

箱玉系の組合せ論的側面 – Combinatorial aspects of box-ball systems –

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

概要

箱玉系はソリトンセルオートマトン (超離散可積分系) の一種であり, 非線型な可積分系の離散化や, 量子代数の表現論における結晶理論と関連して注目すべき研究が進められている. 本講演では, 箱玉系をワードとタブローの組合せ論の観点から考察した [1]. 箱玉系の各時刻の状態はロビンソン・シェンステッド・クヌース対応 (RSK 対応) を用いてタブローの組 (P, Q) と同一視が可能であると述べ, 以下の 2 点を示した.

- (i) P シンボルは箱玉系の保存量を与える.
- (ii) Q シンボルは P シンボルに依存せずに時間発展する.

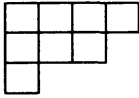
また Q シンボルの時間発展のアルゴリズムは明示的に書けることを示した.

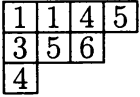
1 準備

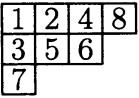
まず最初に, ここで使用するワードとタブローの組合せ論における基本事項をまとめておく. 詳細は文献 [3][5][6][7][12] 等を参照して頂きたい.

自然数 n の分割 $\lambda = (\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_l \geq 0)$ に対応させて, 上から k 段目に λ_k 個の箱を左端を揃えて敷き詰めた図形はヤング図形と呼ばれている. 本講演では, この図形の各箱に整数を, 右に非減少, 下に増大となるように入れたものをヤングタブロー (または単にタブロー) と定義し¹, 1 から n までの自然数を 1 つずつ, 右に増大, 下に増大となるように入れたものをスタンダードタブローと定義した.

例 1. 以下に例を示す.

• ヤング図形  $\longleftrightarrow \lambda = (4, 3, 1)$

• (ヤング) タブロー  $\left\{ \begin{array}{l} \leq \\ \wedge \end{array} \right\}$

• スタンダードタブロー  $\left\{ \begin{array}{l} < \\ \wedge \end{array} \right\}$

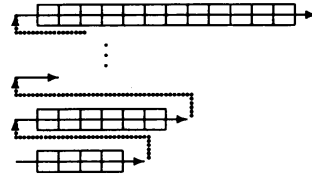
¹セミスタンダードタブローのこと.

下の段から上の段へ向かって、各々の段で左から右へと、タブロー T に記入された数字を全て拾い読み、左から右へ一列に並べた数列を、そのタブローのワードといい、 $W(T)$ と表記する。何か数列が与えられた時、それをワードとするタブローが存在するならば、そのワードはタブローワードと呼ばれる。

タブローに整数を挿入して新たなタブローを作るバンピングという操作がある。ある段に数字 i を挿入する時は、その段に i より大きな数字があるかどうかを見て、無ければ段の右端に新しい箱を付け加えて i を入れて終了、 i より大きい数字があればそのうち最小の数字 j を i で置き換えて、 j を 1 つ下の段へ同様に挿入する。この操作をタブローの一番上の段から下の段へ向かって順に施すのがバンピングのアルゴリズムである。

バンピングの操作は 3 つの数字を入れ換える操作の繰り返しであると考えられ、その最も基本となる変換はクヌース基本変換と呼ばれている。2 つのワードが、クヌース基本変換、及びその逆変換によって移りあう時、それらをクヌース同値なワードであるという。

例 2. 以下に例を示す。



- タブローからワードを読む方向
- バンピング (ロウ・インサーション)

$$(u x' v) x \longrightarrow x' (u x v) \quad (u \leq x < x' \leq v)$$

- クヌース基本変換

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

- クヌース同値 (記号: $\approx$)

$$5152431245 \approx 5415213245$$

2 つの整数 i, j を上下に置いた組 $\begin{pmatrix} i \\ j \end{pmatrix}$ を、横に辞書式順序に並べて出来る 2 段の配列をバイワードと定義し、また、上下の数字を反転させて改めて辞書式順序に並べ直したものを、元のバイワード w に対してデュアルバイワード w^* と定義する。

