

セル・オートマタによる符号化手法とその分析

福原 義久[†] 武藤 佳恭^{††}

セル・オートマタ（以下 CA）とは有限オートマトンを空間内に並べて配置し、近接するセルどうしを結合したシステムである。CA は並列・同期的に動作し、空間内の状態はオートマトンの動作規則によってさまざまに変化する。通常、CA は 1 つの系で状態変化が完結しているが、本研究では複数の CA システムを同期して動作させることにより、各 CA 空間の情報量を下げ、符号化の手法として用いることができることを示した。また、どのような遷移規則の場合に効果的に符号化が行えるかを遷移規則の持つパターン状態を表すパラメータを定義して、詳細に分析した。

Information Encoding by Using Cellular Automata

YOSHIHISA FUKUHARA[†] and YOSHIYASU TAKEFUJI^{††}

In this paper, we propose a new information encoding method by using cellular automata (CA). In this approach, plural CA systems work simultaneously and input values are encoded into small bits by simple deterministic process. The proposed system cannot work under all situation depend on their transition rules and input values. We studied when the proposed system works correctly from various kinds of views.

1. はじめに

セル・オートマタ（以下、CA）¹⁾は生命現象、化学現象、社会現象などの振舞いを考えるうえでの最もシンプルかつ汎用的なモデルである。現実世界のアナロジーとして CA を用いてさまざまな種類の研究が行われている一方で、CA 自体の性質や振舞いについての研究もさかに行われてきた。

CA を情報処理プロセッサとしてとらえたとき、ライフゲームとして知られる二次元の CA が万能チューリングマシンとして動作することが Berlekamp ら²⁾によって示された。その後、Wolfram^{3),4)}は一次元 CA の動作を詳細に検証した結果、秩序的な状態からカオス的な状態に至る 4 つの状態（クラス）に分けられることを示した。秩序状態とカオス状態のはざまに位置する状態はカオスの縁と呼ばれ、Wolfram のクラス分けではクラス IV に相当する。彼はこのような複雑な状態を示す CA には計算万能性があるのではないかと考えた。カオスの縁における計算能力と相転移の相関については Langton⁵⁾の研究が知られているが、これ

らの問題については Mitchell ら¹⁰⁾によって検証が行われた結果、そのような効果は疑問視されている。

このように、シミュレータとしての用途や数理的分析の研究がさかに行われている一方で、CA を用いて具体的な問題解決を行わせる例はそれほど多く見当たらない。これは CA がその単純な動作原理とはうらはらに予測困難で複雑な振舞いを示すことに起因しているのだろう。情報処理システムとしてセルニューラルネットワーク（CNN）^{7),8)}などの応用例も見られるが、これはむしろ人工ニューラルネットワークのアーキテクチャをより単純かつ高速に運用するためのもので、CA の複雑なダイナミクスの本質に迫るものではない。また、Mitchell らによる進化型 CA (EvCA)^{9),10)}は、遷移規則を遺伝的アルゴリズムを用いて最適化する試みである。このような手法を用いれば問題に対して最適な遷移規則を獲得することが可能である。しかし実際のところ CA で解決できる具体的問題は現在のところ比較的単純なものに限られている。CA の示す複雑な振舞いは、計算万能性や創発現象、生命現象との関係⁵⁾が述べられるほどの可能性を持ちつつも、具体的に利用することが困難なのである。

もう 1 つの問題はオートマトンに目的に応じてさまざまに機能を付け加え、拡張したシステムを構築することは可能であるが、そのことにより逆に系の動作原

[†] 慶應義塾大学 SFC 研究所武藤佳恭研究室

Takefuji Laboratory, Keio Research Institute at SFC

^{††} 慶應義塾大学環境情報学部

Faculty of Environmental Information, Keio University

理の複雑化を招き、個々の要素の役割や影響が見えにくくなってしまふことである。我々は、なるべく CA の基本的枠組みに多くの手を加えることなくその情報処理の可能性を検証したい。

そこで本研究では CA を用いた符号化装置を考案した。提案システムは、複数の CA を同時に実行し、各時間でそれぞれの状態を比較しながら情報の次元数を低下させていくことによってなされる。提案システムは、セルの状態として値のほかにそのセルが使用可能か不可能かのフラグを追加することにより、符号化という最も基本的な情報処理を行うことに成功したものである。

また、提案システムがどのような遷移規則のときに効果的に動作するのかを遷移規則のパターンに対する応答特性を示す新たなパラメータを定義し分析を行った。

2. 手 法

本章では、一次元 CA の原理について述べた後、符号化の手法について述べる。

2.1 一次元セル・オートマトン

CA は離散的時空間の中で次の時刻でのあるセルの状態が、そのセルと近傍のセルの状態によって決定される。シミュレータとして用いる場合には二次元の CA がよく扱われるが、CA そのものの動作振舞いを検証するために一次元のものが用いられることも多い。

