LECTURE 18: MONK'S RULE

MOHAMED OMAR

1. Monk's Rule & Insertion Algorithm

Theorem 1.1 (Monk's Rule)**.** *Let $w \in S_\infty$, and s_r be a simple transposition. Then*

$$\sigma_w \cdot \sigma_{s_r} = \sum_{\substack{k \leq r < l \\ l(wt_{kl}) = l(w)+1}} \sigma_{wt_{kl}}.$$

The proof of Monk's Rule will follow from a combinatorial bijection coming from an Insertion Algorithm on rc-graphs. In this lecture, we describe this algorithm and show how the combinatorial consequences imply Monk's Rule.

We begin by detailing the input and output of the Insertion Algorithm. For brevity, we will say that (i, j) exhibits property $(\star)$ if at the i^{th} row and j^{th} column of the rc-graph, the two strands do not make a cross, and the upper strand and lower strand start at s and t respectively with $s \leq r < t$.

Input: $D \in RC(w)$, and $r, i \in \mathbb{Z}$ with $0 < i \leq r$.

Output: $I_r(D, i) = (D', k, l)$ with $k, l \in \mathbb{N}$, $D' \in RC(w')$ with $l(w') = l(w) + 1$.

(1) $i_0 = i, j_0$ is maximal such that (i_0, j_0) is as in $(\star)$; add a crossing at (i_0, j_0) to D. Set $s_0 = s, t_0 = t$
(2) If the result is an rc-graph, STOP.
(3) Else:
  (a) The two strands cross again at $(i_1, j_1^{'})$. Delete the second crossing at $(i_1, j_1^{'})$ from D.
  (b) Find $j_1 < j_1^{'}$ maximal such that (i_1, j_1) is an in $(\star)$
  (c) Add a crossing at (i_1, j_1). Set (s_1, t_1) to be the corresponding labels of the beginnings of the strands at (i_1, j_1)
  (d) GOTO Step 2.

The rc-graph D' obtained from this algorithm is the result of applying p add and delete steps beginning at the rc-graph D, and a final addition step upon exiting the algorithm. We say that the **insertion path** of this algorithm is the sequence of row-column index pairs $(i_0, j_0), (i_1, j_1^{'}), (i_1, j_1), (i_2, j_2^{'}), \ldots, (i_p, j_p)$. Notice that the indices in the insertion path satisfy the two conditions:

$$i = i_0 > i_1 > \cdots > i_p.$$

$$j_0 < j_1^{'} > j_1 < j_2^{'} > j_2 < \cdots < j_p^{'} > j_p.$$

Date: February 18, 2009.

Now recall that the output of the Insertion Algorithm is a triple (D', k, l) with $D' \in RC(w')$. The statistics k and l are precisely s_p and t_p respectively. Letting t_{kl} be the transposition exchanging k and l, we set $w' = wt_{kl}$.

We now illustrate an example of this algorithm. Our input will be:

$$D \in RC([135642]) \quad (s_0, t_0) = (3, 5) \quad r = 4 \quad i = 3$$

We shall use the recording table for the rc-graphs, as they completely determine the rc-graph structure. In this example, our recording table is

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

Since $i = 3$, we see that $(i_0, j_0) = (3, 2)$ because j_0 is maximal under condition $(\star)$. Thus we add a crossing at $(3, 2)$ to give us

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

This is not an rc-graph since the two strands at $(3, 2)$ cross again at $(1, 4)$, so $(i_1, j_1^{'}) = (1, 4)$. From this, $(i_1, j_1) = (1, 3)$ and hence $(s_1, t_1) = (2, 5)$. We delete the crossing at $(1, 4)$ and add the crossing at $(1, 3)$. This gives us an rc-graph with the table below:

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

Thus the output is (D', k, l) with $k = 2, l = 5$ and D' the rc-graph given by the table above.

A nice consequence of the Insertion Algorithm is the following proof of Monk's Rule.

1.1. **Proof of Monk's Rule.** By the algorithm, we have a monomial preserving bijection between

$$RC(w) \times RC(s_r) \longrightarrow \bigcup_{\substack{w' = wt_{kl} \\ k \leq r < l \\ l(w') = l(w)+1}} RC(w')$$

given by the map that sends $(D, i) \mapsto I_r(D, i)$. The result then follows by the correspondence of rc-graphs and Schubert polynomials.

2. Stanley Symmetric Functions

Let $\{u_1^{'}, \ldots, u_{n-1}^{'}\}$ be the generators of the nil Coxeter algebra. Recall the functions $h_i(x) = 1 + xu_i^{'}$, and $H_1(x) = h_{n-1}(x)h_{n-2}(x)\cdots h_1(x)$.

Definition 2.1 (Stanley Symmetric Function)**.** Let $w \in S_n$. Define

$$F_w(x) = \lim_{s\to\infty} \sigma_{1^s \times w} = \langle H_1(x_1)H_1(x_2)\cdot\cdots 1, w\rangle$$

The action and inner products in the above definition are defined as follows. For any $u_i^{'}$ and $w \in S_n$,

$$u_i^{'} \cdot w = \begin{cases} s_i w & \text{if } l(s_i w) = l(w) + 1 \\ 0 & \text{else} \end{cases}$$

The inner product is defined by $\langle v, w\rangle = \delta_{v,w}$. Also recall we can re-write $F_w(x)$ as

$$F_w(x) = \sum_{\substack{u=(a_1,\ldots,a_t) \\ \text{a composition}}} \langle A_{a_t}(u)\cdots A_{a_1}(u)\cdot 1, w\rangle x_1^{a_1}\cdots x_t^{a_t}.$$

where for any positive integer k, $A_k(u) = \sum_{b_k<\cdots<b_1} u_{b_1}\cdots u_{b_k}$. We now describe some properties of Stanley symmetric functions.

(1) $[x_1x_2\cdots x_{l(w)}]F_w(x) =$ The number of reduced words for w.
(2) $F_w(x)$ is symmetric (since the functions H_i and A_i are).
(3) $F_w(x) = \sum_\lambda \langle s_{\lambda^t}(u)\cdot 1, w\rangle s_\lambda(x)$. (The $s_{\lambda^t}(u)$ functions are non-commutative Schur functions).
(4) $\langle s_{\lambda^t}(w)\cdot 1, w\rangle = c_\lambda^w =$ the number of Semistandard Young Tableaux T of shape λ^t such that $w(T)\cdot 1 = w$.

Statements (3) and (4) will be proved next time.