

1次元接続有限状態セルオートマトンによる 冪乗数列生成の実現

上川 直紀^{1,a)} 梅尾 博司^{1,b)}

概要:セルオートマトン (CA) は J. von Neumann により考案された並列計算モデルの 1 つであり, 現在では複雑系などの多くの分野で研究がなされている. CA はセルと呼ばれる有限状態オートマトンにより構成され, セルは自らと, 隣接するセルの内部状態という局所的な情報を元に, 自らの内部状態を遷移させる機能しか持たない. しかしながら, この局所的な相互作用がモデル全体に影響を及ぼし, CA は巨大で複雑な事象をシミュレートすることができるという特徴を持つ. CA 上で考察されている問題の 1 つに実時間数列生成問題が存在する. これまで, 素数列, Fibonacci 数列, 数列 $\{2^n | n = 1, 2, 3, \dots\}$ などの非正則数列の CA 上の生成アルゴリズムが明らかにされている. 本稿では, CA 上の数列 $\{n^3 | n = 1, 2, 3, \dots\}$ 生成アルゴリズムについて考察を行い, その正当性を明らかにする. 本稿では, k, n を $k \geq 2, n \geq 1$ となる自然数とした場合, $\{k^n | n = 1, 2, 3, \dots\}$ として表される冪乗数列の 1 次元接続 CA 上の生成アルゴリズムについて考察を行ない, すべての k の場合の冪乗数列 $\{k^n | n = 1, 2, 3, \dots\}$ が生成可能であることを示す.

キーワード:セルオートマトン, 並列アルゴリズム, 実時間数列生成問題, 複雑系

Realization of Exponential Sequence Generation by One-dimensional Finite-state Cellular Automata

1. はじめに

セルオートマトン (Cellular Automaton, 以下では CA と略す.) は生物固有の能力である自己増殖, 自己複製機能を形式的に記述するモデルとして, J. von Neumann [9] により考案された並列計算モデルであり, 現在では, 複雑系などの多くの分野で研究がなされている. CA はセルと呼ばれる有限状態オートマトンにより構成され, セルは自らと, 隣接するセルの内部状態という局所的な情報を元に, 自らの内部状態を遷移させる機能しか持たない. しかしながら, この局所的な相互作用がモデル全体に影響を及ぼし, CA は巨大で複雑な事象をシミュレートすることができるという特徴を持つ. CA は生命現象の知られざる秘密の探りたいという目的で提案され, 発展し, 現在では, 複雑系などの多くの分野で研究がなされている. CA の応

用例として脇田, 清水, 玉城, 北 [12] らの交通流のシミュレーションが挙げられる.

CA の研究課題の 1 つとして, 実時間数列生成問題が挙げられる. Wolfram [13], Shackleford et al. [11], Pazo-Roblesa and Fuster-Sabaterb [10] らは 2 状態の CA による乱数生成器について考察を行なった. これらの研究では, 1 次元の 2 状態 CA のパターンにより 2 進数を表現し, 時間経過により生成される乱数列について言及した.

一方, 有沢 [1], Korec [8], Kamikawa and Umeo [2], [4], [5] らは上記の研究とは異なり, 1 次元 CA の左端のセルの内部状態で数列を表現する形式で, CA 上の数列生成問題について考察を行った. 有沢 [1] は CA 上の素数列, Fibonacci 数列, 数列 $\{2^n | n = 1, 2, 3, \dots\}$, 数列 $\{n^2 | n = 1, 2, 3, \dots\}$ の生成アルゴリズムを, Korec [8] は素数列の生成アルゴリズムを明らかにした. 有沢 [1] が明らかにしたアルゴリズムは, 最適となる数列生成時間に対して 2 倍の生成時間が必要となる 2 線形時間生成アルゴリズムであり, Korec [8] が明らかにしたアルゴリズムは, 生成時間について最適となる実時間生成アルゴリズムである. Kamikawa

¹ 大阪電気通信大学
Osaka Electro-Communication University, Neyagawa, Osaka, 572-8530, Japan

a) naoki@osakac.ac.jp

b) umeo@osakac.ac.jp

and Umeo [4], [5] は有沢 [1] のアルゴリズムを発展させ、CA 上の Fibonacci 数列, 数列 $\{2^n | n = 1, 2, 3, \dots\}$, 数列 $\{n^2 | n = 1, 2, 3, \dots\}$ の実時間生成アルゴリズムを設計し, その正当性について明らかにした. また, Kamikawa and Umeo [4], 上川, 梅尾 [6] は特定の数列生成アルゴリズムについて考察を行なうのではなく, 1 状態および 2 状態の CA で生成可能な数列を明らかにした. これらの研究では, CA の計算能力について考察が行われており, 有限状態数の CA で生成 (計算) 可能である多くの数列が明らかにされている.

