

# Chapter 7

## P Colonies of Capacity One and Modularity

**Luděk Cienciala**

*Silesian University in Opava, Czech Republic*

**Lucie Ciencialová**

*Silesian University in Opava, Czech Republic*

**Miroslav Langer**

*Silesian University in Opava, Czech Republic*

### ABSTRACT

*In this paper, the authors continue the investigation of P colonies introduced in Kelemen, Kelemenová, and Păun (2004). This paper examines a class of abstract computing devices composed of independent agents, acting and evolving in a shared environment. The first part is devoted to the P colonies of the capacity one. The authors present improved results concerning the computational power of the P colonies with capacity one and without using checking programs. The second part of the paper examines the modularity of the P colonies. The authors then divide the agents into modules.*

### 1. INTRODUCTION

P colonies were introduced (Kelemen, Kelemenová, & Păun, 2004) as formal models of a computing device inspired by membrane systems and formal grammars called colonies. This model is inspired by structure and functioning of a community of living organisms in a shared environment.

The independent organisms living in a P colony are called agents or cells. Each agent is represented by a collection of objects embedded in a membrane. The number of objects inside each agent is the same and constant during computation. The environment contains several copies of the basic environmental object denoted by  $e$ . The object  $e$  appears in arbitrary large number of copies in the environment.

DOI: 10.4018/978-1-4666-4253-9.ch007

With each agent a set of programs is associated. The program, which determines the activity of the agent, is very simple and depends on content of the agent and on multiset of objects placed in the environment. Agent can change content of the environment by programs and through the environment it can affect the behavior of other agents.

This interaction between agents is a key factor in functioning of the P colony. In each moment each object inside the agent is affected by executing the program.

For more information about P colonies see Kelemen and Kelemenová (2005) and about P systems generally see Păun, Rozenberg, and Salomaa (2009) or P Systems (2004).

## 2. DEFINITIONS

Throughout the paper we assume that the reader is familiar with the basics of the formal language theory.

We use  $NRE$  to denote the family of the recursively enumerable sets of natural numbers. Let  $\Sigma$  be the alphabet. Let  $\Sigma^*$  be the set of all words over  $\Sigma$  (including the empty word  $\varepsilon$ ). We denote the length of the word  $w \in \Sigma^*$  by  $|w|$  and the number of occurrences of the symbol  $a \in \Sigma$  in  $w$  by  $|w|_a$ .

A multiset of objects  $M$  is a pair  $M = (V, f)$ , where  $V$  is an arbitrary (not necessarily finite) set of objects and  $f$  is a mapping  $f: V \rightarrow N$ ;  $f$  assigns to each object in  $V$  its multiplicity in  $M$ . The set of all multisets with the set of objects  $V$  is denoted by  $V^*$ . The set  $U \subseteq V$  is called the support of  $M$  and is denoted by  $supp(M)$  if for all

$x \in U$   $f(x) \neq 0$  holds. The cardinality of  $M$ , denoted by  $|M|$ , is defined by  $|M| = \sum_{a \in V} f(a)$ . Each multiset of objects  $M$  with the set of objects  $U = \{a_1, \dots, a_n\}$  can be represented as a string  $w$  over alphabet  $U$ , where

$$|w|_{a_i} = f(a_i); 1 \leq i \leq n.$$

Obviously, all words obtained from  $w$  by permuting the letters represent the same multiset  $M$ . The  $\varepsilon$  represents the empty multiset.

### 2.1. P Colonies

We briefly recall the notion of P colonies. A P colony consists of agents and an environment. Both the agents and the environment contain objects. With each agent a set of programs is associated. The program is formed from rules. There are two types of rules in the programs. The first type of rules, called the evolution rules, are of the form  $a \rightarrow b$ . It means that the object  $a$  inside the agent is rewritten (evolved) to the object  $b$ . The second type of rules, called the communication rules, are of the form  $c \leftrightarrow d$ . When the communication rule is performed, the object  $c$  inside the agent and the object  $d$  outside the agent swap their places. Thus after execution of the rule, the object  $d$  appears inside the agent and the object  $c$  is placed outside the agent.

