MEMBRANE SYSTEMS AND MULTISET APPROXIMATION

Péter Battyányi György Vaszil

Department of Computer Science, Faculty of Informatics, University of Debrecen
Kassai út 26, 4028 Debrecen, Hungary
{battyanyi.peter,vaszil.gyorgy}@inf.unideb.hu

Abstract

We consider membrane systems where the applicability of the multiset transformation rules is determined by the approximating multisets of the membrane regions. We consider two cases: First, we study systems with inner rules where we allow only rule applications such that the multisets involved in the rules are part of the lower approximation of the respective regions, then we consider systems with boundary rules where rule application is defined on the boundaries, that is, rules can only manipulate the elements outside of the lower approximation. We show that the second variant benefits from the underlying approximation framework by demonstrating an increase in its computational strength. On the other hand, the computational power of systems with inner rule application remains weaker than that of Turing machines (as long as the unsynchronized version is considered).

1. Introduction

Membrane systems, introduced in [15], are biologically inspired models of computation: their operation imitates in a sense the functioning of living cells. The computation proceeds in distinct regions, called membranes or compartments. The compartments allow computation with multisets: they accomplish transformations of their contained multisets by various evolution (multiset rewriting) rules. Several variants of P systems have been introduced and studied, see the monograph [16] for a thorough introduction, or the handbook [17] for a summary of notions and results of the area.

The structure of a membrane system can be represented in various ways, cell-like membrane systems have a membrane structure which can be described by a tree. Systems with graph-like membrane structures called tissue-like P systems were also considered, where the connection between the membranes are established by edges forming the communication routes. Here we study variants of tissue-like systems called generalized P systems (see [3]).

An extended version of this paper was presented at the International Joint Conference on Rough Sets (IJCRS 2019), Debrecen, Hungary, June 17-21, 2019.

Gy. Vaszil was supported by grant K 120558 of the National Research, Development and Innovation Office of Hungary (NKFIH), financed under the K 16 funding scheme.

The question of how to define dynamically changing membrane structures using topological spaces, and how the underlying topologies influence the behaviour of P systems was already examined in [5, 6]. Multiset approximation spaces were defined in [8, 9], which made it possible to talk about lower and upper approximations of the contents of membranes of a P system. This led to various notions of membrane borders, and notions of closeness of membranes. Restricting the interaction to membranes that are close to each other, or permitting only rules that manipulate multisets which are on the boundaries of the membranes can affect the computational strength of the membrane system. The study of this area was initiated in [10], where also an intention to model chemical stability played an important role. The results in [10] were formulated for the so-called symport/antiport P systems, but the investigations were also continued for so called generalized P systems in [2]. In the present paper we also study generalized P systems, but we do not rely on any notion of closeness of membranes. Instead, we focus on the notion of clear observability. We consider lower approximations and boundaries of compartments, and restrict the applicability of the rules accordingly. It will turn out that the use of boundary rules, that is, rules which can only manipulate objects on the boundaries of compartments, results in an increase of the computational power of certain variants of generalized P systems to the level of the power of Turing machines. On the other hand, if we restrict rule applications only to rules that manipulate multisets which lie in the inner approximations of the membranes (inner rules), this restriction is not enough to provide Turing completeness.

In the following, we first recall the necessary definitions, then take up the examination of the two variants of generalized P systems with dynamically changing communication structure based on multiset approximation spaces. As maximal parallel rule application makes already the basic model of generalized P systems computationally complete, we study the weaker, unsynchronized variants. We first show that generalized P systems with inner rules can be simulated by simple place-transition Petri nets, thus, their computational power is less than that of Turing machines. Then we consider systems with boundary rules and show that they are able to simulate so called register machines, which demonstrates that their computational power is the same as the power of Turing machines. Finally, the paper ends with a few concluding remarks.

2. Preliminaries

Let $\mathbb{N}$ and $\mathbb{N}_{>0}$ be the set of non-negative integers and the set of positive integers, respectively, and let O be a finite nonempty set (the set of object). A *multiset* M over O is a pair $M = (O, f)$, where $f : O \to \mathbb{N}$ is a mapping which gives the *multiplicity* of each object $a \in O$. The set $\mathrm{supp}(M) = \{a \in O \mid f(a) > 0\}$ is called the *support* of M. If $\mathrm{supp}(M) = \emptyset$, then M is the empty multiset. If $a \in \mathrm{supp}(M)$, then $a \in M$, and $a \in^n M$ if $f(a) = n$.

