Complex Systems 1 (1987) 1045–1062

Scaling of Preimages In Cellular Automata

Erica Jen
Theoretical Division, MS-B258, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

Abstract. A cellular automaton consists of a lattice of sites whose values evolve deterministically according to a local interaction rule. For a given rule and arbitrary spatial sequence, the preimage of the sequence is defined to be the set of tuples that are mapped by the rule onto the sequence. Recurrence relations are provided that express the number of preimages for a general spatial sequence in terms of the number of preimages for its subsequences. These relations are applied to the analysis of a quantity α_n defined as the total number of preimages for spatial sequences of length n whose probabilities of occurrence are in a certain sense minimal after one iteration of the rule. In particular, the recurrence relations are used to characterize automata rules by parameters representing the amount of information about an arbitrary spatial sequence needed to determine the values that, when appended to the sequence, minimize the number of preimages. On the basis of this characterization, it is proved that, for all nearest-neighbor automata rules on infinite lattices, the quantity α_n scales exactly with the length n of the spatial sequence; that is, $\alpha_n = c \cdot 2^n$ for n sufficiently large, where c is a constant depending on the rule. A symmetric result holds in the case of maximal preimages.

1. Introduction

This paper discusses preimages for one-dimensional cellular automaton rules on infinite lattices. For a given rule and arbitrary spatial sequence, the *preimage* of the sequence is defined to be the set of tuples that are mapped by the rule onto the sequence.

Preimages provide information on the probability distribution of spatial sequences associated with an automaton rule. For example, the number of preimages for a sequence determines its probability of occurrence after one iteration of the rule operating on an initial condition in which all sequences appear with uniform probability. Preimages have been been the subject of many studies focussing on questions including the relationship between number of preimages and statistical measures such as dimension and entropy [1], the relationship among preimages, surjectivity, and reversibility [2,3,4], the existence of spatial sequences termed "gardens of Eden" with no preimages [5,6], and the existence of sequences with infinite-order preimages (that is, sequences that are the image of some other sequence after the rule has been applied for any finite number of time steps) [7].

In this paper, a coupled system of recurrence relations is introduced for finding the number of preimages of general spatial sequences. The recurrence relations categorize and count preimages according to their endtuples. The finite neighborhood size of cellular automata rules implies that if the endtuples are long enough, then these counts for sequences of length k suffice to determine the number of preimages for sequences of length $k+1$. The recurrence relations can be reformulated in matrix form, and thus represent an analytically useful (and computationally feasible) technique for enumeration of preimages.

The recurrence relations for general spatial sequences are used here to study the dependence of the number of preimages on sequence length. This dependence will be analyzed in particular for a set of spatial sequences whose preimages are in a certain sense minimal under the rule. The set is defined as follows: Given any sequence S of length n, construct "extended" sequences of length $n+2$ consisting of S together with new values appended on the left and right. Include in the set the extended sequence with the minimal number of preimages. Then, define a quantity α_n to be the total number of preimages for all the sequences in the set. In the case of nearest-neighbor, binary site-valued automata rules, this quantity α_n will be shown here to scale exactly with the length of the spatial sequences; i.e., $\alpha_n = c \cdot 2^n$ for sufficiently large n, where c is a constant depending on the rule. The crucial observation used in establishing this scaling behavior is that it is not in general necessary to know *all* the values of a sequence in order to determine the values that, when appended to the sequence, minimize its number of preimages.

A symmetric result holds for maximal preimages; i.e., the total number of preimages for the set of extended sequences with maximal number of preimages also scales exactly with the length of the sequence. The difference between the scaling coefficient for maximal versus minimal preimages can be interpreted as indicating the degree to which the rule changes a frequency distribution after one iteration. (See reference [8] for a discussion of this and other implications of the scaling behavior.)

This paper is organized as follows. Section 2 provides the recurrence relations that maintain counts of preimages for spatial sequences for general automata of arbitrary neighborhood size. The last part of the section reformulates these relations in matrix form. Section 3 then provides the proof of the exact scaling behavior in the case of nearest-neighbor automata rules. The questions discussed in this section include: the use of endtuples to "guess" the values that, when appended to a sequence, minimize (maximize) the number of preimages; the derivation of the parameters characterizing the amount of information about a sequence needed to ensure the correctness of this guess; and the features of the rules themselves that give rise to differing values for these parameters.

2. Recurrence relations

The general form of a one-dimensional cellular automaton on an infinite lattice is given by

$$x_i^{t+1} = f(x_{i-r}^t, \cdots, x_i^t, \cdots, x_{i+r}^t), \quad f : F_k^{2r+1} \to F_k,$$

where x_i^t denotes the value of site i at time t, f represents the "rule" defining the automaton, and r is a non-negative integer specifying the radius of the rule. The site values are restricted to a finite set of integers $F_k = \{0, 1, \cdots, k-1\}$, and are computed synchronously (in parallel) at each time step.