In Kelemen and Kelemenová (2005) the set of programs was extended by the checking rules. These rules give an opportunity to the agents to opt between two possibilities. The rules are in the form  $r_1/r_2$ . If the checking rule is performed, then the rule  $r_1$  has higher priority to be executed over the rule  $r_2$ . It means that the agent checks whether the rule  $r_1$  is applicable. If the rule can be executed, then the agent is compulsory to use it. If the rule  $r_1$  cannot be applied, then the agent uses the rule  $r_2$ .

*Definition:* The P colony of the capacity  $k$  is a construct

$\Pi = (A, e, f, V_E, B_1, \dots, B_n)$ , where

- $A$  is an alphabet of the colony, its elements are called objects,
- $e \in A$  is the basic object of the colony,
- $f \in A$  is the final object of the colony,
- $V_E$  is a multiset over  $A - \{e\}$ ,

15 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:  
[www.igi-global.com/chapter/p-colonies-of-capacity-one-and-modularity/80060](http://www.igi-global.com/chapter/p-colonies-of-capacity-one-and-modularity/80060)

## Related Content

---

### Direct Perception and Action Decision for Unknown Object Grasping

Hiroyuki Masuta, Tatsuo Motoyoshi, Kei Sawai, Ken'ichi Koyanagi, Toru Oshima and Hun-Ok Lim (2017). *International Journal of Artificial Life Research* (pp. 38-51).

[www.irma-international.org/article/direct-perception-and-action-decision-for-unknown-object-grasping/182577](http://www.irma-international.org/article/direct-perception-and-action-decision-for-unknown-object-grasping/182577)

### Decoding Algorithm by Cooperation Between Hartmann Rudolph Algorithm and a Decoder Based on Syndrome and Hash

My Seddiq El Kasmi Alaoui and Said Nouh (2021). *International Journal of Natural Computing Research* (pp. 15-27).

[www.irma-international.org/article/decoding-algorithm-by-cooperation-between-hartmann-rudolph-algorithm-and-a-decoder-based-on-syndrome-and-hash/274935](http://www.irma-international.org/article/decoding-algorithm-by-cooperation-between-hartmann-rudolph-algorithm-and-a-decoder-based-on-syndrome-and-hash/274935)

### Slithering Intelligence for Predicting Tectonic Plate Movement

Maheswari Raja, Ashiya Parveen, Manobalan Manokaran, Mythili Palanisamy and P. Vijaya (2025). *Exploring the Micro World of Robotics Through Insect Robots* (pp. 235-252).

[www.irma-international.org/chapter/slithering-intelligence-for-predicting-tectonic-plate-movement/359280](http://www.irma-international.org/chapter/slithering-intelligence-for-predicting-tectonic-plate-movement/359280)

### A Computational Comparison of Swarm Optimization Techniques for Optimal Load Shedding Under the Presence of FACTS Devices to Avoid Voltage Instability

G. V. Nagesh Kumar, B. Venkateswara Rao, D. Deepak Chowdary and Polamraju V. S. Sobhan (2018). *Critical Developments and Applications of Swarm Intelligence* (pp. 182-214).

[www.irma-international.org/chapter/a-computational-comparison-of-swarm-optimization-techniques-for-optimal-load-shedding-under-the-presence-of-facts-devices-to-avoid-voltage-instability/198927](http://www.irma-international.org/chapter/a-computational-comparison-of-swarm-optimization-techniques-for-optimal-load-shedding-under-the-presence-of-facts-devices-to-avoid-voltage-instability/198927)

### Client-Awareness Resource Allotment and Job Scheduling in Heterogeneous Cloud by Using Social Group Optimization

Phani Praveen S. and K. Thirupathi Rao (2018). *International Journal of Natural Computing Research* (pp. 15-31).

[www.irma-international.org/article/client-awareness-resource-allotment-and-job-scheduling-in-heterogeneous-cloud-by-using-social-group-optimization/204881](http://www.irma-international.org/article/client-awareness-resource-allotment-and-job-scheduling-in-heterogeneous-cloud-by-using-social-group-optimization/204881)