Let $M_1 = (O, f_1), M_2 = (O, f_2)$. Then $(M_1 \sqcap M_2) = (O, f)$ where $f(a) = \min\{f_1(a), f_2(a)\}$; $(M_1 \sqcup M_2) = (O, f')$, where $f'(a) = \max\{f_1(a), f_2(a)\}$; $(M_1 \oplus M_2) = (O, f'')$, where $f''(a) = f_1(a) + f_2(a)$; $(M_1 \ominus M_2) = (O, f''')$ where $f'''(a) = \max\{f_1(a) - f_2(a), 0\}$; and $M_1 \sqsubseteq M_2$, if $f_1(a) \leq f_2(a)$ for all $a \in O$.

For any $n \in \mathbb{N}$, n-times addition of M, denoted by $\oplus_n M$, is given by the following inductive definition:

- $\oplus_0 M = \emptyset$;
- $\oplus_1 M = M$;
- $\oplus_{n+1} M = (\oplus_n M) \oplus M$.

Let $M_1 \neq \emptyset, M_2$ be two multisets. For any $n \in \mathbb{N}$, $M_1 \sqsubseteq^n M_2$, if $\oplus_n M_1 \sqsubseteq M_2$ but $\oplus_{n+1} M_1 \not\sqsubseteq M_2$.

The number of copies of objects in a finite multiset $M = (O, f)$ is its cardinality: $\text{card}(M) = \Sigma_{a \in \text{supp}(M)} f(a)$. Such an M can be represented by any string w over O for which $|w| = \text{card}(M)$, and $|w|_a = f(a)$ where $|w|$ denotes the length of the string w, and $|w|_a$ denotes the number of occurrences of symbol a in w.

We define $\mathcal{MS}^n(O)$, $n \in \mathbb{N}$, to be the set of all multisets $M = (O, f)$ over O such that $f(a) \leq n$ for all $a \in O$, and we let $\mathcal{MS}^{<\infty}(O) = \bigcup_{n \geq 0} \mathcal{MS}^n(O)$.

2.1. Generalized P Systems

Now we present the notion of generalized P systems, variants of tissue P systems introduced in [3].

An $(n+3)$-tuple $\Pi = (O, w_1, w_2, \ldots, w_n, R, i_o)$ is a *generalized P system* of degree $n \geq 1$, where

- O is a finite set of objects;
- $w_i \in \mathcal{MS}^{<\infty}(O)$, $1 \leq i \leq n$, is a finite multiset of objects, the initial contents of the ith region of Π;
- R is a finite set of transformation rules of the form $(x_1, \alpha_1) \ldots (x_k, \alpha_k) \rightarrow (y_1, \beta_1) \ldots (y_l, \beta_l)$, where $x_i, y_j \in \mathcal{MS}^{<\infty}(O)$, and $1 \leq \alpha_i, \beta_j \leq n$ indicate labels of the regions of the system for all $1 \leq i \leq k,\ 1 \leq j \leq l$;
- $1 \leq i_o \leq n$ is the label of the output compartment.

The rules of a generalized P system can be considered to model interactions of objects simultaneously affecting several regions of the membrane system. Thus, the links between participating compartments are defined dynamically, through the applicability of the rules by the functioning of the system.

Given a generalized P system Π as above, a *configuration* of Π is an n-tuple $c = (u_1, u_2, \ldots, u_n)$ with $u_i \in MS^{<\infty}(O)$, $1 \leq i \leq n$, and $c_0 = (w_1, w_2, \ldots, w_n)$ is called its *initial configuration.* The multisets $u_1, u_2, \ldots, u_n$ are the *contents* of the corresponding compartments $1, 2, \ldots, n$, in configuration c.

A generalized P system changes its configurations by applying its rules. In the basic setting, a rule $r \in R$ is *applicable* to a configuration c if and only if x_i is a submultiset of u_{α_i} for all $1 \leq i \leq k$. As a result of applying r to c, each multiset x_i is removed from the region u_{α_i}, $1 \leq i \leq k$, and each multiset y_j is added to the region u_{β_j}, $1 \leq j \leq l$.

The configuration $c' = (v_1, \ldots, v_n)$ of Π is obtained directly from the configuration $c = (u_1, \ldots, u_n)$ by applying the rules in the *unsynchronized* manner, if there is a multiset R' of rules from R, such that all of them are simultaneously applicable to different copies of objects in configuration c, and the configuration c' is the result of the application of the rules in R'. The configuration c' is obtained from c by applying the rules in the *maximally parallel* manner, if we add the additional requirement that the set R' is maximal, that is, for any $r \in R$, the rules in the rule multiset $\{r\} \oplus R'$ are not simultaneously applicable to c.