Let $S = s_0 \cdots s_n$ be an arbitrary sequence and R be an arbitrary cellular automaton rule of radius r. Denote the number of preimages of S under the rule R by $N(S)$. The objective is to develop recurrence relations for $N(S)$ based on the number of preimages of its subsequences; i.e., $s_0, s_0s_1, s_0s_1s_2, \cdots$ (moving from left to right), or equivalently, $s_n, s_{n-1}s_n, s_{n-2}s_{n-1}s_n, \cdots$ (moving from right to left). Clearly, at each step the number of preimages of a sequence $N(s_1 \cdots s_j)$ can be found exactly from either $N(s_0 \cdots s_{j-1})$ or $N(s_1 \cdots s_j)$ if the preimages of the shorter sequence are distinguished according to their final $2r$ components. The recurrence relations to be described here therefore maintain individual counts of preimages by final (either on the left or right) components, and then rely on the definition of the automaton rule to update these counts as new values are appended to the sequence.

First consider the recurrence relations for $N(S)$ moving from left to right in the sequence. For any integer $0 \leq m \leq k^{2r} - 1$, with $m = \sum_{i=0}^{2r-1} m_i k^{2r-1-i}$, denote by $\vec{m} = [m_0, \cdots, m_{2r-1}]$ the tuple corresponding to its k-ary representation. The vectors $\vec{m}$ thus range over all possible endtuples of length $2r$. The preimages for sequence S will then be grouped and counted according to their endtuples, with the total number of preimages being the sum over all $\vec{m}$ of the number of preimages beginning with $\vec{m}$; that is,

$$N(S) = \sum_{m=0}^{k^{2r}-1} L_{\vec{m}}^n,$$

where, for any j,

$$L_{\vec{m}}^j = \text{number of preimages beginning with tuple } \vec{m} \text{ for sequence } s_0 \cdots s_j.$$

Now define a characteristic function

$$\begin{aligned} I_j(x) &= 1 \text{ if } s_j = x, \\ &= 0 \text{ if } s_j \neq x, \end{aligned}$$

and for $0 \leq i \leq k - 1$,

$$\begin{aligned} x_i &= f(m_0, \cdots, m_{2r-1}, i), \\ \vec{p_i} &= [m_1, \cdots, m_{2r-1}, i], \end{aligned}$$

It then follows that

$$L^j_{\vec{m}} = \sum_i L^{j-1}_{\vec{p_i}} I_j(x_i), \tag{2.1}$$

and the above coupled system of recurrence relations can be used to express $N(S)$ in terms of starting values L^0 that are easily computed from the definition of the automaton rule.

Similarly, to find $N(S)$ moving from right to left in the sequence, set

$$N(S) = \sum_{m=0}^{k^{2r}-1} R^n_{\vec{m}},$$

with

$$R^j_{\vec{m}} = \text{number of preimages of sequence } s_{n-j} \cdots s_n \text{ ending with tuple } \vec{m}.$$

For all $0 \le i \le k-1$, define

$$\begin{aligned} x_i &= f(i, m_0, \cdots, m_{2r-1}), \\ \vec{p_i} &= [i, m_0, \cdots, m_{2r-2}], \end{aligned}$$

Then, as before, the recurrence relations

$$R^j_m = \sum_i R^{j-1}_{\vec{p_i}} I_j(x_i) \tag{2.2}$$

can be used to express $N(S)$ in terms of known starting values R^0.

For example, consider the binary site-valued nearest-neighbor ($k = 2$, $r = 1$) rule defined by

$$\{000, 001, 011, 100, 101, 110, 111\} \to 0, \quad \{010\} \to 1, \tag{2.3}$$

(Rule 4 according to the labeling scheme of [1]). Then from (2.1),

$$\begin{aligned} L^j_{00} &= (L^{j-1}_{00} + L^{j-1}_{01}) I_j(0), \\ L^j_{01} &= L^{j-1}_{11} I_j(0) + L^{j-1}_{10} I_j(1), \\ L^j_{10} &= (L^{j-1}_{00} + L^{j-1}_{01}) I_j(0), \\ L^j_{11} &= (L^{j-1}_{10} + L^{j-1}_{11}) I_j(0). \end{aligned}$$

and therefore the number of preimages of the sequence 0010, for instance, is given by

$$\begin{aligned} N(0010) &= L^3_{00} + L^3_{01} + L^3_{10} + L^3_{11}, \\ &= 2L^2_{00} + 2L^2_{01} + L^2_{10} + 2L^2_{11}, \\ &= 3L^1_{00} + 3L^1_{01} + 2L^1_{10} + 4L^1_{11}, \\ &= 3L^0_{10}, \\ &= 6, \end{aligned}$$

since $\{100, 101\} \to 0$ and hence, for this rule, $L^0_{10} = 2$. Equivalently, the number of preimages of 0010 moving right to left can be computed from (2.2) as

$$\begin{aligned} N(0010) &= R^3_{00} + R^3_{01} + R^3_{10} + R^3_{11}, \\ &= 2R^2_{00} + R^2_{01} + 2R^2_{10} + 2R^2_{11}, \\ &= 2R^1_{01}, \\ &= 2R^0_{00} + 2R^0_{10}, \\ &= 6, \end{aligned}$$

since the definition of the rule implies that $R^0_{00} = 2$ and $R^0_{10} = 1$.

