Information Sciences 199 (2012) 125–132

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Invertible behavior in elementary cellular automata with memory

Juan C. Seck-Tuoh-Mora [a,*], Genaro J. Martínez [b,c], Ramon Alonso-Sanz [d], Norberto Hernández-Romero [a]

[a] Centro de Investigacion Avanzada en Ingenieria Industrial, Universidad Autonoma del Estado de Hidalgo, Hidalgo, Mexico
[b] Instituto de Ciencias Nucleares, Centro de Ciencias de la Complejidad, Universidad Nacional Autonoma de Mexico, Mexico
[c] Unconventional Computing Center, University of the West of England, United Kingdom
[d] ETSI Agrónomos, Polytechnic University of Madrid, Madrid, Spain

ARTICLE INFO

Article history:
Received 22 February 2011
Received in revised form 11 February 2012
Accepted 29 February 2012
Available online 8 March 2012

Keywords:
Reversibility
Memory
Elementary cellular automata
De Bruijn block

ABSTRACT

Elementary cellular automata (ECAs) have been studied for their ability to generate complex global behavior, despite their simplicity. One variation of ECAs is obtained by adding memory to each cell in a neighborhood. This process generates a provisional configuration in which the application of an evolution rule establishes the dynamics of the system. This version is known as an ECA with memory (ECAM). Most previous work on ECAMs analyzed the complex behavior taking chaotic ECAs. However, the present paper investigates reversible ECAMs as obtained from reversible and permutative ECAs. These ECAs have at least one ancestor for every configuration; thus, the correct permutation of states may specify the memory function to obtain reversible ECAMs. For permutative ECAs, which are often irreversible, we demonstrate that the use of a quiescent state and the correct manipulation of de Bruijn blocks produce reversible ECAMs.

1. Introduction

Elementary cellular automata (ECAs) are dynamical systems that are discrete in time, space, and states. An ECAM is defined by a one-dimensional array of cells. There are two possible states per cell; each cell updates its state according to an evolution rule that considers the current state of the cell and its two immediate neighbors.

ECAs have been studied in detail because, in some cases, they are able to produce chaotic and complex behavior without requiring a centralized control mechanism [33,32,14,23]. Thus, ECAs are useful in various applications, for instance, in unconventional universal systems or random number generators [31,9,19].

One variation of ECAs is an ECA that includes memory (ECAM). In this variant, every cell reviews its past history (if any) to derive a provisional state (this process specifies a provisional configuration); and thereafter, the application of an evolution rule results in the evolution of the automaton.

ECAMs are able to exhibit complex behavior from chaotic ECAs; therefore, it is possible to implement unconventional computing devices using ECAMs [3,6,18,20]. Nevertheless, there are other important aspects to the extension of ECAs to ECAMs; in particular, reversibility. This property is one of the main topics of interest in cellular automata research; in part, to allow better understanding of the conservation of information in discrete dynamical systems. For instance, McIntosh adopts second-order and partitioning techniques to achieve reversibility with ECAs, and de Bruijn diagrams to analyze this behavior [21,22]. Some studies of reversibility in ECAMs were performed by Alonso-Sanz [1,2], who applied an invertible XOR boolean operator to produce reversible ECAMs.

* Corresponding author.
E-mail addresses: jseck@uaeh.edu.mx (J.C. Seck-Tuoh-Mora), genaro.juarez.martinez@googlemail.com (G.J. Martínez), ramon.alonso@upm.es (R. Alonso-Sanz), nhromero@uaeh.edu.mx (N. Hernández-Romero).

0020-0255/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2012.02.063

To our knowledge, however, there is no explanation of how the local properties of reversible ECAs could be extended to obtain reversible ECAMs. In addition, there is no general method of finding the appropriate memory function in ECAMs so as to obtain reversible behavior from irreversible ECAs.

The present article is intended to provide solutions to the previously discussed issues. In particular, this paper employs the inverse rule of reversible ECAs and the local properties of permutative ECAs to characterize the memory function needed to obtain invertible ECAMs.

For an ECAM of memory size n, and given its last n configurations c^{t-n+1}, c^{t-n+2}, ..., c^t, one can calculate the ancestor of c^t and use the remaining $n-1$ configurations to obtain the states in c^{t-n}. For reversible ECAs, the correct permutation of states used as memory achieves the desired behavior. Additional components are considered in the case of permutative ECAs, specifically, a quiescent state, de Bruijn blocks, and the uniform number of ancestors established by Hedlund in [13].

This paper is organized as follows: Section 2 presents the basic concept of an ECAM. Section 3 characterizes invertible ECAMs based on reversible ECAs. Section 4 applies these results to the case of permutative ECAs. Section 5 gives illustrative examples based on the previous results. The final section provides some concluding remarks.

2. Basic concepts

A one-dimensional cellular automaton is an array of locally connected cells. Initially, every cell has a state taken from a finite set of states. The cells update their states simultaneously by the same evolution rule in a discrete time. The rule takes as its argument the current state of every cell and its nearest neighbors. Thus, cellular automata are discrete in time, space, and states. There are several problems that are being studied in one-dimensional cellular automata to understand and take advantage of their complex behavior, for instance, replication of structures [12], fluctuations in the evolution of number-conserving cellular automata [15], simulation of universal systems by means of reversible cellular automata [24], and applications to the optimization of functions [29].

Elementary cellular automata (ECAs) have previously been studied for their ability to produce complex global behavior, in spite of their simplicity [31]. An ECA is defined by a binary set of states $\Sigma = \{0,1\}$ and a mapping φ: $\Sigma^3 \to \Sigma$ (therefore, each sequence in Σ^3 is a neighborhood of size 3), and φ is the evolution rule of the ECA. The dynamics of the ECA begins at a given initial condition $c^1 : \mathbb{Z} \to \Sigma$, where the superscript indicates time, c_i^1 is the i-th cell of c^1, and $\eta(c_i^1) = (c_{i-1}^1, c_i^1, c_{i+1}^1)$ constitutes the neighborhood of c_i^1.

