The Relevance of the Environment
on the Efficiency of Tissue P Systems

Mario J. Pérez-Jiménez[1], Agustín Riscos-Núñez[1], Miquel Rius-Font[2],
and Luis Valencia-Cabrera[1]

[1] Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
University of Sevilla
Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
{marper,ariscosn,lvalencia}@us.es
[2] Department of Applied Mathematics IV
Universitat Politécnica de Catalunya, Spain
mrius@ma4.upc.edu

Abstract. The efficiency of computational devices is usually expressed in terms of their capability to solve computationally hard problems in polynomial time. This paper focuses on tissue P systems, whose efficiency has been shown for several scenarios where the number of cells in the system can grow exponentially, e.g. by using cell division rules or cell separation rules. Moreover, in the first case it suffices to consider very short communication rules with length bounded by two, and in the second one it is enough to consider communication rules with length at most three. This kind of systems have an environment with the property that objects initially located in it appear in an arbitrarily large number of copies, which is a somewhat unfair condition from a computational complexity point of view. In this context, we study the role played by the environment and its ability to handle infinitely many objects, in particular we consider tissue P systems whose environment is initially empty.

1 Introduction

Several different models of cell-like P systems have been successfully used to efficiently solve computationally hard problems by trading space for time. An exponential workspace is created in polynomial time by using some kind of rules, and then massive parallelism is used to simultaneously check all the candidate solutions. Inspired by living cells, several ways for obtaining exponential workspace in polynomial time were proposed: membrane division (*mitosis*) [12], membrane creation (*autopoiesis*) [5], and membrane separation (*membrane fission*) [8][1]. These three ways have given rise to the following models: *P systems with active*

---

[1] The name *separation rule* appeared earlier in [1], but with a slightly different definition.

membranes, *P systems with membrane creation*, and *P systems with membrane separation*, respectively.

A new type of P systems, the so-called *tissue P systems*, was introduced in [7]. The hierarchical membrane structure that was commonly used in the first models, inspired on the way vesicles and compartments are arranged within a cell, is discarded. Instead, an arbitrary graph of connections among elementary membranes (now called *cells*) is considered. That is, the inspiration comes now not from a single cell but from a collection of cooperating cells within a multi-cellular organism, e.g. in a tissue. Moreover, the functioning of tissue P systems heavily relies on the intercellular communication, since objects can move under symport/antiport rules, but cannot be rewritten.

This paper addresses two models of tissue P systems which are of a great interest from a computational complexity point of view. The first one was presented in [14], where the definition of tissue P systems is combined with aspects of the definition of P systems with active membranes, yielding *tissue P systems with cell division*. In these models, cells may replicate, that is, the two new cells generated by a division rule have exactly the same objects except for at most one differing pair of objects. The second model that will be considered is *tissue P systems with cell separation* [9]. In this case, an alternative method for generating an exponential number of cells in linear time is used. When a cell divides, its contents are not replicated, but distributed, according to a fixed partition of the alphabet.

The paper is organized as follows. First, we recall the basic mathematical and theoretical background underlying the definitions of the two tissue P systems models mentioned above, together with the definition of complexity class in the membrane computing framework. Then, Section 3 compares the computational power achieved by cell division and by cell separation, evaluating in both cases the role of the environment. Some concluding remarks summarizing the border-lines of efficiency discussed in the paper are given in Section 4.

2 Tissue P Systems

Let us recall that an *alphabet* Γ is a non–empty set whose elements are called *symbols*. A *multiset* m over an alphabet Γ is a pair $m = (\Gamma, f)$ where $f : \Gamma \to \mathbb{N}$ is a mapping. If $m = (\Gamma, f)$ is a multiset then its *support* is defined as $supp(m) = \{x \in \Gamma \,|\, f(x) > 0\}$. A multiset is finite if its support is a finite set. Let $supp(m) = \{a_1, \ldots, a_k\}$ be the support of a finite multiset, m, then we will denote $m = a_1^{f(a_1)} \ldots a_k^{f(a_k)}$ (here the order is irrelevant), and we say that $f(a_1) + \ldots + f(a_k)$ is the cardinal of m, denoted by $|m|$. The empty multiset is denoted by λ. We also denote by $M_f(\Gamma)$ the set of all finite multisets over Γ.

Let $m_1 = (\Gamma, f_1)$ and $m_2 = (\Gamma, f_2)$ multisets over Γ. The *union* of m_1 and m_2, denoted by $m_1 + m_2$ is the multiset (Γ, g), where $g = f_1 + f_2$, that is, $g(x) = f_1(x) + f_2(x)$ for each $x \in \Gamma$. The *relative complement* of m_2 in m_1, denoted by $m_1 \setminus m_2$ is the multiset (Γ, g), where $g(x) = f_1(x) - f_2(x)$ if $f_1(x) \geq f_2(x)$ and $g(x) = 0$ otherwise.

