A Short Note on Reversibility in P Systems

Artiom Alhazov[1,2], Kenichi Morita[1]

[1] IEC, Department of Information Engineering, Graduate School of Engineering
Hiroshima University, Higashi-Hiroshima 739-8527 Japan
morita@iec.hiroshima-u.ac.jp
[2] Institute of Mathematics and Computer Science
Academy of Sciences of Moldova, Academiei 5, Chişinău MD-2028 Moldova
artiom@math.md

Summary. Membrane computing is a formal framework of distributed parallel computing. In this paper we study the reversibility and maximal parallelism of P systems from the computability point of view. The notions of reversible and strongly reversible systems are considered. The universality is shown for one class and a negative conjecture is stated for a more restricted class of reversible P systems. For one class of strongly reversible P systems, a very strong limitation is found, and it is shown that this limitation does not hold for a less restricted class.

1 Introduction

Reversibility is an important property of computational systems. It has been well studied for circuits of logical elements ([3]), circuits of memory elements ([7]), cellular automata ([8]), Turing machines ([1], [10]), register machines ([6]). Reversibility as a syntactical property is closely related to the microscopic reversibility, so it implies that the computation does not increase the entropy of the system, which in turn assume better miniaturization possibilities for potential implementation.

A slightly different view on reversible systems is given for type-0 grammars ([9]). The so-called uniquely parsable grammars are studied. In very simple words, this property (still being syntactical) implies that the generation of any word in the language is unique (modulo the order of applying the rules in case when the composition of applying them is commutative). The advantage of having such a property is that it is easier to analyze their behavior.

Clearly, this reason remains valid even if the property of reversibility becomes undecidable (just like the property of determinism in certain membrane systems). Moreover, reversibility essentially is backward determinism. Reversible P systems already were considered ([4]), but the model is energy-based (so the parallelism is invariant-driven rather than maximal) and the main result is the simulation of the Fredkin gate (so construction of a universal system in this way would use an infinite structure). In this paper we focus on the interplay between reversibility and maximal parallelism, from the viewpoint of computability.

2 Definitions

In this paper we illustrate the reversibility concepts on P systems with symport/antiport and one membrane, sometimes with inhibitors or priorities. For simplicity, we also assume that the environment contains an unbounded supply of all objects. The system thus can be defined by the alphabet, the initial multiset, the set of rules associated to the membrane, and the set of terminal objects. Throughout this paper we represent multisets by strings. We write an antiport rule sending a multiset x out and bringing a multiset y in as x/y, and the symport case corresponds to $y = \lambda$. If a rule has inhibitor a, we write it as $x/y|_{\neg a}$. The priority relationship is denoted by $>$. It is not difficult to generalize the definitions for the models with multiple membranes and changing membrane structure, but it is not important here.

Consider a P system Π with alphabet O. In our setting, a configuration is defined by the multiset of objects inside the membrane, represented by some string $u \in O^*$. The space $\mathcal{C}$ of configurations is essentially $|O|$-dimensional space with non-negative integer coordinates. We use the usual definitions of maximally parallel transition ([11]). It induces an infinite graph of $\mathcal{C}$. Notice that the halting configurations (and only them) have out-degree zero.

We call Π **strongly reversible** if every configuration has in-degree at most one. We call Π **reversible** if every reachable configuration has in-degree at most one.

A property equivalent to reversibility is determinism of a dual P system ([2]).

The result of a halting computation is the number of terminal objects inside the membrane when the system halts. The set $N(\Pi)$ of numbers generated by a P system Π is the set of results of all its computations. The family of number sets generated by reversible P systems with features α is denoted by $NrOP_1(\alpha)_T$, where $\alpha \subseteq \{sym_*, anti_*, inh, Pri\}$ and the braces of the set notation are omitted. Subscript T means that only terminal objects contribute to the result of computations; if $T = O$, we may omit specifying it in the description and we then also omit the subscript T in the notation. To bound the weight (i.e., maximal number of objects sent in a direction) of symport or antiport rule, associated $*$ is replaced by the actual number. For strong reversible systems, we replace in the notation r by r_s.

2.1 Register machines

In this paper we consider register machines with increment, unconditional decrement and test instructions, see also [6].

A register machine is a tuple $M = (n, Q, q_0, q_f, I)$ where

- n is the number of registers;
- I is a set of instructions labeled by elements of Q;
- $q_0 \in Q$ is the initial label;
- $q_f \in Q$ is the final label.

