New techniques to address the problem P vs. NP

Mario J. Pérez Jiménez
A. Riscos-Núñez

Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
Universidad de Sevilla

3rd Int. School on Biomolecular and Biocellular Computing
June 28-30, Valencia

You may say
I'm a dreamer ...

1. Introduction
   - P vs NP
   - Complexity Theory in Membrane Computing

2. Computational complexity framework
   - Borderlines of tractability

3. Final Comments

Goal:

- Unconventional approaches/tools to attack the **P versus NP problem** are given by using **Membrane Computing**.

Computability versus Complexity

Computability (1931)

- Define the informal idea of mechanical/algorithmic problems resolution in a rigorous way.
- Which problems are **computable** in a (universal) model?

Complexity (1970)

- Provide bounds on the amount of resources necessary for every mechanical procedure (algorithm) that solves a given problem.
- Which (computable) problems are **efficiently** solvable?

- Finding solutions versus checking the correctness of solutions.
- Proofs versus verifying their correctness.

- This is essentially the central problem of Computational Complexity theory

The P versus NP problem (II)

It is widely believed that:

- To solve a problem is **harder** than to check the correctness of a solution
- **P** $\neq$ **NP**.

Attacking the P versus NP problem

Classical approach (1970):

- **P** $\neq$ **NP**.
  - Find an **NP**-complete problem such that it does **not** belong to the class **P**.
- **P** $=$ **NP**.
  - Find an **NP**-complete problem such that it belongs to the class **P**.

Attacking the P versus NP problem

Classical approach (1970):

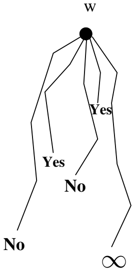
- **P** $\neq$ **NP**.
  - Find an **NP**-complete problem such that it does **not** belong to the class **P**.
- **P** $=$ **NP**.
  - Find an **NP**-complete problem such that it belongs to the class **P**.

Complexity Theory in Membrane Computing

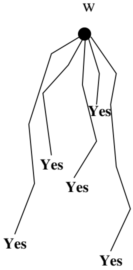
Unconventional approaches/tools to attack the **P versus NP problem** are given by using **Membrane Computing**.

- Polynomial complexity classes associated with (cell–like and tissue–like) P systems are presented.
- A notion of acceptance must be defined in the new framework (<u>different</u> than the classical one for nondeterministic Turing machines)

Nondeterministic Turing machines



Membrane systems



Decision problems

- ⋆ Computational complexity theory deals with **decision problems** which are problems that require a "*yes*" or "*no*" answer.
- ⋆ In real-life, many abstract problems are combinatorial optimization problems not decision problems.
- ⋆ Every decision problem has associated a language in a natural way.
- ⋆ The solvability of decision problems is defined through the **recognition** of the languages associated with them.

Decision problems

- ⋆ Computational complexity theory deals with **decision problems** which are problems that require a "*yes*" or "*no*" answer.
- ⋆ In real-life, many abstract problems are combinatorial optimization problems not decision problems.
- ⋆ Every decision problem has associated a language in a natural way.
- ⋆ The solvability of decision problems is defined through the **recognition** of the languages associated with them.

Recognizer P systems

Decision problems

- ⋆ Computational complexity theory deals with **decision problems** which are problems that require a "*yes*" or "*no*" answer.
- ⋆ In real-life, many abstract problems are combinatorial optimization problems not decision problems.
- ⋆ Every decision problem has associated a language in a natural way.
- ⋆ The solvability of decision problems is defined through the **recognition** of the languages associated with them.

Decision problems

- ⋆ Computational complexity theory deals with **decision problems** which are problems that require a "*yes*" or "*no*" answer.
- ⋆ In real-life, many abstract problems are combinatorial optimization problems not decision problems.
- ⋆ Every decision problem has associated a language in a natural way.
- ⋆ The solvability of decision problems is defined through the **recognition** of the languages associated with them.

Recognizer P systems

Definition

A *recognizer P system* is a P system with input and external output such that:

- *Yes*, *No*
  - for tissue, *Yes*, *No* $\in$ initial multisets (but not in $\mathcal{E}$**!**)
- **all** computations **halt**.
- for every computation, one symbol *Yes* or one symbol *No* (but not both) is sent out (and in the last step of the computation).
  - Accepting/rejecting computations

- The sets $\mathcal{NAM}$, $\mathcal{AM}(-ne)$ and $\mathcal{AM}(+ne)$.
- The sets $\mathcal{TC}$, $\mathcal{TDC}$, $\mathcal{TSC}$, and $\mathcal{TDC}(k)$, $\mathcal{TSC}(k)$, for each $k \geq 1$.

Polynomial time solvability

Definition

Let $\mathcal{R}$ be a class of recognizer P systems. A decision problem $X = (I_X, \theta_X)$ is solvable in polynomial time by a family $F_X = (\Pi(n))_{n \in \mathbb{N}^+}$, of $\mathcal{R}$, if

- F_X is polynomially uniform by Turing machines.

