数理解析研究所講究録 1429 巻 2005 年 1-11

Box-ball systems and Robinson-Schensted-Knuth correspondence

神戸大学大学院・自然科学研究科　福田 香保理 (Kaori Fukuda)
Graduate School of Science and Technology,
Kobe University

Abstract

We study a box-ball system from the viewpoint of combinatorics of words and tableaux. Each state of the box-ball system can be transformed into a pair of tableaux (P, Q) by the Robinson-Schensted-Knuth correspondence. In the language of tableaux, the P-symbol gives rise to a conserved quantity of the box-ball system, and the Q-symbol evolves independently of the P-symbol. The time evolution of the Q-symbol is described explicitly in terms of the box-labels. This report gives a summary of our paper [1], and Japanese version has been already published [2].

1 Preliminaries

In this section, we recall some fundamental facts on combinatorics of words and tableaux, which we will freely use throughout this report.

A *Young diagram* is a finite collection of boxes, arranged in left-justified rows, with a weakly decreasing number of boxes in each row. We usually identify a partition, say $\lambda = (\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_l \geq 0)$, with the corresponding diagram. A *Young tableau*, or simply *tableau*, is a way of putting an integer in each box of a Young diagram that is weakly increasing across each row and strictly increasing down each column. We say that λ is the *shape* of the tableau. A *standard tableau* is a tableau in which the entries are numbers from 1 to n, each occurring once.

Example 1. We show some examples below.

- Young diagram $\begin{array}{|c|c|c|c|}\hline & & & \\ \hline & & \\ \cline{1-3} \\ \cline{1-1} \end{array} \quad \longleftrightarrow \quad \lambda = (4, 3, 1)$

- (Young) tableau $\begin{array}{|c|c|c|c|}\hline 1 & 1 & 4 & 5 \\ \hline 3 & 5 & 6 \\ \cline{1-3} 4 \\ \cline{1-1} \end{array} \quad \left\{ \begin{matrix} & \leq \\ \wedge & \end{matrix} \right\}$

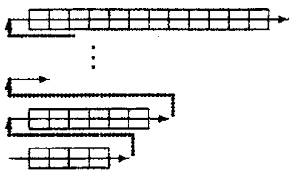
- Standard tableau $\begin{array}{|c|c|c|c|}\hline 1 & 2 & 4 & 8 \\ \hline 3 & 5 & 6 \\ \cline{1-3} 7 \\ \cline{1-1} \end{array} \quad \left\{ \begin{matrix} & < \\ \wedge & \end{matrix} \right\}$

Given a tableau T, we define the *word* $\mathrm{W}(T)$ *of* T by reading the entries of T *from left to right and bottom to top*. We say that a word w is a *tableau word* if it is the word of a tableau.

There is an algorithm called as *bumping* (*row-bumping*, or *row-insertion*), for constructing a new tableau from a tableau by inserting an integer. If there are no integers larger than i in the first row, add a new empty box at the right end, and put i in it. Otherwise, among the integers larger than i, find the leftmost one, say j, and put i in the box by bumping j out (i.e., replace j with i). Then, insert j, the bumped number, into the second row in the same way. Repeat this procedure until the bumped number can be put in a new box at the right end of the row.

This bumping procedure is decomposed into a sequence of rearrangements of three numbers in two ways, and these two transformations are called *elementary Knuth transformations*. We call two words *Knuth equevalent* if they can be transformed into each other by a sequence of elementary Knuth transformations.

Example 2. We show some examples below.

- Reading route of a tableau word $\mathrm{W}(T)$ 
- Bumping(row-insertion)

$$(u\ x'\ v)\ x \longrightarrow x'\ (u\ x\ v) \qquad (u \le x < x' \le v)$$

- The elementary Knuth transformation

$$yzx \longrightarrow yxz \quad (x < y \le z) \qquad xzy \longrightarrow zxy \quad (x \le y < z)$$

- Knuth equivalent (symbol: $\approx$)

$$5152431245 \quad \approx \quad 5415213245$$

We say that a two-rowed array is a *biword* if the columns $\binom{i_k}{j_k}$ are arranged to the lexicographic order. Then we define the *dual biword* $\boldsymbol{w}^*$ of $\boldsymbol{w}$, first by interchanging the top and the bottom rows, and by rearranging the columns so that $\boldsymbol{w}^*$ should be in lexicographic order.