Definition 1. *A basic tissue P system of degree $q \geq 1$ is a tuple*
$\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$, *where:*

1. Γ *is a finite alphabet and $\mathcal{E}$ is a subset of Γ.*
2. Σ *is an (input) alphabet strictly contained in Γ such that $\mathcal{E} \cap \Sigma = \emptyset$.*
3. $\mathcal{M}_1, \ldots, \mathcal{M}_q$ *are finite multisets over $\Gamma \setminus \Sigma$.*
4. $\mathcal{R}$ *is a finite set of communication rules of the form $(i, u/v, j)$,*
    for $i, j \in \{0, 1, 2, \ldots, q\}, i \neq j, u, v \in M_f(\Gamma)$, and $|u + v| \neq 0$;
5. $i_{in} \in \{1, 2, \ldots, q\}$, *and $i_{out} \in \{0, 1, \ldots, q\}$.*

A *basic tissue P system* $\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$ of degree $q \geq 1$ can be viewed as a set of q cells, labelled by $1, \ldots, q$, with an environment labelled by 0 such that: (a) $\mathcal{M}_1, \ldots, \mathcal{M}_q$ are finite multisets over Γ representing the objects (elements in Γ) initially placed in the q cells of the system; (b) Σ is the input alphabet and $\mathcal{E}$ is the set of objects located initially in the environment of the system, all of them appearing in an *arbitrary number of copies*; and (c) i_{in} represents the input cell, and $i_{out} \in \{0, 1, \ldots, q\}$ indicates the *region* that stores the output of the system (which can be either a distinguished cell when $i_{out} \in \{1, \ldots, q\}$, or the environment when $i_{out} = 0$). If $\mathcal{E} = \emptyset$ then we say that the tissue P system is *without environment*.

A communication rule $(i, u/v, j)$ is *applicable* to regions i, j if the multiset u is contained in region i and multiset v is contained in region j. When applying a communication rule $(i, u/v, j)$, the objects of multiset 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 $|u| + |v|$.

The rules are used in a non-deterministic maximally parallel manner as customary in membrane computing. At each step, we apply a multiset of rules which is *maximal*: no further applicable rule can be added.

A *configuration* at any instant of a basic 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 properly changed along the computation. For each multiset m over the input alphabet Σ, the *initial configuration* with input m is $\mathcal{C}_0 = (\mathcal{M}_1, \cdots, \mathcal{M}_{i_{in}} + m, \cdots, \mathcal{M}_q; \emptyset)$. Therefore, we have an initial configuration associated with each input multiset m (over the input alphabet Σ) in this kind of systems. We will use the notation $(\Pi + m)$ to refer to a P system Π such that its initial configuration is the one associated with m. A configuration is a *halting configuration* if no rule of the system is applicable to it. 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 the rules from $\mathcal{R}$ following the previous remarks.

A *computation* of Π is a (finite or infinite) sequence of configurations such that: (a) the first term of the sequence is the initial configuration $\mathcal{C}_0$ of the system associated with a given input; (b) for each $n \geq 2$ the n–th configuration of the sequence is obtained from the previous configuration by applying a maximal multiset of rules of the system as described above; and (c) if the sequence is

finite (called *halting computation*) then the last term of the sequence must be a halting configuration. Only halting computations give a result, which is encoded by the objects present in the output region i_{out} in the halting configuration. The result of a computation can be defined in various ways, just like in the cell-like case. Obviously, when the output is collected in the environment, symbols from $\mathcal{E}$ must be ignored.

If $\mathcal{C} = \{\mathcal{C}_t\}_{0 \leq t \leq r}$ of Π ($r \in \mathbb{N}$) is a halting computation, then the *length of* $\mathcal{C}$, denoted by $|\mathcal{C}|$, is r.

2.1 Cell Division and Cell Separation

Reproduction is doubtlessly one of the fundamental mechanisms on every living being. Thus, there is a clear motivation to try to get inspiration from the various processes that generate new cells (or new membranes, in general) and to adapt them into the tissue P systems framework. Moreover, as mentioned in the Introduction, division rules (*mitosis*), and separation rules (*membrane fission*) have been already introduced for cell-like P systems [12,8].

Definition 2. *A tissue P system with cell division of degree $q \geq 1$ is a tuple $\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$, where:*

1. *$\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}_c, i_{in}, i_{out})$ is a basic tissue P system, where $\mathcal{R}_c$ is the set of communication rules in $\mathcal{R}$.*
2. *$\mathcal{R}$ may also contain cell division rules of the form $[a]_i \rightarrow [b]_i[c]_i$, where $i \in \{1, 2, \ldots, q\}$, $i \neq i_{out}$ and $a, b, c \in \Gamma$.*

Definition 3. *A tissue P system with cell separation of degree $q \geq 1$ is a tuple $\Pi = (\Gamma, \Gamma_1, \Gamma_2, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{out})$, where:*

1. *$\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}_c, i_{in}, i_{out})$ is a basic tissue P system, where $\mathcal{R}_c$ is the set of communication rules in $\mathcal{R}$.*
2. *$\{\Gamma_1, \Gamma_2\}$ is a partition of Γ, that is, $\Gamma = \Gamma_1 \cup \Gamma_2$, $\Gamma_1, \Gamma_2 \neq \emptyset$, $\Gamma_1 \cap \Gamma_2 = \emptyset$.*
3. *$\mathcal{R}$ may also contain cell separation rules of the form $[a]_i \rightarrow [\Gamma_1]_i[\Gamma_2]_i$, where $i \in \{1, \ldots, q\}$, $a \in \Gamma$ and $i \neq i_{out}$.*

A *tissue P system with cell division* is a basic tissue P system that allows cell division rules. When applying a division rule $[a]_i \rightarrow [b]_i[c]_i$, 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 are replicated and copies of them are placed in the two new cells.

