From NP-completeness to DP-completeness: A membrane computing perspective

Mario J. Pérez-Jiménez

Research Group on Natural Computing
Dpt. Computer Science and Artificial Intelligence
University of Seville, Spain
Academia Europaea (The Academy of Europe)

www.cs.us.es/~marper marper@us.es

18th Brainstorming Week on Membrane Computing
Sevilla, Spain, February 4, 2020

Presumably efficient computing models

Efficient computing model: ability to provide polynomial-time solutions for **intractable** problems (with respect to the time).

Presumably efficient computing models

Efficient computing model: ability to provide polynomial-time solutions for **intractable** problems (with respect to the time).

Presumably efficient computing model: ability to provide polynomial-time solutions for **NP-complete** problems.

Presumably efficient computing models

Efficient computing model: ability to provide polynomial-time solutions for **intractable** problems (with respect to the time).

Presumably efficient computing model: ability to provide polynomial-time solutions for **NP-complete** problems.

- If $\mathbf{P} \neq \mathbf{NP}$ then every presumably efficient computing model is an efficient one.

A lower bound for $\mathbf{PMC}_{\mathcal{R}}$

$\mathcal{R}$: class of recognizer membrane systems

$\mathbf{PMC}_{\mathcal{R}}$: time complexity class of problems solvable by families from $\mathcal{R}$.

A lower bound for $\mathbf{PMC}_{\mathcal{R}}$

$\mathcal{R}$: class of recognizer membrane systems

$\mathbf{PMC}_{\mathcal{R}}$: time complexity class of problems solvable by families from $\mathcal{R}$.

- $\star$ $\mathbf{PMC}_{\mathcal{R}}$ is closed under complement and under polynomial-time reduction.

A lower bound for $\mathbf{PMC}_{\mathcal{R}}$

$\mathcal{R}$: class of recognizer membrane systems

$\mathbf{PMC}_{\mathcal{R}}$: time complexity class of problems solvable by families from $\mathcal{R}$.

- $\star$ $\mathbf{PMC}_{\mathcal{R}}$ is closed under complement and under polynomial-time reduction.

If $\mathcal{R}$ is a class of **presumably efficient** recognizer membrane systems then:

A lower bound for $\mathbf{PMC}_{\mathcal{R}}$

$\mathcal{R}$: class of recognizer membrane systems

$\mathbf{PMC}_{\mathcal{R}}$: time complexity class of problems solvable by families from $\mathcal{R}$.

- ⋆ $\mathbf{PMC}_{\mathcal{R}}$ is closed under complement and under polynomial-time reduction.

If $\mathcal{R}$ is a class of **presumably efficient** recognizer membrane systems then:

- ∗ $\mathbf{NP} \cup \mathbf{co\text{-}NP} \subseteq \mathbf{PMC}_{\mathcal{R}}$.

A lower bound for $\mathbf{PMC}_{\mathcal{R}}$

$\mathcal{R}$: class of recognizer membrane systems

$\mathbf{PMC}_{\mathcal{R}}$: time complexity class of problems solvable by families from $\mathcal{R}$.

- $\star$ $\mathbf{PMC}_{\mathcal{R}}$ is closed under complement and under polynomial-time reduction.

If $\mathcal{R}$ is a class of **presumably efficient** recognizer membrane systems then:

- $*$ $\mathbf{NP} \cup \mathbf{co\text{-}NP} \subseteq \mathbf{PMC}_{\mathcal{R}}$.

This lower bound for $\mathbf{PMC}_{\mathcal{R}}$ can be **improved**.

The complexity class DP (difference class)

Introduced by C.H. Papadimitriou and M. Yannakakis[1]

- ⋆ A language L is in the class **DP** iff there are two languages L_1 and L_2 such that $L_1, L_2 \in \mathbf{NP}$ and $L = L_1 \setminus L_2$.

Then, $L \in \mathbf{DP}$ iff there are $L_1 \in \mathbf{NP}$ and $L_2 \in \textbf{co-NP}$ such that $L = L_1 \cap L_2$.