A sequence of configurations $c_0, c_1, \ldots$ of Π is called a *computation* if each configuration in the sequence is obtained directly from the previous one, starting from the initial configuration. Computations halt if no rule can be applied, the result of a *halting computation* is the number of objects that are present in the output compartment (compartment i_o) in the halting configuration.

2.2. Multiset Approximation Spaces

There are different ways of approximating sets originating in rough set theory proposed in the early 1980's, [11, 12]. The theory and its different generalizations uses different kinds of indiscernibility relations to provide lower and upper approximations of sets. An indiscernibility relation on a given set of objects is given by a set of base sets by which lower and upper approximations can be constructed for any set. This way of set approximation was generalized to partial set approximation in [4], giving the possibility to embed available knowledge into an approximation space. The lower and upper approximations also rely on base sets which can be thought of as representants of the available knowledge. Having the concepts of lower and upper approximations, we can also introduce the concept of boundary as the difference between these two.

A multiset approximation space over a finite alphabet O consists of the following:

- A *domain*: in our case it is $\mathcal{MS}^{<\infty}(O)$, the set of finite multisets over some finite set O. The elements of the domain are approximated using the approximation space.
- A *base system*: $\mathfrak{B} \subseteq \mathcal{MS}^{<\infty}(O)$, a nonempty set of finite *base multisets* providing the basis for the approximation process.
- The *approximation functions*: $\mathsf{l}, \mathsf{u}, \mathsf{b} : \mathcal{MS}^{<\infty}(O) \to \mathcal{MS}^{<\infty}(O)$ determining the lower and upper approximations (and the boundaries) of multisets of the domain.

A *multiset approximation space* is a quintuple $(O, \mathfrak{B}, \mathsf{l}, \mathsf{u}, \mathsf{b})$ where O is a finite set, $\mathfrak{B} \subseteq \mathcal{MS}^{<\infty}(O)$ is a base system (a set of base multisets), and $\mathsf{b}, \mathsf{u}, \mathsf{l} : \mathcal{MS}^{<\infty}(O) \to \mathcal{MS}^{<\infty}(O)$ are the approximation functions generated by $\mathfrak{B}$.

For any multiset $M = (O, f) \in \mathcal{MS}^{<\infty}(O)$, we define the *lower approximation function*:

$$\mathsf{l}(M) = \bigsqcup \{\oplus_n B \mid B \in \mathfrak{B},\ B \sqsubseteq M, \text{and } B \sqsubseteq^n M\},$$

the *boundary function*:

$$\mathsf{b}(M) = \bigsqcup\{\oplus_n B \mid B \in \mathfrak{B},\ \text{and } B \sqcap (M \ominus \mathsf{l}(M)) \sqsubseteq^n M \ominus \mathsf{l}(M)\},$$

and the *upper approximation function*:

$$\mathsf{u}(M) = \mathsf{l}(M) \oplus \mathsf{b}(M).$$

In addition, we also define $\mathsf{b}^{\mathsf{e}}(M) = \mathsf{b}(M) \ominus M$ as the *external part* of the boundary of M, and $\mathsf{b}^{\mathsf{i}}(M) = \mathsf{b}(M) \sqcap M$, the *internal part* of the boundary of M.

Intuitively, we can think of the lower approximation of the multiset M as the collection of elements that can be covered by the base multisets in such a way that the covering is inside M completely. If we also cover those elements of M that are left out of the lower approximation, then the union of the covering base sets contains M, thus, it can be thought of as the upper approximation of M, while the difference between the upper and the lower approximations of M is the boundary.

3. Regulating Rule Application in the Multiset Approximation Framework

In [2] we considered P systems with dynamical structure where the dynamic character of the membrane system was encoded in the reformulation of the region structure regarding a closeness property defined among the membranes based on the actual configuration of the system. Here we examine questions that arise when we require that in order for a rule to be applicable, the multisets on its lefthand side must conform to certain properties defined in the multiset approximation framework associated to the system. We discuss the following two approaches: first we require that a rule to be applied should only work with the lower approximations of the compartments' contents. The second approach demands that the multisets on the lefthand sides of the rules should come from the boundaries of the respective compartments.

Conforming the requirement of clear observability when dealing with rough sets, first we stipulate in the following definition that a rule should be applicable in a P system only if the multisets on its lefthand side come from the inner approximations of the containing regions, this means that we are absolutely sure that the rule application affects elements of the corresponding regions. The second requirement, on the other hand, corresponds to a system where rule application can only alter those elements about which our knowledge is vague, so the configuration changes of these systems might be thought of as steps in the direction of reducing vagueness, obtaining more and more determinate knowledge about the objects distributed in the membranes.