本研究では, k, n を $k \geq 2, n \geq 1$ とする自然数とした場合, $\{k^n | n = 1, 2, 3, \dots\}$ として表される冪乗数列の 1 次元接続 CA 上での実時間生成について考察を行なう. 冪乗数列 $\{k^n | n = 1, 2, 3, \dots\}$ は 1 階線形回帰数列であり, CA 上の線形回帰数列生成アルゴリズムについては, Kamikawa and Umeo [3] が言及している. しかしながら, Kamikawa and Umeo [3] が示したのは, アルゴリズム設計のアウトラインのみである. 本項では, Kamikawa and Umeo [3] が示したアルゴリズムをもとに, すべての k の場合の冪乗数列 $\{k^n | n = 1, 2, 3, \dots\}$ が有限状態の 1 次元接続 CA で生成可能であることを示す.

2. セルオートマトン上の数列生成問題

CA はセルと呼ばれる有限状態オートマトン A の有限個のアレイで構成される. 図 1 参照.

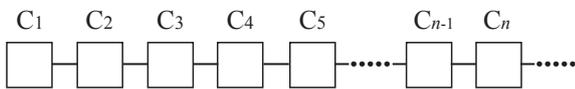


図 1 1次元接続セルオートマトン

$n \geq 1$ とした場合, 左端から $C_1, C_2, C_3, \dots, C_n$ と呼ぶ. 有限状態オートマトン A を定式化すると, $A = (Q, \delta, c, a)$ となる. それぞれ, 以下の意味を持つ.

- (1) Q は内部状態の有限集合である.
- (2) δ は状態遷移関数であり, 次のように定義される.

$$\delta: Q \times Q \times Q \rightarrow Q$$

この場合の状態遷移関数 $\delta(a, b, c) = d$ ($a, b, c, d \in Q$) は次の意味を持つ.

あるステップ t 時に, セル C_i の内部状態が b であり, セル C_{i-1} の内部状態が a , セル C_{i+1} の内部状態が c であると, 次のステップ $t+1$ 時にセル C_i の内部状態が d に遷移する.

左端のセル C_1 は左側からの入力として常に外界を表す特殊な状態 s が入力される. また静止状態 $q(\in Q)$ は隣接する左右のセルの状態が q の場合, q を維持し続ける

という特徴を持つ. すなわち, 遷移規則 $\delta(q, q, q) = q$, $\delta(s, q, q) = q$ が定義される.

- (3) 状態 $c(\in Q)$ は初期計算状況時にセル C_1 がとる特別な状態である.
- (4) 状態 $a(\in Q)$ は数列生成に使用する特別な状態である.

次に, セル空間を記述する記法を導入する. i, j, n を正の自然数, t を正の整数とし, $1 \leq n, 1 \leq i \leq n, 1 \leq j \leq n, i < j, t \geq 0$ とする. 時刻 t 時のセル C_i の内部状態を s_i^t と表し, 時刻 t 時の n 個のセルからなるセル空間を以下の様に表す.

$$t: s_1^t \dots s_n^t$$

また, 時刻 t 時にセル C_1 からセル C_{i-1} の $i-1$ 個のセル全ての内部状態が 0 , セル C_i からセル C_j の $j-i+1$ 個のセル全ての内部状態が s , セル C_{j+1} 以降のセル全ての内部状態が u の場合, 以下の様にまとめて記述する.

$$t: \underbrace{0 \dots 0}_{[1, i-1]} \underbrace{s \dots s}_{[i, j]} \underbrace{u \dots u}_{[j+1, \dots]}$$

次にセル空間の 1 ステップの変化を表す演算記号 " \Rightarrow " を導入する. 以下の様に, \Rightarrow の左右にセル空間の状態を記述した場合, \Rightarrow の左側のセル空間の状態が変化前の状態で, 右側が 1 ステップ後のセル空間の状態となる.

$$t: \underbrace{0}_{[1]} \underbrace{ss \dots}_{[2, \dots]} \Rightarrow t+1: \underbrace{00}_{[1, 2]} \underbrace{ss \dots}_{[3, \dots]}$$

また, 遷移規則の簡略記法も導入する. 遷移規則 $\delta(w, x, y) = z$ の場合, 省略して $w \ x \ y \rightarrow z$ と記述する.

