Interactions in P systems

Artiom Alhazov

Institute of Mathematics and Computer Science
Academy of Sciences of Moldova
artiom@math.md

October 17, 2006
Computational Biomodelling Laboratory
Åbo Academi University, Turku

References:
http://psystems.disco.unimib.it
http://www.geocities.com/aartiom/pub_aa.html

Outline

Outline

Outline

Multiset

- Let O be a finite alphabet
- A multiset is a set with multiplicities
- Represented by string $w \in O^*$

Example

$a \quad c \quad a$
$b \quad a \quad b \quad c$

3 copies of a, 2 copies of b and 2 copies of c

Reaction rules

- $u \longrightarrow v$
- Consume a multiset u, and
- Produce a multiset v

Example

$a \quad c \quad a \qquad ba \longrightarrow bc$

$b \quad a \quad b \quad c \qquad ca \longrightarrow a$

Maximal parallelism; non-determinism

- **Parallelism**
- The same rule may be applied multiple times
- Different rules may be applied simultaneously

Maximal parallelism; non-determinism

- **Parallelism**
- The same rule may be applied multiple times
- Different rules may be applied simultaneously

- **Maximality**
- Objects that are not consumed, remain idle
- No rule should be applicable to the idle objects

Maximal parallelism; non-determinism

- **Parallelism**
- The same rule may be applied multiple times
- Different rules may be applied simultaneously

- **Maximality**
- Objects that are not consumed, remain idle
- No rule should be applicable to the idle objects

Example

$a \quad c \quad a \quad ba \longrightarrow bc$ applied once

$b \quad a \quad b \quad c \quad ca \longrightarrow a$ applied 2 times

Maximal parallelism; non-determinism

- **Parallelism**
- The same rule may be applied multiple times
- Different rules may be applied simultaneously

- **Maximality**
- Objects that are not consumed, remain idle
- No rule should be applicable to the idle objects

Example

a		c		a	$ba \longrightarrow bc$ applied 2 times
	b	a	b	c	$ca \longrightarrow a$ applied once

Computation

Example

Rules $ba \to bc$, $ca \to a$, starting multiset $b^2a^3c^2$.

$bbaaacc \Rightarrow (bc)^2(a)c$:

Computation

Example

Rules $ba \rightarrow bc$, $ca \rightarrow a$, starting multiset $b^2a^3c^2$.

$bbaaacc \Rightarrow (bc)^2(a)c$:

$bbaaacc \Rightarrow b(bc)(a)^2$:

Computation

Example

Rules $ba \to bc$, $ca \to a$, starting multiset $b^2a^3c^2$.

$bbaaacc \Rightarrow (bc)^2(a)c$:

$bbaccc \Rightarrow bbcccc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbccc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbac \Rightarrow bbcc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbac \Rightarrow bba \Rightarrow bbc.$

$bbaaacc \Rightarrow b(bc)(a)^2$:

Computation

Example

Rules $ba \to bc$, $ca \to a$, starting multiset $b^2a^3c^2$.

$bbaaacc \Rightarrow (bc)^2(a)c$:

$bbaccc \Rightarrow bbcccc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbccc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbac \Rightarrow bbcc.$

$bbaccc \Rightarrow bbacc \Rightarrow bbac \Rightarrow bba \Rightarrow bbc.$

$bbaaacc \Rightarrow b(bc)(a)^2$:

$bbaac \Rightarrow bbccc.$

$bbaac \Rightarrow bbac \Rightarrow bbcc.$

$bbaac \Rightarrow bbac \Rightarrow bba \Rightarrow bbc.$

Structure

- Objects are in regions
- Regions are delimited by membranes
- Associated to the region directly inside
- External region is called the environment

Structure

- Objects are in regions
- Regions are delimited by membranes
- Associated to the region directly inside
- External region is called the environment

- **Cell-like systems**
- Membranes are nested
- Serve as channels
- Tree structure

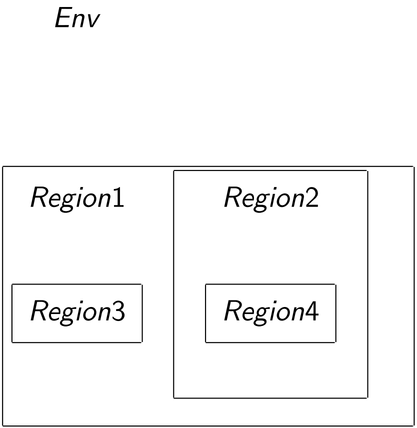
Structure

- Objects are in regions
- Regions are delimited by membranes
- Associated to the region directly inside
- External region is called the environment