A *tissue P system with cell separation* is a basic tissue P system that allows cell separation rules. When applying a separation rule $[a]_i \rightarrow [\Gamma_1]_i[\Gamma_2]_i$, in reaction with an object a, the cell i is separated into two cells with the same label; at the same time, object a is consumed; all the other objects in the cell are distributed (not replicated): those from Γ_1 are placed in the first cell, while those from Γ_2 are placed in the second cell. The output cell i_{out} cannot be divided nor separated.

The label of a cell precisely identifies the rules which can be applied to it. Note that in the previous definitions $\{1, \dots, q\}$ is used as the set of labels, but without loss of generality any finite set can be considered instead. The rules are used in a non-deterministic maximally parallel manner with the following restriction: when a cell is divided (or separated), the objects inside that cell do not get involved in any communication rule during this step. The two new resulting cells could participate in the interaction with other cells or the environment by means of communication rules at the next step – provided that they are not divided (or separated) again.

2.2 Recognizer Tissue P Systems

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. There are many different ways to describe instances of a decision problem, but we assume that each problem has associated with it a fixed *reasonable encoding scheme* (in the sense of [3], page 10) which provides a string associated with each problem instance. The *size* of an instance $u \in I_X$ is the length of the string associated with it by means of a reasonable encoding scheme.

A correspondence between decision problems and languages over a finite alphabet, 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 Σ, its associated decision problem is $X_L = (I_{X_L}, \theta_{X_L})$, where $I_{X_L} = \Sigma^*$, and $\theta_{X_L} = \{(x, 1) \mid x \in L\} \cup \{(x, 0) \mid x \notin L\}$. The solvability of decision problems is defined through the recognition of the languages associated with them by means of language recognizer devices.

Definition 4. *A* tissue P system *of degree* $q \geq 1$ *is a recognizer* system *if:*

1. *The working alphabet Γ has two distinguished objects* yes *and* no *being, at least, one copy of them present in some initial multisets, but none of them are present in the alphabet of the environment.*
2. *All computations halt.*
3. *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.*

Note that, because of the first condition, the presence or absence of objects yes and no in the environment can be accounted for in any configuration. Note also that all computations are finite as a consequence of the second condition, and thus it is possible to refer to their "last step".

Given a recognizer tissue P system Π and a computation $\mathcal{C}$ of Π, we say that $\mathcal{C}$ is an *accepting computation* (respectively, *rejecting computation*) if object yes (respectively, object no) appears in the environment associated with the corresponding halting configuration of $\mathcal{C}$. Note that, since Π is a recognizer system, neither object yes nor no appears in the environment associated with any non–halting configuration of $\mathcal{C}$.

For each natural number $k \geq 1$, we denote by $\mathbf{TDC}(k)$ (respectively, $\mathbf{TSC}(k)$) the class of recognizer tissue P systems with cell division (respectively, with cell separation) and communication rules with length at most k. We denote by $\widehat{\mathbf{TDC}}(k)$ (respectively, $\widehat{\mathbf{TSC}}(k)$) the class of recognizer tissue P systems with cell division (respectively, with cell separation), with communication rules with length at most k, and without environment.

Now, we define what it means to solve a decision problem in the framework of tissue P systems efficiently and in a uniform way. Since we define each tissue P system to work on a finite number of inputs, to solve a decision problem we define a numerable family of tissue P systems.

Definition 5. *We say that a decision problem $X = (I_X, \theta_X)$ is solvable in a uniform way and polynomial time by a family $\mathbf{\Pi} = \{\Pi(n) \mid n \in \mathbb{N}\}$ of recognizer P systems if the following holds:*

1. *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 $n \in \mathbb{N}$.*
2. *There exists a pair (cod, s) of polynomial-time computable functions over I_X such that:*
   (a) *for each instance $u \in I_X$, $s(u)$ is a natural number and $cod(u)$ is an input multiset of the system $\Pi(s(u))$;*
   (b) *for each $n \in \mathbb{N}$, $s^{-1}(n)$ is a finite set;*
   (c) *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 $cod(u)$ is halting and it performs at most $p(|u|)$ steps;*
   (d) *the family $\mathbf{\Pi}$ is* sound *with regard to (X, cod, s), that is, for each $u \in I_X$, if* <u>there exists</u> *an accepting computation of $\Pi(s(u))$ with input $cod(u)$, then $\theta_X(u) = 1$;*
   (e) *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* <u>every</u> *computation of $\Pi(s(u))$ with input $cod(u)$ is an accepting one.*

From the soundness and completeness conditions above 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 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 [16].

3 Computational Efficiency of Tissue P Systems without Environment

It is well known that tissue P systems with cell division and tissue P systems with cell separation are able to solve computationally hard problems efficiently.

