Decidability of Termination Problems for Sequential P Systems with Active Membranes

Michal Kováč

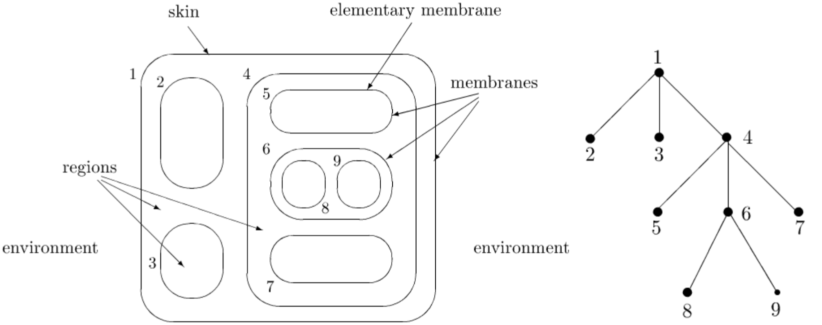
FMFI UK, Slovakia

15.5.2015

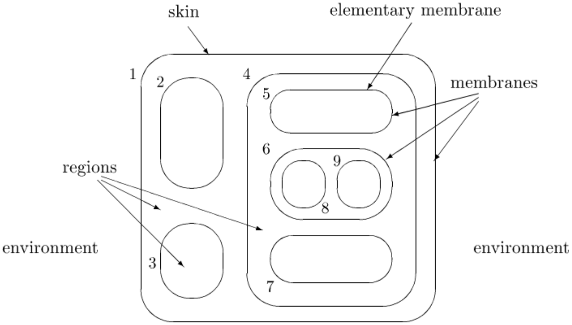
1. P systems
   - Overview
   - Computational power

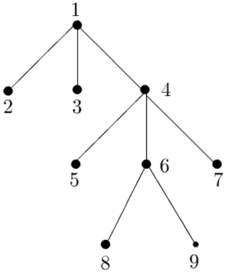
2. Termination problems
   - Halting problem
   - Termination problems in active membranes

Membrane structure



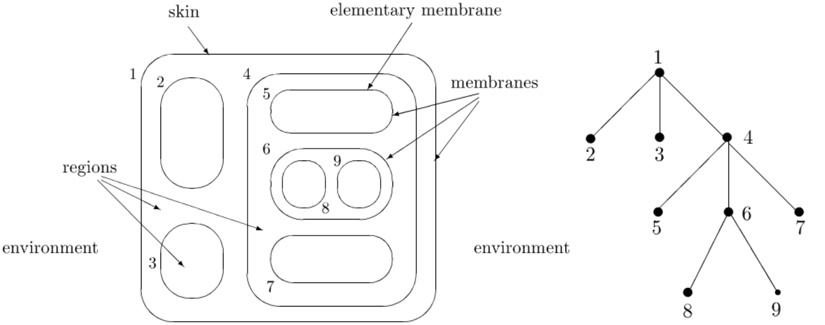
Membrane structure





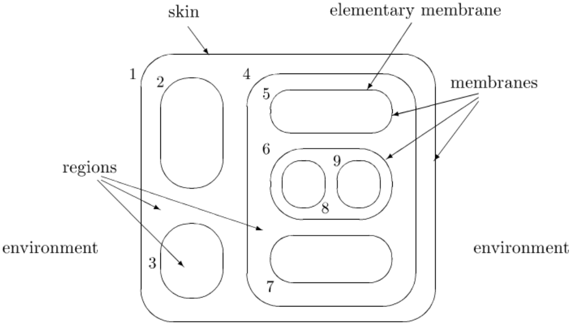
- Multisets

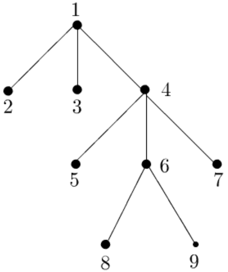
Membrane structure



- Multisets
- Rewriting rules

Membrane structure





- Multisets
- Rewriting rules
- Passive vs. Active

P system with active membranes

- $\Pi = (\Sigma, C_0, R_1, \ldots R_m)$

P system with active membranes

- $\Pi = (\Sigma, C_0, R_1, \ldots R_m)$
- $C = (T, l, c)$
  - $l : V(T) \to \{1, \ldots, m\}$
  - $c : V(T) \to \mathbb{N}^{\Sigma}$

