

# 計算万能性を有する単純な 1 次元可逆セルオートマトン

## Simple Universal One-Dimensional Reversible Cellular Automata

森田 憲一

Kenichi Morita

広島大学大学院工学研究科

Graduate School of Engineering, Hiroshima University

### 概要

計算万能な 1 次元可逆 (単射) セルオートマトン (RCA) でどれほど状態数の少ないものが存在するかという問題について研究した。ここでは、計算万能性がすでに知られているサイクリックタグシステム (CTAG) を模倣できる単純な RCA を 2 種類与えた。最初に、任意の CTAG を無限状相で模倣できる RCA が 36 状態で構成できることを示した。次に、停止性が定義できるような任意の CTAG を有限状相で模倣する RCA が 98 状態で構成できることを示した。

キーワード: 可逆セルオートマトン, 計算万能性, サイクリックタグシステム

Keywords: reversible cellular automata, computation universality, cyclic tag system

## 1 はじめに

セルオートマトン (CA) の理論においては計算万能性 (任意の Turing 機械が模倣できる性質) は非常に重要な問題である。1 次元の場合、わずか 2 状態の計算万能 CA が存在することが Cook [1] によって証明されている。彼は、まずサイクリックタグシステム (CTAG) の計算万能性を示し、次に任意の CTAG が規則番号 110 を持つ 2 状態 3 近傍の CA で無限状相を持つものによって模倣できることを示した。

本論文では、計算万能な 1 次元可逆 CA (RCA) について考察する。従来、森田ら [3] により、任意の可逆 Turing 機械 (その中には万能可逆 Turing 機械も含まれる) に対し、それを模倣する 1 次元 RCA が構成できることが示されており、それから計算万能な RCA の存在が導かれる。しかし、これまでは少数 (例えば数十個) の状態しか持たない計算万能 1 次元 RCA は知られていなかった。ここでは、そのような RCA を得るために、分割 CA の枠組みを用いて、任意の CTAG を模倣できる 1 次元 RCA のモデルを二種類構成した。一つめは 36 状態のモデルで、停止の概念が定義されていない任意の CTAG を無限状相上で模倣できる。二つめは 98 状態のモデルで、停止の概念が定義されている任意の CTAG を有限状相上で模倣できる。

## 2 準備

**定義 2.1** 3近傍決定性1次元分割セルオートマトン (partitioned cellular automaton, PCA) は,  $P = (\mathbf{Z}, (L, C, R), g, (\#, \#, \#))$  によって定義される. 但し,  $\mathbf{Z}$  はすべての整数の集合,  $L, C, R$  は各セルのそれぞれ左, 中央, 右の部分の非空の状態集合,  $g : R \times C \times L \rightarrow L \times C \times R$  は局所関数,  $(\#, \#, \#) \in L \times C \times R$  は静止状態で  $g(\#, \#, \#) = (\#, \#, \#)$  を満たす.

$P$  の (あるいは集合  $Q = L \times C \times R$  上の) 状相 (configuration) は写像  $\alpha : \mathbf{Z} \rightarrow Q$  によって定められる.  $Q$  上のすべての状相の集合を  $\text{Conf}(Q)$  と書く.  $|\{x \mid \alpha(x) \neq (\#, \#, \#)\}|$  が有限であるとき  $\alpha$  を有限状相, そうでないとき無限状相という (但し  $|S|$  は集合  $S$  の要素数を表す).

ここで LEFT (CENTER, RIGHT) を, 3項組  $L \times C \times R$  から左 (中央, 右) の成分を取り出す射影とする. このとき,  $P$  の大域関数  $G : \text{Conf}(Q) \rightarrow \text{Conf}(Q)$  は次のように定義される.

$$\forall x \in \mathbf{Z} : G(\alpha)(x) = g(\text{RIGHT}(\alpha(x-1)), \text{CENTER}(\alpha(x)), \text{LEFT}(\alpha(x+1)))$$

$P$  が大域的に可逆であるとは  $G$  が単射であることを言い, また局所的に可逆であるとは  $g$  が単射であることを言う.

