A polynomial alternative to unbounded environment for tissue P systems with cell division

Mario J. Pérez-Jiménez[a], Agustín Riscos-Núñez[a], Miquel Rius-Font[b], and Francisco J. Romero-Campero[a]

[a]*Department of Computer Science and Artificial Intelligence, University of Sevilla, Avda. Reina Mercedes s/n, 41012, Sevilla, Spain;* [b]*Department of Applied Mathematics IV, Universitat Politécnica de Catalunya, Barcelona, Spain*

The standard definition of tissue P systems includes a special alphabet whose elements are assumed to appear in the initial configuration of the system in an arbitrarily large number of copies. These objects reside in a distinguished place of the system, called the environment. Such potentially infinite supply of objects seems an unfair tool when designing efficient solutions to computationally hard problems in the framework of membrane computing, by performing a space–time trade-off. This paper deals with computational aspects of tissue P systems with cell division where there is no environment having the property mentioned above. Specifically, we prove that the polynomial complexity classes associated with tissue P systems with cell division and with or without environment are actually identical. As a consequence, we conclude that it is not necessary to have infinitely many copies of some objects in the initial configuration in order to solve **NP**–complete problems in an efficient way.

Keywords: membrane computing; tissue P systems; cell division; environment of a tissue; computational complexity

1. Introduction

Membrane computing is a branch of Natural Computing initiated by Gh. Păun at the end of 1998 [8]. It is inspired by the structure and functioning of the living cells, in the following sense: the foundational paper introduces a special kind of distributed parallel devices, called *transition P systems*, based on the notion of a membrane structure containing objects (chemicals) inside. Formally, such a structure consists of a rooted tree whose nodes are an abstraction of biological membranes (the hierarchy induced by the parent–child relation of nodes in the tree captures in a natural way the situation of a membrane being nested into another one). The root of the tree corresponds to the skin membrane that delimits the inner part of the cell separating it from the outside region. The leaves of the tree are called elementary membranes. In transition P systems, there is a distinguished region called environment which plays a passive role. Initially, the environment is empty and along any computation, some objects can be released into the environment moving out of the skin membrane of the system. However, in this model, it is not

allowed to send an object from the environment into any membrane, since the environment has no associated rules.

A computational complexity theory in membrane computing has been developed and polynomial complexity classes have been introduced trying to capture the tractability concept for decision problems. An interesting variant of P systems, called P systems with active membranes, inspired by membrane division processes (*mitosis*) [9], is able to produce an exponential workspace in terms of the number of membranes and objects in linear time. In that framework, efficient solutions to computationally hard problems have been provided.

In this paper, we deal with another type of devices in membrane computing, called tissue P systems, inspired by intercellular communication in tissues and cooperation between neurons. Instead of considering a hierarchical arrangement of membranes structured by a rooted tree, membranes (called cells in this new framework) are placed in the nodes of a directed graph. From the seminal definitions of tissue P systems [5,6], several research lines have been developed and other variants have arisen. For example, it is worth recalling here the solution of SAT presented in [1] using cell-like symport/antiport P systems with membrane division, indicating the possibility to adapt the solution for the case of having an empty environment.

One of the most interesting variants of tissue P systems was presented in [11]. In that paper, the definition of tissue P systems is combined with the one of P systems with active membranes, yielding *tissue P systems with cell division*. In this kind of tissue P systems, there exists replication, that is, the two new cells generated by a division rule have exactly the same objects except for at most a pair of different objects. Another relevant ingredient of these models consists of a special alphabet of objects initially contained in a distinguished region of the system: the environment. In tissue P systems, the environment plays an active role in the sense that not only can it receive objects from different cells of the system but some cells can also receive objects from the environment along a computation of the system. Besides, the objects initially placed in the environment are assumed to appear in an arbitrarily large amount of copies. In this paper, the mentioned property above is analysed from the computational complexity point of view. That is, we study the efficiency of tissue P systems with cell division such that their alphabet of the environment is empty.

The paper is organized as follows. In the next section, some preliminary concepts and definitions are given. Section 3 is devoted to tissue P systems with cell division, and tissue P systems without environment are introduced. Recognizer tissue P systems with cell division and polynomial complexity classes associated with this kind of P systems are also defined in this section. A technical result is given concerning recognizer tissue P systems where only communication rules are allowed. In Section 4, the concept of efficient simulation of a recognizer tissue P system is given, and a tissue P system with cell division and without environment simulating in an efficient way a tissue P system with cell division, is provided. Section 5 is devoted to the main result of the paper: the polynomial complexity classes of recognizer tissue P systems with cell division with or without environment are the same. Finally, conclusions and ideas for further work are given in Section 6.

2. Preliminaries

An *alphabet*, Γ, is a non-empty set whose elements are called *symbols*. An ordered finite sequence of symbols is a *string* or *word*. If u and v are strings over Γ, then so is their *concatenation* uv, obtained by juxtaposition, that is, writing u and v one after the other. The number of symbols in a string u is the *length* of the string and it is denoted by $|u|$. As usual, the empty string (with length 0) will be denoted by λ. The set of all strings over an alphabet Γ is denoted by Γ^*. In algebraic terms, Γ^* is the free monoid generated by Γ under the operation of concatenation. Subsets, finite or infinite, of Γ^* are referred to as *languages* over Γ.

The set of symbols occurring in a string $u \in \Gamma^*$ is denoted by $\mathrm{alph}(u)$.

The *Parikh vector* associated with a string $u \in \Gamma^*$ with respect to the alphabet $\Sigma = \{a_1, \ldots, a_r\} \subseteq \Gamma$ is $\Psi_\Sigma(u) = (|u|_{a_1}, \ldots, |u|_{a_r})$, where $|u|_{a_i}$ denotes the number of occurrences of symbol a_i in string u. This is called the *Parikh mapping* associated with Σ. Notice that, in this definition, the ordering of the symbols from Σ is relevant. If $\Sigma_1 = \{a_{i_1}, \ldots, a_{i_r}\} \subseteq \Gamma$, then we define $\Psi_{\Sigma_1}(u) = (|u|_{a_{i_1}}, \ldots, |u|_{a_{i_r}})$, for each $u \in \Gamma^*$.

A *multiset* m over a set A is a pair (A, f), where $f : A \to \mathbb{N}$ is a mapping. If $m = (A, f)$ is a multiset, then its *support* is defined as $\mathrm{supp}(m) = \{x \in A : f(x) > 0\}$. A multiset is empty (resp. finite) if its support is the empty set (resp. a finite set). If $m = (A, f)$ is a finite multiset over A and $\mathrm{supp}(m) = \{a_1, \ldots, a_k\}$, then it will be denoted as $m = \{a_1^{f(a_1)}, \ldots, a_k^{f(a_k)}\}$. That is, superscripts indicate the multiplicity of each element, and if $f(x) = 0$ for $x \in A$, then element x is omitted. A finite multiset $m = \{a_1^{f(a_1)}, \ldots, a_k^{f(a_k)}\}$ can also be represented by the string $a_1^{f(a_1)} \cdots a_k^{f(a_k)}$ over the alphabet $\{a_1, \ldots, a_k\}$. Nevertheless, all permutations of this string identify the same multiset m precisely. Throughout this paper, we speak about 'the finite multiset m', where m is a string, meaning 'the finite multiset represented by the string m'. If $m_1 = (A, f_1), m_2 = (A, f_2)$ are multisets over A, then we define the union of m_1 and m_2 as $m_1 + m_2 = (A, g)$, where $g = f_1 + f_2$, that is, $g(a) = f_1(a) + f_2(a)$, for each $a \in A$.

For any sets A and B, the *relative complement* $A \setminus B$ of B in A is defined as follows: $A \setminus B = \{x \in A : x \notin B\}$.

Finally, for any set A, we denote $|A|$ the cardinal (number of elements) of A, as usual.