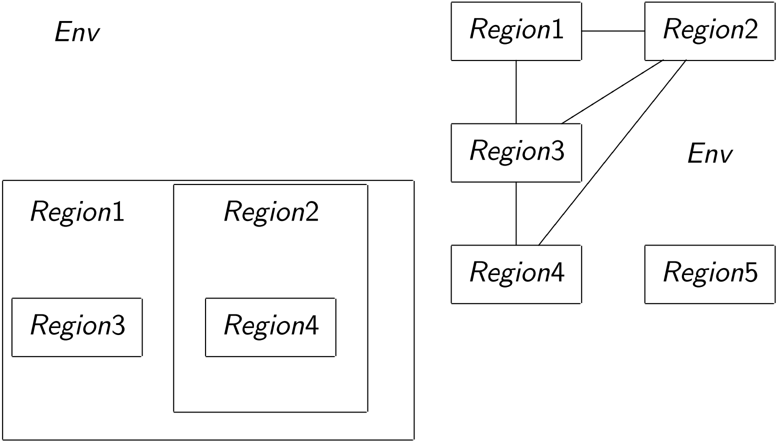
- **Cell-like systems**
- Membranes are nested
- Serve as channels
- Tree structure

- **Tissue-like systems**
- Cells are in the environment
- Connected by channels
- Graph structure

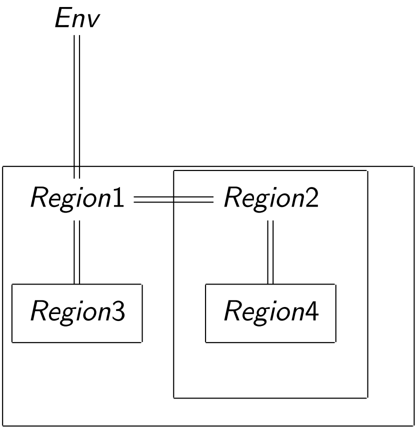
Cell-like VS tissue-like

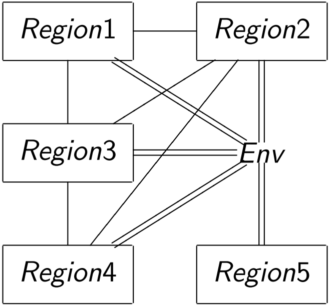


Cell-like VS tissue-like



Cell-like VS tissue-like





Result

- Objects can be moved between regions
- Transitional model: target indications
- Destinations are specified in the right side of the rules
- Designated output region (env.=0 or membrane)

Result

- Objects can be moved between regions
- Transitional model: target indications
- Destinations are specified in the right side of the rules
- Designated output region (env.=0 or membrane)

- Consider at halting
- total number of objects: a number
- number of objects of each kind: a vector
- order in which objects come: a word

Two objects by rewriting

- Non-cooperative rules work like a $0L$ system
- Rules of type $u \to v$, $|u| \leq 2$ are already too powerful.

Two objects by rewriting

- Non-cooperative rules work like a $0L$ system
- Rules of type $u \longrightarrow v$, $|u| \leq 2$ are already too powerful.

Catalysts $C \subseteq O$

- Catalytic rules are of type $ca \longrightarrow cv$, $c \in C$, $a \in O \setminus C$, $v \in (O \setminus C)^*$
- We may assume the catalysts are distinct
- Essentially, a catalyst ensures that out of associated rules $a \longrightarrow v$ at most one will be applied.

Catalysts are universal
[Freund et al.]

Theorem

Purely catalytic P systems with 3 catalysts generate RE.

- Like 3 *CF* grammars
- working on the same sentential form
- only interacting by competing for resources
- With non-cooperative rules this can be improved

Catalysts are universal
[Freund et al.]

Theorem

Purely catalytic P systems with 3 catalysts generate RE.

- Like 3 *CF* grammars
- working on the same sentential form
- only interacting by competing for resources
- With non-cooperative rules this can be improved

Theorem

P systems with 2 catalysts generate RE.

Bi-stable catalysts

- $a \longrightarrow u,$
- $ca \longrightarrow c'u$

Theorem

P systems with 1 bi-stable catalyst generate RE.

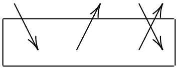
- Like a $0L$ system and a CF grammar
- the latter has 1-bit memory

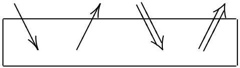
Interaction by moving objects

- Symport rules: (v, out) or (u, in)
- Antiport rule: $(v, out; u, in)$
- weight: $\max(|u|, |v|)$

Interaction by moving objects

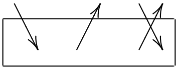
- Symport rules: (v, out) or (u, in)
- Antiport rule: $(v, out; u, in)$
- weight: $\max(|u|, |v|)$
- Minimal cooperation: rules envolve at most 2 objects
- in one way: $(sym_1, anti_1)$ or (sym_2)

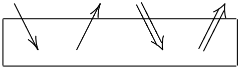




Interaction by moving objects

- Symport rules: (v, out) or (u, in)
- Antiport rule: $(v, out; u, in)$
- weight: $\max(|u|, |v|)$
- Minimal cooperation: rules envolve at most 2 objects
- in one way: $(sym_1, anti_1)$ or (sym_2)





- Evolution–communication: non-cooperative evolution plus communication with minimal cooperation

Protons $P \subseteq O$

A proton

- is one of the interacting objects
- is never mentioned in evolution rules
- is essentially an *m*-stable catalyst of communication

The previous Theorem was proved as a **corollary** of

Theorem

P systems with 1 proton and 2 membranes generate RE

More results

Time-freeness: rules are not necessarily executed in 1 step; the result should be independent.