初期計算状況, すなわち, 時刻 $t=0$ 時のセル空間は以下に示す通り, セル C_1 の内部状態は c をとり, C_1 以外のセルの内部状態は静止状態 q をとる.

$$t=0: \underbrace{c}_{[1]} \underbrace{q \dots}_{[2, \dots]}$$

r, n を自然数とし, $r \geq 1, n \geq 1$ とする. $\{t(n) | n = 1, 2, 3, \dots\}$ を無限に単調増加する正整数の数列とすると, すべての n について, $s_1^{r \cdot t(n)} = a$, すなわち, $t = r \cdot t(n)$ 時のみにセル C_1 の内部状態が a を取ると, r 線形時間で, 数列 $\{t(n) | n = 1, 2, 3, \dots\}$ を生成すると言う. 特に, $r=1$ の時は数列生成時間について最適となり, 実時間で数列 $\{t(n) | n = 1, 2, 3, \dots\}$ を生成すると言う.

3. 冪乗数列生成の実現

本項では 1 次元接続セルオートマトンを用いた冪乗数列の生成について述べる. k, n を自然数とし, $k \geq 2, n \geq 1$ とする. 生成する冪乗数列を a_n とすると, $a_n = k^n$ となる. また数列 a_n は以下の漸化式として表す事ができる.

$$a_1 = k, a_{n+1} = k \cdot a_n$$

数列 a_n は 1 階線形回帰数列であるので, Kamikawa and Umeo[3] が示した 1 階線形回帰数列の生成アルゴリズムを発展させて, 数列 a_n の実時間生成を実現する.

M_k を有限状態 CA とし, $M_k = (Q_k, \delta_k, c, a)$ とする. M_k が数列 $a_n = k^n$ を生成するとき, 有限状態集合 Q_k は以下の通り定まる.

$$Q_k = \{q, a, b, c, \overbrace{d_1, d_2, \dots, d_{k-1}}^{k-1}, \overbrace{e_1, e_2, \dots, e_{k-2}}^{k-2}\}$$

$$|Q_k| = 2k + 1$$

l を自然数とし, $l \geq 1$ とする. 冪乗数列生成の実現について, $k = 2, k = 2l + 2, k = 2l + 1$ の場合に分けて考える.

3.1 $k = 2$ の場合

最初に $k = 2$ の場合を考える. この場合, 生成する数列は $a_n = 2^n$ となり, a_n は以下の漸化式で表すことができる.

$$a_1 = 2, a_{n+1} = 2 \cdot a_n$$

M を 5 状態 CA とし, $M = (Q, \delta, c, a)$, $Q = \{q, a, b, c, d_1\}$ とする. 数列 $\{2^n | n = 1, 2, 3, \dots\}$ は表 1 に示す状態遷移規則集合により生成される.

表 1 数列 $\{2^n | n = 1, 2, 3, \dots\}$ 生成のための状態遷移規則集合

q	Right State	a	Right State	b	Right State	c	Right State
	q a b c d ₁						
q	q q q c	q	q q q	q	q c c	c	c c c
a	q q q c	a	q q q	a	q c c	a	c c c
b	q q q c	b	q q q	b	q c c	b	c c c
c	q q q c	c	q q q	c	q c c	c	c c c
d ₁	q q q a	d ₁	q q q	d ₁	q q q	d ₁	q q q
\$	q q q a	\$	q q q	\$	q q q	\$	q q q

表の最初の行(列)はそれぞれ, 右(左)側に隣接するセルの状態を示し, 表内のそれぞれのエントリは 1 ステップ後のセルの内部状態を示す. 表 1 に示す状態遷移規則集合により, M の各セルは図 2 に示す様に遷移する.

図 2 の横軸はセル空間であり, 左端から C_1, C_2, C_4, \dots , それぞれの内部状態を表す. 縦軸は時間軸で, 上端を時刻 $t = 0$ として, 下方向に向かって時間の経過を表す. 図 2 より, 内部状態がセル空間を伝播している事が見て取れる. このセル空間上の状態の伝播を波と呼ぶ. CA のアルゴリズムは, 状態の伝播を波として表現し, 時間-空間図を用いて幾何学的に設計を行なう. 図 3 に数列 $\{2^n | n = 1, 2, 3, \dots\}$ の生成アルゴリズムの時間-空間図を示す.

