Decidability of Divergence for Catalytic P Systems

Nadia Busi

Dipartimento di Scienze dell'Informazione
Università di Bologna
Mura A. Zamboni 7, I-40127 Bologna, Italy
busi@cs.unibo.it

Summary. P systems are a biologically inspired model introduced by Gheorghe Păun with the aim of representing the structure and the functioning of the cell. Since their introduction, several variants of P systems have been proposed and explored.

We concentrate on the class of catalytic P systems without priorities associated to the rules. We show that the divergence problem (i.e., checking for the existence of an infinite computation) is decidable in such a class of P systems.

As a corollary, we obtain an alternative proof of the nonuniversality of deterministic catalytic P systems, an open problem recently solved by Ibarra and Yen.

1 Introduction

Membrane computing is a branch of natural computing, initiated by Gheorghe Păun with the definition of P systems in [11, 12, 13]. The aim is to provide a formal modeling of the structure and the functioning of the cell, making use especially of automata, languages and complexity theoretic tools.

Quoting from the Milano webpage [17], "A P system is a computing model which abstracts from the way the alive cells process chemical compounds in their compartmental structure. In short, in the regions defined by a membrane structure we have objects which evolve according to given rules. The objects can be described by symbols or by strings of symbols (in the former case their multiplicity matters, that is, we work with multisets of objects placed in the regions of the membrane structure; in the second case we can work with languages of strings or, again, with multisets of strings). By using the rules in a nondeterministic, maximally parallel manner, one gets transitions between the system configurations. A sequence of transitions is a computation. With a halting computation we can associate a result, in the form of the objects present in a given membrane in the halting configuration, or expelled from the system during the computation. Various ways of controlling the transfer of objects from a region to another one and of applying the rules, as well as possibilities to dissolve, divide, create, or move membranes were considered."

Since their introduction, plenty of variants of P systems have been introduced, and a lot of research effort has been carried out, especially concerned with the study of the expressivity and the universality of the proposed models and with the ability to solve NP-complete problems in polynomial time.

In this paper we concentrate on catalytic P systems, namely systems whose rules are of one of the following kinds:

- context–free rules, with the form $a \to v$ and representing the fact that an instance of a is consumed and the objects in v are produced, or
- catalytic rules, with the form $ca \to cv$ and representing the fact that a is consumed and the objects in v are produced, provided that an instance of catalyst c is present inside the membrane, and that such an instance of c has not been used yet by another concurrently executed rule. As we will see in Section 3, catalysts are neither produced nor consumed by evolution rules, but they are used to bound the number of instances of catalytic rules applied in a maximal parallelism step.

The computational power of catalytic P systems has been tackled in various papers: in [12] the universality of P systems with catalysts and priorities is proved; the result has been improved in [15, 16] by showing that priorities are not necessary for universality. In [5, 4] some minimality properties for universality are investigated; in particular, [4] shows that two catalysts are sufficient to get universality.

However, the encodings presented in the aforementioned works present a high degree of nondeterminism. For example, in [4] an encoding of deterministic Minsky's register machines [10] is provided, that satisfies the following properties:

- if the register machine halts, then the encoding of the register machine has a halting computation (but there could also be other nonterminating computations);
- if the register machine does not halt, then all the computations of the encoding are nonterminating.

A corollary of this result is the undecidability of the existence of an halting computation for catalytic P systems, but the nondeterministic nature of the encoding leaves the door open to the possibility for other properties to be decidable. This fact is quite relevant from the point of view of systems biology, where the interest is to predict the behavior of the living matter.

In the present paper we show that divergence, namely, the existence of a nonterminating computation, is decidable for catalytic P systems. The proof is based on the theory of well-structured transition systems [3]: the existence of an infinite computation starting from a given state is decidable for finitely branching transition systems, provided that the set of states can be equipped with a well-quasi-ordering, i.e., a quasi-ordering relation which is compatible with the transition relation and such that each infinite sequence of states admits an increasing subsequence. To this aim, we define a quasi-ordering on the configurations of catalytic P systems that turns out to be a well-quasi-ordering compatible with the maximally parallel evolution rule.

A consequence of this result is the impossibility to provide a deterministic encoding of any Turing powerful formalism in catalytic P systems, as for deterministic systems divergence and the existence of an halting computation are equivalent properties. This yields to an alternative proof of the nonuniversality of deterministic catalytic P systems, an open problem raised in [6, 14] and recently solved by Ibarra and Yen [8, 9].

The paper is organized as follows. After providing some basic definitions in Section 2, in Section 3 we define catalytic P systems. Section 4 is devoted to recalling some basic notions and results concerning well-quasi-orderings and well-structured transition systems that will be used in the following. Section 5 is devoted to the decidability result: after providing a finer notion of the configurations and evolution rules of catalytic P systems that is suitable to our aims, we define a quasi-ordering relation on the configurations and show that it turns out to be a well-quasi-ordering compatible with the maximally parallel evolution rule, and finally we make use of the theory of well-structured transition systems to get the decidability of divergence. Some conclusive remark is reported in Section 6.

2 Basic definitions

In this section we provide some definitions that will be used throughout the paper.

We start with the definition of multisets and multiset operations.

Definition 1. *Given a set S, a* finite multiset *over S is a function $m : S \to I\!\!N$ such that the set $dom(m) = \{s \in S \,|\, m(s) \neq 0\}$ is finite. The* multiplicity *of an element s in m is given by the natural number $m(s)$. The set of all finite multisets over S, denoted by $\mathcal{M}_{fin}(S)$, is ranged over by m. A multiset m such that $dom(m) = \emptyset$ is called* empty*. The empty multiset is denoted by $\emptyset$.*

Given the multiset m and m', we write $m \subseteq m'$ if $m(s) \leq m'(s)$ for all $s \in S$ while $\oplus$ denotes their multiset union*: $m \oplus m'(s) = m(s) + m'(s)$. The operator $\setminus$ denotes* multiset difference*: $(m \setminus m')(s) = \mathit{if}\ m(s) \geq m'(s)\ \mathit{then}\ m(s) - m'(s)\ \mathit{else}\ 0$. The* scalar product*, $j \cdot m$, of a number j with m is $(j \cdot m)(s) = j \cdot (m(s))$. The cardinality of a multiset is the number of occurrences of elements contained in the multiset: $|m| = \sum_{s \in S} m(s)$.*

The set of parts of a set S is defined as $\mathcal{P}(S) = \{X \mid X \subseteq S\}$.

