Characterizing Tractability by Tissue-Like P Systems

Rosa Gutiérrez–Escudero[1], Mario J. Pérez–Jiménez[1], Miquel Rius–Font[2]

[1] Research Group on Natural Computing
   Department of Computer Science and Artificial Intelligence
   University of Sevilla
   Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
   `rgutierrez,marper@us.es`
[2] Department of Applied Mathematics IV
   Universitat Politécnica de Catalunya
   Edifici C3, Despatx 016, Av. del Canal Olímpic, s/n
   08860 Castelldefels, Spain
   `mrius@ma4.upc.edu`

Summary. In the framework of cell–like membrane systems it is well known that the construction of exponential number of objects in polynomial time is not enough to efficiently solve **NP**–complete problems. Nonetheless, it may be sufficient to create an exponential number of membranes in polynomial time. In the framework of recognizer polarizationless P systems with active membranes, the construction of an exponential workspace expressed in terms of number of membranes and objects may not suffice to efficiently solve computationally hard problems.

In this paper we study the computational efficiency of recognizer tissue P systems with communication (symport/antiport) rules and division rules. Some results have been already obtained in this direction: (a) using communication rules and forbidding division rules, only tractable problems can be efficiently solved; (b) using communication rules with length three and division rules, **NP**–complete problems can be efficiently solved. In this paper we show that the allowed length of communication rules plays a relevant role from the efficiency point of view of the systems.

1 Introduction

Membrane Computing is a branch of Natural Computing and starts from the assumption that the processes taking place within the compartmental structure of a living cell can be interpreted as computations [9]. The computational devices in Membrane Computing are called *P systems*. Roughly speaking, a P system consists of a membrane structure. In the compartments of this structure are multisets of

objects which evolve according to given rules in a synchronous, non–deterministic, maximally parallel manner[3].

In recent years, many different models of P systems have been proposed and proved to be computationally universal. The most studied variants are characterized by a *cell-like* membrane structure, where the communication happens between a membrane and the surrounding one. In this model the membrane structure is hierarchical and the graph of the neighborhood relation between compartments is a tree.

We shall focus here on another type of P systems, the so-called (because of their membrane structure) *tissue P Systems*. Instead of considering a hierarchical arrangement, membranes are modeled as nodes of an undirected graph. This variant has two biological inspirations: intercellular communication and cooperation between neurons. The common mathematical model of these two mechanisms is a net of processors dealing with symbols and communicating these symbols along channels specified in advance. The communication between cells is based on symport/antiport rules[4]. Symport rules move a number of objects across a membrane together in the same direction, whereas antiport rules move objects across a membrane in opposite directions.

Since the initial definition of tissue P systems several research lines have been developed and other variants have arisen. One of the most interesting variants of tissue P systems was presented in [12] where the definition of tissue P systems is combined with the corresponding one of P systems with active membranes, yielding the model of *tissue P systems with cell division.*

This model has been studied in depth in [1], where the importance of the cell division rules regarding the computational power of the model is shown. Working with tissue P systems without division rules it is not possible to solve computationally hard problems [2] (unless **P=NP**). We focus now on the influence of the length of communication rules on the computational power of tissue P systems with cell division. In particular, when limiting this length to 1, only tractable problems can be efficiently solved. A proof of this result is presented here.

The paper is organized as follows. In Section 2 we recall some definitions related to tissue P systems (further information can be found in the literature, see [15]). Section 3 is devoted to formalizing the concept of polynomial solvability of decision problems by recognizer tissue P systems. In Section 4 we introduce a dependency graph for tissue P systems and use this technique to prove the main result of the paper. Finally, the last section contains some remarks and raises open questions and future work directions.

---

[3] An informal overview can be found in [11] and further bibliography at [15].

[4] This method of communication for P systems was introduced in [8].

2 Recognizer Tissue P Systems

Firstly, the concept of *tissue P system of degree $q \geq 1$ with cell division* is introduced.