There is a bijection between the biwords $\boldsymbol{w}$ and the pairs of tableaux (P, Q) of the same shape (RSK correspondence). The *P-symbol* P is the tableau obtained from the bottom row $(j_1, j_2, \dots, j_n)$ by bumping. The *Q-symbol* Q is another tableau of the same shape which keeps the itinerary of the bumping procedure; it is obtained by filling the number i_k at each step in the box that has newly appeared when the number j_k is inserted.

Example 3. We show some examples below.

- Biword $\quad \boldsymbol{w} = \begin{pmatrix} i_1 & i_2 & \cdots & i_k & \cdots & i_n \\ j_1 & j_2 & \cdots & j_k & \cdots & j_n \end{pmatrix}$

$$\text{The lexicographic order} \begin{cases} i_1 \le i_2 \le \cdots \le i_n \\ j_k \le j_{k+1} \quad \text{if} \quad i_k = i_{k+1} \quad (k = 1, \dots, n-1) \end{cases}$$

- The dual biword of the biword $\boldsymbol{w} = \left(\begin{smallmatrix} 1 & 2 & 2 & 4 & 5 & 7 \\ 3 & 1 & 5 & 2 & 2 & 1 \end{smallmatrix}\right)$ is $\boldsymbol{w}^* = \left(\begin{smallmatrix} 1 & 1 & 2 & 2 & 3 & 5 \\ 2 & 7 & 4 & 5 & 1 & 2 \end{smallmatrix}\right)$.
- RSK correspondence $\quad \{\text{Biword}\} \overset{1:1}{\longleftrightarrow} \{(P, Q)\}$
- Bumping Procedure $\quad \boldsymbol{w} = \begin{pmatrix} 1 & 2 & 2 & 4 & 5 & 7 \\ 3 & 1 & 5 & 2 & 2 & 1 \end{pmatrix} \quad \rightarrow \quad (P, Q)$

$$
\begin{array}{rlcrl}
P_0 = & \emptyset & & Q_0 = & \emptyset \\
& \downarrow & & & \downarrow \\
P_1 = & \boxed{3} & & Q_1 = & \boxed{1} \\
& \downarrow & & & \downarrow \\
P_2 = & \begin{array}{l} \boxed{1} \\ \boxed{3} \end{array} & & Q_2 = & \begin{array}{l} \boxed{1} \\ \boxed{2} \end{array} \\
& \downarrow & & & \downarrow \\
P_3 = & \begin{array}{l} \boxed{1}\boxed{5} \\ \boxed{3} \end{array} & & Q_3 = & \begin{array}{l} \boxed{1}\boxed{2} \\ \boxed{2} \end{array} \\
& \downarrow & & & \downarrow \\
P_4 = & \begin{array}{l} \boxed{1}\boxed{2} \\ \boxed{3}\boxed{5} \end{array} & & Q_4 = & \begin{array}{l} \boxed{1}\boxed{2} \\ \boxed{2}\boxed{4} \end{array} \\
& \downarrow & & & \downarrow \\
P_5 = & \begin{array}{l} \boxed{1}\boxed{2}\boxed{2} \\ \boxed{3}\boxed{5} \end{array} & & Q_5 = & \begin{array}{l} \boxed{1}\boxed{2}\boxed{5} \\ \boxed{2}\boxed{4} \end{array} \\
& \downarrow & & & \downarrow \\
P = P_6 = & \begin{array}{l} \boxed{1}\boxed{1}\boxed{2} \\ \boxed{2}\boxed{5} \\ \boxed{3} \end{array} & & Q = Q_6 = & \begin{array}{l} \boxed{1}\boxed{2}\boxed{5} \\ \boxed{2}\boxed{4} \\ \boxed{7} \end{array}
\end{array}
$$