In what follows, we give a brief overview on the basic notions and terminology of tissue P systems. For further details, see for example, [10] or [12].

3. Tissue P systems with cell division

DEFINITION 3.1 *A tissue P system with cell division of degree $q \geq 1$ is a tuple $\Pi = (\Gamma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\mathrm{out}})$, where*:

- Γ *is a finite alphabet.*
- $\mathcal{E} \subseteq \Gamma$.
- $\mathcal{M}_1, \ldots, \mathcal{M}_q$ *are strings over* Γ.
- $\mathcal{R}$ *is a finite set of rules of the following forms:*
  (i) *Communication rules:* $(i, u/v, j)$, *for* $i, j \in \{0, 1, \ldots, q\}, i \neq j, u, v \in \Gamma^*, |u| + |v| > 0$;
  (ii) *Division rules:* $[a]_i \to [b]_i[c]_i$, *where* $i \in \{1, \ldots, q\}$, $i \neq i_{\mathrm{out}}$ *and* $a, b, c \in \Gamma$.
- $i_{\mathrm{out}} \in \{0, 1, \ldots, q\}$.

A *tissue P system with cell division* $\Pi = (\Gamma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\mathrm{out}})$, of degree q, can be viewed as a set of q cells, labelled by $1, \ldots, q$ such that: (a) $\mathcal{M}_1, \ldots, \mathcal{M}_q$ represent the finite multisets of objects initially placed into the q cells of the system; (b) $\mathcal{E}$ is the set of objects initially located in the environment of the system, all of them available in an arbitrary number of copies; and (c) i_{out} represents a distinguished *region* which will encode the output of the system. We use the term *region* i ($0 \leq i \leq q$) to refer to cell i in the case $1 \leq i \leq q$ and to refer to the environment in the case $i = 0$.

A communication rule $(i, u/v, j)$ is called a *symport rule* if $u = \lambda$ or $v = \lambda$. A symport rule $(i, u/\lambda, j)$, with $i \neq 0, j \neq 0$, provides a virtual arc from cell i to cell j. A communication rule $(i, u/v, j)$ is called an *antiport rule* if $u \neq \lambda$ and $v \neq \lambda$. An antiport rule $(i, u/v, j)$, with $i \neq 0, j \neq 0$, provides two arcs: one from cell i to cell j and the other from cell j to cell i. Thus, every tissue P system has an underlying directed graph whose nodes are the cells of the system and the arcs are obtained from communication rules. In this context, the environment can be considered as

a virtual node of the graph such that its connections are defined by communication rules of the form $(i, u/v, j)$, with $i = 0$ or $j = 0$.

When applying a rule $(i, u/v, j)$, the objects of the multiset represented by u are sent from region i to region j and, simultaneously, the objects of multiset v are sent from region j to region i. The length of communication rule $(i, u/v, j)$ is defined as $|u| + |v|$.

When applying a division rule $[a]_i \to [b]_i[c]_i$, under the influence of object a, the cell with label i is divided into two cells with the same label; in the first copy, object a is replaced by object b, in the second one, object a is replaced by object c; all the other objects residing in cell i are replicated and copies of them are placed in the two new cells. The output region i_{out} cannot be divided.

The rules of a tissue P system with cell division are applied in a non-deterministic maximally parallel manner. At each step, rules to be applied are selected non-deterministically (some of them may be applied more than once), in such a way that no further applicable rule can be added to the selection, with the following important remark: if a cell divides, then the division rule is the only one which is applied for that cell at that step; the objects inside that cell cannot participate in any communication rule. In other words, before division a cell interrupts all its communication channels with the other cells and with the environment. The new cells resulting from division will interact with other cells or with the environment only at the next step – providing that they do not divide once again. The label of a cell precisely identifies the rules which can be applied to it.

An *instantaneous description* or a *configuration* at any instant of a tissue P system is described by all multisets of objects over Γ associated with all the cells present in the system, and the multiset of objects over $\Gamma \setminus \mathcal{E}$ associated with the environment at that moment. Recall that there are infinitely many copies of objects from $\mathcal{E}$ in the environment, and hence this set is not actually changed along the computation. The *initial configuration* is $(\mathcal{M}_1, \ldots, \mathcal{M}_q; \emptyset)$. A configuration is a *halting configuration* if no rule of the system is applicable to it.

Consider a tissue P system Π. We say that configuration $\mathcal{C}_1$ yields configuration $\mathcal{C}_2$ in one *transition step*, denoted by $\mathcal{C}_1 \Rightarrow_\Pi \mathcal{C}_2$, if we can pass from $\mathcal{C}_1$ to $\mathcal{C}_2$ by applying a maximal multiset of rules from $\mathcal{R}$, following the previous remarks. A *computation* of Π is a (finite or infinite) sequence of configurations such that:

(1) the first term of the sequence is the initial configuration of the system;
(2) each non-initial configuration of the sequence is obtained from the previous configuration by applying rules of the system in a maximally parallel manner as explained above; and
(3) if the sequence is finite (called *halting computation*), then the last term of the sequence is a halting configuration.

All computations start from an initial configuration and proceed as stated above; only halting computations give a result, which is encoded by the objects present in the output cell i_{out} in the halting configuration.

If $\mathcal{C} = \{\mathcal{C}_t\}_{t<r+1}$ $(r \in \mathbb{N})$ is a halting computation of Π, starting from the initial configuration $\mathcal{C}_0$ and having $\mathcal{C}_r$ as a halting configuration, then the *length of* $\mathcal{C}$, denoted by $|\mathcal{C}|$, is r. That is, $|\mathcal{C}|$ measures the number of steps of the computation, or equivalently the number of non-initial configurations $(\mathcal{C}_1, \ldots, \mathcal{C}_r)$.

We denote by $\mathcal{C}_t(i)$ the multiset obtained as the union of the multisets of objects over Γ contained in all cells labelled by i (by applying division rules, different cells with the same label can be created) at configuration $\mathcal{C}_t$, and $\mathcal{C}_t(0)$ denotes the multiset of objects over $\Gamma \setminus \mathcal{E}$ contained in the environment at configuration $\mathcal{C}_t$. Finally, we denote by $\mathcal{C}_t^*$ the multiset $\mathcal{C}_t(0) + \mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q)$.

DEFINITION 3.2 *A tissue P system with cell division and without environment is a tissue P system with cell division such that the alphabet of the environment is an empty set.*

Usually, we omit the alphabet of the environment in the tuple describing a tissue P system with cell division and without environment.

3.1 *Recognizer tissue P systems*

Let us recall that a *decision problem* is a pair (I_X, θ_X), where I_X is a language over a finite alphabet (whose elements are called *instances*) and θ_X is a total Boolean function over I_X. Many abstract problems are not decision problems. For example, in *combinatorial optimization problems*, some value must be optimized (minimized or maximized). In order to deal with such problems, they can be transformed into roughly equivalent decision problems by supplying a target/threshold value for the quantity to be optimized, and then asking whether this value can be attained.

A natural correspondence between decision problems and languages can be established as follows. Given a decision problem $X = (I_X, \theta_X)$, its associated language is $L_X = \{w \in I_X : \theta_X(w) = 1\}$. Conversely, given a language L, over an alphabet Γ, its associated decision problem is $X_L = (I_{X_L}, \theta_{X_L})$, where $I_{X_L} = \Gamma^*$, and $\theta_{X_L} = \{(x, 1) : x \in L\} \cup \{(x, 0) : x \notin L\}$. The solvability of decision problems is defined through the recognition of the languages associated with them.

In order to study the computing efficiency, the notions from classical *computational complexity theory* are adapted for membrane computing, and a special class of cell-like P systems is introduced in [14]: *recognizer P systems* (called *accepting P systems* in a previous paper [13]). For tissue P systems, with the same idea as recognizer cell-like P systems, *recognizer tissue P systems* are introduced in [11].