時間-空間図の横軸はセル空間, 縦軸を時刻を表し, 時間-空間図は時間の経過におけるセル空間の変化を表す. 時間-空間図に示す線はセル空間上の波を示す. 数列

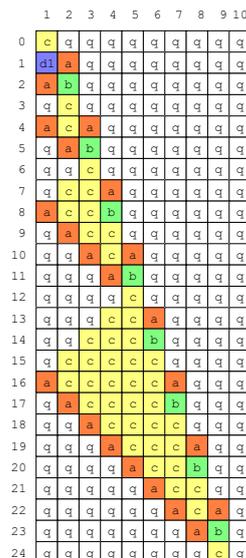


図 2 時刻 $t = 24$ までのシミュレーション状況

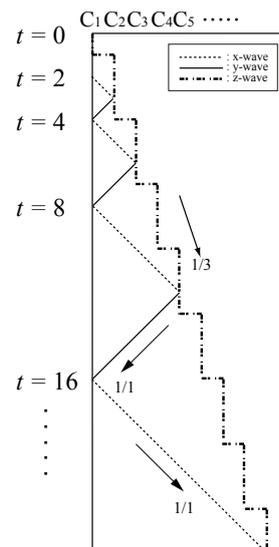


図 3 数列 $\{2^n | n = 1, 2, 3, \dots\}$ 生成のための時間-空間図

$\{2^n | n = 1, 2, 3, \dots\}$ の生成には 3 種類の波, x 波, y 波, z 波を用いる. 図 3 参照.

時刻 $t = 0$ 時に, セル C_1 上で z 波が生成され, 3 ステップにつき 1 ステップだけ右方向に進む. すなわち, z 波の速さは $1/3$ となる. 時刻 $t = 2$ 時に, セル C_1 の内部状態が a に遷移し, セル C_1 上で x 波が生成され, $1/1$ の速さで右方向に進む. x 波がセル空間上を右方向に進み, z 波と衝突する. この時, z 波は右方向に進み続け, x 波は消滅し, 衝突したセル上で y 波が生成され, $1/1$ の速さで左方向に進む. 時刻 $t = 4$ 時, y 波がセル C_1 に到達すると, y 波が消滅し, x 波が生成され, セル C_1 の内部状態が a に遷移する. このように, x 波, y 波をセル C_1 と z 波間を往復運動させ, y 波がセル C_1 に到達した時刻にセル C_1 の内部状態が a に遷移する. セル C_1 の内部状態が a となる時刻は, $t = 2, 4, 8, \dots, 2^n$ 時となる.

m を任意の自然数とし, $m \geq 2$ とする. 時刻 $t = 0$ 時に, セル C_1 上で z 波が生成され, 時刻 $t = m$ 時にセル C_1 の内部状態が a をとり, セル C_1 上で x 波が生成されたとした時, z 波と x 波の衝突により生成された y 波がセル C_1 に到達する時刻を考える. ここでは, m が偶数の場合を考える. N を自然数の集合とし, $P_z(t) : N \cup \{0\} \rightarrow N$ を時刻 t 時に z 波の存在するセルの位置を表す関数とすると, $P_z(t) = \lceil \frac{t}{3} \rceil + 1, t \geq 0$ となる. p を $p \geq 1$ となる自然数とし, 時刻 $t = m + p$ 時に z 波と x 波はセル $C_{P_z(m+p)}$ 上で衝突したとする. 図 4 参照.

x 波は 1 ステップにつき 1 セルだけ伝播する波, すなわち, 速さ $1/1$ の波であるので, 時刻 $t = m + p$ 時にセル C_{p+1} に存在する. よって, $P_z(m + p) = p + 1$ となり, $p = \frac{m}{2}$ となる. 以上より, m が偶数の場合, z 波と x 波は時刻 $t = m + \frac{m}{2}$ 時にセル $C_{\frac{m}{2}+1}$ で衝突する. また, y 波

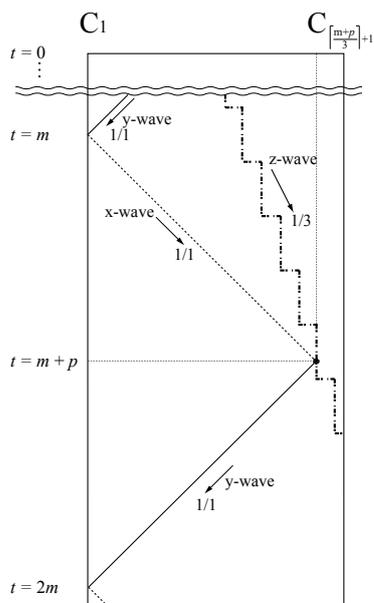


図 4 z 波と x 波の衝突 (m が偶数の場合)