Theorem

4 protons and 2 membranes suffice for generating RE

More results

Time-freeness: rules are not necessarily executed in 1 step; the result should be independent.

Theorem

4 protons and 2 membranes suffice for generating RE

- Accepting: the input multiset is placed in a designated region; it is accepted iff the system halts
- Determinism: the maximal multiset of applicable rules is always unique

Theorem

3 membranes suffice for accepting PsRE

Symport/antiport only

Rules envolving at most 3 objects

- There are infinitely many objects from $E \subseteq O$ in the environment

Known results

Theorem

P systems with $(sym_1, anti_{2/1})$ generate NRE and deterministically accept PsRE, already with 1 membrane.
The accepting case also holds for (sym_3)

Symport/antiport only

Rules envolving at most 3 objects

- There are infinitely many objects from $E \subseteq O$ in the environment

Known results

Theorem

P systems with $(sym_1, anti_{2/1})$ generate NRE and deterministically accept PsRE, already with 1 membrane.
The accepting case also holds for (sym_3)

Improved result:

Theorem

P systems with (sym_3) and 1 membrane generate at least N_7RE

Minimal cooperation

Rules envolving at most 2 objects: $(sym_1, anti_1)$ or (sym_2)

"Clean" result:

Theorem

Such P systems with 3 membranes generate NRE

Minimal cooperation

Rules envolving at most 2 objects: $(sym_1, anti_1)$ or (sym_2)

"Clean" result:

Theorem

Such P systems with 3 membranes generate NRE

Latest and optimal: 1 "garbage" object

Theorem

Such P systems with 2 membranes generate at least N_1RE; if a set containing 0 is generated, that set is finite.

Theorem

Such P systems with 1 membrane only generate finite sets.

Massive cooperation

Limiting membranes and $|O|$

"Heavy" symport/antiport rules are considered; information is stored in big multisets over a small alphabet.

Theorem

P systems with $m \geq 2$ membranes and $n \geq 1$ symbols, $m + n \geq 6$ generate NRE

Massive cooperation

Limiting membranes and $|O|$

"Heavy" symport/antiport rules are considered; information is stored in big multisets over a small alphabet.

Theorem

P systems with $m \geq 2$ membranes and $n \geq 1$ symbols, $m + n \geq 6$ generate NRE

- Generating vectors
- Tissue P systems
- smaller (m, n)

are also studied.

Active membranes

$h \in H$ is a membrane label, $e, e', e'' \in E$ are polarizations

(a) $[\ a \to v\ ]_h^e$ evolution

(b) $a[\ \ ]_h^e \to [\ b\ ]_h^{e'}$ send in

(c) $[\ a\ ]_h^e \to [\ \ ]_h^{e'} b$ send out

(d) $[\ a\ ]_h^e \to b$ dissolution

(e) $[\ a\ ]_h^e \to [\ b\ ]_h^{e'} [\ c\ ]_h^{e''}$ division of elementary membrane

- etc.

Question: how much information does a membrane need to carry for the systems to be complete or efficient

Properties

- Determinism
- Confluence

Properties

- Determinism
- Confluence
- Uniform solution
- Semi-uniform solution

Properties

- Determinism
- Confluence
- Uniform solution
- Semi-uniform solution
- etc.

Polarizations

Theorem

Systems with 2 polarizations are complete (in 1 membrane) and efficient.

Theorem

Systems with 1 polarization are complete (unbounded membrane division) and efficient (with non-elementary membrane division).

Polarizations

Theorem

Systems with 2 polarizations are complete (in 1 membrane) and efficient.

Theorem

Systems with 1 polarization are complete (unbounded membrane division) and efficient (with non-elementary membrane division).

Minimal parallelism: at least one rule associated to every membrane is applied, if possible.

Theorem

Minimally parallel systems are efficient with 6 or 4 polarizations, depending on the rules used.

- Using separation instead of division
- Changing membrane labels instead of polarizations
- Descriptional complexity parameters
- . . .

- Using separation instead of division
- Changing membrane labels instead of polarizations
- Descriptional complexity parameters
- ...
- Cooperation by promoters or inhibitors
- Sorting P systems
- Solving graph problems
- ...

- Using separation instead of division
- Changing membrane labels instead of polarizations
- Descriptional complexity parameters
- ...
- Cooperation by promoters or inhibitors
- Sorting P systems
- Solving graph problems
- ...
- Topics outside the scope of this presentation