,t"]))
sage: H = Sym.macdonald().H()
```

```
sage: ks = Sym.kschur(3)
sage: ks(H[3])
q^3*ks3[1, 1, 1] + (q^2+q)*ks3[2, 1] + ks3[3]
sage: ks(H[3,2])
q^4*ks3[1, 1, 1, 1, 1] + (q^3*t+q^3+q^2)*ks3[2, 1, 1, 1]
 + (q^3*t+q^2*t+q^2+q)*ks3[2, 2, 1]
 + (q^2*t+q*t+q)*ks3[3, 1, 1] + ks3[3, 2]
sage: ks(H[3,1,1])
q^3*ks3[1, 1, 1, 1, 1] + (q^3*t^2+q^2+q)*ks3[2, 1, 1, 1]
 + (q^2*t^2+q^2*t+q*t+q)*ks3[2, 2, 1]
 + (q^2*t^2+q*t^2+1)*ks3[3, 1, 1] + t*ks3[3, 2]
```

5 Directions of Research and Open Problems

In this section we consider further directions of k-Schur research, some in their early stages.

5.1 *A k–Murnaghan-Nakayama Rule*

The Murnaghan–Nakayama rule [118, 126, 128] is a combinatorial formula for the characters $\chi_\lambda(\mu)$ of the symmetric group in terms of ribbon tableaux. Under the Frobenius characteristic map, there exists an analogous statement on the level of symmetric functions, which follows directly from the formula

$$p_r s_\lambda = \sum_\mu (-1)^{\text{height}(\mu/\lambda)} s_\mu . \tag{5.1}$$

Here p_r is the r-th power sum symmetric function, s_λ is the Schur function labeled by partition λ, and the sum is over all partitions $\lambda \subseteq \mu$ for which μ/λ is a border strip of size r. Recall that a border strip is a connected skew shape without any 2×2 squares. The height height(μ/λ) of a border strip μ/λ is one less than the number of rows.

In [7], an analogue of the Murnaghan–Nakayama rule for the product of p_r times $s_\lambda^{(k)}$ is given. This is derived using the k-Pieri rule, and is expressed in terms of the action of the affine symmetric group (resp. nil-Coxeter group) on cores. To give the precise result we need to make a couple of definitions. We define a *vertical domino* in a skew-partition to be a pair of cells in the diagram, with one sitting directly above the other. For the skew of two k-bounded partitions $\lambda \subseteq \mu$ we define the height as

$$\text{height}(\mu/\lambda) = \text{number of vertical dominos in } \mu/\lambda\ . \tag{5.2}$$

For ribbons, that is skew shapes without any 2×2 squares, the definition of height can be restated as the number of occupied rows minus the number of connected components. Notice that this is compatible with the usual definition of the height of a border strip.

Definition 5.1. The skew of two k-bounded partitions, μ/λ, is called a *k-ribbon of size r* if μ and λ satisfy the following properties:

(0) (Containment condition) $\lambda \subseteq \mu$ and $\lambda^{\omega_k} \subseteq \mu^{\omega_k}$;
(1) (Size condition) $|\mu/\lambda| = r$;
(2) (Ribbon condition) $\mathfrak{c}(\mu)/\mathfrak{c}(\lambda)$ is a ribbon;
(3) (Connectedness condition) $\mathfrak{c}(\mu)/\mathfrak{c}(\lambda)$ is k-connected, that is, the contents of $\mathfrak{c}(\mu)/\mathfrak{c}(\lambda)$ form an interval of $[0, k]$ (where 0 is k are adjacent);
(4) (Height statistics condition) $\mathrm{height}(\mu/\lambda) + \mathrm{height}(\mu^{\omega_k}/\lambda^{\omega_k}) = r - 1$.

Then the k-Murnaghan–Nakayama rule states:

Theorem 5.2. *For $1 \le r \le k$ and λ a k-bounded partition, we have*

$$p_r s_\lambda^{(k)} = \sum_\mu (-1)^{\mathrm{height}(\mu/\lambda)} s_\mu^{(k)},$$

where the sum is over all k-bounded partitions μ such that μ/λ is a k-ribbon of size r.

Computer evidence suggests that the ribbon condition (2) of Definition 5.1 might be superfluous because it is implied by the other conditions of the definition. This was checked for $k, r \le 11$ and for all $|\lambda| = n \le 12$ and $|\mu| = n + r$. Also, the k-Murnaghan–Nakayama rule of Theorem 5.2 was only proven for the definition of k-Schur functions $s_\lambda^{(k)}[X; 1]$ and not in terms of $A_\lambda^{(k)}[X; 1]$ or $\tilde{A}_\lambda^{(k)}[X; 1]$.

Note that a Murnaghan–Nakayama rule potentially provides us with a fourth, independent definition of the k-Schur functions in a manner similar to Eq. (2.19). The fault with this approach is that it is not immediately obvious that this system of equations is invertible and consequently defines the elements $s_\lambda^{(k)}$.

An analog of the Murnaghan–Nakayama rule for the elements $\tilde{F}_\lambda^{(k)}[X]$ would give a combinatorial interpretation of the k-Schur functions in the power sum basis at $t = 1$.

Sage Example 5.3. Let us show how to compute the Murnaghan–Nakayama rule for $\tilde{F}_\lambda^{(k)}[X]$. The quotient space $\Lambda^{(k)}$ is implemented in SAGE, but there are several means of computing the coefficients of $p_k \tilde{F}_\lambda^{(k)}[X]$ by duality. As an example, let us compute $p_2 \tilde{F}_{21}^{(3)}[X]$:

```
sage: Sym = SymmetricFunctions(QQ)
sage: Q3 = Sym.kBoundedQuotient(3,t=1)
sage: F = Q3.affineSchur()
sage: p = Sym.power()
```

```
sage: F[2,1]*p[2]
-F3[1, 1, 1, 1, 1] - F3[2, 1, 1, 1] + F3[3, 1, 1] + F3[3, 2]
```

Hence this computation shows that

$$p_2 \tilde{F}^{(3)}_{21} = \tilde{F}^{(3)}_{32} + \tilde{F}^{(3)}_{311} - \tilde{F}^{(3)}_{2111} - \tilde{F}^{(3)}_{11111}.$$

5.2 A Rectangle Generalization at *t* a Root of Unity

Let ζ_m be an mth root of unity (take $\zeta_m = e^{2\pi i/m}$). A result due to Lascoux, Leclerc and Thibon [105] states that if $\lambda = (1^{m_1}, 2^{m_2}, \ldots, d^{m_d})$ and $m_i = q_i m + r_i$ for $0 \leq r_i < m$, then

$$Q'_\lambda[X;\zeta_m] = (Q'_{(1^m)}[X;\zeta_m])^{q_1}(Q'_{(2^m)}[X;\zeta_m])^{q_2} \cdots (Q'_{(d^m)}[X;\zeta_m])^{q_d} Q'_\nu[X;\zeta_m]$$

where $\nu = (1^{r_1}, 2^{r_2}, \ldots, d^{r_d})$. A similar property is shown by Descouens and Morita [39] for the Macdonald symmetric functions. Namely they show

$$H_\lambda[X;q,\zeta_m] = (H_{(1^m)}[X;q,\zeta_m])^{q_1}(H_{(2^m)}[X;q,\zeta_m])^{q_2} \cdots (H_{(d^m)}[X;q,\zeta_m])^{q_d} \\ H_\nu[X;q,\zeta_m]\,.$$

Moreover it is shown in these references that

$$Q'_{(r^m)}[X;\zeta_m] = p_m \circ h_r$$

and

$$H_{(r^m)}[X;q,\zeta_m] = p_m \circ h_r[X/(1-q)] \left(\prod_{i=1}^{r}(1-q^{im})\right)$$

where $\circ$ is the operation of plethysm.

Since at arbitrary t, both of these functions expand positively in k-Schur functions, it is natural to ask if this property is shared by the k-Schur functions themselves. At $t = 1$ the k-Schur functions satisfy (see Sect. 4.6)

$$s^{(k)}_{(\ell^{k-\ell+1})}[X;1] s^{(k)}_\lambda[X;1] = s^{(k)}_{(\ell^{k-\ell+1})\cup\lambda}[X;1]\,. \tag{5.3}$$

At an mth root of unity this property seems to generalize and we conjecture

Conjecture 5.4. *For $\ell \leq k$ and $\zeta_m = e^{2\pi i/m}$*

$$s^{(k)}_{(\ell^{m(k-\ell+1)})}[X;\zeta_m] s^{(k)}_\lambda[X;\zeta_m] = s^{(k)}_{(\ell^{m(k-\ell+1)})\cup\lambda}[X;\zeta_m]$$

and moreover

$$s^{(k)}_{(\ell^{m(k-\ell+1)})}[X;\zeta_m] = p_m \circ s^{(k)}_{(\ell^{k-\ell+1})}[X;1]\,.$$

Sage Example 5.5. We demonstrate an example of this conjecture by building two copies of the k-Schur functions in SAGE, one where the parameter t is specialized to a fourth root of unity, and the other where $t = 1$. Expanding these k-Schur functions in the power sum basis makes it possible to see the relationship between these elements, checking the second relation:

```
sage: R = QQ[I]; z4 = R.zeta(4)
sage: Sym = SymmetricFunctions(R)
sage: ks3z = Sym.kschur(3,t=z4)
sage: ks3 = Sym.kschur(3,t=1)
sage: p = Sym.p()
sage: p(ks3z[2, 2, 2, 2, 2, 2, 2, 2])
1/12*p[4, 4, 4, 4] + 1/4*p[8, 8] - 1/3*p[12, 4]
sage: p(ks3[2,2])
1/12*p[1, 1, 1, 1] + 1/4*p[2, 2] - 1/3*p[3, 1]
sage: p(ks3[2,2]).plethysm(p[4])
1/12*p[4, 4, 4, 4] + 1/4*p[8, 8] - 1/3*p[12, 4]
```

The first relation can be checked as follows:

```
sage: ks3z[3, 3, 3, 3]*ks3z[2, 1]
ks3[3, 3, 3, 3, 2, 1]
```

5.3 A Dual-Basis to $s^{(k)}_{\lambda}[X;t]$

Recall from Sect. 1.7 that $P_\lambda[X;t]$ is the dual basis to $Q'_\lambda[X;t]$ with respect to the $\langle\,.\,,\,.\,\rangle$ scalar product. Moreover, we have the expansion (3.13)

$$Q'_\mu[X;t] = \sum_{\lambda\vdash|\mu|,\lambda_1\le k} K^{(k)}_{\lambda\mu}(t)\tilde{s}^{(k)}_{\lambda}[X;t] \tag{5.4}$$

and

$$\tilde{s}^{(k)}_{\lambda}[X;t] = \sum_{\mu\vdash|\lambda|,\mu_1\le k} K^{(k)}_{\mu\lambda}(t)^{-1} Q'_\mu[X;t]\,, \tag{5.5}$$

where $K^{(k)}_{\lambda\mu}(t)$ t-enumerate weak tableaux of shape $\mathfrak{c}(\lambda)$ and weight μ. A t-generalization for dual k-Schur functions of Eq. (2.17) then comes out of this [36] by duality,

$$\tilde{F}_{\lambda}^{(k)}[X;t] := \sum_{\mu \vdash |\lambda|, \mu_1 \leq k} K_{\lambda\mu}^{(k)}(t) P_{\mu}[X;t] \tag{5.6}$$

These elements clearly live in a space spanned by $\{P_\lambda[X;t]\}_{\lambda_1 \leq k}$. In fact, by triangularity considerations of the symmetric functions $P_\lambda[X;t]$, we have that

$$\Lambda_t^{(k)} := \mathcal{L}\left\{\tilde{F}_{\lambda}^{(k)}[X;t]\right\}_{\lambda_1 \leq k} = \mathcal{L}\{P_\lambda[X;t]\}_{\lambda_1 \leq k} = \mathcal{L}\{s_\lambda[X]\}_{\lambda_1 \leq k} = \mathcal{L}\{m_\lambda[X]\}_{\lambda_1 \leq k} .$$

While this space is not closed under the usual product, it is closed under coproduct.