バイワードは、台の等しいタブローの組 (P, Q) と一対一に対応していて、この対応を Robinson-Schensted-Knuth 対応 (RSK 対応) と呼んでいる。具体的には、バイワードの下段に並んだ数列を左から順にバンピングして得られるタブローが P シンボルであり、その過程において $\begin{pmatrix} i \\ j \end{pmatrix}$ の j をバンピングして新たに加わった箱と同じ位置の箱に i を入れることで得られるもう 1 つのタブローが Q シンボルである。またデュアルバイワードに対しては (Q, P) が対応するという対称性がある。

例 3. 以下に例を示す。

• バイワード $w = \begin{pmatrix} i_1 & i_2 & \cdots & i_k & \cdots & i_n \\ j_1 & j_2 & \cdots & j_k & \cdots & j_n \end{pmatrix}$

辞書式順序 $\begin{cases} i_1 \leq i_2 \leq \cdots \leq i_n \\ j_k \leq j_{k+1} \text{ if } i_k = i_{k+1} \end{cases} \quad (k = 1, \dots, n-1)$

• バイワード $w = \begin{pmatrix} 1 & 2 & 2 & 4 & 5 & 7 \\ 3 & 1 & 5 & 2 & 2 & 1 \end{pmatrix}$ のデュアルバイワードは $w^* = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 5 \\ 2 & 7 & 4 & 5 & 1 & 2 \end{pmatrix}$.

• RSK 対応 $\{\text{バイワード}\} \xleftrightarrow{1:1} \{(P, Q)\}$

• バンピング過程 $w = \begin{pmatrix} 1 & 2 & 2 & 4 & 5 & 7 \\ 3 & 1 & 5 & 2 & 2 & 1 \end{pmatrix} \rightarrow (P, Q)$

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

2 箱玉系

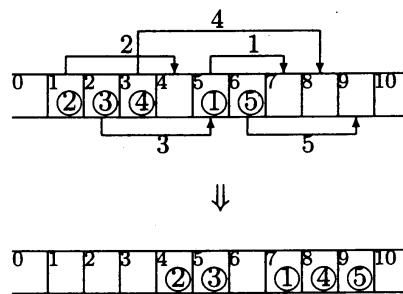
ここでは、箱玉系のスタンダードな場合を用いて、論文 [1] の主結果を述べる。箱玉系とは、整数で番号付けられた箱を左右無限に並べ、その中をある一定の規則に従って、1 から n までの自然数で色分けされた有限個の玉を移動させる事によって得られる一種の力学

系である。この論文では、スタンダード箱玉系を、「全部で n 個の相異なる色の玉を用いて、それぞれの箱には各々 1 個までしか玉を入れてはいけない」という条件に基づいて時間発展させたものであると定義し、また玉の色を表すのに自然数 $1, 2, \dots, n$, 玉を入れる事が出来る空箱の状態を表すのに記号 $e (= n + 1)$ を用いる。

まず、スタンダード箱玉系における時間発展則 (オリジナルアルゴリズムと呼ぶ) を次のように定義する。

- (1) 時刻 t から $t + 1$ において、全ての玉を一度だけ移動させる。
- (2) 「1」の玉を、その右にある一番近くの空箱へ移動させる。
- (3) 同じ要領で他の玉も「2, 3, $\dots$, n 」の順に移動させる。

例 4. 以下は、 $n = 5$ の場合の時間発展 (1 ステップ) の例である。



例 5. 以下に、 $n = 9$ の場合の時間発展の例を示す²。

```

-----15789-----236-----4-----
-----15789-----236-----4-----
-----15789-----236-----4-----
-----15789-----2634-----
-----157682349-----
-----7_568_12349-----
-----7_568_12349-----
-----7_568_12349-----