最も基本的な 2 状態 3 近傍の一次元 CA の場合、自セルと両隣の 3 つのセルの状態から次の時刻での自セルの状態が決定される。2 状態 3 近傍の場合、図 1 (1) に示されたような 8 つのエントリの組合せからなる遷移規則が必要である。遷移規則の組合せの数は k を状態数、 n を自セルを含めた近傍の数とすると k^n 個存在する。2 状態 3 近傍の CA の場合、存在する遷移規則の組合せは $2^{2^3} = 256$ 個あることとなる。

一般に遷移規則を f で表すとすると、セルの状態 a は式 (1)、(2) で表される。ここで $a_{i,j}^t$ は時刻 t における系 i の j 番目のセルの状態をしめしている。

$$a_{i,j}^{t+1} = f(a_{i,j-x}^t, \dots, a_{i,j}^t, \dots, a_{i,j+x}^t) \quad (1)$$

$$x = \frac{n-1}{2} \quad (2)$$

一次元の任意の初期パターンにこの遷移規則を適用することによって、パターンはさまざまに変化していく。本研究では基本的に一次元の CA を取り扱うので、以降 CA と記述した場合は一次元のものを指すこととする。

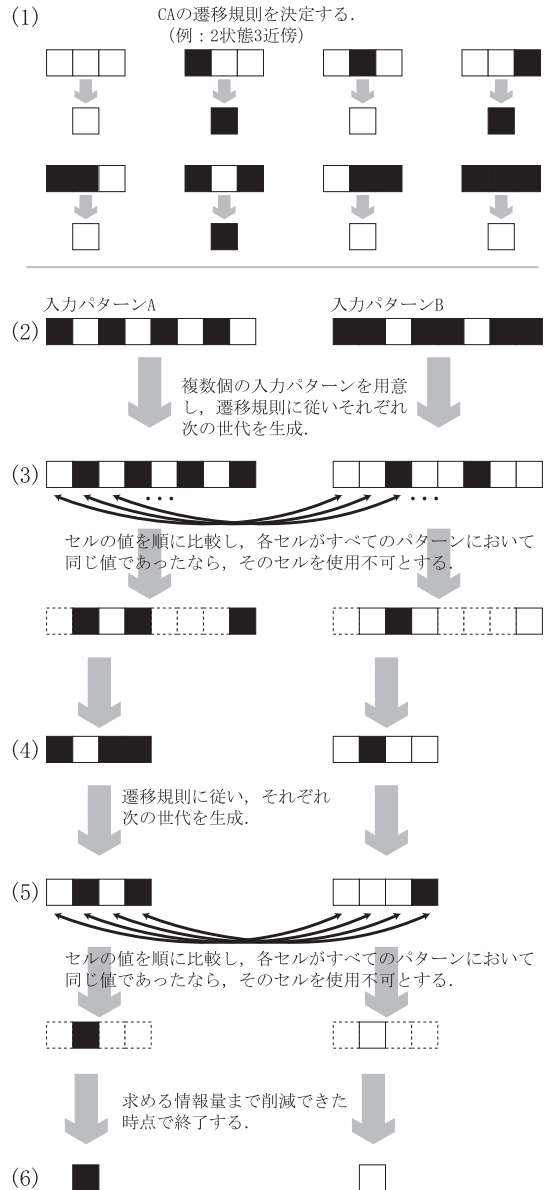


図 1 CA を用いた符号化の例

Fig. 1 An example of the information encoder by using CA.

2.2 CA を用いた符号化装置

本研究での符号化とは、複数の異なる有限の数列があるアルゴリズムに従って、より低次元のそれぞれ異なる数列に写像することを意味する。

まず符号化を行いたい複数のパターンに対してそれぞれ遷移規則を適用する。提案システムの動作は、これら各 CA 系からの出力を各時間でそれぞれ比較し、不要なセルを消滅することによって情報の次元を下げていくことによってなされる。つまりプロセスの進行

により空間からセルを取り除いてやる必要があるのだが、これはいいかえれば、セルのとり値のほかはそのセルが使用可能かどうかのフラグを追加することと同じである。

すべての系を通して、各セルが使用可能かどうかは式 (3) によって導かれる。ここで S_j^t は m 個の CA 系が存在するときの t 時における j 番目が使用可能かどうかのフラグを示す。 S が真の a の集合に対してのみ式 (1) が適用される。本研究では関数 f を a がすべて同じ値のとき使用不可能、それ以外のとき使用可能な状態を返すと定義した (式 (4))。座標 j のセルの生き死にはすべての CA 系において共通である。つまりある系でのセル j は使用可能にもかかわらず、別の系でのセル j が使用不可能であるということはある。また、それぞれの CA 系は同じ遷移規則で動作するものとする。

$$S_j^t = f(a_{1,j}^t, a_{2,j}^t, \dots, a_{i,j}^t) \quad (3)$$

$$f(x) = \begin{cases} 0 & a_{1,j}^t = a_{2,j}^t \dots = a_{i,j}^t \text{ のとき} \\ 1 & \text{それ以外のとき} \end{cases} \quad (4)$$

