(Tissue) P Systems Working in the k-Restricted Minimally Parallel Derivation Mode

Rudolf Freund[1] and Sergey Verlan[2]

[1]Faculty of Informatics, Vienna University of Technology
Favoritenstr. 9, 1040 Vienna, Austria
Email: rudi@emcc.at

[2]LACL, Département Informatique
UFR Sciences et Technologie, Université Paris XII
61, av. Général de Gaulle, 94010 Créteil, France
Email: verlan@univ-paris12.fr

Wien, August 27, 2008

Variants of Derivation Modes

- maximally parallel derivation mode

Gh. Păun: *Membrane Computing. An Introduction*. Springer, Berlin, 2002.

- sequential derivation mode
- asynchronous derivation mode
- minimally parallel derivation mode

G. Ciobanu, L. Pan, Gh. Păun, M. J. Pérez-Jiménez: P systems with minimal parallelism, *Theoretical Computer Science* **378**, 1 (2007), 117–130.

- variants of the minimal parallel derivation mode

R. Freund, S. Verlan: A formal framework for P systems. In: G. Eleftherakis, P. Kefalas, Gh. Păun (Eds.), Membrane Computing, 8th International Workshop, WMC 2007,LNCS **4860**, Springer, Berlin, 2008, 271–284.

- *k*-restricted minimal parallel derivation mode

R. Freund, S. Verlan: P systems working in the *k*-restricted minimally parallel derivation mode. *Proceedings of the Workshop Computing with Biomolecules,* Vienna, Austria (August 27, 2008).

Network of Cells

We consider networks of cells having a static structure.

Definition

A *network of cells of degree* $n \geq 1$ (an *NC of degree* $n \geq 1$, for short) is a construct $\Pi = (V, T, w_1, w_2, \ldots, w_n, R)$ where

1. V is a *finite alphabet*;
2. $T \subseteq V$ is the *terminal alphabet*;
3. $w_i \in \langle V, \mathbb{N}_\infty \rangle$, for all $1 \leq i \leq n$, is the multiset *initially associated to cell* i;
4. R is a finite set of *interaction rules* of the form $(P : X \rightarrow Y)$ where $X = (x_1, \ldots, x_n)$, $Y = (y_1, \ldots, , y_n)$, $x_i, y_i \in \langle V, \mathbb{N} \rangle$, $1 \leq i \leq n$, are vectors of multisets over V and P is a recursive predicate for multisets over V.

Asynchronous Derivation Mode

Definition

For the *asynchronous* derivation mode (*asyn*),

$$Appl\left(\Pi, C, asyn\right) = Appl\left(\Pi, C\right),$$

i.e., there are no particular restrictions on the multisets of rules applicable to C.

Sequential Derivation Mode

Definition

For the *sequential* derivation mode (*sequ*),

$$Appl\left(\Pi, C, sequ\right) = \left\{R' \mid R' \in Appl\left(\Pi, C\right) \text{ and } \left|R'\right| = 1\right\},$$

i.e., any multiset of rules $R' \in Appl\left(\Pi, C, sequ\right)$ has size 1.

Maximally Parallel Derivation Mode

Definition

For the *maximally parallel* derivation mode (max),

$$Appl(\Pi, C, max) = \{R' \mid R' \in Appl(\Pi, C) \text{ and there is no } R'' \in Appl(\Pi, C) \text{ with } R'' \supsetneq R'\}.$$

Minimally Parallel Derivation Mode

In the minimally parallel derivation mode, we need an additional feature for the set of rules R, i.e., we consider a partition of R into subsets R_1 to R_h.

Definition

For the *minimally parallel* derivation mode (*min*),

$$
\begin{aligned}
Appl\,(\Pi, C, min) \quad = \quad & \{R' \mid R' \in Appl\,(\Pi, C, asyn) \text{ and} \\
& \text{there is no } R'' \in Appl\,(\Pi, C, asyn) \\
& \text{with } (R'' - R') \cap R_j \neq \emptyset \\
& \text{and } R' \cap R_j = \emptyset \text{ for some } j,\ 1 \leq j \leq h\}\,.
\end{aligned}
$$

k-restricted Minimally Parallel Derivation Mode

From every rule set R_j, at most k rules can be taken.