```

次に、スタンダード箱玉系の時間発展における各時刻の状態をバイワードに対応させて、主定理を述べる。

スタンダード箱玉系において、ラベル i の箱に入っている玉の色を a_i で表し、空箱の場合は $a_i = e$ と表示することにとすると、各時刻の状態は $1, \dots, n$ 及び $e = n + 1$ からなる無限列 $\dots a_{-1} a_0 a_1 \dots$ として表すことが出来る。更に左から右へ順に見て、玉の入った箱のラベル $i_1, i_2, \dots, i_n$ に対応する玉の色 $a_{i_1}, a_{i_2}, \dots, a_{i_n}$ を記録することで、空箱を無視した形のバイワード表示が可能である。逆に勝手なバイワード w が与えられた時、その中の整数の組 $(i_k)_{a_{i_k}}$ に対して「ラベル i_k の箱に、色 a_{i_k} の玉が入っている」と解釈すること

²アンダーバーは空箱を表している。

で、スタンダード箱玉系の時間発展における状態とバイワードは1対1に対応させることができる。

スタンダード箱玉系での玉の色は、 $1, 2, \dots, n$ が各々1個ずつなので、対応するデュアルバイワードは、上段に $1, 2, \dots, n$ が順序良く並んだような形になる。このデュアルバイワードの下段の数値列 $b = (b_1, b_2, \dots, b_n)$ は箱ラベル $i_1, i_2, \dots, i_n$ を並べ替えたもので(箱ラベル列と呼ぶ)、左から $b_1, \dots, b_n$ の順に、そのラベルの箱に入っている玉を移動させるというのが、オリジナルアルゴリズムになっている事に気付く。RSK 対応で知られているように、バイワードは同じ台のタブローの組 (P, Q) と一対一に対応させることができる。そこでスタンダード箱玉系において、与えられた状態 $\dots a_{-1}a_0a_1\dots$ に対して (P, Q) を対応させて、これをスタンダード箱玉系の状態と見なすことにする。ここで以下の定理を得る。

定理 2.1. スタンダード箱玉系を (P, Q) の時間発展とみなした時、

- (i) P シンボルは時間発展において保存する。
- (ii) Q シンボルは P シンボルに依存せずに発展する。

例 6. ここで定理の主張を例示する。まず例4を思い出し、時間発展の前後の各状態をバイワードに対応させると以下ようになる。

$$\begin{array}{c|c|c|c|c|c|c|c|c|c|c|c} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ \hline & 2 & 3 & 4 & & 1 & 5 & & & & \end{array} \Rightarrow \begin{array}{c|c|c|c|c|c|c|c|c|c|c|c} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ \hline & & & & 2 & 3 & & 1 & 4 & 5 & \end{array}$$

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

更にタブローの組に対応させると次のような結果になる。

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

ここで、 $P = P'$ になっているのが定理の一つ目の主張であり、 P に依存せずに Q だけを見て Q' が分かるというのが二つ目の主張である。

定理の証明は後述する。また Q シンボルの発展則の記述には、次に述べる運搬車アルゴリズムが重要な役割を果たす。

あるワード $w = (w_1, w_2, \dots, w_n)$ から別のワード $w' = (w'_1, w'_2, \dots, w'_n)$ への変換に運搬車と呼ばれる非減少数列 $C = (c_1, \dots, c_m)$ を用いるアルゴリズムを導入する。運搬車が元のワード w に沿って左から右へと数字の積み下ろしをしながら通り過ぎた結果、新しいワード w'_k が得られる。数字の積み下ろし規則は次のように定義する。

$C_{k-1} = (c_1^{(k-1)}, c_2^{(k-1)}, \dots, c_m^{(k-1)})$ ($c_1^{(k-1)} \leq c_2^{(k-1)} \leq \dots \leq c_m^{(k-1)}$) を運搬車に既に積み込まれている数字の列とする。 w_k を今から積み込む数字とする。 w_k と、 C_{k-1} に含まれている数字とを比べて、 w_k より大きい数字が C_{k-1} に存在する時は、その中の最小値が下ろされて、代わりに w_k が積み込まれる。そのような数字が存在しない時は、単に C_{k-1} の中の最小値が下ろされて、代わりに w_k が積み込まれる。

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

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

C_k = the sequence of numbers obtained from C_{k-1}
by replacing a w'_k by w_k .