Now recall from Sect. 1.7, that $\langle Q'_\lambda[X;t], P_\mu[X;t]\rangle = \delta_{\lambda\mu}$. Hence

$$\begin{aligned} \left\langle s_{\mu}^{(k)}[X;t], \tilde{F}_{\lambda}^{(k)}[X;t]\right\rangle &= \sum_{\substack{\gamma \vdash |\mu| \\ \gamma_1 \leq k}} K_{\gamma\mu}^{(k)}(t)^{-1} \left\langle Q'_{\gamma}[X;t], \tilde{F}_{\lambda}^{(k)}[X;t]\right\rangle \\ &= \sum_{\substack{\gamma \vdash |\mu| \\ \gamma_1 \leq k}} K_{\gamma\mu}^{(k)}(t)^{-1} K_{\lambda\gamma}^{(k)}(t) = \delta_{\lambda\mu} . \end{aligned}$$

Therefore we can see that the elements $\left\{\tilde{F}_{\lambda}^{(k)}[X;t]\right\}_{\lambda_1 \leq k}$ are another t-analogue of the Schur functions which, by triangularity considerations, live in the linear span of $\mathcal{L}\{m_\lambda : \lambda_1 \leq k\}$ and are dual to $\left\{s_{\lambda}^{(k)}[X;t]\right\}_{\lambda_1 \leq k}$ with respect to the usual scalar product.

Just as with the space $\Lambda^t_{(k)}$, the linear span of the dual elements for the k-Schur functions is a subspace and not an algebra with respect to the usual product. We can however make it an algebra, by introducing a product

$$\tilde{F}_{\nu}^{(k)}[X;t] \cdot^t \tilde{F}_{\mu}^{(k)}[X;t] := \sum_{\lambda} C_{\nu\mu}^{\lambda(k)}(t) \tilde{F}_{\lambda}^{(k)}[X;t] , \tag{5.7}$$

where the coefficients $C_{\nu\mu}^{\lambda(k)}(t)$ are precisely those defined by Eq. (4.9). The coefficients $C_{\mu\nu}^{\lambda(k)}(t)$ are discussed in Sect. 4.8 and they are conjectured to be polynomials in t with non-negative integer coefficients. To be clear, we take as definition that the space is closed under a product where the structure coefficients are

$$C_{\nu\mu}^{\lambda(k)}(t) = \left\langle \Delta(s_{\lambda}^{(k)}[X;t]), \tilde{F}_{\nu}^{(k)}[X;t] \tilde{F}_{\mu}^{(k)}[Y;t]\right\rangle . \tag{5.8}$$

There are several ways of computing a more explicit formula for these coefficients, but let us consider one that can be found by expanding the elements $s_{\lambda}^{(k)}[X;t]$ and $\tilde{F}_{\lambda}^{(k)}[X;t]$ in the Schur basis. Since

$$s_\lambda^{(k)}[X;t] = \sum_{\mu \vdash |\lambda|} \left\langle s_\lambda^{(k)}[X;t], s_\mu[X] \right\rangle s_\mu[X] \tag{5.9}$$

and

$$\tilde{F}_\lambda^{(k)}[X;t] = \sum_{\mu \vdash |\lambda|} \left\langle \tilde{F}_\lambda^{(k)}[X;t], s_\mu[X] \right\rangle s_\mu[X] \ , \tag{5.10}$$

a formula for these coefficients is found by combining (5.8)–(5.10) to obtain

$$C_{\nu\mu}^{\lambda(k)}(t) = \sum_{\substack{\theta \vdash |\lambda| \\ \tau \vdash |\nu|, \gamma \vdash |\mu|}} c_{\tau\gamma}^{\theta} \left\langle s_\lambda^{(k)}[X;t], s_\theta[X] \right\rangle \left\langle \tilde{F}_\nu^{(k)}[X;t], s_\tau[X] \right\rangle \left\langle \tilde{F}_\mu^{(k)}[X;t], s_\gamma[X] \right\rangle \tag{5.11}$$

where the $c_{\mu\nu}^{\lambda}$ are the Littlewood–Richardson coefficients previously discussed.

Little is known about this product and it would be useful to understand it in terms of another basis. However, it is possible to compute these coefficients as elements of a quotient algebra or, as we do here, define a projection operator and notice that the product $\cdot^t$ is simply the usual product followed by a projection into the space $\Lambda_t^{(k)}$.

Proposition 5.6. *Let $\Theta^{(k)}$ be a projection from Λ to $\Lambda_t^{(k)}$, the space spanned by functions dual to the k-Schur functions, defined by $\Theta^{(k)}(P_\lambda[X;t]) = P_\lambda[X;t]$ if $\lambda_1 \leq k$, and $\Theta^{(k)}(P_\lambda[X;t]) = 0$ if $\lambda_1 > k$, then*

$$\tilde{F}_\nu^{(k)}[X;t] \cdot^t \tilde{F}_\mu^{(k)}[X;t] = \Theta^{(k)}(\tilde{F}_\nu^{(k)}[X;t]\tilde{F}_\mu^{(k)}[X;t]) \ . \tag{5.12}$$

Proof. In order to prove this we need to show that the coefficient of $\tilde{F}_\lambda^{(k)}[X;t]$ in the expression $\Theta^{(k)}(\tilde{F}_\nu^{(k)}[X;t]\tilde{F}_\mu^{(k)}[X;t])$ is equal to $C_{\nu\mu}^{\lambda(k)}(t)$. Since $s_\lambda^{(k)}[X;t]$ is in the linear span of elements $\{Q'_\lambda[X;t]\}_{\lambda_1 \leq k}$, we can conclude that the coefficient of $\tilde{F}_\lambda^{(k)}[X;t]$ in $\Theta^{(k)}(\tilde{F}_\nu^{(k)}[X;t]\tilde{F}_\mu^{(k)}[X;t])$ is equal to

$$\left\langle \Theta^{(k)}(\tilde{F}_\nu^{(k)}[X;t]\tilde{F}_\mu^{(k)}[X;t]), s_\lambda^{(k)}[X;t] \right\rangle = \left\langle \tilde{F}_\nu^{(k)}[X;t]\tilde{F}_\mu^{(k)}[X;t], s_\lambda^{(k)}[X;t] \right\rangle . \tag{5.13}$$

We can use Eqs. (5.9) and (5.10) to expand the right hand side so that it is equal to

$$\sum_{\substack{\theta \vdash |\lambda| \\ \tau \vdash |\nu|, \gamma \vdash |\mu|}} \left\langle s_\tau s_\gamma, s_\theta \right\rangle \left\langle \tilde{F}_\nu^{(k)}[X;t], s_\tau[X] \right\rangle \left\langle \tilde{F}_\mu^{(k)}[X;t], s_\gamma[X] \right\rangle \left\langle s_\lambda^{(k)}[X;t], s_\theta[X] \right\rangle$$

$$= \sum_{\substack{\theta \vdash |\lambda| \\ \tau \vdash |\nu|, \gamma \vdash |\mu|}} c_{\tau\gamma}^{\theta} \left\langle \tilde{F}_\nu^{(k)}[X;t], s_\tau[X] \right\rangle \left\langle \tilde{F}_\mu^{(k)}[X;t], s_\gamma[X] \right\rangle \left\langle s_\lambda^{(k)}[X;t], s_\theta[X] \right\rangle = C_{\nu\mu}^{\lambda(k)}(t)$$

by (5.11). This shows that the coefficients that appear in the t-product in Eq. (5.7) are the same that appear by usual multiplication followed by a projection by $\Theta^{(k)}$. □

Sage Example 5.7. We can compute the coefficients $C^{\nu(k)}_{\lambda\mu}(t)$ as structure coefficients of the dual basis elements $\tilde{F}^{(k)}_{\lambda}[X;t]$. These elements are implemented in SAGE in a space representing the quotient of the ring of symmetric functions Λ by the ideal generated by the Hall-Littlewood symmetric functions $P_{\lambda}[X;t]$ with $\lambda_1 > k$.

```
sage: Sym = SymmetricFunctions(QQ["t"].fraction_field())
sage: Q3 = Sym.kBoundedQuotient(3)
sage: dks = Q3.dual_k_Schur()
sage: dks[2, 1, 1]*dks[3, 2, 1]
(t^7+t^6)*dks3[2, 1, 1, 1, 1, 1, 1, 1, 1]
 + (t^4+t^3+t^2)*dks3[2, 2, 2, 1, 1, 1, 1]
 + (t^3+t^2)*dks3[2, 2, 2, 2, 1, 1]
 + (t^5+2*t^4+2*t^3+t^2)*dks3[2, 2, 2, 2, 2]
 + (t^5+2*t^4+t^3)*dks3[3, 1, 1, 1, 1, 1, 1, 1]
 + (2*t^5+3*t^4+4*t^3+3*t^2+t)*dks3[3, 2, 1, 1, 1, 1, 1]
 + (2*t^2+t+1)*dks3[3, 2, 2, 1, 1, 1]
 + (t^4+3*t^3+4*t^2+3*t+1)*dks3[3, 2, 2, 2, 1]
 + (t^5+t^4+4*t^3+4*t^2+3*t+1)*dks3[3, 3, 1, 1, 1, 1]
 + (2*t^5+3*t^4+5*t^3+6*t^2+4*t+2)*dks3[3, 3, 2, 1, 1]
 + (t^4+t^3+3*t^2+2*t+1)*dks3[3, 3, 2, 2]
 + (t^5+3*t^4+3*t^3+4*t^2+2*t+1)*dks3[3, 3, 3, 1]
```

5.4 A Product on $\Lambda^{t}_{(k)}$

What is interesting about the $\tilde{F}^{(k)}_{\lambda}[X;t]$ elements is that they are clearly closed under the usual coproduct operation of the symmetric functions. It is therefore natural to consider the coefficients that appear in the coproduct of the elements which are dual to the k-Schur functions as the structure constants of the product of k-Schur functions.

Define the coefficients $c^{\lambda(k)}_{\nu\mu}(t)$ by the coproduct formula

$$\Delta(\tilde{F}^{(k)}_{\lambda}[X;t]) = \sum_{\nu,\mu} c^{\lambda(k)}_{\nu\mu}(t)\tilde{F}^{(k)}_{\nu}[X;t]\tilde{F}^{(k)}_{\mu}[Y;t]\,.$$

We will give a more precise calculation of these coefficients below, but assuming that they exist, we then define

$$s^{(k)}_{\nu}[X;t] \cdot_t s^{(k)}_{\mu}[X;t] := \sum_{\lambda\vdash|\nu|+|\mu|} c^{\lambda(k)}_{\nu\mu}(t) s^{(k)}_{\lambda}[X;t] \tag{5.14}$$

where

$$c_{\nu\mu}^{\lambda(k)}(t) := \left\langle \Delta(\tilde{F}_\lambda^{(k)}[X;t]), s_\nu^{(k)}[X;t] s_\mu^{(k)}[Y;t] \right\rangle .$$

We can then derive from Eqs. (5.9) and (5.10), that

$$c_{\nu\mu}^{\lambda(k)}(t) = \sum_{\substack{\theta \vdash |\lambda| \\ \tau \vdash |\nu|, \gamma \vdash |\mu|}} c_{\tau\gamma}^{\theta} \left\langle \tilde{F}_\lambda^{(k)}[X;t], s_\theta[X] \right\rangle \left\langle s_\nu^{(k)}[X;t], s_\tau[X] \right\rangle \left\langle s_\mu^{(k)}[X;t], s_\gamma[X] \right\rangle . \tag{5.15}$$

At this point it is possible to see that while it is not obvious what the product structure on the k-Schur functions should be, if we take this to be the definition then, as in the case with the product $\cdot^t$, the product on k-Schur functions can also be realized as a projection of the usual product.