The restriction to a subset of a multiset is defined as follows:

Definition 2. *Let m be a finite multiset over S and $X \subseteq S$. The multiset $m|_X$ is defined as follows: for all $s \in S$,*

$$m|_X(s) = \begin{cases} m(s) & \mathit{if}\ s \in X, \\ 0 & \mathit{otherwise.} \end{cases}$$

We provide some basic definitions on strings, cartesian products and relations.

Definition 3. *A string over S is a finite (possibly empty) sequence of elements in S. Given a string $u = x_1 \dots x_n$, the length of u is the number of occurrences of elements contained in u and is defined as follows: $|u| = n$.*

With S^* we denote the set of strings over S, and $u, v, w, \dots$ range over S. Given $n \geq 0$, with S^n we denote the set of strings of length n over S.

Given a string $u = x_1 \dots x_n$ and i such that $1 \geq i \geq n$, with $(u)_i$ we denote the i-th element of u, namely, $(u)_i = x_i$.

Given a string $u = x_1 \dots x_n$, the multiset corresponding to u is defined as follows: for all $s \in S$, $m_u(s) = |\{i \mid x_i = s \wedge 1 \leq i \leq n\}|$. With abuse of notation, we use u to denote also m_u.

Definition 4. *With $S \times T$ we denote the cartesian product of sets S and T, with $\times_n S$, $n \geq 1$, we denote the cartesian product of n copies of set S and with $\times_{i=1}^n S_i$ we denote the cartesian product of sets $S_1, \dots, S_n$, i.e., $S_1 \times \dots \times S_n$. The ith projection of $(x_1, \dots, x_n) \in \times_{i=1}^n S_i$ is defined as $\pi_i(x) = x_i$, and lifted to subsets $X \subseteq \times_{i=1}^n S_i$ as follows: $\pi_i(X) = \{\pi_i(x) \mid x \in X\}$.*

Given a binary relation R over a set S, with R^n we denote the composition of n instances or R, with R^+ we denote the transitive closure of R, and with R^* we denote the reflexive and transitive closure of R.

3 Catalytic P Systems

We recall the definition of catalytic P systems without priorities on rules provided in [4]. For a thorough description of the model, motivation and examples see, e.g., [1, 2, 11, 12, 13].

To this aim, we start with the definition of a *membrane structure*:

Definition 5. *Given the alphabet $V = \{[,]\}$, the set MS is the least set inductively defined by the following rules:*

- *$[\,] \in MS$,*
- *if $\mu_1, \mu_2, \dots, \mu_n \in MS$, $n \geq 1$, then $[\mu_1 \dots \mu_n] \in MS$.*

We define the following relation over MS: $x \sim y$ iff the two strings can be written in the form $x = [_1 \dots [_2 \dots]_2 \dots [_3 \dots]_3 \dots]_1$ and $y = [_1 \dots [_3 \dots]_3 \dots [_2 \dots]_2 \dots]_1$ (i.e., if two pairs of parenthesis that are neighbors can be swapped together with their contents).

The set $\overline{MS}$ of membrane structures *is defined as the set of equivalence classes w.r.t. the relation $\sim^*$.*

We call a *membrane* each matching pair of parenthesis appearing in the membrane structure. A membrane structure μ can be represented as a Venn diagram, in which any closed space (delimited by a membrane and by the membranes immediately inside) is called a *region* of μ.

Definition 6. *A* catalytic P system *(of degree d, with $d \geq 1$) is a construct*

$$\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0),$$

where:

1. *V is a finite alphabet whose elements are called* objects*;*
2. *$C \subseteq V$ is a set of* catalysts*;*
3. *μ is a* membrane structure *consisting of d membranes (usually labeled with i and represented by corresponding brackets $[_i$ and $]_i$, with $1 \leq i \leq d$);*
4. *w_i^0, $1 \leq i \leq d$, are strings over V associated with the regions $1, 2, \ldots, d$ of μ; they represent multisets of objects present in the regions of μ (the multiplicity of a symbol in a region is given by the number of occurrences of this symbol in the string corresponding to that region);*
5. *R_i, $1 \leq i \leq d$, are finite sets of* evolution rules *over V associated with the regions $1, 2, \ldots, d$ of μ; these evolution rules are of the forms $a \rightarrow v$ or $ca \rightarrow cv$, where c is a catalyst, a is an object from $V \setminus C$, and v is a string from $((V \setminus C) \times \{here, out, in\})^*$;*
6. *i_0 is a number between 1 and d and it specifies the* output *membrane of Π.*

The membrane structure and the multisets represented by w_i, $1 \leq i \leq d$, in Π constitute the *initial state*[1] of the system. A transition between states is governed by an application of the evolution rules which is done in parallel; all objects, from all membranes, which *can be* the subject of local evolution rules *have to* evolve simultaneously.

The application of a rule[2] $u \rightarrow v$ in a region containing a multiset m results in subtracting from m the multiset identified by u, and then in adding the multiset defined by v. The objects can eventually be transported through membranes due to the targets *in* and *out* (we usually omit the target *here*). Note that the catalysts are neither created nor destroyed by the application of the rules: they simply bound the number of occurrences of (particular sets of) rules in a maximal parallelism step. Moreover, catalysts cannot move across the membranes.

The system continues parallel steps until there remain no applicable rules in any region of Π; then the system halts. We consider the number of objects from V contained in the output membrane i_0 when the system halts as the *result* of the underlying computation of Π.