It is also useful to re-express relations (2.1, 2.2) in matrix form. Matrix multiplication provides a convenient representation of the effects of appending new values to a spatial sequence being operated upon by a cellular automaton rule. (See [9] for an example of the use of matrices in the analysis of fixed points and limit cycles for automata rules with periodic boundary conditions.) The reformulation of (2.1,2.2) will be given here for binary site-valued, nearest-neighbor rules; the generalization is obvious. Define a matrix M with algebraic entries $m_{ij}, 0 \leq i, j \leq 3$ such that, denoting the binary representation of any integer $0 \leq k \leq 3$ as k_0k_1,

$$\begin{aligned} m_{ij} &= 0 \quad \text{if } i_1 \neq j_0, \\ &= a \quad \text{if } i_1 = j_0 \text{ and } f(i_0, i_1, j_1) = 0, \\ &= b \quad \text{if } i_0 = j_1 \text{ and } f(i_0, i_1, j_1) = 1. \end{aligned}$$

Thus, the i, j^{th} entry of M is zero if the second component of i does not match the first component of j (implying that a three-tuple (i, j) cannot be constructed using i and j). Otherwise, m_{ij} is set to a "marker" variable with possible values a and b (corresponding to the characteristic functions $I_j(0)$ and $I_j(1)$ that appear in (2.1) and (2.2)) denoting the value of $f(i, j)$. All possible images of length n are then given by the entries of the matrix M^n, where the product of matrix entries is defined using conventional addition and associative, but non-commutative, multiplication. (For instance, $abb = ab^2$, but $ab^2 \neq b^2a$.) In particular, for any image sequence $S = s_0 \cdots s_{n-1}$, consider the term $X = \prod x_i$, where

$$\begin{aligned} x_i &= a \quad \text{if } s_i = 0, \\ &= b \quad \text{if } s_i = 1, \end{aligned}$$

and the multiplication is again taken to be associative but non-commutative. Then the number of preimages of S beginning on the left with i and ending on the right with j is given by the coefficient of X in the i, j^{th} entry of M^n.

For example, consider again the rule defined by (2.3). The matrix M is given by

$$M = \begin{pmatrix} a & a & 0 & 0 \\ 0 & 0 & b & a \\ a & a & 0 & 0 \\ 0 & 0 & a & a \end{pmatrix}.$$

To find preimages for sequences of length 3, for example, consider

$$M^3 = \begin{pmatrix} a^3+aba & a^3+aba & a^3+a^2b & 2a^3 \\ a^3+ba^2 & a^3+ba^2 & a^3+bab & a^3+ba^2 \\ a^3+aba & a^3+aba & a^3+a^2b & 2a^3 \\ 2a^3 & 2a^3 & a^3+a^2b & 2a^3 \end{pmatrix}.$$

From the above, it is clear that the sequence $S = 100$, corresponding to the term $X = ba^2$, has exactly three preimages, all of which begin with 01 and end with 00, 01, or 11, whereas the sequence $S = 010$, corresponding to $X = aba$, has four preimages, beginning with either 00 or 10, and ending with either 00 or 01.

Finally, recurrence relations (2.1,2.2) can be solved in general for the number of preimages of arbitrary sequences. (See [10] for details.) The relations are also easily modified to provide explicit expressions, rather than counts, for the preimage strings.

3. Scaling of minimal (maximal) preimages

The recurrence relations provided in section 2 will be used in this section to prove that, for all nearest-neighbor automata rules, the number of preimages for a well-defined set of sequences scales exactly with the length of the sequences. For any sequence $S = s_1 \cdots s_n$, denote by S^* any of the (four) extended sequences of length $n+2$ obtained by appending either a 0 or a 1 to the left and right ends of S. Further define

$$N^-(S) = \min_{S^*} N(S^*), \qquad N^+(S) = \max_{S^*} N(S^*). \tag{3.1}$$

Next consider the quantities

$$\alpha_n^- = 2^{-(n+4)} \sum_S N^-(S), \qquad \alpha_n^+ = 2^{-(n+4)} \sum_S N^+(S) \tag{3.2}$$

where the sum is taken over all sequences S of length n. (The denominators serve to convert the number of preimages to a probability; see [8] for discussion of the significance of this quantity.) In reference [8], it is asserted that for any nearest-neighbor automaton rule,

$$\lim_{n\to\infty} 2^6 \alpha_n^- = c^-, \qquad \lim_{n\to\infty} 2^6 \alpha_n^+ = c^+,$$

where $0 \le c^- \le 16$, $16 \le c^+ \le 64$ are constants depending on the rule, and the coefficient 2^6 will be seen to be a convenient normalizing factor. That result will be established below.

The discussion will focus on the limiting behavior of α^- measuring the minimal number of preimages; the arguments for α^+ are exactly symmetrical. The essence of the proof is to show that for any automaton rule and any sequence S, knowledge of the end components of S suffices to determine the values that should be appended to S in order to minimize the number of preimages. Given that minimization depends only on the end components, it is therefore not necessary to find $N^-(S)$ for every sequence S; the quantity α_n^- can be found by considering the number of preimages for the minimizing left- and right-most end tuples, and then using all possible values for the components in the middle. Lemma 1 assumes that knowledge of end components is sufficient to achieve minimization and indicates how this information is used to computing the limiting value of α_n^-. Lemmas 2 and 3 establish the number of end components necessary for minimization for general nearest-neighbor rules. Finally, theorem 1 combines these results to show that the limiting value of α_n^- is constant for all nearest-neighbor rules.