Proposition 5.8. *Let $\Theta_{(k)}$ be a projection from the space Λ to the space $\Lambda_{(k)}^t$ spanned by the k-Schur functions defined by $\Theta_{(k)}(Q'_\lambda[X;t]) = Q'_\lambda[X;t]$ if $\lambda_1 \leq k$ and $\Theta_{(k)}(Q'_\lambda[X;t]) = 0$, then*

$$s_\nu^{(k)}[X;t] \cdot_t s_\mu^{(k)}[X;t] = \Theta_{(k)}(s_\nu^{(k)}[X;t] s_\mu^{(k)}[X;t]) .$$

Proof. The proof proceeds exactly as it did in Proposition 5.6. We compute that the coefficient of $s_\lambda^{(k)}[X;t]$ in $\Theta_{(k)}(s_\nu^{(k)}[X;t] s_\mu^{(k)}[X;t])$ is

$$\left\langle \Theta_{(k)}(s_\nu^{(k)}[X;t] s_\mu^{(k)}[X;t]), \tilde{F}_\lambda^{(k)}[X;t] \right\rangle = \left\langle s_\nu^{(k)}[X;t] s_\mu^{(k)}[X;t], \tilde{F}_\lambda^{(k)}[X;t] \right\rangle$$
$$= \sum_{\substack{\theta \vdash |\lambda| \\ \tau \vdash |\nu|, \gamma \vdash |\mu|}} c_{\tau\gamma}^{\theta} \left\langle s_\nu^{(k)}[X;t], s_\tau[X] \right\rangle \left\langle s_\mu^{(k)}[X;t], s_\gamma[X] \right\rangle \left\langle \tilde{F}_\lambda^{(k)}[X;t], s_\theta[X] \right\rangle = c_{\nu\mu}^{\lambda(k)}(t) .$$

So we see again that the structure coefficients in the definition of the t-product in Eq. (5.14) are exactly those that occur by taking the usual product and then projecting using the map $\Theta_{(k)}$. □

It was discussed in Sect. 4.7 that the coefficients $c_{\mu\nu}^{\lambda(k)}(1)$ are non-negative integers (conjecturally for certain definitions), but in the following example we will see that $c_{\mu\nu}^{\lambda(k)}(t)$ are not generally elements of $\mathbb{N}[t]$. This makes us believe that this product is not 'the' product to define on the space $\Lambda_{(k)}^t$ (if there is such a product). At least in certain cases, the operators $\mathbf{B}_\lambda$ from Eq. (3.7) also provide a means of defining a t-analogue of multiplication and in light of Eq. (4.1), it seems possible that there is some other t-product on $\Lambda_{(k)}^t$ which is the right one to consider on this space.

Sage Example 5.9. We show how a product of k-Schur functions can be computed under the projection $\Theta_{(k)}$. The product is first computed in the $Q'_\mu[X;t]$ basis and then the projection is computed by restricting the support to those partitions whose parts are less than or equal to k, which is 2 in this example.

```
sage: ks2 = SymmetricFunctions(QQ["t"]).kschur(2)
sage: HLQp = SymmetricFunctions(QQ["t"]).hall_littlewood().Qp()
sage: ks2((HLQp(ks2[1,1])*HLQp(ks2[1])).restrict_parts(2))
ks2[1, 1, 1] + (-t+1)*ks2[2, 1]
```

The coefficient $t-1$ should appear as the coefficient of $\tilde{F}_1^{(2)} \otimes \tilde{F}_{11}^{(2)}$ in the expansion of the coproduct $\Delta(\tilde{F}_{21}^{(2)})$. We can calculate this using SAGE using the following commands.

```
sage: Sym = SymmetricFunctions(QQ["t"])
sage: dks = Sym.kBoundedQuotient(2).dks()
sage: dks[2,1].coproduct()
dks2[] # dks2[2, 1] + (-t+1)*dks2[1] # dks2[1, 1]
 + dks2[1] # dks2[2] + (-t+1)*dks2[1, 1] # dks2[1]
 + dks2[2] # dks2[1] + dks2[2, 1] # dks2[]
```

5.5 A Representation Theoretic Model of *k*-Schur Functions

The *Frobenius map* is a map from S_m-modules to symmetric functions of degree m which takes an irreducible module indexed by the partition λ to the Schur function also indexed by the partition $\lambda \vdash m$. Here we denote this map by $\mathcal{F}$ with $\mathcal{F}(V_\lambda) = s_\lambda$, where V_λ is an irreducible S_m-module.

The parameter t in the ring of symmetric functions represents a grading and we can define for a graded module $V = \bigoplus_{d\geq 0} V_d$,

$$\mathcal{F}_t(V) = \sum_{d\geq 0} t^d \mathcal{F}(V_d) \,.$$

For example, if we consider the ring of polynomials $\mathbb{C}[a_1, a_2, \ldots, a_m]$ as a module graded by the degree in the variables a_i, then $\mathcal{F}_t(\mathbb{C}[a_1, a_2, \ldots, a_m]) = h_m\left[\frac{X}{1-t}\right]$. In particular, we also have for any irreducible S_m-module V_λ that the tensor product $V_\lambda \otimes \mathbb{C}[a_1, a_2, \ldots, a_m]$ is an S_m-module, where S_m acts diagonally on the tensors and

$$\mathcal{F}_t(V_\lambda \otimes \mathbb{C}[a_1, a_2, \ldots, a_m]) = s_\lambda\left[\frac{X}{1-t}\right] .$$

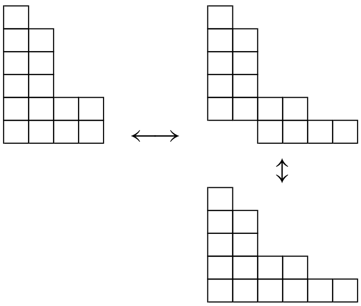
Mark Haiman and Li-Chung Chen [31] defined $\mathcal{M}^{(k)}$ to be the category of graded finitely generated $\mathbb{C}[a_1, a_2, \ldots, a_m] * S_m$-modules V such that V has an S_m-equivariant $\mathbb{C}[a_1, a_2, \ldots, a_m]$ free resolution using only the $V_\lambda \otimes \mathbb{C}[a_1, a_2, \ldots, a_m]$ with $\lambda_1 \leq k$. Then they conjecture the following.

Conjecture 5.10. *The modules which are irreducible in $\mathcal{M}^{(k)}$ have images under the map $\mathcal{F}_t$ which are equal to $\omega s_\lambda^{(k)}[X;t]$ for $\lambda \vdash m$ and $\lambda_1 \leq k$.*

We say that two partitions λ and μ of the same size are called *skew-linked* if there exists a skew partition γ/τ such that λ_i is the number of cells in the ith row of γ/τ and μ_i' is the number of cells in the ith column of γ/τ. The definition of skew-linked is not associated with a particular value of k, but we have seen examples of partitions which are skew-linked through the $(k+1)$-cores. We always have that λ and $(\lambda^{\omega_k})'$ are skew-linked since the skew partition representing the cells of $\mathfrak{c}_k(\lambda)$ with hook less than $k+1$ has λ_i cells in row i and the transpose of $\mathfrak{c}_k(\lambda)$ has $\lambda_i^{\omega_k}$ cells with hook less than $k+1$ in row i. But there are other examples of pairs of partitions which are skew-linked. Of course, if λ and μ are skew-linked, then μ' and λ' are skew-linked.

Example 5.11. For a small example, consider that the partition $(2,2,1)$ is skew-linked to $(2,2,1)$, $(3,2)$, $(4,1)$ and (5) through the skew partitions $(2,2,1)$, $(3,2,1)/(1)$, $(4,2,1)/(2)$, and $(5,3,1)/(3,1)$ respectively.

For a larger example, the partition $(4,4,2,2,2,1)$ is skew linked to $(6,4,2,2,1)$ because the skew partition $(6,4,2,2,2,1)/(2)$ has rows given by $(4,4,2,2,2,1)$ and columns given by the partition $(5,4,2,2,1,1)$.



Notice that $(6,4,2,2,1)$ is not the conjugate of the k-conjugate of $(4,4,2,2,2,1)$ for any k.

Theorem 5.12 (L.-C. Chen). *If λ is skew-linked to μ by a skew partition γ/τ with $\tau \vdash d$, then V_λ occurs with multiplicity 1 at degree d in $V_\mu \otimes \mathbb{C}[a_1, a_2, \ldots, a_m]$ and does not appear at a lower degree.*

As a corollary they construct a module by moding out the module $V_\mu \otimes \mathbb{C}[a_1, a_2, \ldots, a_m]$ by the space generated by all modules V_γ which are at degree d which are not V_λ. In the particular case when λ is a k-bounded partition, λ' is skew-linked to $\mu = \lambda^{\omega_k}$ and the module V_λ is conjectured to have Frobenius image equal to $\omega s_\lambda^{(k)}[X;t]$.

Conjecture 5.13. *For a k-bounded partition λ, let d be the number of cells in $\mathfrak{c}_k(\lambda)$ which have a hook length greater than k. Let W be the module which is generated* *by $V_{\lambda^{\omega_k}}$ as an $S_m * \mathbb{C}[a_1, a_2, \ldots, a_m]$-module and cogenerated by the copy of $V_{\lambda'}$ of degree d. Then $\mathcal{F}_t(W) = \omega s_\lambda^{(k)}[X;t]$.*

Haiman and Li-Chung Chen [31] note that this module is only conjectured to lie within the category of $\mathcal{M}^{(k)}$.

5.6 From Pieri to K-Theoretic k-Schur Functions

As mentioned in the introduction, a trend in Schubert calculus is to generalize the classical setup. The replacement of cohomology by K-theory is a particularly fruitful variation. Lascoux and Schützenberger introduced the Grothendieck polynomials in [110] as representatives for the K-theory classes determined by structure sheaves of Schubert varieties. Grothendieck polynomials have since been connected to representation theory and algebraic geometry and combinatorics is again at the forefront (e.g. [37, 45, 76, 101]). For example, the stable Grothendieck polynomials G_λ are inhomogeneous symmetric polynomials whose lowest homogeneous degree component is a Schur function. Buch proved in [27] that they are the weight generating functions

$$G_\lambda = \sum_{\substack{T \text{ set-valued} \\ \text{shape}(T)=\lambda}} (-1)^{|\lambda|-|\text{weight}(T)|}\, x^{\text{weight}(T)}\,, \tag{5.16}$$

of tableaux called set-valued tableaux. Such a tableau T is a filling of each cell in a shape with a set of integers, where a set X below (west of) Y satisfies $\max X <$ $(\leq\ )\min Y$. The weight of T is α where α_i is the number of cells in T containing an i. Pieri rules are given in [113] in terms of binomial numbers and a generalization for Yamanouchi tableaux gives [27] a combinatorial rule for the structure constants.

Ideas in k-Schur theory extend to the inhomogeneous setting providing combinatorial tools that apply to torus-equivariant K-theory of the affine Grassmannian of SL_{k+1}. Similar to the development described in Sect. 2.2, a close study of a Pieri rule and its iteration is carried out in [123], leading to a family of affine set-valued tableaux that are in bijection with elements of the affine nil-Hecke algebra. These simultaneously generalize set-valued tableaux *and* weak k-tableaux.

These tableaux are defined along the lines described in Remark 2.3; the semi-standard case is given by putting conditions on the reading words of the standard case. Recall from Sect. 3.3 that $T_{\leq x}$ is the subtableau obtained by deleting all letters larger than x from T and note that this is well-defined for a set-valued tableau T. A standard *affine set-valued tableau* T of degree n is then defined as a set-valued filling such that, for each $1 \leq x \leq n$, shape$(T_{\leq x})$ is a core and the cells containing an x form the set of all removable corners of $T_{\leq x}$ with the same residue.

Example 5.14. With $k = 2$, the standard affine set-valued tableaux of degree 5 with shape $\mathfrak{c}(2,1,1) = (3,1,1)$ are

$$
\begin{array}{|c|cc}\cline{1-1} \{3,4\}_1 & & \\ \cline{1-1} \{2\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{3,4\}_1} & \multicolumn{1}{c|}{\{5\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{3,5\}_1 & & \\ \cline{1-1} \{2\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{3\}_1} & \multicolumn{1}{c|}{\{4\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{5\}_1 & & \\ \cline{1-1} \{4\}_2 & & \\ \hline \{1,2\}_0 & \multicolumn{1}{|c|}{\{3\}_1} & \multicolumn{1}{c|}{\{4\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{4\}_1 & & \\ \cline{1-1} \{3\}_2 & & \\ \hline \{1,2\}_0 & \multicolumn{1}{|c|}{\{4\}_1} & \multicolumn{1}{c|}{\{5\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{4\}_1 & & \\ \cline{1-1} \{2,3\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{4\}_1} & \multicolumn{1}{c|}{\{5\}_2} \\ \hline \end{array}
\tag{5.17}
$$

$$
\begin{array}{|c|cc}\cline{1-1} \{5\}_1 & & \\ \cline{1-1} \{4\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{2,3\}_1} & \multicolumn{1}{c|}{\{4\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{5\}_1 & & \\ \cline{1-1} \{3,4\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{2\}_1} & \multicolumn{1}{c|}{\{3,4\}_2} \\ \hline \end{array}
\quad
\begin{array}{|c|cc}\cline{1-1} \{4\}_1 & & \\ \cline{1-1} \{3\}_2 & & \\ \hline \{1\}_0 & \multicolumn{1}{|c|}{\{2\}_1} & \multicolumn{1}{c|}{\{3,5\}_2} \\ \hline \end{array}
\,. \tag{5.18}
$$

For the semi-standard case, first note that a set-valued tableau T of weight α is a standard set-valued tableau with increasing reading words in the alphabets $\mathcal{A}_{\alpha,x}$ of (2.5), where the reading word is obtained by reading letters from a cell in decreasing order (and as usual, cells are taken from top to bottom and left to right). Since letters in standard affine set-valued tableaux can occur with multiplicity, the *lowest reading word* in $\mathcal{A}$ – reading the lowest occurrence of the letters in $\mathcal{A}$ from top to bottom and left to right – is used. Again, letters in the same cell are read in decreasing order. In Example 5.14, the lowest reading words in $\{1, \ldots, 5\}$ are $21435, 52134, 52134, 32145, 32145, 51324, 51243, 41253$.