上記の符号化プロセスについて具体例を用いて説明する。図 1 の模式図の例では、入力パターンとして 2 状態 8 ビットのパターンを 2 つ用意した。これらを 2 つに分類する作業なので、最低 1 ビットの情報量が必要である。よって目標とする次元数は 1 とする。

- (1) CA の遷移規則を適当に決める。図の例では 2 状態 3 近傍の遷移規則を決定した。
- (2) 長さが同じ複数の入力パターンを用意し、それぞれから遷移規則に従って次世代のセルを生成する。境界条件は巡回境界条件とする。
- (3) 生成されたパターンのセルを同じ座標ごとにそれぞれ比較し、すべて同じ値であった場合にはそのセルを使用不可とする。例では、1, 5, 6, 7 番目のセルが 2 つのパターンで同じ値であるため使用不可能とする。
- (4) 残ったセルに再度遷移規則を適用する。ただしここで用いる近傍のセルとは**使用可能なセルのうち最も近くにあるもの**を意味する。図中では説明のため、残ったセルでパターンを再度構成した。
- (5) (3) から (4) を期待する次元にセルの数が減るまで繰り返す。
- (6) 期待する次元数まで情報量が下がった時点で終了する。

しかし、以上の操作は学習などをとまわず、あくまで遷移規則だけに基づいた決定論的操作なので必ず

しも望む結果が得られるとは限らない。入力パターンを図の例のように正しく符号化できるか否かは入力パターンと遷移規則との複雑な関係により事前に予測することは困難である。最終的にシステムのとりうる状態は以下の 4 つが考えられる。

- 期待する次元数までセルの数が減り符号化が成功する。
- 期待する次元数までセルの数が減るが、正しく符号化できていない。
- 期待する次元数よりもセルの数が減ってしまう。もしくはセルがすべて使用不可になる。
- 期待する次元数に達する前にリミットサイクルに陥り、それ以上セルの数が減らなくなる。

提案手法では最終的な出力が望まれるものであった場合、その遷移規則を用いることにより適用したパターンのセットは決定論的操作により必ず符号化することができる。また、各時刻での使用可能なセルの座標を記録しておけば、パターン間の比較・消去の作業が不要になるので単純かつ高速なパターン分類器として用いることもできる。また提案手法は二次元以上の CA にも適用可能である。

しかし実用的なアプリケーションとして提案手法にアプローチする前に、まずどのような状態の CA が符号化能力を持ちうるのかについて検証する必要があるだろう。次章以降では、さまざまな遷移規則と入力パターンを用いて提案手法を検証し、遷移規則と符号化の能力の関係について考察する。

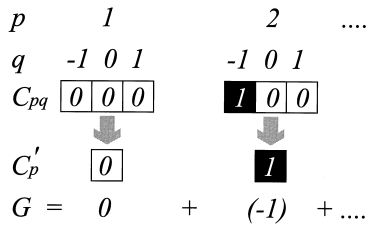
3. 評価手法

我々は前述のようなシステムがどのような遷移規則の場合効率良く動作するかを知りたい。ここではまず Langton⁵⁾の提案した λ パラメータ (式 (5)) を用いる。 λ パラメータを用いることにより遷移規則の入力に対する応答の割合を定めることができる。

$$\lambda = \frac{k^n - n_q}{k^n} \quad (5)$$

k は状態数であり、 n は状態遷移にかかわる近傍セルの数である。 n_q は静状態を示す。静状態とは、セルがとりうるいくつかの状態のうち任意に定められたある 1 つの状態のことである。

λ パラメータは、すべての遷移規則のうちから次の世代で静状態以外の状態に遷移する規則の割合を示したものである。つまり端的に言えば、入力に対する応答の割合を示したものである。しかし λ の解像度は k^n に等しく、CA の持つ全遷移規則数 k^{k^n} に対して粗いといわざるをえない。そこで本論文では遷移規則

図2 G の計算Fig. 2 Calculation of parameter G .

