MATH 821, Spring 2013, Lecture 20

Karen Yeats
(*Scribe: Amy Wiebe*)

March 28, 2013

1 Robinson-Schensted-Knuth

Recall from Lecture 19 what it meant to insert i into tableau T:
`To begin, current row is top row`

(1) `Find smallest` $j > i$ `in the current row, if it exists.`

(2) `If no such` j `exists, add a new box to the end of current row and put` i `in it.`

(3) `Otherwise, replace` j `by` i`, replace current row by next row and go to (1)`

Theorem 1. *There is a bijection between length n lists of pairs of positive integers in lexicographic order (equivalent to multisets of such pairs) and pairs (P, Q) of semistandard Young tableaux of the same shape λ, where λ is a partition of n.*

For semistandard Young tableaux, entries may not be distinct, so we need to adjust the insertion routine, as follows:
`To begin, current row is top row`

(1) `Find` ***`leftmost`*** $j > i$ `in the current row, if it exists.`

(2) `If no such` j `exists, add a new box to the end of current row and put` i `in it.`

(3) `Otherwise, replace` j `by` i`, replace current row by next row and go to (1)`

The Robinson-Schensted-Knuth algorithm:

Given a list L of pairs of positive integers in lexicographic order
`Set` P_0, Q_0 `to be empty tableaux`
`for` $\begin{bmatrix} i \\ j \end{bmatrix}$ `running through` L `with index` k

- Insert j into P_{k-1}, and call the result P_k.
- In the position where P_k, P_{k-1} differ in shape, add a new box to Q_{k-1} and put i in it. Call the result Q_k.

Return $P_{|L|}, Q_{|Q|}$.

Example.

$$L = \begin{pmatrix} 1 & 1 & 1 & 2 & 2 & 3 & 3 \\ 1 & 3 & 3 & 2 & 2 & 1 & 3 \end{pmatrix}$$

k	i	j	P_k	Q_k
1	1	1	1	1
2	1	3	1 3	1 1
3	1	3	1 3 3	1 1 1
4	2	2	1 2 3 3	1 1 1 2
5	2	2	1 2 2 3 3	1 1 1 2 2
6	3	1	1 1 2 2 3 3	1 1 1 2 2 3
7	3	3	1 1 2 3 2 3 3	1 1 1 3 2 2 3

Now it's not as obvious that insertion maintains the tableau property, nor that it's reversible. Let's check!

Lemma 1. *Say we insert i into a semistandard Young tableau T, and say that the bumped entry in row j was b_j originally in column c_j. Then*

$$\begin{array}{rcl} & & c_1 \geq c_2 \geq \cdots \geq c_n \\ i & < & b_1 < b_2 < \cdots < b_n \end{array}$$

Proof. The inequality on the bs is by definition of insertion.

For the c_j, consider the element in row $j+1$, column c_j. Either it doesn't exist, or it is $> b_j$. Then when b_j is placed in row $j+1$ it cannot be placed to the right of this box, so $c_{j+1} \leq c_j$. □

Corollary 2. *Inserting in a semistandard Young tableau yields a semistandard Young tableau.*

Proof. By construction, the rows are weakly increasing. By Lemma 1, the element in row j, column c_{j+1} is $< b_j$, because it always was, or it was what bumped b_j. The element in row $j+2$, col c_{j+1} is $> b_{j+1}$ or it is b_{j+1} after being bumped, which in either case is $> b_j$. □

Lemma 2. *The copies of i in Q are placed in Q strictly from left to right over the course of the algorithm.*

Proof. Say the input to the algorithm was $\begin{pmatrix} \cdots & i & i & \cdots & i & \cdots \\ & j_r \leq & j_{r+1} \leq & \cdots & j_s & \end{pmatrix}$. Let the sequence of boxes involved in the bumps when inserting a given element be called the bumping path. The bumping path for j_t in P_{t-1} lies strictly to the right of the bumping path for j_{t-1}. This is because $j_t \geq j_{t-1}$, so j_t must bump something further to the right than j_{t-1}, so on the next row $b_{t,1} \geq b_{t-1,1}$. Furthermore, the bumping path for j_t will terminate in a row no lower than the bumping path for j_{t-1}, because in the row of P_{t-1} where j_{t-1} terminated, the last box is $b_{t-1,m}$ for some m, so either the path for b_t already terminated or it will also terminate in this row since $b_{t,m} \geq b_{t-1,m}$. □