The affine K-theoretic generalization of a tableau with weight α is then given, for any k-bounded composition α, by a standard affine set-valued tableau of degree $|\alpha|$ where, for each $1 \leq x \leq \ell(\alpha)$,

1. The lowest reading word in $\mathcal{A}_{\alpha,x}$ is increasing
2. The letters of $\mathcal{A}_{\alpha,x}$ occupy α_x distinct residues
3. The letters of $\mathcal{A}_{\alpha,x}$ form a horizontal strip.

Example 5.15. The affine set-valued tableaux in (5.18) of Example 5.14 all have weight $(2, 1, 1, 1)$ and shape $(3, 1, 1) = \mathfrak{c}(2, 1, 1)$, for $k = 2$.

It is proven in [123] that the weight generating functions of affine set-valued tableaux

$$
G_\lambda^{(k)} = \sum_{\substack{T \text{ affine set-valued tableau} \\ \text{shape}(T) = \mathfrak{c}(\lambda)}} (-1)^{|\lambda| + |\text{weight}(T)|} \, x^{\text{weight}(T)} \tag{5.19}
$$

are Schubert representatives for K-theory of the affine Grassmannian of SL_{k+1} called affine stable Grothendieck polynomials [79, 84]. These reduce to Grothendieck polynomials for large k and the term of lowest degree in $G_\lambda^{(k)}$ is the dual k-Schur function $\tilde{F}_\lambda^{(k)}$. As in the k-Schur set-up, these do not form a self-dual basis. Instead, the dual to $\{G_\lambda^{(k)}\}_{\lambda_1 \leq k}$ is an inhomogeneous basis $\{g_\lambda^{(k)}\}_{\lambda_1 \leq k}$ for $\Lambda_{(k)}$.

The set-up discussed in Sects. 2.1 and 2.2 is extended in [123] and gives affine K-theoretic properties for this basis such as Pieri rules and an analog to property (4.2) for an inhomogeneous involution Ω;

$$\Omega g_{\lambda}^{(k)} = g_{\lambda^{\omega_k}}^{(k)} . \tag{5.20}$$

Among some of the open combinatorial problems, it remains to develop an affine K-theoretic set-up in the dual world along the lines discussed in Sect. 2.4; finding the Pieri rules for $\{G_{\lambda}^{(k)}\}_{\lambda_1 \le k}$ and giving a weight-generating formulation for $\{g_{\lambda}^{(k)}\}_{\lambda_1 \le k}$. In addition, there are properties of Grothendieck polynomials that conjecturally extend to these new bases. For example, there is a combinatorial expansion of Grothendieck polynomials into Schur functions described by a family of skew tableaux [113]. It is conjectured that $G_{\lambda}^{(k)}$ and $g_{\lambda}^{(k)}$ have combinatorial expansions in terms of dual k-Schur functions and k-Schur functions, respectively. Noncommutative versions of the affine stable Grothendieck polynomials and $\{g_{\lambda}^{(k)}\}_{\lambda_1 \le k}$ are given in [84], where geometric aspects of these bases are also explored. Further conjectures relating to these bases can be found in [84, 123].

Sage Example 5.16. The basis $\{g_{\lambda}^{(k)}\}_{\lambda_1 \le k}$ has been implemented in SAGE and so it is possible to begin experimenting with these combinatorial problems.

```
sage: Sym = SymmetricFunctions(QQ)
sage: Sym3 = Sym.kBoundedSubspace(3,t=1)
sage: Kks3 = Sym3.K_kschur()
sage: s = Sym.s()
sage: m = Sym.m()
sage: s(Kks3[3,1])
s[3] + s[3, 1] + s[4]
sage: m(Kks3[3,1])
m[1, 1, 1] + 4*m[1, 1, 1, 1] + m[2, 1]
 + 3*m[2, 1, 1] + 2*m[2, 2] + m[3] + 2*m[3, 1] + m[4]
sage: ks3 = Sym3.kschur()
sage: ks3(Kks3[3,1])
ks3[3] + ks3[3, 1]
sage: Kks3[3,1]*Kks3[2]
-Kks3[3, 1, 1] - Kks3[3, 2] + Kks3[3, 2, 1] + Kks3[3, 3]
sage: Kks3[3,1].coproduct()
Kks3[] # Kks3[3, 1] - Kks3[1] # Kks3[2] + Kks3[1] # Kks3[2, 1]
 + 2*Kks3[1] # Kks3[3] + Kks3[1, 1] # Kks3[2]
 - Kks3[2] # Kks3[1] + Kks3[2] # Kks3[1, 1]
 + 2*Kks3[2] # Kks3[2] + Kks3[2, 1] # Kks3[1]
 + 2*Kks3[3] # Kks3[1] + Kks3[3, 1] # Kks3[]
```

Since the elements $\{G_{\lambda}^{(k)}\}_{\lambda_1 \le k}$ are an infinite sum of elements of degree greater than or equal to $|\lambda|$ and we do not know their algebra structure, we cannot currently represent them as we do other bases in SAGE. However, a preliminary implementation of this basis does exist that allows one to compute the elements up to a given degree.

```
sage: SymQ3 = Sym.kBoundedQuotient(3,t=1)
sage: G1 = SymQ3.AffineGrothendieckPolynomial([1],6)
sage: G2 = SymQ3.AffineGrothendieckPolynomial([2],6)
sage: (G1*G2).lift().scalar(Kks3[3,1])
-1
```

Notice how the coproduct applied to $g^{(3)}_{(3,1)}$ agrees with the product structure in this one calculation since we see that coefficient of $G^{(3)}_{(3,1)}$ in the product of $G^{(3)}_{(2)}$ and $G^{(3)}_{(1)}$ is equal to the coefficient of $g^{(3)}_{(2)} \otimes g^{(3)}_{(1)}$ in $\Delta(g^{(3)}_{(3,1)})$.

We may also list all the terms which appear in a product of elements of $\{G^{(k)}_\lambda\}_{\lambda_1 \leq k}$. This is sufficient for generating data for a Pieri rule on $\{G^{(k)}_\lambda\}_{\lambda_1 \leq k}$ since we can compute products and use the duality with the $\{g^{(k)}_\lambda\}_{\lambda_1 \leq k}$ basis to determine the structure coefficients up to a degree higher than what appears in our product.

```
sage: G31 = SymQ3.AffineGrothendieckPolynomial([3,1], 8)
sage: for d in range(5,10):
....:     for la in Partitions(d,max_part=3):
....:         c = (G1*G31).lift().scalar(Kks3(la))
....:         if c!=0:
....:             print la, c
....:
[3, 2] 2
[3, 1, 1] 1
[3, 3] -1
[3, 2, 1] -2
[3, 3, 1] 1
[3, 2, 1, 1] 1
[3, 3, 1, 1] -1
```

This SAGE calculation shows that no terms indexed by partitions of size 9 appear in the product of $G^{(3)}_{(3,1)}$ and $G^{(3)}_{(1)}$, and since we believe that this indicates highest degree of terms which will appear in our product will be 8, then

$$G^{(3)}_{(3,1)}G^{(3)}_{(1)} = 2G^{(3)}_{(3,2)} + G^{(3)}_{(3,1,1)} - G^{(3)}_{(3,3)} - 2G^{(3)}_{(3,2,1)} + G^{(3)}_{(3,3,1)} + G^{(3)}_{(3,2,1,1)} - G^{(3)}_{(3,3,1,1)} \,.$$

6 Duality Between the Weak and Strong Orders

In this section we consider a k-analogue of the Cauchy identity and the Robinson–Schensted–Knuth (RSK) algorithm (or insertion algorithm). The RSK algorithm provides a bijection between permutations and pairs of tableaux satisfying certain conditions. As shown in [81] this can be generalized to the affine setting.

6.1 *k*-Analogue of the Cauchy Identity

In the algebra of symmetric functions (or rather polynomials) an important identity is the *Cauchy identity*, stating that

$$\prod_{i,j=1}^{m} \frac{1}{1-x_i y_j} = \prod_{j=1}^{m} \sum_{r\geq 0} h_r[X_m] y_j^r = \sum_{\lambda} h_\lambda[X_m] m_\lambda[Y_m] = \sum_{\lambda} s_\lambda[X_m] s_\lambda[Y_m] \,, \tag{6.1}$$

where the last two sums run over all partitions λ.

Although there is an algebraic proof of this identity that follows from calculations in Sect. 1.5, there is also a direct combinatorial proof of this result. Recall from Eq. (2.7) that the Schur function is equal to

$$s_\lambda[X_m] = \sum_{T} \mathbf{x}^T \,, \tag{6.2}$$

where the sum is over all semi-standard tableaux of shape λ and $\mathbf{x}^T$ is a monomial which represents the product over i of x_i raised to the number of i in the tableaux.

The Schur functions are in fact characterized as the unique basis which satisfies Eq. (6.1) and which is triangularly related to the monomial symmetric functions. Notice that

$$\prod_{i,j=1}^{m} \frac{1}{1-x_i y_j} = \sum_{M} \prod_{i,j=1}^{m} (x_i y_j)^{m_{ij}} = \sum_{M} (\mathbf{xy})^M \,, \tag{6.3}$$

where the sum is over all $m \times m$ matrices $M = (m_{ij})_{1\leq i,j\leq m}$ with non-negative integer entries m_{ij}. There is a famous bijection due to Robinson–Schensted–Knuth [72, 137, 142] (see for instance [139] for a clear exposition of the bijection) that identifies such a matrix M to a biword in the alphabet of letters $\binom{r}{s}$ with $1 \leq r, s \leq m$. The bijection maps these biwords to pairs (P, Q), where P and Q are semi-standard tableaux of the same shape and $\mathbf{x}^P \mathbf{y}^Q = \prod_{i,j=1}^{m} (x_i y_j)^{m_{ij}}$. As a consequence we conclude that

$$\prod_{i,j=1}^{m} \frac{1}{1-x_i y_j} = \sum_{M} (\mathbf{xy})^M = \sum_{(P,Q)} \mathbf{x}^P \mathbf{y}^Q = \sum_{\lambda} \Bigg(\sum_{\mathrm{shape}(P)=\lambda} \mathbf{x}^P \Bigg) \Bigg(\sum_{\mathrm{shape}(Q)=\lambda} \mathbf{y}^Q \Bigg) . \tag{6.4}$$

Since $\sum_{\mathrm{shape}(P)=\lambda} \mathbf{x}^P$ is triangularly related to the monomial basis, Eq. (6.2) must hold by comparing Eqs. (6.1) and (6.4). In this section we consider the generalization of the Cauchy identity and the RSK algorithm to k-Schur functions and their dual basis at $t = 1$.

By taking the coefficient of $x_1x_2\cdots x_m y_1 y_2\cdots y_m$ in Eq. (6.1), we find the identity

$$m! = \sum_{\lambda\vdash m} f_\lambda^2\ ,$$

where f_λ is the number of standard tableaux of shape λ. This may be seen as an algebraic formulation of the more standard presentation of the RSK algorithm on permutations, namely, there is a bijection between permutations π and pairs of tableaux (P, Q), where P and Q are standard tableaux of the same shape.

In Eq. (2.17), we stated that $\tilde{F}_\lambda^{(k)} = \sum_\mu K_{\lambda\mu}^{(k)} m_\mu$, where $K_{\lambda\mu}^{(k)}$ is equal to the number of weak tableaux of shape $\mathfrak{c}(\lambda)$ and weight μ. The collective results in [79, 93] show that

$$\tilde{F}_\lambda^{(k)}[X_m] = \sum_T \mathbf{x}^T\ , \tag{6.5}$$

where the sum is over all weak tableaux T of shape $\mathfrak{c}(\lambda)$ in the weight $\{1, 2, \ldots, m\}$.