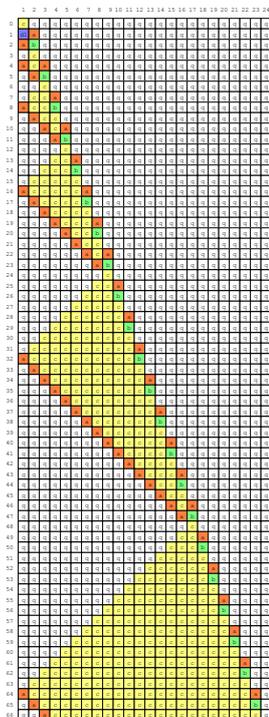


図 5 $k = 2$ の場合の
シミュレーション結果
(時刻 $t = 66$ まで)

の速さも $1/1$ であるので、時刻 $t = m + \frac{m}{2}$ から $\frac{m}{2}$ ステップ後、すなわち、時刻 $t = m + \frac{m}{2} + \frac{m}{2} = 2m$ 時に y 波はセル C_1 に到達し、セル C_1 の内部状態は a に遷移する。この様に、セル C_1 -z 波間の x 波、y 波の往復運動により、 $t = 2, 4, 8, \dots, 2^n$ 時にセル C_1 の内部状態が a となり、数列 $\{2^n \mid n = 1, 2, 3, \dots\}$ が実時間で生成される。図 5 に $k = 2$ の場合 (数列 $\{2^n \mid n = 1, 2, 3, \dots\}$) のシミュレーション結果を示す。

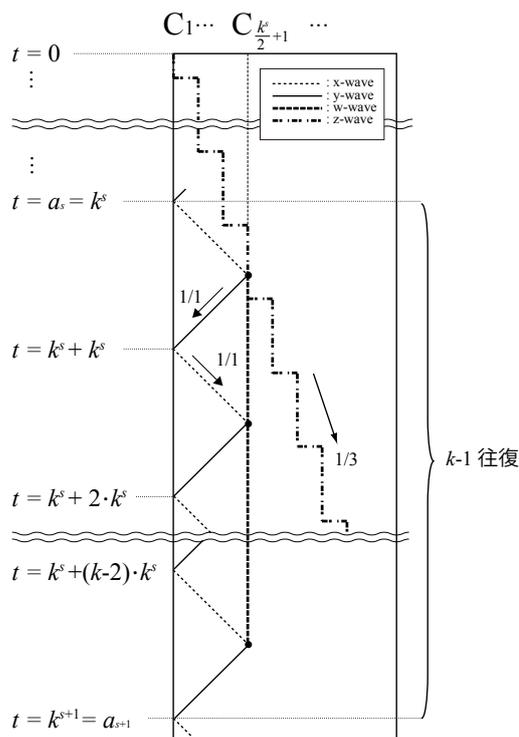


図 6 $k = 2l + 2$ の場合の数列 $\{k^n \mid n = 1, 2, 3, \dots\}$ 生成のための時間-空間図

3.2 $k = 2l + 2$ の場合

次に、 $k = 2l + 2$ の場合を考える。図 6 に $k = 2l + 2$ の場合の時間-空間図を示す。

$k = 2l + 2$ の場合は、x 波、y 波、z 波に加え、その場に留まり続ける速さ 0 の波である w 波を使用する。x 波、y 波、z 波の伝播については、 $k = 2$ の場合と同一であるが、異なるのは、z 波と x 波が衝突した際に、衝突したセル上に w 波を生成し、セル C_1 -w 波間に $k - 1$ 回の往復運動を行う x 波、y 波を生成する点である。s を任意の自然数とし、 $s \geq 1$ とする。時刻 $t = 0$ 時にセル C_1 上で z 波が生成され、時刻 $t = k^s$ 時にセル C_1 の内部状態が a をとり、x 波が生成されたとすると、w 波はセル $C_{\frac{k}{2}+1}$ 上に生成される。セル C_1 -セル $C_{\frac{k}{2}+1}$ を速さ $1/1$ の波が 1 往復するには k^s ステップ必要である。以上より、 $t = k^s$ 時にセル C_1 で生成された x 波により始まる $k - 1$ 回の往復運動によ