DEFINITION 3.3 *A recognizer tissue P system with cell division of degree $q \geq 1$ is a tuple $\Pi = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\text{in}}, i_{\text{out}})$, where*:

- $(\Gamma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\text{out}})$ *is a tissue P system with cell division of degree $q \geq 1$, as defined in the previous section.*
- *The working alphabet Γ has two distinguished objects* `yes` *and* `no`*, at least one copy of them is present in some initial multisets $\mathcal{M}_1$,..., $\mathcal{M}_q$, but none of them is present in $\mathcal{E}$.*
- *Σ is an (input) alphabet strictly contained in Γ such that $\mathcal{E} \cap \Sigma = \emptyset$.*
- *$\mathcal{M}_1, \ldots, \mathcal{M}_q$ are strings over $\Gamma \setminus \Sigma$.*
- *$i_{\text{in}} \in \{1, \ldots, q\}$ is the input cell.*
- *The output region i_{out} is the environment.*
- *All computations halt.*
- *If $\mathcal{C}$ is a computation of Π, then either object* `yes` *or object* `no` *(but not both) must have been released into the environment, and only at the last step of the computation.*

DEFINITION 3.4 *A recognizer tissue P system with cell division and without environment of degree $q \geq 1$ is a tuple $\Pi = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\text{in}}, i_{\text{out}})$, where:*

- $(\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\text{in}}, i_{\text{out}})$ *is a recognizer tissue P system with cell division, according to Definition* 3.3.
- *$\mathcal{E} = \emptyset$.*
- *$i_{\text{out}} \in \{1, \ldots, q\}$, that is, the output region i_{out} is a distinguished cell.*

For each multiset $m \in \Sigma^*$, the *computation of the system Π with input $m \in \Sigma^*$* starts from the configuration of the form $(\mathcal{M}_1, \ldots, \mathcal{M}_{i_{\text{in}}} + m, \ldots, \mathcal{M}_q; \emptyset)$, that is, the input multiset m has been added to the contents of the input cell i_{in}, and we denote it by $\Pi + m$. Therefore, we have an initial configuration associated with each input multiset m (over the input alphabet Σ) in this kind of systems.

Given a recognizer tissue P system (with or without environment) and a halting computation $\mathcal{C} = \{\mathcal{C}_t\}_{t<r+1}$ of Π ($r \in \mathbb{N}$), we define the result of $\mathcal{C}$ as follows:

$$\text{Output}(\mathcal{C}) = \begin{cases} \texttt{yes} & \text{if } \Psi_{\{\texttt{yes},\texttt{no}\}}(M_{r,i_{\text{out}}}) = (1,0) \text{ and} \\ & \quad \Psi_{\{\texttt{yes},\texttt{no}\}}(M_{t,i_{\text{out}}}) = (0,0) \text{ for } t = 0,\ldots,r-1, \\ \texttt{no} & \text{if } \Psi_{\{\texttt{yes},\texttt{no}\}}(M_{r,i_{\text{out}}}) = (0,1) \text{ and} \\ & \quad \Psi_{\{\texttt{yes},\texttt{no}\}}(M_{t,i_{\text{out}}}) = (0,0) \text{ for } t = 0,\ldots,r-1, \end{cases}$$

where Ψ is the Parikh mapping, and $M_{t,i_{\text{out}}}$ is the multiset over $\Gamma \setminus \mathcal{E}$ associated with the output region at the configuration $\mathcal{C}_t$, in particular, $M_{r,i_{\text{out}}}$ is the multiset over $\Gamma \setminus \mathcal{E}$ associated with the output region at the halting configuration $\mathcal{C}_r$.

We say that a computation $\mathcal{C}$ is an *accepting computation* (respectively, *rejecting computation*) if $\text{Output}(\mathcal{C}) = \texttt{yes}$ (respectively, $\text{Output}(\mathcal{C}) = \texttt{no}$), that is, if object `yes` (respectively, object `no`) appears in the output region associated with the corresponding halting configuration of $\mathcal{C}$, and neither object `yes` nor `no` appears in the output region associated with any non-halting configuration of $\mathcal{C}$.

Let us notice that if a recognizer tissue P system

$$\Pi = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\text{in}}, i_{\text{out}})$$

has a symport rule of the type $(i, \lambda/u, 0)$, then $\text{alp}h(u) \cap (\Gamma \setminus \mathcal{E}) \neq \emptyset$, that is, the multiset u must contain some object from $\Gamma \setminus \mathcal{E}$ because on the contrary, all computations of Π would be non-halting.

For each natural number $k \geq 1$, we denote by $\mathbf{TDC}(k)$ (respectively, $\mathbf{TDS}(k)$ or $\mathbf{TDA}(k)$) the class of recognizer tissue P systems with cell division and with communication rules (respectively, allowing only symport or antiport rules) of length at most k. In the case of tissue P systems without environment, we denote by $\widehat{\mathbf{TDC}}(k)$ the class of recognizer tissue P systems with cell division and with communication rules of length at most k.

3.2 *Polynomial complexity classes of recognizer tissue P systems*

Next, we define what solving a decision problem in the framework of tissue P systems in a uniform and an efficient way means. Bearing in mind that they provide devices with a finite description, a numerable family of tissue P systems will be necessary in order to solve a decision problem.

DEFINITION 3.5 *We say that a decision problem $X = (I_X, \theta_X)$ is solvable in a uniform way and polynomial time by a family $\mathbf{\Pi} = \{\Pi(n) : n \in \mathbb{N}\}$ of recognizer tissue P systems with cell division (with or without environment) if the following holds*:

- *The family $\mathbf{\Pi}$ is polynomially uniform by Turing machines, that is, there exists a deterministic Turing machine working in polynomial time which constructs the system $\Pi(n)$ from $n \in \mathbb{N}$.*
- *There exists a pair* (cod, s) *of polynomial-time computable functions over I_X such that*:
  - (i) *for each instance $u \in I_X$, $s(u)$ is a natural number, and* $\text{cod}(u)$ *is an input multiset of the system $\Pi(s(u))$;*
  - (ii) *for each $n \in \mathbb{N}$, $s^{-1}(n)$ is a finite set;*
  - (iii) *the family $\mathbf{\Pi}$ is polynomially bounded with regard to (X, cod, s), that is, there exists a polynomial function p, such that for each $u \in I_X$, every computation of $\Pi(s(u))$ with input* $\text{cod}(u)$ *is halting and it performs at most $p(|u|)$ steps;*
  - (iv) *the family $\mathbf{\Pi}$ is sound with regard to (X, cod, s), that is, for each $u \in I_X$, if there exists an accepting computation of $\Pi(s(u))$ with input* $\text{cod}(u)$, *then $\theta_X(u) = 1$;*

(v) *the family* $\mathbf{\Pi}$ *is complete with regard to* (X, cod, s), *that is, for each* $u \in I_X$, *if* $\theta_X(u) = 1$, *then every computation of* $\Pi(s(u))$ *with input* $\mathrm{cod}(u)$ *is an accepting one.*

From the above soundness and completeness conditions, we deduce that every P system $\Pi(n)$ is *confluent*, in the following sense: every computation of a system with the *same* input multiset must always give the *same* answer.

Let $\mathbf{R}$ be a class of recognizer tissue P systems. We denote by $\mathbf{PMC_R}$ the set of all decision problems which can be solved in a uniform way and polynomial time by means of families of systems from $\mathbf{R}$. The class $\mathbf{PMC_R}$ is closed under complement and polynomial-time reductions [13].

Next, we prove two technical results concerning recognizer tissue P systems.