2 Box-ball system

In this section, we consider a standard version of the BBS, and formulate our main results in terms of the standard BBS. A BBS is a system of finite number of balls of n colors evolving in the infinite array of boxes indexed by $\mathbf{Z}$. By a "standard" BBS, we mean a BBS in which n balls of n different colors are placed in the infinite array of boxes and all the boxes have capacity one. We use the numbers $1, 2, \dots, n$ to denote the colors of balls, and the symbol $e = n + 1$ to indicate a vacant place.

We first formulate the *standard* version of the BBS. A *state* of this system is a way to arrange n balls of different colors $1, 2, \dots, n$ in the array of boxes indexed by $\mathbf{Z}$, under the condition that at most one ball can be placed in each box. One step of time evolution of the standard BBS, from time t to $t + 1$, is defined as follows:

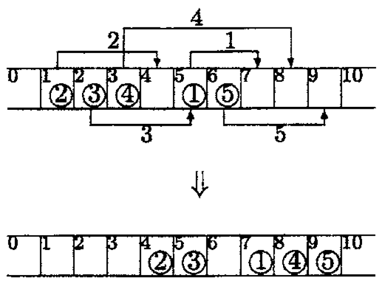
1. Every ball should be moved only once within the interval between time t and $t + 1$.

2. Move the ball of color 1 to the nearest right empty box.

3. In the same way, move the balls of colors $2, 3, \ldots, n$, in this order.

We refer to this rule as the *original algorithm* of the standard BBS.

Example 4. The following figure shows an example with $n = 5$.



Example 5. In the following figure, we show an example of time evolution with $n = 9$.

```
_____15789_______236_____4__________________________________
__________15789_____236___4_________________________________
_______________15789___236_4________________________________
____________________15789_2634______________________________
_________________________157682349__________________________
____________________________7_568_12349_____________________
_____________________________7___568___12349________________
______________________________7_____568_____12349___________
```

We next attach a biword to each state of the standard BBS and formulate our main theorem.

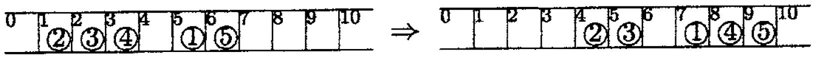
Each state of the standard BBS can be represented by a doubly infinite sequence $\cdots a_{-1}a_0a_1\cdots$ of numbers $1, \ldots, n$ and $e = n + 1$ such that $a_i = e$ except for a finite number of i's; if the box i is not empty, we define a_i to be the color of the ball contained in the box i, and set $a_i = e$ otherwise. Then we make a record of all pairs $\binom{i}{a_i}$ of *box-labels* i and *ball-colors* a_i (such that $a_i \neq e$), by scanning the sequence from left to right. In this way, we obtain a bijection between the possible states of the standard BBS and the biwords.

We remark that the bottom row of the dual biword represents the sequence of the box-labels of all nonempty boxes, arranged according to the ordering of colors. We refer to $b = (b_1, \ldots, b_n)$ as the *box-label sequence* associated with the state $\cdots a_{-1}a_0a_1\cdots$. Given a state $\cdots a_{-1}a_0a_1\cdots$ of the standard BBS, we denote by (P, Q) the pair of tableaux assigned to the biword through the RSK correspondence. Note that P is a standard tableau of n boxes, and that Q is a tableau of the same shape in which the entries are mutually distinct integers. The time evolution of the standard BBS is then translated into the time evolution of the corresponding biword, and also, via the RSK correspondence, into the time evolution of the pair of tableaux (P, Q) of the same shape.