The update of every cell $c_i^2 = \varphi \circ \eta(c_i^1)$ generates a new configuration c^2; the $\circ$symbol indicates the composition of both functions, first applying η and then φ. This process is indefinitely repeated, constituting the evolution of the ECA.

Building on this basic definition, several variants have been also created so as to realize new behaviors, for example, a larger number of states [11,28], different neighborhood sizes [8,17], the application of continuous states [25,26], inhomogeneous neighborhoods and evolution rules [10,16], or (as in the case of this paper) a memory function that evaluates the history of every cell in a neighborhood [3,30]. Memory has been used in other automata models to analyze complex behavior [4,5].

In cellular automata with memory (ECAMs), every cell uses its past states to determine a provisional state. Thereafter, an evolution rule is applied to the new array of provisional states to obtain the subsequent configuration. This paper proposes a characterization of the memory needed to conserve and produce reversible behavior in ECAMs.

For a given ECA and some $n \in \mathbb{Z}^+$, an ECA of memory size n (or ECAM-n) is defined as follows: For every cell c_i^t in configuration c^t, let $\gamma_n(c_i^t) = (c_i^{t-n+1}, \ldots, c_i^t)$ be the sequence composed by the current and the past $n-1$ states of c_i^t. Let us take β_n: $\Sigma^n \to \Sigma$. Additionally, let us define a memory function in Eq. (1) that produces a state based on the history of c_i^t.

$$\tau_n(c_i^t) = \begin{cases} c_i^t & \text{if} \quad t < n \\ \beta_n \circ \gamma_n(c_i^t) & \text{if} \quad t \geqslant n \end{cases} \tag{1}$$

For $t < n$, the memory function in Eq. (1) simply takes the current state of c_i^t. Consequently, with this memory, an ECAM yields the same behavior as that of the original ECA for the first $n-1$ evolutions. In another case, the mapping β_n is applied when there are enough evolutions. The choice of β_n is not restricted, provided that it is specified for the set of sequences of length n. β_n is preserved throughout the evolution of the ECAM-n. Let us notice that every τ_n generates a temporal configuration d^t between c^t and c^{t+1}. In this way, Eq. (2) defines the states of the subsequent configuration.

$$c_i^{t+1} = \varphi \circ \eta\left(d_i^t\right) \tag{2}$$

This study considers the mappings β_n required to produce memory functions τ_n that yield reversible ECAMs based on ECAs. The invertible behavior to be analyzed is defined in the following way.

Definition 1. An ECAM-n is invertible if the configuration c^{t-n} is obtained in a unique way from the configurations c^{t-n+1}, ..., c^t.

Definition 1 implies (at least) that the last n configurations are known to the ECAM-n. Therefore, we propose the following process to re-trace the evolution of an ECAM.

1. Calculate the ancestor of the configuration c^t.
2. Define the states in c^{t-n} in a unique way from this ancestor.

The ECAMs to be analyzed are extensions of reversible and permutative ECAs. The process is based on the well-known properties of the ancestors in this kind of automaton.

3. Invertible memory in reversible ECAs

A reversible ECA (RECA) exhibits invertible behavior, which is induced by an inverse local mapping. RECAs are defined as follows:

Definition 2. An ECA is reversible iff for φ, there exists another evolution rule φ^{-1} such that $c_i^{t-1} = \varphi^{-1} \circ \eta(c_i^t)$.

There are 256 ECAs, and each ECA can be indicated using Wolfram code [31]. In other words, taking the decimal number associated with the binary sequence delineated by the evolution rule. The most significant bit is the one that is related to the evolution of the neighborhood 111.

From these 256 ECAs, there are only six ECAs that fulfill Definition 2 (rules 15, 51, 85, 170, 204, and 240). These rules take an element at the same position in every neighborhood, and retain the same value or change (based on the evolution rule). For instance, rule 15 performs the negation of the leftmost state of every neighborhood (see Table 1). In this way, each rule and its inverse are paired as: $15 \leftrightarrow 85$, $51 \leftrightarrow 51$, $170 \leftrightarrow 240$, and $204 \leftrightarrow 204$ (see page 436 of [31]).

Given an ECAM-n, the inverse rule φ^{-1} can be applied to c^t to determine d^{t-1}. Let $\mathcal{P}$ be the set of permutations of Σ. For each $w \in \Sigma^{n-1}$, let us define $\mathcal{A}_w = \{\beta_n(aw) : a \in \Sigma\}$. The definition of $\mathcal{A}_w$ implies $|\mathcal{A}_w| = 1$ or $|\mathcal{A}_w| = 2$ because the domain of $\mathcal{A}_w$ is $\Sigma = \{0,1\}$. As a consequence, if $\beta_n(0w) = \beta_n(1w)$ then $|\mathcal{A}_w| = 1$; otherwise, $|\mathcal{A}_w| = 2$ and $\mathcal{A}_w$ is a permutation of Σ. These elements are useful to prove the following result.

Proposition 1. *Let an ECAM-n be an extension of an RECA such that it includes memory. Then, the ECAM-n is reversible iff every* $\mathcal{A}_w \in \mathcal{P}$.

Proof. For $t < n$, the ECAM-n has identical behavior to that of the original ECA, because the memory function τ_n is only copying c^t in d^t. Therefore, reversible behavior is (trivially) obtained applying φ^{-1} to c^t to produce $d^{t-1} = c^{t-1}$. In this way, the interesting cases are those in which there are $t \geqslant n$ evolutions.