以下では  $f(q_r, q_c, q_l) = (q'_l, q'_c, q'_r)$  を  $(q_r, q_c, q_l) \rightarrow (q'_l, q'_c, q'_r)$  の形式, あるいは図1や図3の形式で書き,  $P$  の局所遷移規則と呼ぶ.  $\square$

**命題 2.1** [3] 任意の PCA は, 大域的に可逆であるとき, かつそのときに限り局所的に可逆である.

**命題 2.2** [3] 任意の PCA  $P$  に対し, 大域写像が  $P$  と一致するような CA が存在する.

これらの命題から, 望みの性質を持つ可逆 CA を得るには, 局所的に可逆であるような PCA でそのような性質を持つものを設計すれば十分であることがわかる. PCA はこのように可逆 CA の設計に便利な枠組みであるが, 一般に状態数は大きくなりがちである.

サイクリックタグシステム (CTAG) は, 古典的なタグシステムの変種として Cook [1] により提案された. 任意の2タグシステムに対して, それを模倣できる CTAG が存在し, 従って CTAG が計算万能性を有することが示されている [1].

**定義 2.2** サイクリックタグシステム (CTAG) は  $C = (k, \{Y, N\}, (p_0, \dots, p_{k-1}))$  によって定義される. 但し  $k$  ( $k = 1, 2, \dots$ ) はサイクルの長さ (周期),  $\{Y, N\}$  は (固定された) アルファベット, また  $(p_0, \dots, p_{k-1}) \in (\{Y, N\}^*)^k$  は書換規則の  $k$  個組である. 対  $(v, m)$  は  $C$  の時点表現 (instantaneous description, ID) と呼ぶ. 但し  $v \in \{Y, N\}^*$ ,  $m \in \{0, \dots, k-1\}$  であり,  $m$  をこの ID の位相と呼ぶ. ID の集合上の遷移の関係  $\Rightarrow$  を次のように定義する. 任意の  $(v, m), (v', m') \in \{Y, N\}^* \times \{0, \dots, k-1\}$  に対し,

$$\begin{aligned} (Yv, m) &\Rightarrow (v', m') \text{ iff } [m' = m + 1 \bmod k] \wedge [v' = vp_m], \\ (Nv, m) &\Rightarrow (v', m') \text{ iff } [m' = m + 1 \bmod k] \wedge [v' = v]. \end{aligned}$$

ID の列  $((v_0, m_0), \dots, (v_n, m_n))$  は,  $(v_0, m_0) = (v, 0)$  かつ  $(v_i, m_i) \Rightarrow (v_{i+1}, m_{i+1})$  ( $i = 0, 1, \dots, n-1$ ) が成り立つとき,  $v \in \{Y, N\}^*$  から始まる部分計算過程と呼ぶ. (なお以下では, 部分計算過程を  $(v_0, m_0) \Rightarrow (v_1, m_1) \Rightarrow \dots \Rightarrow (v_n, m_n)$  のように書き表す.)  $\square$

### 3 単純な計算万能可逆セルオートマトン

#### 3.1 任意の CTAG を無限状相で模倣できる 36 状態モデル

最初の計算万能モデルは 36 状態可逆 PCA  $P_{36}$  である。これは無限状相 (但し左方向に周期的) 上で働くモデルである。

$$P_{36} = (\mathbf{Z}, (\{\#\}, \{\#, Y, N, +, -, \cdot\}, \{\#, y, n, +, -, *\}), g_{36}, (\#, \#, \#))$$

局所写像  $g_{36}$  は図 1 に示される 33 個の局所遷移規則からなる。なお,  $P_{36}$  は左部分の状態集合が  $\{\#\}$  であるので, 事実上 2 近傍 PCA である。また, どの 2 つの異なる局所遷移規則をとってもその右边が異なっているので,  $P_{36}$  は可逆である。

