

# Sea, North, History, Narrative, Energy, Climate: Papers from the 2012 Academia Europæa Bergen Meeting

## The P versus NP Problem from the Membrane Computing View

---

MARIO J. PÉREZ-JIMÉNEZ

Research Group on Natural Computing, Department of Computer Science and Artificial Intelligence, University of Sevilla, Avda. Reina Mercedes s/n, 41012 Sevilla, Spain.  
E-mail: marper@us.es

In the last few decades several computing models using powerful tools from Nature have been developed (because of this, they are known as bio-inspired models). Commonly, the space-time trade-off method is used to develop efficient solutions to computationally hard problems. According to this, implementation of such models (in biological, electronic, or any other substrate) would provide a significant advance in the practical resolution of hard problems. Membrane Computing is a young branch of Natural Computing initiated by Gh. Păun at the end of 1998. It is inspired by the structure and functioning of living cells, as well as from the organization of cells in tissues, organs, and other higher order structures. The devices of this paradigm, called P systems or membrane systems, constitute models for distributed, parallel and non-deterministic computing. In this paper, a computational complexity theory within the framework of Membrane Computing is introduced. Polynomial complexity classes associated with different models of cell-like and tissue-like membrane systems are defined and the most relevant results obtained so far are presented. Different borderlines between efficiency and non-efficiency are shown, and many attractive characterizations of the  $P \neq NP$  conjecture within the framework of this bio-inspired and non-conventional computing model are studied.

### 1. Introduction

The main objective of Computability Theory is to define the informal idea of mechanical/algorithmic problems resolution in a rigorous way. Each formal definition of the said concept provides a computing model. A basic question is to determine the class of all

problems that can be solved by a computing model when using the algorithms defined in it. In any computing model that captures the informal idea of an algorithm, there will always be undecidable problems, that is, problems that cannot be solved by using the algorithms of the model.

Analysing an algorithm that solves a problem consists of determining an upper bound for the minimal resource requirements with which the problem can be solved. One of the main goals of Complexity Theory is the study of the resources required for solving problems, and to provide tools allowing one to classify the problems with respect to the amount of resources needed for their resolution. These classes allow us to detect some of the inherent difficulties to the computational resolution of some problems, and they provide a classification of the abstract problems according to the resources they need to be solved in a given model.

Membrane Computing is a branch of Natural Computing providing distributed, parallel, synchronous and non-deterministic models of computation whose computational devices are called membrane systems. They are inspired by some basic biological features of the living cells (their structure, organization and functioning), as well as from the cooperation of cells in tissues, organs, and organisms. Membrane Computing was selected by the Thomson Institute for Scientific Information as a fast Emerging Research Front in computer science in 2003.

In this discipline there are basically two ways to consider computational devices: cell-like membrane systems (P systems) and tissue-like membrane systems (tissue P systems). The first one, using the biological membranes arranged hierarchically, inspired from the structure of the cell, and the second one using the biological membranes (called cells in this approach) placed in the nodes of a directed graph, inspired from the cell inter-communication in tissues. The objective of this paper is to introduce a Computational Complexity Theory in the framework of Membrane Computing and to analyse the **P** versus **NP** problem from the new perspective provided by this unconventional bio-inspired model of computing. We also present new frontiers of the efficiency of Membrane Computing models, in the sense of determining their capability to provide feasible solutions to hard problems. Each one of such borderlines provides new tools to attack the aforementioned problem.

The paper is organized as follows. First, we introduce the basic concepts that will be used throughout this paper. Next, we study the efficiency of different models of cell-like membrane systems (Section 3) and tissue-like membrane systems (Section 4). Some conclusions are presented in the final section.

## **2. Membrane Systems**

A membrane system can be viewed as a finite set of basic units of processors (called membranes in the case of cell-like approach and cells in the case of tissue-like) arranged in an organised structure (a rooted tree in the case of cell-like and a directed graph in the case of tissue-like) and delimiting regions able to contain multisets of objects (symbols from a working alphabet  $\Gamma$ ), which are abstractions of chemical substances. A finite set  $R$  of rewriting rules, abstractions of chemical reactions, provides the dynamics of the

system. There exists a double parallelism in the application of rules, once at the level of each region (the rules are used in parallel) and once at the level of the system (all regions evolve concomitantly). Different models of membrane systems can be considered according to the syntax and the semantics of the rules used.

For cell-like systems the tree structure is explicitly included in their definition. The root of the tree is called the skin membrane of the system, and the leaves of the tree are called elementary membranes. However, in tissue-like systems the graph structure is not explicitly given, it comes implicitly determined through some especial rules of the system (communication rules). Every membrane system also has an associated environment, which represents the region surrounding the system. In the case of cell-like approach, it has a passive role, in the sense that it is initially empty and only receives objects from the system through the root of the tree (the skin membrane). On the other hand, in tissue-like systems, the environment is considered to have initially an arbitrarily large number of copies of objects from a distinguished alphabet  $E$ , contained within the working alphabet. Besides, the environment can interact with any cell of the system, not only receiving objects but also sending objects inside the cells. For this reason, the environment is considered as an additional region in tissue-like approach.

