

非同期セル空間における論理回路構成 Construction of Logical Circuits in Asynchronous Cellular Space

広島大学大学院工学研究科 齊金山・森田 憲一

Graduate School of Engineering, Hiroshima University Jin-Shan Qi, and Kenichi Morita

概要

非同期セルオートマトンは各セルが任意の時間遅延をとって動作するセルオートマトンである。これは全セルの完全な同期を必要としないため実装に適していると考えられる。本稿では、任意の組合せ論理回路が実現できるような非常に単純な非同期セルオートマトンを提案する。これは2状態9近傍のセルオートマトンであるが、通常のものとは異なり、「ブロック写像」、つまり原像と像が同一の幾何学的形状であるような局所写像を持っている。これにより状態数と局所写像の規則数を少なくすることができた。このセル空間に、信号とそれを伝えるための各種ワイヤ、信号の分岐と消滅のための素子、および2つの信号の相互作用のための素子を実現する。相互作用素子は、2信号が対向方向から来た場合には共に直進させ、直角方向から来た場合には1つを直進させ、もう1つを左折させるというごく簡単な規則に基づく動作をする。これらの素子をこのセル空間に適切に配置することにより任意の論理関数が実現できることを示す。

キーワード：非同期セルオートマトン, 論理回路

Keywords: asynchronous cellular automaton, logic circuit

1 まえがき

セルオートマトン (cellular automaton, CA) は同一の有限オートマトンを一様に配置、接続した並列システムである。各セルは自分自身と近傍セルの状態に依存して次の時刻の状態を決定する。通常のセルオートマトンは、すべてのセルが同時に状態遷移を行うものとして定式化されている。従って、全体の同期をとるためにはクロックが必要となるが、セルが非常に多数になった場合、全体の同期をとるのは困難となる。そのため近年、非同期セルオートマトン (asynchronous cellular automaton, ACA) の研究が多くみられるようになった [1, 2, 4, 5]。例えば、文献 [4] では、16 状態 5 近傍セルオートマトンで単純な遷移規則を持つものが提案され、任意の非同期回路を構成するための 5 種類の機能モジュールがそのセル空間に実現できることが示されている。

本稿では、2 状態 9 近傍非同期セルオートマトンを新しく提案する。これは通常のセルオートマトンとは異なり「ブロック写像」と呼ばれる形式の局所写像 (局所遷移関数) を持つセルオートマトンである。つまり、原像と像が同一の幾何学的形状であるような局所写像である。また状相遷移の各ステップでは、そのような局所写像がセル空間のただ 1 箇所 (非同期的に) 適用される。従ってこれは、等形アレイ文法 (Isometric Array Grammar, IAG) [6] と同等のシステムであると考えられる (なお文献 [4] のモデルは、1 つのセルがそれぞれ 2 状態からなる 4 部分に分割されているようなセルオートマトンであり、合計 8 個の 2 状態部分の状態から、次の時刻の 8 個の 2 状態部分の状態が決まる)。このような枠組みを用いることで、状態数が 2、局所写像の規則スキーム数が 7 であるような単純な非同期セルオートマトンが得られる。

ここではまず、このセル空間に信号とそれを伝えるための各種ワイヤ、信号の分岐と消滅のための素子、および 2 つの信号の相互作用のための素子を実現できることを示す。この中で特に重要なものは相互作用素子である。これは、2 つの信号が対向方向から来た場合には共に直進させ、直角方向から来た場合には 1 つを直進、もう 1 つを左折させるという動作をする。但しこの素子は、信号が 1 つしか来て

いない場合には、2つ目が来るまで待つ。論理値“0”と“1”を表すのにいわゆる「二線式」[3]の表現法を用いるとAND演算が相互作用素子1つで実現できる。これらの組み合わせにより、任意の組合せ論理回路が構成可能となる。