Polynomial time solvability

Definition

Let $\mathcal{R}$ be a class of recognizer P systems. A decision problem $X = (I_X, \theta_X)$ is solvable in polynomial time by a family $F_X = (\Pi(n))_{n \in \mathbb{N}^+}$, of $\mathcal{R}$, if

- F_X is polynomially uniform by Turing machines.
- There exists a pair (cod, s) of pol-time computable functions such that for every $u \in I_X$ we have $cod(u) \in I_{\Pi(s(u))}$, for every $n \in \mathbb{N}$ we have $s^{-1}(n)$ is a finite set, and
  - F_X is polynomially bounded with regard to (X, cod, s).
  - F_X is sound and complete, with regard to (X, cod, s).

Polynomial time solvability

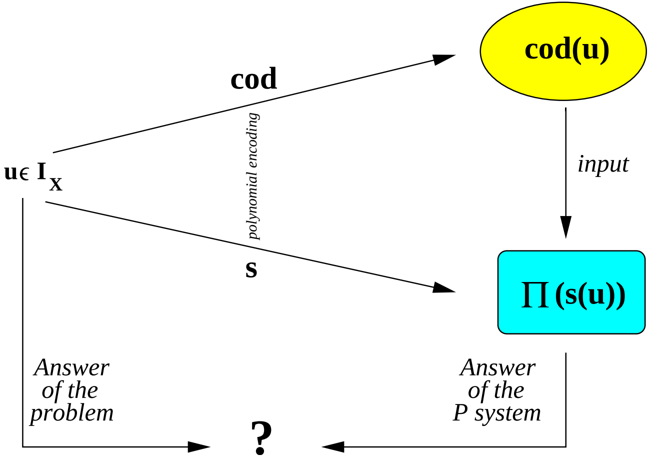
Definition

Let $\mathcal{R}$ be a class of recognizer P systems. A decision problem $X = (I_X, \theta_X)$ is solvable in polynomial time by a family $F_X = (\Pi(n))_{n \in \mathbb{N}^+}$, of $\mathcal{R}$, if

- F_X is polynomially uniform by Turing machines.
- There exists a pair (cod, s) of pol-time computable functions such that for every $u \in I_X$ we have $cod(u) \in I_{\Pi(s(u))}$, for every $n \in \mathbb{N}$ we have $s^{-1}(n)$ is a finite set, and
  - F_X is polynomially bounded with regard to (X, cod, s).
  - F_X is sound and complete, with regard to (X, cod, s).

We denote this by $X \in \mathbf{PMC}_{\mathcal{R}}$.
Closed under complement and polynomial–time reductions

Solvability of a decision problem



Efficiency of a membrane systems class

- **Efficiency**: Capability to solve **NP**-complete problems in polynomial time.
  - **NP** $\cup$ **co-NP** $\subseteq$ $\mathbf{PMC}_{\mathcal{R}}$.
- **Non-Efficiency**: **P** $=$ $\mathbf{PMC}_{\mathcal{R}}$.

Frontiers of the efficiency:

- M_1 efficient.
- M_2 non efficient.
- $M_2 \subseteq M_1$: each solution S of a problem X in M_2 is also a solution in M_1.

Passing from M_2 to M_1 amounts to passing from non efficiency to efficiency.

Efficiency of a membrane systems class

- **Efficiency**: Capability to solve **NP**-complete problems in polynomial time.
  - **NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{R}}$.
- **Non-Efficiency**: **P** $=$ **PMC**$_{\mathcal{R}}$.

Frontiers of the efficiency:

- M_1 efficient.
- M_2 non efficient.
- $M_2 \subseteq M_1$: each solution S of a problem X in M_2 is also a solution in M_1.

Passing from M_2 to M_1 amounts to passing from non efficiency to efficiency.

Efficiency of a membrane systems class

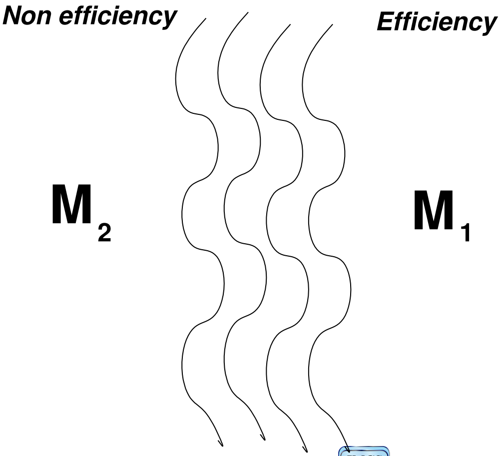
- **Efficiency**: Capability to solve **NP**-complete problems in polynomial time.
  - **NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{R}}$.
- **Non-Efficiency**: **P** $=$ **PMC**$_{\mathcal{R}}$.