のパターンに対する応答の振舞いを検証するために新たにパラメータ G を定義した。 G は式 (6), (7) および式 (2) によって定義される。

$$G = \sum_{p=1}^{k^n} \sum_{q=-x}^x q C_{pq} C'_p \quad (6)$$

$$C = \begin{cases} 0 & \text{セルが静状態のとき} \\ 1 & \text{セルが静状態以外のとき} \end{cases} \quad (7)$$

ここで、 p は遷移規則 1 組の中でのエントリをナンバリングしたものであり、 q は各エントリにおいての自セルを原点とした各セルの座標である。 C_{pq} は p 番目のエントリでの各セルの状態を示し、 C'_p は p 番目のエントリから生成されるセルの状態を示す。また、近傍数 n は奇数であることを前提とする。たとえば図 1 (1) の遷移規則の場合、白を静状態とくと $G = 0 + (-1) + 0 + 1 + 0 + 0 + 0 + 0 = 0$ となる。図 2 に計算例の一部分を示した。

G の値は、CA の状態・近傍数に応じて固有の負から正の値を持ち、0 のとき CA がエントリのパターンに対して対称的な応答をすることを示す。 G はたいていうならば遷移規則のパターンに対する応答の重心といえるだろう。これら λ と G の 2 つの観点から評価することにより、CA の遷移規則と符号化能力の関係についてより詳細に分析することができる。

また、CA に関する研究では、計算量を減らすために近傍の状態の総和だけに注目した合計型推移規則を用いることがあるが、本研究ではエントリのパターンと問題解決能力の関係をより詳細に分析するために考えうるすべての遷移規則を対象とした配列型推移規則を用いた。

4. 実験結果

2 状態 5 近傍の CA を用いて、遷移規則と符号化能力についての検証を行った。実験は 1 つの遷移規則につき、ランダムに生成された 100 ビットの入力パターンのセットを 1,000 回与え、符号化成功回数を測定した。遷移規則は各 λ の値につき 200 個をランダムに

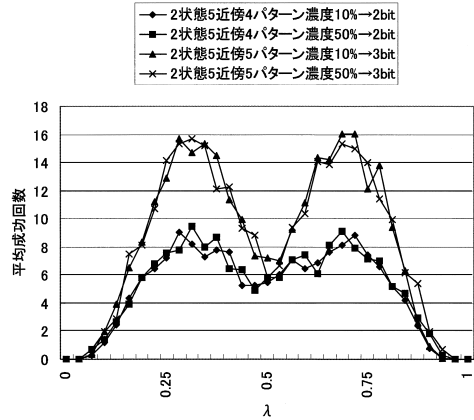
図3 λ からみた 2 状態 5 近傍 CA による符号化能力

Fig. 3 Encode ability of CA (2 states 5 neighbors).

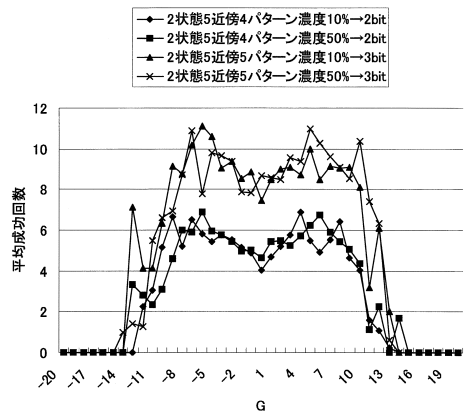
図4 G からみた 2 状態 5 近傍 CA による符号化能力

Fig. 4 Encode ability of CA (2 states 5 neighbors).

抽出したものをを用いた。CA の境界条件は巡回境界条件とし、最大 100 世代までの計算を行った。

図 3 は横軸に λ をとり縦軸にそのときの平均符号化成功回数をとった結果である。実験は 4 つのパターンを 2 ビットに符号化した場合と、5 パターンを 3 ビットに符号化した場合の結果である。また、それぞれの実験は初期入力パターンに 50% の確率で静状態を混合させた場合と 10% の確率で混合させた場合について行った。同様に図 4 は横軸に G をとり縦軸に平均符号化成功回数をとった結果である。

5 パターンを符号化した場合について、上記の結果を二次元の等高線グラフで表したものが図 5 である。

図 6 は図 5 を真上から観察したものである。

図 7, 図 8 は、各 λ , G において最も符号化が成功した例についてその値をプロットしたものである。

2 状態 3 近傍, 3 状態 3 近傍の CA についても同様の実験を行った。図 9, 10, 11, 12 に結果を示す。

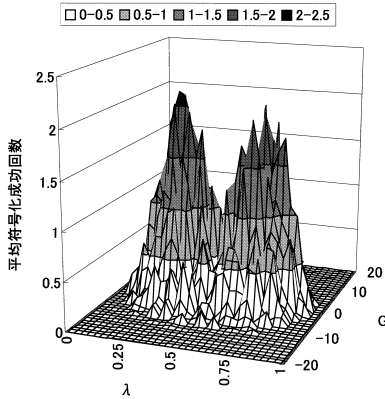


図5 2状態5近傍CAによる符号化能力

Fig. 5 Encode ability of CA (2 states 5 neighbors).