Definition 1. *A* tissue P system of degree $q \geq 1$ with cell division *is a tuple*

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

where:

1. *Γ is a finite alphabet (called working alphabet) whose elements are called objects;*
2. *Σ is a finite alphabet (called input alphabet) strictly contained in Γ ;*
3. *$\Omega \subseteq \Gamma \setminus \Sigma$ is a finite alphabet, describing the set of objects present in the environment in arbitrarily many copies each;*
4. *$\mathcal{M}_1, \ldots, \mathcal{M}_q$ are strings over Γ, describing the multisets of objects placed in the q cells of the system;*
5. *R is a finite set of rules, of the following forms:*
    a) *$(i, u/v, j)$, for $i, j \in \{0, 1, 2, \ldots, q\}, i \neq j$, and $u, v \in \Gamma^*$; communication rules; $1, 2, \ldots, q$ identify the cells of the system, 0 is the environment; 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 the multiset v are sent from region j to region i (we say that the sum of the lengths of u and v is the length of the rule);*
    b) *$[\, a \,]_i \rightarrow [\, b \,]_i [\, c \,]_i$, where $i \in \{1, 2, \ldots, q\}$ and $a, b, c \in \Gamma$; division rules; under the influence of object a, the cell with label i is divided in two cells with the same label; in the first copy the object a is replaced by b, in the second copy the object a is replaced by c; all other objects are replicated and copies of them are placed in the two new cells.*
6. *$i_{in} \in \{1, \ldots, q\}$ is the input cell, and $i_{out} \in \{0, 1, \ldots, q\}$ is the output cell.*

The rules of such a system are applied in a non-deterministic maximally parallel manner as is customary in membrane computing. In each step, all cells which can evolve must evolve in a maximally parallel way (in each step we apply a multiset of rules which is maximal, no further rule can be added), with the following important remark: if a cell divides, then the division rule is the only one which is applied for that cell in that step, its objects do not evolve by means of communication rules. 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 in the next step – providing that they do not divide once again. A cell's label precisely identifies the rules which can be applied to it.

A configuration of Π is a tuple $C = (M_0, M_1, \ldots, M_q)$, where M_0 is a multiset of objects over $\Gamma \setminus \Omega$ (the objects in the environment which are in finitely many copies), and $M_1, \ldots, M_q$ are multisets of objects over Γ (the objects in each cell of

the system). For two configurations C_1, C_2 of Π we write $C_1 \Rightarrow_\Pi C_2$, and we say that we have a *transition* from C_1 to C_2, if we can pass from C_1 to C_2 by applying the rules from $\mathcal{R}$.

The initial configuration of the system is $(\emptyset, \mathcal{M}_1, \ldots, \mathcal{M}_q)$. For each multiset m over the input alphabet, the initial configuration of the system associated with it is $(\emptyset, \mathcal{M}_1, \ldots, \mathcal{M}_{i_{in}} \cup m, \ldots, \mathcal{M}_q)$. Then, m is an *input multiset* of every computation $\mathcal{C} = \{C_i\}_{i<r}$ such that C_0 is the initial configuration of Π associated with m.

All computations start from an initial configuration and proceeds as stated above; only halting computations give a result, which is encoded by the number of objects in the output cell i_{out} in the last configuration. From now on, we will consider that the output is collected in the environment (that is, $i_{out} = 0$, and thus we will omit i_{out} in the definition of tissue P systems). This way, if Π is a tissue P system and $\mathcal{C} = \{C_i\}_{i<r}$ is a halting computation of Π, with $C_i = (M_{i,0}, M_{i,1}, \ldots, M_{i,q})$, then the answer of the computation $\mathcal{C}$ is

$$Output(\mathcal{C}) = \Psi_{\Gamma \setminus \Omega}(M_{r-1,0})$$

where Ψ is the Parikh function.

Let us recall that **NP**–completeness has been usually studied in the framework of *decision problems*, that is problems whose solution is either *yes* or *no*. More formally, 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.

Each decision problem $X = (I_X, \theta_X)$ has a language L_X over the alphabet of I_X associated with it, defined as follows: $L_X = \{a \in I_X \mid \theta_X(a) = 1\}$. Reciprocally, each language L over an alphabet Σ has a decision problem, X_L associated with it as follows: $I_{X_L} = \Sigma^*$, and $\theta_{X_L} = \{(x, 1) \mid x \in L\} \cup \{(x, 0) \mid x \notin L\}$.