Specifically, **NP**–complete problems have been solved in polynomial time in [19] by using families of tissue P systems with cell division and communication rules of length at most 2, and by using families of tissue P systems with cell separation and communication rules of length at most 3. Thus,

$$\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC_{TDC(2)}} \cap \mathbf{PMC_{TSC(3)}}$$

In [4,9,10] it has been proved that only tractable problems can be efficiently solved by using families of tissue P systems with cell division and communication rules of length 1 (or with cell separation and communication rules of length bounded by 2). That is, $\mathbf{P} = \mathbf{PMC_{TDC(1)}} = \mathbf{PMC_{TSC(1)}} = \mathbf{PMC_{TSC(2)}}$. Therefore, in the framework of tissue P systems with cell division (respectively, cell separation), passing the maximum length of communication rules of the systems from 1 to 2 (respectively, from 2 to 3) amounts to passing from non–efficiency to efficiency, assuming that $\mathbf{P} \neq \mathbf{NP}$. That is, the cooperation of 2 objects (respectively, 3 objects) in the communication rules is a key feature that allows efficient solutions of **NP**–complete problems.

3.1 Efficiency of Tissue P Systems with Cell Division and without Environment

In this section, we give a family of tissue P systems with cell division, communication rules of length at most 2, and without environment which solves the `HAM-CYCLE` problem, a well known **NP**–complete problem [3], in polynomial time, according to Definition 5.

Let us recall that the `HAM-CYCLE` problem is the following: *given a directed graph, to determine whether or not there exists a Hamiltonian cycle in the graph.*

Our starting point will be the family $\mathbf{\Pi} = \{\Pi(n) \mid n \in \mathbb{N}\}$ of tissue P systems from **TDC**(2) provided in [19]. We will not recall in detail the definition of this solution, but let us provide an informal overview of the design. The authors follow a brute force approach, generating all possible combination of arcs from the graph, and then checking whether they represent a Hamiltonian cycle or not. Let us consider an arbitrary instance $G = (V, E)$ of the `HAM-CYCLE` problem, where $|V| = n$. In order to represent the generated paths, there are n special obects $(u, v)_1, \ldots (u, v)_n$ in the input multiset of the system for each arc $(u, v) \in E$. Having the object $(u, v)_i$ in the multiset of a cell after the generation stage is completed will mean "the arc (u, v) is the $i-th$ component of the path associated with this cell". All possible subsets of the input multiset are generated in the first stage of the computation, and then there is a checking stage that filters all invalid paths, as well as those which are not Hamiltonian cycles (a collection of auxiliary cells and symbols are used, but we will skip the details here). Finally, the computation ends with a final stage that sends the appropriate answer to the environment, depending on the results of all those checkings.

The idea of the solution presented here is the following: starting from the above mentioned family $\mathbf{\Pi}$, we construct a family $\mathbf{\Pi'} = \{\Pi'(n) \mid n \in \mathbb{N}\}$ of tissue P systems from $\widehat{\mathbf{TDC}}(2)$ such that $\Pi'(n)$ processes all instances G of `HAM-CYCLE`

with n nodes. The construction is implemented according to Definition 6.2 in [15], in such a way that each $\Pi'(n)$ *simulates* its counterpart $\Pi(n)$ in an efficient way. We refer to [15] for details, but informally speaking, each computation from $\Pi'(n)$ matches (or "simulates") an equivalent one from $\Pi(n)$, except for a polynomial amount of additional auxiliary steps.

Let us recall that for each $n \in \mathbb{N}$, $\Pi(n)$ is the following tissue P system:

$$\Pi(n) = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_{in}, \mathcal{M}_h, \mathcal{M}_y, \mathcal{M}_{yes}, \mathcal{M}_{no}, \mathcal{M}_{out},$$
$$\mathcal{M}_{e_{i,j,k}}(1 \le i, j, k \le n), \mathcal{M}_{c_i}(1 \le i \le n), \mathcal{R}, i_{in}, i_{out})$$

- The input alphabet is $\Sigma = \{(i, j)_k \mid 1 \le i, j, k \le n\}$.
- The working alphabet is

$$\Gamma = \{(i, j)_k, (i, j)'_k, (i, j)''_k \mid 1 \le i, j, k \le n\} \cup$$
$$\{(i, j)_{k,r}, (i, j)'_{k,r}, (i, j)''_{k,r} \mid 1 \le i, j, k \le n \ \wedge \ 1 \le r \le n^3\} \cup$$
$$\{w_i \mid 1 \le i \le n^3 + 6\} \cup \{c_r, h_r, y_r \mid 1 \le r \le n^3\} \cup$$
$$\{w, c, c', c'', h, h', h'', h''', y, y', y'', y''', y'''', x, yes, no, \#\}$$

- The alphabet of the environment is

$$\mathcal{E} = \{w_i \mid 1 \le i \le n^3 + 5\} \ \cup \ \{w, c'', y'', h'', y''', h''', y''''\}$$