According to Eq. (1), for $t \geqslant n$, τ_n can be described as in Eq. (3). Each row index is a state $a \in \Sigma$ and every column index is a sequence $w \in \Sigma^{n-1}$. Thus, $\beta_n(aw)$ defines the entry (a,w), and the column w represents the set $\mathcal{A}_w$.

$$\begin{array}{c|cccc} & w_1 & w_2 & \cdots & w_{2^{(n-1)}} \\ \hline 0 & \beta_n(0w_1) & \beta_n(0w_2) & \cdots & \beta_n(0w_{2^{n-1}}) \\ 1 & \beta_n(1w_1) & \beta_n(1w_2) & \cdots & \beta_n(1w_{2^{n-1}}) \end{array} \tag{3}$$

$\Rightarrow$) Suppose that the ECAM-n is reversible; in this case, each c_i^{t-n} can be obtained in a unique way from $(c_i^{t-n+1}, \ldots, c_i^t)$. Let us take $a = c_i^{t-n}$, $w = c_i^{t-n+1} c_i^{t-n+2} \cdots c_i^{t-1}$, and suppose that $\mathcal{A}_w \notin \mathcal{P}$. As a result, $\mathcal{A}_w$ is not a permutation of Σ. Therefore, all rows of column w are identical, because $|\Sigma| = 2$ and $|\mathcal{A}_w| = 1$, as shown in Eq. (4).

$$\begin{array}{c|ccc} & \cdots & w & \cdots \\ \hline 0 & \cdots & b & \cdots \\ 1 & \cdots & b & \cdots \end{array} \quad \text{for } b \in \Sigma \tag{4}$$

Moreover, $\varphi^{-1} \circ \eta(c_i^t) = b = d_i^{t-1}$. However, Eq. (4) indicates that $\beta_n(0w) = \beta_n(1w) = d_i^{t-1}$. Therefore, c_i^{t-n} may be either 0 or 1, which contradicts the reversibility of the ECAM-n.

$\Leftarrow$) Suppose that $\mathcal{A}_w \in \mathcal{P}$ for every $w \in \Sigma^{n-1}$. Then, each column w is a permutation of Σ, as shown in Eq. (5).

Table 1
Evolution rules of the six RECAs.

Neighborhood	15	51	85	170	204	240
000	1	1	1	0	0	0
001	1	1	0	1	0	0
010	1	0	1	0	1	0
011	1	0	0	1	1	0
100	0	1	1	0	0	1
101	0	1	0	1	0	1
110	0	0	1	0	1	1
111	0	0	0	1	1	1

$$\begin{array}{c|ccc} & \cdots & w & \cdots \\ \hline 0 & \cdots & a & \cdots \\ 1 & \cdots & b & \cdots \end{array} \quad \text{for } \{a,b\} = \Sigma \tag{5}$$

For each c_i^t, let us take $w = c_i^{t-n+1}c_i^{t-n+2}\cdots c_i^{t-1}$ and $d_i^{t-1} = \varphi^{-1}\circ\eta(c_i^t)$. Then, as the column w in Eq. (5) is a permutation of Σ, there is a unique row $x\in\Sigma$ in this column such that $\beta_n(xw) = d_i^{t-1}$. Thus $c_i^{t-n} = x$; therefore, the ECAM-n is reversible. □

4. Reversible memory for permutative ECAs

This section analyzes other types of irreversible ECAs that might exhibit reversible behavior when used as ECAMs. In particular, permutative ECAs (PECAs) are discussed, because they are sometimes irreversible; however, their surjective properties are useful to yield reversible ECAMs.

To better interpret the following results, we will consider the simulation of every ECA by another cellular automaton with 4 states and a neighborhood size of 2 [7,27]. For $m\in\mathbb{Z}^+$ and any sequence $w\in\Sigma^m$, let $w_{[i]}$ be the i-th state of w. For $1\leqslant i\leqslant j\leqslant m$, let $w_{[i,j]}$ be the sequence $w_{[i]}w_{[i+1]}\ldots w_{[j]}$ in w from position i to position j. For $m\geqslant 3$ and any $w\in\Sigma^m$, let us define $\phi(w)$ as the sequence in Σ^{m-2} generated by the concatenation of $\varphi(w_{[1,3]})\varphi(w_{[2,4]})\cdots\varphi(w_{[m-2,m]})$. That is to say, $\phi(w)$ is the application of the evolution rule to each neighborhood of w without being required to have periodic boundary conditions. Accordingly, for any two sequences $u,v\in\Sigma^2$ where uv is the usual concatenation of sequences, we know that $\phi(uv) = w\in\Sigma^2$.

Let M be the square matrix in which row and column indices are the elements of Σ^2, such that every entry of M is defined in Eq. (6).

$$M(u,v) = w \text{ iff } \phi(uv) = w \tag{6}$$

The sequences of Σ^2 in M are just the de Bruijn blocks normally used to represent the evolution rule graphically [22,23].

For $Q = \{0,1,2,3\}$, let us define a bijection ς: $\Sigma^2\to Q$ such that ς: $00\to 0$, ς: $01\to 1$, ς: $10\to 2$, and ς: $11\to 3$. With this bijection,[1] the following definition is given.

Definition 3. em An ECA is permutative (PECA) if each row in its corresponding matrix M is a permutation of Q.

Notice that Definition 3 takes the bijection ς to implicitly represent the elements of M.

In every PECA, for $m\in\mathbb{Z}^+$ and each sequence $w\in Q^m$, there exists a set $\Lambda_w = \{v\colon v\in Q^{m+1}, \phi(v) = w\}$ such that $|\Lambda_w| = 4$ [13]. In other words, every sequence of length m has 4 different ancestors of length $m+1$.

In this case, for any $x\in Q$ and each $w\in Q^m$, every sequence in Λ_{wx} is an extension of a unique sequence in Λ_w. This is because every row in M is a permutation of Q; therefore, for each $v\in\Lambda_w$ with rightmost state $v_{[m+1]} = a\in Q$, there exists a unique $b\in Q$ such that $\varphi(ab) = x$. Therefore, $vb\in\Lambda_{wx}$.