Recognizer cell-like P systems were introduced in [14] and they are the natural framework to study and solve decision problems within Membrane Computing, since deciding whether an instance of a given problem has an affirmative or negative answer is equivalent to deciding if a string belongs or not to the language associated with the problem.

In the literature, recognizer cell-like P systems are associated with P systems with *input* in a natural way. The data encoding an instance of the decision problem has to be provided to the P system in order to compute the appropriate answer. This is done by codifying each instance as a multiset placed in an *input membrane*. The output of the computation (`yes` or `no`) is sent to the environment in the last step of the computation. In this way, cell-like P systems with input and external output are devices which can be seen as black boxes, in the sense that the user provides the data before the computation starts, and then waits *outside* the P system until it sends to the environment the output in the last step of the computation.

In order to use these computational devices for solving decision problems, *recognizer tissue P systems* are introduced.

Definition 2. *A tissue P system with cell division of degree $q \geq 1$*

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

is a recognizer system if the following holds:

1. *The working alphabet Γ has two distinguished objects* yes *and* no*, present in at least one copy in some initial multisets $\mathcal{M}_1, \ldots, \mathcal{M}_q$, but not present in Ω.*
2. *All computations halt.*
3. *If $\mathcal{C} = \{C_i\}_{i<r}$ is a computation of Π, then either the object* yes *or the object* no *(but not both) must have been released into the environment, and only in the last step of the computation.*

Given a recognizer tissue P system with cell division, and a computation $\mathcal{C} = \{C_i\}_{i<r}$ of Π ($r \in \mathbf{N}$), we define the result of $\mathcal{C}$ as follows:

$$Output(\mathcal{C}) = \begin{cases} \text{yes, if } \Psi_{\{\text{yes,no}\}}(M_{r-1,0}) = (1,0) \\ \qquad \wedge\ \Psi_{\{\text{yes,no}\}}(M_{k,0}) \quad = (0,0) \text{ for } k = 0, \ldots, r-2 \\ \text{no,}\ \text{ if } \Psi_{\{\text{yes,no}\}}(M_{r-1,0}) = (0,1) \\ \qquad \wedge\ \Psi_{\{\text{yes,no}\}}(M_{k,0}) \quad = (0,0) \text{ for } k = 0, \ldots, r-2 \end{cases}$$

That is, $\mathcal{C}$ is an accepting computation (respectively, rejecting computation) if the object yes (respectively, no) appears in the environment (only) in the halting configuration of $\mathcal{C}$.

3 Polynomial Solvability by Recognizer Tissue P systems

In this section, the definition of polynomial (uniform) solvability of decision problems by a family of cell–like P systems is extended to solvability by a family of tissue P systems.

Definition 3. *We say that a decision problem $X = (I_X, \theta_X)$ is solvable in polynomial time by a family $\mathbf{\Pi} = \{\Pi(n) : n \in \mathbb{N}\}$ of recognizer tissue P systems with cell division if the following hold:*

- *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 (called a polynomial encoding of I_X in $\mathbf{\Pi}$) such that:*
  - *For each instance $u \in I_X$, $s(u)$ is a natural number and $cod(u)$ is an input multiset of the system $\Pi(s(u))$.*
  - *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, moreover, it performs at most $p(|u|)$ steps.*

– *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* $cod(u)$, *then* $\theta_X(u) = 1$.
– *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* $cod(u)$ *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.

We denote by $\mathbf{PMC}_{TD}$ the set of all decision problems which can be solved by means of recognizer tissue P systems with cell division in polynomial time. This class is closed under polynomial–time reduction and under complement (see [13] for a similar result for cell-like P systems). We also denote by $\mathbf{PMC}_{TD(k)}$ the set of all decision problems which can be solved by means of recognizer tissue P systems with cell division in polynomial time, by using communication rules whose length is, at most, k.