2 2状態9近傍非同期ブロックセルオートマトン

本稿の非同期セルオートマトンは次のように定義する。

定義 2.1 2次元9近傍非同期ブロックセルオートマトンとは5項組

$$A = (\mathbf{Z}^2, N, Q, f, q_0)$$

によって定義される。但し、 \mathbf{Z} は整数集合、従って \mathbf{Z}^2 は2次元セル空間を表す。 $N = ((0,0), (0,1), (0,2), (1,0), (2,0), (0,-1), (0,-2), (-1,0), (-2,0))$ は近傍型と呼ばれる。任意のセル $x \in \mathbf{Z}^2$ に対し、 $x_0 = x, x_1 = x + (0,1), x_2 = x + (0,2), x_3 = x + (1,0), x_4 = x + (2,0), x_5 = x + (0,-1), x_6 = x + (0,-2), x_7 = x + (-1,0), x_8 = x + (-2,0)$ と置くと、 $\{x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ が x の近傍となる。 Q は状態の有限かつ空でない集合である。 f は $f: Q^9 \rightarrow Q^9$ なる写像で局所写像と呼ばれる(ブロック写像ともいう)。 $q_0 (\in Q)$ は静止状態であり、 $f(q_0, q_0, q_0, q_0, q_0, q_0, q_0, q_0, q_0) = (q_0, q_0, q_0, q_0, q_0, q_0, q_0, q_0, q_0)$ を満たす。 $(a, b, c, d, e, f, g, h, i), (a', b', c', d', e', f', g', h', i') \in Q^9$ に対して

$$f(a, b, c, d, e, f, g, h, i) = (a', b', c', d', e', f', g', h', i')$$

が成り立つときこれを

$$(a, b, c, d, e, f, g, h, i) \rightarrow (a', b', c', d', e', f', g', h', i')$$

のように、あるいは図1のように表し、 A の遷移規則と呼ぶ。

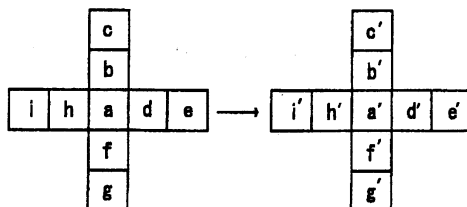


図 1: 9近傍非同期ブロックセルオートマトンの遷移規則

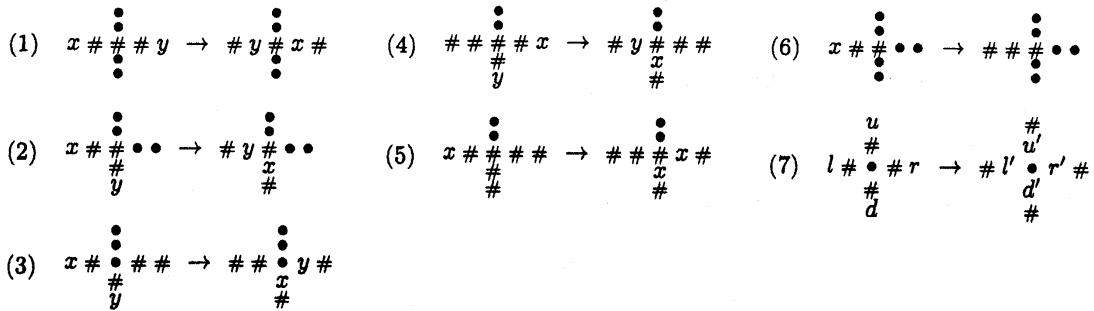
$c: \mathbf{Z}^2 \rightarrow Q$ を Q 上の(または A の)状態と呼ぶ。 Q 上の状態すべての集合を $\text{Conf}(Q)$ と書く。つまり $\text{Conf}(Q) = \{c \mid c: \mathbf{Z}^2 \rightarrow Q\}$ 。 A の大域写像 $F: \text{Conf}(Q) \rightarrow 2^{\text{Conf}(Q)}$ は任意の $c, c' \in \text{Conf}(Q)$ に対して次のように定義される。

$$\begin{aligned} c' \in F(c) \text{ iff} \\ \exists x \in \mathbf{Z}^2 \quad & ((c'(x_0), c'(x_1), c'(x_2), c'(x_3), c'(x_4), c'(x_5), c'(x_6), c'(x_7), c'(x_8)) \\ & = f(c(x_0), c(x_1), c(x_2), c(x_3), c(x_4), c(x_5), c(x_6), c(x_7), c(x_8)) \wedge \\ & \forall y \in \mathbf{Z}^2 (y \notin \{x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\} \rightarrow c'(y) = c(y))) \end{aligned}$$

本稿では特に次のような2状態9近傍非同期ブロックセルオートマトン A_2 を考える。

$$A_2 = (\mathbf{Z}^2, N, \{0, 1\}, f_2, 0)$$

遷移規則 f_2 は次の通りである。ここで、状態 0 と 1 はそれぞれ空白 (#) と \bullet で表している。また、回転対称性を仮定しているため、左辺と右辺を同時に 90, 180, 270 度回転して得られる遷移規則は省略してある。