Theorem 2.1. *We regard the standard BBS as the time evolution of the pairs of tableaux (P, Q) through the RSK correspondence in the way explained above. Then,*

1. *The P-symbol is a conserved quantity under the time evolution of the BBS.*
2. *The Q-symbol evolves independently of the P-symbol.*

Example 6. We illustrate below the main statement of this theorem with Example 4.



$$w = \begin{pmatrix} 1 & 2 & 3 & 5 & 6 \\ 2 & 3 & 4 & 1 & 5 \end{pmatrix} \quad \Rightarrow \quad w' = \begin{pmatrix} 4 & 5 & 7 & 8 & 9 \\ 2 & 3 & 1 & 4 & 5 \end{pmatrix}.$$

We consider the same evolution in terms of the pairs of tableaux.

$$P = \begin{array}{|c|c|c|c|}\hline 1 & 3 & 4 & 5 \\ \hline 2 \\ \cline{1-1}\end{array}, \quad Q = \begin{array}{|c|c|c|c|}\hline 1 & 2 & 3 & 6 \\ \hline 5 \\ \cline{1-1}\end{array} \quad \Rightarrow \quad P' = \begin{array}{|c|c|c|c|}\hline 1 & 3 & 4 & 5 \\ \hline 2 \\ \cline{1-1}\end{array}, \quad Q' = \begin{array}{|c|c|c|c|}\hline 4 & 5 & 8 & 9 \\ \hline 7 \\ \cline{1-1}\end{array}$$

In the above, " $P = P'$ " is the first statement of the theorem, and the second is that we can check Q' by considering only Q without P.

As we will see below, the time evolution of the standard BBS can be described locally by the so-called *carrier algorithm.* Theorem 2.1 will be proved in Section 4 by applying the carrier algorithm. We remark that the time evolution of the Q-symbol can also be described by using the carrier algorithm.

The *carrier algorithm* is a way to transform a finite sequence $w = (w_1, w_2, \dots w_n)$ of numbers into another sequence $w' = (w'_1, w'_2, \dots, w'_n)$, by means of a weakly increasing sequence $C = (c_1, \dots, c_m)$, called the *carrier.* In this transformation, the carrier moves along the word w from left to right; while the carrier passes each number w_k, the carrier loads w_k and unloads w'_k:

$$\left\{\begin{array}{ccccccccccc} & & w_1 & & w_2 & & w_3 & \cdots & & w_n & \\ & & \vdots & & \vdots & & \vdots & \cdots & & \vdots & \\ C = C_0 & \longrightarrow & C_1 & \longrightarrow & C_2 & \longrightarrow & C_3 & \cdots & C_{n-1} & \longrightarrow & C_n = C' \\ & & \vdots & & \vdots & & \vdots & \cdots & & \vdots & \\ & & w'_1 & & w'_2 & & w'_3 & \cdots & & w'_n & \end{array}\right\}$$

The rule of loading and unloading is defined as follows:

The rule of loading/unloading : *Let $C_{k-1} = (c_1^{(k-1)}, c_2^{(k-1)}, \dots, c_m^{(k-1)})$ $(c_1^{(k-1)} \leq c_2^{(k-1)} \leq \cdots \leq c_m^{(k-1)})$ be the sequence of numbers which have already been loaded on the carrier. Let w_k be the number to be loaded. Compare w_k with the numbers in C_{k-1}. If there are some numbers larger than w_k in C_{k-1}, then one of the smallest among them is unloaded, and w_k is loaded instead. If there is no such number, a minimum in C_{k-1} is unloaded, and w_k is loaded instead.*

$$w'_k = \begin{cases} \min\{c_i^{(k-1)} \in C_{k-1} \mid c_i^{(k-1)} > w_k\} \\ \qquad\qquad\qquad \text{if } \{c_i^{(k-1)} \in C_{k-1} \mid c_i^{(k-1)} > w_k\} \neq \emptyset, \\ c_1^{(k-1)} \qquad\qquad \text{otherwise.} \end{cases}$$

$$\begin{aligned} C_k = {} & \text{the sequence of numbers obtained from } C_{k-1} \\ & \text{by replacing a } w'_k \text{ by } w_k. \end{aligned}$$

