

# The Institute of Mathematical Sciences

CIT Campus, Taramani, Chennai 600113, India.

AMRITANSHU PRASAD  
Professor H  
amri@imsc.res.in

Phone: 044 2254 3207  
Fax: 044 2254 1586  
Grams: MATSCIENCE

---

13<sup>th</sup> June, 2018

The Editors  
Discrete Mathematics & Theoretical Compute Science

Dear Sir/Madam,

## Submission to DMTCS

I hereby submit my article titled “A Timed Version of the Plactic Monoid” to be considered for publication in *DMTCS*. I certify that this article is not under consideration at any other journal.

Some of the results of this article were announced at the 83rd annual conference of the Indian Mathematical Society, held at Sri Venkateshwara University, Tirupati where I delivered the Srinivasa Ramanujan Memorial award lecture. A short expository article based on that lecture is to appear in a forthcoming issue of *The Mathematics Student*, a magazine for undergraduate and masters students that is published by the society. That expository article contains statements of some of the results in this paper, but does not contain technical details or proofs that appear here. A copy of the aforementioned article is attached.

Yours Sincerely,

AMRITANSHU PRASAD

# Knuth's moves on Timed Words

Amritanshu Prasad

ABSTRACT. We give an exposition of Schensted's algorithm to find the length of the longest increasing subword of a word in an ordered alphabet, and Greene's generalization of Schensted's results using Knuth equivalence. We announce a generalization of these results to timed words.

## 1. Introduction

The theory of Young tableaux lies at the cross-roads of modern combinatorics, the theory of symmetric functions, enumerative problems in geometry, and representation theory (see [Ful97, Man98, Pra15, Pra18a]). Young tableaux are named after Alfred Young, who introduced them in his study of the representation theory of symmetric groups, which he called *substitutional analysis* [You00]. Young tableaux played an important role in the proof of the Littlewood-Richardson rule. This is a rule for computing the Littlewood-Richardson coefficients  $c_{\mu\nu}^\lambda$ , which arise as (for details, see [Ful97, Man98]):

- the multiplicity of the irreducible polynomial representation  $W_\lambda$  of  $GL_n(\mathbb{C})$  in a tensor product  $W_\mu \otimes W_\nu$ .
- the coefficient of the Schur polynomial  $s_\lambda$  in the expansion of a product  $s_\mu s_\nu$  of Schur polynomials.
- the number of points of intersection of Schubert varieties  $X_\mu$ ,  $X_\nu$  and  $X_\lambda$  in general position.

Robinson [Rob38] outlined an approach to the Littlewood-Richardson rule based on Young tableaux, which was perfected in the work of Lascoux and Schützenberger [LS78] forty years later. In this expository article, our point of departure is Schensted's observation [Sch61] that Robinson's construction of the insertion tableau of a word can be used as an algorithm to determine the longest increasing subword of a word in an ordered language. Schensted used this to give a formula for the number of words with longest increasing subword of a given length. Schensted's results were generalized by Greene [Gre74] using relations which were introduced by Knuth [Knu70] to characterize the class of words with a given insertion tableau.

In a different context, Alur and Dill [AD94] introduced timed words as a part of their description of timed automata. Timed automata are generalizations of finite automata, and are used for the formal verification of real-time systems. The author has extended Greene's theorem to timed words [Pra18b], with the goal of providing a framework to study piecewise linear versions of bijective correspondences involving Young tableau, such as the ones studied by Berenstein and Kirillov [Kir11]. The

salient features of this extension are outlined here. Detailed proofs, technical details, and applications to piecewise-linear bijections will appear in [Pra18b].

## 2. Schensted's Algorithm

**2.1. Words.** Let  $A_n$  denote the set  $\{1, \dots, n\}$ , which we regard as an *ordered alphabet*. A *word* in  $A_n$  is a finite sequence  $w = c_1 \cdots c_k$  of elements of  $A_n$ . The set of all words in  $A_n$  is denoted  $A_n^*$ . A *subword* of  $w = c_1 \cdots c_k$  is defined to be a word of the form

$$w' = c_{i_1} \cdots c_{i_m}, \text{ where } 1 \leq i_1 < \cdots < i_m \leq k.$$

The subword  $w'$  is said to be *weakly increasing* if  $c_{i_1} \leq \cdots \leq c_{i_m}$ .

