Hang Lung Mathematics Awards
Vol. 1 (2004), pp. 35–60

GHOST LEG

TEAM MEMBERS
Man-Kit Ho, Hoi-Kwan Lau, Ting-Fai Man, Shek Yeung[1]

SCHOOL
Sha Tin Government Secondary School

Abstract. People sometimes need to distribute or select things randomly. Some do this by drawing lots, some by throwing dice, and others by drawing "ghost leg". Our group finds that the input and output (elements to be arranged) can be considered as a sequence while the "ghost leg" itself as a permutation. So each horizontal line leads to a transformation of neighboring elements. Also these lines can be transformed and deleted under some conditions without affecting the input and output sequences. And thus a simplified one is obtained.
Our objective in this project is to systemize and solve "ghost leg" mathematically so that result can be obtained quickly and systematically.

1. Introduction

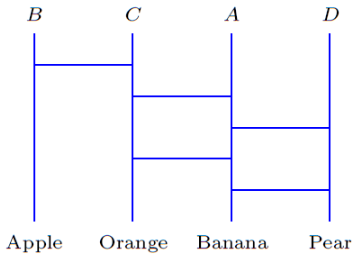
"Drawing ghost leg" is a traditional Chinese game. It is often used to arrange or select things. It consists of some horizontal lines and vertical lines. Very often the number of vertical lines is the same as the number of people playing, and at the bottom lines there are certain items, e.g. things that will be given to the player. Unlike vertical lines, the number of horizontal lines can be zero or more. The horizontal lines can be drawn anywhere between two vertical lines, except that no horizontal lines crossing vertical ones. The general rule for playing this game is: first choose a line on the top, and follow this line downwards. If a horizontal line is encountered, follow the horizontal line to get to another vertical line and go downwards again. Repeat the above procedures until reaching the end of the vertical line. Then the player will be given the thing written at the bottom of the line.

[1]This work is done under the supervision of the authors' teacher, Ms. Fei Wong.

Example 1: Some fruits are distributed to A, B, C, D, and they decide it by drawing ghost leg.

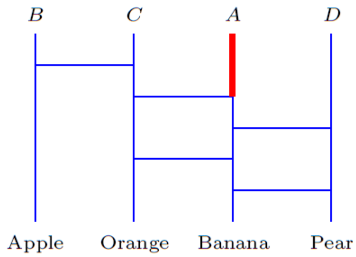
(1)

A, B, C, D select their starting positions.



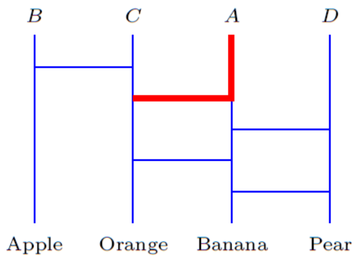
(2)

A draws first.



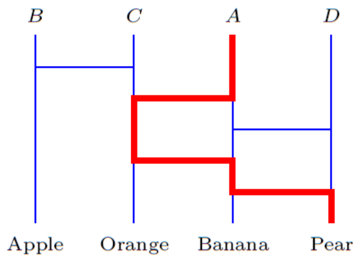
(3)

Turn left after encountering a horizontal line.

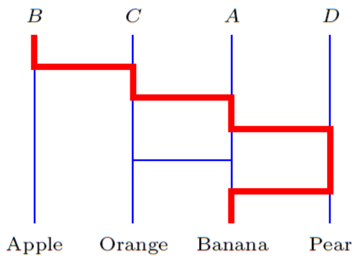


(4)

Go downwards and continue.

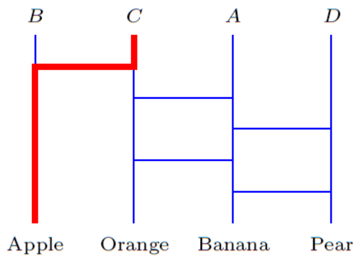


(5)



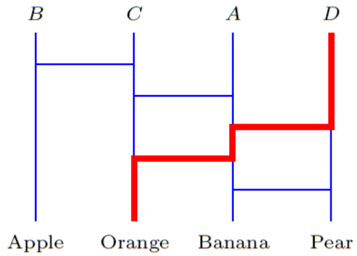
B gets the banana.

(6)



C gets the apple.

(7)

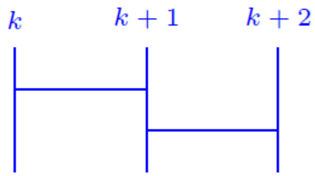


D gets the orange.

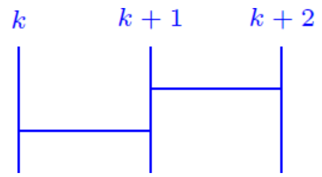
2. Definition

To make things simple, some terms will be used and their definition will be given.

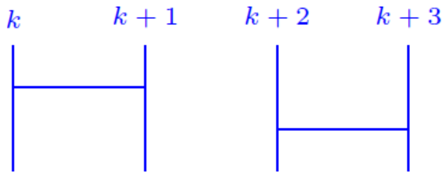
1. A ghost leg is a diagram formed by some vertical and horizontal lines. The elements written above the lines can be treated as a sequence, and after the ghost leg, these elements are of different order, and the sequence has been transformed to another permutation.
2. A track is a vertical line in a ghost leg. The index of a track is the number of which track it is counting from left to right, e.g. the leftmost vertical line is called T_1.
3. A leg is a horizontal line in a ghost leg. A leg has two index, if the leg is between T_k and T_{k+1}, the first index is k, and if it is the Rth line counted from top, the second index is R. $L_{k,R}$ is the leg in this example.
4. After the ghost leg, all the input I_k, where k is any natural number, may be transformed to another track, so the elements below the ghost leg maybe different from the elements on top. As a result, the output sequence may different from the input sequence.
5. A level is the space occupied by a leg. Each level has one and only one leg. When playing the ghost leg, the legs which are encountered first are defined to be at higher level. So when we see the following configuration, $L_k > L_{k+1}$.