A configuration at any instant of a membrane system is described by all multisets of objects over the working alphabet, associated with all the processors (membranes or cells) present in the system. In cell-like systems the current tree structure must be considered, and in tissue  $P$  systems we also consider the multiset of objects from  $\Gamma \setminus E$  present in the environment at that moment. A configuration is a halting configuration if no rule of the system is applicable to it. We can pass from the current configuration to another one in one transition step by applying the rules from  $R$ . The objects to evolve in a transition step and the rules by which they evolve are chosen in a non-deterministic and maximally parallel manner: we assign objects to rules but in such a way that after this assignation no further rule can be applied to the remaining objects.

A computation of the system is a (finite or infinite) sequence of configurations such that: (a) the first term of the sequence is the initial configuration of the system; (b) each remaining term is obtained from the previous configuration by applying rules of the system; and (c) if the sequence is finite (called halting computation) then the last term of the sequence must be a halting configuration.

Throughout this paper any membrane system has a distinguished alphabet, the input alphabet  $\Sigma$ , contained in the working alphabet,  $\Gamma$ , which allows us to encode instances of a problem. The idea is to impose that the initial multisets appearing in the definition of the system can only contain objects from  $\Gamma \setminus \Sigma$ , and an additional multiset over  $\Sigma$ , encoding the instance to be solved, will be introduced into a distinguished (input) region/processor before starting the computation. For each multiset  $m$  over the input alphabet we have an initial configuration associated with it, obtained by adding  $m$  to the initial multiset associated with the input region. It is worth pointing out that the answer of a halting computation of a membrane system will be encoded by the contents of the environment associated with the corresponding halting configuration, as it will be explained below.

### 2.1. Recognizer Membrane Systems

In order to study the computational efficiency of membrane systems, the notions from classical Computational Complexity Theory are adapted for Membrane Computing, and a special class of cell-like P systems was introduced in Ref. 1: recognizer P systems (called accepting P systems in a previous paper<sup>2</sup>). The same idea was adapted for the tissue-like case in Ref. 3, introducing recognizer tissue P systems.

A membrane system is a recognizer system if: (a) the working alphabet has two distinguished objects *yes* and *no*, but in the case of tissue P systems, at least, one copy of them is present in some initial multisets but none of them are present initially in the environment; (b) all computations halt; and (c) for each computation of the system, 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.

We say that a computation C of a recognizer membrane system is an accepting computation (respectively, rejecting computation) if object *yes* (respectively, object *no*) appears in the environment associated with the corresponding halting configuration of C, and neither object *yes* nor object *no* appears in the environment associated with any non-halting configuration of C.

### 2.2. Polynomial Complexity Classes of Membrane Systems

Let us recall that a decision problem is a pair  $(I_X, \Theta_X)$  where  $I_X$  is a language over a finite alphabet (whose elements are called instances) and  $\Theta_X$  is a total Boolean function over  $I_X$ . Next, we define what solving a decision problem efficiently means in the framework of membrane systems.

**Definition 1.** We say that a decision problem  $X = (I_X, \Theta_X)$  is solvable in polynomial time by a family  $\prod = \{\Pi(n) \mid n \in \mathbb{N}\}$  of recognizer membrane systems if the following holds.

- (1) The family  $\prod$  is polynomially uniform by Turing machines, that is, there exists a deterministic Turing machine working in polynomial time such that on input  $1^n$ , constructs the system  $\Pi(n)$ .
- (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  $\text{cod}(u)$  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  $\prod$  is polynomially bounded with regard to  $(X; \text{cod}; s)$ , that is, there exists a polynomial function  $p$ , such that for each instance  $u \in I_X$  every computation of  $\Pi(s(u))$  with input  $\text{cod}(u)$  is halting and it performs at most  $p(|u|)$  steps;
  - (d) the family  $\prod$  is sound with regard to  $(X; \text{cod}; s)$ , that is, for each instance  $u \in I_X$ , if there exists an accepting computation of  $\Pi(s(u))$  with input  $\text{cod}(u)$ , then  $\Theta_X(u) = 1$ ;

- (e) the family  $\prod$  is complete with regard to  $(X; \text{cod}; s)$ , that is, for each instance  $u \in I_X$ , if  $\Theta_X(u) = 1$ , then every computation of  $\Pi(s(u))$  with input  $\text{cod}(u)$  is an accepting one.

From the soundness and completeness conditions above we deduce that every system  $\Pi(n)$  from the family is confluent, in the following sense: every computation of a system  $\Pi(s(u)) + \text{cod}(u)$  processing an instance  $u$  of the problem must always give the same answer.

Let us recall that an instance of a decision problem is accepted by a non-deterministic Turing machine processing it, if there exists, at least, an accepting computation. Then, the classical notion of acceptance is not a true algorithmic concept. Nevertheless, an instance will be accepted by a membrane system processing it, if and only if each computation of the system is an accepting computation. Thus, the presented definition of acceptance by a membrane system substantially differs from the acceptance by a non-deterministic Turing machine.