PROPOSITION 3.6 *Let* $\Pi = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\mathrm{in}}, i_{\mathrm{out}})$ *be a recognizer tissue P system with communication rules with length at most* k, $k \geq 2$, *and without cell division. Let* $M = |\mathcal{M}_1 + \cdots + \mathcal{M}_q|$ *and let* $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_m)$ *be a computation of* Π. *The following holds*:

(a) $|\mathcal{C}_0^*| = M$ *and for each* t, $0 \leq t < m$, *we have*

$$|\mathcal{C}_{t+1}^*| \leq (|\mathcal{C}_t^*| + M) \cdot (k-1).$$

(b) *For each* t, $1 \leq t \leq m$, *we have*

$$|\mathcal{C}_t^*| \leq 2M(k-1)^t + M(k-1)^{t-1} + \cdots + M(k-1).$$

Proof (a) Obviously, $|\mathcal{C}_0^*| = |\mathcal{C}_0(0) + \mathcal{C}_0(1) + \cdots + \mathcal{C}_0(q)| = |\mathcal{M}_1 + \cdots + \mathcal{M}_q| = M$. Let us compute $\mathcal{C}_{t+1}^* = \mathcal{C}_{t+1}(0) + \mathcal{C}_{t+1}(1) + \cdots + \mathcal{C}_{t+1}(q)$.

First, let us see what is the contribution to $\mathcal{C}_{t+1}^*$ of multiset $\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q)$.

- Multiset $(\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q))_a$: some objects from $\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q)$, generically denoted by $a's$, that either do not evolve or they evolve by the application of communication rules between different cells of the system. These objects will pass to $\mathcal{C}_{t+1}(1) + \cdots + \mathcal{C}_{t+1}(q)$, directly in the first case, and after an interchange process, in the second case.
- Multiset $(\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q))_b$: some objects from $\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q)$, generically denoted by $b's$, evolve by the application of rules of the type $(i, u/v, 0)$, where u is a multiset of $b's$ objects. These objects $b's$ allow to bring objects from the environment into cells i. Each such object b can bring in a cell at most $k-1$ objects from the environment. Besides, any object $b \in \Gamma \setminus \mathcal{E}$ acting in these rules must be considered in the multiset $\mathcal{C}_{t+1}(0)$.

Now, let us see what is the contribution to $\mathcal{C}_{t+1}^*$ of multiset $\mathcal{C}_t(0)$.

- Some objects from $\mathcal{C}_t(0)$ do not evolve and they directly pass to $\mathcal{C}_{t+1}(0)$.
- The remaining objects from $\mathcal{C}_t(0)$ will evolve by means of rules of the type $(i, b's/v, 0)$ or rules of the type $(i, \lambda/v, 0)$. In this last case, the string v must contain some objects from $\Gamma \setminus \mathcal{E}$ (at least, some objects from $\mathcal{C}_t(0)$ that evolve). Then, the number of objects that can arrive into cells by the application of these rules is, at most, $M \cdot (k-1)$.

Hence,

$$\begin{aligned} |\mathcal{C}_{t+1}^*| &\leq |\mathcal{C}_t(0)| + |(\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q))_a| + |(\mathcal{C}_t(1) + \cdots + \mathcal{C}_t(q))_b| \cdot (k-1)M \cdot (k-1) \\ &\leq |\mathcal{C}_t^*| \cdot (k-1) + M \cdot (k-1) = (|\mathcal{C}_t^*| + M) \cdot (k-1). \end{aligned}$$

(b) We will prove the second part of the theorem by induction on t.

For $t = 1$, the result is trivial because of $|\mathcal{C}_1^*| \leq (|\mathcal{C}_0^*| + M) \cdot (k-1) = 2M \cdot (k-1)$.

Let t be such that $1 < t < r$ and the result holds for t. Then,

$$\begin{aligned}|\mathcal{C}^*_{t+1}| &\leq (|\mathcal{C}^*_t| + M) \cdot (k-1) \\ &\overset{i.h.}{\leq} (2M(k-1)^t + M(k-1)^{t-1} + \cdots + M(k-1) + M) \cdot (k-1) \\ &\leq 2M(k-1)^{t+1} + M(k-1)^t + \cdots + M(k-1)^2 + M(k-1).\end{aligned}$$

■

PROPOSITION 3.7 *Let $\mathbf{\Pi} = \{\Pi(n) : n \in \mathbb{N}\}$ be a family of recognizer tissue P systems from **TDC**(k), where $k \geq 2$, solving a decision problem $X = (I_X, \theta_X)$ in polynomial time. Let* (cod, s) *be a polynomial encoding associated with that solution. There exists a polynomial function $r(n)$ such that for each instance, $u \in I_X$, $2^{r(|u|)}$ is an upper bound of the number of objects in all cells of the system $\Pi(s(u)) + \mathrm{cod}(u)$ along any computation.*

Proof Let $p(n)$ be a polynomial function such that for each $u \in I_X$, every computation of $\Pi(s(u)) + \mathrm{cod}(u)$ is halting and it performs at most $p(|u|)$ steps.

Let $u \in I_X$ be an instance of X and

$$\Pi(s(u)) + \mathrm{cod}(u) = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\mathrm{in}}, i_{\mathrm{out}}).$$

Let $M = |\mathcal{M}_1 + \cdots + \mathcal{M}_q|$. Let $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_m)$, $0 \leq m \leq p(|u|)$, be a computation of Π.

First, let us suppose that we apply only communication rules at m consecutive transition steps. From Proposition 3.6, we deduce that

(a) $|\mathcal{C}^*_0| = M$ and $|\mathcal{C}^*_{t+1}| \leq (|\mathcal{C}^*_t| + M) \cdot (k-1)$, for each t, $0 \leq t < m$.
(b) $|\mathcal{C}^*_t| \leq 2M(k-1)^t + M(k-1)^{t-1} + \cdots + M(k-1)$, for each t, $1 \leq t \leq m$.

Thus, if we apply in a consecutive way the maximum possible number of communication rules (without applying any division rules) to the system $\Pi(s(u)) + \mathrm{cod}(u)$, at any instant of any computation of the system, $2M \cdot (k-1)^{p(|u|)}$ is an upper bound of the number of objects in the whole system.

Now, let us consider the effect of applying in a consecutive way the maximum possible number of division rules (without applying any communication rules) to the system $\Pi(s(u)) + \mathrm{cod}(u)$ when the initial configuration has $2M \cdot (k-1)^{p(|u|)}$ objects. After that an upper bound of the number of objects in the whole system by any computation is $2M \cdot (k-1)^{p(|u|)} \cdot 2^{p(|u|)} \cdot p(|u|)$. Hence, for each instance $u \in I_X$, the number of objects in all cells of the system $\Pi(s(u)) + \mathrm{cod}(u)$ is, at most, $M \cdot (k-1)^{p(|u|)} \cdot 2^{p(|u|)+1} \cdot p(|u|)$.

Then, we consider a polynomial function $r(n)$ such that

$$r(|u|) \geq \log(M) + p(|u|) \cdot \log(k-1) + p(|u|) + 1 + \log(p(|u|))$$

for each instance $u \in I_X$. The polynomial function $r(n)$ fulfils the property required. ■

COROLLARY 3.8 *Let $\mathbf{\Pi} = \{\Pi(n) : n \in \mathbb{N}\}$ be a family of recognizer tissue P systems with cell division solving a decision problem $X = (I_X, \theta_X)$ in polynomial time. Let* (cod, s) *be a polynomial encoding associated with that solution. Then, there exists a polynomial function $r(n)$ such that for each instance $u \in I_X$, $2^{r(|u|)}$ is an upper bound of the number of objects from $\mathcal{E}$ which are moved from the environment to all cells of the system $\Pi(s(u)) + \mathrm{cod}(u)$ along any computation.*

Proof It suffices to note that from Proposition 3.7, there exists a polynomial function $r(n)$ such that for each instance $u \in I_X$, $2^{r(|u|)}$ is an upper bound of the number of objects in all cells of the system $\Pi(s(u)) + \mathrm{cod}(u)$. ■

4. Simulating tissue P systems with cell division by means of tissue P systems with cell division and without environment

The goal of this section is to show that any recognizer tissue P system with cell division can be simulated by a tissue P system with cell division and without environment in an efficient way.