Corollary 3. *Q_k is a semistandard Young tableau at the end of each step of the algorithm.*

Proof. When i is added to Q_{k-1}, the content of Q_{k-1} consists of elements $< i$ and possibly copies of i. By Lemma 2 any copies of i are in distinct columns from the column where the new i is inserted. □

Corollary 4. *The Robinson-Schensted-Knuth algorithm is reversible.*

Proof. By Lemma 2, we can identify the order in which the boxes containing i were inserted. Then we can reverse in the same manner as we did for the Robinson-Schensted algorithm. □

2 Littlewood-Richardson coefficients

Theorem 5.

$$\prod_{i,j=1}^{\infty} \left(\frac{1}{1 - t x_i y_i} \right) = \sum_{n=0}^{\infty} \sum_{\lambda \text{ partition of } n} s_\lambda(\underline{x}) s_\lambda(\underline{y}) t^n$$

Proof. This is the Robinson-Schensted-Knuth correspondence phrased in terms of generating functions.

LHS: for a single term we have

$$\frac{1}{1 - t x_i y_i} = 1 + t x_i y_i + t^2 x_i^2 y_i^2 + \cdots .$$

View $t^k x_i^k y_j^k$ as saying there are k copies of $\left[\begin{array}{c} i \\ j \end{array}\right]$ in the list of pairs, or equivalently there is a k in the i,j position in the matrix of nonnegative integers. So

$$\prod_{i,j=1}^{\infty} \frac{1}{1-tx_iy_i}$$

counts all multisets of pairs of positive integers, equivalently all matrices with nonnegative integer entries, only finitely many of which are nonzero.

RHS: by definition of Schur functions we have

$$\sum_{n=0}^{\infty} \sum_{\lambda \text{ partition of } n} s_\lambda(\underline{x}) s_\lambda(\underline{y}) t^n = \sum_{n=0}^{\infty} \sum_{\substack{P,Q \text{ semistandard} \\ \text{shape } \lambda \text{ tableaux,} \\ \lambda \text{ partition of } n}} t^n \underline{x}^{\mathrm{cont}(P)} \underline{y}^{\mathrm{cont}(Q)}.$$

So by the Robinson-Schensted-Knuth correspondence, the LHS and RHS are equal. □

Corollary 6. *Write*

$$s_\mu s_\nu = \sum_\lambda c^\lambda_{\mu,\nu} s_\lambda$$

and

$$\Delta(s_\lambda) = \sum_{\mu,\nu} \hat{c}^\lambda_{\mu,\nu} s_\mu \otimes s_\nu.$$

Then $c^\lambda_{\mu,\nu} = \hat{c}^\lambda_{\mu,\nu}$.

Proof. By Theorem 5,

$$\begin{aligned}
\left(\prod_{i,j=1}^{\infty}\left(\frac{1}{1-tx_iz_i}\right)\right)\left(\prod_{i,j=1}^{\infty}\left(\frac{1}{1-ty_iz_i}\right)\right) &= \left(\sum_{n=0}^{\infty}\sum_{\lambda \text{ partition of } n} s_\lambda(\underline{x})s_\lambda(\underline{z})t^n\right)\left(\sum_{n=0}^{\infty}\sum_{\lambda \text{ partition of } n} s_\lambda(\underline{y})s_\lambda(\underline{z})t^n\right) \\
&= \sum_{\mu,\nu} t^{|\mu|+|\nu|} s_\mu(\underline{x}) s_\nu(\underline{y}) s_\mu(\underline{z}) s_\nu(\underline{z}) \\
&= \sum_{\mu,\nu} t^{|\mu|+|\nu|} s_\mu(\underline{x}) s_\nu(\underline{y}) \sum_\lambda c^\lambda_{\mu,\nu} s_\lambda(\underline{z}). \qquad (1)
\end{aligned}$$