Lemma 1 1. *Let R be a cellular automaton rule. For every $q_l, q_r \geq 0$, and for every sequence $S = s_1 \cdots s_n$ with $n \geq q_l,\ q_r$ define*

$$\begin{aligned} S(q_l) &= s_1 \cdots s_{q_l}, \\ S(q_r) &= s_{n-q_r} \cdots s_n, \end{aligned}$$

representing, respectively, the leftmost q_l and the rightmost q_r values of the sequence S. Suppose there exist q_l and q_r with $0 \leq q_l, q_r < \infty$ such that, for every sequence S, the tuples $S(q_l)$ and $S(q_l)$ suffice to determine the values $x, y \in \{0,1\}$ such that

$$N(xs_1 \cdots s_n y) = \min_{S^*} N(S^*),$$

where S^* is any of the four sequences that can be obtained from S by appending either a 0 or a 1 to its left and right ends. Then, for n sufficiently large,

$$2^6 \alpha_n^- = 2^{-q_l - q_r} \sum_{\vec{q_l}, \vec{q_r}} N(x(\vec{q_l})\vec{q_l}) N(\vec{q_r} y(\vec{q_r})), \tag{3.3}$$

where α_n^- is defined in (3.2), and the sum is taken over all 2^{q_l} tuples $\vec{q_l}$ and all 2^{q_r} tuples $\vec{q_l}$.

Remark. Note that the minimizing values may be non-unique, in which case q_l, q_r are taken to be the smallest values that produce no inconsistencies. On the left, for example, q_l is the smallest value such that for every sequence S, there exists an x depending only on the q_l leftmost values of S such that

$$N(xS) \leq N(\bar{x}S),$$

where $\bar{x} = 1 - x$, with equality holding in the cases of non-unique minimizing values.

Proof. Let C_n be the set of sequences constructed by taking (i) the preimages of $x(\vec{q_l})\vec{q_l}$ as their leftmost values, (ii) the preimages of $x(\vec{q_r})\vec{q_r}$ as their rightmost values, with $\vec{q_l}$, $\vec{q_r}$ ranging over all possible tuples, and (iii) all possible values in the remaining $n - (q_l + q_r + 2)$ positions. Let D_n be the set of sequences obtained as follows: Consider all possible sequences S of length

n, find the extended sequence S^* that minimizes the number of preimages. If there is more than one such sequence, then set S^* to match a sequence in C_n. Include in D_n all preimages of S^*. Then, if q_l, q_r are such that $\vec{q_l}$ and $\vec{q_r}$ are sufficient to determine the values of x and y that minimize the number of preimages for the complete sequence (with x and y appended to the left and right ends), then any sequence in C_n belongs to D_n, and vice versa. Therefore, for $n \geq q_l + q_r + 2$,

$$\begin{aligned}\alpha_n^- &= 2^{-(n+4)} \,[\text{number of sequences in } C_n], \\ &= 2^{-(n+4)} \sum_{\vec{q_l},\vec{q_r}} N(x(\vec{q_l})\vec{q_l})N(\vec{q_r}y(\vec{q_r}))2^{n-q_l-q_r-2}, \\ &= 2^{-q_l-q_r-6} \sum_{\vec{q_l},\vec{q_r}} N(x(\vec{q_l})\vec{q_l})N(\vec{q_r}y(\vec{q_r})),\end{aligned}$$

and

$$\lim_{n\to\infty} 2^6\alpha_n^- = 2^{-q_l-q_r} \sum_{\vec{q_l},\vec{q_r}} N(x(\vec{q_l})\vec{q_l})N(\vec{q_r}y(\vec{q_r})),$$

which clearly is a constant value. ∎

To illustrate the lemma, consider the rule defined by (2.3). It is easily shown that given any S, the extended sequence with the minimal number of preimages is that obtained by appending a 1 on both ends of S. In this case, $q_l = q_r = 0$, and

$$N(x\vec{q_l}) = N(\vec{q_r}y) = N(1) = 1,$$

implying that for $n \geq 2$,

$$2^6\alpha_n^- = 1.$$

As a second example, consider the rule defined by

$$\{011, 101, 110, 111\} \to 0, \quad \{000, 001, 010, 100\} \to 1, \tag{3.4}$$

(Rule 23 in the scheme of [1]). It will be shown later in this section that for any sequence S, the minimum number of preimages is attained by appending to each end the "toggled" value of the last component. Here, $q_l = q_r = 1$, and

$$\begin{aligned}N(x(\vec{q_l})\vec{q_l}) &= N(10) = 2, && \text{for } \vec{q_l} = 0, \\ &= N(01) = 2, && \text{for } \vec{q_l} = 1, \\ N(\vec{q_r}y(\vec{q_r})) &= N(01) = 2, && \text{for } \vec{q_r} = 0, \\ &= N(10) = 2, && \text{for } \vec{q_r} = 1,\end{aligned}$$

and therefore for $n \geq 4$,

$$2^6\alpha_n^- = 2^{-2}2^4 = 4.$$

The next two lemmas present a sufficient condition for the existence of finite q_l, q_r with the minimization properties described above. The following definition will be useful.

Definition. For any tuple $\vec{q_l}$, define $T_0(\vec{q_l})$ to be the pseudo-set of the two right components of all tuples mapping under the rule to $0\vec{q_l}$, and $T_1(\vec{q_l})$ to be the pseudo-set of the two right components of all tuples mapping under the rule to $1\vec{q_l}$. The pseudo-sets U_0 and U_1 are defined analogously for tuples with 0 and 1 appended to the right. (Note that the term "pseudo-set" is used since all elements are included even though possibly non-distinct.)