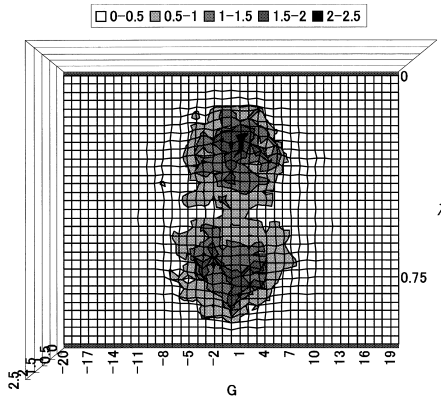


図6 2状態5近傍CAによる符号化能力

Fig. 6 Encode ability of CA (2 states 5 neighbors).

5. 考 察

5.1 実験条件と符号化能力の関係

図3, 図4に示された実験結果から, 2状態5近傍のCAでは入力パターン数, 入力パターンの状態(濃度), 符号化ビット長, 状態近傍数にかかわらず全体的に $\lambda = 0.5$ で平均符号化成功率が落ち込む特徴が観察される. また G を軸に見たとき, 総計では $G = 0$ 付近で符号化成功率が低下しているが, 図5, 図6に示した等高線グラフから判断すると $G = 0$, $\lambda = 0.3$ 付近でも高い値を示している. つまりグラフの中央付近, すなわち $G = 0$, $\lambda = 0.5$ における平均成功率の低下が著しいために, 図4の G を軸とした集計に影響を及ぼしていると考えられる.

平均符号化能力と λ , G との関係は, 図5, 図6の結果よりおおまかに把握できたが, 図7, 図8に示した符号化成功回数の最大値と比較すると, 平均値との間に大きな隔たりがあることが分かる. これはごく一

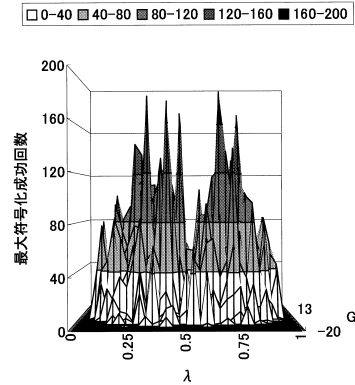


図7 2状態5近傍CAによる符号化能力

Fig. 7 Encode ability of CA (2 states 5 neighbors).

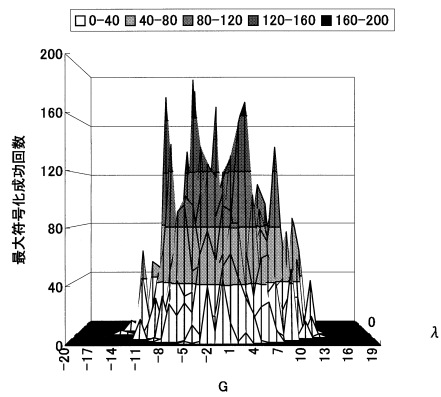


図8 2状態5近傍CAによる符号化能力

Fig. 8 Encode ability of CA (2 states 5 neighbors).

部の成功した例が平均を押し上げていることを意味する. 最大値に注目した場合, $\lambda = 0.3$, $G = |3 \sim 4|$ 付近で符号化能力が著しく高い例があることが分かる. ただし前述の理由から, これらのパラメータを持つ遷移規則のうちごく一部のものだけが低い符号化能力を持つことがうかがえる.

2状態3近傍, 3状態3近傍のCAを用いた実験でもおおむね同じような傾向が見られる. ただし2状態3近傍4パターンを2ビットに符号化する実験のみ $\lambda = 0.5$ で符号化能力が高いことが確認された. すべての実験における最適な λ および G について表1にまとめた.

5.2 応答確率の不等性と符号化能力

λ パラメータは, すべての遷移規則のうちから次の世代で静状態以外の状態に遷移する規則の割合を示したものである. つまり端的にいえば, 入力に対する応答の割合を示したものである. この値が0に近いとき静的な状態, $1.0 - 1/k$ に近いとき, 系はカオス的

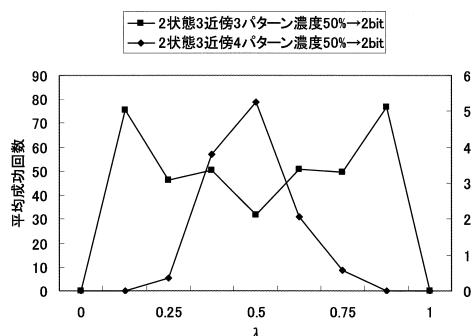


図9 λ からみた 2 状態 3 近傍 CA による符号化能力
Fig. 9 Encode ability of CA (2 states 3 neighbors).