We say that a P system Π *diverges* if there exists an infinite computation starting from the initial state of Π.

For example, a graphical representation of the initial state of the P system $\Pi_1 = (\{a, b, c\}, \{c\}, [_1[_2\ ]_2]_1, aabc, abc, \{a \rightarrow b, cb \rightarrow ca, ca \rightarrow caa\}, \{ca \rightarrow cb, cb \rightarrow cba\}, 1)$ is depicted in Figure 1.

[1] Here we use the term *state* instead of the classical term *configuration* because we will define a (essentially equivalent but syntactically) different notion of configuration in Section 5.

[2] We use $u \rightarrow v$ as a shorthand to denote both the rules of kind $a \rightarrow w$ and $ca \rightarrow cw$.

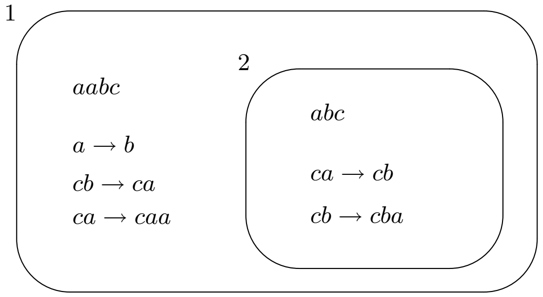

Fig. 1. The initial state of the P system Π_1.

Even if both a copy of a and a copy of b are present in the membrane 1 in the initial state, rules $cb \rightarrow ca$ and $ca \rightarrow caa$ cannot be applied together in the same step, as only a single instance of catalyst c is present in such a state.

We introduce a couple of functions on membrane structures that will be useful in the following:

Definition 7. *Let μ be a membrane structure consisting of d membranes, labelled with $\{1, \ldots, d\}$.*

Given two membranes i and j in μ, we say that the i is contained in j if the surface delimited by the perimeter of i in the Venn diagram representation of μ is contained inside the perimeter of j.

We say that i is the father *of j (and j is a* child *of i) if the membrane j is contained in i, and no membrane exists that contains j and is contained in i.*

The partial function $father : \{1, \ldots, d\} \rightarrow \{1, \ldots, d\}$ returns the father of a membrane i, or is undefined if i is the external membrane.

The function $children : \{1, \ldots, d\} \rightarrow \mathcal{P}(\{1, \ldots, d\})$ returns the set of children of a membrane.

For example, take $\mu = [_1[_2[_3\ ]_3]_2\ [_4\ ]_4]_1$; then, $father(2) = father(4) = 1$, $father(3) = 4$ and $father(1)$ is undefined; moreover, $children(4) = \emptyset$ and $children(1) = \{2, 4\}$.

4 Well-Structured Transition Systems

We start by recalling some basic definitions and results from [3], concerning well-structured transition systems, that will be used in the following.

A *quasi-ordering* (qo) is a reflexive and transitive relation.

Definition 8. *A* well-quasi-ordering *(wqo) is a quasi-ordering $\leq$ over a set X such that, for any infinite sequence $x_0, x_1, x_2, \ldots$ in X, there exist subscripts $i < j$ such that $x_i \leq x_j$.*

Note that, if $\leq$ is a wqo, then any infinite sequence $x_0, x_1, x_2, \ldots$ contains an infinite increasing subsequence $x_{i_0}, x_{i_1}, x_{i_2}, \ldots$ (with $i_0 < i_1 < i_2 < \ldots$).

Transition systems can be formally defined as follows.

Definition 9. *A* transition system *is a structure $TS = (S, \rightarrow)$, where S is a set of* states *and $\rightarrow \subseteq S \times S$ is a set of* transitions*.*
We write $Succ_{\rightarrow}(s)$ to denote the set $\{s' \in S \mid s \rightarrow s'\}$ of immediate successors of $s \in S$.
TS is finitely branching *if $\forall s \in S : Succ(s)$ is finite. We restrict to finitely branching transition systems.*

Well-structured transition systems, defined as follows, provide the key tool to decide properties of computations.

Definition 10. *A* well-structured transition system (with strong compatibility) *is a transition system $TS = (S, \rightarrow)$, equipped with a quasi-ordering $\leq$ on S, also written $TS = (S, \rightarrow, \leq)$, such that the following two conditions hold:*

1. **well-quasi-ordering***: $\leq$ is a well-quasi-ordering, and*
2. **strong compatibility***: $\leq$ is (upward) compatible with $\rightarrow$, i.e., for all $s_1 \leq t_1$ and all transitions $s_1 \rightarrow s_2$, there exists a state t_2 such that $t_1 \rightarrow t_2$ and $s_2 \leq t_2$.*

The following theorem (a special case of a result in [3]) will be used to obtain our decidability result.

Theorem 1. *Let $TS = (S, \rightarrow, \leq)$ be a finitely branching, well-structured transition system with decidable $\leq$ and computable Succ. The existence of an infinite computation starting from a state $s \in S$ is decidable.*

To show that the quasi-ordering relation we will define on P systems is a well-quasi-ordering we need the following results on well-quasi-ordering relations for finite sets and multisets.

Proposition 1. *Let S be a finite set. Then the equality is a wqo over S.*

Lemma 1. *[Dickson] Let S be a finite set. The relation $\subseteq$ is a wqo over $\mathcal{M}_{fin}(S)$.*

The following proposition permits to obtain a well-quasi-ordering on the cartesian product of sets equipped with well-quasi-orderings:

Proposition 2. *Let S_i be sets and $\leq_i$, be wqo over S_i, for $i = 1, \ldots, n$. The relation $\leq$ over $\times_{i=1}^{n} S_i$ is defined as follows: $x \leq y$ iff $\pi_i(x) \leq_i \pi_i(y)$ for $i = 1, \ldots, n$. The relation $\leq$ is a wqo over $\times_{i=1}^{n} S_i$.*

5 Decidability of Divergence for Catalytic P Systems

In this section we show that the existence of a divergent computation is decidable for the class of catalytic P systems defined in Section 3.