- The initial multisets are

$$\begin{cases} \mathcal{M}_{in} = c^n \, y \, h \\ \mathcal{M}_{e_{i,j,k}} = (i, j)''_{k,n^3}, \ 1 \le i, j, k \le n \\ \mathcal{M}_{c_i} = c_{n^3}, \ 1 \le i \le n \\ \mathcal{M}_h = h_{n^3} \\ \mathcal{M}_y = y_{n^3} \\ \mathcal{M}_{yes} = yes \\ \mathcal{M}_{no} = w_{n^3+6} \, no \\ \mathcal{M}_{out} = x \end{cases}$$

- The set $\mathcal{R}$ consists of the following rules:

  (1) $(no, w_r / w_{r-1}, 0)$, for $2 \le r \le n^3 + 6$.
  (2) $(no, w_1 / w, 0)$.
  (3) $[(i, j)_k]_{in} \to [(i, j)'_k]_{in} [\#]_{in}$, for $1 \le i, j, k \le n$.
  (4) $[(i, j)''_{k,r}]_{e_{i,j,k}} \to [(i, j)''_{k,r-1}]_{e_{i,j,k}} [(i, j)''_{k,r-1}]_{e_{i,j,k}}$,
      for $1 \le i, j, k \le n$ and $2 \le r \le n^3$.
  (5) $[(i, j)''_{k,1}]_{e_{i,j,k}} \to [(i, j)''_k]_{e_{i,j,k}} [(i, j)''_k]_{e_{i,j,k}}$, for $1 \le i, j, k \le n$.
  (6) $[c_r]_{c_i} \to [c_{r-1}]_{c_i} [c_{r-1}]_{c_i}$, for $1 \le i \le n \ \wedge \ 1 \le r \le n^3$.
  (7) $[y_r]_y \to [y_{r-1}]_y [y_{r-1}]_y$, for $1 \le r \le n^3$.
  (8) $[h_r]_h \to [h_{r-1}]_h [a_{r-1}]_h$, for $1 \le r \le n^3$.
  (9) $(in, (i, j)'_k / (i, j)''_k, e_{i,j,k})$, for $1 \le i, j, k \le n$.
  (10) $(in, c / c', c_i)$, for $1 \le i \le n$.
  (11) $(in, y / y', y)$.
  (12) $(in, h / h', h)$.

(13) $(in, (i,j)''_k (i,j')''_{k'} / \lambda, 0)$, for $1 \le i, j, j', k, k' \le n$.

(14) $(in, (i,j)''_k (i',j)''_{k'} / \lambda, 0)$, for $1 \le i, i', j, k, k' \le n$.

(15) $(in, (i,j)''_k (i',j')''_{k+1} / \lambda, 0)$, for $1 \le i, i', j, j', k \le n$, and $j \ne i'$.

(16) $(in, (i,j)''_k (i',j')''_k / \lambda, 0)$, for $1 \le i, i', j, j', k \le n$.

(17) $(in, c' / c'', 0)$.

(18) $(in, y' / y'', 0)$.

(19) $(in, h' / h'', 0)$.

(20) $(in, (i,j)''_k c'' / \lambda, 0)$ for $1 \le i, j, k \le n$.

(21) $(in, y'' / y''', 0)$.

(22) $(in, h'' / h''', 0)$.

(23) $(in, c'' h''' / \lambda, 0)$.

(24) $(in, y''' / y'''', 0)$.

(25) $(in, h''' y'''' / \lambda, yes)$.

(26) $(yes, y'''' yes / \lambda, out)$.

(27) $(out, x\, yes / \lambda, 0)$.

(28) $(no, w\, no / \lambda, out)$.

(29) $(out, x\, no / \lambda, 0)$.

- The input cell is $i_{in} = in$.
- The output region is the environment, $i_{out} = 0$.

Let us notice that $|\Gamma| = 3n^4 + 7n^3 + 23$, $|\mathcal{E}| = n^3 + 12$ and the degree of $\Pi(n)$ is $q = n^3 + n + 6$. Let Lab_n denote the set of labels of cells in $\Pi(n)$. Besides, the execution-time is $n^3 + 7$ if the answer is affirmative and it is $n^3 + 8$ if the answer is negative. We thus consider $p(n) = n^3 + 8$ as the polynomial function needed for the construction of $\Pi'(n)$, according to Definition 6.2 in [15].

Now, for each $n \in \mathbb{N}$, let us construct, using $\Pi(n)$ as a starting point, a tissue P system from $\widehat{\mathbf{TDC}}(2)$ of degree $q_1 = 1 + (n^3 + n + 6) \cdot (n^3 + 10) + (n^3 + 12)$,

$$\Pi'(n) = (\Gamma', \Sigma', \mathcal{E}', \mathcal{M}'_0, \mathcal{M}'_1, \ldots, \mathcal{M}'_{q_1-1}, \mathcal{R}', i'_{in}, i'_{out})$$

defined as follows:

- $\Gamma' = \Gamma \cup \{\alpha_j \mid 0 \le j \le n^3 + 7\}$.
- $\Sigma' = \Sigma$ and $\mathcal{E}' = \emptyset$.
- Each one of the q cells of $\Pi(n)$ provides a cell of $\Pi'(n)$ with the same label. In addition, $\Pi'(n)$ has:
  - For each one of the q cells of $\Pi(n)$, $n^3 + 9$ new cells, labelled by $(i,0), \ldots, (i, n^3+8)$, respectively, where i stands for the original label of the cell in $\Pi(n)$.
  - A distinguished cell labelled by 0.
  - A new cell, labelled by l_b, for each $b \in \mathcal{E}$.
- $\mathcal{M}'_{l_b} = \{\alpha_0\}$, for each $b \in \mathcal{E}$, $\mathcal{M}'_{(i,0)} = \mathcal{M}_i$, for each $i \in Lab_n$, and every other multiset of $\Pi'(n)$ is initially empty.