Consider the following computational problem:

Given a word  $w \in A_n^*$ , determine the maximal length of a weakly increasing subword of  $w$ .

**2.2. Tableaux.** Schensted [Sch61] gave an elegant algorithm to solve the preceding computational problem. His algorithm makes one pass over the word. At each stage of its running, it stores a combinatorial object called a *semistandard Young tableau* (see Section 2.2.1). This tableau is modified as each successive letter of the word is read. The length of the longest increasing subword can be read off from the tableau (see Sections 2.5 and 2.6) obtained when all of  $w$  has been read.

**DEFINITION 2.2.1** (Semistandard Young Tableau). A semistandard Young tableau in  $A_n$  is a finite arrangement of integers from  $A_n$  in rows and columns so that the numbers increase weakly along rows, strictly along columns, so that there is an element in the first row of each column, there is an element in the first column of each row, and there are no gaps between numbers.

Let  $l$  be the number of rows in the tableau, and for each  $i = 1, \dots, l$ , let  $\lambda_i$  be the length of the  $i$ th row. Then  $\lambda = (\lambda_1, \dots, \lambda_l)$  is called the *shape* of the tableau.

**EXAMPLE 2.2.2.** The arrangement

1	1	5
2	4	
3		

is a semistandard Young tableau of shape  $(3, 2, 1)$  in  $A_5$ .

The notion of a semistandard Young tableau is a generalization of *Young tableau*, which was introduced by Young [You00, p. 133]. In Young's version, each element of  $A_n$  occurs exactly once in the tableau. For brevity, we shall henceforth use the term *tableau* to refer to a semistandard Young tableau.

**2.3. Row Insertion.** A word  $c_1 c_2 \cdots c_k$  in  $A_n^*$  is called a *row* if  $c_1 \leq \cdots \leq c_k$ . Each row of a tableau is a row in the sense of this definition. For each row  $u = a_1 \cdots a_k \in A_n^*$ , define the *row insertion* of  $a$  into  $u$  by:

$$\text{RINS}(u, a) = \begin{cases} (\emptyset, a_1 \cdots a_k a) & \text{if } a_k \leq a, \\ (a_j, a_1 \cdots a_{j-1} a a_{j+1} \cdots a_k) & \text{otherwise, with} \\ & j = \min\{i \mid a < a_i\}. \end{cases}$$

Here  $\emptyset$  should be thought of as an empty word of length zero.

**EXAMPLE 2.3.1.**  $\text{RINS}(115, 5) = (\emptyset, 1155)$ ,  $\text{RINS}(115, 3) = (5, 113)$ .

It is clear from the construction that, for any row  $u \in A_n^*$  and  $a \in A_n$ , if  $(a', u') = \text{RINS}(u, a)$ , then  $u'$  is again a row. For convenience set  $\text{RINS}(u, \emptyset) = (\emptyset, u)$ .

**2.4. Tableau Insertion.** Let  $t$  be a tableau with rows  $u_1, u_2, \dots, u_l$ . Then  $\text{INSERT}(t, a)$ , the insertion of  $a$  into  $t$ , is defined as follows: first  $a$  is inserted into  $u_1$ ; if  $\text{RINS}(u_1, a) = (a'_1, u'_1)$ , then  $u_1$  is replaced by  $u'_1$ . Then  $a'_1$  is inserted into  $u_2$ ; if  $\text{RINS}(u_2, a'_1) = (a'_2, u'_2)$ , then  $u_2$  is replaced by  $u'_2$ ,  $a'_2$  is inserted into  $u_3$ , and so on. This process continues, generating  $a'_1, a'_2, \dots, a'_k$  and  $u'_1, \dots, u'_k$ . The tableau  $t' = \text{INSERT}(t, a)$  has rows  $u'_1, \dots, u'_k$ , and a last row (possibly empty) consisting of  $a'_k$ . It turns out that  $\text{INSERT}(t, a)$  is a tableau [Knu70].

EXAMPLE 2.4.1. For  $t$  as in Example 2.2.2, we have

$$\text{INSERT}(t, 3) = \begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 4 & 5 \\ \hline 3 & & \\ \hline \end{array},$$

since  $\text{RINS}(115, 3) = (5, 113)$ ,  $\text{RINS}(24, 5) = (\emptyset, 245)$ .