Let  $\mathbf{R}$  be a class of recognizer membrane systems. We denote by  $\mathbf{PMC}_{\mathbf{R}}$  the set of all decision problems that can be solved in polynomial time by means of families of systems from  $\mathbf{R}$ . The class  $\mathbf{PMC}_{\mathbf{R}}$  is closed under complement and polynomial-time reductions.<sup>2</sup>

### 3. Efficiency of Cell-like Membrane Systems

This section studies the computational efficiency of several models of cell-like membrane systems, that is, their capability to solve computationally hard problems in an efficient way. These variants are considered by using specific kind of rules and, also, by adding new ingredients as electrical charges.

#### 3.1. Basic Transition P Systems

Basic transition P systems are a kind of cell-like membrane systems whose membrane structure does not grow, that is, there are no rules that produce new membranes in the system.

First of all, in order to formally define what it means that a family of membrane systems simulates a Turing machine in an efficient way, we shall introduce for each Turing machine a decision problem associated with it.

**Definition 2.** Let  $M$  be a deterministic Turing machine with input alphabet  $\Sigma_M$ . The decision problem associated with  $M$  is the problem  $X_M = (I_M; \Theta_M)$ , where  $I_M = \Sigma_M^*$ , and for every  $u \in \Sigma_M^*$ ,  $\Theta_M(u) = 1$  if and only if  $M$  accepts  $u$ .

Obviously, the decision problem  $X_M$  is solvable by the Turing machine  $M$ .

**Definition 3.** Let  $\mathbf{R}$  be a class of recognizer membrane systems. We say that a deterministic Turing machine  $M$  is simulated in polynomial time by a family of systems from  $\mathbf{R}$  if  $X_M \in \mathbf{PMC}_{\mathbf{R}}$ .

A basic transition P system is a cell-like membrane system whose rules are of the following form:  $[u \rightarrow v]_h$ ,  $u[ ]_h \rightarrow [v]_h$ ,  $[u]_h \rightarrow [ ]_h$ ,  $v$ , and  $[u]_h \rightarrow v$ , where  $h$  is the label of a

membrane and  $u, v$  are multisets over the working alphabet. These rules do not increase the size of the membrane structure, but the application of a rule of the last type decreases the number of membranes. We denote by  $T$  the class of recognizer basic transition P systems.

In Ref. 4 an efficient simulation of deterministic Turing machines by recognizer basic transition P systems was given. Thus, we have the following result.

**Proposition 1.** (Sevilla theorem)

Every deterministic Turing machine working in polynomial time can be simulated in polynomial time by a family of recognizer basic transition P systems.

It was also proved that each confluent basic transition P system can be (efficiently) simulated by a deterministic Turing machine.<sup>4</sup> As a consequence, these membrane systems efficiently solve at most tractable problems. Therefore, we have the following.

**Proposition 2.** If a decision problem is solvable in polynomial time by a family of recognizer basic transition P systems, then there exists a deterministic Turing machine solving it in polynomial time.

From Proposition 1 and Proposition 2 we deduce the following result.

**Theorem 1.**  $P = PMC_T$ .

Thus, the ability of a basic transition P system to create exponential workspace (in terms of number of objects) in polynomial time (e.g. via rules of the type  $[a \rightarrow a^2]_h$ ) is not enough to efficiently solve NP-complete problems (assuming that  $P \neq NP$ ).

Theorem 1 provides a tool to attack conjecture  $P \neq NP$  in the framework of Membrane Computing.

**Corollary 1.**  $P \neq NP$  if and only if every, or at least one, NP-complete problem cannot be solved in polynomial time by families of basic transition P systems.

### 3.2. P Systems with Active Membranes

Replication is one of the most important functions of a cell and, in ideal circumstances, a cell produces two identical copies by division. Mitosis is an elegant process of cell division that results in the production of two daughter cells from a single parent cell. Daughter cells are identical to one another and to the original parent cell. Through a sequence of steps, the replicated genetic material in a parent cell is equally distributed to two daughter cells. While there are some subtle differences, mitosis is remarkably similar across organisms. Bearing in mind that the reactions that take place in a cell are related to membranes, rules for membrane division are considered.

P systems with active membranes were first introduced by Gh. Păun.<sup>5</sup> This kind of cell-like membrane systems has electrical charges associated with membranes, which can be modified by the application of some rules (although labels always remain unchanged). They do not use cooperation, and they incorporate membrane division rules.

The rules of a P system with active membranes are of the following forms:

- (a)  $[a \rightarrow u]_h^\alpha$  (object evolution rules).
- (b)  $a[\ ]_h^{\alpha_1} \rightarrow [b]_h^{\alpha_2}$  (send-in communication rules).
- (c)  $[a]_h^{\alpha_1} \rightarrow [\ ]_h^{\alpha_2} b$  (send-out communication rules).
- (d)  $[a]_h^\alpha \rightarrow b$  (dissolution rules).
- (e)  $[a]_h^{\alpha_1} \rightarrow [b]_h^{\alpha_2} [c]_h^{\alpha_3}$  (division rules for elementary membranes).
- (f)  $[[\ ]_{h_1}^{\alpha_1} [\ ]_{h_2}^{\alpha_2}]_h^\alpha \rightarrow [[\ ]_{h_1}^{\alpha_3} [\ ]_{h_2}^{\alpha_4}]_h^\beta$  (division rules for non-elementary membranes).

where  $a, b, c$  are objects,  $u$  is a multiset of objects, and  $\alpha, \beta, \gamma, \alpha_1, \alpha_2, \alpha_3, \alpha_4$  are electrical charges from the set  $\{+, 0, -\}$ .