First of all, we define the meaning of efficient simulations in the framework of recognizer tissue P systems.

DEFINITION 4.1 *Let Π and Π' be recognizer tissue P systems. We say that Π' simulates Π in an efficient way if the following holds*:

(1) *Π' can be constructed from Π by a deterministic Turing machine working in polynomial time.*
(2) *There exists an injective function, f, from the set $\mathbf{Compz}(\Pi)$ of computations of Π onto the set $\mathbf{Comp}(\Pi')$ of computations of Π' such that:*
   - *There exists a deterministic Turing machine that constructs computation $f(\mathcal{C})$ from computation $\mathcal{C}$ in polynomial time.*
   - *A computation $\mathcal{C} \in \mathbf{Comp}(\Pi)$ is an accepting computation if and only if $f(\mathcal{C}) \in \mathbf{Comp}(\Pi')$ is an accepting one.*
   - *There exists a polynomial function $p(n)$ such that for each $\mathcal{C} \in \mathbf{Comp}(\Pi)$, we have $|f(\mathcal{C})| \leq p(|\mathcal{C}|)$.*

Now, for every family of recognizer tissue P system with cell division solving a decision problem, we design a family of recognizer tissue P systems with cell division and *without environment* efficiently simulating it, according to Definition 4.1.

DEFINITION 4.2 *Let $\mathbf{\Pi} = \{\Pi(n) : n \in \mathbb{N}\}$ be a family of recognizer tissue P systems solving a decision problem $X = (I_X, \theta_X)$ in polynomial time according to Definition* 3.5, *and let $r(n)$ be a polynomial function such that for each instance $u \in I_X$, $2^{r(|u|)}$ is an upper bound of the number of objects from $\mathcal{E}$ which are moved from the environment to all cells of the system by any computation of $\Pi(s(u)) + \mathrm{cod}(u)$.*

For each $n \in \mathbb{N}$, let $\Pi(n) = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{\mathrm{in}}, i_{\mathrm{out}})$ be an element of the previous family of degree q, and for the sake of simplicity, we write r instead of $r(n)$. Let us consider the recognizer tissue P system of degree $q_1 = 1 + q \cdot (r+2) + |\mathcal{E}|$ with cell division and without environment

$$\mathbf{S}(\Pi(n)) = (\Gamma', \Sigma', \mathcal{M}'_0, \mathcal{M}'_1, \ldots, \mathcal{M}'_{q_1}, \mathcal{R}', i'_{\mathrm{in}}, i'_{\mathrm{out}})$$

defined as follows:

- $\Gamma' = \Gamma \cup \{\alpha_i : 0 \leq i \leq r-1\}$.
- $\Sigma' = \Sigma$.
- *For each cell $i \in \{1, \ldots, q\}$ of $\Pi(n)$, there is a cell in $\mathbf{S}(\Pi(n))$ with the same label. In addition, $\mathbf{S}(\Pi(n))$ has*:
  (i) *$r+1$ new cells, labelled by $(i,0), (i,1), \ldots, (i,r)$, respectively, for each $i \in \{1, \ldots, q\}$.*
  (ii) *A distinguished cell labelled by 0.*
  (iii) *A new cell, labelled by l_b, for each $b \in \mathcal{E}$.*

- *Initial multisets*: $\mathcal{M}'_{l_b} = \{\alpha_0\}$, *for each* $b \in \mathcal{E}$, *and*

$$\left.\begin{array}{rcl} \mathcal{M}'_{(i,0)} & = & \mathcal{M}_i \\ \mathcal{M}'_{(i,1)} & = & \emptyset \\ \vdots & & \vdots \\ \mathcal{M}'_{(i,r)} & = & \emptyset \\ \mathcal{M}'_i & = & \emptyset \end{array}\right\} (1 \leq i \leq q)$$

- *Set of rules:*

$$\begin{aligned} \mathcal{R}' = \mathcal{R} &\cup \{[\alpha_j]_{l_b} \to [\alpha_{j+1}]_{l_b}[\alpha_{j+1}]_{l_b} : \ b \in \mathcal{E} \ \wedge \ 0 \leq j \leq r-2\} \\ &\cup \{[\alpha_{p-1}]_{l_b} \to [b]_{l_b}\,[b]_{l_b} : \ b \in \mathcal{E}\} \\ &\cup \left\{\left(l_b, \frac{b}{\lambda}0\right) : \ b \in \mathcal{E}\right\} \\ &\cup \left\{g\left((i,j), \frac{a}{\lambda}, (i,j+1)\right) : a \in \Gamma \ \wedge \ 1 \leq i \leq q \ \wedge \ 0 \leq j \leq p-1\right\} \\ &\cup \{((i,r),\ a/\lambda, i) : \ a \in \Gamma \ \wedge \ 1 \leq i \leq q\} \end{aligned}$$

- $i'_{\text{in}} = (i_{\text{in}}, 0)$ *and* $i'_{\text{out}} = 0$.

Let us notice that $\mathbf{S}(\Pi(n))$ can be considered as an *extension* of $\Pi(n)$ *without environment*, in the following sense:

- $\Gamma \subseteq \Gamma'$, $\Sigma \subseteq \Sigma'$ and $\mathcal{E} = \emptyset$.
- Each cell in $\Pi(n)$ is also a cell in $\mathbf{S}(\Pi(n))$.
- There is a distinguished cell in $\mathbf{S}(\Pi(n))$ labelled by 0 which plays the role of environment of $\Pi(n)$.
- $\mathcal{R} \subseteq \mathcal{R}'$, and now 0 is the label of a 'normal cell' in $\mathbf{S}(\Pi(n))$.

In what follows throughout this section, we use $\Pi(n)$, r and $\mathbf{S}(\Pi(n))$ in the same context as in the Definition 4.2.

Next, we analyse the structure of the computations of system $\mathbf{S}(\Pi(n))$ and we compare them with the computations of $\Pi(n)$.

LEMMA 4.3 *Let* $\mathcal{C}' = (\mathcal{C}'_0, \mathcal{C}'_1, \ldots)$ *be a computation of* $\mathbf{S}(\Pi(n))$. *For each* t $(1 \leq t \leq r)$, *the following holds*:

- $\mathcal{C}'_t(i) = \emptyset$, *for* $0 \leq i \leq q$.
- *For each* $1 \leq i \leq q$ *and* $0 \leq j \leq r$, *we have*

$$\mathcal{C}'_t(i,j) = \begin{cases} \mathcal{M}_i & \text{if } j = t, \\ \emptyset & \text{if } j \neq t, \end{cases}$$

- *For each* $b \in \mathcal{E}$, *there exist* 2^t *cells labelled by* l_b *whose contents are*

$$\mathcal{C}'_t(l_b) = \begin{cases} \alpha_t & \text{if } 1 \leq t \leq r-1, \\ b & \text{if } t = r, \end{cases}$$

Proof By induction on t.

Let us start with the basic case $t = 1$. In the initial configuration of system $\mathbf{S}(\Pi(n))$, $\mathcal{C}'_0$, the following holds:

- $\mathcal{C}'_0(i) = \emptyset$, for $0 \leq i \leq q$.
- For each $1 \leq i \leq q$, we have $\mathcal{C}'_0(i,0) = \mathcal{M}_i$, and $\mathcal{C}'_0(i,j) = \emptyset$, for $1 \leq j \leq r$.
- For each $b \in \mathcal{E}$, there exists only one cell labelled by l_b whose contents is $\{\alpha_0\}$.

