Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science 253 (2010) 23–31
Electronic Notes in Theoretical Computer Science
www.elsevier.com/locate/entcs

Universality Issues in Reversible Computing Systems and Cellular Automata (Extended Abstract)

Kenichi Morita[1]

Department of Information Engineering, Graduate School of Engineering, Hiroshima University Higashi-Hiroshima, 739-8527, Japan

Abstract

In this survey, we deal with the problem how a universal computer can be constructed in a reversible environment. We discuss this problem based on the frameworks of reversible Turing machines, reversible logic circuits, and reversible cellular automata. We can see that in spite of the constraint of reversibility, there are several very simple reversible systems that have universal computing ability.

Keywords: reversible Turing machine, reversible logic element, reversible cellular automaton

1 Introduction

Since reversibility is one of the fundamental microscopic physical laws of Nature, it is important to investigate how universal computer can be implemented efficiently in a system having this property. This is because the size of future computing devices will become nanoscale ones. Here, we give a survey on universality issues of such systems, in particular on reversible Turing machines, reversible logic circuits, and reversible cellular automata. We can see that even very simple reversible systems have universal computing ability. In these reversible systems, computation can be often carried out in a very different manner from conventional computing systems, and thus they give new ways and concepts for future computing.

2 Reversible Turing machines

A reversible Turing machine (RTM) is a standard model of computation in the theory of reversible computing. Lecerf [11] first investigated RTMs, and showed

[1] Email: morita@iec.hiroshima-u.ac.jp

doi:10.1016/j.entcs.2010.02.004

unsolvability of their halting problem. Bennett [1,2,3] studied them from the standpoint of thermodynamics of computing.

2.1 Definitions on reversible Turing machines

Bennett [1] used a quadruple formulation for RTMs, because an "inverse" RTM for a given RTM is easily defined. Here, we use a quintuple formulation instead. This is because classical TMs are usually defined in the quintuple form, hence making comparison easier.

Definition 2.1 *A* Turing machine (TM) in the quintuple form *is defined by*

$$T = (Q, S, q_0, q_f, s_0, \delta),$$

where Q is a set of states, S is a set of tape symbols, $q_0 \in Q$ is an initial state, $q_f \in Q$ is a final state, and $s_0 \in S$ is a blank symbol. δ is a move relation, which is a subset of $(Q \times S \times S \times \{L, N, R\} \times Q)$, where $L, N,$ and R stand for left-shift, no-shift, and right-shift of the head. Each element of δ is a quintuple of the form $[p, s, s', d, q]$. It means if T reads the symbol s in the state p, then write s', shift the head to the direction d, and go to the state q.

T is called deterministic *iff the following condition holds for any pair of distinct quintuples $[p_1, s_1, s_1', d_1, q_1]$ and $[p_2, s_2, s_2', d_2, q_2]$ in δ.*

$$\text{If } p_1 = p_2, \text{ then } s_1 \neq s_2.$$

T is called reversible *iff the following condition holds for any pair of distinct quintuples $[p_1, s_1, s_1', d_1, q_1]$ and $[p_2, s_2, s_2', d_2, q_2]$ in δ.*

$$\text{If } q_1 = q_2, \text{ then } s_1' \neq s_2' \ \wedge \ d_1 = d_2.$$

Example 2.2 *Consider a simple RTM $T_{\mathrm{parity}} = (Q, \{0, 1\}, q_0, q_{\mathrm{acc}}, 0, \delta)$, where $Q = \{q_0, q_1, q_2, q_{\mathrm{acc}}, q_{\mathrm{rej}}\}$, and δ is as follows: $\delta = \{[\ q_0, 0, 1, R, q_1\ ],\ [\ q_1, 0, 1, N, q_{\mathrm{acc}}\ ],\ [\ q_1, 1, 0, R, q_2\ ],\ [\ q_2, 0, 1, N, q_{\mathrm{rej}}\ ],\ [\ q_2, 1, 0, R, q_1\ ]\}$. T_{parity} checks if a given unary number n is even or odd. If it is even, T_{parity} halts in the accepting state q_{acc}, otherwise halts in the rejecting state q_{rej}. All the symbols read by T_{parity} are complemented (see Fig. 1).*

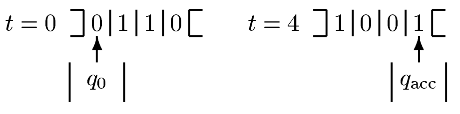

Fig. 1. The initial and the final configuration of an RTM T_{parity} for a unary input 11.