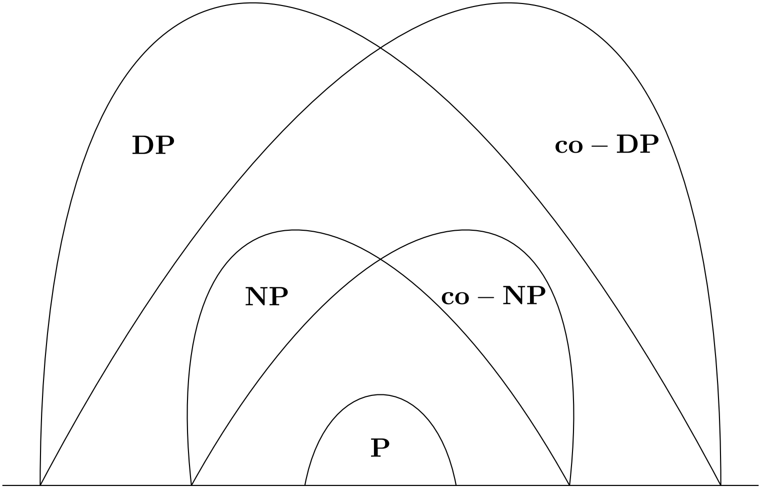
Class **DP**:

- ∗ lies between the first two levels of the **polynomial hierarchy**.
- ∗ is the second level in the **Boolean hierarchy**.

$\mathbf{NP} \subseteq \mathbf{DP} \subseteq \mathbf{P}^{\mathbf{NP}}$.

$\mathbf{NP} \cup \textbf{co-NP} \subseteq \mathbf{DP} \cap \textbf{co-DP}$.

[1] C.H. Papadimitriou, M. Yannakis. The complexity of facets (and some facets of complexity). **Proceedings of the 24th ACM Symposium on the Theory of Computing**, 1982, pp. 229-234.


DP
co – DP
NP
co – NP
P

Product of two decision problems

Let $X_1 = (I_{X_1}, \theta_{X_1})$ and $X_2 = (I_{X_2}, \theta_{X_2})$ be two decision problems.

Product of two decision problems

Let $X_1 = (I_{X_1}, \theta_{X_1})$ and $X_2 = (I_{X_2}, \theta_{X_2})$ be two decision problems.

The **product problem** $X_1 \otimes X_2 = (I_{X_1 \otimes X_2}, \theta_{X_1 \otimes X_2})$ is defined as follows:

Product of two decision problems

Let $X_1 = (I_{X_1}, \theta_{X_1})$ and $X_2 = (I_{X_2}, \theta_{X_2})$ be two decision problems.

The **product problem** $X_1 \otimes X_2 = (I_{X_1 \otimes X_2}, \theta_{X_1 \otimes X_2})$ is defined as follows:

- $I_{X_1 \otimes X_2} = I_{X_1} \times I_{X_2}$.
- $\theta_{X_1 \otimes X_2}(u_1, u_2) = 1 \Leftrightarrow \theta_{X_1}(u_1) = 1 \land \theta_{X_2}(u_2) = 1$.

Product of two decision problems

Let $X_1 = (I_{X_1}, \theta_{X_1})$ and $X_2 = (I_{X_2}, \theta_{X_2})$ be two decision problems.

The **product problem** $X_1 \otimes X_2 = (I_{X_1 \otimes X_2}, \theta_{X_1 \otimes X_2})$ is defined as follows:

- $I_{X_1 \otimes X_2} = I_{X_1} \times I_{X_2}$.
- $\theta_{X_1 \otimes X_2}(u_1, u_2) = 1 \Leftrightarrow \theta_{X_1}(u_1) = 1 \wedge \theta_{X_2}(u_2) = 1$.

Proposition: If X_1 is an **NP** complete problem and X_2 is a **co-NP** complete problem then $X_1 \otimes X_2$ is a **DP** complete problem.

Product of two decision problems

Let $X_1 = (I_{X_1}, \theta_{X_1})$ and $X_2 = (I_{X_2}, \theta_{X_2})$ be two decision problems.

The **product problem** $X_1 \otimes X_2 = (I_{X_1 \otimes X_2}, \theta_{X_1 \otimes X_2})$ is defined as follows:

- $I_{X_1 \otimes X_2} = I_{X_1} \times I_{X_2}$.
- $\theta_{X_1 \otimes X_2}(u_1, u_2) = 1 \Leftrightarrow \theta_{X_1}(u_1) = 1 \wedge \theta_{X_2}(u_2) = 1$.

Proposition: If X_1 is an **NP** complete problem and X_2 is a **co-NP** complete problem then $X_1 \otimes X_2$ is a **DP** complete problem.

Corollary: If X is an **NP** complete problem, then $X \otimes \overline{X}$ is a **DP** complete problem.