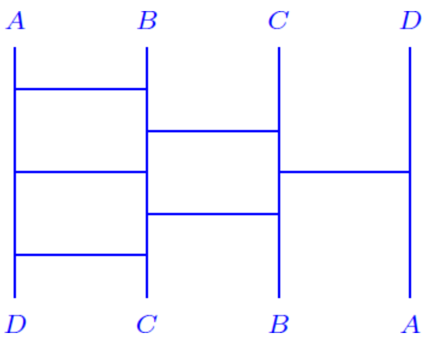
And if we see the following configuration, $L_{k+1} > L_k$.



There is a special case of configuration. In the following ghost leg, $L_{k+2} > L_k$.



Take the following ghost leg as an example:

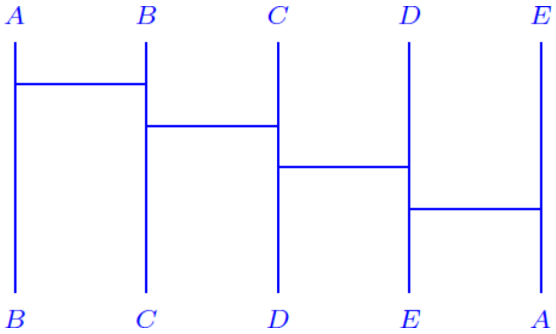


$L_{11} > L_{21} > L_{31} > L_{12} > L_{22} > L_{13}$

6. A specific permutation can be represented by infinitely many ghost legs (to be discussed later). The ghost leg with the smallest number of legs is called "prime". For some permutation, there maybe more than one prime. The only difference between those primes is that they are of different shapes, and they must of same number of legs.

3. Methods of representation

1. Graphical: a ghost leg can be represented by a graph



2. Represent by a sequence: the above graph can be represented by $(abcde) \to (bcdae)$.

3. Represent by a matrix: the above graph can be represented by $\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

   and
$$\begin{pmatrix} a & b & c & d & e \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} b & c & d & e & a \end{pmatrix}$$

4. Simplified matrix: actually each leg can be represented by a matrix, and the product of all the matrix gives the representation in 3. Thus, $M = L_1L_2L_3L_4$ can be used to represent the graph in 1.

After passing a leg, some of the input In will change its position. We use (I_x, T_y) to represent the new position of the input. For example, in the graph in 1, after the first leg, I_1 has changed its position. We write $(I_1, T_1) \to (I_1, T_2)$.

4. Properties of Ghost Leg

The permutation of ghost leg has many different properties, we will use a ghost leg representative matrix of n by n, M, to represent and prove the properties of ghost leg.

4.1. Idempotent

A ghost leg is idempotent $\Leftrightarrow$ The ghost leg representative matrix, M, has the property of $M^k = I$, $\exists k \in \mathbb{N}$, where I is the identity matrix.

All ghost leg representative matrix is idempotent.

Proof. (**Prove by contradiction**)
Given X_i a $(1\times n)$ matrix, $\forall i \in \mathbb{N}$, where M is a $(n\times n)$ matrix. $X_{i+1} = X_i M$, $\forall i \in \mathbb{N}$. And the input of ghost leg be X_1. Therefore, X_i represents the new sequence due to permutation of one leg.

Let $M^k \neq I$, $\forall k \leqslant n!$. Since $X_{i+1} = X_i M$, $\forall i \in \mathbb{N}$, it follows that $X_1 \neq X_2 \neq X_3 \neq \cdots \neq X_k$, $\forall k \leqslant n!$. Since n numbmers has at most $n!$ permutations, $\exists a, b \leqslant (n!+1)$ and $a < b$ such that $X_a = X_b$, which leads to a contradiction. Therefore, $\exists j = (b-a) \leqslant n!$ and a, $b \leqslant (n!+1)$ and $a < b$ such that $M^j = I$. Hence all ghost legs are idempotent. □

4.2. Periodicity

One ghost leg is periodic $\Leftrightarrow$ $\exists k \in \mathbb{N}$ such that ghost leg matrix, $M = M^k$.

All ghost legs are periodic.

Proof. Since all ghost legs are idempotent, $\exists x \in \mathbb{N}$ such that $M^x = I$. Then $M^{x+1} = M$, so all ghost legs are periodic. □

4.3. Reversibility

One ghost leg is reversible $\Leftrightarrow$ there exist another ghost leg whose permutation can cancel out that of the given one, i.e. the combination of the two ghost legs does not permutate the inputs.

All ghost legs are are reversible.

Proof. For all ghost leg with n legs, its matrix representation can be expressed as simplified matrix: i.e. $M = L_{x_1} L_{x_2} \ldots L_{x_n}$.

Consider one matrix M', $M' = L_{x_n} L_{x_{n-1}} \ldots L_{x_1}$.

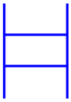
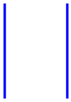
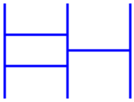
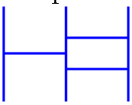
$$MM' = L_{x_1} L_{x_2} \ldots L_{x_{n-1}} L_{x_n} L_{x_n} L_{x_{n-1}} \ldots L_{x_2} L_{x_1}.$$

As $L_k L_k = I$, $\forall k \in \mathbb{N}$ (prove later)

$$\begin{aligned} MM' &= L_{x_1} L_{x_2} \dots L_{x_{n-1}} L_{x_{n-1}} \dots L_{x_2} L_{x_1} \\ &= \dots \\ &= L_{x_1} L_{x_1} \\ &= I. \end{aligned}$$

Therefore, all ghost legs are reversible. □

4.4. $L_k L_k = I$ and $L_k L_{k+1} L_k = L_{k+1} L_k L_{k+1}$

Graphically,  is equivalent to  and  is equivalent to . Before proving the above identity, we prove the two identities below. (note: matrix below are all of n by n order, and AB 3×3 matrix)

$$\begin{pmatrix} 100........0 \\ 010.......... \\ .....\ddots \\ .00.....A \end{pmatrix} \begin{pmatrix} 100........0 \\ 010.......... \\ .....\ddots \\ .00.....B \end{pmatrix} = \begin{pmatrix} 100........0 \\ 010.......... \\ .....\ddots \\ .00.....AB \end{pmatrix} \tag{1}$$

$$\begin{pmatrix} A............ \\ \dots\ddots \\ ....10 \\ .00.....001 \end{pmatrix} \begin{pmatrix} B............ \\ \dots\ddots \\ ....10 \\ .00.....001 \end{pmatrix} = \begin{pmatrix} AB............ \\ \dots\ddots \\ ....10 \\ .00.....001 \end{pmatrix} \tag{2}$$