Thus, the ancestors of a given sequence may have different states at the same position. Consequently, the previous property usually avoids the specification of an inverse evolution rule. In particular, PECA are examples of surjective automata [21]. However, this property can be used to redefine the memory function τ_n in Eq. (1) and obtain reversible behavior in any PECA.

First, a PECA is transformed into an ECAM where all configurations are represented between semi-infinite sequences of quiescent states #. For any $a\in Q$, let us define a state # such that $\varphi(a\#) = \varphi(\#a) = \varphi(\#\#) = \#$. State # is called 'quiescent' because all its neighborhoods evolve into #. For $m\in\mathbb{Z}^+$ and $1\leqslant i\leqslant m$, every initial configuration $c = c_1\cdots c_m$ is represented as $\mathcal{C} = \cdots\#\#c\#\#\cdots$ with $c_i\in Q$. That is to say, c is finite, and the states of c are not quiescent. For $i\in\mathbb{Z}$, let $\mathcal{C}_i$ be the i-th element of $\mathcal{C}$; thus, $\mathcal{C}_1\cdots\mathcal{C}_m = c$. The use of the quiescent state avoids the need to specify periodic boundary conditions in the evolution of the ECAM.

Let us take $\gamma_n(\mathcal{C}_i^t) = (\mathcal{C}_i^{t-n+1},\ldots,\mathcal{C}_i^t)$. Then, $\gamma_n(\mathcal{C}_i^t)$ gives the current and the previous $n-1$ states of $\mathcal{C}_i$. For $n\in\mathbb{Z}^+$, let us redefine β_n as follows:

$$\beta_n : Q^n \to \Lambda_a \text{ for some } a\in Q \tag{7}$$

In Eq. (7), β_n maps every sequence of states of length n onto a complete set of ancestors of $a\in Q$. The selection of a is unrestricted. For $x,y\in Q$ and $w = xy$, let us define $\alpha(w) = x$ and $\omega(w) = y$. With these definitions, the memory function τ_n is reformulated as Eq. (8), specifying a reversible ECAM-n for any PECA.

$$\tau_n(\mathcal{C}_i^t) = \begin{cases} \# & \text{if } \mathcal{C}_{i-1}^t = \mathcal{C}_i^t = \# \\ w\in\Lambda_{\mathcal{C}_i^t} & \text{if } t<n \text{ and } w\approx\tau_n(\mathcal{C}_{i-1}^t) \\ w\in(\beta_n\circ\gamma_n(\mathcal{C}_i^t)) & \text{if } t\geqslant n \text{ and } w\approx\tau_n(\mathcal{C}_{i-1}^t) \end{cases} \tag{8}$$

where, for $v = \tau_n(\mathcal{C}_{i-1}^t)$, $w\approx v$ means that $v = \#$ or $\omega(v) = \alpha(w)$. For $1\leqslant i\leqslant m$ and $t<n$, the memory function τ_n maps the state in $\mathcal{C}_i^t$ onto an ancestor w of the same state. For $t\geqslant n$, τ_n takes an ancestor w in the set specified by β_n. For the cell $\mathcal{C}_1^t$, the

[1] Of course, any other bijection can be used.

$$
\begin{array}{l}
\mathcal{C}^1 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{C}^1_1}\boxed{\mathcal{C}^1_2}\cdots\boxed{\mathcal{C}^1_m}\boxed{\#}\boxed{\#}\boxed{\#}\cdots \\
\mathcal{D}^1 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{D}^1_1}\boxed{\mathcal{D}^1_2}\cdots\boxed{\mathcal{D}^1_m}\boxed{\mathcal{D}^1_{m+1}}\boxed{\#}\boxed{\#}\cdots \\
\quad\mathcal{C}^2 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{C}^2_1}\boxed{\mathcal{C}^2_2}\cdots\boxed{\mathcal{C}^2_m}\boxed{\#}\boxed{\#}\boxed{\#}\cdots \\
\quad\mathcal{D}^2 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{D}^2_1}\boxed{\mathcal{D}^2_2}\cdots\boxed{\mathcal{D}^2_m}\boxed{\mathcal{D}^2_{m+1}}\boxed{\#}\boxed{\#}\cdots \\
\qquad\mathcal{C}^3 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{C}^3_1}\boxed{\mathcal{C}^3_2}\cdots\boxed{\mathcal{C}^3_m}\boxed{\#}\boxed{\#}\boxed{\#}\cdots \\
\qquad\mathcal{D}^3 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{D}^3_1}\boxed{\mathcal{D}^3_2}\cdots\boxed{\mathcal{D}^3_m}\boxed{\mathcal{D}^3_{m+1}}\boxed{\#}\boxed{\#}\cdots \\
\qquad\quad\mathcal{C}^4 = \cdots \boxed{\#}\boxed{\#}\boxed{\mathcal{C}^4_1}\boxed{\mathcal{C}^4_2}\cdots\boxed{\mathcal{C}^4_m}\boxed{\#}\boxed{\#}\boxed{\#}\cdots \\
\qquad\quad\vdots \qquad\qquad\qquad \vdots \qquad\qquad\qquad\qquad \vdots
\end{array}
$$

Fig. 1. Evolution of an ECAM-n from a PECA. For $1 \leqslant i \leqslant m$, $\mathcal{D}^t_i\mathcal{D}^t_{i+1} \in \Lambda_{\mathcal{C}^t_i}$ if $t < n$; and $\mathcal{D}^t_i\mathcal{D}^t_{i+1} \in (\beta_n \circ \gamma_n(\mathcal{C}^t_i))$ if $t \geqslant n$.

choice of w is unrestricted. In contrast, from the second to the m-th cell, the initial state of w must be the same as the one that the final state of the sequence v produced by τ_n in $\mathcal{C}^t_{i-1}$. This is always possible, because all the rows of M are permutations of Q. Therefore, for any $v = \tau_n(\mathcal{C}^t_{i-1})$, the row in M corresponding to the state $\omega(v)$ is a permutation of Q. Thus, there is a unique selection of w such that $\omega(v) = \alpha(w)$.