2.2 Universal reversible Turing machines

Bennett [1] proved universality of RTMs showing that any irreversible TM can be converted to a garbage-less RTM.

Theorem 2.3 *(Bennett [1]) For any irreversible 1-tape TM T, we can construct a garbage-less 3-tape RTM T' that simulates the former and gives only an input string and an output string on the tapes when it halts.*

A universal Turing machine (UTM) can simulate any TM, or equivalently can compute every recursive function. Finding small (irreversible) UTMs has a long history (see e.g. [32]), and the following UTMs are known to be minimal, where UTM(m,n) denotes an m-state n-symbol UTM. We have the UTM(2,18), UTM(4,6), and UTM(5,5) proposed by Rogozhin [27]; the UTM(3,9) by Kudlek and Rogozhin [10]; and the UTM(6,4), UTM(9,3) and UTM(18,2) developed by Neary and Woods [24]. By Theorem 2.3, we can obtain a URTM from a UTM. But, if we convert any one of these UTMs to a URTM by this method, its size becomes large.

For RTMs, the following small URTMs are constructed by simulating a cyclic tag system (CTAG) defined by Cook [4], which is a very simple string rewriting system with computation-universality. They are the 17-state 5-symbol URTM (URTM(17,5)) with 67 quintuples proposed by Morita and Yamaguchi [20] and the URTM(15,6) with 62 quintuples defined by Morita [22]. Fig. 2 shows these results.

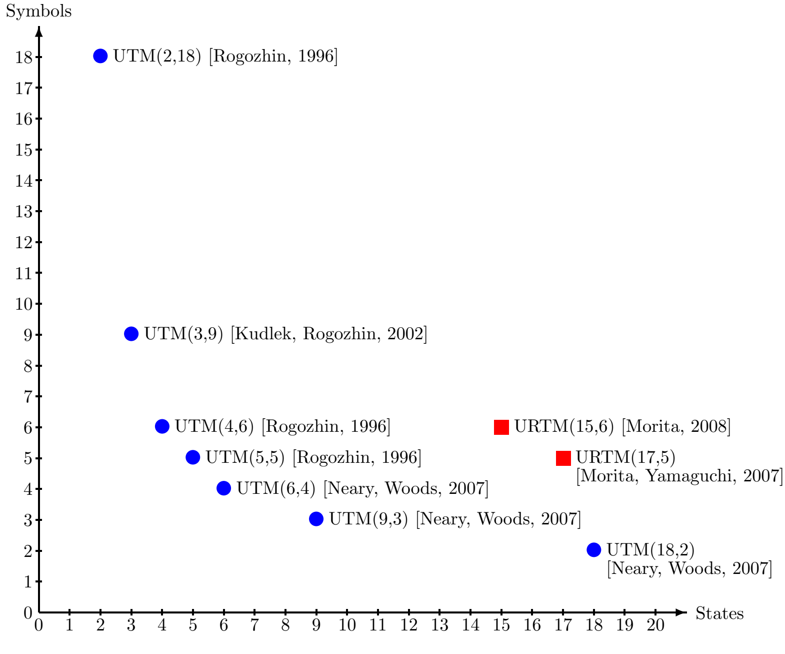

Fig. 2. Small universal TMs (UTMs) and universal reversible TMs (URTMs).

3 Reversible logic elements and circuits

A reversible logic element is a primitive for composing reversible logic circuits, whose function is described by a one-to-one mapping. There are two types of such elements: one without memory, which is usually called a reversible logic gate, and one with memory. Early study on reversible gates was made by Petri [26]. Later, Toffoli [29,30], and Fredkin and Toffoli [5] studied them in connection with physical reversibility. Among many reversible logic gates, Toffoli gate [29,30] and Fredkin gate [5] have been studied extensively, and are known to be logically universal. On the other hand, reversible logic elements with memory are also useful in reversible computing. Hereafter, we mainly discuss reversible logic elements with memory of two states, and in particular the so-called rotary element.

