Complex Systems 1 (1987) 1035–1044

On Invertible Cellular Automata*

Karel Culik II
Department of Computer Science, University of South Carolina, Columbia, SC 29208, USA

Abstract. The problem of testing whether a given cellular automaton rule has an inverse was stated by Toffoli and Margolus. This problem is equivalent to the invertibility of the given CA-rule (injectivity of the global function it defines). It is decidable for one-dimensional cellular automata and open for higher dimensions. We show some related results and give automata-theoretic proofs of some known results.

1. Introduction

Recently, cellular automata (CA) have been intensively studied as models of complex natural systems containing large numbers of simple identical components with local interactions [13,14]. In these applications, an important property of a CA is reversibility (invertibility). A cellular automaton is given by its local rule (CA-rule) that specifies a deterministic global function on the configurations of the system. If this function is injective (the system is backward-deterministic), then the rule is called *invertible*; moreover, if there is a CA-rule that makes the system to go backwards, then it is called the *inverse* rule with respect to the original, or *direct*, rule. For more discussion, see reference [13, chapter 14]; specifically, it is noted there that for locally interacting systems having a finite amount of information per site, such as cellular automata, reversibility is equivalent to the second law of thermodynamics.

The injectivity of the global function defined by a cellular automaton was studied in [1,9,11] using the terminology "tessellations with local transformations" rather than cellular automata. In particular, Amoroso and Patt [1] have shown that given a one-dimensional CA-rule, it is decidable whether it is invertible (i.e., whether the global function is injective), and Richardson [11] has shown that a CA-rule is invertible iff it has an inverse. This result holds for any n-dimensional CA. We will give an alternative automata theoretic proofs of these results for one-dimensional CA and show some related

*This research was supported by the National Sciences Foundation under Grant No. CCR-8702752 and the Natural Sciences and Engineering Research Council of Canada under Grant No. A-7403.

results. The above results give a solution for one-dimensional CA of a problem stated by Toffoli and Margolus. We quote from [13, p. 146]: "The theory of invertible cellular automata has many open problems [12]; in particular, no general decision procedure is known for determining whether a given rule has an inverse (and this question may well be undecidable)." Since a CA-rule has an inverse iff it is invertible, this problem is equivalent to the testing of invertibility (injectiveness of the global function). This is decidable for one-dimensional CA and open for two-dimensional. Our results will be limited to one-dimensional (linear) CA only. A linear CA consists of biinfinite string of identical cells. The next state of each cell is determined by the current state of the cells in its neighborhood which consists of the cell itself and r of its neighbors at each side, for some $r \geq 0$. It is clear that our results can be generalized for different types of neighborhood.

Our arguments rely on the fact that the function defined by CA on biinfinite configurations is continuous in the space S^Z endowed by the product topology [3,4]. We will also use the fact that CA-functions can be defined by our $\omega\omega$-sequential machines which are a generalization of sequential machines [5] to biinfinite words or a special case of $\omega\omega$-finite transducers considered in [4]. We will use some well-known results on finite transducers [6,2] and their extensions to $\omega\omega$-sequential machines.

2. Preliminaries

Let Z be set of integers. A *cellular automaton* (CA) is an infinite array, indexed by Z, of *cells*. Each cell is identified by its *location* $I \in Z$.

At any time, each cell has a *state*, which belongs to a finite set S. The dynamic behavior of the CA is determined by a rule that describes the state of each cell at time $t+1$ as a function of the states of the cell itself, the r neighboring cells on its left, and the r neighboring cells on its right at time t. The rule is invariant with respect to translations (shifts) of Z.

Formally, a cellular automaton is a triple $A = (S, r, f)$, where S is a finite set of *states*, r specifies the size of the *neighborhood*, and $f : S^{2r+1} \to S$ is the local function called the *CA-rule*.

A *configuration* c of the CA is a function $c : Z \to S$, which assigns a state S to each cell of the CA. The set of configurations is denoted S^Z. The local function f is extended to the *global function*

$$G_f : S^Z \to S^Z$$

of the set of configurations into itself. By definition, for $c_1, c_2 \in S^Z$,

$$G_f(c_1) = c_2$$

if and only if

$$c_2(i) = f(c_1(i-r), c_1(i-r+1), \ldots, c_1(i+r))$$

for all $i \in Z$.