Definition

For the *k-restricted minimally parallel* derivation mode (min_k),

$$\begin{aligned} Appl\,(\Pi, C, min_k) \quad = \quad & \{R' \mid R' \in Appl\,(\Pi, C, min) \text{ and} \\ & |R' \cap R_j| \leq k \text{ for all } j,\ 1 \leq j \leq h\}\,. \end{aligned}$$

Transition Step

Definition

Given a configuration C of Π and a derivation mode ϑ, we may choose a multiset of rules $R' \in \mathit{Appl}(\Pi, C, \vartheta)$ in a non-deterministic way and apply it to C. The result of this *transition step* from the configuration C with applying R' is the configuration $\mathit{Apply}(\Pi, C, R')$, and we also write $C \Longrightarrow_{(\Pi,\vartheta)} C'$. The reflexive and transitive closure of the transition relation $\Longrightarrow_{(\Pi,\vartheta)}$ is denoted by $\Longrightarrow^*_{(\Pi,\vartheta)}$.

Definition

A configuration C is said to be *accessible* in Π with respect to the derivation mode ϑ if and only if $C_0 \Longrightarrow^*_{(\Pi,\vartheta)} C$ (C_0 is the initial configuration of Π). The set of all accessible configurations in Π is denoted by $\mathit{Acc}(\Pi, \vartheta)$.

(Total) Halting

The most important halting condition used from the beginning in the P systems area is the *total halting*, usually considered as THE *halting*:

Definition

An accessible configuration C is said to fulfill the *total halting* condition (H) if and only if no multiset of rules can be applied to C with respect to the derivation mode anymore, i.e.,

$$H(\Pi, \vartheta) = \{C' \mid C' \in Acc(\Pi, \vartheta) \text{ and } Appl(\Pi, C', \vartheta) = \emptyset\}.$$

Computations

Definition

A *computation* in a network of cells Π, $\Pi = (n, V, w, R)$, starts with the initial configuration $C_0 = w$ and continues with transition steps according to the chosen derivation mode ϑ; it is called *successful* if we reach a configuration C to which no multiset of rules can be applied with respect to the derivation mode ϑ anymore, i.e., $Appl(\Pi, C, \vartheta) = \emptyset$ (we also say that the computation *halts*).

Results of Computations

Definition

As the results of a halting computation we take the number of objects in a specified output cell. We shall use the notation

$$NO_m C_n(\vartheta)\,[\text{parameters for rules}]$$

to denote the family of sets of natural numbers generated by networks of cells $\Pi = (n, V, w, R)$ with $m = |V|$ as well as ϑ indicating the derivation mode; the *parameters for rules* describe the specific features of the rules in R. If any of the parameters m and n is unbounded, we replace it by $*$.

A General Result for min_1

For networks of cells and therefore for (tissue) P systems of any arbitrary type, we can show that derivation mode *seq* can be seen as a special variant of the derivation mode min_1.

Theorem

For networks of cells with any arbitrary type X of rules we have

$$NO_* C_* ((min_1, \Phi_0))[X] = NO_* C_* (seq)[X]$$

with Φ_0 being the trivial partitioning where the rule set itself is the single element of the partitioning.

(Purely) Catalytic Tissue P Systems

Definition

A *noncooperative evolution* rule is of the form

$$((a, i) \rightarrow (y_1, 1) \dots (y_n, n))$$

where a is a single symbol.
A *catalytic* rule is of the form

$$((c, i)(a, i) \rightarrow (c, i)(y_1, 1) \dots (y_n, n))$$

where c is from a distinguished subset $V_C \subset V$ such that in all rules (noncooperative evolution rules, catalytic rules) of the whole system the symbols y_i are from $(V - V_C)^*$ and the symbols a are from $(V - V_C)$.

(Purely) Catalytic P Systems

Imposing the restriction that the noncooperative evolution rules and the catalytic rules in a network of cells allow for finding a hierarchical tree structure of membranes such that symbols either stay in their membrane region or are sent out to the surrounding membrane region or sent into an inner membrane, then we get the classical catalytic P systems priorities.

Catalytic P systems using only catalytic rules are called *purely catalytic P systems*.

Universality of (Purely) Catalytic P Systems

As we know, only two (three) catalysts in one membrane are needed to obtain NRE with (purely) catalytic P systems without priorities working in the maximally parallel derivation mode (the subscript $_{-cat}$ to N indicates that we do not count the catalysts):

Theorem

$$\begin{aligned} NRE &= N_{-cat} O_* C_1 (max) [cat_2] \\ &= N_{-cat} O_* C_1 (max) [pcat_3] . \end{aligned}$$

R. Freund, L. Kari, M. Oswald, P. Sosík: Computationally universal P systems without priorities: two catalysts are sufficient. *Theoretical Computer Science* **330** (2005), 251–266,