The decidability proof is based on the theory of well-structured transition systems [3]: the existence of an infinite computation starting from a given state is decidable for finitely branching transition systems, provided that the set of states can be equipped with a well-quasi-ordering, i.e., a quasi-ordering relation which is compatible with the transition relation and such that each infinite sequence of states admits an increasing subsequence.

To this aim, we need a finer definition of the computation of a P system, where a maximal parallelism evolution step is represented as a (maximal) sequence of simple evolution steps, which are obtained by the application of a single evolution rule.

After defining such preliminary notions, we propose a well-quasi-ordering that turns out to be compatible with the maximal parallelism evolution step.

5.1 Partial configurations, reaction relation, and maximal parallelism step

To represent the states of the system reached after the execution of a non-maximal sequence of simple evolution rules, we introduce the notion of *partial configuration* of a system. In a partial configuration, the contents of each region is represented by two multisets:

- The multiset of *active objects* contains the objects that were in the region at the beginning of the current maximal parallelism evolution step. These objects can be used by the next simple evolution step.
- The multiset of *frozen objects* contains the objects that have been produced in the region during the current maximal parallelism evolution step. These objects will be available for consumption in the next maximal parallelism evolution step.

Definition 11. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system.*

A partial configuration *of Π is a tuple $(w_1, \bar{w}_1), \ldots, (w_d, \bar{w}_d) \in \times_d(V \times V)$.*

We use $\times_{i=1}^{d}(w_i, \bar{w}_i)$ to denote the partial configuration above.

The set of partial configurations of Π is denoted by $Conf_\Pi$. We use $\gamma, \gamma', \gamma_1, \ldots$ to range over $Conf_\Pi$.

In the above definition, $w_1, \ldots, w_d$ represent the active multisets, whereas $\bar{w}_1, \ldots, \bar{w}_d$ represent the frozen multisets.

For example, $((aabc, \emptyset), (abc, \emptyset))$ and $((abc, cb), (c, \emptyset))$ are partial configurations (not necessarily reachable from the initial state) of the P system in Figure 1.

A *configuration* is a partial configuration containing no frozen objects; configurations represent the states reached after the execution of a maximal parallelism computation step.

Definition 12. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system.*

A configuration *of Π is a partial configuration $\times_{i=1}^{d}(w_i, \bar{w}_i)$ satisfying the following: $\bar{w}_i = \emptyset$ for $i = 1, \ldots, d$.*

The initial configuration *of Π is the configuration $\times_{i=1}^{d}(w_i^0, \emptyset)$.*

For example, $((aabc, \emptyset), (abc, \emptyset))$ is a configuration (actually the initial configuration) of the P system in Figure 1, whereas the partial configuration $((ab, caa), (abc, \emptyset))$ is not a configuration.

The size of a partial configuration is the number of active objects contained in the configuration; it will be used to prove the results in the following part of the paper:

Definition 13. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system and $\gamma = \times_{i=1}^{d}(w_i, \bar{w}_i)$ be a partial configuration. The* size *of γ is $\#(\gamma) = \sum_{i=1}^{d} |w_i|$.*

The execution of a simple evolution rule is formalized by the notion of reaction relation, defined as follows:

Definition 14. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system.*

The reaction relation $\mapsto$ *over $Conf_\Pi \times Conf_\Pi$ is defined as follows:*

$\times_{i=1}^{d}(w_i, \bar{w}_i) \mapsto \times_{i=1}^{d}(w'_i, \bar{w}'_i)$ iff there exist k, with $1 \leq k \leq d$, an evolution rule $u \to v \in R_k$ and a migration string *$\rho \in \{1, \ldots, d\}^{|v|}$ such that*

- *$u \subseteq w_k$,*
- *$w'_k = w_k \setminus u$,*
- *$\forall i : 1 \leq i \leq d$ and $i \neq k$ implies $w'_i = w_i$,*
- *$\forall j : 1 \leq j \leq |v|$ the following holds:*
  - *if $\pi_2((v)_j) = here$, then $(\rho)_j = k$,*
  - *if $\pi_2((v)_j) = out$, then*[3] *$(\rho)_j = father(k)$,*
  - *if $\pi_2((v)_j) = in$, then*[4] *$(\rho)_j \in children(k)$,*
- *$\forall i, 1 \leq i \leq d : \bar{w}'_k = \bar{w}_k \oplus \bigoplus_{1 \leq j \leq |v|, (\rho)_j = k} (v)_j$.*

For example, in the P system of Figure 1

$$\begin{array}{rcl} ((aabc, \emptyset), (abc, \emptyset)) & \mapsto & ((ab, caa), (abc, \emptyset)), \\ ((ab, caa), (abc, \emptyset)) & \mapsto & ((ab, caa), (a, cba)), \\ ((ab, caa), (a, cba)) & \not\mapsto & ((ab, cab), (a, cba)). \end{array}$$

Note that the size of a configuration represents an upper bound to the length of the sequences of reactions starting from that configuration. Hence, infinite sequences of reactions are not possible.

[3] As $\rho \in \{1, \ldots, d\}^{|v|}$, this implies that $father(k)$ is defined.

[4] This implies that $children(k)$ is not empty.

Proposition 3. *Let $\Pi = (V, C, \mu, w_1^0, \dots, w_d^0, R_1, \dots, R_d, i_0)$ be a catalytic P system and γ be a configuration. If $\gamma \mapsto^n \gamma'$, then $n \leq \#(\gamma)$.*

The size of a configuration is also used to provide an upper bound to the set of configurations reachable by firing a single reaction:

Proposition 4. *Let $\Pi = (V, C, \mu, w_1^0, \dots, w_d^0, R_1, \dots, R_d, i_0)$ be a catalytic P system and γ be a configuration and $Succ_{\mapsto}(\gamma) = \{\gamma' \mid \gamma \mapsto \gamma'\}$. Then $|Succ_{\mapsto}(\gamma)| \leq \#(\gamma) \cdot max\{|R_i| \mid i = 1, \dots, d\} \cdot max\{|v| \mid \exists i, u : 1 \leq i \leq d \wedge u \rightarrow v \in R_i\} \cdot d$.*

The *heating function heat* transforms the frozen objects of a configuration in active objects, and will be used in the definition of the maximal parallelism computation step.