Now consider the following k-bounded analogue of the kernel (6.3) given by

$$\begin{aligned}&\prod_{j=1}^{m}(1 + h_1[X_m]y_j + h_2[X_m]y_j^2 + \cdots + h_k[X_m]y_j^k)\\ &\quad = \sum_{\lambda:\lambda_1\le k} h_\lambda[X_m]m_\lambda[Y_m] = \sum_M (\mathbf{xy})^M,\end{aligned} \tag{6.6}$$

where the sum on the right hand side of the equation is over all k-bounded matrices $M = (m_{ij})_{1\le i,j\le n}$, whose entries are non-negative integers and satisfy $\sum_{i=1}^n m_{ij} \le k$, and $(\mathbf{xy})^M = \prod_{i,j}(x_i y_j)^{m_{ij}}$.

In [81] the authors provide a bijection between the set of k-bounded matrices and pairs of tableaux (P, Q) such that P is a strong tableau and Q is a weak tableau that are both of the same $(k+1)$-core shape. This is done by introducing an insertion algorithm which generalizes that of the RSK bijection (and reduces to RSK case when $k \ge \sum_{i,j} m_{ij}$).

Here we provide an exposition of a special case of this bijection, namely between permutation matrices (with entries in $\{0, 1\}$ with a single 1 in each row and column) and pairs of tableaux (P, Q), where P is a strong standard tableau and Q is a weak standard tableau.

The bijection in [81] (which is called 'affine insertion' in analogy with RSK-insertion) shows that the duality of the weak and strong functions can be expressed through the duality of the kernel in (6.6) and hence

$$\prod_{j=1}^{m}(1 + h_1[X_m]y_j + h_2[X_m]y_j^2 + \cdots + h_k[X_m]y_j^k)$$
$$= \sum_{\lambda:\lambda_1 \le k} \mathrm{Strong}_{\mathfrak{c}(\lambda)}[X_m]\mathrm{Weak}_{\mathfrak{c}(\lambda)}[Y_m], \tag{6.7}$$

where the functions are defined as

$$\mathrm{Weak}_\kappa[X_m] = \sum_T \mathbf{x}^T \tag{6.8}$$

with the sum over all weak tableaux of shape κ (a $(k+1)$-core) and

$$\mathrm{Strong}_\kappa[X_m] = \sum_T \mathbf{x}^T \tag{6.9}$$

with the sum is over all strong tableaux of shape κ. In both cases $\mathbf{x}^T$ represents a monomial associated to the weight. The equality $\tilde{F}_\lambda^{(k)}[X_m] = \mathrm{Weak}_{\mathfrak{c}(\lambda)}[X_m]$ with dual k-Schur functions relies on a permutation action on the weight which can be found in [92]. The equality $s_\lambda^{(k)}[X_m] = \mathrm{Strong}_{\mathfrak{c}(\lambda)}[X_m]$ follows by a duality argument after showing that the functions $\mathrm{Strong}_\kappa[X_m]$ form a basis [81].

Let f_κ^{weak} be the number of standard weak tableaux of shape κ and $f_\kappa^{\mathrm{strong}}$ be the number of standard strong tableaux of shape κ, where κ is a $(k+1)$-core. If we take the coefficient of $x_1x_2\cdots x_my_1y_2\cdots y_m$ in Eq. (6.7), we find the following combinatorial result

$$m! = \sum_{\lambda:\lambda_1 \le k} f_{\mathfrak{c}(\lambda)}^{\mathrm{strong}} f_{\mathfrak{c}(\lambda)}^{\mathrm{weak}}, \tag{6.10}$$

where the sum is over k-bounded partitions of λ of m. This formula can be seen as a manifestation of a bijection between the set of permutations σ of S_m and pairs of tableaux $(P^{(k)}, Q^{(k)})$, where $P^{(k)}$ is a strong standard tableau and $Q^{(k)}$ is a weak standard tableau.

6.2 A Brief Introduction to Fomin's Growth Diagrams

Affine insertion is proved using Fomin's growth diagrams [42] which is a tool of presenting insertion algorithms on graded posets via certain local rules. To put the affine insertion algorithm in context, we give a brief presentation of the usual RSK algorithm between permutations and pairs of standard tableaux in terms of growth diagrams in order to show how the algorithms compare. In Sect. 6.3 we will demonstrate how the algorithm can be generalized to the bijection which explains

Eq. (6.10). The treatment we present in this section follows roughly the way of viewing Fomin's growth diagrams that is presented in [139, Sect. 5.2] with a few modifications in orientation.

We begin with an $n \times n$ permutation matrix corresponding to a permutation π (we use the convention that row i has a 1 in column π_i) and convert it into a pair of standard tableaux of the same shape. To do this we draw an $n \times n$ lattice of squares and label the vertices of this lattice with partitions and the centers of these squares with the entries of the permutation matrix. At the start of the procedure we begin by labeling only the first row and first column of vertices with empty partitions and fill in the rest of the diagram by a recursive procedure using a set of *local rules*.

To describe the RSK algorithm we describe a 'local rule' which is a bijection between two types of arrays.

Case 1:

$$\begin{array}{ccc} \lambda & \rightarrow & \mu \\ \downarrow & 0 & \\ \nu & & \end{array} \qquad \longleftrightarrow \qquad \begin{array}{ccc} & & \mu \\ & & \downarrow \\ \nu & \rightarrow & \gamma \end{array}$$

(a) If $\lambda = \mu = \nu$, then $\gamma = \nu$.
(b) If $\mu \neq \nu$, then $\gamma = \mu \cup \nu$.
(c) If λ is strictly contained in $\mu = \nu$, then if μ is obtained from λ by adding a cell in row i, then γ is obtained from μ by adding a cell in row $i + 1$.

Case 2:

$$\begin{array}{ccc} \lambda & \rightarrow & \mu \\ \downarrow & 1 & \\ \nu & & \end{array} \qquad \longleftrightarrow \qquad \begin{array}{ccc} & & \mu \\ & & \downarrow \\ \nu & \rightarrow & \gamma \end{array}$$

This case can only occur when $\lambda = \mu = \nu$, and then γ is obtained from μ by adding a cell in the first row.

By successively applying these local rules, the growth diagram is filled in until it is an $(n + 1) \times (n + 1)$ array of partitions. Because each of the rules we apply is a bijection, we need only remember the last row and last column of the array and the rest of the table can be recovered by applying the local rule in reverse. The last row of this table is a sequence of partitions each of which differ by a single cell and so can be interpreted as a standard tableau which agrees with the insertion tableau of π. The last column of table is also a sequence of partitions each of which differ by a single cell; the corresponding standard tableau agrees with the recording tableau corresponding to the permutation π.

The important thing to notice is that the local rules can be reversed. For this reason if we are given just the pair of tableaux that represent the last row and the last column of the table, it is possible to reconstruct the entire table and hence the permutation matrix.

A beautiful feature of the growth diagram perspective on the RSK insertion algorithm is also that it makes it completely manifest that interchanging the insertion tableau P and recording tableau Q inverts the permutation π. This can be seen by interchanging the rows and columns of the array, which inverts the permutation matrix and interchanges P and Q.

Example 6.1. Let us consider the permutation 4132 as a running example. We begin with a row and a column of 5 empty partitions and the entries of the permutation matrix in an array pictured below.

$$
\begin{array}{ccccccccc}
\emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset \\
\downarrow & 0 & & 0 & & 0 & & 1 & \\
\emptyset & & & & & & & & \\
\downarrow & 1 & & 0 & & 0 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 0 & & 1 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 1 & & 0 & & 0 & \\
\emptyset & & & & & & & &
\end{array}
$$

The local rules may be applied at first only in one place:

$$
\begin{array}{ccccccccc}
\emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset \\
\downarrow & 0 & \downarrow & 0 & & 0 & & 1 & \\
\emptyset & \rightarrow & \emptyset & & & & & & \\
\downarrow & 1 & & 0 & & 0 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 0 & & 1 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 1 & & 0 & & 0 & \\
\emptyset & & & & & & & &
\end{array}
$$

In successive steps, the local rules may be applied in each corner.

$$
\begin{array}{ccccccccc}
\emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset \\
\downarrow & 0 & \downarrow & 0 & \downarrow & 0 & & 1 & \\
\emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & & & & \\
\downarrow & 1 & \downarrow & 0 & & 0 & & 0 & \\
\emptyset & \rightarrow & \square & & & & & & \\
\downarrow & 0 & & 0 & & 1 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 1 & & 0 & & 0 & \\
\emptyset & & & & & & & &
\end{array}
$$

$$
\begin{array}{ccccccccc}
\emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset \\
\downarrow & 0 & \downarrow & 0 & \downarrow & 0 & \downarrow & 1 & \\
\emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset & & \\
\downarrow & 1 & \downarrow & 0 & \downarrow & 0 & & 0 & \\
\emptyset & \to & \square & \to & \square & & & & \\
\downarrow & 0 & \downarrow & 0 & & 1 & & 0 & \\
\emptyset & \to & \square & & & & & & \\
\downarrow & 0 & & 1 & & 0 & & 0 & \\
\emptyset & & & & & & & &
\end{array}
$$

By continuing to apply the local rules, we arrive at the following completed table:

$$
\begin{array}{cccccccccc}
\emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset & \\
\downarrow & 0 & \downarrow & 0 & \downarrow & 0 & \downarrow & 1 & \downarrow & \\
\emptyset & \to & \emptyset & \to & \emptyset & \to & \emptyset & \to & \square & \\
\downarrow & 1 & \downarrow & 0 & \downarrow & 0 & \downarrow & 0 & \downarrow & \\
\emptyset & \to & \square & \to & \square & \to & \square & \to & \ydiagram{1,1} & Q \\
\downarrow & 0 & \downarrow & 0 & \downarrow & 1 & \downarrow & 0 & \downarrow & \\
\emptyset & \to & \square & \to & \square & \to & \ydiagram{2} & \to & \ydiagram{1,2} & \\
\downarrow & 0 & \downarrow & 1 & \downarrow & 0 & \downarrow & 0 & \downarrow & \\
\emptyset & \to & \square & \to & \ydiagram{2} & \to & \ydiagram{1,2} & \to & \ydiagram{1,1,2} & \\
 & & & & P & & & & &
\end{array}
$$

Now in order to reconstruct this entire table, we need only remember the two standard tableaux $(P, Q) = \left(\ytableaushort{4,3,12}, \ytableaushort{4,2,13} \right)$ which correspond to the last row and the last column of the table. This indeed agrees with the usual insertion tableau P and recording tableau Q when row-inserting 4132 (see for example [139]).

6.3 *Affine Insertion*

Now that we have presented Fomin's growth diagrams as a tool for understanding RSK, we will give the local rules necessary to understand k-affine insertion for permutations. This is the algorithm presented in [81]. Parts of these local rules use operations which are described in Sects. 1.2–1.4 (in particular the action of s_i, the notion of strong and weak cover, and the vocabulary of content and residue).

We have stripped down the algorithm presented in [81] in hopes of making it clearer by having fewer details to follow. The algorithm presented in [81] is slightly more general because it includes an additional rule that generalizes the bijection from k-bounded non-negative integer matrices to pairs tableaux of the same shape, the first is a strong semi-standard tableau, the second is a weak semi-standard tableau. This more general bijection is sufficient to show Eq. (6.7). Here we are pairing down the presentation rule to only demonstrate how Eq. (6.10) works.

To do this we construct again a table that will be a growth diagram, where there is an $(n+1) \times (n+1)$ array of $(k+1)$-cores and between these entries we put the $n \times n$ permutation matrix.

There is an additional piece of information which is recorded in this matrix besides the shapes of the $k+1$-cores. The horizontal connectors in our growth diagrams keep track of (possibly empty) strong covers and the vertical connectors keep track of (possibly empty) weak covers. If there are two $(k+1)$-cores that are adjacent in the same row, $\tau \rightarrow \kappa$, then either $\tau = \kappa$ or κ covers τ in the strong order. In the second case, we also need to mark one of the connected components of κ/τ in the diagram of κ or keep track of this marking on the arrow $\xrightarrow{c}$, where c represents the content of the diagonal of the marked cell. When working with the diagram in the examples below we only record this information by marking a cell of κ/τ within core κ for compactness of notation.

We begin with the first row and column of this array consisting of empty cores only (and hence there are no markings necessary). Given a corner of the table that is partially filled in, we complete the rest with the following local rules.

Case 1:

$$\begin{array}{ccc} \tau & \xrightarrow{c} & \kappa \\ \downarrow & 0 & \\ \theta & & \end{array} \qquad \longleftrightarrow \qquad \begin{array}{ccc} & & \kappa \\ & & \downarrow \\ \theta & \xrightarrow{c'} & \zeta \end{array}$$

Try to apply (a)–(c) in this order. If a case does not apply, proceed to the next case.