Proof (1). Apparently,

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & A & \\ 0 & & & \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & B & \\ 0 & & & \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & AB & \\ 0 & & & \end{pmatrix}.$$

Let $A' = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & A \\ 0 & & \end{pmatrix}$, $B' = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & B \\ 0 & & \end{pmatrix}$.

$$\begin{pmatrix} 1 & \dots & 0 \\ \vdots & & A' \\ 0 & & \end{pmatrix}\begin{pmatrix} 1 & \dots & 0 \\ \vdots & & B' \\ 0 & & \end{pmatrix}$$

$$= \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & A \\ 0 & & \end{pmatrix} & \\ 0 & & \end{pmatrix}\begin{pmatrix} 1 & \dots & 0 \\ \vdots & \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & B \\ 0 & & \end{pmatrix} & \\ 0 & & \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & \vdots \\ \dots & \dots & & \\ 0 & \dots & & AB \end{pmatrix}$$

Similarly, let $A'' = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & A' \\ 0 & & \end{pmatrix}$, $B'' = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & B' \\ 0 & & \end{pmatrix}$.

$$\begin{pmatrix} 1 & \dots & 0 \\ \vdots & & A'' \\ 0 & & \end{pmatrix}\begin{pmatrix} 1 & \dots & 0 \\ \vdots & & B'' \\ 0 & & \end{pmatrix}$$

$$= \begin{pmatrix} 1 & \dots & 0 \\ \vdots & \begin{pmatrix} 1 & \dots & 0 \\ \vdots & 1 & \vdots \\ 0 & \dots & A \end{pmatrix} & \\ 0 & & \end{pmatrix}\begin{pmatrix} 1 & \dots & 0 \\ \vdots & \begin{pmatrix} 1 & \dots & 0 \\ \vdots & 1 & \vdots \\ 0 & \dots & B \end{pmatrix} & \\ 0 & & \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & \vdots \\ \dots & \dots & 1 & \vdots \\ 0 & \dots & \dots & AB \end{pmatrix}$$

Inductively,

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \vdots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & A \end{pmatrix}\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \vdots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & B \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \vdots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & AB \end{pmatrix} \qquad \square$$

Proof (2). Apparently,

$$\begin{pmatrix} & A & & 0 \\ & & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} & B & & 0 \\ & & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} & AB & & 0 \\ & & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Let $A' = \begin{pmatrix} & A & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix}$, $B' = \begin{pmatrix} & B & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix}$.

$$\begin{aligned} & \begin{pmatrix} & A' & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix}\begin{pmatrix} & B' & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix} \\ = & \begin{pmatrix} \begin{pmatrix} & A & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix} & \begin{matrix} 0 \\ \vdots \end{matrix} \\ 0 \quad \dots & 1 \end{pmatrix}\begin{pmatrix} \begin{pmatrix} & B & 0 \\ & & \vdots \\ 0 & \dots & 1 \end{pmatrix} & \begin{matrix} 0 \\ \vdots \end{matrix} \\ 0 \quad \dots & 1 \end{pmatrix} \\ = & \begin{pmatrix} AB & \vdots & 0 \\ \dots & 1 & \vdots \\ & & 0 \\ 0 \quad \dots & 0 & 1 \end{pmatrix} \end{aligned}$$

Use method in proof (1), inductively

$$\begin{aligned} & \begin{pmatrix} A & & & \vdots & 0 \\ & & & \vdots & \vdots \\ & & \ddots & & \vdots \\ \dots & \dots & & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}\begin{pmatrix} B & & & \vdots & 0 \\ & & & \vdots & \vdots \\ & & \ddots & & \vdots \\ \dots & \dots & & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix} \\ = & \begin{pmatrix} AB & & & \vdots & 0 \\ & & & \vdots & \vdots \\ & & \ddots & & \vdots \\ \dots & \dots & & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}. \end{aligned}$$

Furthermore, let

$$A' = \begin{pmatrix} A & & & \vdots & 0 \\ & & & \vdots & \vdots \\ & & \ddots & & \vdots \\ \dots & \dots & & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}, \; B' = \begin{pmatrix} B & & & \vdots & 0 \\ & & & \vdots & \vdots \\ & & \ddots & & \vdots \\ \dots & \dots & & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{pmatrix}.$$

$$\begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & A & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & B & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & A' \end{pmatrix} \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & B' \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \ddots & \vdots \\ 0 & 0 & \dots & A'B' \end{pmatrix} \quad \text{(from (1))}$$

$$= \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & AB & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \quad \text{(from (2))} \qquad \square$$

Proof of $L_kL_k = I$. Let $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.

$$L_kL_k = \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix} & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix} & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix}$$

$$= I$$

□

Proof of $L_kL_{k+1}L_k = L_{k+1}L_kL_{k+1}$. Let

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \ B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

$$L_kL_{k+1}L_k$$

$$= \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & \begin{matrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{matrix} & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 & 0 \\ 0 & 1 & \dots & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \vdots & \vdots & & \begin{matrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{matrix} & & \vdots & \vdots \\ \vdots & \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & \dots & \dots & \dots & 0 & 1 \end{pmatrix}$$

$$\times L_k$$

$$= \begin{pmatrix} 1 & 0 & \dots & & \dots & & \dots & 0 & 0 \\ 0 & 1 & \dots & & \dots & & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & & & \vdots & \vdots \\ & & & 0 & 0 & 1 & & & \\ \vdots & \vdots & & 1 & 0 & 0 & & \vdots & \vdots \\ & & & 0 & 1 & 0 & & & \\ \vdots & \vdots & & & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & & \dots & & \dots & 1 & 0 \\ 0 & 0 & \dots & & \dots & & \dots & 0 & 1 \end{pmatrix} L_k$$