3.1 A rotary element

A rotary element (RE) [17] has two internal states called H-state ( ) and V-state ( ), and four input lines $\{n, e, s, w\}$ and four output lines $\{n', e', s', w'\}$. We can interpret it as a one having a "rotating bar" to control the moving direction of an input signal (or a particle). When no particle exists, nothing happens on the RE. If a particle comes from the direction parallel to the rotating bar, then it goes out from the output line of the opposite side without affecting the direction of the bar (Fig. 3 (a)). If a particle comes from the direction orthogonal to the bar, then it makes a right turn, and rotates the bar by 90 degrees counterclockwise (Fig. 3 (b)). It is reversible in the following sense: from the next state and the output, the previous state and the input are uniquely determined.

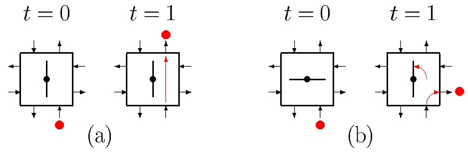

Fig. 3. Operations of an RE: (a) the parallel case, and (b) the orthogonal case.

The following theorem is proved by constructing a circuit made only of REs that simulates a Fredkin gate.

Theorem 3.1 *(Morita, et al. [17,19]) The set $\{RE\}$ is logically universal.*

Universality of an RE is also shown by giving a direct construction method of an RTM by REs [17]. Actually, this method is much simpler than to use reversible logic gates to construct RTMs. Fig. 4 shows a garbage-less circuit composed only of REs that simulates the RTM T_{parity} in Example 2.2.

3.2 Realization of a rotary element by the Billiard Ball Model

The Billiard Ball Model (BBM) is a reversible physical model of computing proposed by Fredkin and Toffoli [5]. It is an idealized mechanical model consisting of balls and reflectors. Balls can collide with other balls or reflectors. It is assumed that collisions are elastic, and there is no friction. Fredkin and Toffoli [5] showed that Fredkin gate is realizable in BBM.

It is also possible to realize an RE in BBM as shown in Fig. 5 [22]. It consists of one stationary ball, and many reflectors indicated by small rectangles. If a stationary ball is put at the position V (or H, respectively), then we regard it as being in the V-state (H-state). An input ball can be given to any one of the input lines n, e, s, and w, at any moment and at any speed.

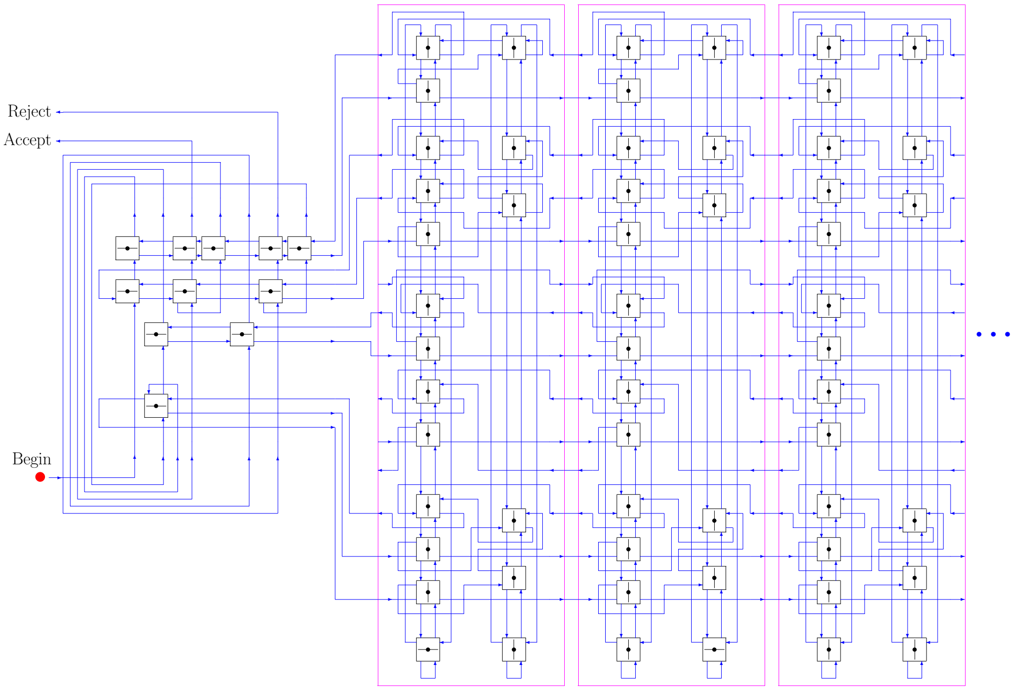

Fig. 4. A circuit made of REs that simulates the RTM T_{parity} in Example 2.2.