(a) If $\tau = \kappa$, then $\zeta = \theta$ and neither κ nor ζ will be marked; if $\tau = \theta$, then $\zeta = \kappa$ and $c' = c$.

(b) If κ/τ is not contained in θ/τ, let r be the residue of the cells $\theta/\tau \pmod{k+1}$ (there is exactly one). In this case ζ is s_r applied to κ. One cell of κ on the diagonal with content c is marked. In ζ mark the component that has an overlap with the marked ribbon in κ (alternatively, if c does not have residue r, then $c' = c$; otherwise c' is on one diagonal higher than c).

(c) If τ is strictly contained in $\kappa = \theta$, let c' be the content of the first diagonal which is weakly to the left of the marked ribbon of κ/τ and is an addable cell of κ. Then $\zeta = s_{c'}\kappa$ and the marked cell of ζ is on the diagonal with content c'.

Case 2:

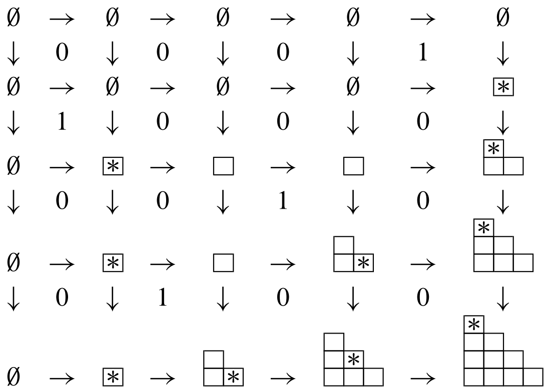
$$\begin{array}{ccc} \tau & \rightarrow & \kappa \\ \downarrow & 1 & \\ \theta & & \end{array} \qquad \longleftrightarrow \qquad \begin{array}{ccc} & & \kappa \\ & & \downarrow \\ \theta & \xrightarrow{c'} & \zeta \end{array}$$

This case can only occur when $\tau = \kappa = \theta$, and then ζ is s_{τ_1} applied to τ (the effect of adding a cell in the first row of τ, but as a $(k+1)$-core). A marking c' is added in last cell of the first row of ζ.

Example 6.2. Let us compute the growth diagram for the matrix corresponding to the permutation 4132. As in Example 6.1, we begin our growth diagram with the first row and column consisting of empty cores.

$$\begin{array}{ccccccccc}
\emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset & \rightarrow & \emptyset \\
\downarrow & 0 & & 0 & & 0 & & 1 & \\
\emptyset & & & & & & & & \\
\downarrow & 1 & & 0 & & 0 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 0 & & 1 & & 0 & \\
\emptyset & & & & & & & & \\
\downarrow & 0 & & 1 & & 0 & & 0 & \\
\emptyset & & & & & & & &
\end{array}$$

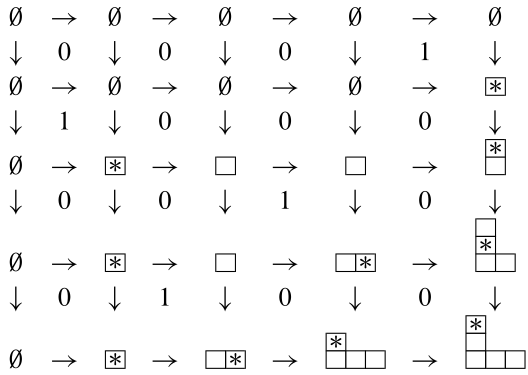
Below is the growth diagram for $k = 1$:



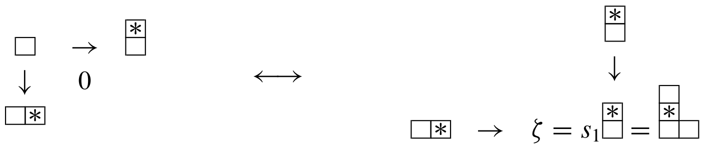
Note that since the value of k is too small, Case 1(b) is not used. By reading the last row of this table we can encode it as a single strong tableau. When $k = 1$ there is only one weak tableau of shape $(m, m-1, \ldots, 2, 1)$. The last row and column of this table can then be encoded as

$$\left(\begin{array}{|c|ccc} \hline 4^* & & & \\ \hline 3 & \multicolumn{1}{|c|}{4} & & \\ \hline 2 & \multicolumn{1}{|c|}{3^*} & \multicolumn{1}{c|}{4} & \\ \hline 1^* & \multicolumn{1}{|c|}{2^*} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} \\ \hline \end{array} \; , \; \begin{array}{|c|ccc} \hline 4 & & & \\ \hline 3 & \multicolumn{1}{|c|}{4} & & \\ \hline 2 & \multicolumn{1}{|c|}{3} & \multicolumn{1}{c|}{4} & \\ \hline 1 & \multicolumn{1}{|c|}{2} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} \\ \hline \end{array} \right) .$$

For $k = 2$ and starting with the same permutation the situation is a little more complicated:



Now it is necessary to apply all four rules to fill in the growth diagram. The first time that Case 1(b) occurs is constructing the last entry in the fourth row of cores (the last entry of the third row of the permutation matrix).

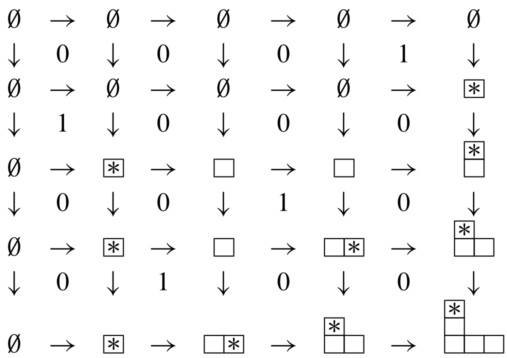


If we just record the last row as a strong tableau and the last column as a weak tableau, we have the follow pair:

$$\left(\begin{array}{|c|cc} \hline 4^* & & \\ \hline 3^* & & \\ \hline 1^* & \multicolumn{1}{|c|}{2^*} & \multicolumn{1}{c|}{3} \\ \hline \end{array}, \begin{array}{|c|cc} \hline 3 & & \\ \hline 2 & & \\ \hline 1 & \multicolumn{1}{|c|}{3} & \multicolumn{1}{c|}{4} \\ \hline \end{array} \right)$$

and this pair of tableaux is sufficient to reconstruct the entire table.

For $k = 3$, the case is similar to the $k = 2$ case in that there are examples where all four rules are applied in order to construct the table:



Because we can reconstruct the table from the last row and column of this table, it is necessary to keep track only of the strong tableau representing the last row and the weak tableau representing the last column. This is represented by the pair,

$$\left(\begin{array}{lll}4^*&&\\3^*&&\\1^*&2^*&4\end{array},\ \begin{array}{lll}4&&\\2&&\\1&3&4\end{array}\right).$$

Example 6.3. In the following table we have presented permutations of 3 (corresponding to permutation matrices) which are in bijection with pairs of weak and strong tableaux for $k = 1, 2$ and 3. When $k = 3$, the tableaux are in bijection with pairs of standard tableaux of the same shape by dropping the markings in the strong tableaux.

σ	$(P^{(1)}, Q^{(1)})$	$(P^{(2)}, Q^{(2)})$	$(P^{(\infty)}, Q^{(\infty)})$
123	$\left(\begin{array}{lll}3&&\\2&3&\\1^*&2^*&3^*\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{lll}3&&\\1^*&2^*&3^*\end{array}, \begin{array}{lll}3&&\\1&2&3\end{array}\right)$	$\left(\begin{array}{lll}1^*&2^*&3^*\end{array}, \begin{array}{lll}1&2&3\end{array}\right)$
132	$\left(\begin{array}{lll}3&&\\2&3^*&\\1^*&2^*&3\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{lll}3^*&&\\1^*&2^*&3\end{array}, \begin{array}{lll}3&&\\1&2&3\end{array}\right)$	$\left(\begin{array}{ll}3^*&\\1^*&2^*\end{array}, \begin{array}{ll}3&\\1&2\end{array}\right)$
213	$\left(\begin{array}{lll}3&&\\2^*&3&\\1^*&2&3^*\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{ll}3&\\2^*&\\1^*&3^*\end{array}, \begin{array}{ll}3&\\2&\\1&3\end{array}\right)$	$\left(\begin{array}{ll}2^*&\\1^*&3^*\end{array}, \begin{array}{ll}2&\\1&3\end{array}\right)$
231	$\left(\begin{array}{lll}3&&\\2^*&3^*&\\1^*&2&3\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{lll}2^*&&\\1^*&3&3^*\end{array}, \begin{array}{lll}3&&\\1&2&3\end{array}\right)$	$\left(\begin{array}{ll}2^*&\\1^*&3^*\end{array}, \begin{array}{ll}3&\\1&2\end{array}\right)$
312	$\left(\begin{array}{lll}3^*&&\\2&3&\\1^*&2^*&3\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{ll}3&\\3^*&\\1^*&2^*\end{array}, \begin{array}{ll}3&\\2&\\1&3\end{array}\right)$	$\left(\begin{array}{ll}3^*&\\1^*&2^*\end{array}, \begin{array}{ll}2&\\1&3\end{array}\right)$
321	$\left(\begin{array}{lll}3^*&&\\2^*&3&\\1^*&2&3\end{array}, \begin{array}{lll}3&&\\2&3&\\1&2&3\end{array}\right)$	$\left(\begin{array}{ll}3^*&\\2^*&\\1^*&3\end{array}, \begin{array}{ll}3&\\2&\\1&3\end{array}\right)$	$\left(\begin{array}{l}3^*\\2^*\\1^*\end{array}, \begin{array}{l}3\\2\\1\end{array}\right)$

Example 6.4. In the following table we have permutations of 4 (corresponding to permutation matrices) which are in bijection with pairs of weak and strong tableaux for $k = 1, 2$ and $3, 4$. When $k = 4$, the tableaux are in bijection with pairs of standard tableaux of the same shape by dropping the markings in the strong tableaux.