Purely Catalytic Networks of Cells

If we now partition the rule set in a purely catalytic P system according to the catalysts present in each membrane, this partitioning replaces the use of the catalysts when working in the 1-restricted minimally parallel derivation mode, because by definition from each of these sets then – if possible – exactly one rule (as with the use of the corresponding catalyst) is chosen: from the set of purely catalytic rules R we obtain the corresponding set of noncooperative rules R' as

$$\begin{aligned} R' &= \{((a,i) \to (y_1,1)\ldots(y_n,n)) \mid \\ &\quad ((c,i)(a,i) \to (c,i)(y_1,1)\ldots(y_n,n)) \in R\} \end{aligned}$$

as well as the corresponding partitioning of R' as

$$\begin{aligned} R'_{i,c} &= \{((a,i) \to (y_1,1)\ldots(y_n,n)) \mid \\ &\quad ((c,i)(a,i) \to (c,i)(y_1,1)\ldots(y_n,n)) \in R\}\,. \end{aligned}$$

Purely Catalytic P Systems

Considering purely catalytic P systems in one membrane, we therefore infer the following quite astonishing result that when using the 1-restricted minimally parallel derivation mode for a suitable partitioning of rules we only need noncooperative rules:

Theorem

$NRE = N_{-cat} O_* C_1 (min_1) [noncoop]$.

Extended Spiking Neural P Systems

An *extended spiking neural P system* (ESNP) is a construct $\Pi = (m, S, R)$ where

- m is the number of *neurons*; the neurons are uniquely identified by a number between 1 and m;
- S describes the *initial configuration* by assigning an initial value (of spikes) to each neuron;
- R is a finite set of *rules* of the form $\left(i, E/a^k \rightarrow P\right)$ such that $i \in [1..m]$ (specifying that this rule is assigned to neuron i), $E \subseteq REG\left(\{a\}\right)$ is the *checking set* (the current number of spikes in the neuron has to be from E if this rule shall be executed), $k \in \mathbb{N}$ is the "number of spikes" consumed by this rule, and P is a (possibly empty) set of *productions* of the form (l, a^w) where $l \in [1..m]$, $w \in \mathbb{N}$ is the *weight* of the energy sent along the axon from neuron i to neuron l.

ESNP as a Network of Cells

We now consider the ESNP system $\Pi = (m, S, R)$ as a network of cells $\Pi' = (m, \{a\}, S, R', \Phi_S)$ working in the 1-restricted minimally parallel derivation mode, with

$$\begin{aligned} R' \quad = \quad & \{ \left(E : \left(a^k, i\right) \rightarrow \left(a^{w_1}, l_1\right) \ldots \left(a^{w_n}, l_n\right)\right) \mid \\ & \left(i, E/a^k \rightarrow \left(l_1, a^{w_1}\right) \ldots \left(l_n, a^{w_n}\right)\right) \in R \} \end{aligned}$$

and the *standard* partitioning Φ_S of R' into R'_i, $1 \leq i \leq m$, according to the set of neurons, i.e.,

$$\begin{aligned} R'_i \quad = \quad & \{ \left(E : \left(a^k, i\right) \rightarrow \left(a^{w_1}, l_1\right) \ldots \left(a^{w_n}, l_n\right)\right) \mid \\ & \left(E : \left(a^k, i\right) \rightarrow \left(a^{w_1}, l_1\right) \ldots \left(a^{w_n}, l_n\right)\right) \in R' \} . \end{aligned}$$

Extended Spiking Neural P Systems

The 1-restricted minimally parallel derivation mode chooses one rule – if possible – from every set R_i and then applies such a multiset of rules in parallel, which directly corresponds to applying one spiking rule in every neuron where a rule can be applied.

Theorem

$NRE = NO_1 C_3 (min_1) [ESNP]$.

A. Alhazov, R. Freund, M. Oswald, M. Slavkovik: Extended spiking neural P systems generating strings and vectors of non-negative integers. In: H.J. Hoogeboom, Gh. Păun, G. Rozenberg, (Eds.): *Pre-Proceedings of the Seventh International Workshop on Membrane Computing, WMC7*, Leiden, The Netherlands (2006), 88–101.

Conclusion

The main purpose of this talk was to introduce the k-restricted minimally parallel derivation mode.

P systems with purely catalytic rules working in the maximally parallel derivation mode can be considered as P systems working with the corresponding noncooperative rules in the 1-restricted minimally parallel derivation mode when partitioning the rule sets for each membrane with respect to the catalysts.

The 1-restricted minimally parallel derivation mode, allows us to interpret the way of how spiking neural P systems (without delays) work in a sequential way on the level of cells, but in the maximally parallel way on the level of the whole system (with a partitioning of the rules given by the individual neurons).

THANK YOU FOR YOUR ATTENTION !