Given two finite sequences $C = (c_1, c_2, \dots, c_m)$ $(c_1 \le c_2 \le \cdots \le c_m)$ and $w = (w_1, w_2, \dots, w_n)$, from $C_0 = C$, we obtain the new sequences $C' = C_n$ and w' by repeating the rule of loading/unloading above. We call this transformation $(C, w) \to (C', w')$ the *carrier algorithm.* Note that the carrier algorithm can be understood as a repetition of Knuth transformations.

$$\begin{aligned} Cw &= C_0 w_1 w_2 w_3 \cdots w_n \approx w_1' C_1 w_2 w_3 \cdots w_n \\ &\approx w_1' w_2' C_2 w_3 \cdots w_n \\ &\approx \quad \vdots \\ &\approx w_1' w_2' w_3' \cdots w_n' C_n = w'C' \end{aligned}$$

In our paper [1], two propositions are given in order to prove the above-mentioned theorem. These both explains that time evolution of the BBS can be described by using the carrier algorithm. On one hand, the carrier runs along with the sequence $\dots, a_{-1}, a_0, a_1, \dots$ meaning the state of the BBS; on the other, the carrier runs along with the "box-label sequence" represented by the dual word. It is effective in seeing the essence of time evolution of the BBS that we consider the dual version.

3 Carrier algorithm

In the following, we give two propositions that will be used in the proof of Theorem 2.1. The time evolution of the standard BBS from one state to the next can be described in two different ways; the original algorithm and the transformation of the box-label sequences. We describe these two algorithms by using the carrier as introduced above.

Note that a state of the standard BBS, represented as an infinite sequence $\cdots a_{-1} a_0 a_1 \cdots$, is identified with a function $a : \mathbf{Z} \to \{1, \dots, n, n+1 = e\}$ of *finite support*; the support of a is defined by $\operatorname{supp}(a) = \{i \in \mathbf{Z}; a_i \ne e\}$. The time evolution a' of a is determined by the injective mapping $f : \operatorname{supp}(a) \to \mathbf{Z}$ such that $a'_{f(i)} = a_i$ for $i \in \operatorname{supp}(a)$, and by $a'_j = e$ for $j \notin \operatorname{Im}(f)$. We first describe how the mapping f is defined by the original algorithm.

We take an interval $[p, q]$ of $\mathbf{Z}$ so that it contains all i with $a_i \ne e$, and all i with $a'_i \ne e$ as well. A choice of such an interval is given by $p = \min\{i \in \mathbf{Z} \mid a_i \ne e\}$, $q = \max\{i \in \mathbf{Z} \mid a_i \ne e\} + n$. We remark that, in this procedure, the final state of the carrier is identical to the initial state: $C' = (e, \dots, e)$.

$$[p, q] : \begin{cases} p = \min\{i \in \mathbf{Z} \mid a_i \ne e\} \\ q = \max\{i \in \mathbf{Z} \mid a_i \ne e\} + n \end{cases}$$

Example 7.

			p									q		
Box-label :	$\cdots$	0	1	2	3	4	5	6	7	8	9	10	11	12	$\cdots$
	$\cdots$	e	**2**	**3**	**4**	e	**1**	**5**	e	e	e	e	e	e	$\cdots$
	$\cdots$	e	e	e	e	**2**	**3**	e	**1**	**4**	**5**	e	e	e	$\cdots$

Proposition 3.1. *For a given state of the standard BBS, by ignoring the infinite sequences of e's on both sides, let* $A = (a_p, a_{p+1}, \dots, a_{q-1}, a_q)$ *be the remaining sequence of numbers; with* p, q *defined as above. Then, the original algorithm* $A \to A'$ *from time* t *to* $t+1$*, can be described by the carrier algorithm with a sequence* $C = (e, e, \dots, e)$ *of* n *e's chosen as the initial state.*

Example 8. We show an example below.