### 2.5. Insertion Tableau of a Word.

DEFINITION 2.5.1. The insertion tableau  $P(w)$  of a word  $w$  is defined recursively as follows:

- (1)  $P(\emptyset) = \emptyset$
- (2)  $P(c_1 \cdots c_k) = \text{INSERT}(P(c_1 \cdots c_{k-1}), c_k)$ .

EXAMPLE 2.5.2. Take  $w = 3421153$ . Sequentially inserting the terms of  $w$  into the empty tableau  $\emptyset$  gives the sequence of tableaux:

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

and finally, the insertion tableau  $P(w) = \begin{array}{|c|c|c|} \hline 1 & 1 & 3 \\ \hline 2 & 4 & 5 \\ \hline 3 & & \\ \hline \end{array}$ .

**2.6. Schensted's Theorem.** Schensted [Sch61] proved the following:

THEOREM 2.6.1. *The length of the longest increasing subword of any  $w \in A_n^*$  is the length of the first row of  $P(w)$ .*

In other words, the algorithm for constructing the insertion tableau of  $w$  solves the computational problem posed in Section 2.1.

The proof of Schensted's theorem is not very difficult, and the reader is invited to attempt it. The proof is by induction on  $k$ , and uses the observation that the last entry of the first row of  $P(a_1 \cdots a_k)$  is the *least last element* of all maximal length weakly increasing subword of  $a_1 \cdots a_k$ .

**2.7. Greene's Theorem.** The insertion tableau  $P(w)$  obtained from a word  $w$  seems to contain a lot more information than just the length of the longest weakly increasing subword. For example, what do the lengths of the remaining rows of  $P(w)$  signify? The answer to this question was given by Greene [Gre74]. We say that subwords  $c_{i_1} \cdots c_{i_r}$  and  $c_{j_1} \cdots c_{j_s}$  of  $c_1 \cdots c_k$  are *disjoint* if the subsets  $\{i_1, \dots, i_r\}$  and  $\{j_1, \dots, j_s\}$  are disjoint.

**DEFINITION 2.7.1 (Greene Invariants).** The  $r$ th Greene invariant of a word  $w \in A_n^*$  is defined to be the maximum cardinality of a union of  $r$  pairwise disjoint weakly increasing subwords of  $w$ .

**EXAMPLE 2.7.2.** For  $w = 3421153$  from Example 2.5.2, the longest weakly increasing subwords have length 3 (for example, 113 and 345). The subwords 345 and 113 are disjoint, and no pair of disjoint weakly increasing subwords of  $w$  can have cardinality greater than 6. However, the entire word  $w$  is a union of three disjoint weakly increasing subwords (for example 345, 23 and 15). So the Greene invariants of  $w$  are  $a_1(w) = 3$ ,  $a_2(w) = 6$ , and  $a_3(w) = 7$ .

**THEOREM 2.7.3 (Greene).** For any  $w \in A_n$ , if  $P(w)$  has shape  $\lambda = (\lambda_1, \dots, \lambda_l)$ , then for each  $r = 1, \dots, l$ ,  $a_r(w) = \lambda_1 + \dots + \lambda_l$ .

Example 2.7.2 is consistent with Greene's theorem as the shape of  $P(w)$  is  $(3, 3, 1)$  and the Greene invariants are 3,  $6 = 3 + 3$  and  $7 = 3 + 3 + 1$ , respectively.

**2.8. Knuth Equivalence.** Greene's proof of Theorem 2.7.3 is based on the notion of Knuth equivalence. Knuth [Knu70] identified a pair of elementary moves on words:

$$\begin{aligned} (K1) \quad & xzy \equiv zxy \text{ if } x \leq y < z, \\ (K2) \quad & yxz \equiv yzx \text{ if } x < y \leq z. \end{aligned}$$

For example, in the word 4213443, the segment 213 is of the form  $yxz$ , with  $x < y \leq z$ . A Knuth move of type (K2) replaces this segment by  $yzx$ , which is 231. Thus a Knuth move of type (K2) transforms 4213443 into 4231443. Knuth equivalence is the equivalence relation on  $A_n^*$  generated by Knuth moves:

**DEFINITION 2.8.1 (Knuth Equivalence).** Words  $w, w' \in A_n^*$  are said to be Knuth equivalent if  $w$  can be transformed into  $w'$  by a series of Knuth moves (K1) and (K2). If this happens, we write  $w \equiv w'$ .

