An efficient simulation of polynomial-space Turing machines by P systems with active membranes

Andrea Valsecchi Antonio E. Porreca Alberto Leporati
Giancarlo Mauri Claudio Zandron

Dipartimento di Informatica, Sistemistica e Comunicazione
Università degli Studi di Milano-Bicocca, Italy

Tenth Workshop on Membrane Computing

Talk outline

Definitions

- P systems with active membranes
- Space complexity for P systems

Simulating Turing machines

- Representing a TM as a P system
- Computing via communication rules

Space hierarchies

- The space hierarchy theorem for TMs
- A pseudo-hierarchy for P systems

Conclusions

- Future directions

P systems with active membranes

Definition

A P system with active membranes is a structure

$$\Pi = (\Gamma, \Lambda, \mu, w_1, \dots, w_m, R)$$

where

- Γ is the alphabet of objects
- Λ the set of labels for the membranes
- μ is a (cell-like) structure of m membranes
  labelled by elements of Λ
  and having polarization $\{+, 0, -\}$ initially neutral
- $w_1, \dots, w_m$ are the multisets of objects
  placed inside the membranes
- R is a finite set of rules

Standard kinds of rules

- Object evolution: $[a \to w]_h^\alpha$
- Communication: $a\ [\ ]_h^\alpha \to [b]_h^\beta$
- Communication: $[a]_h^\alpha \to [\ ]_h^\beta\ b$
- Dissolution: $[a]_h^\alpha \to b$
- Elementary division: $[a]_h^\alpha \to [b]_h^\beta [c]_h^\gamma$
- Non-elementary division:

$$\left[[\ ]_{h_1}^{+} \cdots [\ ]_{h_k}^{+}\ [\ ]_{h_{k+1}}^{-} \cdots [\ ]_{h_n}^{-} \right]_h^\alpha$$
$$\downarrow$$
$$\left[[\ ]_{h_1}^{\delta} \cdots [\ ]_{h_k}^{\delta} \right]_h^\beta \quad \left[[\ ]_{h_{k+1}}^{\epsilon} \cdots [\ ]_{h_n}^{\epsilon} \right]_h^\gamma$$

- The rules are chosen nondeterministically in a maximally parallel way

Semi-uniform recogniser P systems

Definition

A family of P systems $\mathbf{\Pi} = \{\Pi_x : x \in \Sigma^\star\}$ decides $L \subseteq \Sigma^\star$ in a confluent way iff for each $x \in \Sigma^\star$

- Exactly one object *yes* (acceptance) or *no* (rejection) is sent out from the skin during each computation of Π_x
- All computations of Π_x agree on the result
- The output of Π_x is *yes* iff $x \in L$
- The mapping $x \mapsto \Pi_x$ can be computed in polynomial time by a deterministic Turing machine

Space complexity for P systems

Definition

- The size of a P system configuration is the sum of the number of objects and the number of membranes
- The space required by a P system is the size of the largest configuration among all the computations

Definition

A family of P systems $\mathbf{\Pi} = \{\Pi_x : x \in \Sigma^\star\}$
operates within space bound $f \colon \mathbf{N} \to \mathbf{N}$
iff every P system Π_x requires at most $f(|x|)$ space

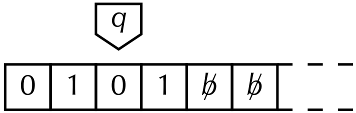
Representing the tape of a TM via membranes

- Each tape cell is represented by a membrane
- The membranes are located one inside the other (a linear chain) to preserve the ordering
- The outermost membrane corresponds to the leftmost cell
- We assume the tape alphabet is $\Gamma = \{\not{b}, 0, 1\}$
- Instead of using objects, we represent the symbol in a cell by the polarization of the corresponding membrane:

Symbol	Polarization
$\not{b}$	0
0	$-$
1	$+$

Representing head position and state of the TM

- An object is locate inside the n-th outermost membrane to indicate that the tape head is located on the n-th leftmost tape cell
- This object also represents the state of the simulated Turing machine
- Example:



$$[^{-}\ [^{+}\ [^{-}\ q\ [^{+}\ [^{0}\ [^{0} \cdots]\ ]\ ]\ ]\ ]\ ]$$

Representing the transition function of the TM

- We simulate a step of the TM via communication rules applied to the head/state object
- Change the polarization of the membrane (symbol in the cell)
- Move the object to an adjacent membrane (head movement)
- Change the object itself (state change)