4 Dependency Graph Associated with Tissue P Systems

Let Π be a tissue P system with cell division and let all communication rules be of length 1. In this case, each rule of the system can be activated by a single object. Hence, there exists in a certain sense, a *dependency* between the object triggering the rule and the object or objects produced by its application. This dependency allows to adapt the ideas developed in [5] for cell-like P systems with active membranes to tissue P systems with cell division and communication rules of length 1.

We can consider a general pattern $(a, i) \rightarrow (b_1, j) \ldots (b_s, j)$ where $i, j \in \{0, 1, 2, \ldots, q\}, i \neq j$, and $a, b \in \Gamma$. Communication rules correspond to the case $s = 1$ and $b_1 = a$, and division rules correspond to the case $s = 2$ and $j = i \neq 0$. The above pattern can be interpreted as follows: from the object a in the cell (or in the environment) labeled with i we can *reach* the objects $b_1, \ldots, b_s$ in the cell (or in the environment) labeled with j.

Without loss of generality we can assume that all communication rules in the system obey the syntax $(i, a/\lambda, j)$, since every rule of the form $(j, \lambda/a, i)$ can be rewritten to follow the above syntax, with equivalent semantics.

Next, we formalize these ideas in the following definition.

Definition 4. *Let* $\Pi = (\Gamma, \Sigma, \Omega, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in})$ *be a tissue P system of degree* $q \geq 1$ *with cell division. Let* $H = \{0, 1, \ldots, q\}$. *The dependency graph associated with* Π *is the directed graph* $G_\Pi = (V_\Pi, E_\Pi)$ *defined as follows:*

$V_\Pi = \{(a, i) \in \Gamma \times H : \ \exists j \in H \ ((i, a/\lambda, j) \in R \ \lor \ (j, a/\lambda, i) \in R) \ \lor$

$$\exists b, c \in \Gamma \ ([a]_i \rightarrow [b]_i[c]_i \ \in R \ \lor \ [b]_i \rightarrow [a]_i[c]_i \ \in R)\},$$

$$E_\Pi = \{((a,i),(b,j)) : (a = b \ \wedge \ (i,a/\lambda, j) \in R) \ \vee$$

$$\exists c \in \Gamma \ ([a]_i \rightarrow [b]_i[c]_i \in R \wedge j = i)\}.$$

Proposition 1. *Let* $\Pi = (\Gamma, \Sigma, \Omega, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in})$ *be a tissue P system with cell division, in which the length of all communication rules is 1. Let* $H = \{0, 1, \ldots, q\}$. *There exists a deterministic Turing machine that constructs the dependency graph* G_Π *associated with* Π, *in polynomial time (that is, in a time bounded by a polynomial function depending on the total number of rules).*

Proof. A deterministic algorithm that, given a P system Π with the set R of rules, constructs the corresponding dependency graph, is the following:

```
Input: Π (with R as its set of rules)
V_Π ← ∅;  E_Π ← ∅
for each rule r ∈ R of Π do
   if  r = (i, a/λ, j) then
      V_Π ← V_Π ∪ {(a, i), (a, j)};  E_Π ← E_Π ∪ {((a, i), (a, j))}
   if  r = [a]_i → [b]_i[c]_i then
      V_Π ← V_Π ∪ {(a, i), (b, i), (c, i)};
      E_Π ← E_Π ∪ {((a, i), (b, i)), ((a, i), (c, i))}
```

The running time of this algorithm is bounded by $O(|R|)$. $\qquad\square$

Proposition 2. *Let* $\Pi = (\Gamma, \Sigma, \Omega, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in})$ *be a tissue P system with cell division, in which the length of all communication rules is 1. Let* $H = \{0, 1, \ldots, q\}$. *Let* Δ_Π *be defined as follows:*

$$\Delta_\Pi = \{(a,i) \in \Gamma \times H : \textit{there exists a path (within the dependency graph)}$$
$$\textit{from } (a,i) \textit{ to } (\texttt{yes}, 0)\}.$$

Then, there exists a Turing machine that constructs the set Δ_Π *in polynomial time (that is, in a time bounded by a polynomial function depending on the total number of rules).*

Proof. We can construct the set Δ_Π from Π as follows:

- We construct the dependency graph G_Π associated with Π.
- Then we consider the following algorithm:

```
Input: G_Π = (V_Π, E_Π)
   Δ_Π ← ∅
   for each (a, i) ∈ V_Π do
      if reachability (G_Π, (a, i), (yes, 0)) = yes then
         Δ_Π ← Δ_Π ∪ {(a, i)}
```

The running time of this algorithm is of order $O(|V_\Pi| \cdot |V_\Pi|^2)$, hence[5] it is of order $O(|\Gamma|^3 \cdot |H|^3)$.   □

Notation: Let $\Pi = (\Gamma, \Sigma, \Omega, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$ be a tissue P system with cell division. Let m be a multiset over Σ. Then we denote $\mathcal{M}_j^* = \{(a,j) : a \in \mathcal{M}_j\}$, for $1 \le j \le q$, and $m^* = \{(a, i_{in}) : a \in m\}$.

  Below we characterize accepting computations of a recognizer tissue P system with cell division and communication rules of length 1 by distinguished paths in the associated dependency graph.

Lemma 1. *Let* $\Pi = (\Gamma, \Sigma, \Omega, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in})$ *be a recognizer confluent tissue P system with cell division in which the length of all communication rules is 1. The following assertions are equivalent:*

(1) There exists an accepting computation of Π.
(2) There exists $(a_0, i_0) \in \bigcup_{j=1}^{q} \mathcal{M}_j^*$ and a path in the dependency graph associated with Π, from (a_0, i_0) to $(\texttt{yes}, 0)$.

Proof. $(1) \Rightarrow (2)$ First, we show that for each accepting computation $\mathcal{C}$ of Π there exists $(a_0, i_0) \in \bigcup_{j=1}^{q} \mathcal{M}_j^*$ and a path $\gamma_\mathcal{C}$ in the dependency graph associated with Π from (a_0, i_0) to $(\texttt{yes}, 0)$. By induction on the length n of $\mathcal{C}$.

  If $n = 1$, a single step is performed in $\mathcal{C}$ from C_0 to C_1. A rule of the form $(j, \texttt{yes}/\lambda, 0)$, with $a \in \Gamma, j \ne 0$, has been applied in that step. Then, $(\texttt{yes}, j) \in \mathcal{M}_j^*$, for some $j = 1, \ldots, q$. Hence, $((\texttt{yes}, j), (\texttt{yes}, 0))$ is a path in the dependency graph associated with Π.

  Let us suppose that the result holds for n. Let $\mathcal{C} = (C_0, C_1, \ldots, C_n, C_{n+1})$ be an accepting computation of Π. Then $\mathcal{C}' = (C_1, \ldots, C_n, C_{n+1})$ is an accepting computation of the system $\Pi' = (\Gamma, \Sigma, \Omega, \mathcal{M}_1', \ldots, \mathcal{M}_q', \mathcal{R}, i_{in})$, being $\mathcal{M}_j'$ the contents of cell j in configuration C_1, for $1 \le j \le q$. By induction hypothesis there exists an object b_0 in a cell i_0 from C_1, and a path in the dependency graph associated with Π' from (b_0, i_0) to $(\texttt{yes}, 0)$. If (b_0, i_0) is an element of configuration C_0 (that means that in the first step a division rule has been applied to cell i_0), then the result holds. Otherwise, there is an element (a_0, j_0) in C_0 producing (b_0, i_0). So, there exists a path $\gamma_\mathcal{C}$ in the dependency graph associated with Π from (a_0, j_0) to $(\texttt{yes}, 0)$.

---

[5] The Reachability Problem is the following: *given a (directed or undirected) graph, G, and two nodes a, b, determine whether or not the node b is reachable from a, that is, whether or not there exists a path in the graph from a to b.* It is easy to design an algorithm running in polynomial time solving this problem. For example, given a (directed or undirected) graph, G, and two nodes a, b, we consider a depth–first–search with source a, and we check if b is in the tree of the computation forest whose root is a. The total running time of this algorithm is $O(|V| + |E|)$, that is, in the worst case is quadratic in the number of nodes. Moreover, this algorithm needs to store a linear number of items (it can be proved that there exists another polynomial time algorithm which uses $O(\log^2(|V|))$ space).