**EXAMPLE 2.8.2.** The word 3421153 is Knuth equivalent to 3245113:

$$3421153 \equiv_{K2} 3241153 \equiv_{K2} 3214153 \equiv_{K1} 3241153 \equiv_{K1} 3241513 \equiv_{K1} 3245113.$$

At each stage, the letters to which the Knuth moves will be applied to obtain the next stage are highlighted.

**2.9. Reading Word of a Tableau.** Given a tableau, its reading word is obtained by reading its rows from left to right, starting with the bottom row, and moving up to its first row.

**EXAMPLE 2.9.1.** The reading word of the tableau:

1	1	3
2	4	5
3		

is 3245113.

**2.10. Proof of Greene's Theorem.** The proof of Greene's theorem is based on three observations, all fairly easy to prove:

- (1) If  $w$  is the reading word of a tableau of shape  $\lambda = (\lambda_1, \dots, \lambda_l)$ , then  $a_r(w) = \lambda_1 + \dots + \lambda_r$  for  $r = 1, \dots, l$ .
- (2) Every word is Knuth equivalent to the reading word of its insertion tableau.

(3) Greene invariants remain unchanged under Knuth moves.

We illustrate these points with examples (for detailed proofs, see Lascoux, Leclerc and Thibon [LLT02], or Fulton [Ful97]). For the first point, in Example 2.9.1 the first  $k$  rows of the tableau

1	1	3
2	4	5
3		

are indeed disjoint weakly increasing subwords of its reading word of maximal cardinality. For the second point, observe that the sequence of Knuth moves in Example 2.8.2 transform 3421153 to the reading word of its insertion tableau. For the third point, consider the case of the Knuth move ( $K1$ ). A word  $w$  of the form  $w = uxzyv$  is transformed into the word  $w' = uzxyv$ . The only issue is that a weakly increasing subword  $g$  of  $w$  may contain both the letters  $x$  and  $z$ . Then it no longer remains a weakly increasing subword of  $w'$ . However, the subword, being weakly increasing, cannot contain  $y$ , so the  $z$  can be swapped for a  $y$ . This could be a problem if  $y$  is part of another weakly increasing subword  $g'$  in a collection of pairwise disjoint weakly increasing subwords. In that case, we have  $g = g_1 x z g_2$  and  $g' = g'_1 y g'_2$ . We may replace them with  $g_1 x y g'_2$  and  $g'_1 z g_2$ , which would still be weakly increasing, and would have the same total length as  $g$  and  $g'$ .

**2.11. Characterization of Knuth Equivalence.** Knuth equivalence can be characterized in terms of Greene invariants (see [LS78, Theorem 2.15]).

**THEOREM 2.11.1.** *Two words  $w$  and  $w'$  in  $A_n^*$  are Knuth equivalent if and only if  $a_r(uwv) = a_r(uw'v)$  for all words  $u$  and  $v$  in  $A_n^*$ , and all  $r \geq 1$ .*

### 3. Timed Words

**3.1. From Words to Timed Words.** Words, in the sense of Section 2, play an important role in computer science, specifically in the formal verification of systems. Each letter of the alphabet is thought of as an event. A sequence of events is then nothing but a word in  $A_n^*$ . The system is modeled as an automaton having a starting state, and each time an event occurs, its state changes, depending both, on its current state, and the event that has occurred. Following the groundbreaking work of Rabin and Scott [RS59], finite state automata are widely used to model and formally verify the integrity of systems.

For many real-time systems, such as controllers of washing machines, industrial processes, and air or railway traffic control, the *time gaps* between the occurrences of the events modeled by words are as important as the events themselves.

To deal with real-time systems, Alur and Dill [AD94] developed the theory of *timed automata*. A timed automaton responds to a sequence of events that come with time stamps for their occurrence. They represented a sequence of events with time stamps by timed words. We introduce a finite variant of the notion of timed word that they used:

**DEFINITION 3.1.1 (Timed Word).** A timed word in  $A_n$  is a sequence of the form:

$$(3) \quad w = c_1^{t_1} c_2^{t_2} \cdots c_k^{t_k},$$

where  $c_1, \dots, c_k \in A_n$ , and  $t_1, \dots, t_k$  are positive real numbers, and  $c_i \neq c_{i+1}$  for  $i = 1, \dots, k-1$ . The length of the timed word  $w$  above is  $l(w) = t_1 + \cdots + t_k$ .