り、 $t = k^s + \overbrace{k^s + k^s + \dots + k^s}^{k-1} = k \cdot k^s = k^{s+1}$ 時にセル C_1 の内部状態が a に遷移する。また、w 波は $k - 1$ 回目の往復運動の際に消滅する。 $k = 2$ の場合に対して、内部状態 $\overbrace{e_1, e_2, \dots, e_{k-2}}^{k-2}$ が追加となっているが、これらの状態は、 $k - 1$ 回の往復運動を実現するために用いられる。1 回目の往復運動には状態 a, e_1 が、2 回目の往復運動には状態 b, e_2 が、 \dots 、 $k - 2$ 回目の往復運動には状態 b, e_{k-2} が、 $k - 1$ 回目の往復運動には状態 b, c がそれぞれ用いられる。また、状態 $\overbrace{d_1, d_2, \dots, d_{k-1}}^{k-1}$ については、 a_1 の生成

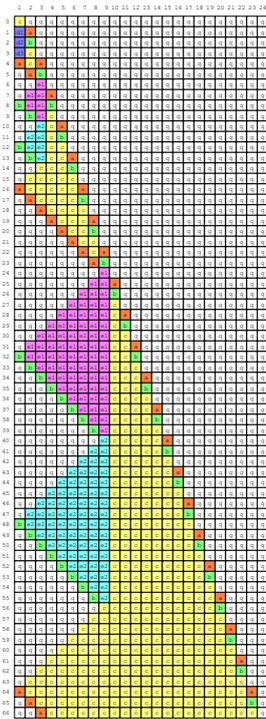


図 7 $k = 4$ の場合の
シミュレーション結果
(時刻 $t = 66$ まで)

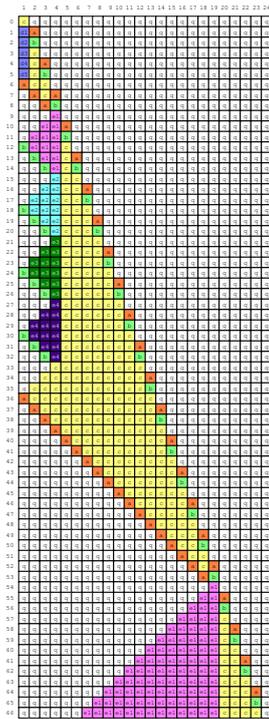


図 8 $k = 6$ の場合の
シミュレーション結果
(時刻 $t = 66$ まで)

の実現, すなわち, セル C_1 を最初に状態 a に遷移させるために用いる. 時刻 $t = 0$ にセル C_1 の内部状態は c をとり, 1ステップごとに d_1, d_2, \dots, d_{k-1} に遷移し, 時刻 $t = k$ 時にセル C_1 は状態 a に遷移する. 以上より, $k = 2\ell + 2$ の場合も数列 $\{k^n \mid n = 1, 2, 3, \dots\}$ が実時間で生成される. 例として, 図 7 に $k = 4$ の場合 (数列 $\{4^n \mid n = 1, 2, 3, \dots\}$) の, 図 8 に $k = 6$ の場合 (数列 $\{6^n \mid n = 1, 2, 3, \dots\}$) のシミュレーション結果を示す.

3.3 $k = 2\ell + 1$ の場合

次に, $k = 2\ell + 1$ の場合を考える. 図 9 に $k = 2\ell + 1$ の場合の時間-空間図を示す.

この場合も, $k = 2\ell + 2$ の場合と同様に, z 波と x 波が衝突したセルに w 波を生成し, セル C_1 - w 波間の x 波, y 波の往復運動を $k - 1$ 回行うことで, 数列 $\{k^n \mid n = 1, 2, 3, \dots\}$ の生成を実現する. しかしながら, $k = 2\ell + 2$ の場合と比較して, w 波を生成するセルの位置が異なり, y 波がセル C_1 に到達して x 波を生成するまで 1 ステップの遅延が発生する.