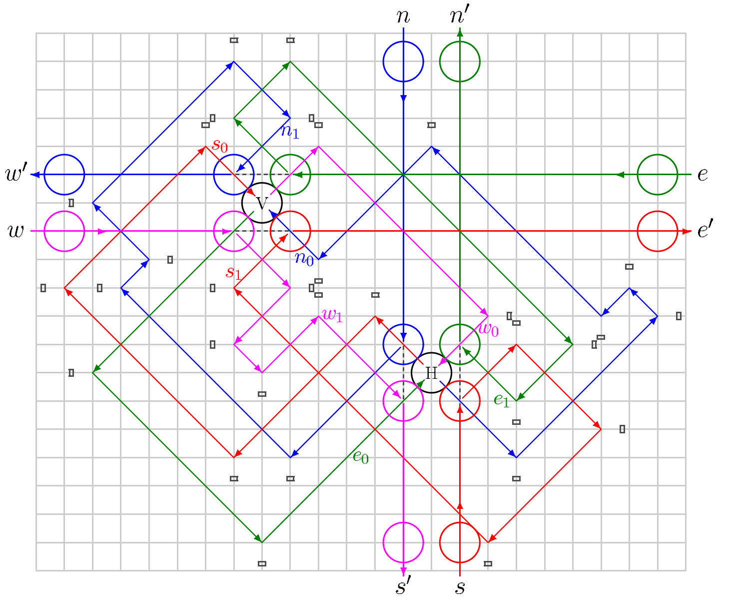

Fig. 5. A rotary element (RE) realized in BBM.

4 Reversible cellular automata

A reversible cellular automaton (RCA) is a mathematical model of a reversible space defined as a CA whose global function (a mapping from configurations to configurations) is one-to-one. Toffoli [28] first studied universality of RCAs. It is, however, in general difficult to design an RCA if we use traditional CAs [9]. So far, a few frameworks are proposed for designing RCAs. They are CAs with block rules [13,31], partitioned CAs (PCAs) [14], and CAs with second order rules [13,31].

Here we consider the framework of PCAs. In a one-dimensional three-neighbor PCA, each cell is divided into three parts, i.e., left, center, and right parts, and their state sets are L, C, and R. The next state of a cell is determined by the present states of the left part of the right-neighbor cell, the center part of this cell, and the right part of the left-neighbor cell . Fig. 6 shows its cellular space, and how the local function f is applied. Higher dimensional PCAs can be also defined similarly.

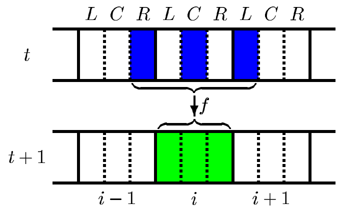

Fig. 6. Cellular space of a 1-dimensional 3-neighbor PCA, and its local function f.

It has been shown that a local function of a PCA is one-to-one iff its global function is one-to-one [14]. Furthermore, PCAs are a subclass of traditional CAs. Hence, if we want to have an RCA, it is sufficient to construct a PCA whose local function is one-to-one.

4.1 One-dimensional universal RCAs

In the case of one-dimensional irreversible CA, Cook [4] showed that the elementary cellular automaton (ECA) of rule 110, which has only 2 states and 3 neighbours, is universal. He proved it by showing that the ECA of rule 110 with infinite (but ultimately periodic) configurations can simulate any cyclic tag system.

How is it in the reversible case? The following result has been shown by constructing reversible PCAs that can simulate cyclic tag systems.

Theorem 4.1 *(1) There is a 24-state universal reversible PCA with infinite (but ultimately periodic) configurations (Morita [23]). (2) There is a 98-state universal reversible PCA with finite configurations (Morita [21]).*

4.2 Two-dimensional universal RCAs

In the two-dimensional case, there are several simple universal RCAs. Margolus [13] first showed a universal model using a 2-state CA with block rules. Imai and Morita [7] gave a universal 8-state triangular RCA using the framework of PCAs as stated in the following theorem. In their model, each triangular cell has three parts with two states, and has an extremely simple local function.