- $\mathcal{R}' = \mathcal{R} \cup \{[\alpha_j]_{l_b} \to [\alpha_{j+1}]_{l_b} \ [\alpha_{j+1}]_{l_b} \mid b \in \mathcal{E} \ \wedge \ 0 \le j \le n^3 + 6\}$
  $\cup \{[\alpha_{n^3+7}]_{l_b} \to [b]_{l_b} \ [b]_{l_b} \mid b \in \mathcal{E}\}$
  $\cup \{(l_b, \ b/\lambda \ , 0) \mid b \in \mathcal{E}\}$
  $\cup \{((i,j), \ a/\lambda \ , (i, j+1)) \mid a \in \Gamma \ \wedge \ i \in Lab_n \ \wedge \ 0 \le j \le n^3 + 7\}$
  $\cup \{((i, n^3 + 8), \ a/\lambda \ , i) \mid a \in \Gamma \ \wedge \ i \in Lab_n\}$
- $i'_{in} = (i_{in}, 0)$, and $i'_{out} = 0$.

Let us notice that $\Pi'(n)$ 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 $\Pi'(n)$.
- ★ There is a distinguished cell in $\Pi'(n)$ 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 $\Pi'(n)$.

Note also that this construction does not affect the maximum length of the communication rules, since the communication rules in $\mathcal{R}' \backslash \mathcal{R}$ are of type symport and length 1.

An Overview of the Computations

Let $G = (V, E)$, with $V = \{1, \dots, n\}$ and $E = \{(u_1, v_1), \dots, (u_p, v_p)\}$, be an arbitrary instance of the HAM-CYCLE problem.

The *size* mapping on the set of instances is defined as $s(G) = n$, and the encoding of the instance is the multiset

$$cod(G) = \{(u_i, v_i)_k \mid 1 \le i \le p \ \wedge \ 1 \le k \le n \ \wedge \ (u_i, v_i) \in E\}$$

Each object $(u_i, v_i)_k$ can be interpreted as considering arc (u_i, v_i) being "placed" in the "k-th position" in a sequence of n arcs that could be a Hamiltonian cycle.

This way of encoding arcs by means of objects is one of the keys to understand the design of the solution. A brute force approach is followed, generating all possible combinations by division and subsequently checking for each subset of n objects from $cod(G)$ whether it represents a Hamiltonian cycle or not.

Let us now informally describe how system $\Pi'(s(G))$ with input multiset $cod(G)$, denoted by $\Pi'(s(G)) + cod(G)$, works, in order to process the instance G of the HAM-CYCLE problem.

At the initial configuration of $\Pi'(s(G)) + cod(G)$ we have the following:

- Cell labelled by 0 is empty.
- For each $i \in Lab_n$, the contents of cell i is empty and the contents of cell $(i, 0)$ is $\mathcal{M}_i$ (except for the case $i = i_{in}$, where $\mathcal{M}'_{(in,0)} = \mathcal{M}_{in} + cod(G)$).
- For each i, j ($i \in Lab_n$ and $1 \le j \le n^3 + 8$), the contents of cell (i, j) is empty.
- For each $b \in \mathcal{E}$, cell labelled by l_b contains only object α_0.

It is easy to check that the rules of a system $\Pi(n)$ of the family are recursively defined from n and the amount of resources needed to build an element of the

family is of a polynomial order in n. Therefore, there exists a deterministic Turing machine that builds the system $\Pi(n)$ in time polynomial with respect to n. The same holds for $\Pi'(n)$, since only a polynomial number of cells, objects and rules have been added to the definition.

At the first $n^3 + 9$ steps of any computation $\mathcal{C}'$ of $\Pi'(n)$, only the following rules can be applied:

- $\{[\alpha_j]_{l_b} \to [\alpha_{j+1}]_{l_b} [\alpha_{j+1}]_{l_b} \mid b \in \mathcal{E} \ \wedge \ 0 \le j \le n^3 + 6\}$
- $\{[\alpha_{n^3+7}]_{l_b} \to [b]_{l_b} [b]_{l_b} \mid b \in \mathcal{E}\}$
- $\{(l_b, \ b/\lambda \ , 0) \mid b \in \mathcal{E}\}$
- $\{((i,j), \ a/\lambda \ , (i, j+1)) \mid a \in \Gamma \ \wedge \ i \in Lab_n \ \wedge \ 0 \le j \le n^3 + 7\}$
- $\{((i, n^3 + 8), \ a/\lambda \ , i) \mid a \in \Gamma \ \wedge \ i \in Lab_n\}$

The purpose of the division rules is to generate an exponential amount of copies of each element of the environment alphabet. After the division process is completed, all copies of these objects are transferred to cell 0 by symport rules. In the meantime, the rest of the objects initially present in the system are "delayed", by being forced to travel through a sequence of auxiliary cells. More precisely, the initial multiset of cell i starts from cell $(i, 0)$, then goes through every intermediate cell (i, j) until reaching cell $(i, n^3 + 8)$. After that, the multiset can finally be transferred to cell i.