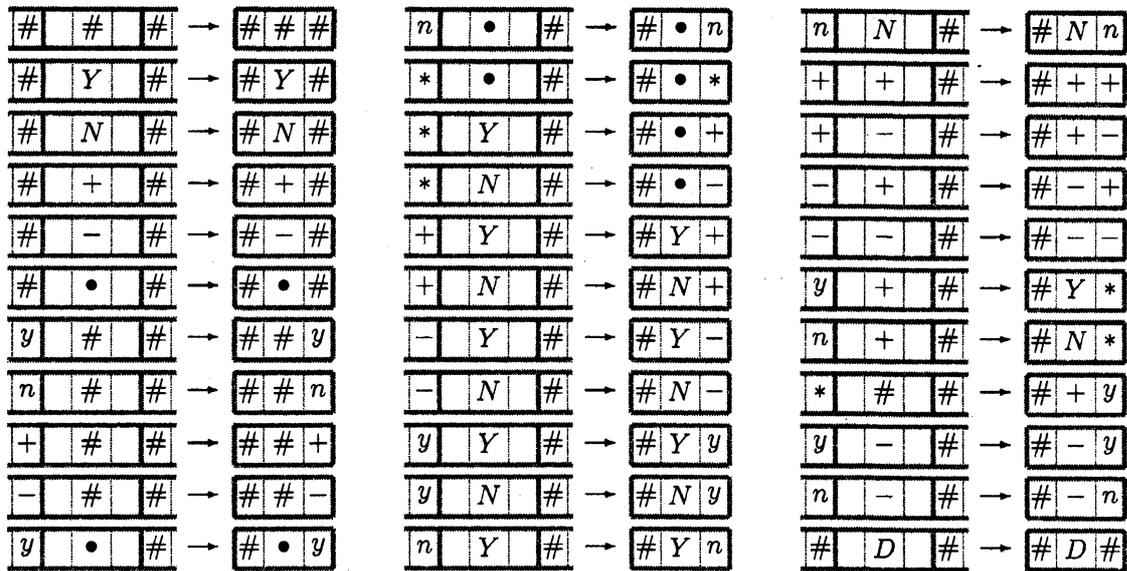


図 1: PCA  $P_{36}$  の局所遷移規則.

$P_{36}$  がどのように CTAG を模倣するかを次の CTAG の例  $C_0$  を用いて説明する。  $C_0 = (3, \{Y, N\}, (\varepsilon, NY, YY))$  (但し  $\varepsilon$  は空列を表す)。初期記号列を  $YYN$  とすると,  $(YYN, 0) \Rightarrow (YN, 1) \Rightarrow (NNY, 2) \Rightarrow (NY, 0)$  は, それから始まる部分計算過程である。

$C_0$  と初期記号列  $YYN$  に対する  $P_{36}$  の初期状態は次のように定める (図 2 の一行目を参照)。初期記号列  $YYN$  は, 連続する 3 セルの中央部分に与える。それらのすぐ右のセルの中央部分は  $-$  に, また, それらよりも左にあるセルはすべて中央部分を  $\cdot$  にセットする。書換規則は, 右部分の状態  $y, n, *$  からなる左右反転列によって表現する (なお, 右部分の状態は右に進む信号として働くことに注意)。ここで  $*$  は各規則の先頭を示す区切り記号である。従って, 1 サイクルの規則列  $(\varepsilon, NY, YY)$  は状態列  $yy*yn**$  によって表される。この状態列の間には状態  $\#$  を任意に挿入してもよい。従って例えば  $\#\#yy* \#y\#n* \#\#$  も  $(\varepsilon, NY, YY)$  を表す。CTAG では 1 サイクルの規則列  $(\varepsilon, NY, YY)$  を繰り返し適用しなければならないので, 状態の列  $yy*yn**$  の無限個のコピーを  $P_{36}$  の初期状態に与える必要がある。