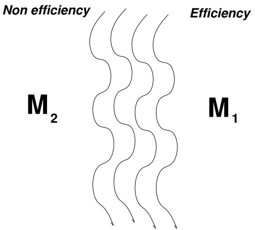
Frontiers of the efficiency:

- M_1 efficient.
- M_2 non efficient.
- $M_2 \subseteq M_1$: each solution S of a problem X in M_2 is also a solution in M_1.

Passing from M_2 to M_1 amounts to passing from non efficiency to efficiency.

Managing frontiers of the efficiency





P = NP

- Finding an **NP**-complete problem efficiently solvable in M_2.
  - Translating a polynomial time solution of an **NP**-complete problem in M_1, to a polynomial time solution in M_2.

P ≠ NP

- Finding an **NP**-complete problem not polynomial time solvable in M_2.

Non efficiency of basic P systems

- **Proposition 1** (Sevilla theorem, 2004)
  Every DTM working in polynomial time can be simulated in polynomial time by a family of recognizer basic P systems.

- **Proposition 2** (Milano theorem, 2000)
  If a decision problem is solvable in polynomial time by a family of recognizer basic P systems with input membrane, then there exists a DTM solving it in polynomial time.

- **Theorem: P $=$ PMC$_{\mathcal{T}}$** (Sevilla team, 2004).

  - **Corollary: P $\neq$ NP** if and only if every, or at least one, **NP**–complete problem is not in **PMC**$_{\mathcal{T}}$.

Non efficiency of basic P systems

- **Proposition 1** (Sevilla theorem, 2004)
  Every DTM working in polynomial time can be simulated in polynomial time by a family of recognizer basic P systems.

- **Proposition 2** (Milano theorem, 2000)
  If a decision problem is solvable in polynomial time by a family of recognizer basic P systems with input membrane, then there exists a DTM solving it in polynomial time.

- **Theorem: P $=$ PMC$_{\mathcal{T}}$** (Sevilla team, 2004).

  - **Corollary: P $\neq$ NP** if and only if every, or at least one, **NP**–complete problem is not in **PMC**$_{\mathcal{T}}$.

Non efficiency of basic P systems

- **Proposition 1** (Sevilla theorem, 2004)
  Every DTM working in polynomial time can be simulated in polynomial time by a family of recognizer basic P systems.

- **Proposition 2** (Milano theorem, 2000)
  If a decision problem is solvable in polynomial time by a family of recognizer basic P systems with input membrane, then there exists a DTM solving it in polynomial time.

- **Theorem: $\mathbf{P} = \mathbf{PMC}_{\mathcal{T}}$** (Sevilla team, 2004).

  - **Corollary: $\mathbf{P} \neq \mathbf{NP}$** if and only if every, or at least one, **NP**–complete problem is not in $\mathbf{PMC}_{\mathcal{T}}$.

On efficiency of P systems with active membranes

- **Proposition 3:** A deterministic P system with active membranes but without membrane division can be simulated by a DTM with a polynomial slowdown.

  Theorem: P $=$ **PMC**$_{\mathcal{NAM}}$.

- Efficient solutions to **NP**–complete problems in $\mathcal{AM}(-ne)$:
  - **NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{AM}(-ne)}$ (Sevilla team 2003, A. Alhazov, C. Martín and L. Pan, 2004).

A borderline of the efficiency

- (elementary) division rules in $\mathcal{AM}(-ne)$

On efficiency of P systems with active membranes

- **Proposition 3:** A deterministic P system with active membranes but without membrane division can be simulated by a DTM with a polynomial slowdown.

  Theorem: P $=$ **PMC**$_{\mathcal{NAM}}$.

- Efficient solutions to **NP**–complete problems in $\mathcal{AM}(-ne)$:
  - **NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{AM}(-ne)}$ (Sevilla team 2003, A. Alhazov, C. Martín and L. Pan, 2004).

A borderline of the efficiency

- (elementary) division rules in $\mathcal{AM}(-ne)$

On efficiency of P systems with active membranes

- **Proposition 3:** A deterministic P system with active membranes but without membrane division can be simulated by a DTM with a polynomial slowdown.

  Theorem: P $=$ **PMC**$_{\mathcal{NAM}}$.

- Efficient solutions to **NP**–complete problems in $\mathcal{AM}(-ne)$:
  - **NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{AM}(-ne)}$ (Sevilla team 2003, A. Alhazov, C. Martín and L. Pan, 2004).

A borderline of the efficiency

- (elementary) division rules in $\mathcal{AM}(-ne)$

On efficiency of P systems with active membranes

- **Characterization: PSPACE $=$ PMC**$_{\mathcal{AM}(+ne)}$

  (A.E. Porreca, G. Mauri and C. Zandron, 2006, 2008).

- **Conclusion:** $\mathcal{AM}$ is too powerful from the complexity point of view.

Polarizationless P systems with active membranes

- $\Pi = (\Gamma, \Sigma, H, \mu, \mathcal{M}_1, \dots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$:

  (a) $[\, a \to u \,]_h$ (object evolution rules).

  (b) $a [\ \ ]_h \to [\, b \,]_h$ (send–in communication rules).