The intuition for the next lemmas is best obtained by referring again to the rule defined by (2.3). Consider the question of computing two quantities, $N(0S)$ and $N(1S)$, where S represents an arbitrary sequence. From (2.1),

$$\begin{aligned} N(0S) &= 2L_{00}^{n-1} + 2L_{01}^{n-1} + L_{10}^{n-1} + 2L_{11}^{n-1}, \\ N(1S) &= L_{10}^{n-1}, \end{aligned}$$

implying that independent of the sequence S,

$$N(1S) \leq N(0S).$$

Therefore, the value 1 is always the minimizing value on the left, and $q_l = 0$. (Similarly, $q_r = 0$.) Lemma 3 asserts that the value of q_l can be derived directly from the definition of the rule since the "pseudo-set" T_0 consisting of the two right components of the tuples mapping to 0—that is, $\{00, 00, 01, 01, 10, 11, 11\}$—"contains" that of the tuples mapping to 1—that is, $\{10\}$.

Suppose now the rule is defined by (3.4), and again compare $N(0S)$ and $N(1S)$. The pseudo-set T_0 is $\{01, 10, 11\}$, and the pseudo-set T_1 is $\{00, 00, 01, 10\}$. Neither contains the other, suggesting that $q_l > 0$. It is thus necessary to do a pairwise comparison of $\{N(00S), N(10S)\}$, and of $\{N(01S), N(11S)\}$. First consider the tuples mapping to 00 and 10. Since

$$\{1101, 0110, 1110, 1011, 0111, 1111\} \to 00, \quad \{0101, 0011\} \to 10,$$

the pseudo-set $T_0(0)$ is $\{01, 10, 10, 11, 11, 11\}$ and that for $T_1(0)$ is $\{01, 11\}$. It follows that for any sequence beginning with a 0, the number of preimages is minimized by appending a 1 on the left. Formally, the same result can be seen from (1.2) since

$$\begin{aligned} N(00S) &= L_{01}^{n-2} + 2L_{10}^{n-2} + 3L_{11}^{n-2}, \\ N(10S) &= L_{01}^{n-2} + L_{11}^{n-2}, \end{aligned}$$

implying $N(10S) \leq N(00S)$. Similarly, $N(01S) \leq N(11S)$, and therefore $q_l = 1$.

The sufficient condition suggested by the above two examples is strengthened slightly by the following lemma. The lemma states that in certain cases, the "containment" condition may appear to be violated, and yet still be satisfied given a certain equivalence of certain tuples under the rule. This equivalence is intuitively described as implying that, under the particular rule being considered and in the context of finding minimum preimages on the left, a tuple xy may be replaced by a tuple zy without affecting the enumeration of preimages. Conversely, on the right, under certain conditions a tuple yx may be replaced by yz.

Lemma 2. *Let L^j, R^j be defined as in (2.1) and (2.2), respectively. If $\{xy0, \bar{x}y0\}$ and $\{xy1, \bar{x}y1\}$, with $\bar{x} = 1 - x$, are mapped to pairwise equal values, then $L^j_{xy} = L^j_{\bar{x}y}$. Similarly, if $\{0yx, 0y\bar{x}\}$ and $\{1yx, 1y\bar{x}\}$ are mapped to pairwise equal values, then $R^j_{yx} = R^j_{y\bar{x}}$.*

Proof. The lemma follows directly from the definitions (2.1) and (2.2).

To illustrate the use of lemma 2, consider the rule defined by

$$\{000, 010, 011, 100, 110, 111\} \to 0, \ \{001, 101\} \to 1. \tag{3.5}$$

Then, $T_0 = \{00, 00, 10, 10, 11, 11\}$ and $T_1 = \{01, 01\}$, suggesting that $q_l > 0$. But, $\{010, 110\} \to 0$ and $\{011, 111\} \to 0$ imply that, according to lemma 2, $L^j_{01} = L^j_{11}$ for all j. Hence, T_1 is contained in T_0, and $q_l = 0$. ∎

The above arguments suffice to establish the following lemma.

Lemma 3. *Consider an arbitrary nearest-neighbor automaton rule. Fix $q_l \geq 0$, and let $\vec{q_l}$ be any vector of length q_l. If for all possible tuples $\vec{q_l}$, the pseudoset T_0 either contains, or is contained in, T_1 (the direction need not be the same for all $\vec{q_l}$), subject to the definition of equivalence in lemma 2, then the value q_l is sufficient to determine minimization on the left. A similar result holds on the right.*

Remark. The above represents a sufficient condition, and thus, the values of q_l and q_r obtained from the lemma are upper bounds. An example is given later in this section of a rule with particular tuples $\vec{q_l}$ for which the pseudosets U_0 defined above never contain each other and yet additional arguments establish that the number of preimages is always minimized by appending a 1 on the right.