The application of τ_n to $\mathcal{C}^t$ yields a temporal configuration $\mathcal{D}^t = \cdots\#\#\mathcal{D}^t_1\ldots\mathcal{D}^t_{m+1}\#\#\cdots$ where $\mathcal{D}_i \in Q$ for $1 \leqslant i \leqslant m+1$. With regard to the quiescent state, notice that $\tau_n(\#)$ is not defined if it is on the right of a non-quiescent state. This is not a problem because any $\tau(\mathcal{C}^t_i)$ produces a block $w \in Q^2$. As a result, every position in D^t has a state of Q or a quiescent state.

For $t \geqslant n$, τ_n can be represented as a table analogous to the one in Eq. (3). The row indices of this table are the states in Q. The column indices are the sequences in Q^{n-1}. For each $x \in Q$ and every $w \in Q^{n-1}$, $\beta_n(xw)$ describes the entry (x, w). τ_n is illustrated in Eq. (9) for $j = 4^{n-1}$, $w_i \in Q^{n-1}$, and $t \geqslant n$.

$$
\begin{array}{c|cccccc}
 & w_1 & w_2 & \cdots & w_i & \cdots & w_j \\
\hline
0 & \beta_n(0w_1) & \beta_n(0w_2) & \cdots & \beta_n(0w_i) & \cdots & \beta_n(0w_j) \\
1 & \beta_n(1w_1) & \beta_n(1w_2) & \cdots & \beta_n(1w_i) & \cdots & \beta_n(1w_j) \\
2 & \beta_n(2w_1) & \beta_n(2w_2) & \cdots & \beta_n(2w_i) & \cdots & \beta_n(2w_j) \\
3 & \beta_n(3w_1) & \beta_n(3w_2) & \cdots & \beta_n(3w_i) & \cdots & \beta_n(3w_j)
\end{array} \tag{9}
$$

Fig. 1 describes the generic evolution of an ECAM-n related to a PECA.

As a consequence, the following result can be stated:

Proposition 2. *An ECAM-n associated with a PECA is reversible iff for any $x, y \in Q$ with $x \neq y$ and for each column w in Eq. (9), $\beta_n(xw) \neq \beta_n(yw)$ is also satisfied.*

Proof. This proof is analogous to Proposition 1. For $t < n$ and $1 \leqslant i \leqslant m$, the memory function of the ECAM-n simply takes an ancestor of the same $\mathcal{C}^t_i$. Thus, $\mathcal{C}^{t+1} = \mathcal{C}^t$, and therefore, obtaining reversible behavior is trivial in this case. The interesting cases are those in which $t \geqslant n$.

$\Rightarrow$) Suppose that the ECAM-n is reversible and there exists a column w in Eq. (9) such that $\beta_n(xw) = \beta_n(yw)$. Let us take a configuration $\mathcal{C}^t$ with $\Lambda_{\mathcal{C}^t_i} = \beta_n(xw)$. Such a configuration exists because the ECAM-n is reversible. This implies that $\mathcal{C}^{t-n}_i$ can be defined by either x or y, which is a contradiction.

$\Leftarrow$) For $x \neq y$, suppose that each column w in Eq. (9) satisfies $\beta_n(xw) \neq \beta_n(yw)$. There are $|Q|$ rows in Eq. (9), and there are $|Q|$ different Λ sets as well; thus, for every $\mathcal{C}^t_i$, the set $\Lambda_{\mathcal{C}^t_i}$ only appears in one entry of each column w. Therefore, $\mathcal{C}^{t-n}_i$ is uniquely defined, which means that the ECAM-n is reversible. □

5. Examples

To illustrate Proposition 1, consider an ECAM-3 that is an extension of RECA rule 15. The functions β_3 and τ_3 for $t \geqslant 3$ are presented in Eq. (10).[2]

$$
\begin{array}{ll}
\beta_3(000) \to 0 & \beta_3(100) \to 1 \\
\beta_3(001) \to 1 & \beta_3(101) \to 0 \\
\beta_3(010) \to 1 & \beta_3(110) \to 0 \\
\beta_3(011) \to 0 & \beta_3(111) \to 1
\end{array}
\Rightarrow \tau_3 :
\begin{array}{c|cccc}
 & 00 & 01 & 10 & 11 \\
\hline
0 & 0 & 1 & 1 & 0 \\
1 & 1 & 0 & 0 & 1
\end{array} \tag{10}
$$

Eq. (10) specifies τ_3 in such a way that each column is a permutation of Σ. This kind of memory is shown in Fig. 2, which depicts the evolution of the ECAM-3 taking an initial configuration of 12 cells and periodic boundary conditions. In this ECAM, black squares are cells in state 1 and white squares are cells in state 0.

[2] In particular, τ_3 is the parity rule.

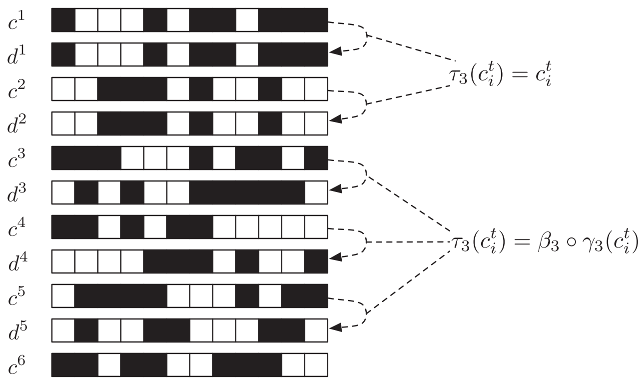

Fig. 2. Evolution of the ECAM-3. The memory function is the identity for the first two steps. From the third step onwards, the memory function is the composition of β_3 and γ_3 according to Eq. (1).