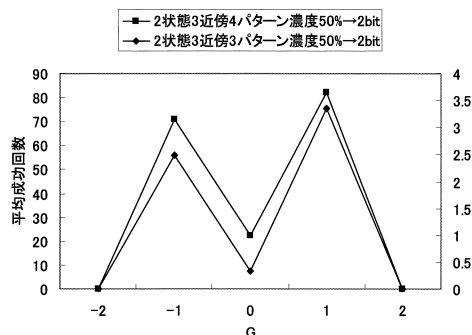


図10 G からみた 2 状態 3 近傍 CA による符号化能力
Fig. 10 Encode ability of CA (2 states 3 neighbors).

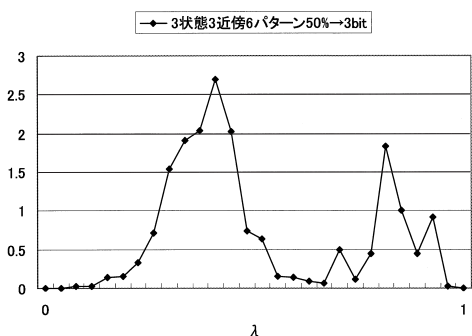


図11 λ からみた 3 状態 3 近傍 CA による符号化能力
Fig. 11 Encode ability of CA (3 states 3 neighbors).

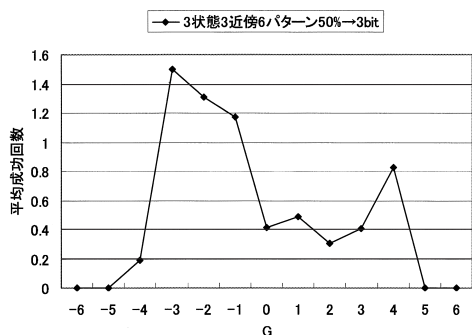


図12 G からみた 3 状態 3 近傍 CA による符号化能力
Fig. 12 Encode ability of CA (3 states 3 neighbors).

表1 平均符号化能力の高い遷移規則が含まれる変数値
(ただし $1 > G > -1$ で正規化)

Table 1 Effective parameters for the encoder.

実験	λ	G
2 状態 5 近傍 4 パターン → 2 bit	0.34 (0.66)	[0.3]
2 状態 5 近傍 5 パターン → 3 bit	0.34 (0.66)	[0.3]
2 状態 3 近傍 3 パターン → 2 bit	0.125 (0.875)	[0.5]
2 状態 3 近傍 4 パターン → 2 bit	0.5	[0.5]
3 状態 3 近傍 5 パターン → 2 bit	0.4	-0.5

態⁴⁾を示すといわれている。

提案手法を用いた実験では一部の例を除き、 $\lambda = 1.0 - 1/k$ で表されるカオス的な状態のとき、符号化能力が下がる傾向が見られる。これは状態が偏りなくすべての状態に均等に遷移するため情報量が下がりにくいためと推測される。また、カオス的な状態は周囲のセルに対する影響作用が強く、状態が混ぜ合わされる力が働くことも阻害要因と推測される。逆に λ が 0 や 1 に近い静的状態では急激に情報量が減衰してしまい、正しい符号化が行えなくなるものと考えられる。一方、 λ が 0.3 から 0.4 の間で符号化能力が高まる傾向が見られるが、これは提案手法が遷移した状態の中から差異の見られない部分を削除していくことによってなされるため、各状態が等確率で出力されないような応答が効率的に情報量を減らせるものと考えられる。ただし 2 状態 3 近傍の実験のような例外も見られることから、この視点からの考察についてはさらに実験を重ねる必要があるだろう。

$\lambda = 0.3$ 付近の CA はクラス IV と呼ばれる非常に複雑な振舞いを示すことがあるといわれているが⁵⁾、提案システムの成功例ははたしてどのような性質を持っているのだろうか。図 13 は、各実験で用いた CA の中で最も符号化能力が高かった遷移規則に 100 ビットのランダムな初期値を与え、その遷移を 200 世代にわたり記録したものである。(a), (b), (d) では、系の振舞いは単純なアトラクタに収束するクラス II に相当するものであることが分かり共通した傾向が見られる。一方、(c) はカオス状態を示すクラス III に相当している。これは (c) の例が、2 状態 3 近傍の実験結果であり、 $\lambda = 0.5$ の遷移規則から生成されているためである。この結果から見ると、一部の例をのぞきクラス II に属する遷移規則が有効であることが分かる。

提案手法は、複数の CA 系を用いること、CA 空間の大きさが変化すること、の 2 点から通常の CA とは異なるものの、CA の計算能力や系の相転移などに対するさまざまな議論に対してなんらかの示唆を含んでいるかもしれない。なぜなら符号化とは最も基本的かつユニバーサルな計算能力を表すといえるからである。