Corollary 1. *The values of q_l, q_r determined from the above lemma for the 88 distinct nearest-neighbor rules (distinct under symmetries) are given in table 1.*

Proof. The results with finite-valued q_l, q_r are proved using Lemma 3. The case of rule 44 for which $q_l = 0$, $q_r = \infty$ typifies the other cases, and is treated in detail here. Rule 44 is defined by

$$\{000, 001, 100, 110, 111\} \to 0, \qquad \{010, 011, 101\} \to 1. \tag{3.6}$$

To show $q_r = \infty$, it suffices to show that for any finite q_r, there exists a tuple $\vec{q_r}$ of length q_r such that

q_l	q_r	rule numbers
0	0	0 1 2 3 4 5 6 7 8 9 10 12 15 18 22 24 25 26 30 32 33 34 36 37 40 41 45 50 51 60 62 72 73 74 76 90 94 104 105 106 110 122 126 128 130 132 134 136 138 140 146 150 152 154 160 164 170 204
0	1	28 35 56
0	3	13 14 162
0	∞	11 44
1	1	19 23 29 54 57 77 108 156 178 184 200 232
1	3	78
1	∞	27
2	0	168
2	2	43 142
3	0	42
3	1	58
∞	0	38
∞	1	172
∞	∞	46

Table 1: For each rule, the parameters q_l, q_r represent the number of site values on the left and right ends, respectively, of an arbitrary sequence S required to determine exactly the values that, when appended to S, minimize (maximize) the preimages of the extended sequence.

$$y(S0\vec{q_r}) \neq y(S1\vec{q_r}),$$

where S is an arbitrary sequence, and $y(\vec{a})$ is the value that minimizes the number of preimages when appended to the right of the sequence $\vec{a}$. Let $\vec{q_r}$ be the sequence of length q_r consisting of all 1's. Then it is straightforward to show that for any even q_r,

$$\begin{aligned} N(S\vec{q_r}0) &= 3R_{11}^{n-q_r}, \\ N(S\vec{q_r}1) &= 2R_{01}^{n-q_r} + R_{11}^{n-q_r}, \end{aligned}$$

where n is the length of the sequence $S\vec{q_r}$. Further iterations of the recurrence relations (2.2) for rule 44 yield

$$\begin{aligned} N(S0\vec{q_r}0) &= 3R_{11}^{n-q_r-1}, \\ N(S0\vec{q_r}1) &= 2R_{00}^{n-q_r-1} + R_{11}^{n-q_r}, \end{aligned} \tag{3.7}$$

and

$$\begin{aligned} N(S00\vec{q_r}0) &= 3R_{11}^{n-q_r-1}, \\ N(S00\vec{q_r}1) &= 2R_{00}^{n-q_r-1} + 3R_{11}^{n-q_r}, \\ N(S10\vec{q_r}0) &= 3R_{01}^{n-q_r-1}, \\ N(S10\vec{q_r}1) &= R_{01}^{n-q_r-1}. \end{aligned} \tag{3.8}$$

Comparison of the first two shows that preimages are minimized by appending a 0 on the right, whereas comparison of the last two indicates minimization with a 1 on the right. Since the relations hold for all values of q_r even, it follows that $q_r = \infty$. ∎

Theorem 1. *Let α_n^-, α_n^+ be defined as in (3.2). Then, for any nearest-neighbor automaton rule,*

$$\lim_{n\to\infty} 2^6\alpha_n^- = c^-, \qquad \lim_{n\to\infty} 2^6\alpha_n^+ = c^+,$$

where $0 \leq c^- \leq 16$, $16 \leq c^- \leq 64$ are constants depending on the rule.

Remark. The values of c^- for all nearest-neighbor rules are given in table 2.

Proof. First, show that $0 \leq c^- \leq 16$ for all rules. Clearly, α_n^- is nonnegative, and the lower bound of 0 is in fact assumed, for instance, for rules that map all tuples to the same value. α_n^- is maximized for rules such that every spatial sequence has exactly the same number of preimages as any other sequence of the same length (including linear, "toggle," and surjective rules); for these rules, $q_l = q_r = 0$, and $2^6\alpha_n^- = 16$.

Next, show that α_n^- does in fact converge to a constant value. From lemmas 1 through 3 and the corollary to lemma 3, the only cases for which the result needs to be established are those for which either $q_l = \infty$ or $q_r = \infty$; i.e., rules 11, 27, 38, 44, 46, and 172. Two representative cases–rules 11 and 44–will be treated in detail.

c^-	Rule numbers
0	0
1	1 2 4 8 32 128
4	3 5 6 9 10 12 18 19 23 24 33 34 36 40 46 72 77 126 130 132 136 160 178 200 232
7.5	38 44
8.25	13 14 42 58 78 162
8.5	27 172
9	7 11 22 25 26 28 29 35 37 41 43 50 54 56 57 62 73 74 76 94 104 108 110 122 134 138 140 142 146 152 156 164 168 184
16	15 30 45 51 60 90 105 106 150 154 170 204

Table 2: Parameters q_l, q_r from table 1 are used to compute exact values of

$$c^- = \lim_{n\to\infty} 2^6 \alpha_n = 2^{-n+2} \sum_S N^-(S),$$

where the sum is taken over all strings S of length n, and $N^-(S)$ is equal to the minimum number of preimages for the string S with a new value appended to both the left and right ends.