The rules of a P system with active membranes are applied according to the following principles (see Ref. 5 for details).

- The rules associated with membranes labelled by  $h$  are used for all copies of this membrane. At one step, a membrane can be the subject of only one rule of types (b)–(f).
- All the rules are applied in parallel and in a maximal manner. In one step, one object of a membrane can be used by only one rule (chosen in a non-deterministic way), but any object that can evolve by one rule of any form, must evolve, with the restriction just mentioned above.
- If a membrane is dissolved, its contents (multiset and internal membranes) are left free in the surrounding region, or in the nearest non-dissolved ancestor. The skin membrane cannot be dissolved.
- If at the same time a membrane labelled by  $h$  is divided by a rule of type (e) and there are objects in this membrane that evolve by means of rules of type (a), then we suppose that first the evolution rules of type (a) are used, and then the division is produced. Of course, this process takes only one step. Analogously for rules of type (f). The skin membrane cannot be divided.

Let us denote by  $AM$  the class of recognizer P systems with active membranes, and let  $NAM$  be the class of recognizer P systems with active membranes that do not make use of division rules.

In the framework of cell-like membrane systems, confluent recognizer P systems with active membranes making use of no division rule, can be efficiently simulated by a deterministic Turing machine. This can be shown by adapting the proof of the following result, given in Ref. 6.

**Proposition 3.** (Milano theorem.) A deterministic P system with active membranes but without membrane division can be simulated by a deterministic Turing machine with a polynomial slowdown.

As a consequence of the previous result, the following holds.

**Corollary 2.**  $PMC_{NAM} \subseteq P$ .

In Ref. 7, a simple proof of each tractable problem being solvable by a family of recognizer P systems with active membranes (without polarizations) operating in exactly

one step and using only send-out communication rules, has been provided. Thus, we have the following.

**Proposition 4.**  $P \subseteq \text{PMC}_{NAM}$ .

From Corollary 2 and Proposition 4 we deduce the following result.

**Proposition 4.**  $P = \text{PMC}_{NAM}$ .

Let us notice that Theorem 2 can be considered as a version of Theorem 1 for the class *NAM*.

**Remark 1.** In the framework of P systems with active membranes, the use or not of the division rules provides a borderline for the tractability of decision problems, assuming that  $P \neq NP$ , that is, by using division rules we can solve NP-complete problems in polynomial time, but without division rules only problems in P can be solved in an efficient way.

We denote by  $AM(+n)$  (respectively,  $AM(-n)$ ) the class of recognizer P systems with active membranes using division rules for elementary and non-elementary membranes (respectively, only for elementary membranes).

In the framework of  $AM(-n)$ , efficient solutions to weakly NP-complete problems (Knapsack,<sup>8</sup> Subset Sum,<sup>9</sup> Partition<sup>10</sup>), and strongly NP-complete problems (SAT,<sup>2</sup> Clique,<sup>11</sup> Bin Packing,<sup>12</sup> Common Algorithmic Problem<sup>13</sup>) have been given. In particular, the following.

**Proposition 5.**  $SAT \in \text{PMC}_{AM(-n)}$ .

Since  $\text{PMC}_R$  is closed under complement and polynomial-time reductions, for any class R of recognizer membrane systems, the following result holds.

**Proposition 6.**  $NP \cup \text{co-NP} \subseteq \text{PMC}_{AM(-n)}$ .

In the framework of  $AM(+n)$  it has been shown<sup>14</sup> that the QBF-SAT problem can be solved in a linear time by a family of recognizer P systems with active membranes (without using dissolution rules) and using division rules for elementary and non-elementary membranes. Thus, we have the following result.

**Proposition 7.**  $\text{PSPACE} \subseteq \text{PMC}_{AM(+n)}$ .

In Ref. <sup>15</sup>, a (deterministic and efficient) algorithm simulating a single computation of any confluent recognizer P system with active membranes, has been described. Such P systems can be simulated by a deterministic Turing machine working with exponential space, and spending a time of the order  $O(2^{p(n)})$ , for some polynomial  $p(n)$ . Thus, we have  $\text{PMC}_{AM(+n)} \subseteq \text{EXP}$ , and hence we have the following.

**Proposition 8.**  $\text{PSPACE} \subseteq \text{PMC}_{AM(+n)} \subseteq \text{EXP}$ .

Previous results show that the usual framework of P systems with active membranes for solving decision problems is too powerful from the computational complexity point



of view. Therefore, it would be interesting to investigate weaker models of P systems with active membranes able to characterize classical complexity classes below NP and providing borderlines between efficiency and non-efficiency.

