SEMI-STANDARD YOUNG TABLEAUX AND $\mathfrak{sl}_n$ CRYSTALS

PETER TINGLEY (TEXED BY DAVID JORDAN)

Today we will explicitly describe the combinatorics of $\mathfrak{sl}_n$ crystals in terms of semi-standard Young tableaux. As an application, we will discuss Littlewood-Richardson (LR) rules. This roughly follows [K2, Section 5], where citations to original sources can be found.

1. The semi-setandard Young tableau realization

Recall that a semi-standard Young tableaux of shape λ for $\mathfrak{sl}_n$ is a filling of λ with the numbers $\{1, \ldots, n\}$, which is weakly increasing in rows. and strictly increasing in columns. For example,

$$\begin{array}{|c|cccccccc}
\hline
5 \\
\hline
3 & \multicolumn{1}{|c|}{4} & \multicolumn{1}{c|}{5} \\
\hline
2 & \multicolumn{1}{|c|}{2} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} & \multicolumn{1}{c|}{4} & \multicolumn{1}{c|}{5} \\
\hline
1 & \multicolumn{1}{|c|}{1} & \multicolumn{1}{c|}{1} & \multicolumn{1}{c|}{2} & \multicolumn{1}{c|}{2} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} \\
\hline
\end{array}\ .$$

Call the set of such tableau $SSYT_n(\lambda)$. If λ has more then n rows, there are no such tableaux. In fact, we often assume λ has at most $n-1$ rows, as if it has n rows then the columns of height n must all be filled with exactly $1, 2, \ldots n$, and can essentially be ignored. The difference between allowing columns of height n and not is essentially the difference between working with $\mathfrak{gl}_n$ representations and working with $\mathfrak{sl}_n$ representations.

Theorem 1.1. *The set $SSYT_n(\lambda)$ parameterizes the $\mathfrak{sl}_n$ crystal $B(\lambda)$, where the highest weight λ is expressed in terms of fundamental weights by*

$$\lambda \mapsto \sum (\lambda_j - \lambda_{j-1})\omega_j.$$

In order for this theorem to be meaningful (beyond saying the the number of semi-standard Young tableau of shape λ is the dimension of $V(\lambda)$), we must describe the action of the crystal operators. Each f_k will change a "k" to a "$k+1$", or else set the tableau to 0. Let us describe the algorithm by example, rather than by formulas. Let us apply f_2 to the previous example. We begin by creating a parentheses sequence, by putting a "(" above each "2", and a ")" above each "3", so that they cancel as parentheses do. If "2"s and "3"s appear in the same column, the "("s appear before the ")", but such brackets always cancel, so many people do not include them. Here they will be colored red.

$$\begin{array}{cccccccc}
() &) & (& () &) & (& (& \\
\end{array}$$

$$\begin{array}{|c|cccccccc}
\hline
5 \\
\hline
3 & \multicolumn{1}{|c|}{4} & \multicolumn{1}{c|}{5} \\
\hline
2 & \multicolumn{1}{|c|}{2} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} & \multicolumn{1}{c|}{4} & \multicolumn{1}{c|}{5} \\
\hline
1 & \multicolumn{1}{|c|}{1} & \multicolumn{1}{c|}{1} & \multicolumn{1}{c|}{2} & \multicolumn{1}{c|}{2} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{3} & \multicolumn{1}{c|}{4} \\
\hline
\end{array}$$

We then look for the rightmost unmatched ")", and change the "2" inside to a "3". In the previous example we end up with the tableau:

Date: March 18, 2011.

$$\begin{array}{llllllll}
\boxed{5} \\
\boxed{3} & \boxed{4} & \boxed{5} \\
\boxed{2} & \boxed{3} & \boxed{3} & \boxed{3} & \boxed{4} & \boxed{4} & \boxed{5} \\
\boxed{1} & \boxed{1} & \boxed{1} & \boxed{2} & \boxed{2} & \boxed{3} & \boxed{3} & \boxed{4}
\end{array}.$$

If there were no uncanceled), f_2 would kill the tableaux (i.e. send it to 0). The rules for e_i are similar: you form the same string of brackets, and change the $i+1$ corresponding to the leftmost uncanceled (to an i.