Main result

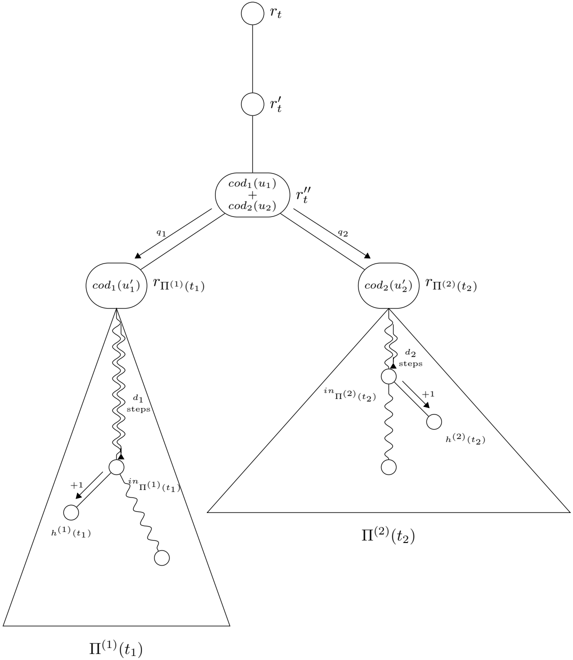
Let $\mathcal{R}$ be a computing model of recognizer non-cooperative P systems allowing **dissolution**, **object evolution** *and* **communication** *rules.*

- *If $X_1 \in \mathbf{PMC}_{\mathcal{R}}$ and $X_2 \in \mathbf{PMC}_{\mathcal{R}}$ then $X_1 \otimes X_2 \in \mathbf{PMC}_{\mathcal{R}}$.*

Sketch:
For $i = 1, 2$,

- Let $\mathbf{\Pi}^{(i)} = \{\Pi^{(i)}(t) \mid t \in \mathbf{N}\}$ a family of systems from $\mathcal{R}$ solving X_i in polynomial-time.
- Let (cod_i, s_i) be a polynomial encoding from X_i into $\mathbf{\Pi}^{(i)}$.

A family $\mathbf{\Pi} = \{\Pi(t) \mid t \in \mathbf{N}\}$ of membrane systems from $\mathcal{R}$ will be defined from $\mathbf{\Pi}^{(1)}$ and $\mathbf{\Pi}^{(2)}$, providing a uniform and polynomial-time solution to $X_1 \otimes X_2$.


r_t
r'_t
$cod_1(u_1) + cod_2(u_2)$
r''_t
q_1
q_2
$cod_1(u'_1)$
$r_{\Pi^{(1)}(t_1)}$
$cod_2(u'_2)$
$r_{\Pi^{(2)}(t_2)}$
d_1 steps
$+1$
$in_{\Pi^{(1)}(t_1)}$
$h^{(1)}(t_1)$
d_2 steps
$in_{\Pi^{(2)}(t_2)}$
$+1$
$h^{(2)}(t_2)$
$\Pi^{(2)}(t_2)$
$\Pi^{(1)}(t_1)$

A thinner lower bound for $\mathrm{PMC}_{\mathcal{R}}$

Let $\mathcal{R}$ be a **presumably efficient** computing model of recognizer P systems allowing **dissolution**, **object evolution** and **communication** rules.

A **thinner** lower bound for $\mathbf{PMC}_{\mathcal{R}}$

Let $\mathcal{R}$ be a **presumably efficient** computing model of recognizer P systems allowing **dissolution**, **object evolution** and **communication** rules.

– Then, $\mathbf{DP} \cup \mathbf{co\text{-}DP} \subseteq \mathbf{PMC}_{\mathcal{R}}$.

A thinner lower bound for $\mathbf{PMC}_{\mathcal{R}}$

Let $\mathcal{R}$ be a **presumably efficient** computing model of recognizer P systems allowing **dissolution**, **object evolution** and **communication** rules.

– Then, $\mathbf{DP} \cup \mathbf{co\text{-}DP} \subseteq \mathbf{PMC}_{\mathcal{R}}$.

Proof: If X is an **NP**-complete problem such that $X \in \mathbf{PMC}_{\mathcal{R}}$, then $X \otimes \overline{X}$ is a **DP**-complete problem such that $X \otimes \overline{X} \in \mathbf{PMC}_{\mathcal{R}}$.

THANK YOU

FOR YOUR ATTENTION!