•y	•y	•*	•	•y	•n	•*	•	•*	•	•*	•	Y	Y	N	-				
•	•y	•y	•*	•	•y	•n	•*	•	•*	•	•*	Y	Y	N	-				
•	•	•y	•y	•*	•	•y	•n	•*	•	•+	•	Y	N	-					
•	•	•	•y	•y	•*	•	•y	•n	•*	•	•+	Y	N	-					
•	•	•	•	•y	•y	•*	•	•y	•n	•*	•	Y	N+	-					
•	•	•	•	•	•y	•y	•*	•	•y	•n	•+	N	+	-					
•	•	•	•	•	•	•y	•y	•*	•	•y	•n	N+	+	-					
•	•	•	•	•	•	•	•y	•y	•*	•	•y	Nn	++	-					
•	•	•	•	•	•	•	•	•y	•y	•*	•	Ny	N*	+					
•	•	•	•	•	•	•	•	•	•y	•y	•*	N	Ny	+y	+				
•	•	•	•	•	•	•	•	•	•	•y	•y	•-	N	Y*	y	+			
•	•	•	•	•	•	•	•	•	•	•	•y	•y	N-	Y	+y	y			
•	•	•	•	•	•	•	•	•	•	•	•	•y	Ny	Y-	+	y			
•	•	•	•	•	•	•	•	•	•	•	•	•	Ny	Yy	-+				
•	•	•	•	•	•	•	•	•	•	•	•	•	N	Yy	-y	+			
•	•	•	•	•	•	•	•	•	•	•	•	•	N	Y	-y	y			
•	•	•	•	•	•	•	•	•	•	•	•	•	N	Y	-	y			
•	•	•	•	•	•	•	•	•	•	•	•	•	N	Y	-				

図 2: 可逆 PCA  $P_{36}$  による CTAG  $C_0$  の模倣 (状態 # は空白で表示).

CTAG における書換プロセスは次のように模倣される。信号 \* が書換えられる記号列の先頭の Y (あるいは N) に会うと、それを状態●に変え、この信号自身は + (-) になる。この信号 + (-) は右方に進んで Y と N からなる記号列を通り抜ける。この信号が右方にある中央部の状態 + または - と出会うと、後者は + (-) に置き換えられ、先頭記号が Y (N) であったことを記録する。一方、信号 y と n は + または - に出会うまで右方に進む。信号 y (あるいは n) が + に出会うと、それは Y (N) に変化し、この位置に固定される。そして + は右に 1 セル分シフトする。信号 y と n が - に出会った場合には、単に素通りして右方に無限に進み、信号が固定されることはない。以上により、書換過程が正しく模倣される。図 2 は  $C_0$  における 1 サイクルの計算過程がどのように模倣されるかを示している。

### 3.2 停止性が定義された任意の CTAG を有限状相で模倣できる 98 状態モデル

Cook [1] によって定義された CTAG (定義 2.2 参照) には「停止」の概念がなく、実際、書換えられる記号列が空列にならない限り計算が停止しない。ここでは、停止したときの記号列中に計算結果が得られるように、CTAG に停止の概念を次の方法で導入する。

**定義 3.1** 停止性が定義された CTAG は  $C = (k, \{Y, N\}, (\text{halt}, p_1, \dots, p_{k-1}))$  によって与えられる。但し  $k (k = 1, 2, \dots)$  はサイクルの長さ、 $\{Y, N\}$  はアルファベット、 $(p_1, \dots, p_{k-1}) \in (\{Y, N\}^*)^{k-1}$  は書換規則の  $k-1$  個組である。停止 ID は  $(Yv, 0) (v \in \{Y, N\}^*)$  の形の ID である。遷移の関係  $\Rightarrow$  は、 $(Yv, 0) \Rightarrow (v', m')$  となるような ID  $(v', m')$  が存在しないよ

うに、次のように定義される。任意の  $(v, m), (v', m') \in \{Y, N\}^* \times \{0, \dots, k-1\}$  に対し、

$$(Yv, m) \Rightarrow (v', m') \text{ iff } [m \neq 0] \wedge [m' = m + 1 \bmod k] \wedge [v' = vp_m],$$

$$(Nv, m) \Rightarrow (v', m') \text{ iff } [m' = m + 1 \bmod k] \wedge [v' = v].$$

ID の列  $((v_0, m_0), \dots, (v_n, m_n))$  は、 $(v_0, m_0) = (v, 0)$ ,  $(v_i, m_i) \Rightarrow (v_{i+1}, m_{i+1})$  ( $i = 0, 1, \dots, n-1$ ) かつ  $(v_n, m_n)$  が停止 ID のときに  $v \in \{Y, N\}^*$  から始まる完全な計算過程と呼ばれる。□