The function G_f describes the dynamic behavior of the CA: the CA moves from the configuration c at time t to the configuration $G_f(c)$ at time $t+1$. The state of cell i at time $t+1$ depends only on the states of the cells in the neighborhood $(i-r, i-r+1, \ldots, i+r)$ at time t. Notice that besides being locally defined, the global function G_f is total and translation-invariant.

We call a mapping $h : S^Z \to S^Z$ a CA-function if there is a CA-rule f such that $h = G_f$.

Example 1. *Let $A = (S, 2, f)$ be a CA, where $S = \{0, 1\}$, and*

$$f(x_1, x_2, x_3, x_4, x_5) = \begin{cases} 1 & \text{if } x_1 + x_2 + x_3 + x_4 + x_5 = 4; \\ 0 & \text{otherwise.} \end{cases}$$

If c is a configuration consisting of all 1's and c' is a configuration of all 0's, then $G_f(c) = c'$. ∎

Frequently, a state $\tilde{q}$ with the property

$$f(\tilde{q}, \tilde{q}, \ldots, \tilde{q}) = \tilde{q}$$

is distinguished and called the *quiescent state*. In CA, there may be more than one state with the above property, but at most, one of them is distinguished as the quiescent state. The configuration with all cells in the quiescent state is called the *quiescent configuration*, denoted by $\tilde{Q}$.

The configuration space S^Z is a product of infinitely many finite sets S. When S is endowed with the discrete topology, the *product topology* on S^Z is compact by Tychonoff's theorem [7, theorem 5.13]. A subbasis of open sets for the product topology consists of all sets of the form

$$\{\, c \in S^Z \mid c(i) = a \,\}, \tag{2.1}$$

where $i \in Z$ and $a \in S$. A subset of S^Z is open if and only if it is a union of finite intersections of sets of the form (2.1). It is easy to show that for every CA-rule, the global function G_f is continuous from S^Z to S^Z. (Thus, the pair (S^Z, G_f) is a classical dynamical system, in the sense of [3]).

We assume that the reader is familiar with finite automata and regular sets (see [6]); finite transducer is a nondeterministic finite automaton (possibly with ϵ-moves) that produces an output string at each step; for a formal definition, see [2]. Their generalizations to biinfinite words ($\omega\omega$-words) are called $\omega\omega$-finite automata, $\omega\omega$-regular sets, $\omega\omega$-finite transducers. They were studied in [10] and [4].

A biinfinite word c is a mapping $Z \to S$, where Z is the set of all integers and S is a finite alphabet. The symbol $c(j)$, $j \in Z$, denotes the j^{th} letter of c.

The infinite repetition of the word w to the right is denoted by w^ω, the infinite repetition to the left is denoted by ${}^\omega w$. So, for example, ${}^\omega abc^\omega$ denotes an infinite number of a's followed by one b and an infinite number of c's. More precisely, ${}^{\omega}abc^{\omega}$ is the (shift invariant) class of biinfinite words $\alpha_k : Z \to S$, for all $k \in Z$, where

$$\begin{aligned} \alpha_k(j) &= a \text{ for } j < k, \\ a_k(k) &= b, \text{ and} \\ a_k(j) &= c \text{ for } j > k. \end{aligned}$$

Configurations of a CA $A = (S, r, f)$ can be viewed as biinfinite words over alphabet S. Alternatively, they can be viewed as equivalence classes of biinfinite words that are shift equivalent. Since G_f is always shift invariant, we may view G_f as function over these equivalence classes.

Let $\tilde{q} \in S$ be the quiescent state. The biinfinite words in ${}^{\omega}\tilde{q}S^{\star}\tilde{q}^{\omega}$ are called *pseudofinite*; that is, a configuration (word) is pseudofinite if it has only finitely many nonquiescent states.

Now, we introduce the notion of a (nondeterministic) $\omega\omega$-sequential machine which defines a mapping on biinfinite words. It is a nondeterministic $\omega\omega$-finite automaton [4,10] that consumes exactly one input symbol and produces exactly one output symbol at each step of the computation.

Formally, an $\omega\omega$-sequential machine M is a six-tuple $(Q, \Sigma, \Delta, \alpha, F_L, F_R)$ where

(i) Q is the finite set of states,

(ii) Σ is the input alphabet,

(iii) Δ is the output alphabet,

(iv) $\alpha : Q \times \Sigma \to 2^{Q \times \Delta}$ is the transition function,

(v) $F_L \subseteq Q$ is the set of left (accepting) states, and

(vi) $F_R \subseteq Q$ is the set of right (accepting) states.