### 3.3. Polarizationless P Systems with Active Membranes

In this kind of P system with active membranes, the membranes of the system have no electrical charges. We denote  $AM^0(\alpha, \beta)$  as the class of all recognizer polarizationless P systems with active membranes such that: (a) if  $\alpha = +d$  (respectively,  $\alpha = -d$ ) then dissolution rules are permitted (respectively, forbidden); and (b) if  $\beta = +n$  (respectively,  $-n$ ) then division rules for elementary and non-elementary membranes (respectively, only for elementary membranes) are permitted.

At the beginning of 2005, Gh. Păun (problem F from Ref. 16) wrote:

My favorite question (related to complexity aspects in P systems with active membranes and with electrical charges) is that about the number of polarizations. Can the polarizations be completely avoided? The feeling is that this is not possible – and such a result would be rather sound: passing from no polarization to two polarizations amounts to passing from non-efficiency to efficiency.

This so-called Păun's conjecture can be formally formulated in terms of membrane computing complexity classes as follows:  $\mathbf{P} = \mathbf{PMC}_{AM0(+d, -n)}$ .

Let  $\Pi$  be a recognizer polarizationless P system with active membranes that do not make use of dissolution rules. A directed graph (called a *dependency graph*) can be associated with  $\Pi$  verifying the following property: every accepting computation of  $\Pi$  is characterized by the existence of a path in the graph between two distinguished nodes. Based on this concept and by using the tractability of the *reachability problem*, the following result has been proved.<sup>17</sup>

**Theorem 3.**  $\mathbf{P} = \mathbf{PMC}_{AM0(-d, -n)} = \mathbf{PMC}_{AM0(-d, +n)}$

Thus, polarizationless P systems with active membranes that do not make use of dissolution rules cannot solve NP-complete problems in polynomial time (unless  $\mathbf{P} = \mathbf{NP}$ ). This result can be considered as a partial affirmative answer to the Păun's conjecture.

Let us now consider polarizationless P systems with active membranes making use of dissolution rules. Will it be possible to solve NP-complete problems in that framework? Several authors<sup>17,18</sup> gave a positive answer when division for non-elementary membranes are allowed. The mentioned papers provide solutions in a linear time to SAT and Subset Sum problems, respectively. Thus, we have the following result.

**Theorem 4.**  $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{AM0(+d, +n)}$ .

As a consequence of Theorem 4, a partial negative answer to Păun's conjecture is given: assuming that  $\mathbf{P} \neq \mathbf{NP}$  and making use of dissolution rules and division rules for elementary and non-elementary membranes, computationally hard problems can be efficiently solved avoiding polarizations. The answer is partial because efficient resolvability of NP-complete problems by polarizationless P systems with active membranes making use of dissolution rules and division only for elementary membranes is unknown.

**Remark 2.** In the framework of polarizationless P systems with active membranes, the use or not of the dissolution rules provides a borderline for the tractability of decision problems, assuming that  $\mathbf{P} \neq \mathbf{NP}$ , that is, in the framework  $\mathbf{AM}^0(+n)$  by using dissolution rules we can solve  $\mathbf{NP}$ -complete problems in polynomial time, but without dissolution rules only problems in  $\mathbf{P}$  can be solved in an efficient way.

#### 4. Efficiency of Tissue-like Membrane Systems

In this section, we study the efficiency of different models of tissue-like membrane systems. These variants are considered by using specific kind of rules. The basic one only considers communication rules but we also analyse tissue P systems where division or separation rules are also permitted.

##### 4.1. Basic Tissue P Systems

Basic tissue P systems are characterized by the fact that they use only communication rules, that is, rules of the symport/antiport type:  $(i,u/v,j)$ , where  $i,j$  are different regions of the system (one of which might be the environment) and  $u,v$  are multisets of objects not simultaneously empty. These rules provide one or two arcs according to the following: one arc from region  $i$  to region  $j$  (if  $u$  is nonempty), and another arc from region  $j$  to region  $i$  (if  $v$  is nonempty). When applying a rule  $(i,u/v,j)$ , the objects of the multiset represented by  $u$  are transferred from region  $i$  to region  $j$  and, simultaneously, the objects of multiset  $v$  are sent from region  $j$  to region  $i$  in exchange. The length of the communication rule  $(i,u/v,j)$  is the total sum of objects that appear in multisets  $u$  and  $v$ .

We denote by TC the class of all recognizer basic tissue P systems. In Ref. 19 it was shown that every family of recognizer basic tissue P systems, which solves a decision problem, can be efficiently simulated by a family of recognizer basic transition P systems solving the same problem. Then, we have the following result.

##### **Theorem 5.** $\mathbf{P} = \mathbf{PMC}_{\text{TC}}$

That is, basic tissue P system models are non-efficient in the sense that only problems in  $\mathbf{P}$  can be solved by this kind of membrane systems in polynomial time, assuming that  $\mathbf{P} \neq \mathbf{NP}$ .

##### 4.2. Tissue P Systems with Cell Division

In tissue P systems with cell division, the rules of the system are communication rules and cell division rules, which are of the form  $[a]_i \rightarrow [b]_i [c]_i$ . When applying this rule, 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.