σ	$(P^{(1)}, Q^{(1)})$	$(P^{(2)}, Q^{(2)})$	$(P^{(3)}, Q^{(3)})$	$(P^{(\infty)}, Q^{(\infty)})$
1234	$\left(\begin{smallmatrix}4\\3&4\\2&3&4\\1^*&2^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3&4\\1^*&2^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\1^*&2^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}1^*&2^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}1&2&3&4\end{smallmatrix}\right)$
1243	$\left(\begin{smallmatrix}4\\3&4\\2&3&4^*\\1^*&2^*&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3&4^*\\1^*&2^*&3^*&4\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\1^*&2^*&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\1^*&2^*&3^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3\end{smallmatrix}\right)$
1324	$\left(\begin{smallmatrix}4\\3&4\\2&3^*&4\\1^*&2^*&3&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3^*&4\\1^*&2^*&3&4^*\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\3^*\\1^*&2^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3^*\\1^*&2^*&4^*\end{smallmatrix}, \begin{smallmatrix}3\\1&2&4\end{smallmatrix}\right)$
1342	$\left(\begin{smallmatrix}4\\3&4\\2&3^*&4^*\\1^*&2^*&3&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3^*&4^*\\1^*&2^*&3&4\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3^*\\1^*&2^*&4&4^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3^*\\1^*&2^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3\end{smallmatrix}\right)$
1423	$\left(\begin{smallmatrix}4\\3&4^*\\2&3&4\\1^*&2^*&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\3\\1^*&2^*&3^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\4^*\\1^*&2^*&3^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\1^*&2^*&3^*\end{smallmatrix}, \begin{smallmatrix}3\\1&2&4\end{smallmatrix}\right)$
1432	$\left(\begin{smallmatrix}4\\3&4^*\\2&3^*&4\\1^*&2^*&3&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\3^*\\1^*&2^*&3\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\3^*\\1^*&2^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\3^*\\1^*&2^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2\end{smallmatrix}\right)$
2134	$\left(\begin{smallmatrix}4\\3&4\\2^*&3&4\\1^*&2&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3\\2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}3\\2\\1&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\2\\1&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}2\\1&3&4\end{smallmatrix}\right)$
2143	$\left(\begin{smallmatrix}4\\3&4\\2^*&3&4^*\\1^*&2&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\3\\2^*&4^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\2&4\\1&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}2&4\\1&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}2&4\\1&3\end{smallmatrix}\right)$
2314	$\left(\begin{smallmatrix}4\\3&4\\2^*&3^*&4\\1^*&2&3&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4\\1^*&3&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4\\2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}3\\1&2&4\end{smallmatrix}\right)$
2341	$\left(\begin{smallmatrix}4\\3&4\\2^*&3^*&4^*\\1^*&2&3&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4^*\\1^*&3&3^*&4\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*\\1^*&3^*&4&4^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\1&2&3\end{smallmatrix}\right)$
2413	$\left(\begin{smallmatrix}4\\3&4^*\\2^*&3&4\\1^*&2&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}3\\2^*\\1^*&3^*&4^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}2^*&4^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}3&4\\1&2\end{smallmatrix}\right)$
2431	$\left(\begin{smallmatrix}4\\3&4^*\\2^*&3^*&4\\1^*&2&3&4\end{smallmatrix}, \begin{smallmatrix}4\\3&4\\2&3&4\\1&2&3&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\2^*\\1^*&3&3^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&3\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\2^*\\1^*&3^*&4\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2&4\end{smallmatrix}\right)$	$\left(\begin{smallmatrix}4^*\\2^*\\1^*&3^*\end{smallmatrix}, \begin{smallmatrix}4\\3\\1&2\end{smallmatrix}\right)$

σ	$(P^{(1)}, Q^{(1)})$	$(P^{(2)}, Q^{(2)})$	$(P^{(3)}, Q^{(3)})$	$(P^{(\infty)}, Q^{(\infty)})$
3124	(4 / 3* 4 / 2 3 4 / 1* 2* 3 4*, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(3 / 3* / 1* 2* 4*, 3 / 2 / 1 3 4)	(4 / 3* / 1* 2* 4*, 4 / 2 / 1 3 4)	(3* / 1* 2* 4*, 2 / 1 3 4)
3142	(4 / 3* 4 / 2 3 4* / 1* 2* 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4 / 3 / 3* 4* / 1* 2*, 4 / 3 / 2 4 / 1 3)	(3* 4* / 1* 2*, 2 4 / 1 3)	(3* 4* / 1* 2*, 2 4 / 1 3)
3214	(4 / 3* 4 / 2* 3 4 / 1* 2 3 4*, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(3* / 2* / 1* 3 4*, 3 / 2 / 1 3 4)	(4 / 3* / 2* / 1* 4*, 4 / 3 / 2 / 1 4)	(3* / 2* / 1* 4*, 3 / 2 / 1 4)
3241	(4 / 3* 4 / 2* 3 4* / 1* 2 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4 / 3* / 2* 4* / 1* 3, 4 / 3 / 2 4 / 1 3)	(3* / 2* / 1* 4 4*, 4 / 2 / 1 3 4)	(3* / 2* / 1* 4*, 4 / 2 / 1 3)
3412	(4 / 3* 4* / 2 3 4 / 1* 2* 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(3 / 3* / 1* 2* 4*, 4 / 3 / 1 2 3)	(3* 4* / 1* 2*, 3 4 / 1 2)	(3* 4* / 1* 2*, 3 4 / 1 2)
3421	(4 / 3* 4* / 2* 3 4 / 1* 2 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(3* / 2* / 1* 3 4*, 4 / 3 / 1 2 3)	(3* / 2* / 1* 4 4*, 4 / 3 / 1 2 4)	(3* / 2* / 1* 4*, 4 / 3 / 1 2)
4123	(4* / 3 4 / 2 3 4 / 1* 2* 3* 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 3 / 1* 2* 3*, 3 / 2 / 1 3 4)	(4 / 4* / 1* 2* 3*, 4 / 2 / 1 3 4)	(4* / 1* 2* 3*, 2 / 1 3 4)
4132	(4* / 3 4 / 2 3* 4 / 1* 2* 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 3* / 1* 2* 3, 3 / 2 / 1 3 4)	(4* / 3* / 1* 2* 4, 4 / 2 / 1 3 4)	(4* / 3* / 1* 2*, 4 / 2 / 1 3)
4213	(4* / 3 4 / 2* 3 4 / 1* 2 3* 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 3 / 2* 4 / 1* 3*, 4 / 3 / 2 4 / 1 3)	(4 / 4* / 2* / 1* 3*, 4 / 3 / 2 / 1 4)	(4* / 2* / 1* 3*, 3 / 2 / 1 4)
4231	(4* / 3 4 / 2* 3* 4 / 1* 2 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 2* / 1* 3 3*, 3 / 2 / 1 3 4)	(4* / 2* / 1* 3* 4, 4 / 2 / 1 3 4)	(4* / 2* / 1* 3*, 4 / 2 / 1 3)
4312	(4* / 3* 4 / 2 3 4 / 1* 2* 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 3 / 3* 4 / 1* 2*, 4 / 3 / 2 4 / 1 3)	(4 / 4* / 3* / 1* 2*, 4 / 3 / 2 / 1 4)	(4* / 3* / 1* 2*, 3 / 2 / 1 4)
4321	(4* / 3* 4 / 2* 3 4 / 1* 2 3 4, 4 / 3 4 / 2 3 4 / 1 2 3 4)	(4* / 3* / 2* 4 / 1* 3, 4 / 3 / 2 4 / 1 3)	(4* / 3* / 2* / 1* 4, 4 / 3 / 2 / 1 4)	(4* / 3* / 2* / 1*, 4 / 3 / 2 / 1)

6.4 The t-Compatible Affine Insertion Algorithm

Since the expression on the left of Eq. (6.10) is independent of k, we realize that since $w \leftrightarrow (P^{(k)}, Q^{(k)}) \leftrightarrow (P^{(k+1)}, Q^{(k+1)})$, where $P^{(k)}$ and $P^{(k+1)}$ are strong k and $(k+1)$-tableaux may help us to see how $s_\lambda^{(k)}$ can be expressed as a positive sum of elements $s_\mu^{(k+1)}$. It may be possible to understand the expansion of $s_\lambda^{(k)}$ in terms of Schur functions or in terms of $s_\lambda^{(k+1)}$ using this bijection, but this would impose certain conditions on the shapes of tableaux $P^{(k)}$ and $P^{(k+1)}$ that do not seem to hold.

In this section we present some evidence suggesting that one might hope for an affine insertion bijection which has additional properties that are not shared with the affine insertion algorithm of Sect. 6.3.

There is also a t-analogue of Eq. (6.7) which can be used as a stronger guide for the combinatorics of an analogue of the Robinson–Schensted–Knuth bijection. The coefficient of $m_{(1^m)}[X]$ in $Q'_{(1^m)}[X;t]$ is equal to the t-analogue of $m!$, $[m]_t! = (1-t)^{-m}\prod_{i=1}^m(1-t^i)$, and hence we have

$$
\begin{aligned}
[m]_t! = \left\langle Q'_{(1^m)}[X;t], h_{(1^m)}[X]\right\rangle &= \sum_{\lambda\vdash m} K_{\lambda(1^m)}(t)\left\langle s_\lambda[X], h_{(1^m)}[X]\right\rangle \\
&= \sum_{\lambda\vdash m} K_{\lambda(1^m)}(t) f_\lambda = \sum_{\lambda:\lambda_1\le k} K^{(k)}_{\lambda(1^m)}(t)\left\langle s_\lambda^{(k)}[X;t], h_{(1^m)}[X]\right\rangle . \qquad (6.11)
\end{aligned}
$$

The polynomial $\left\langle s_\lambda^{(k)}[X;t], h_{(1^m)}[X]\right\rangle$ is equal to $\sum_P t^{\mathrm{spin}(P)}$, where the sum is over all strong standard k-tableaux of shape $\mathfrak{c}(\lambda)$ by Eq. (3.16). The coefficient $K^{(k)}_{\lambda(1^m)}(t)$ is a t-analogue of the number of standard weak k-tableaux of shape $\mathfrak{c}(\lambda)$. This equation is just one possible refinement of Eq. (6.10). Similarly, the right hand side of this equation depends on k while the left hand side does not and this indicates that we might hope to see some relationship between the bijection at level k and level $k+1$ that relates the t statistic.

If we are looking to explain this algebraic expression with a bijection, we would like to find a statistic $charge$ on standard weak k-tableaux and a bijection between permutations and pairs of strong and weak tableaux of the same shape $w^{-1} \leftrightarrow (P^{(k)}, Q^{(k)})$ such that

$$
\mathrm{charge}(w) = \mathrm{spin}(P^{(k)}) + \mathrm{charge}(Q^{(k)}) . \qquad (6.12)
$$

Note that we taking the association with w^{-1} to ensure that everything agrees since as $k \to \infty$ the statistic charge was defined so that the charge of a permutation is the charge of its insertion tableau. The statistic spin on k-strong tableaux is different in nature than the charge statistic and in general $\mathrm{spin}(P^{(\infty)}) = 0$.

The reason the previous affine insertion algorithm is not quite ‘real’ is that we are unable to use it to explain this t-analogue. A ‘real’ affine insertion algorithm would allow us to take a definition of the charge statistic so that if $w^{-1} \leftrightarrow (P^{(k)}, Q^{(k)})$, then $\mathrm{charge}(Q^{(k)}) = \mathrm{charge}(w) - \mathrm{spin}(P^{(k)})$. It would need to be the case that if u and v are two different permutations such that $u^{-1} \leftrightarrow (P^{1(k)}, Q^{(k)})$ and $v^{-1} \leftrightarrow (P^{2(k)}, Q^{(k)})$, then $\mathrm{charge}(u) - \mathrm{spin}(P^{1(k)}) = \mathrm{charge}(v) - \mathrm{spin}(P^{2(k)})$. The following example, based on the calculations from Example 6.3, shows that this affine insertion algorithm is not compatible with the spin statistic in this sense.

Example 6.5. It is probably easiest to see that Eq. (6.12) cannot hold in our example unless $\mathrm{charge}(w) = \mathrm{spin}(P^{(1)})$ because when $k = 1$ all of the $Q^{(1)}$ tableaux are the same. Consider the case $k = 1$ and $u = u^{-1} = 132$ with $\mathrm{charge}(u) = 2$. Then

$P_1^{(1)} = \begin{array}{|c|c|c|}\cline{1-1} 3 \\ \cline{1-2} 2 & 3^* \\ \hline 1^* & 2^* & 3 \\ \hline \end{array}$ and $\mathrm{spin}(P_1^{(1)}) = 1$.

Also, $v = v^{-1} = 213$ with $\mathrm{charge}(v) = 1$. Then $P_2^{(1)} = \begin{array}{|c|c|c|}\cline{1-1} 3 \\ \cline{1-2} 2^* & 3 \\ \hline 1^* & 2 & 3^* \\ \hline \end{array}$ and $\mathrm{spin}(P_2^{(1)}) = 1$, but $Q^{(1)} = \begin{array}{|c|c|c|}\cline{1-1} 3 \\ \cline{1-2} 2 & 3 \\ \hline 1 & 2 & 3 \\ \hline \end{array}$ is the same in both cases and if there is a charge statistic it should be the same.

7 The k-Shape Poset and a Branching Rule for Expressing k-Schur in $(k+1)$-Schur Functions

One of the more recent developments with the definition of k-Schur functions defined as the sum over strong tableaux is an explanation of why they expand positively in the $(k+1)$-Schur functions. That is, there are nonnegative integer coefficients $b_{\mu\lambda}^{(k)}$ such that

$$s_\mu^{(k-1)} = \sum_\lambda b_{\mu\lambda}^{(k)} s_\lambda^{(k)}$$

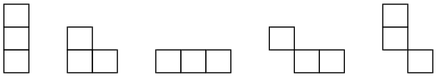
and in this section we will describe a combinatorial interpretation for the coefficients $b_{\mu\lambda}^{(k)}$.

If we consider a partition λ as a collection of cells, then $\mathrm{Int}^k(\lambda) = \{b \in \mathrm{dg}(\lambda) : \mathrm{hook}_\lambda(b) > k\}$ and $\partial^k(\lambda) = \lambda/\mathrm{Int}^k(\lambda)$. We define the row shape (resp. column shape) of λ to be the composition $rs^k(\lambda)$ (resp. $cs^k(\lambda)$) consisting of the number of cells in each of the rows of $\partial^k(\lambda)$. The partition λ is said to be a k-shape if both $rs^k(\lambda)$ and $cs^k(\lambda)$ are partitions. Let Π^k denote the set of k-shapes and Π_N^k represent the set of k-shapes λ such that $|\partial^k(\lambda)| = N$. Notice that both the k-cores and the $(k+1)$-cores of size N are a subset of Π_N^k.