Example of simulated step

- If the tape of the TM is $01\underline{0}1$, the state is q_1, and $\delta(0, q_1) = (1, q_2, \rightarrow)$

$$[^- \; [^+ \; [^- \; q_1 \; [^+ \cdots]_h]_h]_h]_h$$

- First we apply $[q_1]_h^- \rightarrow [\,]^+ \; q_2''$

$$[^- \; [^+ \; q_2'' \; [^+ \; [^+ \cdots]_h]_h]_h]_h$$

- Then we move the object to the right via $q_2'' \; [\,]_h^+ \rightarrow [q_2']_h^+$

$$[^- \; [^+ \; [^+ \; q_2' \; [^+ \cdots]_h]_h]_h]_h$$

- Lastly we move it to its final position via $q_2' \; [\,]_h^\gamma \rightarrow [q]_h^\gamma$

$$[^- \; [^+ \; [^+ \; [^+ \; q_2 \; \cdots]_h]_h]_h]_h$$

Setting up the initial configuration

- Let M be a TM using space $f(n)$ and $x = x_1 \cdots x_n$ its input
- The initial configuration of the P system simulating M on x is

$$\Big[[\hat{q}_0]^0_w \overbrace{\underbrace{[x_1[x_2 \cdots [x_n}_{n \text{ membranes}}[\cdots[\;}^{f(n) \text{ membranes}} \overbrace{]^0_h \cdots]^0_h \; \underbrace{]^0_h \cdots]^0_h]^0_h}_{n \text{ membranes}}}^{f(n) \text{ membranes}} \Big]^0_{h_0}$$

- Objects $x_1, \ldots, x_n$ are used to set up the polarizations of the n outermost tape-membranes according to the initial configuration of M
  - They are sent out by a communication rule and discarded
- Object $\hat{q}_0$ waits one step, then enters the "tape" and becomes the head/state object q_0
- Now the real simulation begins

Example of initial setup

- We simulate a TM on input $x = 0101$
- $t = 0$: initial configuration of the P system

$$\Big[[\hat{q}_0]_w^0 \ \overbrace{\underbrace{[^0 \ 0 \ [^0 \ 1 \ [^0 \ 0 \ [^0 \ 1}_{\text{input}} \ [\cdots]_h]_h]_h]_h]_h}^{\text{tape}} \Big]_{h_0}^0$$

- $t = 1$: $\hat{q}_0$ and the objects representing x have been sent out

$$\Big[[\ ]_w^0 \ \hat{q}_0 \ \# \ \overbrace{[^- \ \# \ [^+ \ \# \ [^- \ \# \ [^+}^{\text{input}=0101} \ [\cdots]_h \ ]_h]_h]_h]_h \Big]_{h_0}^0$$

- $t = 2$: the head/state objects has entered the "tape"

$$\Big[[\ ]_w^0 \ \# \ [^- \ q_0 \ \# \ [^+ \ \# \ [^- \ \# \ [^+ \ [\cdots]_h \ ]_h]_h]_h]_h \Big]_{h_0}^0$$

Halting the simulation

- As soon as the simulated machine enters
  an accepting (resp. rejecting) halting state,
  the head/state object becomes *yes* (resp. *no*)...
- ...while simultaneously being sent out
  through the whole membrane structure
  and expelled from the skin as the result
- Each step of the TM is simulated by at most three steps
  of the P system
- The P system uses $f(n) + 2$ membranes and $n + 1$ objects
- Hence it requires (asymptotically) the same amount
  of time and space as the simulated TM

Main result

Definition

A function $f\colon \mathbf{N} \to \mathbf{N}$ is time- (resp. space-) constructible
iff the mapping $1^n \mapsto 1^{f(n)}$ is TM-computable
in $O(f(n))$ time (resp. space).