The rules of a tissue P system with cell division are applied in a non-deterministic maximally parallel manner as it is customary in Membrane Computing, with the following important remark: if a cell divides, only the division rule is applied to that cell at that step; the objects inside that cell do not evolve by means of communication rules.

In other words, we can think that before division a cell interrupts all its communication channels with the other cells and with the environment. The new cells resulting from division will only interact with other cells or with the environment at the next step – providing they do not divide once again. The label of a cell identifies the rules that can be applied to it precisely.

For each natural number  $k \geq 1$ , we denote by  $\mathbf{TDC}(k)$  the class of recognizer tissue P systems with cell division such that each communication rule of the system has a length at most  $k$ .

By using the concept of a dependency graph associated with tissue P systems with cell division and communication rules with length at most 1, it has been proved that this kind of membrane system can only efficiently solve tractable problems (see Ref. 20 for details). Thus, the following holds.

**Theorem 6.**  $\mathbf{P} = \mathbf{PMC}_{\mathbf{TDC}(1)}$ .

In Ref. 21, a polynomial-time solution of the HAM-CYCLE problem was given by using a family of recognizer tissue P systems with cell division and communication rules of length at most 2. Therefore, we have the following.

**Proposition 9.**  $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{\mathbf{TDC}(2)}$ .

**Remark 3.** From Theorem 6 and Proposition 9, we deduce that in the framework of recognizer tissue P systems with cell division, the length of the communication rules provides a borderline of the tractability of decision problems. Specifically, passing from length 1 to length 2 amounts to passing from non-efficiency to efficiency, assuming that  $\mathbf{P} \neq \mathbf{NP}$ . Moreover, another frontier is obtained when we add cell division rules in basic tissue P systems.

#### 4.3. Tissue P Systems with Cell Separation

The biological inspiration of *tissue P systems with cell separation* is the following: alive tissues are not *static* network of cells, since new cells are generated by membrane fission in a natural way. In this kind of tissue P system, the rules of the system are communication rules and separation rules. A separation rule is of the form  $[a]_i \rightarrow [\Gamma_1]_i [\Gamma_2]_i$ , being  $\{\Gamma_1, \Gamma_2\}$  a fixed partition of the working alphabet associated with the system. This rule is applicable to cell  $i$  if object  $a$  is contained in that cell. When applying such a separation rule, 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; while the rest of the contents of the cell are distributed as follows: objects from  $\Gamma_1$  are placed in the first cell, and those from  $\Gamma_2$  are placed in the second cell; the output cell  $i_{\text{out}}$  cannot be separated.

The rules are used in a non-deterministic maximally parallel manner, with a restriction: when a cell is separated, the separation rule is the only one that is applied for that cell at that step; thus, the objects inside that cell do not evolve by means of communication rules.

For each natural number  $k \geq 1$ , we denote by  $\mathbf{TSC}(k)$  the class of recognizer tissue P systems with cell separation such that each communication rule of the system has a length at most  $k$ .

By using the concept of dependency graph associated with tissue P systems with cell separation and communication rules with length at most 1, it has been proved that this kind of membrane systems can only efficiently solve tractable problems (see Ref. 22 for details). This result has been extended to tissue P systems with cell separation and communication rules with length at most two<sup>23</sup> through an algorithmic method. Then, we have the following result.

**Theorem 7.**  $P = PMC_{TSC(2)}$ .

**Remark 4.** From Proposition 9 and Theorem 7 we deduce that the use of cell division rules instead of cell separation rules in tissue P systems with communication rules with length at most 2, provides efficient computing models.

In Ref. 24, a polynomial-time solution of the SAT problem was given by using a family of recognizer tissue P systems with cell separation and communication rules of length at most 3. Thus, we have the following result.

**Proposition 10.**  $NP \cup co-NP \subseteq PMC_{TSC(3)}$ .

**Remark 5.** From Theorem 7 and Proposition 10, we deduce that in the framework of recognizer tissue P systems with cell separation the length of the communication rules provides a borderline of the tractability of decision problems. Specifically, passing from length 2 to length 3 amounts to passing from non-efficiency to efficiency, assuming that  $P \neq NP$ . Moreover, another frontier is obtained by adding separation rules to basic tissue P systems.

#### 4.4. Tissue P Systems without Environment

Classical tissue P systems with cell division have a special alphabet whose elements initially appear in an arbitrary large number of copies. These objects are shared in a distinguished place of the system, called the environment. This property is not so nice from the complexity point of view. In this section we deal with tissue-like membrane systems where there is not an environment having the property mentioned above. Specifically, we analyse the role played by the environment in tissue-like membrane systems where division or separation rules are considered.

A *tissue P system without environment* is a tissue P system such that the alphabet of the environment is an empty set. For each natural number  $k \geq 1$ , we denote by  $\overline{TDC}(k)$  (respectively,  $\overline{TSC}(k)$ ) the class of recognizer tissue P systems with cell division (respectively, with cell separation), with communication rules of length at most  $k$ , and without environment.

In Ref. 25, it is shown that every family  $\Pi$  of tissue P systems with cell division solving a decision problem can be simulated by a family  $\Pi'$  of tissue P systems with cell division and without environment, in an efficient way. Moreover, the upper bound of the length of communication rules of the systems from  $\Pi$  is equal to the maximum length of communication rules of the system from  $\Pi'$ . That is, we have the following result.