$$= \begin{pmatrix} 1 & 0 & \dots & & \dots & & \dots & 0 & 0 \\ 0 & 1 & \dots & & \dots & & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & & & \vdots & \vdots \\ & & & 0 & 0 & 1 & & & \\ \vdots & \vdots & & 1 & 0 & 0 & & \vdots & \vdots \\ & & & 0 & 1 & 0 & & & \\ \vdots & \vdots & & & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & & \dots & & \dots & 1 & 0 \\ 0 & 0 & \dots & & \dots & & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & \dots & & \dots & & \dots & 0 & 0 \\ 0 & 1 & \dots & & \dots & & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & & & \vdots & \vdots \\ & & & 0 & 1 & 0 & & & \\ \vdots & \vdots & & 1 & 0 & 0 & & \vdots & \vdots \\ & & & 0 & 0 & 1 & & & \\ \vdots & \vdots & & & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & & \dots & & \dots & 1 & 0 \\ 0 & 0 & \dots & & \dots & & \dots & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \dots & & \dots & & \dots & 0 & 0 \\ 0 & 1 & \dots & & \dots & & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & & & & \vdots & \vdots \\ & & & 0 & 0 & 1 & & & \\ \vdots & \vdots & & 0 & 1 & 0 & & \vdots & \vdots \\ & & & 1 & 0 & 0 & & & \\ \vdots & \vdots & & & & & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & & \dots & & \dots & 1 & 0 \\ 0 & 0 & \dots & & \dots & & \dots & 0 & 1 \end{pmatrix}$$

$$
\begin{aligned}
&L_{k+1}L_kL_{k+1}\\
&=\begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0\\
0 & 1 & \dots & & \dots & & \dots & 0 & 0\\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots\\
 & & & 1 & 0 & 0 & & & \\
\vdots & \vdots & & 0 & 0 & 1 & & \vdots & \vdots\\
 & & & 0 & 1 & 0 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots\\
0 & 0 & \dots & & \dots & & \dots & 1 & 0\\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0\\
0 & 1 & \dots & & \dots & & \dots & 0 & 0\\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots\\
 & & & 0 & 1 & 0 & & & \\
\vdots & \vdots & & 1 & 0 & 0 & & \vdots & \vdots\\
 & & & 0 & 0 & 1 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots\\
0 & 0 & \dots & & \dots & & \dots & 1 & 0\\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix}\\
&\quad\times L_{k+1}\\
&=\begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0\\
0 & 1 & \dots & & \dots & & \dots & 0 & 0\\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots\\
 & & & 0 & 1 & 0 & & & \\
\vdots & \vdots & & 0 & 0 & 1 & & \vdots & \vdots\\
 & & & 1 & 0 & 0 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots\\
0 & 0 & \dots & & \dots & & \dots & 1 & 0\\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix} L_{k+1}\\
&=\begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0\\
0 & 1 & \dots & & \dots & & \dots & 0 & 0\\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots\\
 & & & 0 & 1 & 0 & & & \\
\vdots & \vdots & & 0 & 0 & 1 & & \vdots & \vdots\\
 & & & 1 & 0 & 0 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots\\
0 & 0 & \dots & & \dots & & \dots & 1 & 0\\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix}
\begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0\\
0 & 1 & \dots & & \dots & & \dots & 0 & 0\\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots\\
 & & & 1 & 0 & 0 & & & \\
\vdots & \vdots & & 0 & 0 & 1 & & \vdots & \vdots\\
 & & & 0 & 1 & 0 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots\\
0 & 0 & \dots & & \dots & & \dots & 1 & 0\\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix}
\end{aligned}
$$

$$
= \begin{pmatrix}
1 & 0 & \dots & & \dots & & \dots & 0 & 0 \\
0 & 1 & \dots & & \dots & & \dots & 0 & 0 \\
\vdots & \vdots & \ddots & & & & & \vdots & \vdots \\
 & & & 0 & 0 & 1 & & & \\
\vdots & \vdots & & 0 & 1 & 0 & & \vdots & \vdots \\
 & & & 1 & 0 & 0 & & & \\
\vdots & \vdots & & & & & \ddots & \vdots & \vdots \\
0 & 0 & \dots & & \dots & & \dots & 1 & 0 \\
0 & 0 & \dots & & \dots & & \dots & 0 & 1
\end{pmatrix}
$$

$$= L_k L_{k+1} L_k$$

□

4.5. Permutation even odd property

For any leg, L_k, of a given ghost leg, in drawing ghost leg, the input in T_k will be transformed to T_{k+1} via L_k. Similarly, the input in T_{k+1} will be transformed to T_k via L_k. So, a leg of the ghost leg can be regarded as a transposition of I_k and I_{k+1}. Each matrix representation of ghost leg can be decomposed into products of transpositions. Even odd property of a ghost leg refers to the number of the transpositions in that ghost legs, e.g. a ghost leg is even if and only if it has even number of transposition.

Because the number of transposition of a ghost leg equals the number of legs. So a ghost leg's even odd property is related to the number of legs.

In later part of this report, we will deal with simplification of ghost legs which involves a reduction of two legs in the ghost leg each time. So the even odd property of the ghost leg is independent of simplification of ghost legs.

4.6. Infinite number of ghost legs with same permutation

For a specific permutation, there exist infinite number of ghost legs with same permutation.

Proof. Assume there is only finite number of ghost legs with the same permutation, so ghost leg representative matrix $M_k(k = 1, 2, 3, \dots, n)$ are all different.

Let S be a set containing all M_k. $M_k \in S$, $k = 1, 2, 3, \ldots, n$.

Consider the ghost leg M_i, for $I \leqslant n$, with the maximum number of legs. Express this ghost leg in the simplified matrix form $M_i = L_{x_1} L_{x_2} L_{x_3} \ldots L_{x_n}$.

Consider

$$\begin{aligned} M' &= L_1 L_1 L_{x_1} L_{x_2} L_{x_3} \ldots L_{x_n} \\ &= L_{x_1} L_{x_2} L_{x_3} \ldots L_{x_n} \\ &= M_i. \end{aligned}$$

Then ghost leg corresponds to M' and original ghost leg have same permutation. But ghost leg corresponds to M' has $n+2$ legs. Hence $M \notin S$, which leads to contradiction. Therefore, there exist infinite number of ghost legs with same permutation. □

5. Research

In this section, we mainly concern on 3 topics: Method of constructing a ghost leg according to a specified permutation, property of ghost legs constructed by bubble sort, and the method can be used to simplify any ghost leg into the simplest form.