但し、 $x, y \in \{0, 1\}$ である。規則 (7) は、 $u, r, d, l \in \{0, 1\}$ かつ、 u, r, d, l のうちちょうど 2 つが 1 の場合を表しており、このとき u', r', d', l' は次のようになる。

$$u' = ud + rd + ul, \quad r' = rl + dl + ru, \quad d' = du + lu + dr, \quad l' = lr + ur + ld.$$

上記 (1)-(7) によって指定される場合以外は両辺が同じ規則となる。(1)-(7) の 7 個の遷移規則図式は次の働きを持つ。規則 (1) により信号 x と y は対向方向に直進し、伝搬を実現する。規則 (2) により信号 x と y はそれぞれ右折と左折する。規則 (3) により信号 x と y は両方右折する。規則 (4) により信号 x と y は両方左折する。規則 (5) は信号を分岐させるときに使われる。規則 (6) は信号を消去するときに使われる。規則 (7) は信号の相互作用を実現し、相互作用素子に使われる。

3 非同期セル空間上の論理回路

3.1 信号とその伝播

信号は 1 つの \bullet によって表現する。これはあらかじめ用意された信号通路の中を通る。進行方向は信号通路の形状と信号の置かれる位置によって決まる。

信号の伝搬は図 2 に示すようになる。1 つの信号路を 2 つの信号 $x, y \in \{\bullet, \#\}$ が双方向にお互いに影響することなく伝搬する。

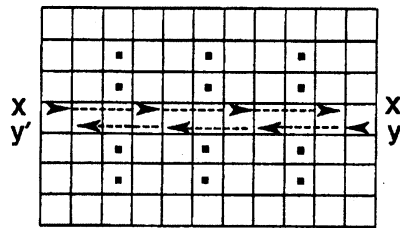


図 2: 信号の伝搬

3.2 信号の右折と左折

図 3 と図 4 のように信号の右折と左折が実現できる。

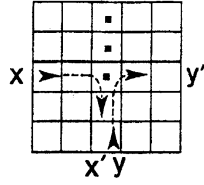


図 3: 信号の右折

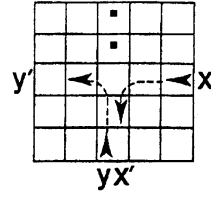


図 4: 信号の左折

3.3 信号の交差

信号 $x, y \in \{\bullet, \#\}$ の交差は図 5 に示したようである。

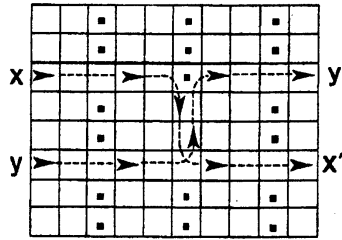


図 5: 信号の交差

3.4 信号の消去

消去素子は信号を消す働きを持つ。図 6 に示すように正面から来た信号を消去し、何の出力もしない。

3.5 信号の分岐

分岐素子は入力信号を分岐させ、同じ信号を 2 つ出力する。図 7 に示すように 1 入力 2 出力であり、信号を 2 つに分岐させ前と左方向に 1 個ずつ出力する。

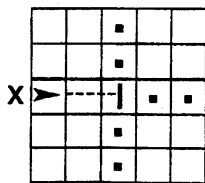


図 6: 消去素子とその動作

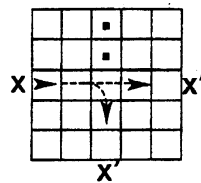


図 7: 分岐素子とその動作

3.6 信号の相互作用

相互作用素子は 4 入力 4 出力の素子 (図 8(a)) である。4 本の入力線にちょうど 2 つの信号が揃ったとき動作する。信号が 1 つしかない場合にはもう 1 つの信号が到着するまで待つことになる。

- ・ 2 つの信号が対向方向 (つまり平行) に来た場合には、2 つとも直進する (図 8(b)).
- ・ 2 つの信号が直角方向に来た場合には、左方からの信号が直進、右方からののが左折する (図 8(c)).