  (c) $[\, a \,]_h \to [\ \ ]_h\, b$ (send–out communication rules).

  (d) $[\, a \,]_h \to b$ (dissolution rules).

  (e) $[\, a \,]_h \to [\, b \,]_h\, [\, c \,]_h$ (division rules for elementary membranes).

  (f) $[\, [\,]_{h_1}\ [\,]_{h_2}\,]_h \to [\, [\,]_{h_1}\,]_h\, [\, [\,]_{h_2}]_h$ (division rules for non–elementary membranes).

- The sets $\mathcal{NAM}^0$, $\mathcal{AM}^0(\alpha, \beta)$,where $\alpha \in \{-d, +d\}$ and $\beta \in \{-ne, +ne\}$.

A Păun's conjecture

At the beginning of 2005, Gh. Păun (problem **F** from [1]) 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.*

The so–called Păun's conjecture can be formally formulated:

$$\mathbf{P} = \mathbf{PMC}_{\mathcal{AM}^0(+d,-ne)}$$

[1] Gh. Păun: Further twenty six open problems in membrane computing. *Third Brainstorming Week on Membrane Computing* (M.A. Gutiérrez et al. eds.), Fénix Editora, Sevilla, 2005, pp. 249–262.

Partial answers

AFFIRMATIVE

- Non efficiency of $\mathcal{AM}^0(-d, +ne)$

Theorem: P $=$ PMC$_{\mathcal{AM}^0(-d,+ne)}$
(Sevilla team, 2006).

NEGATIVE

- Efficiency of $\mathcal{AM}^0(+d, +ne)$:

Theorem: PSPACE $\subseteq$ PMC$_{\mathcal{AM}^0(+d,+ne)}$
(A. Alhazov, Pérez-Jiménez, 2007).

- The notion of dependency graph.

A borderline of the efficiency

- dissolution rules in $\mathcal{AM}^0(+ne)$.

Partial answers

AFFIRMATIVE

- Non efficiency of $\mathcal{AM}^0(-d, +ne)$

Theorem: P $=$ PMC$_{\mathcal{AM}^0\,(-d,+ne)}$

(Sevilla team, 2006).

NEGATIVE

- Efficiency of $\mathcal{AM}^0\,(+d, +ne)$:

Theorem: PSPACE $\subseteq$ PMC$_{\mathcal{AM}^0\,(+d,+ne)}$

(A. Alhazov, Pérez-Jiménez, 2007).

- The notion of dependency graph.

A borderline of the efficiency

- dissolution rules in $\mathcal{AM}^0(+ne)$.

Partial answers

AFFIRMATIVE

- Non efficiency of $\mathcal{AM}^0(-d,+ne)$

Theorem: P $=$ PMC$_{\mathcal{AM}^0(-d,+ne)}$

(Sevilla team, 2006).

NEGATIVE

- Efficiency of $\mathcal{AM}^0\ (+d,+ne)$:

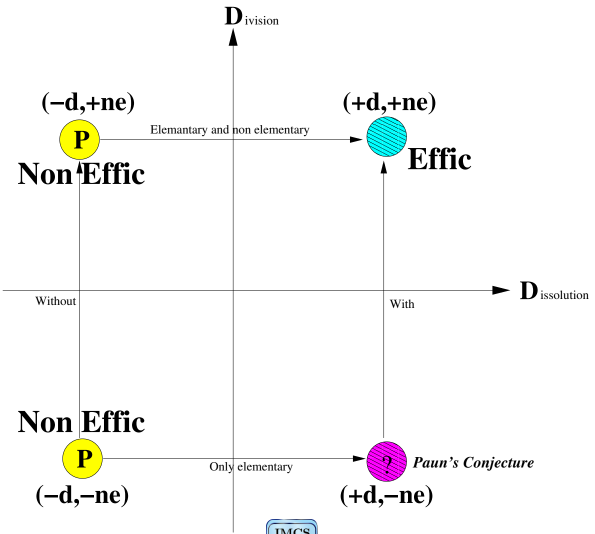
Theorem: PSPACE $\subseteq$ PMC$_{\mathcal{AM}^0(+d,+ne)}$

(A. Alhazov, Pérez-Jiménez, 2007).

- The notion of dependency graph.

A borderline of the efficiency

- dissolution rules in $\mathcal{AM}^0(+ne)$.