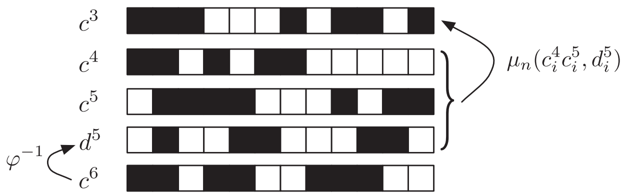

Fig. 3. Inverse evolution of the ECAM-3 used to obtain c^3.

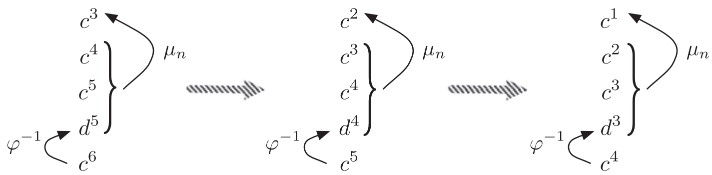

Fig. 4. Representation of the inverse evolution in the ECAM-3.

Fig. 3 presents the inverse evolution used to obtain c^3. Here, the application of the inverse rule[3] to c^6 produces d^5. Next, for c^5 and c^4, configuration c^3 is uniquely defined by the permutations of Σ in each column of Eq. (10). For instance, values $d_3^5 = 0,\ c_3^5 = 1$, and $c_3^4 = 0$ are associated with a unique entry $(a, 01) = 0$ in Eq. (10); therefore, $c_3^3 = 1$.

Fig. 4 describes the complete backwards evolution.

To illustrate Proposition 2, PECA rule 165 is presented in Eq. (11). The equation also shows the associated matrix M (with entries in Σ^2 and Q) applying the bijection ς defined in Section 4.

Neigh.	165
000	1
001	0
010	1
011	0
100	0
101	1
110	0
111	1

$$M_{165} = \begin{array}{c|cccc} & 00 & 01 & 10 & 11 \\ \hline 00 & 11 & 10 & 01 & 00 \\ 01 & 10 & 11 & 00 & 01 \\ 10 & 01 & 00 & 11 & 10 \\ 11 & 00 & 01 & 10 & 11 \end{array} \quad \rightarrow \varsigma \rightarrow \quad \begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 3 & 2 & 1 & 0 \\ 1 & 2 & 3 & 0 & 1 \\ 2 & 1 & 0 & 3 & 2 \\ 3 & 0 & 1 & 2 & 3 \end{array} \tag{11}$$

The irreversibility of this PECA can be easily demonstrated by evaluating periodic boundary conditions. For each $m \geqslant 2$, the sequence 3^m has multiple ancestors, because it is generated by 0^m, 1^m, 2^m, and 3^m (Fig. 5).

Consider the matrix M_{165} that exemplifies Proposition 2. Then, Eq. (12) presents the associated Λ_a sets for each $a \in Q$.

[3] In this case, ECA rule 85.

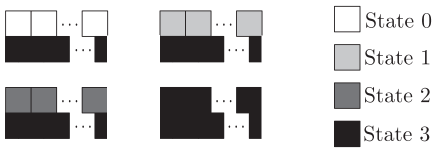

Fig. 5. Irreversibility of PECA rule 165 with four different ancestors for the sequence 3^m.

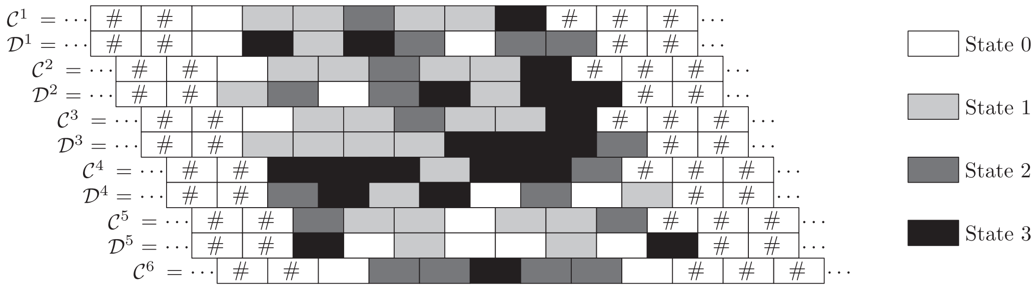

Fig. 6. 5-step evolution of the ECAM-3 from the initial configuration 0 1 1 2 1 1 3.

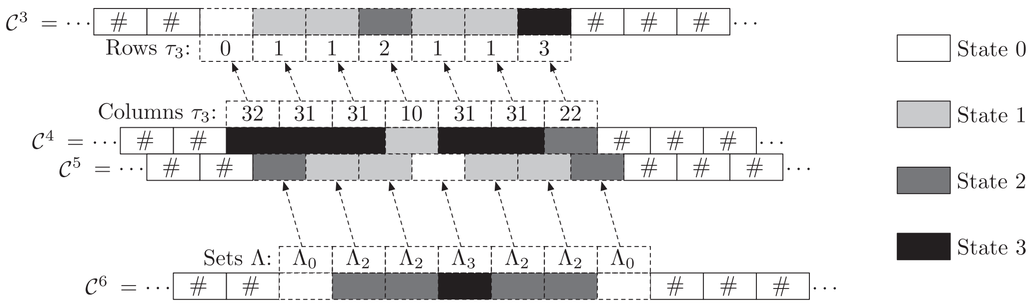

Fig. 7. Obtaining $\mathcal{C}^3$ for the ECAM-3 associated with ECA rule 165.