P system with active membranes

- $\Pi = (\Sigma, C_0, R_1, \dots R_m)$
- $C = (T, l, c)$
  - $l : V(T) \to \{1, \dots, m\}$
  - $c : V(T) \to \mathbb{N}^\Sigma$
- Rewriting rules
  - $u \to v$
  - $u \to v\delta$
  - $u \to [_j v]_j$,
    where $u \in \mathbb{N}^\Sigma, |u| \geq 1$ and $v \in \mathbb{N}^{\Sigma \times \{\cdot, \uparrow, \downarrow_j\}}$

Computation

- Maximal parallel vs. sequential

Computation

- Maximal parallel vs. sequential
- Language
  - generating mode
  - accepting mode

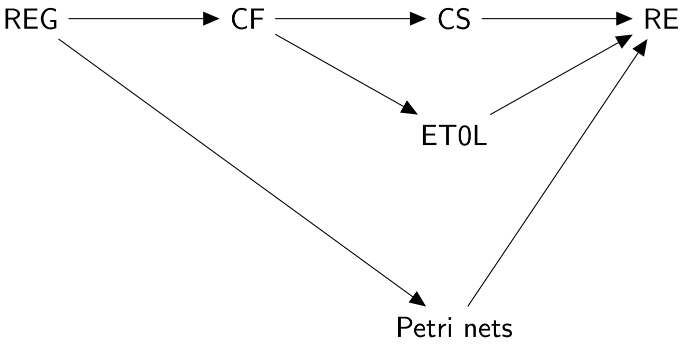
Chomski hierarchy

$$\mathrm{REG} \longrightarrow \mathrm{CF} \longrightarrow \mathrm{CS} \longrightarrow \mathrm{RE}$$

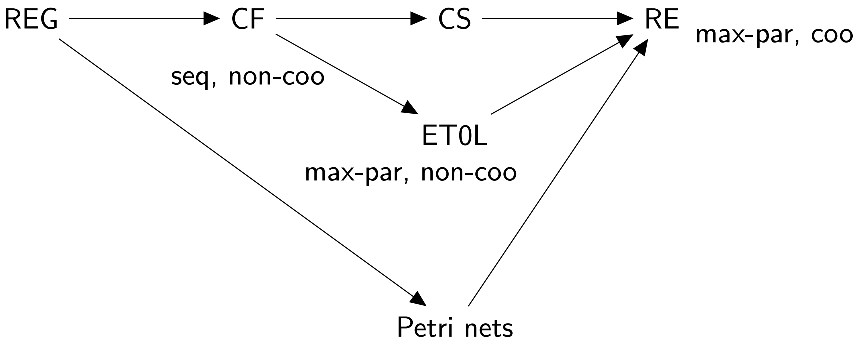
Chomski hierarchy

$$\begin{tikzcd} \mathrm{REG} \arrow[r] & \mathrm{CF} \arrow[r] \arrow[dr] & \mathrm{CS} \arrow[r] & \mathrm{RE} \\ & & \mathrm{ET0L} \arrow[ur] & \end{tikzcd}$$