In configuration $\mathcal{C}'_0$, only the following rules are applicable:

- $[\alpha_0]_{l_b} \to [\alpha_1]_{l_b}\, [\alpha_1]_{l_b}$, for each $b \in \mathcal{E}$.
- $((i,0), a/\lambda, (i,1))$, for each $a \in \mathrm{supp}(\mathcal{M}_i)$.

Thus,

- For each i $(1 \leq i \leq q)$, we have

$$\begin{aligned} \mathcal{C}'_1(i) &= \emptyset, \\ \mathcal{C}'_1(0) &= \emptyset, \\ \mathcal{C}'_1(i,0) &= \emptyset, \\ \mathcal{C}'_1(i,1) &= \mathcal{M}_i, \\ \mathcal{C}'_1(i,j) &= \emptyset \quad \text{for } 2 \leq j \leq r. \end{aligned}$$

- For each $b \in \mathcal{E}$, there are two cells labelled by l_b whose contents is $\{\alpha_1\}$.

Hence, the result holds for $t = 1$.

Let t be such that $1 \leq t < r$, and let us assume, by induction hypothesis, the result holds for t, that is,

- $\mathcal{C}'_t(i) = \emptyset$, for $0 \leq i \leq q$.
- For each $1 \leq i \leq q$, and $0 \leq j \leq r$, we have

$$\mathcal{C}'_t(i,j) = \begin{cases} \mathcal{M}_i & \text{if } j = t, \\ \emptyset & \text{if } j \neq t, \end{cases}$$

- For each $b \in \mathcal{E}$, there exist 2^t cells labelled by l_b whose contents is $\mathcal{C}'_t(l_b) = \{\alpha_t\}$ (because $t \leq r - 1$).

Then, in configuration $\mathcal{C}'_t$ only the following rules are applicable:

(1) If $t \leq r - 2$, the rules $[\alpha_t]_{l_b} \to [\alpha_{t+1}]_{l_b}\, [\alpha_{t+1}]_{l_b}$, for each $b \in \mathcal{E}$.
(2) If $t = r - 1$, the rules $[\alpha_t]_{l_b} \to [b]_{l_b}\, [b]_{l_b}$, for each $b \in \mathcal{E}$.
(3) $((i,t), a/\lambda, (i,t+1))$, for each $a \in \Gamma$.

From the application of rules of types (1) or (2) on the configuration $\mathcal{C}'_t$, we deduce that there are 2^{t+1} cells labelled by l_b in $\mathcal{C}'_{t+1}$, for each $b \in \mathcal{E}$, whose contents is $\{\alpha_{t+1}\}$, if $t \leq r - 2$, or $\{b\}$, if $t = r - 1$.

From the application of rules of type (3) on the configuration $\mathcal{C}'_t$, we deduce that

$$\mathcal{C}'_{t+1}(i,j) = \begin{cases} \mathcal{M}_i & \text{if } j = t+1, \\ \emptyset & \text{if } 0 \leq j \leq r \ \wedge\ j \neq t+1. \end{cases}$$

Bearing in mind that no other rule of system $\mathbf{S}(\Pi(n))$ is applicable, we deduce that $\mathcal{C}'_{t+1}(i) = \emptyset$, for $0 \leq i \leq q$.

This completes the proof of this Lemma. ■

LEMMA 4.4 *Let $\mathcal{C}' = (\mathcal{C}'_0, \mathcal{C}'_1, \dots)$ be a computation of the tissue P system $\mathbf{S}(\Pi(n))$. In configuration $\mathcal{C}'_{r+1}$ the following holds*:

- $\mathcal{C}'_{r+1}(0) = b_1^{2^r} \dots b_\alpha^{2^r}$, *where* $\mathcal{E} = \{b_1, \dots, b_\alpha\}$.
- $\mathcal{C}'_{r+1}(i) = \mathcal{M}_i = C_0(i)$, *for* $1 \leq i \leq q$.
- $\mathcal{C}'_{r+1}(i,j) = \emptyset$, *for* $1 \leq i \leq q$, $0 \leq j \leq r$.
- *There exist* 2^r *empty cells labelled by* l_b, *for each* $b \in \mathcal{E}$.

Proof From Lemma 4.3, in the configuration $\mathcal{C}'_r$ the following holds:

- $\mathcal{C}'_r(i) = \emptyset$, for $0 \leq i \leq q$.
- For each i $(1 \leq i \leq q)$ we have

$$\mathcal{C}'_r(i,j) = \begin{cases} \mathcal{M}_i & \text{if } j = r, \\ \emptyset & \text{if } j \neq r. \end{cases}$$

- For each $b \in \mathcal{E}$, there exist 2^r cells labelled by l_b whose contents is $\{b\}$.

In configuration $\mathcal{C}'_r$, only the following rules are applicable:

- $((i,r), a/\lambda, i)$, for each $a \in \Gamma \cap \mathrm{supp}(\mathcal{M}_i)$.
- $(l_b, b/\lambda, 0)$, for each $b \in \mathcal{E}$.

Thus,

- $\mathcal{C}'_{r+1}(0) = b_1^{2^r} \dots b_\alpha^{2^r}$, where $\mathcal{E} = \{b_1, \dots, b_\alpha\}$.
- $\mathcal{C}'_{r+1}(i) = \mathcal{M}_i = C_0(i)$, for $1 \leq i \leq q$.
- $\mathcal{C}'_{r+1}(i,j) = \emptyset$, for $1 \leq i \leq q$ and $0 \leq j \leq r$.
- There exist 2^r empty cells labelled by l_b, for each $b \in \mathcal{E}$. ■

DEFINITION 4.5 *Let $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1, \dots, \mathcal{C}_m)$ be a halting computation of $\Pi(n)$. Then, we define the computation $\mathbf{S}(\mathcal{C}) = (\mathcal{C}'_0, \mathcal{C}'_1, \dots, \mathcal{C}'_r, \mathcal{C}'_{r+1}, \dots, \mathcal{C}'_{r+1+m})$ of $\mathbf{S}(\Pi(n))$ as follows*:

- *The initial configuration is:*

$$\begin{aligned} \mathcal{C}'_0(i) &= \emptyset \quad \text{for } 0 \leq i \leq q, \\ \mathcal{C}'_0(i,0) &= C_0(i) \quad \text{for } 1 \leq i \leq q, \\ \mathcal{C}'_0(i,j) &= \emptyset \quad \text{for } 1 \leq i \leq q \text{ and } 1 \leq j \leq r, \\ \mathcal{C}'_0(l_b) &= \alpha_0 \quad \text{for each } b \in \mathcal{E}. \end{aligned}$$

- *The configuration $\mathcal{C}'_t$, for $1 \leq t \leq r$, is described by Lemma* 4.3.
- *The configuration $\mathcal{C}'_{r+1}$ is described by Lemma* 4.4.
- *The configuration $\mathcal{C}'_{r+1+s}$, for $0 \leq s \leq m$, coincides with the configuration $\mathcal{C}_s$ of Π, that is, $\mathcal{C}_s(i) = \mathcal{C}'_{r+1+s}(i)$, for $1 \leq i \leq q$. The contents of the remaining cells (excluding cell 0) at configuration $\mathcal{C}'_{r+1+s}$ are equal to the contents of that cell at configuration $\mathcal{C}'_{r+1}$, that is, these cells do not evolve after step $r+1$.*

That is, every computation $\mathcal{C}$ of $\Pi(n)$ can be 'reproduced' by a computation $\mathbf{S}(\mathcal{C})$ of $\mathbf{S}(\Pi(n))$ with a delay: the evolution of the contents of cells $1, \ldots, q$ in the computation of $\mathbf{S}(\mathcal{C})$ from step $r+1$ to step $r+1+m$ is a replica of the computation $\mathcal{C}$ of $\Pi(n)$.