$k = 2$ の場合と同様に, 時刻 $t = m + p$ 時に z 波と x 波はセル $C_{P_z(m+p)}$ 上で衝突したとする. 図 4 参照. $k = 2\ell + 1$ の場合, m は奇数となるので, $P_z(m+p) = \lceil \frac{m+p}{3} \rceil + 1 = p + 1$ より, $p = \lfloor \frac{m}{2} \rfloor = \frac{m-1}{2}$ となる. 以上より, m が奇数の場合, z 波と x 波は時刻 $t = m + \frac{m-1}{2}$ 時にセル $C_{\frac{m-1}{2}+1}$ で衝突し, セル $C_{\frac{m-1}{2}+1}$ で生成された速さ $1/1$ の y 波は時

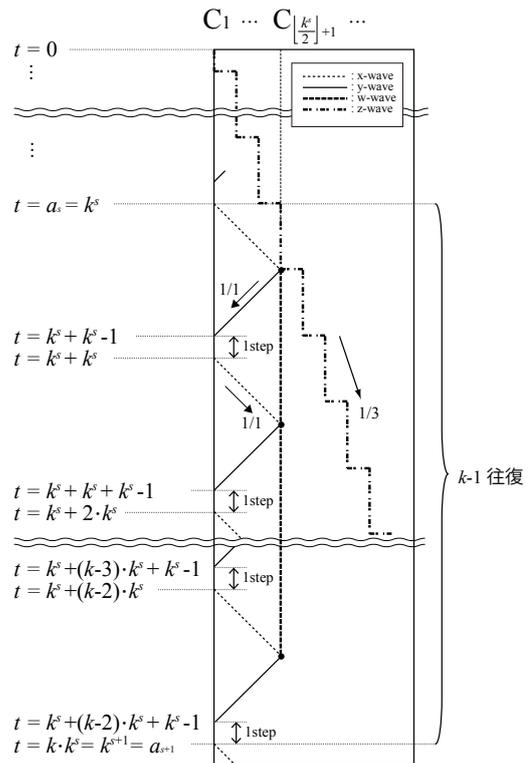


図 9 $k = 2\ell + 1$ の場合の数列 $\{k^n \mid n = 1, 2, 3, \dots\}$ 生成のための時間-空間図

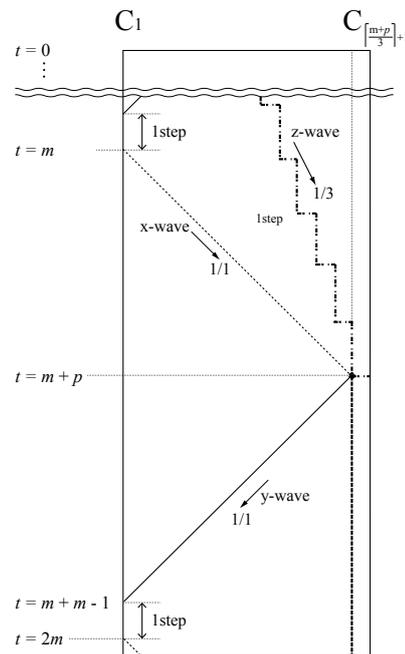


図 10 z 波と x 波の衝突 (m が奇数の場合)

刻 $t = m + \frac{m-1}{2} + \frac{m-1}{2} = 2m - 1$ 時にセル C_1 に到達する. 到達した 1 ステップ後である時刻 $t = 2m$ 時に x 波を生成する. この波の往復を $k - 1$ 回繰り返すことで, $k = 2\ell + 1$ の場合も数列 $\{k^n \mid n = 1, 2, 3, \dots\}$ を実時間で生成することが可能である. 例として, 図 11 に $k = 3$ の場合 (数列 $\{3^n \mid n = 1, 2, 3, \dots\}$) の, 図 12 に $k = 5$ の場合 (数列 $\{5^n \mid n = 1, 2, 3, \dots\}$) のシミュレーション結果を示す.

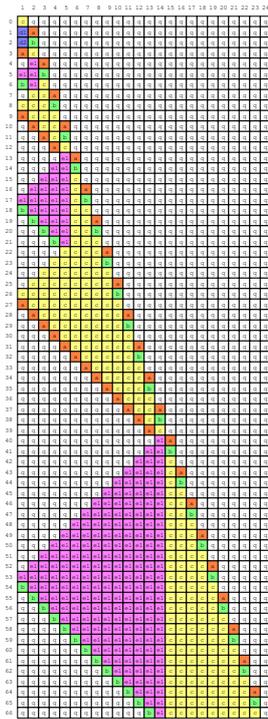


図 11 $k = 3$ の場合の
 シミュレーション結果
 (時刻 $t = 66$ まで)