A sequence (3) where  $c_i = c_{i+1}$  also represents a timed word; segments of the form  $c^{t_1}c^{t_2}$  are replaced by  $c^{t_1+t_2}$  until all consecutive pairs of terms have different letters. The timed word  $w$  in (3) may also be regarded as a piecewise constant left-continuous function  $\mathbf{w} : [0, l(w)) \rightarrow A_n$ , where

$$\mathbf{w}(t) = c_i \text{ if } t_1 + \dots + t_{i-1} \leq t < t_1 + \dots + t_i.$$

The function  $\mathbf{w} : [0, l(w)) \rightarrow A_n$  is called the *function associated to the timed word*  $w$ . We say that the timed word  $w$  is a *timed row* if  $c_1 < \dots < c_k$ . Timed words form a monoid under concatenation. The monoid of timed words in the alphabet  $\{1, \dots, n\}$  is denoted  $A_n^\dagger$ . The map:

$$a_1 \dots a_k \mapsto a_1^1 a_2^1 \dots a_k^1$$

defines an embedding of  $A_n^*$  in  $A_n^\dagger$  as a submonoid.

EXAMPLE 3.1.2. An example of a timed word in  $A_6^\dagger$  of length 7.19 is:

$$w = 3^{0.82}5^{0.08}2^{0.45}6^{0.64}5^{0.94}1^{0.15}5^{0.09}1^{0.52}4^{0.29}1^{0.59}3^{0.97}4^{0.42}2^{0.61}1^{0.07}4^{0.55}$$

Using a color-map to represent the integers 1 to 6,



the timed word  $w$  can be visualized as a colored ribbon:



### 3.2. Subwords of Timed Words.

DEFINITION 3.2.1 (Time Sample). A *time sample* of a word  $w$  is a subset of  $[0, l(w))$  of the form:

$$S = [a_1, b_1) \cup \dots \cup [a_k, b_k),$$

where  $0 \leq a_1 < b_1 < a_2 < b_2 < \dots < a_k < b_k \leq l(w)$ . The length of the time sample  $S$  is  $\sum_i (b_i - a_i)$ , the Lebesgue measure  $\mu(S)$  of  $S$ .

Given a time sample  $S \subset [0, l(w))$ , and  $0 \leq t \leq l(S)$ , the set

$$\{\tilde{t} \mid \mu(S \cap [0, \tilde{t})) = t\}$$

is a closed interval  $[a_t, b_t] \subset [0, l(S))$ . This happens because the function  $t' \mapsto \mu(S \cap [0, t'))$  is a piecewise-linear continuous function on  $[0, l(w)]$  which takes value 0 at  $t' = 0$ , and  $l(S)$  at  $t' = 1$ .

DEFINITION 3.2.2 (Subword of a Timed Word). The subword of a timed word with respect to a time sample  $S \subset [0, l(w))$  is the timed word  $w_S$  of length  $\mu(S)$  whose associated function is given by:

$$\mathbf{w}_S(t) = \mathbf{w}(b_t) \text{ for } 0 \leq t < \mu(S),$$

where  $b_t$  is the largest number in  $[0, l(w))$  such that  $\mu(S \cap [0, \tilde{t})) = t$ .

### 3.3. Timed Tableau.

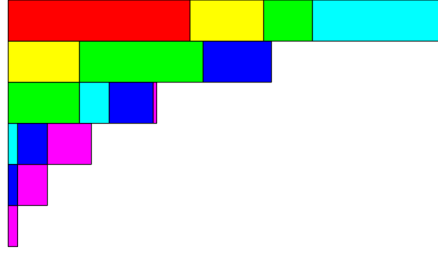
DEFINITION 3.3.1 (Timed Tableau). A timed tableau is a collection  $u_1, u_2, \dots, u_l$  of timed words such that

- (1) Each  $u_i$  is a timed row (in the sense of Section 3.1).
- (2) For each  $i = 1, \dots, l-1$ ,  $l(u_i) \geq l(u_{i+1})$ .
- (3) For each  $i = 1, \dots, l-1$  and  $0 \leq t < l(u_{i+1})$ ,  $u_i(t) < u_{i+1}(t)$ .