5.1. Bubble Sort and its most simplicity

Bubble sort is a method to rearrange a group of numbers into ascending order. The idea of bubble sort (bbsort) is to continue comparing the magnitude of neighboring numbers, if the number in the right hand side is larger, then the two numbers exchange in position, otherwise, they stay unchanged. After several times of exchanging, the numbers will be in the right order. Here is an example of rearranging 5 numerical elements $(3, 4, 6, 8, 9)$ by using bbsort.

1st round

$$\begin{array}{ccccccccc} 9 & \rightarrow & 8 & & 6 & & 4 & & 3 \\ 8 & & 9 & \rightarrow & 6 & & 4 & & 3 \\ 8 & & 6 & & 9 & \rightarrow & 4 & & 3 \\ 8 & & 6 & & 4 & & 9 & \rightarrow & 3 \\ 8 & & 6 & & 4 & & 3 & & 9 \end{array}$$

The largest no. has transferred to the right most.

2nd round

8	→	6		4		3		9
6		8	→	4		3		9
6		4		8	→	3		9
6		4		3		8	→	9
6		4		3		8		9

The 2nd largest no. has transferred to the 2nd right most.

3rd round

6	→	4		3		8		9
4		6	→	3		8		9
4		3		6		8		9

The 3rd largest no. has transferred to the correct position.

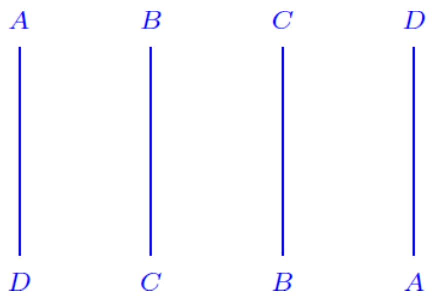
4th round

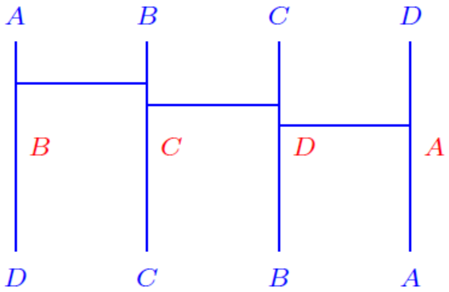
4	→	3		6		8		9
3		4	→	3		8		9

All numbers are arranged in ascending order, permutation thus succed.
(→ Exchange of two neighbouring elements)

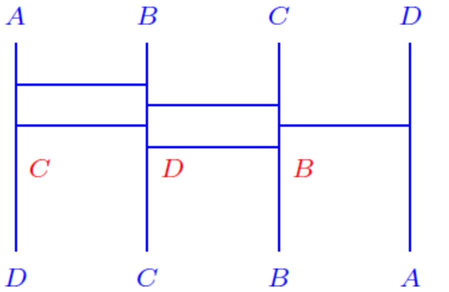
Ghost Leg can be constructed by using the idea of bbsort, here is an example:

Example 2: Using bbsort to construct a Ghost Leg representing the permutation of $(ABCD) \rightarrow (DCBA)$.

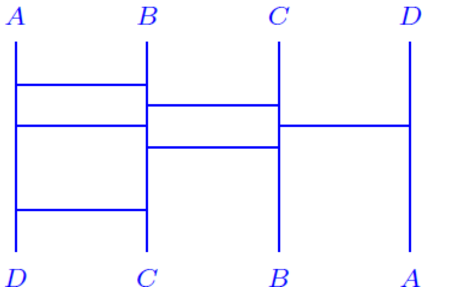




Transfer A to T_4



Then transfer B to T_3



Finally, C is transferred to T_2, D appears on T_1 naturally, and the Ghost Leg is completely constructed.

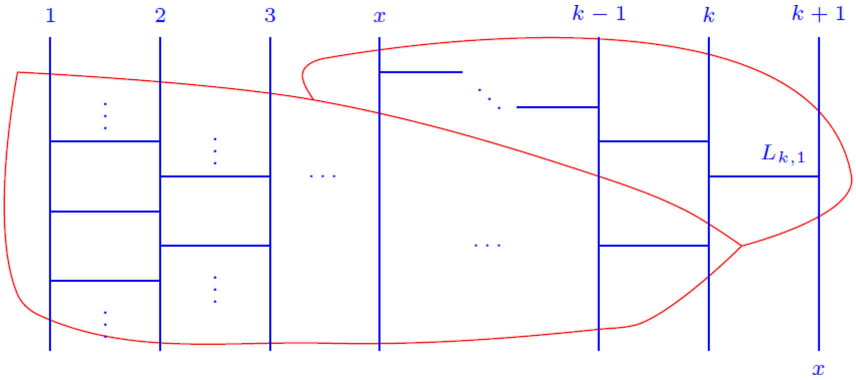
Ghost Leg constructed by bbsort must be in its simplest form, which is defined as Prime before. That means the Ghost Leg must contain the least number of Legs that can represent the permutation.

Proof. All Ghost Legs constructed by bbsort must not be further simplified. Let $P(n)$ be the preposition that n tracked Ghost Legs constructed by bbsort cannot be further simplified, for all $n > 1$.

Obviously, $P(2)$ is true.

Assume $P(k)$ is true, i.e. k tracked Ghost Leg constructed by bbsort cannot be simplified.

Consider $P(k+1)$.



From the figure, the right group of legs cannot be simplified.

The left group of legs can be regarded as a k tracked Ghost Leg constructed by bbsort. Therefore, according to $P(k)$, left group of legs cannot be simplified.

Right group of legs are all necessary (Otherwise the element X cannot be transferred to its final position. Therefore, the left and right groups cannot be simplified.

Hence the whole Ghost Leg cannot be simplified. Therefore, $P(k+1)$ is true.