We formalize these notions in the following definition.

Definition 3.1 Let $\Pi = (O, \mathfrak{B}, w_1, w_2, \ldots, w_n, R, i_o)$ where $\mathfrak{B} \subseteq \mathcal{MS}^{<\infty}(O)$ is a base system and $(O, w_1, w_2, \ldots, w_n, R, i_o)$ is a generalized P system.

We call Π a *generalized P system with an associated multiset approximation space and inner rules*, if the applicability of a rule $r = (x_1, \alpha_1) \dots (x_k, \alpha_k) \to (y_1, \beta_1) \dots (y_l, \beta_l) \in R$ in a configuration $c = (u_1, \dots, u_n)$ is defined by the requirement that x_i is a submultiset of $\mathsf{l}(u_{\alpha_i})$, the inner approximation of the respective region, $1 \leq i \leq k$. If $r \in R$ is applicable to c in this sense, then we call r an *inner rule* (with respect to c).

We call Π a *generalized P system with an associated multiset approximation space and boundary rules*, if the applicability of a rule $r = (x_1, \alpha_1) \dots (x_k, \alpha_k) \to (y_1, \beta_1) \dots (y_l, \beta_l) \in R$ in a configuration $c = (u_1, \dots, u_n)$ is defined by the requirement that x_i is a submultiset of $\mathsf{b}^{\mathsf{i}}(u_{\alpha_i})$, the internal part of the boundary of the respective region, $1 \leq i \leq k$. If $r \in R$ is applicable to c in this sense, then we call r a *boundary rule* (with respect to c).

Example 3.2 *Assume that $C = (w_1, w_2)$ is the initial configuration of a generalized P system with an associated multiset approximation space $\Pi = (\{a, b, c, d\}, \{B_1, B_2\}, w_1, w_2, \{r_1, r_2\}, i_o)$, with $w_1 = a^3b^3c^2$ and base sets $B_1 = a^2$, $B_2 = bc$. Further, let $r_1 = (ab^2, 1) \to (c, 1)(d^3, 2)$ and $r_2 = (ab, 1) \to (e^2, 1)$.*

If Π is a system with inner rules, then both rules are applicable in C, as $B_1 \sqcup \oplus_2 B_2 = a^2b^2c^2$ is the lower approximation of w_1.

If Π is a system with boundary rules, then only the rule r_2 is applicable in C, as a^2bc is the boundary of w_1 with inner part ab.

We claim that the use of inner rules do not add much to the computational strength of the P system in the sense that in the non-synchronized mode a generalized P system with an associated multiset approximation space and inner rules is not Turing complete. To show this, we might construct a simple place-transition Petri net that simulates the P system in question. This is sufficient, because Petri nets in this simple setting are strictly weaker in computational power than Turing machines, see for example [13, 14].

Theorem 3.3 *For any generalized P system Π with an associated multiset approximation space and inner rules, there is a place-transition Petri net N, such that N generates the same set of numbers as Π in the unsynchronized manner of rule application.*

As we have already mentioned, the expressive power of place-transition Petri nets are less than that of Turing machines, so we obtain the following corollary.

Corollary 3.4 *Generalized membrane systems with multiset approximation spaces and inner rules using the unsynchronized manner of rule application are strictly weaker in computational power than Turing machines, that is, they are not computationally complete.*

Now we continue with the investigation of the case of boundary rules. We show that generalized P systems with boundary rules generate any recursively enumerable set of numbers. We might do this by demonstrating how these systems simulate the computations of register machines, a computational model equivalent in power to Turing machines.

Theorem 3.5 *Generalized P systems with associated multiset approximation spaces and boundary rules generate any recursively enumerable set of numbers, even in the unsynchronized manner of rule application.*

4. Concluding Remarks

We have used multiset approximation spaces to restrict the applicability of multiset evolution rules of generalized P systems. This way we incorporated some additional "dynamics" into the system, as not only the presence or absence of elements, but also the underlying approximation spaces have a role in determining the applicability of the rules.

It turned out that restricting the operation of the rules to the boundaries of compartments increases the computational power of generalized P systems, as they are able to generate any recursively enumerable sets of numbers even in the unsynchronized manner of rule application. On the other hand, a similar restriction allowing the rules to manipulate only elements of the lower approximation of the compartments of the system does not result in a similar increase of the computational power.