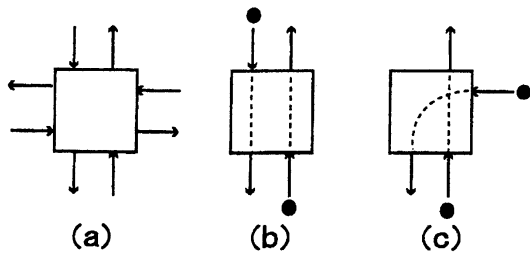


図 8: 相互作用素子の概念図

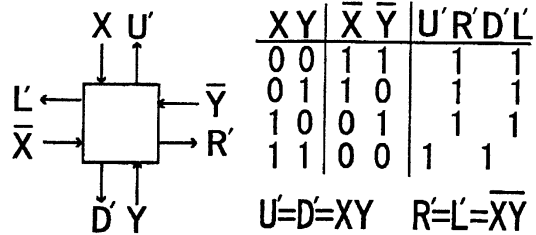


図 9: 相互作用素子の真理表

図 8 は相互作用素子の動作を表している。相互作用素子の入出力を図 9 の左部分のように接続する。右の真理表から分かるように、 $U' = D' = XY, R' = L' = \overline{XY}$ 。出力 U' と R' を信号 XY の出力に対応させると、論理演算 AND が実現できる。

3.7 NOT と AND の実現

3.7.1 NOT

図 10 で示したように入力信号を入れ替えることにより論理 NOT が実現できる。

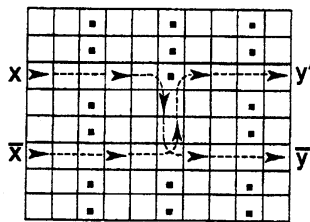


図 10: NOT の実現

3.7.2 AND

相互作用素子を基本にし、信号の消去、左折と右折を利用しながら論理演算 AND が実現できる。相互作用素子の真理表より、図 11 が AND 演算を行ってることを確認できる。

4 むすび

本稿では任意の組合せ論理回路がその空間に構成できるような、非同期ブロックセルオートマトンを提案した。ここでは近傍を 9 近傍とし、また、局所写像を原像と像が同じ形状であるような写像にすることにより、2 状態で実現した。このとき、非常に単純な動作を行う相互作用素子が回路構成の基礎となっている。このようなセル空間への順序回路の簡潔な埋め込みが今後の課題である。

参考文献

[1] 中村克彦, 非同期セルオートマトンとその計算能力, 電子通信学会論文誌, 57-D, No.10, 573-580, 1974.

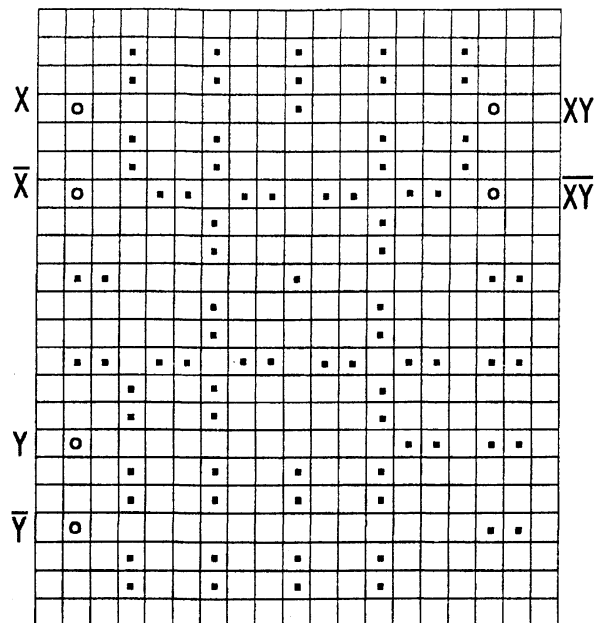


図 11: AND の実現 (○で信号の入出力の場所を示している)

- [2] J. Lee, S. Adachi, F. Peper, and K. Morita, Embedding universal delay-insensitive circuits in asynchronous cellular spaces, *IEEE Trans. Computers* (to appear).
- [3] J. von Neumann, Probabilistic logic and the synthesis of reliable organisms from unreliable components, *Automata Studies* (eds. C.E. Shannon and J. McCarthy), Princeton University Press, 1956.
- [4] F. Peper, J. Lee, S. Adachi, and S. Mashiko, Laying out circuits on asynchronous cellular arrays: a step towards feasible nanocomputers, *Nanotechnology*, **14**, 469–485, 2003.
- [5] H. K. Büning, and L. Priese, Universal asynchronous iterative arrays of Mealy automata, *Acta Informatica*, **13**, 269–285, 1980.
- [6] A. Rosenfeld, *Picture Languages*, Academic Press, New York, 1979.