Theorem

Let M be a single-tape TM halting on every input and operating in time $g(n)$ and space $f(n)$, where

- $f(n)$ *time-constructible*
- $f(n) \in \Omega(n)$ *and bounded by a polynomial*

Then there exists a semi-uniform family $\mathbf{\Pi}_M$ of P systems using only communication rules such that

- $\mathbf{\Pi}_M$ *operates in time $O(g(n))$ and space $O(f(n))$*
- $\mathbf{\Pi}_M$ *can be constructed by a TM in time $O(f(n))$*
- $\mathbf{\Pi}_M$ *and M decide the same language*

The space hierarchy theorem for TMs

Definition

$L(f)$ is the language of strings x of the form $\langle M \rangle 10^\star$
where $\langle M \rangle$ is the encoding of a single-tape TM
that rejects x using at most $f(|x|)$ space

Theorem (Space hierarchy theorem)

If $f\colon \mathbf{N} \to \mathbf{N}$ *is space-constructible and* $f(n) \in \Omega(n)$
then $L(f)$ *is TM-decidable in* $O(f(n))$ *space*
but not in $o(f(n))$ *space*

Deciding $L(f)$ via TM

Theorem

$L(f)$ is TM-decidable in $O(f(n))$ space

Proof sketch.

Let D be the following TM:

- On input $x = \langle M \rangle 10^k$, start simulating M on x
- If the computation exceeds $f(|x|)$ space, halt and reject
- If the computation time exceeds $2^{f(|x|)}$, halt and reject
- Otherwise, complete the simulation
  and return the opposite result

The TM D runs in $O(f(n))$ space □

Deciding $L(f)$ via P systems

Theorem

If f is bounded by a polynomial then $L(f)$ is P system-decidable in $O(f(n))$ space

Proof.

Change the TM D as follows:

- On input $x = \langle M \rangle 10^k$, construct a P system $\Pi_{M,x,f}$

Where $\Pi_{M,x,f}$ simulates M on x but

- Rejects if the computation exceeds $f(|x|)$ space
- Rejects if the computation exceeds $2^{f(|x|)}$ time
- Flips the result (accept/reject)

This defines a family of P systems deciding $L(f)$
and using only communication rules, which can be constructed in $O(f(n))$ time (hence it is semi-uniform) □

Modifying the simulation of TMs

- Reject if the computation exceeds $f(|x|)$ space
  - If the head/state object crosses the innermost membrane then it is sent out as a *no*
- Reject if the computation exceeds $2^{f(|x|)}$ time
  - We can construct a binary counter in $O(f(|x|))$ space using the polarizations of the membranes
  - When the counter reaches $2^{f(|x|)}$ an object is produced which makes the P system reject
- Flip the result (accept/reject)
  - When *yes* (resp. *no*) crosses the skin it is changed into *no* (resp. *yes*)

Limitations of P systems

Theorem

$L(f)$ cannot be decided by any semi-uniform family of P systems which can be constructed in $o(f(n))$ time and operating in $o(f(n)/\log f(n))$ space

Proof.

Otherwise we could simulate this family via a TM using $o(f(n))$ space[1], hence contradicting the space hierarchy theorem for TMs □

[1] A. E. Porreca, G. Mauri, C. Zandron, Complexity classes for membrane systems, *RAIRO Theoretical Informatics and Applications* 40(2), 141–162, 2006

Summarising

Theorem

Let $f\colon \mathbf{N} \to \mathbf{N}$ be a function such that

- *$f(n)$ is time-constructible*
- *$f(n) \in \Omega(n)$ and bounded by a polynomial*

Then there exists a language L such that

- *L can be decided by a semi-uniform family of P systems using only communication rules, working in $O(f(n))$ space and constructible in $O(f(n))$ time*
- *L cannot be decided by any semi-uniform family of P systems working in $o(f(n)/\log f(n))$ space and constructible in $o(f(n))$ time*

Hence there exists an infinite (pseudo-) hierarchy of languages: larger enough space bound $\Rightarrow$ harder problems become solvable

Future directions

- Provide a uniform construction for the family of P systems
- Can we avoid the inverse logarithmic factor?
- Can we prove a time hierarchy theorem?
  - We need to simulate multitape TMs
- What about simulating nondeterministic TMs?
- Can we make our constructions work also with minimal parallelism?

Thank you!