To see why the above gives the crystal $B(\lambda)$, we will show that it can be embedded in a tensor product of many copies of the crystal $B(\omega_1)$ corresponding to the vector representation. It is easy to check that the crystal $B(\omega_1)$ looks like

$$\boxed{1} \xrightarrow{F_1} \boxed{2} \xrightarrow{F_2} \cdots\cdots \xrightarrow{F_{n-1}} \boxed{n}.$$

Consider the map $SSYT_n(\lambda) \to B(\omega_1)^{\otimes|\lambda|}$ which simply reads the entries of λ moving up columns then right to left. For example,

$$\begin{array}{c}
\begin{array}{llllllll}
\boxed{5} \\
\boxed{3} & \boxed{4} & \boxed{5} \\
\boxed{2} & \boxed{2} & \boxed{3} & \boxed{3} & \boxed{4} & \boxed{4} & \boxed{5} \\
\boxed{1} & \boxed{1} & \boxed{1} & \boxed{2} & \boxed{2} & \boxed{3} & \boxed{3} & \boxed{4}
\end{array} \\
\downarrow \\
\boxed{4} \otimes \boxed{3} \otimes \boxed{5} \otimes \boxed{3} \otimes \boxed{4} \otimes \boxed{2} \otimes \boxed{4} \otimes \boxed{2} \otimes \boxed{3} \otimes \boxed{1} \otimes \boxed{3} \otimes \boxed{5} \otimes \boxed{1} \otimes \boxed{2} \otimes \boxed{4} \otimes \boxed{1} \otimes \boxed{2} \otimes \boxed{3} \otimes \boxed{5}.
\end{array} \tag{1.2}$$

Recall the tensor product rule for crystals from Lecture 2. This can be rephrased algebraically as follows: Let $b \in B(\lambda), c \in B(\mu)$ then

$$f_i(b \otimes c) = \begin{cases} f_i(b) \otimes c & \text{if } \varphi_i(b) > \varepsilon_i(c) \\ b \otimes f_i(c) & \text{if } \varphi_i(b) \leq \varepsilon_i(c). \end{cases} \tag{1.3}$$

Even more useful to us, it can be displayed as follows: above b, put a string of brackets $))\cdots))((\cdots((,$ where the number of) is $\varepsilon(b)$ and the number of (is $\varphi(b)$. Do the same for c, and concatenate to get one string of brackets

$$\begin{array}{ccc}))\cdots((& &))\cdots((\\ b & \otimes & c. \end{array}$$

If the leftmost uncanceled (is in the string above b, then $f_i(b \otimes c) = f_i(b) \otimes c$. Otherwise, $f_i(b \otimes c) = b \otimes f_i(c)$.

At this point, it should be clear that the map from (1.2) intertwines the f_i defined on $SSYT(\lambda)$ with the f_i defined on the tensor product of the $B(\omega_1)$ (caution: because of poor choices of conventions, one string of brackets is actually the reverse of the other). Thus our rules define an element of the unique closed family of crystals, and by checking the highest weight we see that it is $B(\lambda)$. In fact, the above version of the tensor product rule is probably the main reason that canceling brackets occur so often in definitions of crystal operators.

2. Jeu de Taquin

One might want to go the other way, and produce a tableaux from a sequence. This can be done by first creating a skew tableau from the sequence, and then performing the "Jeu de Taquin." For example,

$$\boxed{3} \otimes \boxed{1} \otimes \boxed{2} \quad \to \quad \begin{array}{ccc} \boxed{2} & & \\ & \boxed{1} & \\ & & \boxed{3} \end{array} \xrightarrow{\text{Jeux de Taquin}} \begin{array}{ll} \boxed{2} & \\ \boxed{1} & \boxed{3} \end{array}.$$

The "Jeu de Taquin" changes a skew semi-standard tableau to an ordinary semi-standard tableau by the following algorithm

(1) choose some box on the boundary of the inner shape.
(2) Interchange that box with either the box above it or the box to the right. Exactly one of these will be possible without creating something with is no longer semi-standard (i.e. weakly increasing along rows and strictly increasing along columns).
(3) Again, interchange the same box with either the box above it or the box to the right, whichever is possible. Continue doing this until the box is an outer corner. Then delete it.
(4) Choose another box on the boundary of the inner shape, and repeat the above procedure.
(5) Continue until you have an ordinary tableau.

It turns out the the resulting tableau does not depend on the choices made. One way to see this is to show that performing Jeu-de-taquin commutes with crystal operators (defined on skew tableau by the obvious generalization).