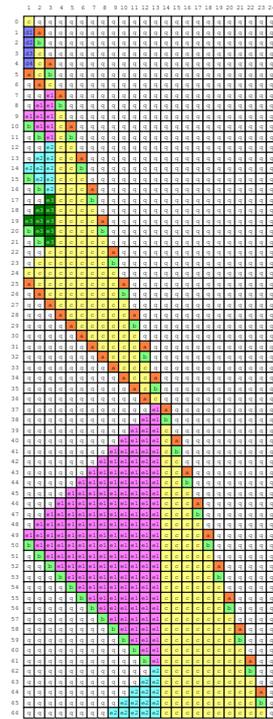


図 12 $k = 5$ の場合の
 シミュレーション結果
 (時刻 $t = 66$ まで)

4. おわりに

本稿では、CA 上の実時間数列生成問題について考察し、 k, n を $k \geq 2, n \geq 1$ となる自然数とした場合、 $\{k^n | n = 1, 2, 3, \dots\}$ として表される冪乗数列の 1 次元接続 CA 上の生成アルゴリズムを設計し、すべての k の場合の冪乗数列 $\{k^n | n = 1, 2, 3, \dots\}$ が有限状態の CA で実時間生成が可能であることを示した。今後の課題としては、内部状態数について考察を行なう事と、線形回帰数列生成アルゴリズムの考察が挙げられる。

参考文献

- [1] 有沢: 有限状態機械の一次元繰り返し形配列による数列の生成方式について. 電子通信学会論文誌 C, Vol. 54, No. 8, pp. 759-766, 1971.
- [2] N. Kamikawa and H. Umeo: Some state-efficient algorithms for real-time generation of non-regular sequences on cellular automata, *Proc. of the 13th International Symposium on Artificial Life and Robotics*, pp.47-50, 2008.
- [3] N. Kamikawa and H. Umeo: A design of algorithms for real-time generation of linear-recursive sequences on cellular automata. The Fourteenth International Symposium on Artificial Life and Robotics, pp.281-286, 2009.
- [4] N. Kamikawa and H. Umeo: A Study on Sequence Generation Powers of Small Cellular Automata. *SICE Journal of Control, Measurement, and System Integration*, Vol. 5, No. 4, pp. 191-199, DOI: 10.9746/jcmsi.5.191, 2012.
- [5] N. Kamikawa and H. Umeo: A construction of five-state real-time Fibonacci sequence generator. *Artificial*

- Life and Robotics*, Vol. 21, No. 4, pp. 531-539, DOI: 10.1007/s10015-016-0309-2, 2016.
- [6] 上川, 梅尾: 能力の小さい 1 ビットセルオートマトンで生成可能な数列についての考察. 情報処理学会論文誌 数理モデル化と応用 (TOM), Vol. 10, No. 1, pp. 1-13, 2017.
- [7] 上川, 梅尾: セルオートマトン上の数列 $\{n^3 | n = 1, 2, 3, \dots\}$ 生成アルゴリズムの設計. 情報処理学会研究報告 数理モデル化と問題解決 (MPS) 2017-MPS-113 (15), pp. 1-6, 2017.
- [8] I. Korec: Real-time generation of primes by a one-dimensional cellular automaton with 9 states, *Proc. of the 2nd International Colloquium on Universal Machines and Computations*, pp.101-116, 1998.
- [9] J. von Neumann: *Theory of Self-Reproducing Automata*, A. W. Burks, Ed., p. 388, Univ. of Illinois Press, 1968.
- [10] M. E. Pazo-Roblesa and A. Fuster-Sabaterb: Modeling pseudorandom sequence generators using cellular automata: The alternating step generator, *Computation in Modern Science and Engineering, Proc. International Conference on Computational Methods in Science and Engineering 2007*, Vol.963, pp.969-972, 2007.
- [11] B. Shackleford, M. Tanaka, R. J. Carter, and G. Snider: FPGA implementation of neighborhood-of-four cellular automata random number generators, *Proc. 2002 ACM/SIGDA Tenth International Symposium on Field-Programmable Gate Arrays*, pp.106-112, 2002.
- [12] 脇田, 清水, 玉城, 北: 合流車両による交通渋滞緩和のためのセルオートマトンシミュレーション, 日本計算数理工学論文集, Vol.10, pp.75-80, 2010.
- [13] S. Wolfram: Random sequence generation by cellular automata, *Advances in Applied Mathematics*, Vol. 7, No. 2, pp. 123-169, 1986.