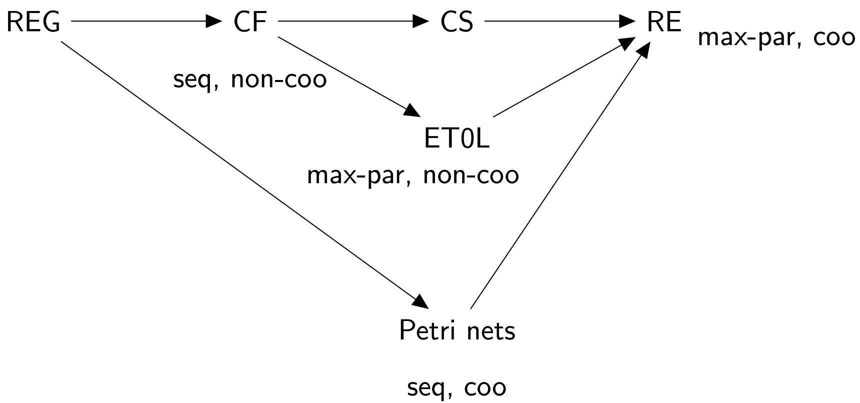
Chomski hierarchy



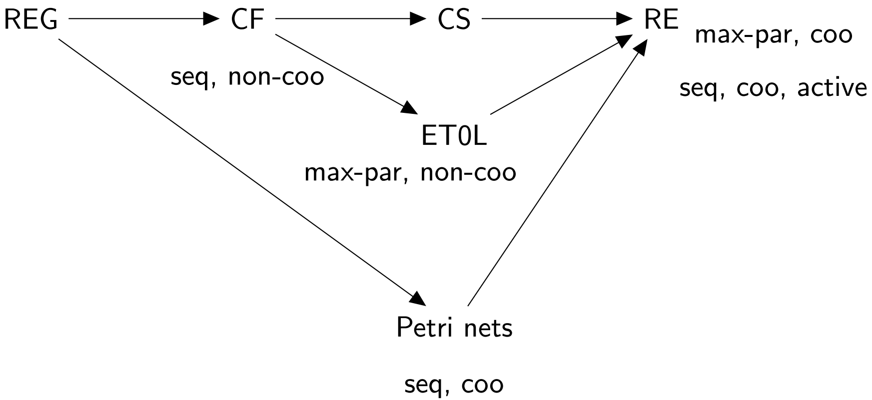
Chomski hierarchy



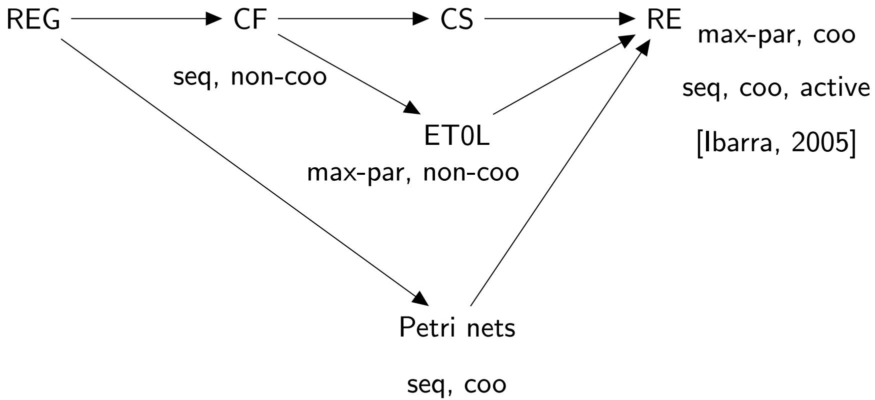
Chomski hierarchy



Chomski hierarchy



Chomski hierarchy



Vector addition systems

- $G = (x, W), x \in \mathbb{N}^n, W \subseteq Z^n$

Vector addition systems

- $G = (x, W), x \in \mathbb{N}^n, W \subseteq Z^n$
- Reachability set $R(G) = \{z | \exists v_1 \dots v_j \in W : z = x + v_1 + \dots v_j \text{ and } \forall 1 \leq i \leq j : x + v_1 + \dots + v_i \geq 0\}$

Vector addition systems

- $G = (x, W), x \in \mathbb{N}^n, W \subseteq Z^n$
- Reachability set $R(G) = \{z | \exists v_1 \dots v_j \in W : z = x + v_1 + \dots v_j$ and $\forall 1 \leq i \leq j : x + v_1 + \dots + v_i \geq 0\}$
- Same reachability set as Petri nets

Termination problems

- Halting problem

Termination problems

- Halting problem
- Existence of (in)finite computation

Termination problems

- Halting problem
- Existence of (in)finite computation
- Reachability graph

Termination problems

- Halting problem
- Existence of (in)finite computation
- Reachability graph
- Two conditions:
  - $C_1 \leq C_2 \Rightarrow$ each transition in C_1 can be fired in C_2