Definition 15. *Let $\Pi = (V, C, \mu, w_1^0, \dots, w_d^0, R_1, \dots, R_d, i_0)$ be a catalytic P system and $\times_{i=1}^d (w_i, \bar{w}_i)$ be a partial configuration of Π.*

The heating function $heat : Conf_\Pi \rightarrow Conf_\Pi$ is defined as follows:
$heat(\times_{i=1}^d (w_i, \bar{w}_i)) = \times_{i=1}^d (w_i \oplus \bar{w}_i, \emptyset)$

For example, $heat(((ab, caa), (a, cba))) = ((abcaa, \emptyset), (acba, \emptyset))$.

Now we are ready to define the maximal parallelism computational step $\Mapsto$:

Definition 16. *Let $\Pi = (V, C, \mu, w_1^0, \dots, w_d^0, R_1, \dots, R_d, i_0)$ be a catalytic P system.*

The maximal parallelism computational step $\Mapsto$ *over (nonpartial) configurations of Π is defined as follows: $\gamma_1 \Mapsto \gamma_2$ iff there exists a partial configuration γ' such that $\gamma_1 \mapsto^+ \gamma'$, $\gamma' \not\mapsto$ and $\gamma_2 = heat(\gamma')$.*

For example, in the P system of Figure 1

$$\begin{array}{l} ((aabc, \emptyset), (abc, \emptyset)) \mapsto \\ ((ab, caa), (abc, \emptyset)) \mapsto \\ ((ab, caa), (b, cb)) \mapsto \\ ((b, caab), (b, cb)) \not\mapsto \end{array}$$

Hence, as $heat(((b, caab), (b, cb))) = ((bcaab, \emptyset), (bcb, \emptyset))$, we obtain

$$((aabc, \emptyset), (abc, \emptyset)) \Mapsto ((bcaab, \emptyset), (bcb, \emptyset))$$

Now we can provide a formal definition of the notion of divergence.

Definition 17. *Let $\Pi = (V, C, \mu, w_1^0, \dots, w_d^0, R_1, \dots, R_d, i_0)$ be a catalytic P system.*

We say that Π is divergent if there exists an infinite sequence of configurations γ_i, $i = 0, 1, \dots$ such that

- *γ_0 is the initial configuration of Π, i.e., $\gamma_0 = \times_{i=1}^d (w_i^0, \emptyset)$,*
- *$\gamma_i \Mapsto \gamma_{i+1}$ for all $i \geq 0$.*

A P system is deterministic if the configuration reached by the execution of a maximal parallelism step is univocally determined.[5]

Definition 18. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system and let γ_0 be the initial configuration of Π.*

We say that P is deterministic *if, for each $\gamma, \gamma', \gamma'' \in Reach_\Pi$: $\gamma \Mapsto \gamma'$ and $\gamma \Mapsto \gamma''$ imply $\gamma' = \gamma''$.*

The set of partial configurations reachable in a computation is defined as follows:

Definition 19. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system and let γ_0 be the initial configuration of Π. The set of partial configurations* reachable *from the initial configuration of Π is defined as follows:*
$Reach_\Pi = \{\gamma \in Conf_\Pi \mid \exists \gamma_1 : \gamma_0 \Mapsto^* \gamma_1 \mapsto^* \gamma\}$.

To make use of the tools illustrated in Section 4 is is necessary to show that the transition system is finitely branching:

Proposition 5. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system. The transition system $(Reach_\Pi, \Mapsto)$ is finitely branching.*

Proof. (Sketch) Let $\gamma \in Reach_\Pi$ and $Succ_{\Mapsto}(\gamma) = \{\gamma' \mid \gamma \Mapsto \gamma'\}$.

We show that $Succ_{\Mapsto}(\gamma)$ is finite.

By definition of $\Mapsto$, it is easy to see that $|Succ_{\Mapsto}(\gamma)| \leq |\{\gamma' \mid \gamma \mapsto^+ \gamma'\}|$.

We know by Proposition 3 that $\#(\gamma)$ is an upper bound to the length of any path starting from γ in the transition system $(Reach_\Pi, \Mapsto)$. Now we consider the tree obtained as unfolding of the part of transition system reachable from γ. Thanks to the above observation, this tree has a finite depth, that is not greater than $\#(\gamma)$.

Now we provide an upper bound to the set of arcs exiting from each node of the tree.

Let $Branch(\gamma) = \#(\gamma) \cdot max\{|R_i| \mid i = 1, \ldots, d\} \cdot max\{|v| \mid \exists i, u : 1 \leq i \leq d \wedge u \rightarrow v \in R_i\} \cdot d$.

By Proposition 4 we have that $|Succ_{\mapsto}(\gamma)| \leq Branch(\gamma)$.

By definition of $\mapsto$ it is easy to see that the number of active objects decreases after the firing of a reaction rule, i.e., if $\gamma \mapsto \gamma'$, then $\#(\gamma) \geq \#(\gamma')$. Hence, from the fact above we obtain that $|Succ_{\mapsto}(\gamma')| \leq Branch(\gamma)$ for all γ' such that $\gamma \mapsto \gamma'$.

Thus, the number of arcs exiting from each node of the tree is bounded by $Branch(\gamma)$.

As the depth of the tree is bounded by $\#(\gamma)$, the number of nodes of in the tree is not greater than $(Branch(\gamma))^{\#(\gamma)}$.