$(2) \Rightarrow (1)$. Let us see that for each $(a_0, i_0) \in \bigcup_{j=1}^{q} \mathcal{M}_j^*$ and for each path in the dependency graph associated with Π from (a_0, i_0) to $(\text{yes}, 0)$, there exists an accepting computation of Π. By induction on the length n of the path.

If $n = 1$, we have a path $((a_0, i_0), (\text{yes}, 0))$. Then, $a_0 = \text{yes}$ and the computation $\mathcal{C} = (C_0, C_1)$ where the rule $(i_0, \text{yes}/\lambda, 0)$ belongs to a multiset of rules m_0 that produces configuration C_1 from C_0 is an accepting computation of Π.

Let us suppose that the result holds for n. Let
$$((a_0, i_0), (a_1, i_1), \ldots (a_n, i_n), (\text{yes}, 0))$$
be a path in the dependency graph of length $n + 1$. If $(a_0, i_0) = (a_1, i_1)$, then the result holds by induction hypothesis. Otherwise, let C_1 be the configuration of Π reached from C_0 by the application of a multiset of rules containing the rule that produces (a_1, i_1) from (a_0, i_0). Then $((a_1, i_1), \ldots (a_n, i_n), (\text{yes}, 0))$ is a path of length n in the dependency graph associated with the system
$$\Pi' = (\Gamma, \Sigma, \Omega, \mathcal{M}_1', \ldots, \mathcal{M}_q', \mathcal{R}, i_{in})$$
being $\mathcal{M}_j'$ the content of cell j in configuration C_1, for $1 \leq j \leq q$. By induction hypothesis, there exists an accepting computation $\mathcal{C}' = (C_1, \ldots, C_t)$ of Π'. Hence, $\mathcal{C} = (C_0, C_1, \ldots, C_t)$ is an accepting computation of Π. $\qquad\square$

Next, given a family $\mathbf{\Pi} = (\Pi(n))_{n \in \mathbf{N}}$ of recognizer tissue P system with cell division in which the length of all communication rules is 1, solving a decision problem, we will characterize the acceptance of an instance of the problem, w, using the set $\Delta_{\Pi(s(w))}$ associated with the system $\Pi(s(w))$, that processes the given instance w. More precisely, the instance is accepted by the system if and only if there is an object in the initial configuration of the system $\Pi(s(w))$ with input $cod(w)$ such that there exists a path in the associated dependency graph starting from that object and reaching the object yes in the environment.

Proposition 3. *Let $X = (I_X, \theta_X)$ be a decision problem. Let $\mathbf{\Pi} = (\Pi(n))_{n \in \mathbf{N}}$ be a family of recognizer tissue P system with cell division in which the length of all communication rules is 1 solving X, according to Definition 3. Let (cod, s) be the polynomial encoding associated with that solution. Then, for each instance w of the problem X the following assertions are equivalent:*

(a) $\theta_X(w) = 1$ (that is, the answer to the problem is yes for w).

(b) $\Delta_{\Pi(s(w))} \cap ((cod(w))^* \cup \bigcup_{j=1}^{p} \mathcal{M}_j^*) \neq \emptyset$, where $\mathcal{M}_1, \ldots, \mathcal{M}_p$ are the initial multisets of the system $\Pi(s(w))$.

Proof. Let $w \in I_X$. Then $\theta_X(w) = 1$ if and only if there exists an accepting computation of the system $\Pi(s(w))$ with input multiset $cod(w)$. From Lemma 1 this condition is equivalent to the following: in the initial configuration of $\Pi(s(w))$ with input multiset $cod(w)$ there exists at least one object $a \in \Gamma$ in a cell labeled with i such that in the dependency graph the node $(\text{yes}, 0)$ is reachable from (a, i).