**Theorem 8.** For each  $k \geq 1$  we have  $\text{PMC}_{TDC(k)} = \text{PMC}_{\overline{TDC(k)}}$ .

From this theorem we deduce that the environment is not relevant for the efficiency of tissue P systems with cell division. That is, for each polynomial-time solution of a decision problem  $X$  in  $TDC(k)$  we can construct a family of systems from  $\overline{TDC(k)}$  solving  $X$  in polynomial time. Moreover, the proof of the previous theorem provides a method to construct such a family.

Concerning the role of the environment in tissue P systems with cell separation, in Ref. 26 it is shown that only tractable problems can be solved efficiently by using tissue P systems with communication rules, separation rules and without environment. That is, the following holds.

**Theorem 9.** For each  $k \geq 1$  we have  $\text{PMC}_{\overline{TSC(k)}} = \text{P}$ .

From this theorem we deduce that the environment is relevant for the efficiency of tissue P systems with cell separation.

**Remark 6.** From Theorem 9 and Proposition 10, we deduce that in the framework of recognizer tissue P systems with cell separation and communication rules with length at most 3, passing from having an environment to not having it amounts to passing from efficiency to non-efficiency, assuming that  $\text{P} \neq \text{NP}$ .

**Remark 7.** Bearing in mind that NP-complete problems can be efficiently solved by using tissue P systems with cell division and without environment, we deduce that in the framework of tissue P systems without environment, the kind of rules (separation versus division) provides a new frontier of the tractability of decision problems, assuming that  $\text{P} \neq \text{NP}$ .

## 5. Conclusions

In this paper, a computational complexity in the membrane computing field has been presented. Assuming that  $\text{P} \neq \text{NP}$ , different boundaries of the efficiency have been described in terms of syntactical ingredients of membrane systems. Each of them provide new tools to attack the  $\text{P} \neq \text{NP}$  conjecture. Specifically, in the framework of membrane systems, the following ingredients provide new borderlines of the efficiency of computing models.

- The use of division rules in P systems with active membranes.
- The use of dissolution rules in polarizationless P systems with active membranes.
- The length of communication rules (passing from 1 to 2) in tissue P systems with cell division.
- The length of communication rules (passing from 2 to 3) in tissue P systems with cell separation.
- The use of cell division (or cell separation) rules in basic tissue P systems.
- The use of the environment in tissue P systems with cell separation.

It is worth highlighting that each one of the frontiers just mentioned can be seen as a new tool to attack the **P** versus **NP** conjecture, in the sense indicated in Corollary 1.

## References

1. M. J. Pérez-Jiménez, A. Romero-Jiménez and F. Sancho-Caparrini (2006) A polynomial complexity class in P systems using membrane division. *Journal of Automata, Languages and Combinatorics*, **11**(4), pp. 423–434.
2. M. J. Pérez-Jiménez, A. Romero-Jiménez and F. Sancho-Caparrini (2003) Complexity classes in models of cellular computing with membranes. *Natural Computing*, **2**(3), pp. 265–285.
3. Gh. Păun, M. J. Pérez-Jiménez and A. Riscos-Núñez (2008) Tissue P Systems with cell division. *International Journal of Computers, Communications and Control*, **3**(3), pp. 295–303.
4. M. A. Gutiérrez-Naranjo, M. J. Pérez-Jiménez, A. Riscos-Núñez, F. J. Romero-Campero and A. Romero-Jiménez (2006) Characterizing tractability by cell-like membrane systems. In: K. G. Subramanian, K. Rangarajan, M. Mukund (eds) *Formal models, languages and applications* (Singapore: World Scientific), pp. 137–154.
5. Gh. Păun (2001) P systems with active membranes: attacking NP-complete problems. *Journal of Automata, Languages and Combinatorics*, **6**(1), pp. 75–90.
6. C. Zandron, C. Ferretti and G. Mauri (2000) Solving NP-complete problems using P systems with active membranes. In: I. Antoniou, C. S. Calude and M. J. Dinneen (Eds) *Unconventional Models of Computation* (Berlin: Springer), pp. 289–301.
7. A. E. Porreca (2008) Computational complexity classes for membrane systems. Master Degree Thesis, Università di Milano-Bicocca, Italy.
8. M. J. Pérez-Jiménez and A. Riscos-Núñez (2004) A linear-time solution to the Knapsack problem using P systems with active membranes. *Lecture Notes in Computer Science*, **2933**, pp. 250–268.
9. M. J. Pérez-Jiménez and A. Riscos-Núñez (2005) Solving the Subset-Sum problem by active membranes. *New Generation Computing*, **23**(4), pp. 367–384.
10. M. A. Gutiérrez-Naranjo, M. J. Pérez-Jiménez and A. Riscos-Núñez (2005) A fast P system for finding a balanced 2-partition. *Soft Computing*, **9**(9), pp. 673–678.
11. A. Alhazov, C. Martín-Vide and L. Pan (2004) Solving graph problems by P systems with restricted elementary active membranes. *Lecture Notes in Computer Science*, **2950**, pp. 1–22.
12. M. J. Pérez-Jiménez and F. J. Romero-Campero (2004) An efficient family of P systems for packing items into bins. *Journal of Universal Computer Science*, **10**(5), pp. 650–670.
13. M. J. Pérez-Jiménez and F. J. Romero-Campero (2005) Attacking the Common Algorithmic Problem by recognizer P systems. *Lecture Notes in Computer Science*, **3354**, pp. 304–315.
14. A. Alhazov, C. Martín-Vide and L. Pan (2003) Solving a PSPACE-complete problem by recognizing P systems with restricted active membranes. *Fundamenta Informaticae*, **58**, pp. 67–77.
15. A. E. Porreca, G. Mauri and C. Zandron (2006) Complexity classes for membrane systems. *Informatique théorique et applications*, **40**(2), pp. 141–162.
16. Gh. Păun (2005) Further twenty six open problems in membrane computing. In: M.A. Gutiérrez-Naranjo, A. Riscos-Núñez, F.J. Romero-Campero and D. Surlan (eds) *Third Brainstorming Week on Membrane Computing* (Sevilla: Fénix Editora), pp. 249–262.