例 3.1 2 タグシステムの例として次の  $T_1$  を考える (2 タグシステムについては例えば [2] を参照のこと)。  $T_1 = (2, \{a_0, a_1, a_2\}, \{a_0 : \text{halt}, a_1 \rightarrow a_0 a_2, a_2 \rightarrow a_1\})$ 。  $T_1$  は、次のような停止性が定義された CTAG  $C_1$  によって模倣できる。  $C_1 = (6, \{Y, N\}, (\text{halt}, YNNNNY, NYN, \varepsilon, \varepsilon, \varepsilon))$ 。ここで、 $T_1$  における記号  $a_0, a_1, a_2$  は、 $C_1$  においては記号列  $YNN, NYN, NNY$  によって表現される。そして  $T_1$  における計算過程  $a_2 a_1 a_0 \Rightarrow a_0 a_1$  は、 $C_1$  においては次の完全な計算過程によって模倣される。  $(NNYNYNYNN, 0) \Rightarrow (NYNYNYNN, 1) \Rightarrow (YNYNYNN, 2) \Rightarrow (NYNYNNNYN, 3) \Rightarrow (YNYNNNYN, 4) \Rightarrow (NYNNNYN, 5) \Rightarrow (YNNNYN, 0)$ 。 □

停止性が定義された任意の CTAG を有限状相で模倣できる 98 状態計算万能可逆 PCA を与える。

$$P_{98} = (\mathbf{Z}, (L, C, R), g_{98}, (\#, \#, \#)).$$

ここで  $L = \{\#, e\}$ ,  $C = \{\#, Y, N, +, -, \cdot, Y_1\}$ ,  $R = \{\#, y, n, +, -, *, e\}$  である。局所関数  $g_{98}$  は図 3 に示される 48 個の局所遷移規則によって定められる。  $P_{98}$  の可逆性とは容易に確かめられる。

ここで次のような CTAG の例を考える。  $C_2 = (3, \{Y, N\}, (\text{halt}, NY, YY))$ 。このとき、  $(NYY, 0) \Rightarrow (YY, 1) \Rightarrow (YNY, 2) \Rightarrow (NYYY, 0) \Rightarrow (YYY, 1) \Rightarrow (YNYN, 2) \Rightarrow (YNYYY, 0)$  は、初期記号列  $NYY$  からの完全な計算過程である。図 4 は  $P_{98}$  におけるこの計算過程の模倣を示す。初期記号列の配置と、記号列を書換える方法は  $P_{36}$  におけるのと同様である。  $P_{36}$  との相違点は次の通りである。

まず状相を有限にするために次のようにする。1 サイクルの書換規則列の「雛型」を、図 4 にあるように、初期記号列の左方に置く (但し  $p_0 (= \varepsilon)$  をダミーの規則として与えている)。各規則は静止状態  $(\#, \#, \#)$  によって区切り、また 1 サイクルの規則列全体は左端マーカー  $(\#, +, e)$ ,  $(e, +, \#)$  と右端マーカー  $(\#, Y_1, \#)$  によって区切る。書換規則の雛型を左方に進む信号  $e$  で活性化することで、 $y, n, -$  からなる書換規則を表す信号列が生み出される (但し信号  $-$  は右端マーカー  $(\#, Y_1, \#)$  を通過すると  $*$  に変わる)。

次に  $P_{98}$  は、1 サイクルの書換えが終わる度毎に、得られた ID が停止 ID か否かを調べ、もしそうならば、  $P_{98}$  はこの ID を含む部分状相がそれ以上変化しないように保存する必要がある (但し  $P_{98}$  の可逆性から、全体の状相は安定にはなりえない)。このため、左方に進む信号  $e$  は、1 サイクルの規則列を生み出し終わると、左端マーカーの位置で反射して右方に進み、最終的に記号列の先頭に達する。このときの ID の位相は 0 であるので、もし先頭記号が  $N$  ならば信号  $e$  は進行方向を左方に変え、次の 1 サイクルの規則列を生み出す。先頭記号が  $Y$  なら停止 ID であるので、信号  $e$  は  $+$  に変化して右方に送られる。これにより停止 ID は  $P_{98}$  の状相の中で固定される。