Hence, $\theta_X(w) = 1$ if and only if $\Delta_{\Pi(s(w))} \cap \mathcal{M}_j^* \neq \emptyset$ for some $j \in \{1, \ldots, p\}$, or $\Delta_{\Pi(s(w))} \cap (cod(w))^* \neq \emptyset$. $\qquad\square$

Theorem 1. P = $PMC_{TD(1)}$

Proof. We have $\mathbf{P} \subseteq \mathbf{PMC}_{TD(1)}$ because the class $\mathbf{PMC}_{TD(1)}$ is closed under polynomial time reduction. Next, we show that $\mathbf{PMC}_{TD(1)} \subseteq \mathbf{P}$. Let $X \in \mathbf{PMC}_{TD(1)}$ and let $\mathbf{\Pi} = (\Pi(n))_{n \in \mathbf{N}}$ be a family of recognizer tissue P systems with cell division solving X, according to Definition 3. Let (cod, s) be the polynomial encoding associated with that solution.

We consider the following deterministic algorithm:

```
Input: An instance w of X
  - Construct the system Π(s(w)) with input multiset cod(w).
  - Construct the dependency graph G_Π(s(w)) associated with Π(s(w)).
  - Construct the set Δ_Π(s(w)) as indicated in Proposition 2
    answer ← no;  j ← 1
    while j ≤ p  ∧  answer = no do
        if  Δ_Π(s(w)) ∩ M*_j ≠ ∅ then
            answer ← yes
        j ← j + 1
    endwhile
    if  Δ_Π(s(w)) ∩ (cod(w))* ≠ ∅ then
        answer ← yes
```

On one hand, the answer of this algorithm is yes if and only if there exists a pair (a, i) belonging to $\Delta_{\Pi(s(w))}$ such that the symbol a appears in the cell labeled with i in the initial configuration (with input the multiset $cod(w)$).

On the other hand, a pair (a, i) belongs to $\Delta_{\Pi(s(w))}$ if and only if there exists a path from (a, i) to $(\mathbf{yes}, 0)$, that is, if and only if we can obtain an accepting computation of $\Pi(s(w))$ with input $cod(w)$. Hence, the algorithm above described solves the problem X.

The cost to determine whether or not $\Delta_{\Pi(s(w))} \cap \mathcal{M}_j^* \neq \emptyset$ (or $\Delta_{\Pi(s(w))} \cap (cod(w))^* \neq \emptyset$) is of order $O(|\Gamma|^2 \cdot |H|^2)$.

Hence, the running time of this algorithm can be bounded by $f(|w|) + O(|R|) + O(q \cdot |\Gamma|^2 \cdot n^2)$, where f is the (total) cost of a polynomial encoding from X to $\mathbf{\Pi}$, R is the set of rules of $\Pi(s(w))$, and q is the number of (initial) cells of $\Pi(s(w))$. But from Definition 3 we have that all involved parameters are polynomials in $|w|$. That is, the algorithm is polynomial in the size $|w|$ of the input. $\qquad\square$

In [3] a polynomial time solution of the Vertex Cover problem was given by using a family of recognizer tissue P systems with cell division and communication rules of length at most 3. Then $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TD}(3)}$.

Hence, in the framework of recognizer tissue P systems with cell division the length of the communication rules provides a borderline between efficiency and non-efficiency. Specifically, a frontier is obtained when we pass from length 1 to length 3.

5 Final Remarks and Future Work

It is well known [2] that tissue P systems with communication rules and without division rules can efficiently solve only tractable problems. It is also well known that by adding division rules we can efficiently solve **NP**–complete problems in linear time by using communication rules with length at most 3 [3].

In order to obtain new borderlines between tractability and intractability of problems, we study the possibility to restrict the length of communication rules to 1, allowing division rules. By using the dependency graph technique of cell–like P systems, we have shown that only tractable problems can be efficiently solved in that scenario.

Several questions regarding the role of the length remain open, for example:

- What happens if we consider tissue P systems using communication rules of length at most 2?
- In the solution provided in [3], antiport rules of length at most 3 were used. Would it be possible to provide another solution in which all rules of length 3 were symport?