By induction, $P(n)$ is true for all n. Therefore, the Ghost Leg constructed by bbsort must not be simplified, by definition, they are regarded as prime. □

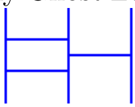
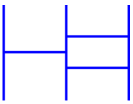
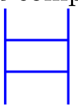
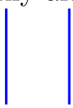
5.2. Maximum number of legs for prime $= \dfrac{n(n-1)}{2}$

In each round of bbsort, at least 1 element can be moved to its correct position of order, therefore, mostly $n-1$ round of bbsort is needed to construct a complete Ghost Leg (After $n-1$ elements are transferred to the correct position, the nth element will appear at correct position at once). In first round, $n-1$ transpositions is needed at most to transfer the element to track T_n. In second round, $n-2$ transpositions is needed at most to transfer the particular element to track T_{n-1}. Similarly, in $(n-1)$th round, only 1 transposition is needed.

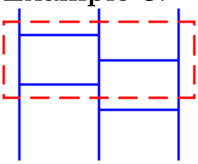
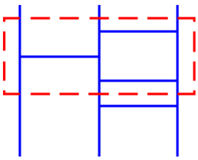
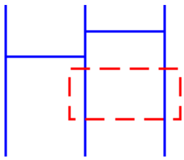
Obviously, when constructing a n-tracked Ghost Leg, $\dfrac{n(n-1)}{2}$ transpositions is needed at most to achieve the resultant Ghost Leg. As a n-tracked Ghost Leg can be regarded as a permutation with n elements, the number of legs for prime should not be more than $\dfrac{n(n-1)}{2}$. Therefore, if a Ghost Leg consists of more than $\dfrac{n(n-1)}{2}$ legs, it must be possible to be simplified.

Furthermore, in bbsort, the element results on T_n is transferred to its correct position at round 1, the Ghost Leg generated by bbsort can have 1 L_{n-1} at most. For instance, in Example 10, T_4 is involved only at round 1, thus the Ghost Leg can only consist of 1 L_3.

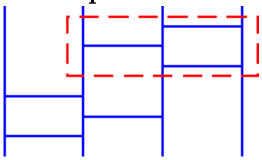
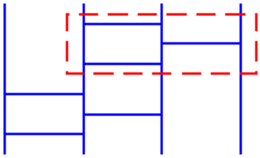
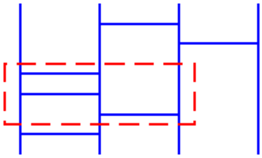
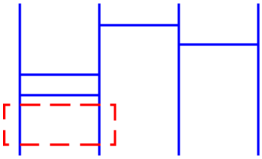
5.3. Bubblization

Bubblization is the method that must change any Ghost Legs into the form constructed by bbsort for the particular permutation. Since bbsort must construct the simplest Ghost Leg, Bubblization can simplify any Ghost Leg, unless that is already a Prime. Its idea is to use the identity,  $\equiv$ , to transfer disordered and complexly arranged legs to a particular side. And then use the identity,  $\equiv$  to eliminate extra and useless legs. Prime can be obtained by repeating this procedure.

Example 3:

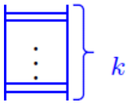
 is changed to  and further .

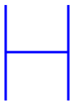
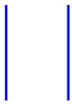
Example 4:

 is changed to  and further  and finally becomes .

Proof. (Bubblization can simplify all Ghost Leg.)

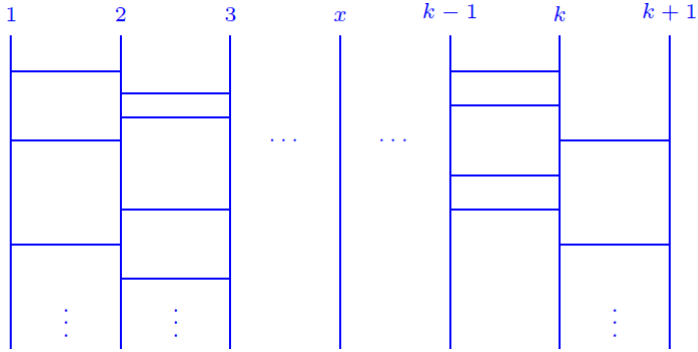
Let $P(n)$ be the proposition that all n-tracked Ghost Leg can be simplified to Prime by Bubblization, for all $n > 1$.

When $n = 2$, 

can be simplified as  (k is odd) or  (k is even). Therefore, $P(2)$ is true.

Assume $P(k)$ is true, i.e. all k-tracked Ghost Leg can be simplified to Prime by Bubblization.

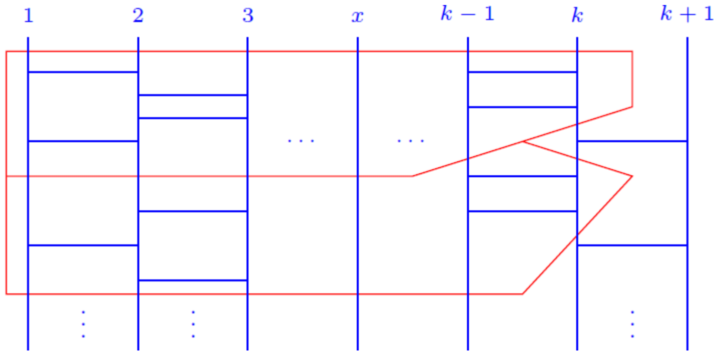
When $n = k + 1$



Legs with level higher than that of $L_{k,1}$ are grouped as Legs group A_1, and those lie between $L_{k,1}$ and $L_{k,2}$ are grouped as A_2.

Similarly, Legs lie between $L_{k,R-1}$ and $L_{k,R}$ are grouped as A_R.