The allowed instructions are:

- $(q : i?, q', q'')$ - jump to instruction q'' if the contents of register i is zero, otherwise proceed to instruction q';
- $(q : i+, q', q'')$ - add one to the contents of register i and proceed to either instruction q' or q'', non-deterministically;
- $(q : i-, q', q'')$ - subtract one from the contents of register i and proceed to either instruction q' or q'', non-deterministically;
- $(q_f : halt)$ - finish the computation; it is a unique instruction with label q_f.

If $q' = q''$ for every instruction $(q : i+, q', q'')$ and for every instruction $(q : i-, q', q'')$, then the machine is called deterministic.

A register machine is called reversible if for some state q there is more than one instruction leading to it, then exactly two exist, they test the same register, one leads to q if the register is zero and the other one leads to q if the register is positive. More formally, for any two different instructions $(q_1 : i_1\alpha_1, q_1', q_1'')$ and $(q_2 : i_2\alpha_2, q_2', q_2'')$, it holds that $q_1' \neq q_2'$ and $q_1'' \neq q_2''$. Moreover,

$$\text{if } q_1' = q_2'' \text{ or } q_1'' = q_2', \text{ then } \alpha_1 = \alpha_2 =? \text{ and } i_1 = i_2.$$

It has been shown ([6]) that reversible register machines are universal. It follows that non-deterministic reversible register machines can generate any recursively enumerable set of non-negative integers as a value of the first register by all its possible computations starting from all registers having zero value.

3 Examples and Universality

We now present a few examples to illustrate the definitions.

Example 0: Consider a P system $\Pi_0 = (\{a, b\}, a, \{a/ab\})$. It is strongly reversible (for a preimage, remove as many copies of b as there are copies of a, in case it is possible and there is at least one copy of a), but no halting configuration is reachable. Therefore, $\emptyset \in Nr_sOP_1(anti_*)$.

Example 1: Consider a P system $\Pi_1 = (\{a, b, c\}, a, \{a/ab,\ a/c\})$. It generates the set of positive integers and it is reversible (for the preimage, replace c with a or ab with b), but not strongly reversible (e.g., $aa \Rightarrow cc$ and $ac \Rightarrow cc$). Hence, $\mathbb{N}_+ \in NrOP(anti_2)$.

Example 2: Consider a P system $\Pi_2 = (\{a, b\}, aa, \{aa/ab,\ ab/bbb\})$. It is reversible ($aa$ has in-degree 0, while ab and bbb have in-degree 1, and no other configuration is reachable), but not strongly reversible (e.g., $aab \Rightarrow abbb$ and $aabb \Rightarrow abbb$).

Example 3: Any P system with a rule x/λ, $x \in O^+$, is not reversible. Therefore, symport rules cannot be actually used in a reversible P systems with one membrane.

Example 4: Any P system with rules x_1/y, x_2/y that applied at least one of them in some computation is not reversible.

We now show that reversible P systems with either inhibitors or priorities are universal.

Theorem 1. *$NrOP_1(anti_2, Pri)_T = NrOP_1(anti_2, inh)_T = NRE$.*

Proof. We reduce the theorem statement to the claim that such P systems simulate the work of any reversible register machine $M = (n, Q, q_0, q_f, I)$. Consider a P system

$$\begin{aligned} \Pi &= (O, q_0, R, \{r_1\}), \text{ where} \\ O &= \{r_i \mid 1 \leq i \leq n\} \cup Q, \\ R &= \{q/q'r_i, q/q''r_i \mid (q : i+, q', q'') \in I\} \\ &\quad \cup \{qr_i/q', qr_i/q'' \mid (q : i-, q', q'') \in I\} \cup R_t, \\ R_t &= \{q/q''|_{\neg r_i}, qr_i/q'r_i \mid (q : i?, q', q'') \in I\}. \end{aligned}$$

Inhibitors can be replaced by priorities by redefining R_t as follows.

$$R_t = \{qr_i/q'r_i > q/q'' \mid (q : i?, q', q'') \in I\}.$$

Since there is a bijection between the configurations of Π containing one symbol from Q and the configurations of M, the reversibility of Π follows from the correctness of simulation, the reversibility of M, and from the fact that the number of symbols from Q is preserved by transitions of Π.