1. Rule 11 is defined by

$$\{010, 100, 101, 110, 111\} \to 0, \quad \{000, 001, 011\} \to 1.$$

Lemma 3 implies that $q_l = 0$. On the right, the recurrence relations are obtained from (1.3) as

$$\begin{aligned} R^j_{00} &= R^{j-1}_{10} I_j(0) + R^{j-1}_{00} I_j(1), \\ R^j_{01} &= R^{j-1}_{10} I_j(0) + R^{j-1}_{00} I_j(1), \\ R^j_{10} &= (R^{j-1}_{01} + R^{j-1}_{11}) I_j(0), \\ R^j_{11} &= R^{j-1}_{11} I_j(0) + R^{j-1}_{01} I_j(1). \end{aligned}$$

It is easy to show that for a sequence of the form $S1$, $N(S10) \geq N(S11)$, and hence for any sequence ending with a 1, the number of preimages is minimized by adding another 1. If, instead, the sequence is of the form $S0$ and is of length n, then

$$\begin{aligned} N(S00) &= 2R^n_{00} + R^n_{10} + 4R^n_{11}, \\ N(S01) &= 3R^n_{00}, \end{aligned}$$

Since the only equivalence (as per lemma 2) is between 00 and 01, it is not obvious that one of the above two terms is less than the other. It is, however, straightforward to show that

$$N(S100) \geq N(S101),$$

and in fact, for any sequence of the form $S1\cdots$, the number of preimages is minimized by adding a 1 on the right. The only case that remains is a sequence S^* consisting of all 0's. In this case, for S^* of length n, the general form of the number of preimages is given by

$$\begin{aligned} N(S^*0) &= R^0_{00} + 2R^0_{10} + [\tfrac{3n}{2} + 2]R^0_{11}, && n \text{ even}, \\ N(S^*1) &= 3R^0_{10} + [\tfrac{3(n-2)}{2} + 2]R^0_{11}, && n \text{ even}, \\ N(S^*0) &= 2R^0_{00} + R^0_{10} + [\tfrac{3(n+1)}{2} + 2]R^0_{11}, && n \text{ odd}, \\ N(S^*1) &= 3R^0_{00} + [\tfrac{3(n-1)}{2} + 2]R^0_{11}, && n \text{ odd}, \end{aligned}$$

and since $R^0_{00} = R^0_{01} = 1$, $R^0_{10} = 2$, $R^0_{11} = 0$, it follows that

$$N(S0) \geq N(S1)$$

for arbitrary sequences S. Hence, for $n \geq 2$,

$$2^6 \alpha^-_n = N(1)N(1) = 1.$$

2. Rule 44 is defined by (3.6). Lemma 3 implies that $q_l = 0$, and specifically, preimages are minimized for any sequence by appending a 1 on the left.

It was shown earlier that $q_r = \infty$, implying that minimization of preimages in general requires complete knowledge of the sequence. Nonetheless, it will be shown here that α_n^- can be computed exactly using only finite-length end tuples and appropriate correction terms. Let $\alpha_n^-(q), 0 \leq q \leq n$, be an approximation to the true value of α_n^- as defined in (3.2) such that

$$\begin{aligned}\alpha_n^-(q) &= 2^{-(n+4)} \sum_{\vec{q}} 2^{n-q-2} N(1) N^r(\vec{q}),\\ &= 2^{-(n+4)} 3 \sum_{\vec{q}} 2^{n-q-2} N^r(\vec{q}),\end{aligned}$$

where $\vec{q}$ ranges over all tuples of length q, and $N^r(\vec{q})$ denotes the minimum number of preimages for the sequence $\vec{q}$ with a new value appended on the right. Since $q_r = \infty$, $\alpha_n^-(q)$ cannot represent the exact value of α_n^-, and

$$\alpha_n^-(q+1) = \alpha_n^-(q) + \delta(q),$$

where $\delta(q) < 0$ for all q. It can be shown by iterating recurrence relations (2.2) for rule 44 that, for $q \geq 2$, $\delta(q)$ represents the q^{th} level error in estimating exactly one term, namely $N^r(011\cdots1)$. Formally, using $\sum'$ to denote summation over all tuples $\vec{q}$ not equal to $011\cdots1$,

$$\begin{aligned}\alpha_n^-(q+1) &= 2^{-(n+4)} 3 [\sum_{\vec{q}}{}' N^r(\vec{q}) 2^{n-q-2} + (N^r(001\cdots1)\\ &+ N^r(101\cdots1)) 2^{n-q-3}],\\ &= \alpha_n^-(q) - 3 \cdot 2^{-4} [2N^r(01\cdots1) - N^r(001\cdots1)\\ &- N^r(101\cdots1)] 2^{-q-3},\\ &= \alpha_n^-(q) + \delta(q).\end{aligned}$$

To find $\delta(q)$, note that equations (3.7) imply that for the sequence $\vec{q}$ consisting of a 0 followed by an even number of 1's,

$$\begin{aligned}N(\vec{q}0) = N(011\cdots10) &= 3R_{11}^0 = 3,\\ N(\vec{q}1) = N(011\cdots11) &= 2R_{01}^0 + R_{11}^0 = 3,\end{aligned}$$

and equations (3.8) imply

$$\begin{aligned}N(0\vec{q}0) &= N(0011\cdots10) &&= 3R_{11}^0 = 3,\\ N(0\vec{q}1) &= N(0011\cdots11) &&= 2R_{00}^0 + R_{11}^0 = 5,\\ N(1\vec{q}0) &= N(1011\cdots10) &&= 3R_{11}^0 = 3,\\ N(1\vec{q}1) &= N(1011\cdots11) &&= 2R_{00}^0 + R_{11}^0 = 1.\end{aligned}$$