与えられた2つの有限列 $C = (c_1, c_2, \dots, c_m)$ ($c_1 \leq \dots \leq c_m$) と $w = (w_1, w_2, \dots, w_n)$ に対して, $C_0 = C$ として, 上記の積み下ろし規則を繰り返す事により新たに $C' = C_n$ と w' が得られる. この変換 $(C, w) \rightarrow (C', w')$ を運搬車アルゴリズムと定義する. ここで運搬車アルゴリズムはクヌース変換の繰り返しである事に注意しておく.

$$\begin{aligned} Cw &= C_0 w_1 w_2 w_3 \cdots w_n \approx w'_1 C_1 w_2 w_3 \cdots w_n \\ &\approx w'_1 w'_2 C_2 w_3 \cdots w_n \\ &\quad \vdots \\ &\approx w'_1 w'_2 w'_3 \cdots w'_n C_n = w' C' \end{aligned}$$

論文では前述の定理の証明をする為に2つの命題を与えているが, 共に箱玉系の時間発展が運搬車アルゴリズムを用いて記述できるという内容のものである. 片方は箱玉系の状態を表す数列 $\dots, a_{-1}, a_0, a_1, \dots$ に沿って運搬車を走らせ, もう片方ではデュアルに着目して「箱ラベル列」に沿って運搬車を走らせる. このようにデュアルで考えることにより, 時間発展の本質が見やすくなる.

3 運搬車アルゴリズム

ここでは既に述べた運搬車アルゴリズムに関する2つの命題を与える. 時間発展の1ステップを考えるにあたり, 左右無限の数列を考慮する必要は無く, 以下で定める最低限必要な区間 $[p, q]$ のみを採用する.

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

例 7.

		p											q		
箱ラベル:	$\dots$	0	1	2	3	4	5	6	7	8	9	10	11	12	$\dots$
	$\dots$	e	2	3	4	e	1	5	e	e	e	e	e	e	$\dots$
	$\dots$	e	e	e	2	3	e	1	4	5	e	e	e	e	$\dots$

命題 3.1. スタンダード箱玉系の与えられた状態から左右無限に伸びる e の列を無視して、残った数字の列を $A = (a_p, a_{p+1}, \dots, a_{q-1}, a_q)$ とする. 但し, p, q は上で定めたものとする. このとき, 時刻 t から $t+1$ へのオリジナルアルゴリズム $A \rightarrow A'$ は, n 個の e から成る列 $C = (e, e, \dots, e)$ を初期の運搬車とみなした運搬車アルゴリズムとして記述可能である.

例 8. 以下に例を示す.

$$\begin{array}{cccccccccccccccc}
 \dots & e & 2 & 3 & 4 & e & 1 & 5 & e & e & e & e & e & e & \dots \\
 \dots & e & e & e & e & 2 & 3 & e & 1 & 4 & 5 & e & e & e & \dots
 \end{array}$$

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

次にデュアルの場合を考える. 既に述べたように, スタンダード箱玉系において, デュアルバイワード表記を用いた場合, 下の段に現われる列は箱ラベルを並べ替えたものであり, これを「箱ラベル列」と呼ぶことにする.

例 9. 以下に例を示す. まず例 6 のバイワード表記を思い出す.

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

これをデュアルバイワード表記に直すと次のようになる.

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

このとき箱ラベル列の発展とは $b = (51236) \rightarrow b' = (74589)$ のことである.

命題 3.2. スタンダード箱玉系の与えられた状態に対し, 時刻 t から $t+1$ への, 箱ラベル列の発展 $b \rightarrow b'$ は, $C = (l_1, l_2, \dots, l_m)$ を初期の運搬車とみなした運搬車アルゴリズムとして記述可能である. 但し, C は上で定めた区間 $[p, q]$ における空箱のラベルを集めて出来る非減少列として定義される.