$$\begin{aligned}
\Lambda_0 &= \{03, 12, 21, 30\}\\
\Lambda_1 &= \{02, 13, 20, 31\}\\
\Lambda_2 &= \{01, 10, 23, 32\}\\
\Lambda_3 &= \{00, 11, 22, 33\}
\end{aligned} \tag{12}$$

Eq. (13) describes, in a tabular way and for $t \geqslant 3$, the memory τ_3 of an ECAM-3 that yields reversible behavior via PECA rule 165. Notice that there are no identical entries in the columns of Eq. (13).

$$\begin{array}{c|cccccccccccccccc}
 & 00 & 01 & 02 & 03 & 10 & 11 & 12 & 13 & 20 & 21 & 22 & 23 & 30 & 31 & 32 & 33\\
\hline
0 & \Lambda_3 & \Lambda_2 & \Lambda_1 & \Lambda_2 & \Lambda_0 & \Lambda_1 & \Lambda_2 & \Lambda_0 & \Lambda_1 & \Lambda_3 & \Lambda_2 & \Lambda_3 & \Lambda_0 & \Lambda_0 & \Lambda_0 & \Lambda_1\\
1 & \Lambda_2 & \Lambda_1 & \Lambda_2 & \Lambda_0 & \Lambda_2 & \Lambda_3 & \Lambda_0 & \Lambda_1 & \Lambda_2 & \Lambda_1 & \Lambda_3 & \Lambda_0 & \Lambda_3 & \Lambda_2 & \Lambda_1 & \Lambda_0\\
2 & \Lambda_1 & \Lambda_3 & \Lambda_0 & \Lambda_3 & \Lambda_3 & \Lambda_2 & \Lambda_1 & \Lambda_2 & \Lambda_0 & \Lambda_0 & \Lambda_1 & \Lambda_2 & \Lambda_1 & \Lambda_1 & \Lambda_3 & \Lambda_3\\
3 & \Lambda_0 & \Lambda_0 & \Lambda_3 & \Lambda_1 & \Lambda_1 & \Lambda_0 & \Lambda_3 & \Lambda_3 & \Lambda_3 & \Lambda_2 & \Lambda_0 & \Lambda_1 & \Lambda_2 & \Lambda_3 & \Lambda_2 & \Lambda_2
\end{array} \tag{13}$$

Fig. 6 presents a 5-step evolution from the initial configuration 0 1 1 2 1 1 3 using the quiescent state #. In this case, $\mathcal{C}_1^1 = 0$; thus, $\tau_3(\mathcal{C}_1^1) = v \in \Lambda_0$ according to Eq. (8). Taking Λ_0 from Eq. (12), v is set to 03; however, any other sequence in Λ_0 could have been selected. Therefore, $\omega(v) = 3$ and $\mathcal{C}_2^1 = 1$; thus, $\tau_3(\mathcal{C}_2^1) = w \in \Lambda_1$. Nevertheless, w must maintain $\alpha(w) = 3$; for this reason, $w = 31$. In fact, Proposition 2 dictates that for every $v \in \Lambda_0$, there exists a unique $w \in \Lambda_1$ such that $\omega(v) = \alpha(w)$.

There is a similar situation for $t \geqslant 3$ when making use of the memory function τ_3 defined by $\beta_3 \circ \gamma_3$. For instance, let us take $\mathcal{C}_1^3 = 0$. Then, $\gamma_3(\mathcal{C}_1^3) = 000$ which is related to entry (0,00) in Eq. (13). Further, $\beta_3(000) = \Lambda_3$ and $\tau_3(\mathcal{C}_1^3) = v \in \Lambda_3$. v has been selected as 11 from Eq. (12); thus, $\omega(v) = 1$. Any other sequence in Λ_3 could have been chosen. For $\mathcal{C}_2^3 = 1$, $\gamma_3(\mathcal{C}_2^3) = 111$. Then, taking the entry (1,11) in Eq. (13), we have that $\beta_3(111) = \Lambda_3$, so $\tau_3(\mathcal{C}_2^3) = w \in \Lambda_3$. However, w must satisfy $\alpha(w) = \omega(v) = 1$. For this reason, the only possible choice of Λ_3 is 11 and $\mathcal{D}_2^3\mathcal{D}_3^3 = 11$.

Let us take the configurations $\mathcal{C}_4, \mathcal{C}_5$, and $\mathcal{C}_6$ to show the invertible behavior of the ECAM-3. Following the proof of Proposition 2, for $1 \leqslant i \leqslant 7$, the entry $\Lambda_{\mathcal{C}_i^6}$ must be taken from column $\mathcal{C}_i^4\mathcal{C}_i^5$ in Eq. (13). Thus, the row related to this entry determines the state of $\mathcal{C}_i^3$. For instance, $\mathcal{C}_1^6 = 0$ in Fig. 7, so we must use Λ_0. Thereafter, $\mathcal{C}_1^4\mathcal{C}_1^5 = 32$; thus, we must select the row with entry Λ_0 in column *32* from Eq. (13). This row has index 0; therefore, $\mathcal{C}_1^3 = 0$. This process is applied to every cell in $\mathcal{C}^6$ to obtain $\mathcal{C}^3$.

6. Final remarks

This paper shows that the correct permutation of states used as memory extends the reversibility of RECAs to create ECAMs. A slight modification of this result yields reversible ECAMs from surjective PECAs. This process consists of representing every PECA with another cellular automaton of 4 states and a neighborhood size of 2 to produce an identical number of rows and Λ sets. The application of a quiescent state avoids the need for periodic boundary conditions. These elements facilitate the specification of memory functions, which allows reversible behavior.