Now

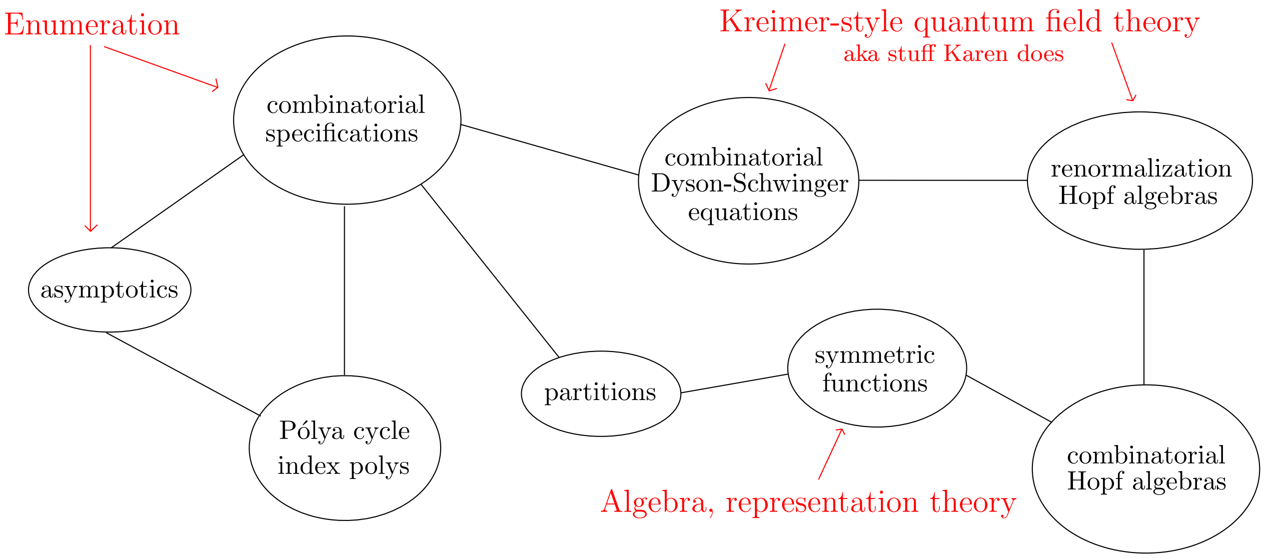
$$\begin{aligned}
\left(\prod_{i,j=1}^{\infty}\left(\frac{1}{1-tx_iz_i}\right)\right)\left(\prod_{i,j=1}^{\infty}\left(\frac{1}{1-ty_iz_i}\right)\right) &= \prod_{\substack{w_i \text{ running} \\ \text{over } (\underline{x},\underline{y}), \\ z_i \text{running} \\ \text{over } \underline{z}}} \frac{1}{1-tw_iz_i} \\
&= \sum_\lambda t^{|\lambda|} s_\lambda(\underline{x},\underline{y}) s_\lambda(\underline{z}), \qquad \text{by Theorem 5} \\
&= \sum_\lambda t^{|\lambda|} \sum_{\mu,\nu} \hat{c}^\lambda_{\mu,\nu} s_\mu(\underline{x}) s_\nu(\underline{y}) s_\lambda(\underline{z}), \quad \text{as } s_\lambda(\underline{x},\underline{y}) = \Delta(s_\lambda) \ (2)
\end{aligned}$$

and taking the coefficient of $s_\mu(\underline{x})s_\nu(\underline{y})s_\lambda(\underline{z})$ in (1) and (2), (and noting that $|\mu| + |\nu| = |\lambda|$ since $\cdot$ and Δ are graded) we get $c^\lambda_{\mu,\nu} = \hat{c}^\lambda_{\mu,\nu}$. □

As a consequence, we get that Λ is self-dual.
The $c^\lambda_{\mu,\nu}$ are called Littlewood-Richardson coefficients. They have combinatorial interpretations, etc., but that's another story.

3 What did we do this semester?

An ecclectic collection of combinatorics, classic and new.



THE END!