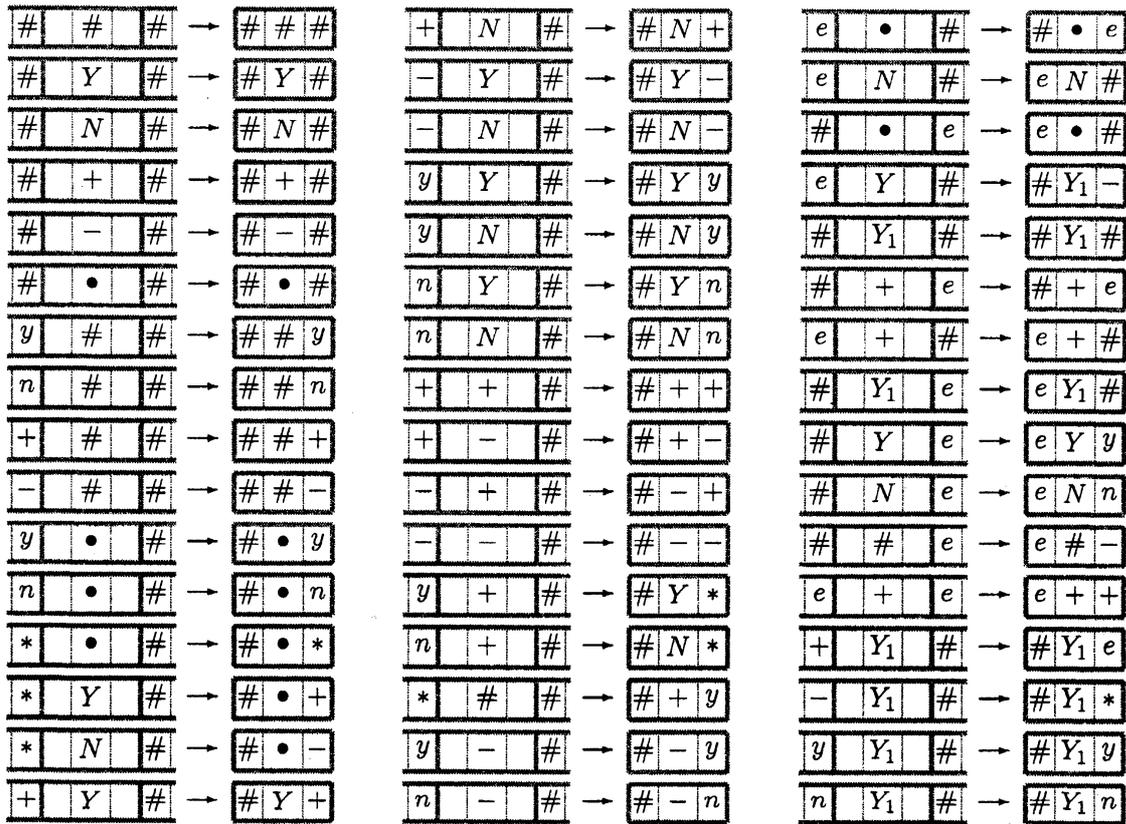


図 3: 可逆 PCA  $P_{98}$  の局所遷移規則.

## 4 むすび

本論文では計算万能で単純な 1 次元可逆 PCA のモデルを 2 つ示した。計算万能な可逆 CA の状態数を大幅に減少させるには PCA ではなく通常の CA の枠組みを用いる必要があると考えられるが、これは今後の課題として残される。

謝辞 本研究は一部、科学研究費補助金・基盤研究 (C)(2) 16500012 による。

## 参考文献

- [1] Cook, M., Universality in elementary cellular automata, *Complex Systems*, **15**, 1–40 (2004).
- [2] Minsky, M.L., *Computation: Finite and Infinite Machines*, Prentice-Hall (1967).
- [3] Morita, K., and Harao, M., Computation universality of one-dimensional reversible (injective) cellular automata, *Trans. IEICE Japan*, **E72**, 758–762 (1989).