Theorem 4.2 *(Imai and Morita [7]) The reversible 8-state triangular PCA having the local function shown in Fig. 7 with infinite configurations can simulate any* *reversible Turing machine.*

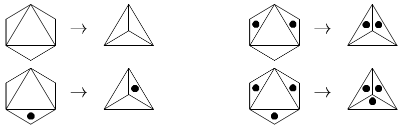

Fig. 7. The local function of a 2-dimensional 8-state triangular reversible PCA.

It is also possible to construct a universal reversible PCA that works on finite configurations. Morita et al. [18] gave a 81-state model in which any reversible counter machine [15] can be embedded. In their model, a rotary element can be simulated as in Fig. 8, and reversible counter machines are built from rotary elements and some other elements as shown in Fig. 9.

Theorem 4.3 *(Morita et al. [18]) There is a reversible 81-state PCA P_3 with finite configurations that can simulate any reversible counter machine.*

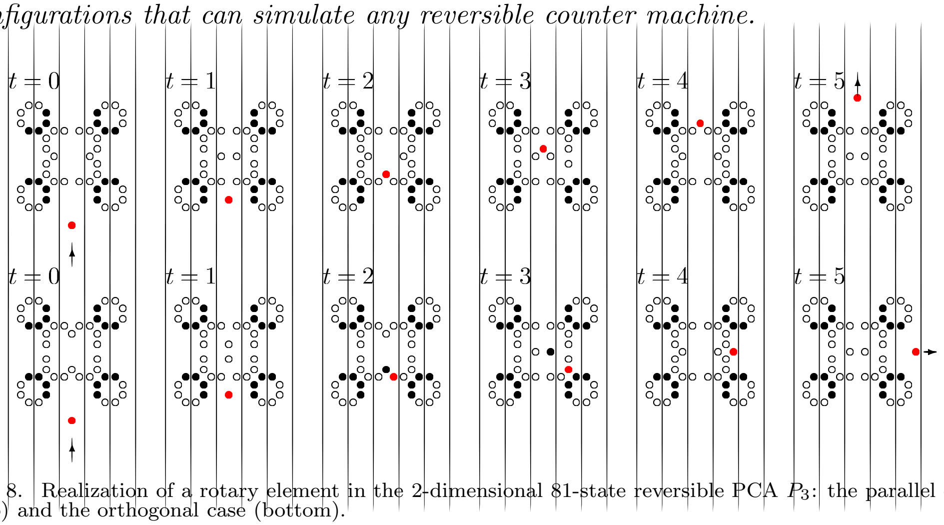

Fig. 8. Realization of a rotary element in the 2-dimensional 81-state reversible PCA P_3: the parallel case (top) and the orthogonal case (bottom).

5 Concluding remarks

We described how reversible systems can be computation-universal. We saw that in spite of the strong constraint of reversibility such systems have full computing power. We also argued that universal computing systems can be composed of very simple primitives such as 2-state reversible logic elements or reversible local functions of PCAs.

In this paper we presented three models of reversible computation, namely reversible Turing machines, reversible logic circuits, and reversible CAs. However, we omitted several other interesting topics concerning these models such as, for example, universality of 2-state reversible logic elements simpler than a rotary element [12,19,25], self-reproduction in reversible CAs [8,16], and firing squad synchronization in reversible CAs [6].