A biinfinite word v is an output on biinfinite input u under $\omega\omega$-sequential machine M if there is a biinfinite sequence of states in Q

$$\ldots, q_{-2}, q_{-1}, q_0, q_1, q_2, \ldots$$

such that for all $j \in Z$.

(i) $(q_{j+1}, v(j)) \in \alpha(q_j, u(j))$, and

(ii) there exist $m, n \in Z, m \leq j \leq n$, such that $q_m \in F_L$ and $q_n \in F_R$.

The input-output relation defined by $\omega\omega$-sequential machine M is denoted by $R^{\omega\omega}(M)$. Now, we state the quite obvious relation between CA, $\omega\omega$-sequential machines and finite transducers that will be useful later.

Lemma 1. *For each CA-rule $f : S^{2r+1} \to S$, we can construct an $\omega\omega$-sequential machine M such that*

(i) $R^{\omega\omega}(M) = G_f$.
Moreover, if $\tilde{q} \in S$ (the quiescent symbol), then we can construct $\omega\omega$-sequential machine $\hat{M}$ (the restriction of M to pseudofinite words) and finite transducer $\tilde{M}$ such that

(ii) $R^{\omega\omega}(\hat{M}) = G_f \cap ({}^{\omega}\tilde{q}S^{*}\tilde{q}^{\omega} \times {}^{\omega}S^{\omega})$

(iii) $R(\tilde{M}) = \{(u,v) | ({}^{\omega}\tilde{q}u\tilde{q}^{\omega}, {}^{\omega}\tilde{q}v\tilde{q}^{\omega}) \in R^{\omega\omega}(M)$ and $u, v \in (S-\{\tilde{q}\})S^{*}(S-\{\tilde{q}\})\}$.

Proof. Parts (i) and (ii) are obvious; part (iii) is a routine exercise on finite transducers. ∎

3. Invertible cellular automata

A CA-rule $f : S^{2r+1} \to S$ is *invertible* if the global function G_f is injective, that is, if for $\alpha, \beta \in S^Z$ $G_f(\alpha) = G_f(\beta)$ implies $\alpha = \beta$. If there is a CA-rule $g : S^{2s+1} \to S$ for some $s \geq 0$ such that the composition of G_f and G_g is the identity mapping on S^Z we call g *an inverse* rule of CA-rule f and say that *f has an inverse*. Actually, we will show that every invertible CA-rule has an inverse, which certainly is not obvious.

The following easy observation is in [13].

Lemma 2. *Given two CA-rules $f : S^{2r+1} \to S$ and $g : S^{2s+1} \to S$, $r, s \geq 0$, we can test whether g is an inverse of f.*

The above result, clearly holds even if g and f are defined by single-valued $\omega\omega$-sequential machines.

Since there are only finitely many possible CA-rules with fixed set of states and fixed r (the size of neighborhood), it immediately follows by lemma 2 that it is decidable whether a given CA-rule has an inverse with the same neighborhood (same r). Later (theorem 2), we will show that we can also decide whether there exists an inverse with arbitrary size of neighborhood.

Now, we show that two variants of the invertibility problem for CA are decidable.

Theorem 1. *Given a CA-rule f, it is decidable whether G_f is injective on pseudofinite configurations.*

Proof. By lemma 1, we can construct $\omega\omega$-sequential machine M such that $R^{\omega\omega}(M) = G_f$ and finite transducer $\tilde{M}$ such that $R(M)$ is the restriction of G_f to pseudofinite strings with the "blanks" omitted.

Using well-known results on finite transducers, see [2] or [6], we can construct finite transducer N such that $R(N) = R(\tilde{M})^{-1}$ and test whether N is single-valued. Clearly, G_f is invertible on pseudofinite strings iff N is single-valued. ∎

Using a combinatorial argument, it was shown in [1] that, given a CA-rule f, it is decidable whether f is invertible (whether G_f is injective). We give an alternative automata theoretic proof of this result. First, we need a lemma which is of interest on its own.