$$\begin{array}{ccccccccccccccc} \cdots & e & \mathbf{2} & \mathbf{3} & \mathbf{4} & e & \mathbf{1} & \mathbf{5} & e & e & e & e & e & e & \cdots \\ \cdots & e & e & e & e & \mathbf{2} & \mathbf{3} & e & \mathbf{1} & \mathbf{4} & \mathbf{5} & e & e & e & \cdots \end{array}$$

$$e,e,e,e,e \underset{e}{\overset{\mathbf{2}}{+}} 2,e,e,e,e \underset{e}{\overset{\mathbf{3}}{+}} 2,3,e,e,e \underset{e}{\overset{\mathbf{4}}{+}} 2,3,4,e,e \underset{\mathbf{2}}{\overset{e}{+}} 3,4,e,e,e \underset{\mathbf{3}}{\overset{\mathbf{1}}{+}} 1,4,e,e,e \underset{e}{\overset{\mathbf{5}}{+}} \to$$

$$\to 1,4,5,e,e \underset{\mathbf{1}}{\overset{e}{+}} 4,5,e,e,e \underset{\mathbf{4}}{\overset{e}{+}} 5,e,e,e,e \underset{\mathbf{5}}{\overset{e}{+}} e,e,e,e,e \underset{e}{\overset{e}{+}} e,e,e,e,e \underset{e}{\overset{e}{+}} e,e,e,e,e$$

Next, we discuss the transformation of the box-label sequences. Recall that the box-label sequence $b = (b_1, \dots, b_k, \dots, b_n)$ is defined as the bottom row of the dual biword $\boldsymbol{w}^*$. Notice that $b_k \in [p, q]$ for all $k = 1, 2, \dots, n$, with p, q defined as before.

Example 9. We first recall the biword formulation of Example 6.

$$\begin{pmatrix} 1 & 2 & 3 & 5 & 6 \\ \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{1} & \mathbf{5} \end{pmatrix} \to \begin{pmatrix} 4 & 5 & 7 & 8 & 9 \\ \mathbf{2} & \mathbf{3} & \mathbf{1} & \mathbf{4} & \mathbf{5} \end{pmatrix}$$

We next transform into the dual biwords below.

$$\begin{pmatrix} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} \\ 5 & 1 & 2 & 3 & 6 \end{pmatrix} \to \begin{pmatrix} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} \\ 7 & 4 & 5 & 8 & 9 \end{pmatrix}$$

Here, the evolution of the box-label sequence is as follows: $b = (51236) \to b' = (74589)$.

Proposition 3.2. *For a given state of the standard BBS, the transformation of the box-label sequence $b \to b'$ from time t to $t+1$ can be described by the carrier algorithm with the initial state of the carrier $C = (l_1, l_2, \dots, l_m)$ defined as the increasing sequence consisting of the labels of all empty boxes in the interval $[p, q]$.*

Example 10. We show an example below.

$$b = (51236) \to b' = (74589)$$

$$4,7,8,9,10,11 \underset{7}{\overset{5}{+}} 4,5,8,9,10,11 \underset{4}{\overset{1}{+}} 1,5,8,9,10,11 \underset{5}{\overset{2}{+}} 1,2,8,9,10,11 \underset{8}{\overset{3}{+}} 1,2,3,9,10,11 \underset{9}{\overset{6}{+}} 1,2,3,6,10,11$$

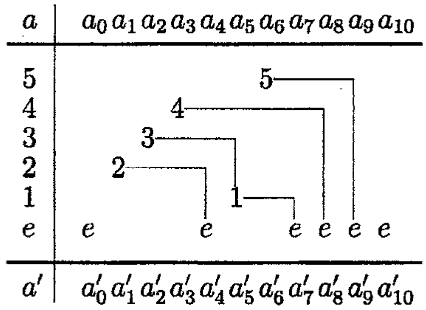
We use the symbol T^* in order to imply the evolution of the box-label sequence: $b' = T^*(b)$.

4 Proof of the main results (Summary)

In this section, we prove two propositions given in the foregoing section, and prove the main theorem by using these propositions.