4 Limitations

The construction in the theorem above uses both cooperation and additional control. It is natural to ask whether both inhibitors and priorities can be avoided. Yet, consider the following situation. Let $(p : i?, s, q''), (q : i?, q', s) \in I$. It is usual for reversible register machines to have this, since the preimage of configuration sC depends on register i. Nevertheless, P systems with maximal parallelism without additional control can only implement a zero-test by try-and-wait-then-check strategy. In this case, the object containing the information about the register p finds out the result of checking after a possible action of the object related to the register. Therefore, when the state represented in the configuration of the system changes to s, it obtains an erroneous preimage representing state q. This leads to the following

Conjecture 1. Reversible P systems without priorities and without inhibitors are not universal.

Now consider a strongly reversible P system. The following theorem establishes a very serious limitation on such systems if no additional control is used.

Theorem 2. *In strongly reversible P systems without inhibitors and without priorities, every configuration is either halting or induces only infinite computation(s).*

Proof. If the right-hand side of every rule contains a left-hand side of some rule, then the claim holds. Otherwise, let x/y be a rule of the system such that y does not contain the left-hand side of any rule. Then $x \Rightarrow y$ and y is a halting configuration. It is not difficult to see that $xy \Rightarrow yy$ (objects y are idle) and $xx \Rightarrow yy$ (the rule can be applied twice). Therefore, such a system is not strongly reversible, which proves the theorem.

Therefore, the strongly universal systems without additional control can only generate singletons, i.e., $Nr_sOP_1(anti_*)_T = \{\{n\} \mid n \in \mathbb{N}\}$, and only in a degenerate way, i.e., without actually computing.

It turns out that the theorem above does not hold if inhibitors are used. Consider a system $\Pi_3 = (\{a, b\}, a, \{a/b|_{\neg b}\})$. If at least one object b is present or no objects a are present, such a configuration is a halting one. Otherwise, all objects a are exchanged by objects b. Therefore, the only possible transitions in the space of all configurations are of the form $a^n \Rightarrow b^n$, $n > 1$, and the system is strongly reversible.

5 Discussion

We outlined the concepts of reversibility and strong reversibility for P systems, concentrating on the case of symport/antiport rules (possibly with control such as priorities or inhibitors) with one membrane, assuming that the environment contains an unbounded supply of all objects.

We showed that reversible P systems with control are universal, and we conjectured that this result does not hold without control. Moreover, the strongly reversible P systems without control do not halt unless the starting configuration is halting, but this is no longer true if inhibitors are used.

Showing related characterizations might be quite interesting. Many other problems are still open, e.g., reversibility of P systems with active membranes.

Acknowledgments

Artiom Alhazov gratefully acknowledges the support of the Japan Society for the Promotion of Science and the Grant-in-Aid for Scientific Research, project 20·08364. He also acknowledges the support by the Science and Technology Center in Ukraine, project 4032.

References

1. C.H. Bennett: Logical reversibility of computation. *IBM J. Res. Dev.*, 17 (1973), 525-532.
2. O. Agrigoroaiei, G. Ciobanu: Dual P systems. *Membrane Computing - 9th International Workshop*, LNCS 5391, 2009, 95–107.
3. E. Fredkin, T. Toffoli: Conservative logic. *Int. J. Theoret. Phys.*, 21 (1982), 219-253.
4. A. Leporati, C. Zandron, G. Mauri: Reversible P systems to simulate Fredkin circuits. *Fundam. Inform.*, 74, 4 (2006), 529–548.
5. M.L. Minsky: *Computation: Finite and Infinite Machines.* Prentice-Hall, Englewood Cliffs, NJ, 1967.
6. K. Morita: Universality of a reversible two-counter machine. *Theoret. Comput. Sci.*, 168 (1996), 303-320.
7. K. Morita: A simple reversible logic element and cellular automata for reversible computing. *Proc. 3rd Int. Conf. on Machines, Computations, and Universality*, LNCS 2055, 2001, 102-113.
8. K. Morita: Simple universal one-dimensional reversible cellular automata. *J. Cellular Automata*, 2 (2007), 159-165.
9. K. Morita, N. Nishihara, Y. Yamamoto, Zh. Zhang: A hierarchy of uniquely parsable grammar classes and deterministic acceptors. *Acta Inf.*, 34, 5 (1997), 389–410.
10. K. Morita, Y. Yamaguchi: A universal reversible Turing machine. *Proc. 5th Int. Conf. on Machines, Computations, and Universality*, LNCS 4664, 2007, 90-98.
11. Gh. Păun: *Membrane Computing. An Introduction.* Springer, Berlin, 2002.
12. P systems website: `http://ppage.psystems.eu/`.