Hence,

$$2N^r(01\cdots1) - N^r(001\cdots1) - N^r(101\cdots1) = 2,$$

and since the same result holds for sequences with an odd number of 1's,

$$\delta(q) = -3 \cdot 2^{-4} 2^{-q-2}.$$

Therefore, since

$$\lim_{q\to\infty} \alpha_n^-(q) = \alpha_n^-(2) + \delta(2) + \delta(3) + \cdots,$$

and

$$\begin{aligned}\alpha_n^-(2) &= 2^{-8}3[N^r(00) + N^r(01) + N^r(10) + N^r(11)], \\ &= 2^{-6}9,\end{aligned}$$

it follows that

$$\begin{aligned}\lim_{q\to\infty} 2^6\alpha_n^-(q) &= 9 - 3[\tfrac{1}{4} + \tfrac{1}{8} + \cdots], \\ &= 7.5,\end{aligned}$$

for n sufficiently large.

Finally, the limiting behavior of α_n^+ follows directly from the observation that equation (3.3) can be modified to give

$$2^6\alpha_n^+ = 2^{-q_l-q_r} \sum_{\vec{q_l},\vec{q_r}} [2N(\vec{q_l}) - N(x(\vec{q_l})\vec{q_l})] \cdot [2N(\vec{q_r}) - N(\vec{q_r}y(\vec{q_r}))],$$

where all terms are defined as before. ∎

Examples illustrating the computation of α_n^+ are given in [8]. For additive and other surjective cellular automata rules, the values of c^- and c^+ (respectively, the constants for the scaling laws governing the minimal and maximal preimages) are clearly equal. The opposite extreme is represented by rules that map all tuples to the same value (e.g., rules 0 and 255), for which $c^+ = 64$ and $c^- = 0$. In general, the difference between c^- and c^+ can be interpreted [8] as reflecting the extent to which the rule changes the probability distribution associated with the automaton's spatial sequences in one iteration.

4. Summary

This paper is concerned with preimages for one-dimensional cellular automaton rules on infinite lattices. For a given rule and arbitrary spatial sequence of values, the preimage of the sequence is defined to be the set of tuples that are mapped by the rule onto the sequence. The number of preimages of a sequence can be interpreted as determining the a priori probability of occurrence of the sequence after one iteration of the rule applied to an initial condition with uniform measure.

Recurrence relations are presented here for finding the number of preimages of general spatial sequences. These relations group and count preimages according to their endtuples, and then, for any sequence, express the number of its preimages beginning (either on the left or right) with a particular endtuple in terms of the number of preimages beginning with other endtuples for its subsequences.

These recurrence relations are then applied to the analysis of a quantity defined as the number of preimages for spatial sequences whose probabilities are in a certain sense minimal (maximal) under one iteration of the rule. In particular, the recurrence relations are used to prove that, for all nearest-neighbor automata rules, this quantity scales exactly with the length of the spatial sequence.

The proof of the exact scaling behavior relies on the observation that for general automaton rules and arbitrary spatial sequences, certain features of the preimages for these sequences depend only on the sequences' end values. Specifically, each nearest-neighbor cellular automaton may be characterized by parameters q_l, q_r representing the length of the endtuples of an arbitrary sequence needed to determine what values, when added to the ends of the sequence, minimize (maximize) the number of preimages of the new sequence. For a few rules, the parameters q_l, q_r are shown to be infinite; surprisingly, however, the bulk of nearest-neighbor rules are characterized by $q_l, q_r \leq 3$.

The size of the parameters q_l, q_r is determined by the detailed structure of the automaton rule. In particular, the parameters are equal to the smallest tuple length such that a "containment" condition is satisfied for the rule. The left-hand condition, for example, requires that for all tuples $\vec{q}$ of length q_l, the set T_0 containing the two right components of the tuples mapping to $0\vec{q}$ must either contain, or be contained in, the set T_1 containing the two right components of the tuples mapping to $1\vec{q}$, and similarly on the right.

The implications of the scaling behavior described in this paper are discussed in [8].

Acknowledgements

This problem was suggested by Howard Gutowitz on the basis of numerical evidence of exact scaling behavior. The work was supported in part by the Applied Mathematics Program of the Department of Energy under Contract No. KC-07-01-01.

References

[1] S. Wolfram, "Universality and Complexity in Cellular Automata," *Physica*, **10D** (1984) 1.

[2] J. Milnor, "Notes on Surjective Cellular Automaton-Maps," Institute for Advanced Study preprint (June 1984).

[3] A. Rouet, "A Mathematical Classification of One-Dimensional Cellular Automata" (preprint).

[4] K. Sutner, "On σ-automata," (preprint).

[5] S. Amoroso and G. Cooper, "The Garden-of-Eden theorem for Finite Configurations," *Proceedings of the American Mathematical Society*, **26** (1970) 158.

[6] S. Skyum, "Confusion in the Garden of Eden," *Proceedings of the American Math Society*, **50** (1975) 332.

[7] K. Culik and Y. Sheng, "On the Limit Sets of Cellular Automata," (University of Waterloo Research Report CS-87-47).

[8] H. Gutowitz and E. Jen, "Approximate Factorization of Block Preimages under Cellular Automata," (in preparation).

[9] E. Jen, "Cylindrical Cellular Automata," Los Alamos National Laboratory Report No. LAUR-87-353.

[10] E. Jen, "Enumeration of Preimages in Cellular Automata," (preprint).