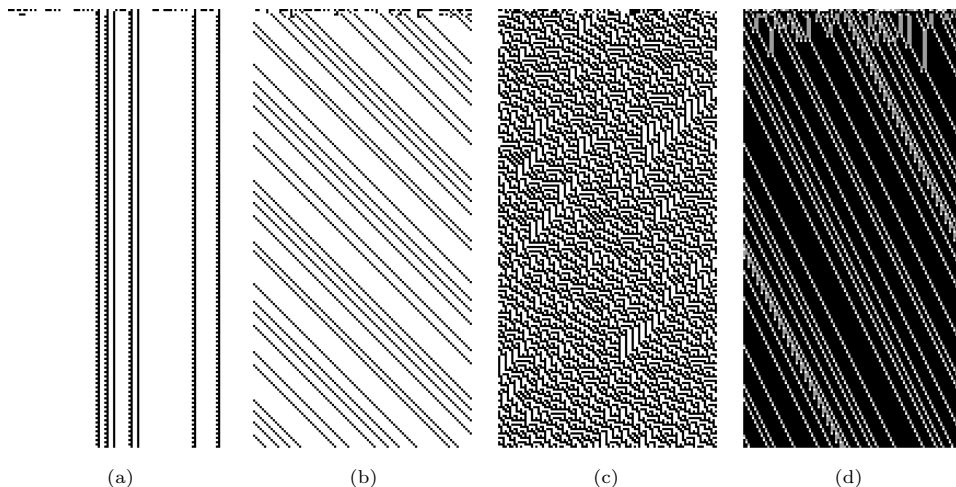


図 13 符号化能力の高い CA の時系列表示：(a) 2 状態 5 近傍 4 パターンを 2 ビットに落とした場合、(b) 2 状態 5 近傍 5 パターンを 3 ビットに落とした場合、(c) 2 状態 3 近傍 4 パターンを 2 ビットに落とした場合、(d) 3 状態 3 近傍 6 パターンを 2 ビットに落とした場合

Fig. 13 Encode ability of CA: (a) 2 states 5 neighbors with 4 patterns to 2 bit, (b) 2 states 5 neighbors with 5 patterns to 3 bit, (c) 2 states 3 neighbors with 4 patterns to 2 bit, (d) 3 states 3 neighbors with 6 patterns to 2 bit.

今回の実験結果からは、実際の遷移規則はクラス II や III に属するものであることが分かり、カオスの縁における計算能力についての議論^{(5),(6),(11)}との関連性は確認されなかった。このような議論に関しては Mitchell らの再検証⁽¹⁰⁾も同じように否定的な見解を示している。クラス III に関しては、二次元の CA において結晶化とよばれる状態が見られるという報告もあり⁽¹²⁾、これらとの比較検討も今後の課題としたい。

5.3 パターンに対する非対称性と符号化能力

G を軸とした解析ではほとんどの実験で $G = 0$ 付近で符号化能力が落ち込むことが分かった。 G を -1 から 1 の値で正規化した場合、ほぼ $G = |0.3| \sim |0.5|$ で符号化能力が高いことが表 1 から理解できる。3 状態 3 近傍の実験については、やや左肩上がりのグラフになり、最小値は正規化値で $G = 0.33$ を示している。これは状態の増加によって符号化能力の重心位置がずれることを示している。これらのことから、パターンに対して適度に非対称性を持った応答が問題解決能力に深く寄与していることがうかがえる。

5.4 符号化能力と遷移規則の非対称的特徴

以上の結果をまとめると、提案手法では以下の 3 つの特徴を持つ遷移規則の符号化能力が高いという傾向を得た。

- (1) 遷移規則のエントリのパターンに対する応答が非対称な場合。
- (2) λ パラメータが約 0.2 から 0.3.

- (3) 遷移規則がクラス II に属する。

ただし λ や G が結果に対して相関を持っていると思われるものの、状態・近傍数および入力パターン数に応じてその影響の度合いが変化する場合がある。また、パラメータで指定した遷移規則のうちのごく一部だけが符号化能力を有しており、すべてのケースにおいて普遍的かつ的確に符号化能力の高い遷移規則を抽出する基準は発見できなかった。

一方、以下の 2 点の特徴のうちどちらかを有している場合はすべての実験に共通して符号化能力が低いことが判明した。

- (1) $\lambda = 1 - 1/k$ でなおかつ $G = 0$ に近いとき。
- (2) λ ないし G がそれぞれがとりうる最大・最小値に近いとき。

以上のことから、符号化能力には入力に対して各状態が等確率で出力されないような確率の不等性、ならびに遷移規則のパターンに対する非対称的応答性が関与していると推測される。