EXAMPLE 3.3.2. A timed tableau of shape  $(3.20, 1.93, 1.09, 0.61, 0.29, 0.07)$  is given by:

$$\begin{aligned} t = & 1^{1.33} 2^{0.54} 3^{0.36} 4^{0.97} \\ & 2^{0.52} 3^{0.91} 5^{0.50} \\ & 3^{0.52} 4^{0.22} 5^{0.32} 6^{0.03} \\ & 4^{0.07} 5^{0.22} 6^{0.32} \\ & 5^{0.07} 6^{0.22} \\ & 6^{0.07} \end{aligned}$$

In using the color-map from Section 3.1, it can be visualized as:



The three properties of Definition 3.3.1 are easily perceived from the figure.

DEFINITION 3.3.3 (Reading Word of a Timed Tableau). The reading word of a timed tableau with rows  $u_1, \dots, u_l$  is the timed word

$$u_l u_{l-1} \cdots u_1.$$

EXAMPLE 3.3.4. The reading word of the timed tableau in Example 3.3.2 is  $6^{0.07} 5^{0.07} 6^{0.22} 4^{0.07} 5^{0.22} 6^{0.32} 3^{0.52} 4^{0.22} 5^{0.32} 6^{0.03} 2^{0.52} 3^{0.91} 5^{0.50} 1^{1.33} 2^{0.54} 3^{0.36} 4^{0.97}$ .

**3.4. Timed Insertion.** Given a timed word  $w$  and  $0 \leq a < b \leq l(w)$ , according to Definition 3.2.2,  $w_{[a,b]}$  is the timed word of length  $b - a$  such that:

$$w_{[a,b]}(t) = w(a + t) \text{ for } 0 \leq t < b - a.$$

DEFINITION 3.4.1 (Timed Row Insertion). Given a timed row  $w$ , define the insertion  $\text{RINS}(w, c^{t_c})$  of  $c^{t_c}$  into  $w$  as follows: if  $w(t) \leq c$  for all  $0 \leq t < l(w)$ , then

$$\text{RINS}(w, c^{t_c}) = (\emptyset, wc^{t_c}).$$

Otherwise, there exists  $0 \leq t < l(w)$  such that  $\mathbf{w}(t) > c$ . Let

$$t_0 = \min\{0 \leq t < l(w) \mid \mathbf{w}(t) > c\}.$$



Define

$$\text{RINS}(w, c^{t_c}) = \begin{cases} (w_{[t_0, t_0+t_c]}, w_{[0, t_0]} c^{t_c} w_{[t_0+t_c, l(w)]}) & \text{if } l(w) - t_0 > t_c, \\ (w_{[t_0, l(w)]}, w_{[0, t_0]} c^{t_c}) & \text{if } l(w) - t_0 \leq t_c. \end{cases}$$

It is obvious that the above definition is compatible with the definition of RINS from Section 2.4 when  $u$  is a row in  $A_n^*$ , and  $t_c = 1$ . If  $u = c_1^{t_1} \cdots c_l^{t_l}$  is a timed word, define  $\text{RINS}(w, u)$  by induction on  $l$  as follows: Having defined  $(v', w') = \text{RINS}(w, c_1^{t_1} \cdots c_{l-1}^{t_{l-1}})$ , let  $(v'', w'') = \text{RINS}(w', c_l^{t_l})$ . Then define

$$\text{RINS}(w, u) = (v'v'', w'').$$

EXAMPLE 3.4.2.  $\text{RINS}(1^{1.4}2^{1.6}3^{0.7}, 1^{0.7}2^{0.2}) = (2^{0.7}3^{0.2}, 1^{2.1}2^{1.1}3^{0.5})$ .

DEFINITION 3.4.3 (Timed Tableau Insertion). Let  $w$  be a timed tableau with row decomposition  $u_l \dots u_1$ , and let  $v$  be a timed row. Then  $\text{INSERT}(w, v)$ , the insertion of  $v$  into  $w$  is defined as follows: first  $v$  is inserted into  $u_1$ . If  $\text{RINS}(u_1, v) = (v'_1, u'_1)$ , then  $v'_1$  is inserted into  $u_2$ ; if  $\text{RINS}(u_2, v'_1) = (v'_2, u'_2)$ , then  $v'_2$  is inserted into  $u_3$ , and so on. This process continues, generating  $v'_1, \dots, v'_l$  and  $u'_1, \dots, u'_l$ .  $\text{INSERT}(w, v)$  is defined to be  $v'_l u'_l \cdots u'_1$ . Note that it is quite possible that  $v'_l = \emptyset$ .