Other open issues related to tissue P systems that may be interesting are:

- Analyzing a new role for the environment. More specifically, consider in the initial configuration only permitting objects with finite multiplicity in the environment . It seems that this new scenario would be equivalent to tissue P systems without environment, with a new distinct cell with no division rules associated. Is it still possible to solve **NP**–complete problems in polynomial time in this new framework, permitting division rules?
- Considering variations in the semantics of division rules, for example, dispensing with replication or with evolution. Division rules without replication would obey the syntax $[\, a\,]_i \to [\;]_i [\, u\,]_i$, where $i \in \{1, 2, \ldots, q\}$, $a \in \Gamma$ and $u \in \Gamma^*$, meaning that under the influence of object a, the cell with label i is divided in two cells with the same label. The first copy contains all objects of the original cell except for a and in the second copy the content of the original cell is replaced by the multiset u. Division rules without evolution would be either of the form $[\, a\,]_i \to [\;]_i [\;]_i$ or $[\, a\,]_i \to [\, a\,]_i [\, a\,]_i$, where $i \in \{1, 2, \ldots, q\}$ and $a \in \Gamma$. In both cases, under the influence of object a, the cell with label i is divided in two cells. All objects are replicated and copies of them are placed in the two new cells, except for a in the first case.

Acknowledgement

References

1. D. Díaz–Pernil: *Sistemas celulares de tejidos: Formalización y eficiencia computacional.* Ph D. Thesis, University of Sevilla, 2008.
2. D. Díaz–Pernil, M.J. Pérez–Jiménez, A. Romero–Jiménez: Efficient simulation of tissue-like P systems by transition cell-like P systems. *Natural Computing*, online version (http://dx.doi.org/10.1007/s11047-008-9102-z).
3. D. Díaz–Pernil, M.J. Pérez–Jiménez, A. Riscos–Núñez, A. Romero–Jiménez: Computational efficiency of cellular division in tissue-like membrane systems. *Romanian Journal of Information Science and Technology*, 11, 3 (2008), 229–241.
4. P. Frisco, H.J. Hoogeboom: Simulating counter automata by P systems with symport/antiport. *Lecture Notes in Computer Science*, 2597 (2003), 288–301.
5. M.A. Gutiérrez–Naranjo, M.J. Pérez–Jiménez, A. Riscos–Núñez, F.J. Romero–Campero: On the power of dissolution in P systems with active membranes. *Lecture Notes in Computer Science*, 3850 (2006), 224–240.
6. M.A. Gutiérrez–Naranjo, M.J. Pérez–Jiménez, A. Riscos–Núñez, F.J. Romero–Campero, A. Romero–Jiménez: Characterizing tractability by cell-like membrane systems. In K.G. Subramanian, K. Rangarajan, M. Mukund, eds., *Formal models, languages and applications*, World Scientific, 2006, chapter 9, 137–154.
7. C. Martín–Vide, J. Pazos, Gh. Păun, A. Rodríguez–Patón: Tissue P systems. *Theoretical Computer Science*, 296 (2003), 295–326.
8. A. Păun, Gh. Păun: The power of communication: P systems with symport/antiport. *New Generation Computing*, 20, 3 (2002), 295–305.
9. Gh. Păun: Computing with membranes. *Journal of Computer and System Sciences*, 61, 1 (2000), 108–143.
10. Gh. Păun: *Membrane Computing. An Introduction.* Springer–Verlag, Berlin, 2002.
11. Gh. Păun, M.J. Pérez–Jiménez: Recent computing models inspired from biology: DNA and membrane computing. *Theoria*, 18, 46 (2003), 72–84.
12. Gh. Păun, M.J. Pérez–Jiménez, A. Riscos–Núñez: Tissue P Systems with cell division. In Gh. Păun, A. Riscos–Núñez, A. Romero–Jiménez and F. Sancho–Caparrini, eds., *Second Brainstorming Week on Membrane Computing*, Sevilla, Report RGNC 01/2004, (2004), 380–386.
13. M.J. Pérez–Jiménez, A. Romero–Jiménez, F. Sancho–Caparrini: Complexity classes in cellular computing with membranes, *Natural Computing*, 2, 3 (2003), 265–285.
14. M.J. Pérez–Jiménez, A. Romero–Jiménez, F. Sancho–Caparrini: A polynomial complexity class in P systems using membrane division, *Journal of Automata, Languages and Combinatorics*, 11, 4 (2006), 423–434.
15. P systems website: `http://ppage.psystems.eu/`