Division
(−d,+ne)
P
Non Effic
(+d,+ne)
Effic
Elemantary and non elementary
Dissolution
Without
With
Non Effic
P
(−d,−ne)
Only elementary
?
Paun's Conjecture
(+d,−ne)

On efficiency of tissue P systems

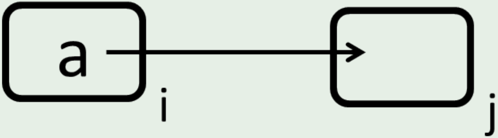
Only communication

$$\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$$

- tissue P systems with only communication rules can be efficiently simulated by basic transition P systems

Relevance of the rules' length

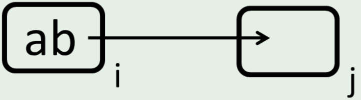
Length 1 (only symport)



$$(i, a/\lambda, j)$$

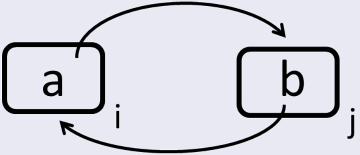
Relevance of the rules' length

Length 2 (symport)



$(i, ab/\lambda, j)$

Length 2 (antiport)



$(i, a/b, j)$

Borderlines of tractability (rules length)

Division

$\star$**NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{TDC}(2)}$
$\star$**P** $=$ **PMC**$_{\mathcal{TDC}(1)}$

AE Porreca, N Murphy, MJ Pérez.
An Optimal Frontier of the Efficiency of Tissue P Systems with Cell Division.
10thBWMC 2012 (vol.II), 141-166.

R Gutiérrez, MJ Pérez, M Rius.
Characterizing tractability by tissue-like P systems.
LNCS 5957 (2010), 289-300.

Borderlines of tractability (rules length)

Separation

$\star$**NP** $\cup$ **co-NP** $\subseteq$ **PMC**$_{\mathcal{TSC}(3)}$
$\star$**P** $=$ **PMC**$_{\mathcal{TSC}(2)}$

MJ Pérez, P. Sosík.
Improving the efficiency of tissue P systems with cell separation.
10thBWMC 2012 (vol.II), 105-140.

L Pan, MJ Pérez, A Riscos, M Rius.
New frontiers of the efficiency in tissue P systems.
Pre-proceedings ACMC 2012, pp. 61-73.

Tissue P systems without environment

- Tissue-like P systems: $\Pi = (\Gamma, \Sigma, \mathcal{E}, \mathcal{M}_1, \ldots, \mathcal{M}_q, \mathcal{R}, i_{in}, i_{out})$
  - The objects of $\mathcal{E}$ initially appear located in the environment in an arbitrary number of copies.

- Tissue-like P systems without environment: $\mathcal{E} = \emptyset$.

- The classes $\widehat{\mathcal{TC}}$, $\widehat{\mathcal{TDC}}$, $\widehat{\mathcal{TSC}}$, and $\widehat{\mathcal{TC}(k)}$, $\widehat{\mathcal{TDC}(k)}$, $\widehat{\mathcal{TSC}(k)}$, for $k \geq 1$.

Division makes environment irrelevant

Theorem

$$\forall k \in \mathbb{N} \ \left(\mathbf{PMC}_{\mathcal{TDC}(k+1)} = \mathbf{PMC}_{\widehat{\mathcal{TDC}}(k+1)} \right)$$

MJ Pérez, A Riscos, M Rius, FJ Romero.
A polynomial alternative to unbounded environment for tissue P systems with cell division.
IJCM, 90 (4) (2013)

- $\mathbf{P} = \mathbf{PMC}_{\widehat{\mathcal{TDC}}(1)}$.
- $\mathbf{NP} \cup \mathbf{co}-\mathbf{NP} \subseteq \mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$.

A borderline of the efficiency

- Length of communication rules in $\widehat{\mathcal{TD}}$.

Borderlines of tractability (environment)

Separation

$$\mathbf{P} = \mathbf{PMC}_{\widehat{\mathcal{TSC}}}$$

LF Macías, MJ Pérez, A Riscos, M Rius.
The efficiency of tissue P systems with cell separation relies on the environment.
Proceedings CMC 2012, pp. 277-290.

- $\mathbf{P} = \mathbf{PMC}_{\widehat{\mathcal{TSC}}(3)}$.
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$.

A borderline of the efficiency

- Environment in $\mathcal{TSC}(3)$.

Conclusions: "New" Frontiers of the efficiency

- Kind of rules:
  - Division rules in $\mathcal{AM}$.
  - Division rules in $\mathcal{TC}$.
  - Dissolution rules in $\mathcal{AM}^0(+ne)$.
- The length of communication rules:
  - Passing from 1 to 2 in $\mathcal{TD}$.
  - Passing from 1 to 2 in $\widehat{\mathcal{TD}}$.
  - Passing from 2 to 3 in $\mathcal{TS}$.
- The environment:
  - In the framework $\mathcal{TSC}(3)$.

Each of them provides a new way to attack the P versus NP problem.

Conclusions: “New” Frontiers of the efficiency

- Kind of rules:
  - Division rules in $\mathcal{AM}$.
  - Division rules in $\mathcal{TC}$.
  - Dissolution rules in $\mathcal{AM}^0(+ne)$.
- The length of communication rules:
  - Passing from 1 to 2 in $\mathcal{TD}$.
  - Passing from 1 to 2 in $\widehat{\mathcal{TD}}$.
  - Passing from 2 to 3 in $\mathcal{TS}$.
- The environment:
  - In the framework $\mathcal{TSC}(3)$.

Each of them provides a new way to attack the P versus NP problem.