EXAMPLE 3.4.4. Take

$$w = 1^{1.4}2^{1.6}3^{0.7} \\ 3^{0.8}4^{1.1},$$

a timed tableau in  $A_5$  of shape  $(3.7, 1.9)$ . Then

$$\text{INSERT}(w, 1^{0.7}2^{0.2}) = 1^{1.7}2^3 3^{0.2} \\ 2^{0.3}3^{1.2}4^{0.4} \\ 3^{0.3}4^{0.7}$$

of shape  $(4.9, 1.9, 1.0)$ .

### 3.5. Insertion Tableau of a Timed Word.

DEFINITION 3.5.1 (Insertion Tableau of a Timed Word). The insertion tableau  $P(w)$  of a timed word  $w$  is defined recursively by the rules:

- (1)  $P(\emptyset) = \emptyset$ ,
- (2)  $P(wc^t) = \text{INSERT}(P(w), c^t)$ .

EXAMPLE 3.5.2. The tableau in Example 3.3.2 is the insertion tableau of the timed word in Example 3.1.2.

### 3.6. Greene Invariants for Timed Words.

DEFINITION 3.6.1 (Greene Invariants for Timed Words). The  $r$ th Greene invariant for a timed word  $w$  is defined as:

$$a_r(w) = \sup \left\{ \mu(S_1) + \cdots + \mu(S_r) \mid \begin{array}{l} S_1, \dots, S_r \text{ are pairwise disjoint time samples} \\ \text{of } w \text{ such that } w_{S_i} \text{ a timed row for each } i \end{array} \right\}.$$

**3.7. Greene's Theorem for Timed Words.** All the ingredients are now in place to state Greene's theorem for timed words:

**THEOREM 3.7.1** (Greene's Theorem for Timed Words). *Let  $w \in A_n^+$  be a timed word. Suppose that  $P(w)$  has shape  $\lambda = (\lambda_1, \dots, \lambda_l)$ , then the Greene invariants of  $w$  are given by:*

$$a_r(w) = \lambda_1 + \dots + \lambda_r \text{ for } r = 1, \dots, l.$$

*For the word  $w$  from Example 3.1.2, the insertion tableau has shape*

$$(3.20, 1.93, 1.09, 0.61, 0.29, 0.07),$$

*(given in Example 3.3.2) so the Greene invariants are given by:*

$$\begin{aligned} a_1(w) &= 3.20 \\ a_2(w) &= 3.20 + 1.93 = 5.13 \\ a_3(w) &= 3.20 + 1.93 + 1.09 = 6.22 \\ a_4(w) &= 3.20 + 1.93 + 1.09 + 0.61 = 6.83 \\ a_5(w) &= 3.20 + 1.93 + 1.09 + 0.69 + 0.29 = 7.12 \\ a_6(w) &= 3.20 + 1.93 + 1.09 + 0.69 + 0.29 + 0.07 = 7.19 \end{aligned}$$

**3.8. Knuth Moves on Timed Words.** As explained in Section 2.10, the proof of Greene's theorem in [Gre74] uses Knuth moves to reduce to the case of reading words of tableau. The main difficulty in generalizing his theorem to timed words is to identify the analogues of Knuth relations  $(K1)$  and  $(K2)$ . These relations need to be simple enough so that it can be shown that if two words differ by such a relation, then they have the same Knuth invariants. At the same time, they need to be strong enough to reduce any timed word to its insertion tableau.

Consider the relations:

$$\begin{aligned} (\kappa_1) \quad & xzy \equiv zxy \text{ when } xyz \text{ is a timed row, } l(z) = l(y), \text{ and } \lim_{t \rightarrow l(y)^-} \mathbf{y}(t) < \mathbf{z}(0), \\ (\kappa_2) \quad & yxz \equiv yzx \text{ when } xyz \text{ is a timed row, } l(x) = l(y), \text{ and } \lim_{t \rightarrow l(x)^-} \mathbf{x}(t) < \mathbf{y}(0). \end{aligned}$$

**EXAMPLE 3.8.1.** We have:

$$w = 5^{1.10} 3^{2.19} 4^{0.89} 5^{1.20} 1^{0.32} 2^{0.44} \equiv w' = 5^{1.10} 3^{2.19} 4^{0.62} 1^{0.32} 2^{0.41} 4^{0.27} 5^{1.20} 2^{0.03},$$

because we may write

$$\begin{aligned} w &= 5^{1.10} 3^{2.08} yzx 2^{0.03}, \\ w' &= 5^{1.10} 3^{2.08} yxz 2^{0.03}, \end{aligned}$$

where  $x = 1^{0.32} 2^{0.41}$ ,  $y = 3^{0.11} 4^{0.62}$ , and  $z = 4^{0.27} 5^{1.20}$ , so  $w$  and  $w'$  differ by a Knuth move of the form  $(\kappa_2)$ .