Termination problems

- Halting problem
- Existence of (in)finite computation
- Reachability graph
- Two conditions:
  - $C_1 \leq C_2 \Rightarrow$ each transition in C_1 can be fired in C_2
  - for each infinite computation there is C_1, C_2, such that $C_1 \rightarrow^* C_2$ and $C_1 \leq C_2$

Termination problems

- Halting problem
- Existence of (in)finite computation
- Reachability graph
- Two conditions:
  - $C_1 \leq C_2 \Rightarrow$ each transition in C_1 can be fired in C_2
  - for each infinite computation there is C_1, C_2, such that $C_1 \rightarrow^* C_2$ and $C_1 \leq C_2$
- Dickson's lemma: For every infinite sequence of tuples over $\mathbb{N}$ $\{a_i\}_{i=0}^{\infty}$ there are $i < j$ such that $a_i \leq a_j$

Termination problems in active membranes

- How to use Dickson's lemma for active membranes?
- Idea: encode configuration to k-tuple maintaining two conditions

Termination problems in active membranes

- How to use Dickson's lemma for active membranes?
- Idea: encode configuration to k-tuple maintaining two conditions

Definition

$C_1 = (T_1, l_1, c_1) \leq C_2 = (T_2, l_2, c_2) \Leftrightarrow$
$\exists$ isomorphism $f : T_1 \to T_2$ such that $\forall d \in V(T_1)$:

- $l_1(d) = l_2(f(d))$
- $c_1(d) \subseteq c_2(f(d))$

Termination problems in active membranes

- How to use Dickson's lemma for active membranes?
- Idea: encode configuration to k-tuple maintaining two conditions

Definition

$C_1 = (T_1, l_1, c_1) \leq C_2 = (T_2, l_2, c_2) \Leftrightarrow$
$\exists$ isomorphism $f : T_1 \to T_2$ such that $\forall d \in V(T_1)$:

- $l_1(d) = l_2(f(d))$
- $c_1(d) \subseteq c_2(f(d))$

Lemma

$C_1 = (T_1, l_1, c_1) \leq C_2 = (T_2, l_2, c_2) \Rightarrow \exists$ isomorphism $f : T_1 \to T_2$ such that rule r is applicable in $d \in T_1 \Rightarrow r$ is applicable in $f(d)$.

Termination problems in active membranes

Definition

Define $end(C)$ as k-tuple satisfying
$enc(C_1) \leq enc(C_2) \Rightarrow C_1 \leq C_2$.

Termination problems in active membranes

Definition

Define $end(C)$ as k-tuple satisfying
$enc(C_1) \leq enc(C_2) \Rightarrow C_1 \leq C_2$.

Lemma

For every infinite computation there is $i < j$ such that $C_i \leq C_j$.

Termination problems in active membranes

Definition

Define $end(C)$ as k-tuple satisfying
$enc(C_1) \leq enc(C_2) \Rightarrow C_1 \leq C_2$.

Lemma

For every infinite computation there is $i < j$ such that $C_i \leq C_j$.

Proof.

Assume infinite sequence $\{enc(C_i)\}_{i=0}^{\infty}$. From Dickson's lemma there is $i < j$ such that $enc(C_1) \leq enc(C_2)$. Our property of enc implies $C_i \leq C_j$. $\square$

Termination problems in active membranes

Theorem

Existence of infinite computation in sequential P systems with active membranes is decidable.

Termination problems in active membranes

Theorem

Existence of infinite computation in sequential P systems with active membranes is decidable.

Theorem

Existence of finite computation in sequential P systems with active membranes is decidable.

Termination problems in active membranes

Theorem

Existence of infinite computation in sequential P systems with active membranes is decidable.

Theorem

Existence of finite computation in sequential P systems with active membranes is decidable.

Proof.

Reduction to reachability of register machines. □

Ibarra (2005).
On sequential and 1-deterministic p systems.
In Wang, L., editor, *Computing and Combinatorics*, volume 3595 of *Lecture Notes in Computer Science*, pages 905–914. Springer Berlin Heidelberg.

Thanks for your attention!