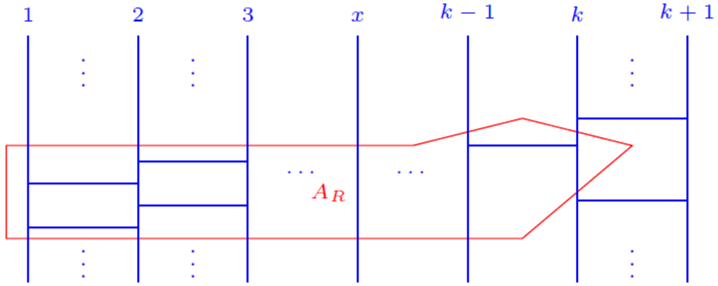
After reformation, we get



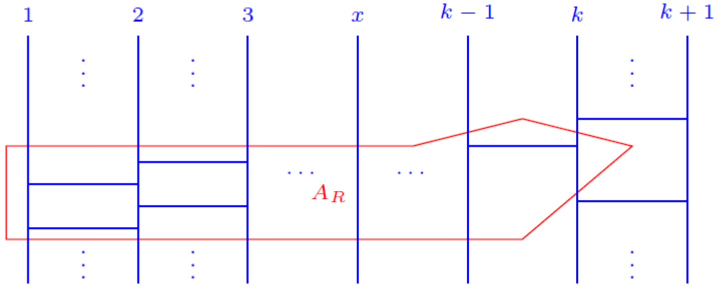
Since Ax(for all x) is a k-tracked Ghost Leg, by $P(k)$, Ax(for all x) can be simplified to Prime.

After Bubblization, there is only 1 Leg between T_{k-1} and T_k, therefore there is only 0 or 1 L_{k-1} lying between $L_{k,R-1}$ and $L_{k,R}$ (Refer to fig).

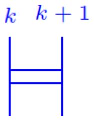
Case (1)

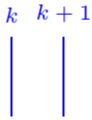


Case (2)

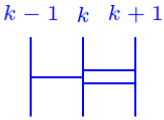


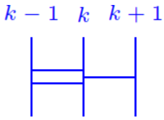
For any R, T_{k-1}, T_k, T_{k+1} are bubblized.

For case (1), 

is converted to 

.

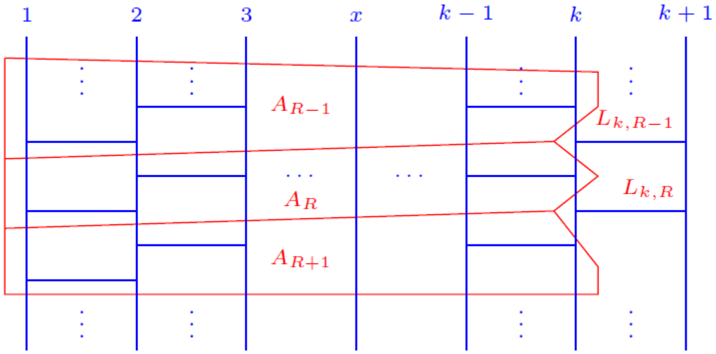
For case (2), 

is converted to 

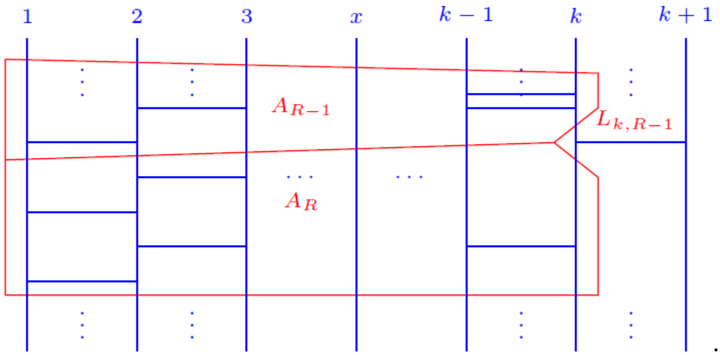
.

After a round of Bubblization, at least 1 L_k is reduced.

For case (2),



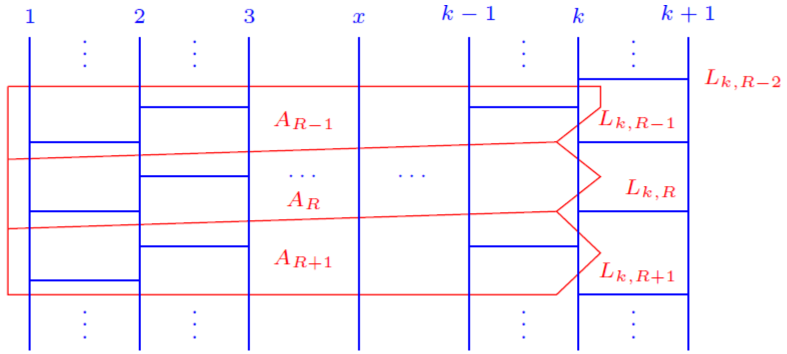
is converted to



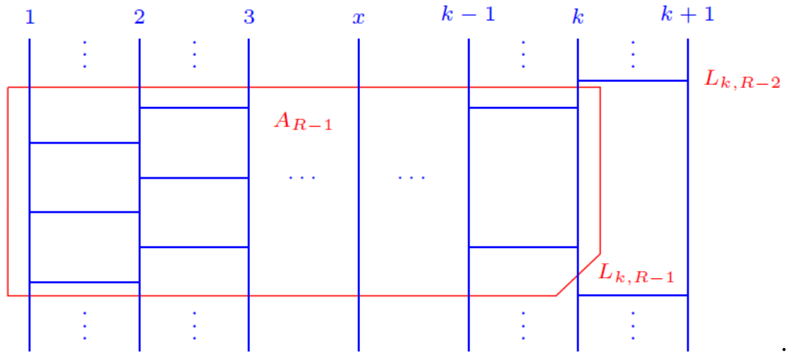
.

Since the level of L_k is changed, all Leg groups A are needed to redefine. In case (2), the new A_R is the combination of former A_R and A_{R+1}, and the new A_{R-1} is the former A_{R-1} with one extra L_{k-1}. Even though the former A_{R+1}, A_R, A_{R-1} are all Prime, but the A_{R+1}, A_R may not be Prime after combined, and an extra L_{k-1} added to A_{R-1} can change it into non-Prime, thus they can be further simplified.