[5] Here we consider a slightly more general notion of determinism, namely, a system is deterministic even if the target configuration can be reached by firing different sets of of rules. This notion is sometimes called *confluence*. In [8, 9] a more restrictive notion of determinism is considered: namely, a system is deterministic if at each step there exists at most one maximally parallel multiset of rules that can be applied.

As $|Succ_{\Mapsto}(\gamma)| \leq |\{\gamma' \mid \gamma \mapsto^{+} \gamma'\}|$, and $|\{\gamma' \mid \gamma \mapsto^{+} \gamma'\}|$ is not greater than the number of nodes in the tree, we obtain that $|Succ_{\Mapsto}(\gamma)| \leq (Branch(\gamma))^{\#(\gamma)}$. Hence, $Succ_{\Mapsto}(\gamma)$ is finite. □

A crucial property for the proof of decidability of divergence is the fact that no catalyst is created or destroyed during the computation; hence, the number of each catalyst (either active or frozen) in each region of the system is left unchanged by the execution of both single reaction steps and maximal parallelism steps:

Proposition 6. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system and let $\gamma_0 = \times_{i=1}^{d}(w_i^0, \emptyset)$ be the initial configuration of Π.*

If $\gamma_0 \mapsto^* \times_{i=1}^{d}(w_i, \bar{w}_I)$, then $w_i^0|_C = (w_i \oplus \bar{w}_i)|_C$ for $i = 1, \ldots, d$.

If $\gamma_0 \Mapsto^* \times_{i=1}^{d}(w_i, \bar{w}_I)$, then $w_i^0|_C = (w_i \oplus \bar{w}_i)|_C$ for $i = 1, \ldots, d$.

As a consequence, the above property also holds for all reachable partial configurations:

Corollary 1. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system, let $\gamma_0 = \times_{i=1}^{d}(w_i^0, \emptyset)$ be the initial configuration of Π and $\gamma = \times_{i=1}^{d}(w_i, \bar{w}_i) \in Reach(\Pi)$.*

Then $w_i^0|_C = (w_i \oplus \bar{w}_i)|_C$ for $i = 1, \ldots, d$.

5.2 Well-quasi-ordering on partial configurations compatible with $\Mapsto$

Now we are ready to define a preorder relation over partial configurations of Π that turns out to be a well-quasi-ordering compatible with $\Mapsto$.

The relation $\gamma_1 \leq \gamma_2$ essentially requires that the multisets of active (resp. frozen) catalysts in each region of the two partial configurations is the same, whereas the multiset of active (resp. frozen) objects that are not catalysts of γ_1 is contained in the corresponding multiset of γ_2. We start defining a relation $\preceq$ over the pairs of active and frozen multisets of a single region, then we extend the notion to a partial configuration.

Definition 20. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system.*

The relation $\preceq$ over $\mathcal{M}_{fin}(V) \times \mathcal{M}_{fin}(V)$ is defined as follows: $(v, \bar{v}) \preceq (w, \bar{w})$ iff

- $v|_C = w|_C$,
- $v|_{V \setminus C} \subseteq w|_{V \setminus C}$,
- $\bar{v}|_C = \bar{w}|_C$,
- $\bar{v}|_{V \setminus C} \subseteq \bar{w}|_{V \setminus C}$.

The relation $\leq$ over $Conf_\Pi$ is defined as follows: $\times_{i=1}^{d}(v_i, \bar{v}_i) \leq \times_{i=1}^{d}(w_i, \bar{w}_i)$ iff, for $i = 1, \ldots, d$, $(v_i, \bar{v}_i) \preceq (w_i, \bar{w}_i)$.

It is easy to see that the relation $\leq$ is a quasi-ordering over the partial configurations of Π.

Now we show that $\leq$ is a well-quasi-ordering over the partial configurations of Π.

Lemma 2. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system. The relation* $\leq$ *is a well-quasi-ordering over* $Reach_\Pi$.

Proof. Let $\gamma \in Reach_\Pi$. By Corollary 1 we obtain an upper limit to the multiset of catalysts in γ, i.e., if $\gamma = \times_{i=1}^{d}(w_i, \bar{w}_i)$, then $w_i|_C \subseteq w_i^0|_C$ and $\bar{w}_i|_C \subseteq w_i^0|_C$ for $i = 1, \ldots, d$.

Now we show that $\preceq$ is a wqo over $\pi_i(Reach_\Pi)$, for $i = 1, \ldots, d$.

Let $(w_1, \bar{w}_1), \ldots, (w_k, \bar{w}_k), \ldots$ be an infinite sequence of elements of $\pi_i(Reach_\Pi)$. Consider now the sequence $w_1|_C, \ldots, w_k|_C, \ldots$ of the multisets of active catalysts in the sequence above. As there exists an upper bound to such multisets, i.e., $w_k|_C \subseteq w_i^0|_C$ for $k \geq 0$, the set $\{w_k|_C \mid k \geq 0\}$ is finite; hence, by Proposition 1 it is possible to extract an infinite increasing subsequence w.r.t. the ordering relation $=$, or, in other words, it is possible to extract an infinite subsequence $w_{i_1}|_C, \ldots, w_{i_k}|_C, \ldots$ of equal elements, i.e., such that $w_{i_k}|_C = w_{i_h}|_C$ for all $h, k \geq 0$.

Consider now the sequence $w_{i_1}|_{(V \setminus C)}, \ldots, w_{i_k}|_{(V \setminus C)}, \ldots$ of active noncatalyst objects in the extracted subsequence. As $V \setminus C$ is a finite set, by Dickson Lemma 1 it is possible to extract an infinite subsequence that is increasing w.r.t. the multiset inclusion relation $\subseteq$.

Following the same reasoning it is possible to extract from the subsequence obtained in the previous step a subsequence satisfying the following: the multisets of frozen catalysts in each element of the subsequence are all equal and the multisets of frozen noncatalysts in each element of the subsequence are increasing w.r.t. $\subseteq$.