Note however that many different elements of $B(\omega_1)^{\otimes N}$ can be sent to the same tableau by this procedure. This is because there are usually many copies of a given $B(\lambda)$ in $B(\omega_1)^{\otimes|\lambda|}$. A related issue is that the rule for applying f_i to a tableaux can be modified by reading the boxes in various other orders, without changing the final operator. For instance, you can read right to left along rows, then bottom to top.

3. Littlewood-Richardson Rules

A classical problem is to compute the Littlewood-Richardson (LR) coefficient $c^{\gamma}_{\lambda,\mu}$, which is the multiplicity of $V(\gamma)$ in $V(\lambda) \otimes V(\mu)$. We will discuss how crystal theory and the combinatorics of Young tableau can be used to give an answer to this question. Consider the example

$$\gamma = \begin{array}{llll} \square & \square & & \\ \square & \square & \square & \\ \square & \square & \square & \square \end{array}, \quad \lambda = \begin{array}{ll} \square & \\ \square & \\ \square & \square \end{array}, \quad \mu = \begin{array}{lll} \square & \square & \\ \square & \square & \square \end{array},$$

say for $\mathfrak{sl}_4$. To make the rule precise, it is better to consider representations of $\mathfrak{gl}_4$, so that highest weights can have columns of height 4. $c^{\gamma}_{\lambda\mu}$ will always be 0 unless $|\gamma| = |\lambda| + |\mu|$. Representations whose highest weights differ by adding columns of height 4 are isomorphic an $\mathfrak{sl}_4$ representations, but not as $\mathfrak{gl}_n$ representations.

We need only count the dimension of the space of highest weight vectors of weight γ that appear. Using crystal theory, we should count the number of highest weight elements of weight γ that appear in $B(\lambda) \otimes B(\mu)$. Such a highest weight vector must by of the form $b_\lambda \otimes c$ for some c in $B(\mu)$, where

$$b_\lambda = \begin{array}{ll} \boxed{3} & \\ \boxed{2} & \\ \boxed{1} & \boxed{1} \end{array}$$

is the highest weight element in $B(\lambda)$ (highest weight elements always have this form). Since the highest weight vector b_γ is given by the tableau

$$b_\gamma = \begin{array}{llll} \boxed{3} & \boxed{3} & & \\ \boxed{2} & \boxed{2} & \boxed{2} & \\ \boxed{1} & \boxed{1} & \boxed{1} & \boxed{1} \end{array},$$

we must count the number of tableaux of type $\begin{array}{|c|c|c|}\cline{1-2} & & \multicolumn{1}{c}{} \\ \hline & & \\ \hline\end{array}$ filled with $1, 1, 2, 2, 3$ (so that the weight of the tensor product is right), such that the product is highest weight. There are two tableaux of this shape:

$$\begin{array}{|c|c|c|}\cline{1-2} 2 & 3 & \multicolumn{1}{c}{} \\ \hline 1 & 1 & 2 \\ \hline\end{array}\,, \quad \text{and} \quad \begin{array}{|c|c|c|}\cline{1-2} 2 & 2 & \multicolumn{1}{c}{} \\ \hline 1 & 1 & 3 \\ \hline\end{array}\,. \tag{3.1}$$

The first satisfies the property $b_\lambda \otimes c$ is highest weight, while the second does not. So in this case $c^\gamma_{\lambda,\mu} = 1$.

More often, LR rules are described as follow: $c^\gamma_{\lambda,\mu}$ is the number of skew tableaux of shape γ/λ, satisfying the additional condition that, if you read the entries right to left along rows, starting with the bottom row then moving up, any initial sequence has at least as many i as $i + 1$, for all i. Such tableaux are called Littlewood-Richardson tableaux. The one corresponding to the above LR-coefficient of 1 is

$$\begin{array}{|c|c|c|}\cline{1-1} 2 & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\ \cline{1-2} 1 & 2 & \multicolumn{1}{c}{} \\ \hline \multicolumn{1}{c|}{} & 1 & 1 \\ \cline{2-3}\end{array}\,.$$

The correspondence between these two versions of the rule is that row r of the LR-tableaux records the heights of all the r appearing in the corresponding tableaux from (3.1). The condition on initial sequences exactly translates to the tableaux from (3.1) actually being semi-standard.

References

[K2] Masaki Kashiwara. On crystal bases, *Representations of groups (Banff, AB, 1994)*, 155-197, CMS Conf. Proc. **16**, Amer. Math. Soc., Providence, RI, 1995.