Besides, the above mentioned rules are applied in a deterministic manner. Then, the configuration $\mathcal{C}'_{n^3+9}$ of any computation $\mathcal{C}'$ of $\Pi'(s(G)) + cod(G)$ is characterized by the following:

   (1) The contents of cell 0 is $b_1^{2^{n^3+8}} \dots b_\alpha^{2^{n^3+8}}$, where $\mathcal{E} = \{b_1, \dots, b_\alpha\}$.
   (2) For each $i \in Lab_n$, the contents of cell i is $\mathcal{M}_i$ (except for the case $i = i_{in}$, that contains $\mathcal{M}_{in} + cod(G)$).
   (3) For i, j ($i \in Lab_n$ and $0 \le j \le n^3 + 8$) the contents of cell (i, j) is empty.
   (4) For each $b \in \mathcal{E}$, there exist 2^{n^3+8} cells labelled by l_b whose content is empty.

Basically, this is the "initial" configuration of the system $\Pi(s(G)) + cod(G)$, with a standard cell labelled by 0 that will play the role of the environment, and with a large number of spare empty cells. Therefore, from step $n^3 + 9$ any computation of $\Pi'(s(G)) + cod(G)$ "reproduces" a computation of the system $\Pi(s(G)) + cod(G)$ with a delay.

Bearing in mind that the family $\mathbf{\Pi} = \{\Pi(n) \mid n \in \mathbb{N}\}$ solves HAM-CYCLE problem in polynomial time, we deduce that the family $\mathbf{\Pi}' = \{\Pi'(n) \mid n \in \mathbb{N}\}$ also solves HAM-CYCLE problem in polynomial time. Hence, we have the following result:

Theorem 1. HAM-CYCLE $\in \mathbf{PMC}_{\widehat{\mathbf{TDC}(2)}}$.

That is, a uniform solution working in polynomial time has been found for an **NP**–complete problem using an empty environment alphabet. Hence, the environment does not play a relevant role in recognizer tissue P systems with cell division with respect to the efficiency of these models.

3.2 Non-efficiency of Tissue P Systems with Cell Separation and without Environment

In [6] it has been proved that only tractable problems can be efficiently solved by using tissue P systems with cell separation where there is no environment having infinitely many copies of some objects. Thus, tissue P systems with cell separation and without environment are non-efficient in the sense that they are not capable to solve **NP**–complete problems in polynomial time, according to Definition 5, assuming that $\mathbf{P} \neq \mathbf{NP}$.

Theorem 2. *For each $k \in \mathbb{N}, k \geq 1$ we have $\mathbf{P} = \mathbf{PMC}_{\widehat{\mathbf{TSC}}(k)}$.*

Hence, the environment plays a relevant role in recognizer tissue P systems with cell separation with respect to the efficiency of these models. That is, by using the environment, **NP**–complete problems can be solved in polynomial time, but this is not possible when the initial environment is empty.

Another interesting consequence of the previous result is the following. In the framework of recognizer tissue P systems without environment, the kind of rules provides a frontier for the efficiency, that is, passing from division rules to separation rules amounts to passing from efficiency to non-efficiency, assuming that $\mathbf{P} \neq \mathbf{NP}$.

4 Conclusions

In this paper we have discussed how allowing an infinite supply of objects in the environment determines (or not) that the model of tissue P systems considered will be efficient or not.

More precisely, we have highlighted the key role that the environment plays in the case of tissue P systems with cell separation. It does actually constitute a borderline between efficiency and non-efficiency for the classes $\mathbf{TSC}(k)$ and $\widehat{\mathbf{TSC}}(k)$, for every $k \geq 3$. However, it is important to note that cooperation (of at least 3 objects) in the communication rules is another important ingredient, since we cannot get efficient solutions with tissue P systems with cell separation and communication rules of length bounded by 2, irrespectively of using the environment or not [10].

On the other hand, the environment has been shown to be an irrelevant ingredient in the case of tissue P systems with cell division. Indeed, a uniform polynomial solution has been described for HAM-CYCLE using a family of tissue P systems with cell division and without environment from $\widehat{\mathbf{TDC}}(2)$. Note that the borderline of efficiency concerning the length of communication rules remains the same as what was already known when the environment is exploited: symport of length 1 versus cooperation of 2 objects.