Thus, we have built an infinite subsequence of $(w_1, \bar{w}_1), \ldots, (w_k, \bar{w}_k), \ldots$ that is increasing w.r.t. the ordering relation $\preceq$ of Definition 20. Hence, the relation $\preceq$ is a wqo over $\pi_i(Reach_\Pi)$.

By definition of $\leq$ and by Lemma 2, we obtain that $\leq$ is a wqo over $Reach_\Pi$. □

Here we show that the relation $\leq$ is strongly compatible with the maximal parallelism relation $\Mapsto$. To this aim, we need an auxiliary ordering relation $\ll$ on the set of partial configurations.

Definition 21. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system.*

The relation $\prec\!\!\prec$ *over* $\mathcal{M}_{fin}(V) \times \mathcal{M}_{fin}(V)$ *is defined as follows:* $(v, \bar{v}) \prec\!\!\prec (w, \bar{w})$ *iff*

- $v|_C \supseteq w|_C$,
- $v|_{V \setminus C} \subseteq w|_{V \setminus C}$,
- $(v \oplus \bar{v})|_C = (w \oplus \bar{w})|_C$,
- $\bar{v}|_{V \setminus C} \subseteq \bar{w}|_{V \setminus C}$.

The relation $\ll$ over $Conf_{\Pi}$ is defined as follows: $\times_{i=1}^{d}(v_i, \bar{v}_i) \ll \times_{i=1}^{d}(w_i, \bar{w}_i)$ *iff, for* $i = 1, \ldots, d$, $(v_i, \bar{v}_i) \lessdot\!\!\lessdot (w_i, \bar{w}_i)$.

It is easy to see that $\leq$ is stronger than $\ll$:

Proposition 7. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system and* $\gamma_1, \gamma_2 \in Conf_{\Pi}$. *If* $\gamma_1 \leq \gamma_2$, *then* $\gamma_1 \ll \gamma_2$.

Moreover, if $\gamma_1 \ll \gamma_2$ and no reaction is fireable in γ_1, then γ_1 is smaller than any partial configuration that can be reached from γ_2 by executing a reaction rule:

Proposition 8. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system and* $\gamma_1, \gamma_2 \in Conf_{\Pi}$. *If* $\gamma_1 \ll \gamma_2$, $\gamma_1 \not\mapsto$ *and* $\gamma_2 \mapsto \gamma_2'$ *then* $\gamma_1 \ll \gamma_2'$.

Proof. (Sketch) If $\gamma_2 \mapsto \gamma_2'$ two cases can happen:

- An instance of rule $a \to v$ has been applied in region i of γ_2; if $a \to v$ cannot be applied on γ_1, then object a does not belong to the active objects of region i of γ_1, hence $\gamma_1 \ll \gamma_2'$.
- An instance of rule $ca \to cv$ has been applied in region i of γ_2; by definition of $\ll$, the set of active catalysts in region i of γ_1 is greater or equal to the set of active catalysts in region i of γ_2; hence c belongs to the active catalysts in region i of γ_1; as rule $ca \to cv$ is not fireable in γ_1, this means that a does not belong to the active objects of region i of γ_1. Hence $\gamma_1 \ll \gamma_2'$. Note that the fact that an active catalyst is removed from region i when moving from γ_2 to γ_2' does not prevent the relation $\gamma_1 \ll \gamma_2'$ to hold. □

Another result that will be useful in the proof of strong compatibility of $\leq$ is the following:

Proposition 9. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system and* $\gamma_1, \gamma_2 \in Conf_{\Pi}$. *If* $\gamma_1 \ll \gamma_2$, *then* $heat(\gamma_1) \leq heat(\gamma_2)$.

Lemma 3. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system. The relation* $\leq$ *is a strongly compatible with* $\mapsto$ *over* $Conf_{\Pi}$.

Proof. (Sketch) Let γ_1, γ_1' and $\gamma_2 \in Conf_{\Pi}$ such that $\gamma_1 \leq \gamma_2$. Suppose that $\gamma_1 \mapsto \gamma_1'$. Then there exists a migration string ρ satisfying the conditions of Definition 14. The key idea consists in using the migration string ρ to show that there exists $\gamma_2' \in Conf_{\Pi}$ such that $\gamma_2 \mapsto \gamma_2'$. □

Theorem 2. *Let* $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ *be a catalytic P system. The relation* $\leq$ *is a strongly compatible with* $\Mapsto$ *over* $Conf_{\Pi}$.

Proof. Let γ, γ' and $\sigma \in Conf_{\Pi}$ such that $\gamma \leq \sigma$. Suppose that $\gamma \Mapsto \gamma'$. We show that there exists $\sigma' \in Conf_{\Pi}$ such that $\sigma \Mapsto \sigma'$ and $\gamma' \leq \sigma'$.

As $\gamma \Mapsto \gamma'$, there exists a sequence $\gamma_1, \ldots, \gamma_n$ such that $\gamma \mapsto \gamma_1 \mapsto \ldots \mapsto \gamma_n$, $\gamma_n \not\mapsto$ and $\gamma' = heat(\gamma_n)$.

By repeated application of Lemma 3, there exists $\sigma_1, \ldots, \sigma_n$ such that $\sigma \mapsto \sigma_1 \mapsto \ldots \mapsto \sigma_n$ and $\gamma_i \leq \sigma_i$ for $i = 1, \ldots, n$.

Now, as $\sigma_n \geq \gamma_n$, it may happen that $\sigma_n \mapsto$. By Proposition 3 there exists a (possibly empty) reactions sequence starting from σ_n of the following form: $\sigma_n \mapsto \sigma_{n+1} \mapsto \ldots \sigma_{n+k}$.

We obtained above that $\gamma_n \leq \sigma_n$; by Proposition 7 we obtain $\gamma_n \ll \sigma_n$; as $\gamma_n \not\mapsto$, by repeated application of Lemma 8 we obtain $\gamma_n \ll \sigma_{n+k}$.