From Lemmas 4.3 and 4.4, we deduce the following:

(a) $\mathbf{S}(\mathcal{C})$ is a computation of $\mathbf{S}(\Pi(n))$.
(b) $\mathbf{S}$ is an injective function from $\mathbf{Comp}(\Pi(n))$ onto $\mathbf{Comp}(\mathbf{S}(\Pi(n)))$.

Moreover, if r is a polynomial function on the size of $\Pi(n)$, then we have the following result.

PROPOSITION 4.6 *The tissue P system* $\mathbf{S}(\Pi(n))$ *defined in* 4.2 *simulates* $\Pi(n)$ *in an efficient way.*

Proof In order to show that $\mathbf{S}(\Pi(n))$ can be constructed from $\Pi(n)$ by a deterministic Turing machine working in polynomial time, it is enough to note that the amount of resources needed to construct $\mathbf{S}(\Pi(n))$ from $\Pi(n)$ is polynomial in the size of the initial resources of $\Pi(n)$. Indeed,

(1) The size of the alphabet of $\mathbf{S}(\Pi(n))$ is $|\Gamma'| = |\Gamma| + r$.
(2) The initial number of cells of $\mathbf{S}(\Pi(n))$ is $1 + q \cdot (r+2) + |\mathcal{E}|$.
(3) The initial number of objects of $\mathbf{S}(\Pi(n))$ is the initial number of objects of $\Pi(n)$ plus $|\mathcal{E}|$.
(4) The number of rules of $\mathbf{S}(\Pi(n))$ is $|\mathcal{R}'| = |\mathcal{R}| + (r+1) \cdot |\mathcal{E}| + |\Gamma| \cdot q \cdot (r+1)$.
(5) The maximal length of a communication rule of $\mathbf{S}(\Pi(n))$ is equal to that of $\Pi(n)$.

From Lemmas 4.3 and 4.4, we deduce that:

(a) Every computation $\mathcal{C}'$ of $\mathbf{S}(\Pi(n))$ has associated a computation $\mathcal{C}$ of $\Pi(n)$ such that $\mathbf{S}(\mathcal{C}) = \mathcal{C}'$.
(b) The function $\mathbf{S}$ is injective.
(c) A computation $\mathcal{C}$ of Π is an accepting computation if and only if $\mathbf{S}(\mathcal{C})$ is an accepting computation of $\mathbf{S}(\Pi(n))$.

Finally, let us notice that if $\mathcal{C}$ is a computation of $\Pi(n)$ with length m, then $\mathbf{S}(\mathcal{C})$ is a computation of $\mathbf{S}(\Pi(n))$ with length $r+1+m$. ■

5. Computational complexity classes of tissue P systems with cell division and without environment

In this section, we analyse the role of the environment in the efficiency of tissue P systems with cell division. That is, we study the power of these P systems with respect to the computational efficiency when the alphabet of the environment is an empty set.

THEOREM 5.1 *For each* $k \in \mathbb{N}$, *we have* $\mathbf{PMC}_{\mathbf{TDC}(k+1)} = \mathbf{PMC}_{\widehat{\mathbf{TDC}}}(k+1)$.

Proof Let us recall that $\mathbf{PMC}_{\mathbf{TDC}(1)} = \mathbf{P}$ (see [3] for details). Then,

$$\mathbf{P} \subseteq \mathbf{PMC}_{\widehat{\mathbf{TDC}}(1)} \subseteq \mathbf{PMC}_{\mathbf{TDC}(1)} = \mathbf{P}.$$

Thus, the result holds for $k = 0$.

Let us show the result for $k \geq 1$. Since $\widehat{\mathbf{TDC}}(k+1) \subseteq \mathbf{TDC}(k+1)$, it suffices to prove that $\mathbf{PMC}_{\mathbf{TDC}(k+1)} \subseteq \mathbf{PMC}_{\widehat{\mathbf{TDC}}(k+1)}$. For that, let $X \in \mathbf{PMC}_{\mathrm{TDC}(k+1)}$.

Let $\{\Pi(n) : n \in \mathbb{N}\}$ be a family of tissue P systems from $\mathbf{TDC}(k+1)$ solving X according to Definition 3.5. Let (cod, s) be a polynomial encoding associated with that solution. Let $u \in I_X$ be an instance of the problem X that will be processed by the system $\Pi(s(u)) + \mathrm{cod}(u)$. According to Proposition 3.7, let $r(n)$ be a polynomial function such that $2^{r(|u|)}$ is an upper bound of the

number of objects from $\mathcal{E}$ which are moved from the environment to all cells of the system by any computation of $\Pi(s(u)) + \mathrm{cod}(u) = (\Gamma, \mathcal{E}, \Sigma, \mathcal{M}_1, \ldots, \mathcal{M}_{i_{\mathrm{in}}} + \mathrm{cod}(u), \ldots, \mathcal{M}_{q_1}, \mathcal{R}, i_{\mathrm{in}}, i_{\mathrm{out}})$.

Then, we consider the tissue P system without environment

$$\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u) = (\Gamma', \Sigma', \mathcal{M}'_0, \mathcal{M}'_1, \ldots, \mathcal{M}'_{i_{\mathrm{in}}} + \mathrm{cod}(u), \ldots, \mathcal{M}'_{q_1}, \mathcal{R}', i'_{\mathrm{in}}, i'_{\mathrm{out}})$$

according to Definition 4.2, where $q_1 = 1 + q \cdot (r(|u|) + 2) + |\mathcal{E}|$.

Therefore, $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$ is a tissue P system from $\widehat{\mathbf{TDC}}(k+1)$ such that it verifies the following:

- A distinguished cell labelled by 0 has been considered, which will play the role of the environment at the system $\Pi(s(u)) + \mathrm{cod}(u)$.
- At the initial configuration, it has enough objects in cell 0 in order to simulate the behaviour of the environment of the system $\Pi(s(u)) + \mathrm{cod}(u)$.
- After $r(n) + 1$ steps, the computations of $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$ replicate the computations of $\Pi(s(u)) + \mathrm{cod}(u)$, as far as cells $1, \ldots, q$ are concerned.

In order to simulate any computation $\mathcal{C}$ of $\Pi(s(u)) + \mathrm{cod}(u)$ by a tissue P system without environment in an efficient way, we need to have enough objects from $\mathcal{E}$ available ($2^{r(n)}$ copies of each) in the cell of $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$ labelled by 0. For this purpose,

- For each $b \in \mathcal{E}$, we consider a cell in $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$ labelled by l_b which only contains object α_0 initially. We also consider the following rules:
  - $[\alpha_j]_{l_b} \rightarrow [\alpha_{j+1}]_{l_b}\, [\alpha_{j+1}]_{l_b}$, for $0 \leq j \leq r(|u|) - 2$.
  - $[\alpha_{p(n)-1}]_{l_b} \rightarrow [b]_{l_b}\, [b]_{l_b}$.
  - $(l_b, b/\lambda, 0)$.
- By applying the previous rules, after $r(|u|)$ transition steps we get $2^{r(|u|)}$ cells labelled by l_b, for each $b \in \mathcal{E}$ in such a way that each of them contains only an object b. Finally, by applying the third rule, we get $2^{r(|u|)}$ copies of objects b in cell 0, for each $b \in \mathcal{E}$.

Therefore, after the execution of $r(|u|) + 1$ transition steps in each computation of $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$ in cell 0 of the corresponding configuration, we have $2^{r(|u|)}$ copies of each object $b \in \mathcal{E}$. This number of copies is enough to simulate any computation $\mathcal{C}$ of $\Pi(s(u)) + \mathrm{cod}(u)$ with the system $\mathbf{S}(\Pi(s(u))) + \mathrm{cod}(u)$.