例 10. 以下に例を示す.

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

$$\begin{array}{cccccc}
 4, 7, 8, 9, 10, 11 \begin{array}{c} 5 \\ \hline 7 \end{array} & 4, 5, 8, 9, 10, 11 \begin{array}{c} 1 \\ \hline 4 \end{array} & 1, 5, 8, 9, 10, 11 \begin{array}{c} 2 \\ \hline 5 \end{array} & 1, 2, 8, 9, 10, 11 \begin{array}{c} 3 \\ \hline 8 \end{array} & 1, 2, 3, 9, 10, 11 \begin{array}{c} 6 \\ \hline 9 \end{array} & 1, 2, 3, 6, 10, 11
 \end{array}$$

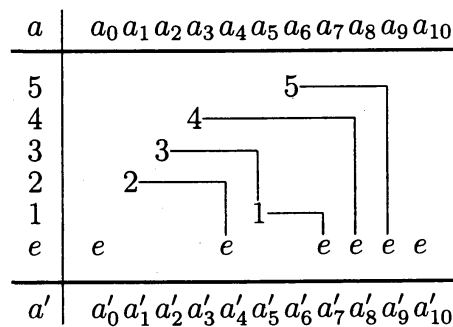
尚, 箱ラベル列の時間発展を T^* という記号を用いて表すことにする: $b' = T^*(b)$.

4 主結果の証明(概略)

ここでは、前節で与えた2つの命題を証明し、それらの命題を用いて定理を証明する。

箱玉系のオリジナルアルゴリズムを注意深く分析することで、これが運搬車アルゴリズムを用いて2通りに記述できる事が分かり、2つの命題が証明できる。更に運搬車アルゴリズムがクヌース変換の繰り返しであるという事実から、時間発展においてクヌース同値性が保たれる事が分かる。概ね、これらのアイデアを基本にして定理の証明が完成する。

2つの命題の証明には次のような図を利用する。ここでも例4と同じデータを用いる事にして、図の見方を例で説明する。まず一番上の段には最初の状態 a_i が並んでいて、一番下には1ステップ後の状態 a'_i が並んでいる。真ん中は、最初の状態 a_i の各々の値、即ち玉の色を示す1から $n=5$ までの数値と空箱を示す記号 e が、その値ごとに段を変えて並んでいると思うことにする。次にオリジナルアルゴリズムに従って、1から5までと、それらが移動する先にある数値とを線で結んでいる。例えば、最初に1を動かす時、その右にあって最も近い空箱のラベルは7であるので、 a_7 の真下に存在する e と線で結ぶ。同様に2は a_4 の真下にある e と線で結ぶ。次に3を動かす時は、1が動いた後の空箱(ラベル5)が最も近いので、 a_5 の真下に存在する1と線で結ぶ。同様にして1から5全ての数値から線を引いたら作業は終了し、以下の図が出来あがる。



さて、オリジナルアルゴリズムに従って上の図が出来たが、線の引き方としては

各々の数値と、その右下にある最も近くの数値とを線で結ぶ。

但し、互いに交わらないようにする。

という法則がある事に気付く。以下、この法則に従う事にして、別の視点から同じ図を作ることを考える。

命題 3.1 の証明

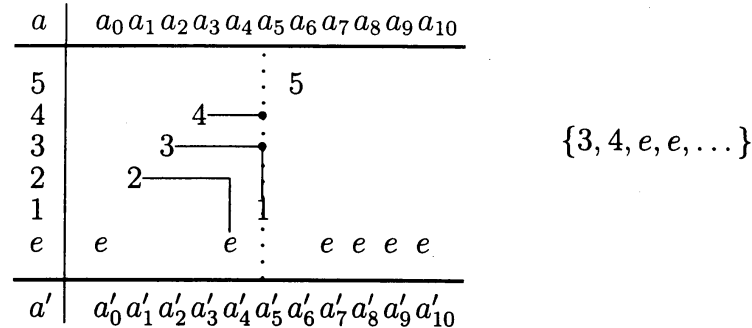
まず、左の玉から右の玉へと順番に(今の例だと、2, 3, 4, 1, 5の順序で)着目して線を引くことにすると、命題 3.1 の証明が出来る。