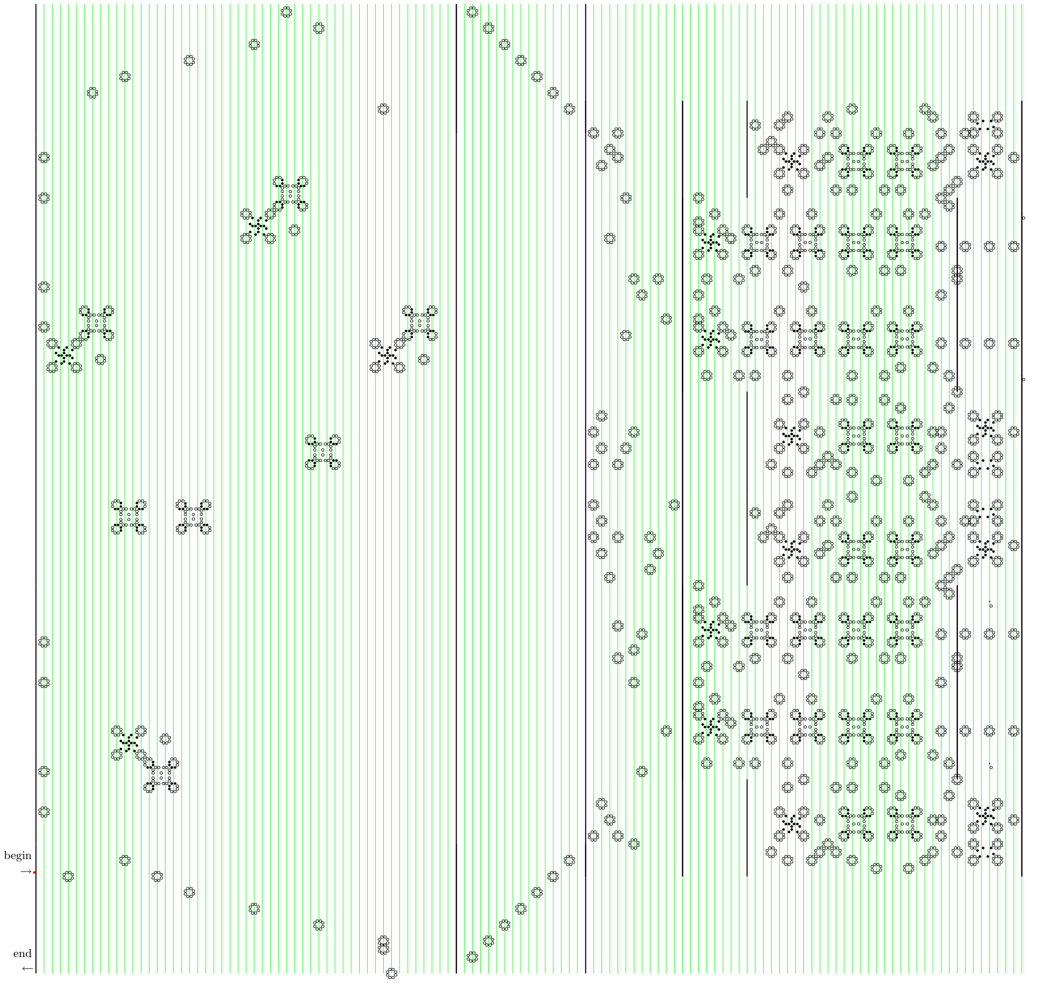

Fig. 9. An example of a reversible counter machine embedded in the 81-state reversible PCA P_3.

References

[1] Bennett, C.H., Logical reversibility of computation, *IBM J. Res. Dev.*, **17**, 525–532 (1973).

[2] Bennett, C.H., The thermodynamics of computation, *Int. J. Theoret. Phys.*, **21**, 905–940 (1982).

[3] Bennett, C.H., Notes on the history of reversible computation, *IBM J. Res. Dev.*, **32**, 16–23 (1988).

[4] Cook, M., Universality in elementary cellular automata, *Complex Systems*, **15**, 1–40 (2004).

[5] Fredkin, E., and Toffoli, T., Conservative logic, *Int. J. Theoret. Phys.*, **21**, 219–253 (1982).

[6] Imai, K., and Morita, K., Firing squad synchronization problem in reversible cellular automata, *Theoretical Computer Science*, **165**, 475–482 (1996).

[7] Imai, K., and Morita, K., A computation-universal two-dimensional 8-state triangular reversible cellular automaton, *Theoret. Comput. Sci.*, **231**, 181–191 (2000).

[8] Imai, K., Hori, T., and Morita, K., Self-reproduction in three-dimensional reversible cellular space, *Artificial Life*, **8**, 155–174 (2002).

[9] Kari, J., Reversibility and surjectivity problems of cellular automata, *J. Comput. Syst. Sci.*, **48**, 149–182 (1994).

[10] Kudlek, M., and Rogozhin, Y., A universal Turing machine with 3 states and 9 symbols, *Developments in Language Theory (DLT 2001)*, LNCS 2295, Springer-Verlag, 311–318 (2002).