Conclusions: "New" Frontiers of the efficiency

- Kind of rules:
  - Division rules in $\mathcal{AM}$.
  - Division rules in $\mathcal{TC}$.
  - Dissolution rules in $\mathcal{AM}^0(+ne)$.
- The length of communication rules:
  - Passing from 1 to 2 in $\mathcal{TD}$.
  - Passing from 1 to 2 in $\widehat{\mathcal{TD}}$.
  - Passing from 2 to 3 in $\mathcal{TS}$.
- The environment:
  - In the framework $\mathcal{TSC}(3)$.

Each of them provides a new way to attack the P versus NP problem.

Conclusions: “New” Frontiers of the efficiency

- Kind of rules:
  - Division rules in $\mathcal{AM}$.
  - Division rules in $\mathcal{TC}$.
  - Dissolution rules in $\mathcal{AM}^0(+ne)$.
- The length of communication rules:
  - Passing from 1 to 2 in $\mathcal{TD}$.
  - Passing from 1 to 2 in $\widehat{\mathcal{TD}}$.
  - Passing from 2 to 3 in $\mathcal{TS}$.
- The environment:
  - In the framework $\mathcal{TSC}(3)$.

Each of them provides a new way to attack the P versus NP problem.

Borderlines of tractability

- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$ (Sevilla team, 2009).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (Sevilla team, 2010).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TSC}(2)}$ (L. Pan, Pérez-Jiménez, A. Riscos, M. Rius, 2012).
- $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{\mathcal{TDC}(2)}$ (A. Porreca, N. Murphy, Pérez-Jiménez, 2012).
- $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$ (P. Sosík, Pérez-Jiménez, 2012).

Borderlines of the efficiency

- division rules in the framework of $\mathcal{TC}$.
- length of communication rules in the framework of $\mathcal{TD}$: passing from 1 to 2 amounts to passing from non–efficiency to efficiency.
- length of communication rules in the framework of $\mathcal{TS}$: passing from 2 to 3 amounts to passing from non–efficiency to efficiency.

Borderlines of tractability

- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$ (Sevilla team, 2009).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (Sevilla team, 2010).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TSC}(2)}$ (L. Pan, Pérez-Jiménez, A. Riscos, M. Rius, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TDC}(2)}$ (A. Porreca, N. Murphy, Pérez-Jiménez, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$ (P. Sosík, Pérez-Jiménez, 2012).

Borderlines of the efficiency

- division rules in the framework of $\mathcal{TC}$.
- length of communication rules in the framework of $\mathcal{TD}$: passing from 1 to 2 amounts to passing from non–efficiency to efficiency.
- length of communication rules in the framework of $\mathcal{TS}$: passing from 2 to 3 amounts to passing from non–efficiency to efficiency.

Borderlines of tractability

- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$ (Sevilla team, 2009).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (Sevilla team, 2010).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TSC}(2)}$ (L. Pan, Pérez-Jiménez, A. Riscos, M. Rius, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TDC}(2)}$ (A. Porreca, N. Murphy, Pérez-Jiménez, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$ (P. Sosík, Pérez-Jiménez, 2012).

Borderlines of the efficiency

- division rules in the framework of $\mathcal{TC}$.
- length of communication rules in the framework of $\mathcal{TD}$: passing from 1 to 2 amounts to passing from non–efficiency to efficiency.
- length of communication rules in the framework of $\mathcal{TS}$: passing from 2 to 3 amounts to passing from non–efficiency to efficiency.

Borderlines of tractability

- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$ (Sevilla team, 2009).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (Sevilla team, 2010).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TSC}(2)}$ (L. Pan, Pérez-Jiménez, A. Riscos, M. Rius, 2012).
- $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{\mathcal{TDC}(2)}$ (A. Porreca, N. Murphy, Pérez-Jiménez, 2012).
- $\mathbf{NP} \cup \mathbf{co-NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$ (P. Sosík, Pérez-Jiménez, 2012).

Borderlines of the efficiency

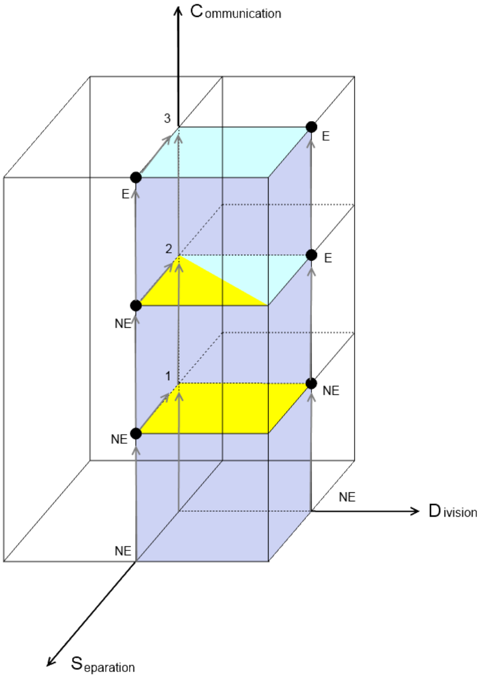
- division rules in the framework of $\mathcal{TC}$.
- length of communication rules in the framework of $\mathcal{TD}$: passing from 1 to 2 amounts to passing from non–efficiency to efficiency.
- length of communication rules in the framework of $\mathcal{TS}$: passing from 2 to 3 amounts to passing from non–efficiency to efficiency.