具体的に今、空の運搬車³が左から走って来たとする。 $a_1 = 2$ の所で運搬車に2が乗り、 $a_2 = 3$ の所で運搬車に3が乗り、 $a_3 = 4$ の所で運搬車に4が乗る。次に $a_4 = e$ の所では運搬車に乗っている2, 3, 4のうち最小の2が降りる。これは各々2, 3, 4から横に線が伸び

³ $C = \{e, e, e, e, e\}$

ていて、そのうち一本と e とを結ぶと考えた時に、互いに交わらないという規則に従えば、2 が選択されるからである。

以下の図は、 $a_5 = 1$ に着目した時の例である。1 と線で結ばれる数値の候補は 3, 4 であり、運搬車には 3, 4 が乗っている⁴という状態である。



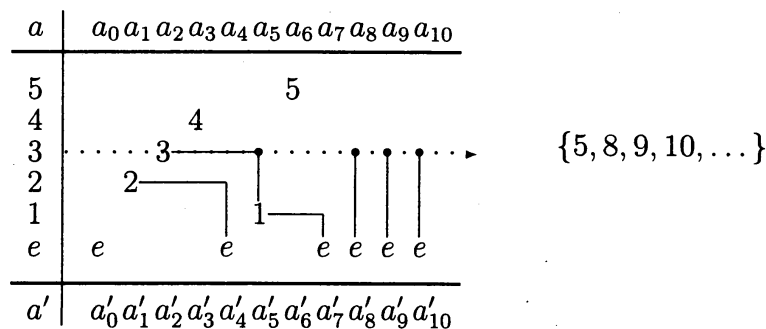
このように、線で結ばれる数値の候補を運搬車に乗せていると解釈することで、命題 3.1 のように運搬車アルゴリズムが適用される事は自然に理解できる。

命題 3.2 の証明

次に、図の下の方から上の玉へ、即ちオリジナルアルゴリズムの場合と同じ順序で、数値に着目して線を引いていく。但し今度は a_i の値 (玉の色) では無く箱ラベル i に注目することにする。

まず $a_5 = 1$ に着目した時、その右にある空箱たちのラベルは 7, 8, 9, 10, ... であるので、それらのラベルが運搬車に乗っていると考える。当然の事ながら、 $a_5 = 1$ と線で結ばれるのは、ラベル 7 の空箱であるので、運搬車からは 7 が降ろされて、代わりにラベル 5 が乗ることになる。運搬車には常に空箱のラベルが乗っていることに注意されたい。

以下の図は、 $a_2 = 3$ に着目した時の例である。これと線で結ばれる空箱ラベルの候補は 5, 8, 9, 10, ... であり、運搬車には 5, 8, 9, 10, ... が乗っているという状態である。



このように、線で結ばれる空箱ラベルの候補を運搬車に乗せていると解釈することで、命題 3.2 のように運搬車アルゴリズムが適用される事は先の命題 3.1 よりも素直に理解できる⁵。

⁴具体的には、 $C = \{3, 4, e, e, e\}$

⁵論文 [1] では、こちらを先に証明している。

以下、定理 2.1 の証明を簡単に説明する．詳細は論文 [1] を参照して頂きたい．

定理 2.1 の証明 (i)

まず天下りであるが次の補題を与える．証明は文献 [3] 等を参照の事．

補題 1. w と w' が互いにクヌース同値なワードであり，それらから最大の数 N を全て取り除いて出来るワードが w_0 と w'_0 であるとき， w_0 と w'_0 もまた互いにクヌース同値なワードである．

命題 3.1 より，スタンダード箱玉系の時間発展は運搬車アルゴリズムで記述でき，それがクヌース変換の繰り返しであるという事から，発展の様子は一般に以下のような形で表記できる．