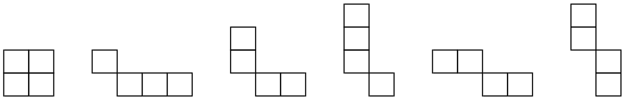
Example 7.1. If $k = 3$, then

$$\Pi_3^3 = \{(1,1,1), (2,1), (3), (3,1), (2,1,1)\}$$

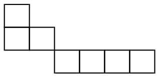
since they correspond to the 3-boundaries



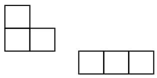
$$\Pi_4^3 = \{(2,2), (4,1), (3,1,1), (2,1,1,1), (4,2), (2,2,1,1)\}$$



Example 7.2. The partition $\lambda = (6,2,1)$ is a 4-shape but it is not a 3-shape. We calculate that

$\partial^4(\lambda) =$ 

and hence $rs^4(\lambda) = (4,2,1)$ and $cs^4(\lambda) = (2,1,1,1,1,1)$, but

$\partial^3(\lambda) =$ 

so that $rs^3(\lambda) = (3,2,1)$ and $cs^3(\lambda) = (2,1,0,1,1,1)$ and hence it is not a 3-shape.

Example 7.3. We include a table of the number of k-shapes for $1 \leq k \leq 9$ and $1 \leq N \leq 13$. In the limit (for $N < k$) it is the case that $|\Pi_N^k|$ is equal to the number of partitions of N.

$k \backslash N$	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$k=1$	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$k=2$	1	1	3	3	6	6	10	10	15	15	21	21	28	28
$k=3$	1	1	2	5	6	10	15	21	27	40	48	65	81	103
$k=4$	1	1	2	3	8	9	15	23	35	42	69	86	116	155
$k=5$	1	1	2	3	5	11	14	21	30	49	67	90	120	177
$k=6$	1	1	2	3	5	7	16	19	30	41	60	89	127	163
$k=7$	1	1	2	3	5	7	11	21	27	40	56	79	107	163
$k=8$	1	1	2	3	5	7	11	15	29	36	54	73	105	138
$k=9$	1	1	2	3	5	7	11	15	22	38	49	70	97	134

We will define a poset structure on the set Π_N^k by describing how the elements are related by a set of row and column moves. In order to define row and column moves we need to define a notion of row-type and column-type strings which describe the movement of cells to get from one k-shape to another.

For a cell $b = (x, y)$ we say that the diagonal index of b is $d(b) = y - x$. Two cells b, b' are called contiguous if $|d(b) - d(b')| \in \{k, k + 1\}$. A string of length ℓ is a skew shape μ/λ consisting of cells $\{a_1, a_2, \ldots, a_\ell\}$ such that a_{i+1} and a_i are contiguous for each $1 \le i < \ell$ and a_{i+1} lies strictly below a_i.

For a skew diagram D, define $\mathrm{left}_a(D)$ to be the leftmost cell in the same row as the cell a and $\mathrm{bot}_a(D)$ to be the bottommost cell in same column as a.

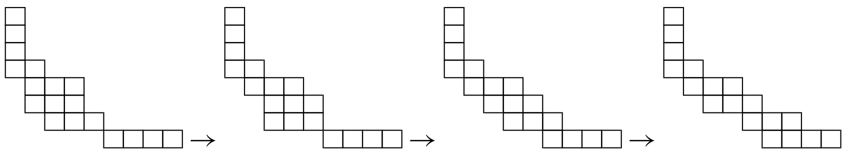
A string $\mu/\lambda = \{a_1, a_2, \ldots, a_\ell\}$ is called a row-string if $\mathrm{hook}_\lambda(\mathrm{left}_{a_1}(\partial^k(\lambda))) = k$ and $\mathrm{hook}_\lambda(\mathrm{bot}_{a_\ell}(\partial^k(\lambda))) < k$. It is called a column-string if the transpose picture is a row-string, or, if $\mathrm{hook}_\lambda(\mathrm{left}_{a_1}(\partial^k(\lambda))) < k$ and $\mathrm{hook}_\lambda(\mathrm{bot}_{a_\ell}(\partial^k(\lambda))) = k$.

A row move (resp. column move) of rank r and length ℓ is a chain of partitions $\lambda = \lambda^0 \subset \lambda^1 \subset \cdots \subset \lambda^r = \mu$ that satisfies

- $\lambda, \mu \in \Pi^k$
- $s_i = \lambda^i/\lambda^{i-1}$ is a row-type (resp. column-type) string consisting of ℓ cells for all $1 \le i \le r$.
- The strings s_i are all translates of each other
- The top cells (resp. rightmost cells) of $s_1, s_2, \ldots, s_r$ occur in consecutive columns (resp. rows) from left to right (resp. bottom to top).

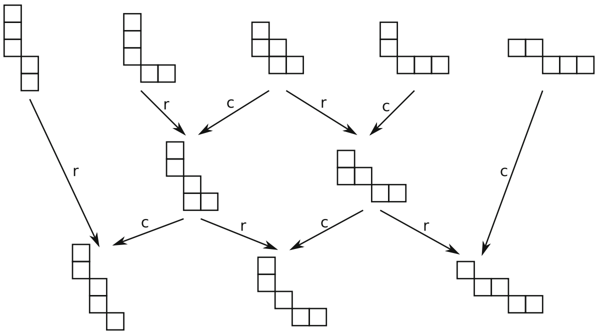
To be clear, a column move is a sequence of partitions whose conjugate partitions are a row move.

Example 7.4. If $k = 5$ and $N = 18$, then $\lambda = (9, 5, 4, 4, 2, 1, 1, 1)$ and $\mu = (9, 7, 5, 4, 2, 1, 1, 1)$ are both 5-shapes and the sequence of partitions $\lambda^0 = \lambda \subset \lambda^1 = (9, 5, 5, 4, 2, 1, 1, 1) \subset \lambda^2 = (9, 6, 5, 4, 2, 1, 1, 1) \subset \lambda^3 = \mu$ have the following boundaries:

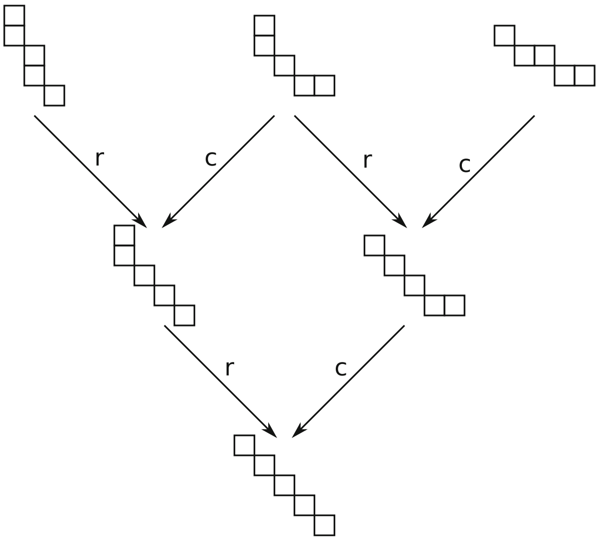


The set Π_N^k is endowed with a poset structure with an edge from λ to μ (or a cover relation $\lambda \gtrdot \mu$) if there is a row or a column move from λ to μ. With some analysis one can show that the minimal elements of the poset structure on Π_N^k are the k-cores and the maximal elements are the $(k + 1)$-cores.

Example 7.5. The set of elements Π_5^3 is endowed with the the following poset structure. The edges representing a row move are labeled with an r and those with a column move with a c.



Example 7.6. In the set of elements of Π_5^2, we see that the highest elements in this partial order are those that are the lowest elements of Π_5^3. The Hasse diagram resembles the following.



Now the combinatorial interpretation for the branching coefficients of the k-Schur functions is given in terms of paths within this poset (up to an equivalence on diamonds).

Define a charge of a move to be 0 if it is a row move and $r\ell$ for a move of length ℓ and rank r.

If $m, M, \tilde{m}, \tilde{M}$ are moves relating the k-shapes λ and γ through the following diagram,

$$\begin{tikzcd} & \lambda \arrow[dl, "m"'] \arrow[dr, "M"] & \\ \mu \arrow[dr, "\tilde{M}"'] & & \nu \arrow[dl, "\tilde{m}"] \\ & \gamma & \end{tikzcd} \tag{7.1}$$

such that

$$\mathrm{charge}(m) + \mathrm{charge}(\tilde{M}) = \mathrm{charge}(\tilde{m}) + \mathrm{charge}(M)$$

then the two paths from λ to γ are equivalent. Now consider two k-shapes $\kappa, \tau \in \Pi_N^k$ where κ is a $(k+1)$-core and τ is a k-core. Let $\mathcal{P}(\kappa, \tau)$ be the set of paths from κ to τ with respect to this equivalence relation.

The reason the k-shapes are related to k-Schur functions is that we have the following theorem.

Theorem 7.7 ([82, Theorem 2]). *Let λ be $k-1$ bounded partition,*

$$s_\lambda^{(k-1)}[X] = \sum_\mu |\mathcal{P}(\mathfrak{c}_k(\mu), \mathfrak{c}_{k-1}(\lambda))| s_\mu^{(k)}[X]$$

where the sum is over all k-bounded partitions μ.

While there is not a complete proof, the charge is defined so that the following conjecture should also hold.

Conjecture 7.8 ([82, Conjecture 3]). *Let λ be $k-1$ bounded partition,*

$$s_\lambda^{(k-1)}[X;t] = \sum_\mu \sum_{\mathbf{p} \in \mathcal{P}(\mathfrak{c}_{k+1}(\mu), \mathfrak{c}_k(\lambda))} t^{\mathrm{charge}(\mathbf{p})} s_\mu^{(k)}[X;t]$$

where the sum is over all k-bounded partitions μ.

Example 7.9. Example 7.5 and Theorem 7.7 can be used to calculate the following expansions of 2-Schur functions in 3-Schur functions.

$$s_{11111}^{(2)}[X] = s_{11111}^{(3)}[X] + s_{2111}^{(3)}[X] + s_{221}^{(3)}[X]$$

$$s_{2111}^{(2)}[X] = s_{2111}^{(3)}[X] + s_{221}^{(3)}[X] + s_{311}^{(3)}[X]$$

$$s_{221}^{(2)}[X] = s_{221}^{(3)}[X] + s_{311}^{(3)}[X] + s_{32}^{(3)}[X]\,.$$

This is because the two paths from the shape $(3, 2, 1)$ to $(4, 2, 1, 1)$ are equivalent under the diamond relation.

The relation that appears in Conjecture 7.8 says that

$$s_{11111}^{(2)}[X;t] = s_{11111}^{(3)}[X;t] + t^2 s_{2111}^{(3)}[X;t] + t^3 s_{221}^{(3)}[X;t]$$

$$s_{2111}^{(2)}[X;t] = s_{2111}^{(3)}[X;t] + t s_{221}^{(3)}[X;t] + t^2 s_{311}^{(3)}[X;t]$$

$$s_{221}^{(2)}[X;t] = s_{221}^{(3)}[X;t] + t s_{311}^{(3)}[X;t] + t^2 s_{32}^{(3)}[X;t]\,.$$

This is because the column moves with charge 2 are those from $(3,2,1,1)$ to $(3,2,2,1,1)$ and from $(5,2)$ to $(5,3,1)$. The others all have charge 1.

Example 7.10. The other poset we have drawn shows that $s^{(1)}_{11111}[X] = s^{(2)}_{11111}[X] + 2s^{(2)}_{2111}[X] + s^{(2)}_{221}[X]$. In this example the two paths from $(2,1,1,1)$ to $(1,1,1,1,1)$ are not equivalent. We can check that the charge of the move from $(4,2,1,1)$ to $(4,3,2,1,1)$ is 3, while the move from $(5,3,2,1)$ to $(5,4,3,2,1)$ is 4. Since the charge from $(5,3,1)$ to $(5,3,2,1)$ is 2, we conclude that Conjecture 7.8 can be used to compute

$$s^{(1)}_{11111}[X;t] = s^{(2)}_{11111}[X;t] + (t^3 + t^4)s^{(2)}_{2111}[X;t] + t^6 s^{(2)}_{221}[X;t].$$

Springer

http://www.springer.com/978-1-4939-0681-9

k-Schur Functions and Affine Schubert Calculus
Lam, Th.; Lapointe, L.; Morse, J.; Schilling, A.;
Shimozono, M.; Zabrocki, M.
2014, VIII, 219 p. 126 illus., Hardcover
ISBN: 978-1-4939-0681-9