Borderlines of tractability

- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}}$ (Sevilla team, 2009).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (Sevilla team, 2010).
- $\mathbf{P} = \mathbf{PMC}_{\mathcal{TSC}(2)}$ (L. Pan, Pérez-Jiménez, A. Riscos, M. Rius, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TDC}(2)}$ (A. Porreca, N. Murphy, Pérez-Jiménez, 2012).
- $\mathbf{NP} \cup \mathbf{co} - \mathbf{NP} \subseteq \mathbf{PMC}_{\mathcal{TSC}(3)}$ (P. Sosík, Pérez-Jiménez, 2012).

Borderlines of the efficiency

- division rules in the framework of $\mathcal{TC}$.
- length of communication rules in the framework of $\mathcal{TD}$: passing from 1 to 2 amounts to passing from non–efficiency to efficiency.
- length of communication rules in the framework of $\mathcal{TS}$: passing from 2 to 3 amounts to passing from non–efficiency to efficiency.


Communication
3
E
E
2
E
NE
1
NE
NE
NE
Division
NE
Separation

Final Comments (environment in tissue)

Division: No influence of the environment

$\star$**NP** $\cup$ **co-NP** $\subseteq$ $\mathbf{PMC}_{\mathcal{TDC}(2)} = \mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$
$\star$**P** $= \mathbf{PMC}_{\mathcal{TDC}(1)} = \mathbf{PMC}_{\widehat{\mathcal{TDC}}(1)}$

Separation: Efficiency relies on the environment

$\star$**NP** $\cup$ **co-NP** $\subseteq$ $\mathbf{PMC}_{\mathcal{TSC}(3)}$
$\star$**P** $= \mathbf{PMC}_{\widehat{\mathcal{TSC}}(3)}$

Separation vs. Division

$\star$**NP** $\cup$ **co-NP** $\subseteq$ $\mathbf{PMC}_{\mathcal{TDC}(2)} = \mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$
$\star$**P** $= \mathbf{PMC}_{\mathcal{TSC}(2)} = \mathbf{PMC}_{\widehat{\mathcal{TSC}}(2)}$

Thank you!

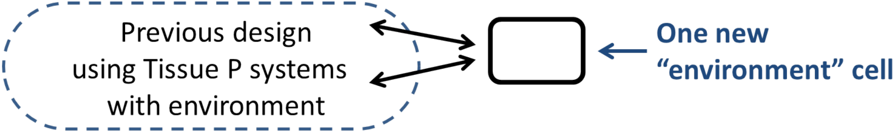
4. Strategies
    - Algorithmic (Dependency graph)
    - Simulation

Borderlines of tractability (environment)

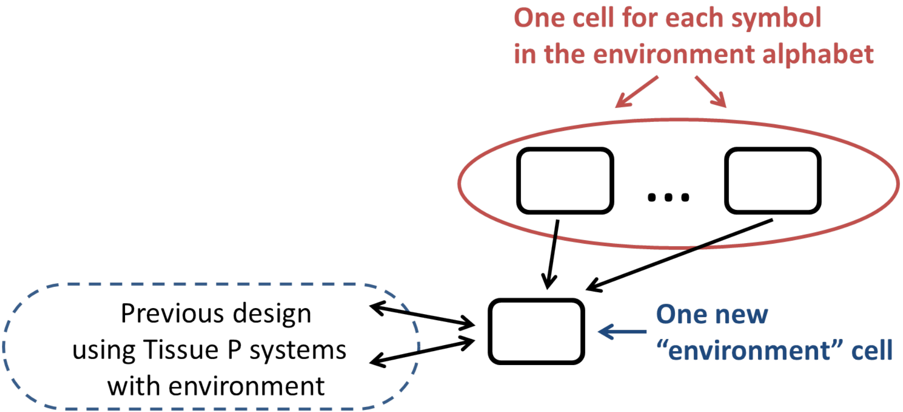
HAM−CYCLE $\in$ $\mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$