6. 結 論

本研究では、CA を用いた符号化手法を提案した。提案手法は決定論的操作で簡便かつ高速に符号化を行えるものである。提案手法は二次元以上および二値以上のパターンに対しても適用可能である。符号化可能な遷移規則を用いれば、簡単なパターン分類器として用いることが可能である。提案手法を具体的なアプリ

ケーションに応用するには、改善しなければならない問題や検証しなければならない問題が多いが、CAを用いた情報処理に1つの可能性を示すものとして提示したい。

特に提案手法は入力とCAの遷移規則の関係からすべての場合で正しく符号化できるとは限らないという制限がある。本研究ではLangtonの提案した λ パラメータと新たに定義したパラメータ G を用いてこれを検証し、入力に対する応答確率の不等性ならびに遷移規則のパターンに対する非対称的応答性が符号化能力に寄与しているのではないかの推測を行った。

7. 今後の展望

符号化能力の高い遷移規則をよりの確に指し示す指標を探索したい。このような指標がパラメータ G とともに他のCA研究にも寄与することを願いたい。また、いくつかの拡張したCAが提案されている¹³⁾。これらに対しても提案手法を適用して、その効果を検証したい。

参考文献

- 1) von Neumann, J.: *Theory of self-reproducing automata*, edited and completed by A. Burks, University of Illinois Press (1966).
- 2) Berlekamp, E., Conway, J. and Guy, R.: *Winning Ways for your Mathematical Plays*, Academic Press (1982).
- 3) Wolfram, S.: Statistical mechanics of cellular automata, *Rev. Mod. Phys.*, Vol.55, pp.601–644 (1983).
- 4) Wolfram, S.: Universality and Complexity in Cellular Automata, *Physica D*, Vol.10, pp.1–35 (1984).
- 5) Langton, C.G.: Computation at the Edge of Chaos: Phase Transitions and Emergent Computation, *Physica D*, Vol.42, pp.12–37 (1990).
- 6) Packard, N.H.: Adaptation toward the edge of chaos, *Dynamic Patterns in Complex Systems*, pp.293–301, Singapore, World Scientific (1988).
- 7) Chua, L.O. and Yang, L.: Cellular neural networks: Theory, *IEEE Trans. Circuits and Syst.*, Vol.35, No.10, pp.1257–1272 (1988).
- 8) Chua, L.O. and Yang, L.: Cellular neural networks: Applications, *IEEE Trans. Circuits and Syst.*, Vol.35, No.10, pp.1273–1290 (1988).
- 9) Mitchell, M., et al.: Revisiting the edge of chaos: evolving cellular automata to perform computations, *Complex Systems*, Vol.7, pp.89–130 (1993).
- 10) Mitchell, M., Crutchfield, J.P. and Hrabar, P.T.: Dynamics, Computation and the “Edge of Chaos”: A Re-Examination, Cowan, G., Pines, D. and Melzner, D. (Eds), *Complexity: Metaphors, Models and Reality, Santa Fe Institute Studies in the Sciences of Complexity*, Vol.19. Addison-Wesley (1993).
- 11) Kauffman, S.A.: *The origin of order*, Oxford University Press (1993).
- 12) Suzudo, T.: Crystallization of two-dimensional cellular automata, *Complexity International*, Vol.6 (1999).
- 13) Mori, T., et al.: Edge of Chaos in Rule-changing Cellular Automata, *Physica D*, Vol.116, pp.275–282 (1998).

(平成 14 年 10 月 30 日受付)

(平成 14 年 12 月 5 日再受付)

(平成 14 年 12 月 22 日採録)

福原 義久 (正会員)



平成 9 年慶應義塾大学環境情報学部卒。平成 11 年同大学院政策・メディア研究科修士課程了。平成 14 年同博士課程了。現在同大 SFC 研究所訪問所員。主な専門はニューラルネットワーク・コンピューティング。現在は対称性の破れと情報処理の関係性についての研究も行っている。共著に「複雑系入門」(井庭 崇, 福原 義久, NTT 出版, 1998)。

武藤 佳恭



Dr. Yoshiyasu Takefuji: professor of Keio University from 1992, was on the faculty of Case Western Reserve Univ. since 1988, Univ. of South Carolina (1985~1988), Univ. of South Florida (1983~1985). Research: neural computing and hyperspectral computing. Awards: NSF-RIA in 1989, distinguished service from IEEE Trans. on NN in 1992, best paper of IPSJ in 1980 and that of IFAC at AIRTC'98, TEPCO and KAST in 1993, Takayanagi award in 1995, KDD award in 1997, NTT tele-education courseware award in 1999. Government advisor: NCC of Philippines, VITTI of Vietnam, Jordan CTTISC, Thailand, Srilanka, Hong Kong, Multimedia Univ. in Malaysia. Published: 22 books and more than 200 papers.

10) Mitchell, M., Crutchfield, J.P. and Hrabar,