Lemma 3. *Let $f : S^{2r+1} \to S$ be a CA-rule. The global function G_f is injective iff it is injective on the periodic configurations ($\omega\omega$-strings over S).*

Proof. Assume f is not invertible (G_f is not injective); that is, there are two distinct $\omega\omega$-strings α, β such that

$$G_f(\alpha) = G_f(\beta) = \gamma.$$

In the following, we consider $\omega\omega$-strings as mappings $Z \to S$, not as equivalence classes of these mappings with respect to shifting. We need to fix the mutual position of α, β and their image. Since α, β are biinfinite, we can find their substrings of length $2r$ (for any fixed r) that repeat in the same positions, and since they are distinct, we can choose such repetition that the substrings between them are distinct. Formally, we can find $s, t \in Z$, $s+2r < t$, such that

$$\alpha(s+i) = \alpha(t+i)\ ,\ \beta(s+i) = \beta(t+i)$$

for $i = 0, \ldots, 2r-1$ and $\alpha(j) \neq \beta(j)$ for some j, $s+2r \leq j \leq t-1$.

Now, we transform α and β into distinct periodic $\omega\omega$-strings on which G_f agrees. Consider the following finite strings u_1, u_2, v_1, v_2, x, y and w, (see figure 1). Note that $x \neq y$ because $\alpha(j) \neq \beta(j)$.

$$\begin{aligned} u_1 &= \alpha(s)\ldots\alpha(s+r-1) = \alpha(t)\ldots\alpha(t+r-1) \\ u_2 &= \alpha(s+r)\ldots\alpha(s+2r-1) = \alpha(t+r)\ldots\alpha(t+2r-1) \\ v_1 &= \beta(s)\ldots\beta(s+r-1) = \beta(t)\ldots\beta(t+r-1) \\ v_2 &= \beta(s+r)\ldots\beta(s+2r-1) = \beta(t+r)\ldots\beta(t+2r-1) \\ x &= \alpha(s+2r)\ldots\alpha(t-1) \\ y &= \beta(s+2r)\ldots\beta(t-1) \\ w &= \gamma(s+r)\ldots\gamma(t+r-1) \end{aligned}$$

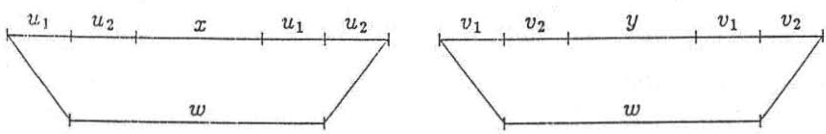

Figure 1: Two distinct substrings with the same image.

Now, we define the periodic $\omega\omega$-strings

$$\tau = {}^{\omega}(u_2 x u_1)^{\omega} \text{ and } \sigma = {}^{\omega}(v_2 y v_1)^{\omega},$$

or, more precisely,

$$\tau(s+r+i+k(t-s)) = \alpha(s+r+i) \text{ and}$$

$$\sigma(s+r+i+k(t-s)) = \beta(s+r+i)$$

for $i = 0, \ldots, t-s-1$ and $k \in Z$.

Clearly, $G_f(\tau) = {}^\omega w^\omega = G_f(\sigma)$. Since $\tau \neq \sigma$, we have shown that G_f is not injective on periodic configurations. ∎

A combinatorial proof of the following result was given in [1].

Theorem 2. *Given a CA-rule $f : S^{2r+1} \to S$, it is decidable whether f is invertible (G_f is injective).*

Proof. By lemma 3, it is sufficient to test the injectivity of G_f on the periodic configurations. We will do this by constructing a finite transducer that will simulate G_f on one period of any periodic configuration.

Let $\rho \subseteq S^* \times S^*$ be defined by

$$\rho = \{(u,v) \in S^* \times S^* \mid |u| = |v|, G_f({}^\omega u^\omega) = {}^\omega v^\omega\}.$$

More precisely, we are requiring that u and v are in the "same position", i.e. that

$$u = {}^\omega u^\omega(s) \ldots {}^\omega u^\omega(s+k) \text{ and}$$

$$v = {}^\omega v^\omega(s) \ldots {}^\omega v^\omega(s+k)$$

for some $s \in Z$ and $k \geq 0$.

A finite transducer T_1 defining ρ can be constructed as follows. First, T_1 guesses the last r symbols in its input u, then reads u and simulates G_f. It remembers the first r symbols in order to be able to compute the last r outputs. It guesses that the last r inputs are coming, and if they match its original guess, then T_1 goes into a final state.