Previous design
using Tissue P systems
with environment

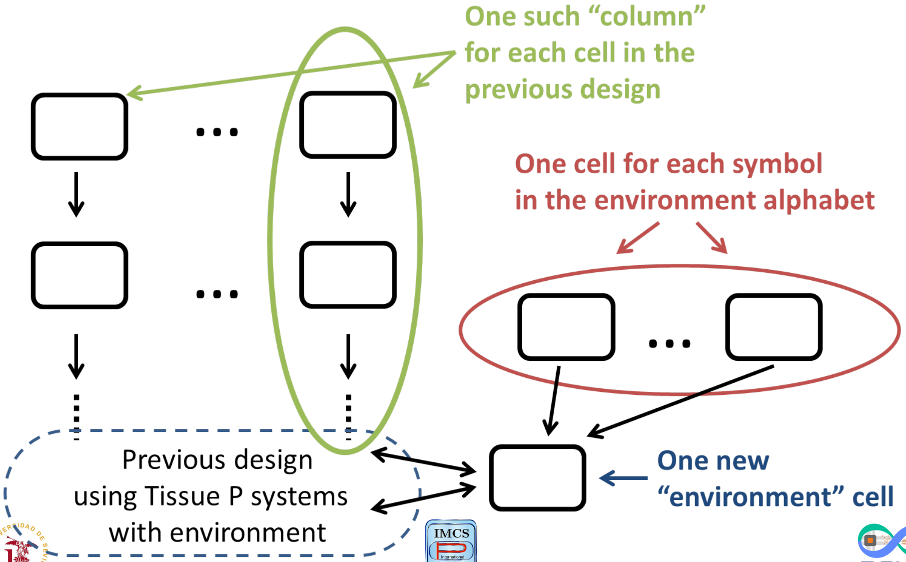
HAM−CYCLE ∈ $\mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$



HAM−CYCLE $\in$ $\mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$



HAM−CYCLE $\in \mathbf{PMC}_{\widehat{\mathcal{TDC}}(2)}$



P = PMC$_{\widehat{\mathcal{T}SC}}$ (Sketch of the proof)

- Key idea: **only one computation** is simulated
  - ⋆ sufficient because of **confluence** condition

Selection stage

1. Loop over the (ordered) set of communication rules
   - apply maximally
2. Loop over the (ordered) set of separation rules
   - apply if possible

- cells with same label get unique tags (strings over 0's and 1's)
- used cells are tracked to avoid incorrect application of separation rules

$\mathbf{P} = \mathbf{PMC}_{\widehat{\mathcal{TSC}}}$ (Sketch of the proof)

- Key idea: **only one computation** is simulated
  - ⋆ sufficient because of **confluence** condition

Selection stage

1. Loop over the (ordered) set of communication rules
   - apply maximally
2. Loop over the (ordered) set of separation rules
   - apply if possible

Execution stage

- Loop over the applied rules
  - update objects from RHS of communication rules
  - create cells (new tags) for separation rules, distributing objects

Pseudocode (Selection stage)

Input: A configuration $\mathcal{C}_t$ of Π at instant t

for $r \equiv (i, u/v, j) \in R_C$ (ordered) **do**
  for each pair $(i, \sigma_i), (j, \sigma_j)$ of $\mathcal{C}'_t$ (ordered) **do**
    $n_r \leftarrow$ max number of applicablications
    if $n_r > 0$ **then**
      $\mathcal{C}'_t \leftarrow \mathcal{C}'_t \setminus n_r \cdot LHS(r, (i, \sigma_i), (j, \sigma_j))$
      $A \leftarrow A \cup \{(r, n_r, (i, \sigma_i), (j, \sigma_j))\}$
      $B \leftarrow B \cup \{(i, \sigma_i), (j, \sigma_j)\}$

for $r \equiv [a]_i \to [\Gamma_0]_i [\Gamma_1]_i \in R_S$ (ordered) **do**
  for each $(a, i, \sigma_i) \in \mathcal{C}'_t$ (ordered), s.t. $(i, \sigma_i) \notin B$ **do**
    $\mathcal{C}'_t \leftarrow \mathcal{C}'_t \setminus \{(a, i, \sigma_i)\}$
    $A \leftarrow A \cup \{(r, 1, (i, \sigma_i))\}$
    $B \leftarrow B \cup \{(i, \sigma_i)\}$

Pseudocode (Execution stage)

Input: The output $\mathcal{C}'_t$ and A of the selection stage

for each $(r, n_r, (i,\sigma_i), (j,\sigma_j)) \in A$ **do**
$\quad \mathcal{C}'_t \leftarrow \mathcal{C}'_t + n_r \cdot RHS(r, (i,\sigma_i), (j,\sigma_j))$

for each $(r, 1, (i,\sigma_i)) \in A$ **do**
$\quad \mathcal{C}'_t \leftarrow \mathcal{C}'_t + \{(\lambda, i, \sigma_i)/\sigma_i 0\} + \{(\lambda, i, \sigma_i 1)\}$
$\quad$ **for** each $(x, i, \sigma_i) \in \mathcal{C}'_t$ (ordered) **do**
$\quad\quad$ **if** $x \in \Gamma_0$ **then** $\mathcal{C}'_t \leftarrow \mathcal{C}'_t + \{(x, i, \sigma_i)/\sigma_i 0\}$
$\quad\quad\quad$ **else** $\mathcal{C}'_t \leftarrow \mathcal{C}'_t + \{(x, i, \sigma_i)/\sigma_i 1\}$

This algorithm is **deterministic** and works in **polynomial time**.

Thank you!