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

今，空箱を意味する e は他の数 $1, 2, \dots, n$ よりも大きいと見なしている．故に CA 及び $A'C$ から全ての e を取り除いて出来るワードを A_e と A'_e であるとする．補題 1 より，これらは互いにクヌース同値である事が分かる．(i.e., $A_e \approx A'_e$.) ここで，クヌース同値なワードは等しいタブローに対応している事が知られているので， A_e 及び A'_e をバンピングして得られるタブロー P は等しい．この P シンボルが時間発展における保存量であると考えられる．また A_e というのが，既に導入したバイワードの下の段に現われる列に他ならない事を注意しておく．このようにして定理 2.1 の一つ目の主張を証明する事が出来る．

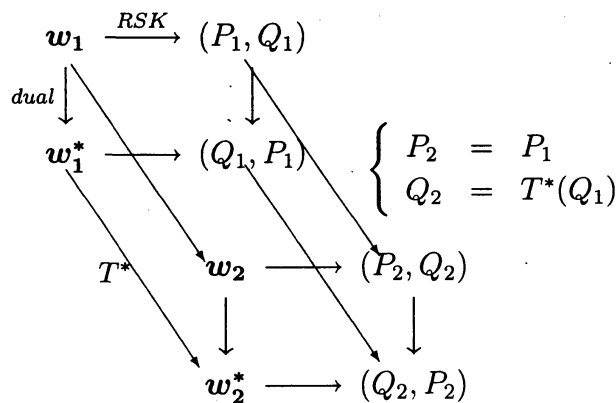
定理 2.1 の証明 (ii)

再度，天下りであるが以下に 2 つの補題を与える．証明は論文 [1] を参照して頂きたい．

補題 2. a と b がクヌース同値なワードであるならば， $T^*(a)$ と $T^*(b)$ もクヌース同値なワードである．

補題 3. b がタブローワードならば， $T^*(b)$ もタブローワードである．

以下，これら 2 つの補題を使って定理 2.1 の二つ目の主張を証明する．次に与える図を参考に後の説明を読んで頂きたい．



簡単の為、証明の要点を箇条書きにする。

- $b \approx W(Q_1) \rightarrow T^*(b) \approx T^*(W(Q_1))$ ($\because$ 補題 2)
- $T^*(b) = b'$ and $b' \approx W(Q_2) \rightarrow W(Q_2) \approx T^*(W(Q_1))$
- $W(Q_1)$ はタブローワード $\rightarrow T^*(W(Q_1))$ もタブローワード ($\because$ 補題 3)
- よって, $W(Q_2) = T^*(W(Q_1))$.

ここで、箱ラベル列 b 及び b' は、デュアルバイワード w_1^* と w_2^* の下の段に現われる列の事である。(前の図を参照。)

上に箇条書きした内容を少し詳しく説明しておく。 $b \approx W(Q_1)$ というのは即ち、 b をバンピングした結果、タブロー Q_1 が得られるというのと等しい。補題 2 より、互いにクヌース同値なワードはデュアルの意味での時間発展 (T^*) 後も互いにクヌース同値になっているので、 $T^*(b) \approx T^*(W(Q_1))$ を得る。今、 $T^*(b) = b'$ であり、 b' をバンピングして得られるタブローは Q_2 である。故に $T^*(b) \approx W(Q_2)$ を得る。先ほどの $T^*(b) \approx T^*(W(Q_1))$ を考慮すると、結果として $W(Q_2) \approx T^*(W(Q_1))$ が得られる。 $W(Q_1)$ は明らかにタブローワードであるので、補題 3 より、 $T^*(W(Q_1))$ もタブローワードである事が分かる。2 つのタブローワードが互いにクヌース同値であるという事は即ち、全く等しいワードである事を意味するので、最終的に $W(Q_2) = T^*(W(Q_1))$ を得る。このことから、 Q_2 は Q_1 だけから決まっている事が分かり、定理 2.1 の二つ目の主張が示せた。

証明終.