By interchanging inputs and outputs of T_1, we obtain a finite transducer T_2 defining ρ^{-1}. Clearly, G_f is injective on periodic configurations iff T_2 is single-valued. The latter is decidable [2]. ∎

Remark. A more detailed analysis of the proof of lemma 3 allows to strengthen lemma 3 and theorem 2. From the number of states in S and the size of the neighborhood r, we can compute bound B such that the function G_f is injective iff it is injective on periodic configurations with period at most B. This gives the possibility of an alternate proof of (strengthened) theorem 2 by bounding the size of the neighborhood of a possible inverse rule. The latter is particularly interesting in view of potential cryptographic applications.

Lemma 4. *Given a (nondeterministic) $\omega\omega$-sequential machine M, it is decidable whether M defines a CA-function, that is whether there is a CA-rule $f : S^{2r+1} \to S$ for some $r \geq 0$ such that $R^{\omega\omega}(M) = G_f$. Moreover, if $R^{\omega\omega}(M)$ is a CA-function, then we can effectively find the smallest integer r_0 and CA-rule f_0 such that $f_0 : S^{2r_0+1} \to S$ and $R^{\omega\omega}(M) = G_{f_0}$.*

Proof. Define the set $R_M \subseteq (S \times S)^*$ consisting of all strings of pairs

$$(a_1, a_{2m+1})(a_2, a_{2m}) \ldots (a_r, a_{r+2})(a_{r+1}, a_{r+1})$$

such that the output symbol that M produces after reading the symbol a_{m+1} in the biinfinite word $\ldots\ a_1a_2 \ldots a_{2m+1}\ \ldots$ is not uniquely determined. See figure 2 for substring a_1, a_2, a_3, a_4, a_5 yielding b or c, $b \neq c$, after reading a_3. This substring will be folded into the string $(a_1, a_5)(a_2, a_4)(a_3, a_3)$ in R_M.

By constructing a nondeterministic finite automaton that simulates M forward on the odd components of the quadruples and backwards on the even components (starting from an arbitrary state), we show that the set R_M is (effectively) regular, and therefore, we can test whether R_M is finite [6]. Clearly, $R^{\omega\omega}(M)$ is a CA-function iff R_M is finite.

Finally, if R_M is finite, then let $2r_0 + 1$ be the length of the longest string in R_M. Clearly, r_0 is the smallest integer such that $f_0 : S^{2r_0+1} \to S$ and $G_{f_0} = R^{\omega\omega}(M)$. ∎

$$\begin{array}{ccccc} a_1 & a_2 & a_3 & a_4 & a_5 \\ \cdot & \cdot & b & \cdot & \cdot \end{array} \qquad \begin{array}{ccccc} a_1 & a_2 & a_3 & a_4 & a_5 \\ \cdot & \cdot & c & \cdot & \cdot \end{array}$$

Figure 2: If $b \neq c$, then there is no CA-rule with $r = 2$.

In reference [11], the following result was shown for arbitrary n-dimensional CA as a corollary of properties of nondeterministic local transformations. We give a direct proof for one-dimensional CA. Our proof can be extended to n-dimensions for arbitrary $n \geq 1$.

Theorem 3. *A CA-rule $f : S^{2r+1} \to S$ has an inverse if and only if it is invertible.*

Proof. 1. Obviously, if f has an inverse, then it is invertible. 2. We extend function f to subconfigurations of length at least $2r + 1$ in the obvious way. For all $m \geq 1$, let

$$f(a_1a_2 \ldots a_{2r+m}) = f(a_1 \ldots a_{2r+1})f(a_2 \ldots a_{2r+2}) \ldots f(a_m \ldots a_{2r+m}).$$

Now, assume that f is invertible but does not have an inverse. That means that for all $s \geq 0$ there exist $u_s, v_s, w_s, x_s, y_s \in S^*, |w_s| = s$, and $a_s, b_s \in S, a_s \neq b_s$ such that

$$f(u_sa_sv_s) = f(x_sb_sy_s) = w_s \tag{3.1}$$

(see figure 3).