Therefore, these results offer a general method of obtaining reversible ECAMs from reversible and permutative ECAs; a method which, to our knowledge, has not been reported before. This technique can be applied to specify systems in which conservation of information is essential; for instance, in the implementation of computable processes, the main area of interest regarding ECAMs.

In future work, both the extension of these results to one-dimensional cellular automata with any number of states, and the transformation of non-permutative cellular automata into reversible cellular automata with memory will be pursued.

Acknowledgements

This work has been partially supported by CONACYT project No. CB2007-83554 and by PROMEP Collaboration Network "Modelado y análisis de sistemas complejos".

References

[1] R. Alonso-Sanz, Reversible cellular automata with memory: two-dimensional patterns from a single site seed, Physica D 175 (2003) 1–30.
[2] R. Alonso-Sanz, Reversible structurally dynamic cellular automata with memory: a simple example, J. Cell. Autom. 2 (2007) 179–201.
[3] R. Alonso-Sanz, Cellular Automata with Memory, Old City Publishing, 2009.
[4] R. Alonso-Sanz, A. Adamatzky, On beta-skeleton automata with memory, J. Comput. Sci. 2 (2010) 57–66.
[5] R. Alonso-Sanz, L. Bull, One-dimensional coupled cellular automata with memory: initial investigations, J. Cell. Autom. 5 (2010) 29–49.
[6] R. Alonso-Sanz, M. Martin, Elementary cellular automata with elementary memory rules in cells: the case of linear rules, J. Cell. Autom. 1 (2006) 70–86.
[7] T. Boykett, Comparison of radius 1/2 and radius 1 paradigms in one dimensional reversible cellular automata, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.55.6675.
[8] E. Czeizler, On the size of the inverse neighborhoods for one-dimensional reversible cellular automata, Theor. Comput. Sci. 325 (2004) 273–284.
[9] M. Cook, Universality in elementary cellular automata, Complex Syst. 15 (2004) 1–40.
[10] S. Das, B.K. Sikdar, P. Pal Chaudhuri, Characterization of reachable/nonreachable cellular automata states, Lect. Notes Comput. Sci. 3305 (2004) 813–822.
[11] R. Fisch, Clustering in the one-dimensional three-color cyclic cellular automaton, Ann. Probab. 20 (1992) 1528–1548.
[12] J. Gravner, G. Gliner, M. Pelfrey, Replication in one-dimensional cellular automata, Physica D 240 (2011) 1460–1474.
[13] G.A. Hedlund, Endomorphisms and automorphisms of the shift dynamical system, Math. Syst. Theory 3 (1969) 320–375.
[14] A. Ilachinski, Cellular Automata: A Discrete Universe, World Scientific, 2001.
[15] J. Lee, K. Imai, Q. Zhu, Fluctuation-driven computing on number-conserving cellular automata, Inform. Sci. 187 (2012) 266–276.
[16] P. Majia, P. Pal Chaudhuri, Non-uniform cellular automata based associative memory: evolutionary design and basins of attraction, Inform. Sci. 178 (2008) 2315–2336.
[17] C. Marr, M.-Th. Hütt, Outer-totalistic cellular automata on graphs, Phys. Lett. A 373 (2009) 546–549.
[18] G.J. Martínez, A. Adamatzky, R. Alonso-Sanz, Juan C. Seck-Tuoh-Mora, Complex dynamics emerging in Rule 30 with majority memory, Complex Syst. 18 (2009) 345–365.
[19] G.J. Martínez, A. Adamatzky, H.V. McIntosh, Phenomenology of glider collisions in cellular automaton Rule 54 and associated logical gates, Chaos Soliton. Fract. 28 (2006) 00–111.
[20] G.J. Martínez, A. Adamatzky, Juan C. Seck-Tuoh-Mora, R. Alonso-Sanz, How to make dull cellular automata complex by adding memory: rule 126 case study, Complexity 15 (2010) 34–49.
[21] H.V. McIntosh, Reversible cellular automata, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.3214.
[22] H.V. McIntosh, Linear cellular automata via de Bruijn diagrams, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.8763.
[23] H.V. McIntosh, One Dimensional Cellular Automata, Luniver Press, 2009.
[24] K. Morita, Simulating reversible turing machines and cyclic tag systems by one-dimensional reversible cellular automata, Theor. Comput. Sci. 412 (2011) 3856–3865.
[25] D.N. Ostrov, R. Rucker, Continuous-valued cellular automata for nonlinear wave equations, Complex Syst. 10 (1996) 91–119.
[26] J.C. Seck Tuoh Mora, M. González-Hernández, N. Hernández-Romero, A. Rodríguez-Trejo, S.V. Chapa-Vergara, Modeling linear dynamical systems by continuous-valued cellular automata, Int. J. Mod. Phys. C 18 (2007) 833–848.
[27] J.C. Seck Tuoh Mora, M. González-Hernández, H.V. McIntosh, S.V. Chapa-Vergara, Construction of reversible cellular automata by amalgamations and permutations of states, J. Cell. Autom. 4 (2009) 311–322.
[28] T. Serizawa, Three-state Neumann neighbor cellular automata capable of constructing self-reproducing machines, Syst. Comput. Jpn. 18 (1987) 33–40.
[29] Y. Shi, H. Liu, L. Gao, G. Zhang, Cellular particle swarm optimization, Inform. Sci. 181 (2011) 4460–4493.
[30] C. Stone, L. Bull, Evolution of cellular automata with memory: the density classification task, Biosystems 97 (2009) 108–116.
[31] S. Wolfram, A New Kind of Science, Wolfram Media, 2002.
[32] S. Wolfram, Cellular Automata And Complexity: Collected Papers, Westview Press, 1994.
[33] A. Wuensche, M. Lesser, The Global Dynamics of Cellular Automata: An Atlas of Basin of Attraction Fields of One-Dimensional Cellular Automata, Perseus Books, 1992.