As a final remark, we would like to add some thoughts on a related model called P systems with anti-matter [1, 7]. In P systems with anti-matter, objects have complementary "anti objects", and when they are both present, they annihilate (disappear). In this paper we considered boundary rules which cannot be applied to objects that are not on the boundary: when all the elements of a base multiset are present in a region, they "disappear" from the scope of boundary rules. This effect is similar to the effect of annihilation rules, although not exactly the same. The difference can be seen from a simple example: let two base multisets be $ab, bc \in \mathfrak{B}$. The fact that they form base multisets is not directly modeled by the annihilation rules $ab \to \varepsilon, bc \to \varepsilon$ (as used in the case of P systems with anti-matter), because of the following. If a region contains ab, then these are "invisible" for the boundary rules, but they are not annihilated, as can be seen when an object c enters the region. As bc is also a base multiset, c immediately "disappears" by becoming part of the inner, lower approximation part of the region contents. As we see, the relationship of boundary rules and anti-matter is not as simple as it might look, but it definitely seems to be an interesting topic for further investigations.

References

[1] A. ALHAZOV, B. AMAN, R. FREUND, P systems with anti-matter. In: M. GHEORGHE, G. ROZENBERG, A. SALOMAA, P. SOSÍK, C. ZANDRON (eds.), *Membrane Computing*. Springer, 2014, 66–85.

[2] P. BATTYÁNYI, T. MIHÁLYDEÁK, GY. VASZIL, Generalized membrane systems with dynamical structure, Petri nets, and multiset approximation spaces. In: I. MCQUILLAN, S. SEKI (eds.), *18th International Conference on Unconventional Computation and Natural Computation. UCNC 2019*. Springer, 2019, 15–29.

[3] F. BERNARDINI, M. GHEORGUE, M. MARGENSTERN, S. VERLAN, Networks of cells and Petri nets. In: D. DÍAZ-PERNIL, C. GRACIANI, M. A. GUTIÉRREZ-NARANJO, GH. PĂUN, I. PÉREZ-HURTADO, A. RISCOS-NÚÑEZ (eds.), *Proceedings of the Fifth Brainstorming Week on Membrane Computing*. Fénix Editora, Sevilla, 2007, 33–62.

[4] Z. CSAJBÓK, T. MIHÁLYDEÁK, Partial approximative set theory: a generalization of the rough set theory. *International Journal of Computer Information Systems and Industrial Management Applications* 4 (2012), 437–444.

[5] E. CSUHAJ-VARJÚ, M. GHEORGHE, M. STANNETT, P systems controlled by general topologies. In: J. DURAND-LOSE, N. JONOSKA (eds.), *Unconventional Computation and Natural Computation*. Springer, 2012, 70–81.

[6] E. CSUHAJ-VARJÚ, M. GHEORGHE, M. STANNETT, GY. VASZIL, Spatially localised membrane systems. *Fundamenta Informaticae* 138 (2015) 1-2, 193–205.

[7] A. LEPORATI, L. MANZONI, G. MAURI, A. E. PORRECA, C. ZANDRON, The counting power of P systems with antimatter. *Theoretical Computer Science* 701 (2017), 161 – 173.

[8] T. MIHÁLYDEÁK, Z. E. CSAJBÓK, Membranes with boundaries. In: E. CSUHAJ-VARJÚ, M. GHEORGHE, G. ROZENBERG, A. SALOMAA, GY. VASZIL (eds.), *Membrane Computing*. Springer, 2013, 277–294.

[9] T. MIHÁLYDEÁK, Z. E. CSAJBÓK, On the membrane computations in the presence of membrane boundaries. *Journal of Automata, Languages and Combinatorics* 19 (2014) 1, 227–238.

[10] T. MIHÁLYDEÁK, GY. VASZIL, Regulating rule application with membrane boundaries in P systems. In: G. ROZENBERG, A. SALOMAA, J. M. SEMPERE, C. ZANDRON (eds.), *Membrane Computing*. Springer, 2015, 304–320.

[11] Z. PAWLAK, Rough sets. *International Journal of Computer & Information Sciences* 11 (1982) 5, 341–356.

[12] Z. PAWLAK, *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, 1991.

[13] J. L. PETERSON, *Petri Net Theory and the Modeling of Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1981.

[14] L. POPOVA-ZEUGMANN, *Time and Petri Nets*. Springer, 2013.

[15] GH. PĂUN, Computing with Membranes. *Journal of Computer and System Sciences* 61 (2000) 1, 108–143.

[16] GH. PĂUN, *Membrane Computing: An Introduction*. Springer, 2002.

[17] GH. PĂUN, G. ROZENBERG, A. SALOMAA, *The Oxford Handbook of Membrane Computing*. Oxford University Press, Inc., New York, NY, USA, 2010.