From Proposition 4.6, we deduce that the family $\{\mathbf{S}(\Pi(n)) : n \in \mathbb{N}\}$ solves X in polynomial time according to Definition 3.5. Hence, $X \in \mathbf{PMC}_{\widehat{\mathbf{TDC}}(k+1)}$. ■

5.1 *Borderlines of efficiency*

Let us highlight two consequences of Theorem 5.1, in the context of existing results in the literature about computational complexity classes associated with tissue P systems. On the one hand, in [4] it has been shown that $\mathbf{P} = \mathbf{PMC}_{\widehat{\mathbf{TSC}}}$. In particular, $\mathbf{P} = \mathbf{PMC}_{\widehat{\mathbf{TSC}}(3)}$. On the other hand, Vertex-Covere $\mathbf{PMC}_{\mathbf{TDC}(3)}$ [2], and from Theorem 5.1, we deduce that $\mathbf{PMC}_{\mathbf{TDC}(3)} = \mathbf{PMC}_{\widehat{\mathbf{TDC}}(3)}$. Thus, in the framework of tissue P systems without environment and with communication rules with length at most 3, the kind of rules (separation versus division) provides a new borderline between the efficiency and non-efficiency.

On the one hand, it is well known that $\mathbf{P} = \mathbf{PMC}_{\mathbf{TDC}(1)}$ [3]. On the other hand, in [15], a uniform and polynomial time solution of the HAM-CYCLE problem by a family of tissue P systems from $\mathbf{TDC}(2)$. Thus, $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathbf{TDC}(2)} = \mathbf{PMC}_{\widehat{\mathbf{TDC}}(2)}$. Hence, from Theorem 5.1 we deduce that in the framework of tissue P systems with cell division and without environment, the length of communication rules provides a new borderline of the tractability, assuming that $\mathbf{P} \neq \mathbf{NP}$: passing from one to two amounts to passing from non-efficiency to efficiency.

6. Conclusions and further work

Initial configurations of ordinary tissue P systems have an arbitrarily large amount of copies of some kind of objects belonging to a distinguished alphabet which specifies the *environment* of the system.

The previous condition seems completely unfair from the computational complexity point of view. In this paper, we study tissue P systems with cell division where there are no objects that verify the above property. Specifically, we show that in tissue P systems with cell division, the environment can be 'removed' without loss of efficiency.

Let us recall that by applying division rules, the two new cells generated by a division rule have exactly the same objects except for at most a pair of different objects, that is, the replication of objects takes place [11]. However, in the biological phenomenon of separation, the contents of the two new cells evolved from a cell can be significantly different. More precisely, separation rules in tissue P systems, as presented in [7], produce two new cells keeping the former label. The object triggering the rule is consumed and the remaining objects are distributed between the new cells, according to a fixed partition of the alphabet.

Recently, it has been proved that the environment plays a significant role in the framework of tissue P systems with cell separation: only tractable problems can be solved in an efficient way by these kinds of P systems without environment [4]. Hence, the environment matters in tissue P systems with cell separation, but does not matter in tissue P systems with cell division.

As future work, we plan to do further research about the relevance of direction in the application of communication rules joint with cell division. In [2], a polynomial time solution of the `Vertex-Cover` problem by a family of tissue P systems from $\mathbf{TDC}(3)$ is shown. Besides, the symport rules that are used in that solution have length at most 2. Then, we can rearrange the rules by adding a new object # to the alphabet $\mathcal{E}$ of the environment, and by replacing rules $(i, u/\lambda, 0)$ by rules $(i, u/\#, 0)$. This proves that `Vertex-Cover` $\in \mathbf{PMC}_{\mathbf{TDA}(3)}$. Thus, $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathbf{TDA}(3)}$. In the case of tissue P systems from $\mathbf{TDC}(2)$, it remains to be investigated whether allowing only symport rules or only antiport rules constitutes a new borderline.

Acknowledgements

The work was supported by Project of Excellence with *Investigador de Reconocida Valía*, from Junta de Andalucía, grant P08 – TIC 04200, and by Project TIN2009-13192 of the Ministerio de Ciencia e Innovación of Spain, cofinanced by FEDER funds.

References

[1] A. Alhazov, *Solving SAT by symport/antiport P systems with membrane division*, Cellular Computing (Complexity Aspects), ESF PESC Exploratory Workshop, Sevilla, Spain, 2005, pp. 1–6.

[2] D. Díaz-Pernil, M.A. Gutiérrez-Naranjo, M.J. Pérez-Jiménez, A. Riscos-Núñez, and A. Romero-Jiménez. *Computational efficiency of cellular division in tissue-like P systems*, Rom. J. Inf. Sci. Tech. 11(3) (2008), pp. 229–241.

[3] R. Gutiérrez-Escudero, M.J. Pérez-Jiménez, and M. Rius-Font, *Characterizing tractability by tissue-like P systems*, Lect. Notes Comput. Sci. 5957 (2010), pp. 289–300. Available at http://link.springer.com/chapter/10.1007/978-3-642-11467-0_21.

[4] L.F. Macías-Ramos, M.J. Pérez-Jiménez, A. Riscos-Núñez, M. Rius-Font, and L. Valencia-Cabrera, *The efficiency of tissue P systems with cell separation relies on the environment*, International Conference on Membrane Computing, Budapest, Hungary, 2012, pp. 277–290.

[5] C. Martín Vide, J. Pazos, Gh. Păun, and A. Rodríguez Patón, *A new class of symbolic abstract neural nets: Tissue P systems*. Lect. Notes Comput. Sci. 2387 (2002), pp. 290–299. Available at http://link.springer.com/chapter/10.1007/3-540-45655-4_32.

[6] C. Martín Vide, J. Pazos, Gh. Păun, and A. Rodríguez Patón, *Tissue P systems*, Theor. Comput. Sci. 296(2) (2003), pp. 295–326.

[7] L. Pan and T.O. Ishdorj, *P systems with active membranes and separation rules*, J. Univ. Comput. Sci. 10(5) (2004), pp. 630–649.
[8] Gh. Păun, *Computing with membranes*, J. Comput. Syst. Sci. 61(1) (2000), pp. 108–143.
[9] Gh. Păun, *Computing with membranes (P systems): Attacking NP-complete problems*, in *Unconventional Models of Computation, UMC'2K, Proceedings of the Second International Conference, Brussel, Belgium, 13–16 December 2000*, I. Antoniou, C. Calude, and M.J. Dinneen, eds., Springer-Verlag, London, 2001, pp. 94–115.
[10] Gh. Păun, *Membrane Computing: An Introduction*, Springer-Verlag, Berlin, 2002.
[11] Gh. Păun, M.J. Pérez-Jiménez, and A. Riscos-Núñez, *Tissue P system with cell division*, Int. J. Comput. Commun. 3(3) (2008), pp. 295–303.
[12] Gh. Păun, G. Rozenberg, and A. Salomaa (eds.), *The Oxford Handbook of Membrane Computing*, Oxford University Press, New York, 2009.
[13] M.J. Pérez-Jiménez, A. Romero-Jiménez, and F. Sancho-Caparrini, *Complexity classes in models of cellular computing with membranes*, Nat. Comput. 2(3) (2003), pp. 265–285.
[14] M.J. Pérez-Jiménez, A. Romero-Jiménez, and F. Sancho-Caparrini, *A polynomial complexity class in P systems using membrane division*, J. Autom. Lang. Comb. 11(4) (2006), pp. 423–434.
[15] A.E. Porreca, N. Murphy, and M.J. Pérez-Jiménez, *An efficient solution of Ham Cycle problem in tissue P systems with cell division and communication rules with length at most 2,* in *Proceedings of the Tenth Brainstorming Week on Membrane Computing, Seville, Spain, 30 January–3 February 2012*, Vol. II, M. García-Quismondo, L.F. Macías-Ramos, Gh. Păun, and L. Valencia-Cabrera, eds., Fénix Editora, Seville, 2012, pp. 141–166.