References

1. Alhazov, A., Ishdorj, T.O.: Membrane operations in P systems with active membranes. In: Păun, G., et al. (eds.) Proceedings of the Second Brainstorming Week on Membrane Computing, Seville, Spain, February 2-7, Technical Report 01/2004, University of Seville, pp. 37-44 (2004)
2. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Pérez-Jiménez, M.J., Riscos-Núñez, A., Romero–Campero, F.J.: Computational efficiency of cellular division in tissue-like P systems. Romanian Journal of Information Science and Technology 11(3), 229–241 (2008)
3. Garey, M.R., Johnson, D.S.: Computers and Intractability A Guide to the Theory of NP-Completeness. W.H. Freeman and Company (1979)
4. Gutiérrez–Escudero, R., Pérez–Jiménez, M.J., Rius–Font, M.: Characterizing tractability by tissue-like P systems. In: Păun, G., Pérez-Jiménez, M.J., Riscos-Núñez, A., Rozenberg, G., Salomaa, A. (eds.) WMC 2009. LNCS, vol. 5957, pp. 289–300. Springer, Heidelberg (2010)
5. Ito, M., Martín Vide, C., Păun, G.: A characterization of Parikh sets of ET0L laguages in terms of P systems. In: Ito, M., Păun, G., Yu, S. (eds.) Words, Semigroups and Transducers, pp. 239–254. World Scientific, Singapore (2001)
6. Macías-Ramos, L.F., Pérez-Jiménez, M.J., Riscos-Núñez, A., Rius-Font, M., Valencia-Cabrera, L.: The efficiency of tissue P systems with cell separation relies on the environment. In: Csuhaj-Varjú, E., Gheorghe, M., Rozenberg, G., Salomaa, A., Vaszil, G. (eds.) CMC 2012. LNCS, vol. 7762, pp. 243–256. Springer, Heidelberg (2013)
7. Martín-Vide, C., Pazos, J., Păun, G., Rodríguez-Patón, A.: A New Class of Symbolic Abstract Neural Nets: Tissue P Systems. In: Ibarra, O.H., Zhang, L. (eds.) COCOON 2002. LNCS, vol. 2387, pp. 290–299. Springer, Heidelberg (2002)
8. Pan, L., Ishdorj, T.-O.: P systems with active membranes and separation rules. Journal of Universal Computer Science 10(5), 630–649 (2004)
9. Pan, L., Pérez-Jiménez, M.J.: Computational complexity of tissue–like P systems. Journal of Complexity 26(3), 296–315 (2010)
10. Pan, L., Pérez-Jiménez, M.J., Riscos-Núñez, A., Rius-Font, M.: New frontiers of the efficiency in tissue P systems. In: Pan, L., Păun, G., Song, T. (eds.) Pre-proceedings of Asian Conference on Membrane Computing, October 15-18, pp. 61–73. Huazhong University of Science and Technology, Wuhan (2012)
11. Păun, A., Păun, G.: The power of communication: P systems with symport/antiport. New Generation Computing 20(3), 295–305 (2002)
12. Păun, G.: Attacking **NP**-complete problems. In: Antoniou, I., Calude, C., Dinneen, M.J. (eds.) Unconventional Models of Computation, UMC'2K, pp. 94–115. Springer (2000)
13. Păun, G.: Membrane Computing. An Introduction. Springer, Berlin (2002)
14. Păun, G., Pérez-Jiménez, M.J., Riscos-Núñez, A.: Tissue P systems with cell division. Int. J. of Computers, Communications and Control 3(3), 295–303 (2008)
15. Pérez-Jiménez, M.J., Riscos-Núñez, A., Rius-Font, M., Romero-Campero, F.J.: The role of the environment in tissue P systems with cell division. In: García-Quismondo, M., et al. (eds.) Proceedings of the Tenth Brainstorming Week on Membrane Computing, Seville, Spain, January 30-February 3, Report RGNC 02/2012, Fénix Editora, vol. II, pp. 89–104 (2012); A revised and improved version can be found in: Pérez-Jiménez, M.J., Riscos-Núñez, A., Rius-Font, M., Romero-Campero, F.J.: A polynomial alternative to unbounded environment for tissue P systems with cell division. Int. J. Comput. Math. 90(4), 760–775 (2013)

16. Pérez-Jiménez, M.J., Romero-Jiménez, A., Sancho-Caparrini, F.: Complexity classes in models of cellular computing with membranes. Natural Computing 2(3), 265–285 (2003)
17. Pérez-Jiménez, M.J., Romero-Jiménez, A., Sancho-Caparrini, F.: A polynomial complexity class in P systems using membrane division. Journal of Automata, Languages and Combinatorics 11(4), 423–434 (2006)
18. Pérez-Jiménez, M.J., Sosík, P.: Improving the efficiency of tissue P systems with cell separation. In: García-Quismondo, M., et al. (eds.) Proceedings of the Tenth Brainstorming Week on Membrane Computing, Seville, Spain, January 30-February 3, Report RGNC 02/2012, Fénix Editora, vol. II, pp. 105–140 (2012)
19. Porreca, A.E., Murphy, N., Pérez-Jiménez, M.J.: An optimal frontier of the efficiency of tissue P systems with cell division. In: García-Quismondo, M., et al. (eds.) Proceedings of the Tenth Brainstorming Week on Membrane Computing, Seville, Spain, January 30-February 3, Report RGNC 02/2012, Fénix Editora, vol. II, pp. 141–166 (2012)