[11] Lecerf, Y., Machines de Turing réversibles — Recursive insolubilité en $n \in \mathbf{N}$ de l'équation $u = \theta^n u$, où θ est un isomorphisme de codes, *Comptes Rendus Hebdomadaires des Séances de L'académie des Sciences*, **257**, 2597–2600 (1963).

[12] Lee, J., Peper, F., Adachi, S., and Morita, K., An asynchronous cellular automaton implementing 2-state 2-input 2-output reversed-twin reversible elements, *Proc. ACRI 2008*, Springer-Verlag, LNCS 5191, 67–76 (2008).

[13] Margolus, N., Physics-like model of computation, *Physica*, **10D**, 81–95 (1984).

[14] Morita, K., and Harao, M., Computation universality of one-dimensional reversible (injective) cellular automata, *Trans. IEICE Japan*, **E-72**, 758–762 (1989).

[15] Morita, K., Universality of a reversible two-counter machine, *Theoret. Comput. Sci.*, **168**, 303–320 (1996).

[16] Morita, K., and Imai, K., Self-reproduction in a reversible cellular space, *Theoret. Comput. Sci.*, **168**, 337–366 (1996).

[17] Morita, K., A simple reversible logic element and cellular automata for reversible computing, in *Proc. 3rd Int. Conf. on Machines, Computations, and Universality*, LNCS-2055, Springer-Verlag, 102–113 (2001).

[18] Morita, K., Tojima, Y., Imai, K., and Ogiro, T., Universal computing in reversible and number-conserving two-dimensional cellular spaces, in *Collision-based Computing* (ed. A. Adamatzky), 161–199, Springer-Verlag (2002).

[19] Morita, K., Ogiro, T., Tanaka, K., and Kato, H., Classification and universality of reversible logic elements with one-bit memory, *Proc. 4th Int. Conf. on Machines, Computations, and Universality*, LNCS-3354, Springer-Verlag, 245–256 (2005).

[20] Morita, K., and Yamaguchi, Y., A universal reversible Turing machine, *Proc. 5th Int. Conf. on Machines, Computations, and Universality* LNCS-4664, Springer-Verlag, 90–98 (2007).

[21] Morita, K., Simple universal one-dimensional reversible cellular automata, *J. Cellular Automata*, **2**, 159–165 (2007).

[22] Morita, K., Reversible computing and cellular automata — a survey, *Theoret. Comput. Sci.*, **395**, 101–131 (2008).

[23] Morita, K., A 24-state universal one-dimensional reversible cellular automaton, *Proc. of AUTOMATA 2008*, Bristol, Luniver Press, 106-112 (2008).

[24] Neary, T., and Woods, D., Four small universal Turing machines, *Proc. 5th Int. Conf. on Machines, Computations, and Universality* LNCS-4664, Springer-Verlag, 242–254 (2007).

[25] Ogiro, T., Kanno, A., Tanaka, K., Kato, H.. and Morita, K., Nondegenerate 2-state 3-symbol reversible logic elements are all universal, *Int. Journ. of Unconventional Computing*, **1**, 47–67 (2005).

[26] Petri, C.A., Grundsätzliches zur Beschreibung diskreter Prozesse, *Proc. 3rd Colloquium über Automatentheorie*, Birkhäuser Verlag, 121–140 (1967).

[27] Rogozhin, Y., Small universal Turing machines, *Theoretical Computer Science*, **168**, 215–240 (1996).

[28] Toffoli, T., Computation and construction universality of reversible cellular automata, *J. Comput. Syst. Sci.*, **15**, 213–231 (1977).

[29] Toffoli, T., Reversible computing, *Automata, Languages and Programming* (eds. J.W. de Bakker and J. van Leeuwen), LNCS-85, Springer-Verlag, 632–644 (1980).

[30] Toffoli, T., Bicontinuous extensions of invertible combinatorial functions, *Mathematical Systems Theory*, **14**, 12–23 (1981).

[31] Toffoli, T., and Margolus, N., Invertible cellular automata: a review, *Physica D*, **45**, 229–253 (1990).

[32] Woods, D., and Neary, T., The complexity of small universal Turing machines: A survey, *Theoret. Comput. Sci.*, **410**, 443–450 (2009).