最後に、タブローワードとタブローを同一視することで、 Q シンボルの時間発展は以下のように定義することが可能である。但し、ワード w をバンピングして得られるタブローを $\text{Tab}(w)$ と表記する。

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

ここで再度、区間 $[p, q] \subset \mathbb{Z}$ を用いて、以下のような命題を与える。

命題 4.1. スタンダード箱玉系において、 Q シンボルの時間発展則は箱ラベルの運搬車アルゴリズムによって記述できる。この時、運搬車の初期状態は、区間 $[p, q]$ にある全ての空箱のラベルを列挙した非減少列 $C = (l_1, \dots, l_m)$ で定義される。また運搬車は、タブロー Q の各々の段を左から右へ、下の段から上の段へ向かって走る。

スタンダード箱玉系の、ある状態が (P, Q) というタブローの組で与えられた時、次の時刻の状態は Q のみを見れば分かるが、この時に (Q, P) から、これに対応するバイワードを再現⁶し、その下の段のワード (箱ラベル列) に運搬車アルゴリズムを適用する必要は無く、直接 Q から読み取ったタブローワードに運搬車アルゴリズムを適用すれば良いというのが、命題 4.1 の主張である。

⁶実際に逆バンピングを行う場合は P も見る必要がある。

5 箱玉系の一般化

論文 [1] では, 一般に「全部で m 個の玉が存在し, 同色の玉がいくつか存在してもよく, それぞれの箱には各々 1 以上の容量が予め設定されていて, 同じ箱に 1 個より多くの玉が入ることもある」という条件まで拡張して, スタンダード箱玉系の場合と同じ内容の定理を述べた. この定理はスタンダード箱玉系の場合と同様の手順で証明をする事が出来る⁷.

参考文献

- [1] K. Fukuda, Box-ball systems and Robinson-Schensted-Knuth correspondence, (math.CO/0105226 v3).
- [2] K. Fukuda, M. Okado and Y. Yamada, Energy functions in box ball systems, *Internat. J. Modern Phys. A* **15** (2000), no. 9, 1379–1392.
- [3] W. Fulton, *Young Tableaux*, London Mathematical Society Student Texts **35**, Cambridge University Press (1997).
- [4] G. Hatayama, A. Kuniba and T. Takagi, Soliton cellular automata associated with finite crystals, *Nuclear Phys. B* **577**(2000), 619–645.
- [5] D.E. Knuth, *The art of computer programming*, Volume 3, *Sorting and searching*, Addison-Wesley Series in Computer Science and Information Processing. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1973.
- [6] I.G. Macdonald, *Symmetric functions and Hall polynomials*, Second Edition, Oxford University Press, 1995.
- [7] B.E. Sagan, *The symmetric group. Representations, combinatorial algorithms, and symmetric functions*, Second edition, Graduate Texts in Mathematics, 203. Springer-Verlag, New York, 2001.
- [8] D. Takahashi, On some soliton systems defined by using boxes and balls, *Proceedings of the International Symposium on Nonlinear Theory and Its Applications (NOLTA '93)*, 1993, pp.555–558.
- [9] D. Takahashi and J. Matsukidaira, Box and ball system with a carrier and ultradiscrete modified KdV equation, *J. Phys. A* **30** (1997) L733–L739.
- [10] D. Takahashi and J. Satsuma, A soliton cellular automaton, *J. Phys. Soc. Jpn.* **59** (1990) 3514–3519.

⁷具体例と共に論文 [1] を参照のこと.

- [11] T. Tokihiro, A. Nagai, and J. Satsuma, Proof of solitonical nature of box and ball systems by means of inverse ultra-discretization, *Inverse Problems* **15** (1999), no.6, 1639–1662.
- [12] 堀田良之, 「加群十話–代数学入門–」 sugaku books 3 朝倉書店, 1988.
- [13] 数理科学 NO.435, SEPTEMBER 1999.
- [14] 時弘哲治, 「箱玉系の数理–超離散ソリトンの世界」 2002 年度日本数学会秋季総合分科会 総合講演・企画特別講演 アブストラクト pp.32–41.