For case (1),



is converted to



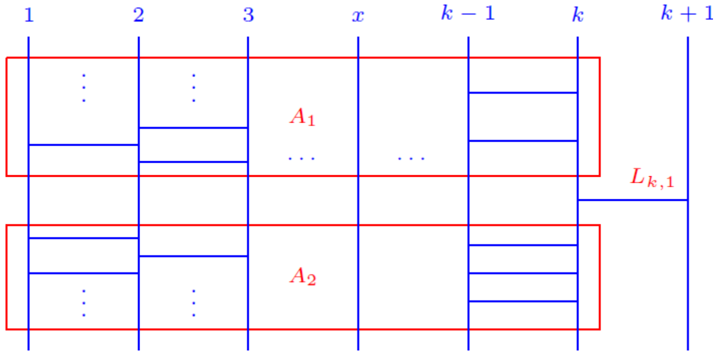
.

In case (2), the new A_{R-1} is fused by former A_{R-1}, A_R and A_{R+1}, thus the new A_{R-1} may be non-Prime and can be further simplified.

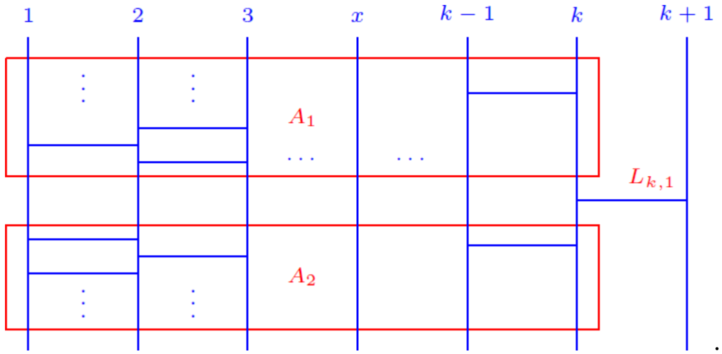
After a round of Bubblization, the Ghost Leg has come to a situation that, between every L_k, there is a Leg group, which can be regarded as a k-tracked Ghost Leg. It is similA_R to the situation before the round of Bubblization, thus the above procedures can be carried again. After each round of Bubblization, the number of L_k is reduced at least by 1, so finally there is only 0 or 1 L_k remained.

For the case with no L_k remained, the $(k+1)$-tracked Ghost Leg is actually a k-tracked Ghost Leg, as the $(k+1)$th track does not involve in permutation, thus it can be simplified to Prime by $P(k)$.

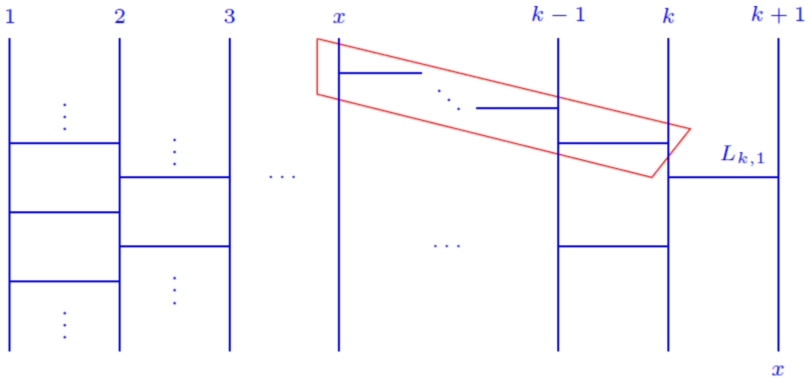
For the case with only 1 L_k remained, the L_k must be $L_{k,1}$, then the remaining Legs can be divided into groups A_1 and A_2. A_1 and A_2 can be bubblized into 2 Primes as follows.



is converted to

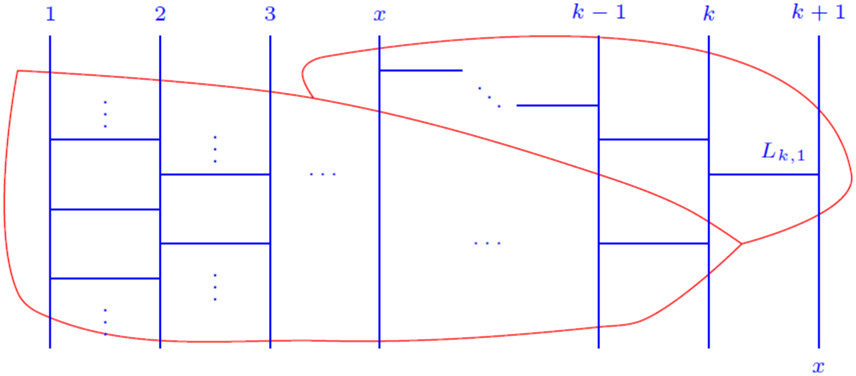


Since A_1 is bubblized Prime, the following situation should present.



that a group of terraced Legs (as the enclosed Legs in the figure) must lie at higher level than other legs, and they are the only Legs that have a higher level than $L_{k,1}$.

Then, applying Bubblization to the Legs with level lower than $L_{k,1}$, the following result would appear,



where the group lower than $L_{k,1}$ is Prime and the Legs higher than $L_{k,1}$ is a group of terraced Legs from X to $k+1$. That's the same as the Ghost Legs constructed by bbsort, and thus it is Prime.

Therefore, $P(k+1)$ is true.

By induction, $P(n)$ is true for all n.

Therefore, all n-tracked Ghost Leg can be simplified to Prime by Bubblization, for all $n > 1$. □

6. Conclusion

Drawing Ghost Leg is a game with long history. Traditionally, it was treated as a randomness generator, just like dices. It is because very few people were eager to put energy and time to research on Ghost Leg and thus it was thought as a mystery that is complicated and unable to be solved unless "Drawing" it.

But we are the exceptions. In this project, Ghost Leg has been successfully quantified, analyzed and solved. A modern mathematical way is used to analysis this mystery game, and it is found that Ghost Leg can be regarded as a way of expressing a permutation. Some other properties of Ghost Leg are also found. More important than that, Bubble Sort is suggested to be used as a method of constructing a Ghost Leg according to any specified permutation, and a method, Bubblization, is developed to simplify any Ghost Leg into the simplest form.

Although our project does not involve complicated mathematical theorems, and it does not have a close relationship with our normal life, but we still think it is a perfect job. Because the whole project, from the choosing of topic to using mathematical skill to solve the problems, are originally and entirely developed by us, at least we cannot find any people have done the same before. Through this project, we have learnt that there are still many things that are not totally understood, if we can take care about our surroundings, we may discover a lot.