17. M. A. Gutiérrez-Naranjo, M. J. Pérez-Jiménez, A. Riscos-Núñez and F. J. Romero-Campero (2006) On the power of dissolution in P systems with active membranes. *Lecture Notes in Computer Science*, **3850**, pp. 224–240.
18. A. Alhazov, L. Pan and Gh. Păun (2004) Trading polarizations for labels in P systems with active membranes. *Acta Informaticae*, **41**(2–3), pp. 111–144.
19. D. Díaz-Pernil, M. A. Gutiérrez-Naranjo, M. J. Pérez-Jiménez and A. Romero-Jiménez (2009) Efficient simulation of tissue-like P systems by transition cell-like P systems. *Natural Computing*, **8**(4), pp. 797–806.
20. R. Gutiérrez-Escudero, M. J. Pérez-Jiménez and M. Rius-Font (2010) Characterizing tractability by tissue-like P systems. *Lecture Notes in Computer Science*, **5957**, pp. 289–300.
21. A. E. Porreca, N. Murphy and M. J. Pérez-Jiménez (2012) An optimal frontier of the efficiency of tissue P systems with cell division. In: M. García-Quismondo, L.F. Macías-Ramos, Gh. Păun, I. Pérez-Hurtado and L. Valencia-Cabrera (eds) *Proceedings of the Tenth Brainstorming Week on Membrane Computing, Volume II* (Seville: Fénix Editora), pp. 141–166.
22. L. Pan and M. J. Pérez-Jiménez (2010) Computational complexity of tissue-like P systems. *Journal of Complexity*, **26**(3), pp. 296–315.
23. L. Pan, M. J. Pérez-Jiménez, A. Riscos-Núñez and M. Rius-Font (2012) New frontiers of the efficiency in tissue P systems. In: L. Pan, Gh. Păun and T. Song (eds) *Pre-proceedings of Asian Conference on Membrane Computing* (Wuhan: Huazhong University of Science and Technology), pp. 61–73.
24. M. J. Pérez-Jiménez and P. Sosík (2013) An optimal frontier of the efficiency of tissue P systems with cell separation. *Computational Complexity*. Submitted.
25. M. J. Pérez-Jiménez, A. Riscos-Núñez, M. Rius-Font and F. J. Romero-Campero (2013) A polynomial alternative to unbounded environment for tissue P systems with cell division. *International Journal of Computer Mathematics*, **90**(4), pp. 760–775.
26. L. F. Macías, M. J. Pérez-Jiménez, A. Riscos-Núñez and M. Rius-Font (2013) The efficiency of tissue P systems with cell separation relies on the environment. *Lecture Notes in Computer Science*, **7762**, pp. 243–256.

### About the Author

**Mario J. Pérez-Jiménez** received a BSc degree in Mathematics from Barcelona University, Spain, in 1971, and currently is a full Professor in the Department of Computer Science and Artificial Intelligence at the University of Seville. His research interests include the theory of computation, computational complexity theory, natural computing, bioinformatics, and computational modelling for systems biology and population biology. Some years ago, he was a Guest Professor of the Huazhong University of Science and Technology, Wuhan, China (2005–2007). He is member of the Academia Europaea, the European Molecular Computing Consortium, the Board of Directors at the Foundation for the Research and Development of Technology in Andalusia as well as a member of the Institute of Mathematics of the University of Seville. He was the first scientist to receive an award for ‘Important Contributions to Membrane Computing’ under the auspices of the European Molecular Computing Consortium (Edinburgh 2008). Nowadays, he is a member of the Editorial Board of four ISI journals. From 2003 on, he has been an expert-reviewer of the *Prospective* and

Evaluation National Agency of Spain. In addition, he has been an independent expert to the evaluation of NEST (New and Emergent Science and Technology) proposals under the Sixth Framework Programme of the European Community, and, from May 2006 onwards, he has been a European Science Foundation peer reviewer.