We define a partial ordering $\prec$ on S^* by:

$$x \prec y \text{ iff } y = uxv \text{ for some } u, v \in S^*,\ |u| = |v|$$

and we extend $\prec$ to $S^* \times S^*$ by:

$$(u, v) \prec (x, y) \text{ if } u \prec x \text{ and } v \prec y.$$

Partial ordering $\prec$ defines an infinite rooted tree of finite degree on pairs $(u_s a_s v_s, x_s b_s y_s)$; thus, by König infinity lemma [8, p. 381], there exists an infinite strictly increasing sequence of integers $i_1, i_2, i_3, \ldots$ such that

$$(u_{i_1} a_{i_1} v_{i_1}, x_{i_1} b_{i_1} y_{i_1}) \prec (u_{i_2} a_{i_2} v_{i_2}, x_{i_2} b_{i_2} y_{i_2}) \prec \ldots$$

The limits α, β, with $\alpha \neq \beta$, clearly exist in the product topology on S^Z

$$\alpha = \lim_{n\to\infty} u_{i_n} a_{i_n} v_{i_n}, \quad \beta = \lim_{n\to\infty} x_{i_n} b_{i_n} y_{i_n} .$$

Since $f(u_{i_n} a_{i_n} v_{i_n}) = f(x_{i_n} b_{i_n} y_{i_n}) = w_{i_n}$ for all $n \geq 1$, by continuity of function G_f we have $G_f(\alpha) = G_f(\beta)$, a contradiction with the invertibility of f. ∎

Theorem 4. *Given a CA-rule $f : S^{2r+1} \to S$, it is decidable whether f has an inverse, that is, whether there is a CA-rule $g : S^{2s+1} \to S$, for some $s \geq 0$ such that $G_f \circ G_g$ is the identity on S^Z. If it exists, then g with the minimal s (size of the neighborhood) can be constructed.*

Figure 3: Two distinct substrings with the same image.

Proof. By theorem 2, f has an inverse iff it is invertible. The invertibility of a CA-function is decidable by theorem 2. If f has an inverse, we can, by lemma 4, find the inverse rule $g : S^{2s+1} \to S$ with the minimal $s \geq 0$. ∎

An alternative proof is as follows. By lemma 1, we can construct an $\omega\omega$-sequential machine M such that $R^{\omega\omega}(M) = G_f$. By interchanging inputs and outputs of M we obtain $\omega\omega$-sequential machine N, such that $R^{\omega\omega}(N) = G_f^{-1}$. By lemma 4, we can test whether N defines a CA-function, that is, whether there exists an inverse g of f. ∎

Remark. Let $f : S^{2r+1} \to S$ be a CA-rule with the quiescent state $\tilde{q}$. It follows easily by our previous considerations that f has an inverse (on S^Z) iff f has an inverse on pseudofinite configurations.

Acknowledgement

An anonymous referee suggested the remark after theorem 2.

References

[1] S. Amoroso and Y. N. Patt, "Decision Procedures for Surjectivity and Injectivity of Parallel Maps for Tessellation Structures", *Journal of Computer and System Sciences*, 6 (1972) 448–464.

[2] J. Berstel, *Transductions and Context-Free Languages*, (Teubner, Stuttgart, 1979).

[3] J. R. Brown, *Ergodic Theory and Topological Dynamics*, (Academic Press, 1976).

[4] K. Culik II, J. Pachl, and S. Yu, "On the Limit Sets of Cellular Automata", University of Waterloo, Res. Rep. CS-87-47.

[5] S. Ginsburg, *The Mathematical Theory of Context-Free Languages*, (McGraw-Hill, 1966).

[6] M. A. Harrison, *Introduction to Formal Language Theory*, (Addison-Wesley, 1978).

[7] J. L. Kelley, *General Topology*, (American Book Company, 1955).

[8] D. E. Knuth, *The Art of Computer Programming, Vol. I* (second edition), (Addison-Wesley, 1973).

[9] E. F. Moore, "Machine Models of Self-Reproduction", *Proceedings of the Symposium on Applied Mathematics 14*, (American Mathematical Society, Providence R.I., 1962) 17–33.

[10] M. Nivat and D. Perrin, "Ensembles Reconnaissables de Mots Biinfinis", *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, (1982) 47–59.

[11] D. Richardson, "Tessellations with Local Transformations", *Journal of Computer System Science*, **6** (1972) 373–388.

[12] T. Toffoli, "Reversible Computing", in *Automata, Languages and Programming*, ed. J. W. Bakker and J. van Leeuwen, 7th ICALP, *Lecture Notes in Computer Science*, **85** (Springer-Verlag, 1980).

[13] T. Toffoli and N. Margolus, *Cellular Automata Machines*, (MIT Press, Cambridge, 1987).

[14] S. Wolfram, *Theory and Applications of Cellular Automata*, (World Scientific, Singapore, 1986).