By analyzing carefully the original algorithm of BBS, it turns out that this algorithm can be described in two ways by using carrier algorithm; thus, two propositions can be proved. Furthermore, the fact that carrier algorithm is a repetition of Knuth transformations shows that Knuth equivalence is maintained in the time evolution. The proof of the theorem is substantially completed on the basis of these ideas.

We now visualize the original algorithm of BBS by means of a 2-dimensional diagram. First, write the state a at time t at the top; write each a_i again, down in the same column at the row corresponding to the number itself; here we are using the datum of Example 4. Then, following the original algorithm of BBS, connect " 1 " to its partner, nearest e on the right. Then look at " 2 ", draw lines by the same method. In this example, " 3 " should be moved to the empty box which had originally been occupied by the 1 on the right. Considering this 1 as the partner of the 3, connect the 3 to it. Do the same thing until all $a_i(\neq e)$ have been connected to their partners $a'_{f(i)}$.



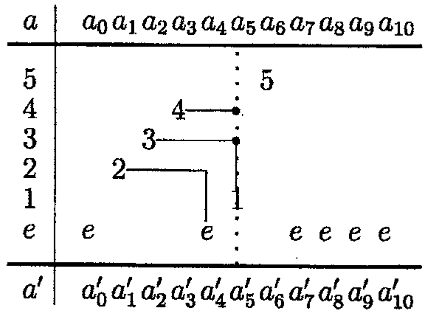
Then the general rule for drawing lines can be described as follows:

> Connect each number with the leftmost one among all the smaller numbers on the right that have not been connected from above.

Notice that the perfect chains never intersect with each other. In view of this fact, we see that the same set of non-intersecting perfect chains can be obtained by observing the sequence of numbers at time t *from left to right*, rather than *from bottom to top* as in the rule above. In the following, we consider how to make the same diagram from another viewpoint by using this law.

Sketch of proof of Proposition 3.1

First, we draw lines to connect balls from left to right, and we can prove the proposition 3.1.

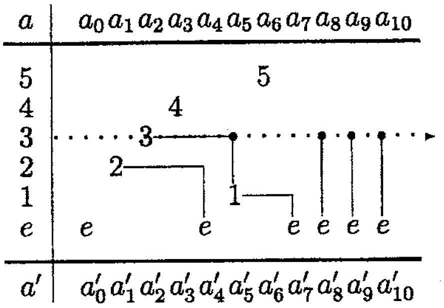


$\{3, 4, e, e, \dots\}$

Thus, it can be understood automatically that carrier algorithm is applied like a proposition 3.1 by considering that we load the candidate of the numbers connected with lines into the carrier.

Sketch of proof of Proposition 3.2

Next, we draw lines from the ball under a figure to the upper ball by paying attention to the numerical value, i.e., in the same order as the original algorithm. However, we see the box label i but not the value of a_i (color of balls).



$\{5, 8, 9, 10, \dots\}$

Thus, it can be understood more obediently than a previous proposition 3.1, that the carrier algorithm is applied like the proposition 3.2 by considering that we load the candidate of the empty box-labels tied with lines into the carrier[1].

Hereafter, we explain the proof of the theorem 2.1 briefly. Please refer to our paper [1] for details.

Proof of Theorem 2.1 (i)

We first give the following lemma.

Lemma 1. *If w and w' are Knuth equivalent words, and w_0 and w'_0 are the results of removing the p largest numbers from each, for any p, then w_0 and w'_0 are Knuth equivalent words.*

In the notation of Proposition 3.1, we get $CA \approx A'C$.

$$\begin{aligned} CA &= C_p(a_p, a_{p+1}, \dots, a_{q-1}, a_q) \\ &\approx a_p' C_{p+1}(a_{p+1}, a_{p+2}, \dots, a_{q-1}, a_q) \\ &\;\vdots \\ &\approx (a_p', a_{p+1}', \dots, a_{q-1}', a_q')C' \qquad = A'C \end{aligned}$$

We know that Knuth equivalent words correspond to the same tableau. Since e is thought of as larger than any other number, we see that the results A_e and A'_e of removing e's from CA and $A'C$, respectively, are Knuth equivalent, i.e., $A_e \approx A'_e$. Hence the bumping of A_e and A'_e give the same tableau P; this P-symbol is conserved by the time evolution. We remark that the sequence A_e is nothing but the bottom row of the biword $\boldsymbol{w}$ we introduced before. We have completed the proof of the first statement of Theorem 2.1.