From $\sigma \mapsto \ldots \mapsto \sigma_n \mapsto \ldots \mapsto \sigma_{n+k}$ and $\sigma_{n+k} \not\mapsto$ we get $\sigma \Mapsto heat(\sigma_{n+k})$.

From $\gamma_n \ll \sigma_{n+k}$ and Proposition 9 we obtain $heat(\gamma_n) \leq heat(\sigma_{n+k})$.

Hence, by taking $\sigma' = heat(\sigma_{n+k})$ we have shown that there exists σ' such that $\sigma \Mapsto \sigma'$ and $\gamma' \leq \sigma'$). □

5.3 Decidability of divergence

Now we are ready to state the main result of the paper.

Theorem 3. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system. The transition system $(Reach_\Pi, \Mapsto, \leq)$ is a well-structured transition system with decidable $\leq$ and computable $Succ_\Mapsto$.*

Proof. Strong compatibility of $\leq$ with the maximal parallelism computational step $\Mapsto$ has been proved in Theorem 2. By Lemma 2 we have that $\leq$ is a wqo over $Reach_\Pi$. From Definition 20 it is easy to deduce an effective procedure to check $\leq$. From Definitions 14 and 16 – and by proof of Proposition 3 – it is easy to deduce an effective procedure to compute $Succ_\Mapsto$. Proposition 5 shows that the transition system $(Reach_\Pi, \Mapsto, \leq)$ is finitely branching. □

By the above theorem and by Theorem 1 we obtain the following

Corollary 2. *Let $\Pi = (V, C, \mu, w_1^0, \ldots, w_d^0, R_1, \ldots, R_d, i_0)$ be a catalytic P system. The existence of an infinite computation starting from the initial configuration of Π is decidable.*

As a consequence of the above corollary we obtain an alternative proof of the result in [8, 9], showing that deterministic catalytic P systems are not universal. In deterministic P systems there exists a unique (possibly infinite) computation starting from the initial configuration; hence, checking for the existence of an halting computation is equivalent to checking that all computations halt, or, equivalently, that there exists a divergent computation. By Corollary 2 we obtain the decidability of the existence of an halting computation for deterministic catalytic P systems. This means that there exists no encoding of any Turing powerful formalism in deterministic catalytic P systems that preserves the existence of an halting computation.

6 Conclusion

In this paper we showed that the existence of a divergent computation is a decidable property for catalytic P systems.

As a byproduct, we also obtain an alternative proof of the nonuniversality of deterministic catalytic P systems, an open problem recently solved by Ibarra and Yen [8, 9]. Actually, the decidability of divergence permits to derive a slightly more general result, i.e., the nonuniversality of the class of P systems which is *uniform w.r.t. termination* (we say that a P system is uniform w.r.t. termination iff the following property holds: the P systems has a terminating computation iff all of its computations terminate).

Finally, as in [4], we consider P systems without priorities associated to the rules. A detailed investigation of the expressivity of various classes of deterministic P systems with priorities has been carried out in [8, 9]. An investigation of the impact of the introduction of priorities on our result is left for future investigation.

In the present paper we showed that divergence is a decidable property for catalytic P systems. The technique employed to prove the decidability of divergence is based on the theory of well-structured transition systems: besides universal termination, such a theory permits to analyse other interesting properties, such as, e.g., coverability, boundedness, and eventuality properties [3]. We plan to investigate the possibility to use this theory for the analysis of other (biologically relevant) properties.

Acknowledgement: I'm grateful to George Păun, Oscar Ibarra and, last but not least, Claudio Zandron for their precious comments and suggestions.

References

1. C.S. Calude, G. Păun: *Computing with Cells and Atoms.* Taylor & Francis, London, 2001.
2. J. Dassow, G. Păun: On the power of membrane computing. J. Univ. Comput. Sci., 5, 2 (1999).
3. A. Finkel, Ph. Schnoebelen: Well-structured transition systems everywhere! *Theoretical Computer Science*, 256 (2001), 63–92.
4. R. Freund, L. Kari, M. Oswald, P. Sosik: Computationally universal P systems without priorities: two catalysts are sufficient. *Theoretical Computer Science*, 330 (2005).
5. R. Freund, M. Oswald, P. Sosik: Reducing the number of catalysts needed in computationally universal systems without priorities. In *Proc. Fifth International Workshop on Descriptional Complexity of Formal Systems*, Budapest, 2003.
6. R. Freund, Gh. Păun: On deterministic P systems. Manuscript, 2003. (See the Milano webpage [17]).
7. O.H. Ibarra: Some recent results concerning deterministic P systems. In *Proc 6th International Workshop on Membrane Computing* (WMC6), LNCS 3850, Springer, 2006.

8. O.H. Ibarra, H.-C. Yen: On deterministic catalytic P systems. In *Proc. 10th International Conference on Implementation and Application of Automata* (CIAA2205), LNCS 3845, Springer, 2005.
9. O.H. Ibarra, H.-C. Yen: Deterministic catalytic systems are not universal. Submitted, 2006.
10. M.L. Minsky: *Computation: Finite and Infinite Machines.* Prentice-Hall, Englewood Cliffs, 1967.
11. Gh. Păun: Computing with membranes: an introduction. *Bull. EATCS*, 67 (1999).
12. Gh. Păun: Computing with membranes. *Journal of Computer and System Sciences*, 61, 1 (2000), 108–143.
13. Gh. Păun: *Membrane Computing. An Introduction.* Springer, Berlin, 2002.
14. Gh. Păun: Further twenty–six open problems in membrane computing. In *Proc. Third Brainstorming Week on Membrane Computing*, Sevilla, 2005.
15. P. Sosik: The power of catalysts and priorities in membrane systems. *Grammars*, 6, 1 (2003).
16. P. Sosik, R. Freund: P Systems without priorities are computationally universal. In *Proc. Membrane Computing. International Workshop, WMC-CdeA 2002*, LNCS 2597, Springer, 2003.
17. P Systems webpage: `http://psystems.disco.unimib.it`.