We say that two timed words  $w$  and  $w'$  are Knuth equivalent (denoted  $w \equiv w'$ ) if  $w$  can be obtained from  $w'$  by a sequence of Knuth moves of the form  $(\kappa_1)$  and  $(\kappa_2)$ .

With these definitions, we have the following results, which suffice to complete the proof of Theorem 3.7.1:

- (1) if  $w$  is the reading word of a timed tableau of shape  $\lambda = (\lambda_1, \dots, \lambda_l)$ , then  $a_r(w) = \lambda_1 + \dots + \lambda_r$  for  $r = 1, \dots, l$ .
- (2) for every  $w \in A_n^+$ ,  $w \equiv P(w)$ ,
- (3) if  $w \equiv w'$ , then  $a_r(w) = a_r(w')$  for all  $r$ .

**3.9. Characterization of Knuth Equivalence.** Finally, it turns out that Knuth equivalence for timed words is characterized by Greene invariants, just as in the classical setting (Section 2.11):

THEOREM 3.9.1. *Given timed words  $w, w' \in A_n^+$ ,  $w \equiv w'$  if and only if, for all  $u, v \in A_n^+$ ,*

$$a_r(uwv) = a_r(uw'v) \text{ for all } r > 0.$$

### References

- [AD94] Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126, 1994.
- [Rob38] Gilbert de Beauregard Robinson. On the representations of the symmetric groups. *American Journal of Mathematics*, 60:745–760, 1938.
- [Ful97] William Fulton. *Young Tableaux: With Applications to Representation Theory and Geometry*. Cambridge University Press, 1997.
- [Gre74] Curtis Greene. An extension of Schensted’s theorem. *Advances in Mathematics*, 14:254–265, 1974.
- [Kir11] Anatol N. Kirillov. Introduction to tropical combinatorics. In *Physics and Combinatorics*, pages 82–150. World Scientific, 2011.
- [Knu70] Donald E. Knuth. Permutations, matrices, and generalized Young tableaux. *Pacific Journal of Mathematics*, 34:709–727, 1970.
- [LLT02] M. Lothaire. *Algebraic Combinatorics on Words*. Cambridge University Press, 2002. Chapter 5: The Plactic Monoid, by Alain Lascoux, Bernard Leclerc and Yves Thibon; also available from <http://www-igm.univ-mlv.fr/~berstel/Lothaire/AlgCWContents.html>.
- [LS78] Alain Lascoux and Marcel-Paul Schützenberger. Le monoïde plaxique. In *Noncommutative structures in algebra and geometric combinatorics (Naples)*. Consiglio Nazionale delle Ricerche, Roma, 1978.
- [Man98] Laurent Manivel. *Symmetric Functions, Schubert Polynomials and Degeneracy Loci*. AMS/SMF, 1998.
- [Pra15] Amritanshu Prasad. *Representaion Theory: A Combinatorial Viewpoint*. Cambridge University Press, 2015.
- [Pra18a] Amritanshu Prasad. An introduction to Schur polynomials. <https://arxiv.org/abs/1802.06073>, 2018. Notes from a course at the NCM school on Schubert Varieties, held at The Institute of Mathematical Sciences, Chennai, in November 2017.
- [Pra18b] Amritanshu Prasad. A timed version the plactic monoid. in preparation, 2018.
- [RS59] Michael O. Rabin and Dana Scott. Finite automata and their decision problems. *IBM journal of research and development*, 3(2):114–125, 1959.
- [Sch61] Craige Schensted. Longest increasing and decreasing subsequences. *Order*, 7:179–191, 1961.
- [You00] Alfred Young. On quantitative substitutional analysis. *Proceedings of the London Mathematical Society*, s1-33(1):97–145, 1900.

THE INSTITUTE OF MATHEMATICAL SCIENCES, CHENNAI

HOMI BHABHA NATIONAL INSTITUTE, MUMBAI