Proof of Theorem 2.1 (ii)

We next give the following lemmas.

[1]In our paper [1], we first prove this proposition.

Lemma 2. *If a and b are two Knuth equivalent words, then so are the resulting $T^*(a)$ and $T^*(b)$.*

Lemma 3. *If b is a tableau word, $T^*(b)$ is a tableau word of the same shape.*

Here, we prove simply the second statement of the theorem 2.1 using these two lemmas. Please read the explanation after seeing the following figure.

$$\begin{tikzcd}
w_1 \arrow[r, "RSK"] \arrow[d, "dual"'] \arrow[ddr] & (P_1, Q_1) \arrow[d] \arrow[ddr] & \\
w_1^* \arrow[r] \arrow[dddr, "T"'] & (Q_1, F_1) \arrow[ddr] & \\
& w_2 \arrow[r] \arrow[d] & (P_2, Q_2) \arrow[d] \\
& w_2^* \arrow[r] & (Q_2, F_2)
\end{tikzcd}
\qquad
\begin{cases} P_2 = P_1 \\ Q_2 = T^*(Q_1) \end{cases}$$

We easily itemize the main points of the proof.

- $b \approx W(Q_1) \to T^*(b) \approx T^*(W(Q_1))$ $\quad$ ($\because$ Lemma2)
- $T^*(b) = b'$ and $b' \approx W(Q_2) \to W(Q_2) \approx T^*(W(Q_1))$
- $W(Q_1)$ is a tableau word. $\to T^*(W(Q_1))$ is a tableau word too. $\quad$ ($\because$ Lemma3)
- Therefore, $W(Q_2) = T^*(W(Q_1))$.

$\square$

Identifying tableau words with tableaux, we can define the time evolution of the Q-symbol Q by

$$T^*(Q) = \mathrm{Tab}(T^*(\mathrm{W}(Q))).$$

Summarizing, with the interval $[p, q] \subset \mathbf{Z}$ again, we have

Proposition 4.1. *In the standard BBS, the time evolution of the Q-symbol Q is described by the box-label algorithm with a carrier. The initial state of the carrier is given with $C = (l_1, \ldots, l_m)$ defined as the increasing sequence consisting of the labels of all empty boxes in the interval $[p, q]$. The carrier runs along the rows of the tableau Q from left to right, and bottom to top.*

Therefore, the evolution of the Q-symbol can be directly computed by the box-label algorithm at the level of tableau words read off from the tableau, without the need to recompute a tableau from the resulting word.

5 Generalization of the BBS

In our paper [1], generally we expanded into the conditions as follows: There exist m balls in all, we allow to use an arbitrary finite number of balls for each color, the capacity of each box is specified individually, and many balls may go into the same box from one piece: And then we gave the theorem of the same contents as the case of the standard BBS. This theorem can be proved in the same procedure as the case of the standard BBS[2].

Acknowledgements

The author would like to thank Prof. M. Okado and Prof. A. Kuniba for their requesting me to speak at this conference CAIS 2004. She terribly regretted that Prof. Kuniba had to be absent from this conference. What supported her to participate in CAIS was his helpful encouragement. Anyway, we need to send grand applause to organizers Mr. Okado and Mr. Kuniba once again.

References

[1] ________, Box-Ball Systems and Robinson-Shensted-Knuth Correspondence, Journal of Algebraic Combinatorics, 19, 67–89, (2004).

[2] ________, Combinatorial aspects of box-ball systems (in Japanese), RIMS Kokyuroku 1310, 16–28, (2003).

[2]Please refer to our paper [1] with examples.