P Systems with Active Membranes, without Polarizations and without Dissolution: A Characterization of P

Miguel A. Gutiérrez–Naranjo, Mario J. Pérez-Jiménez,
Agustín Riscos–Núñez, Francisco J. Romero–Campero

Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
University of Sevilla
Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
{magutier,marper,ariscosn,fran}@us.es

Abstract. We study the computational efficiency of recognizer P systems with active membranes without polarizations and without dissolution. The main result of the paper is the following: the polynomial computational complexity class associated with the class of recognizer P systems is equal to the standard complexity class **P**.

1 Introduction

The *theory of computation* deals with the *mechanical solvability* of problems, that is, searching solutions that can be described by a finite sequence of elementary processes or instructions. The first goal of this theory is general problem solving; that is, develop principles and methods that are able to solve any problem from a certain class of questions.

A *computational model* tries to capture those aspects of mechanical solutions of problems that are relevant to these solutions, including their inherent limitations. In some sense, we can think that computational models design machines according to certain necessity.

If we have a mechanically solvable problem and we have a specific algorithm solving it that can be implemented in a real machine, then it is very important to know how much computational resources (time or memory) are required for a given instance, in order to recognize the limitations of the real device.

Thus, one of the main goals of the *theory of computational complexity* is the study of the efficiency of algorithms and their data structures through the analysis of the resources required for solving problems (that is, according to their intrinsic computational difficulty). This theory provides a classification of the *abstract problems* that allows us to detect their inherent complexity from the computational solutions point of view.

Many interesting problems of the real world are presumably intractable and hence it is not possible to execute algorithmic solutions in an electronic computer when we deal with instances of those problems whose size is large. The theoretical limitations of the Turing machines in terms of computational power are also practical limitations to the digital computers.

Natural Computing is a new computing area inspired by nature, using concepts, principles and mechanisms underlying natural systems. *Evolutionary Computation* uses computational models of evolutionary processes as key elements in the design and implementation of computer–based problem solving systems [18]. *Neural Networks* are inspired in the structures of the brain and nervous system. *DNA Computing* is based on the computational potential of DNA molecules and on the capacity to handle them. *Membrane Computing* is inspired by the structure and functioning of living cells, and it is a cross-disciplinary field with contributions by computer scientists, biologists, formal linguists and complexity theoreticians, enriching each others with results, open problems and promising new research lines.

This emergent branch of Natural Computing was introduced by Gh. Păun in [8]. Since then it has received important attention from the scientific community. In fact, Membrane Computing has been selected by the Institute for Scientific Information, USA, as a fast *Emerging Research Front* in Computer Science, and [6] was mentioned in [19] as a highly cited paper in October 2003.

This new non-deterministic model of computation starts from the assumption that the processes taking place in the compartmental structure of a living cell can be interpreted as computations. The devices of this model are called *P systems*.

Roughly speaking, a P system consists of a cell-like membrane structure, in the compartments of which one places multisets of objects which evolve according to given rules in a synchronous non-deterministic maximally parallel manner[1].

Inspired in living cells, P systems abstract the way of obtaining new membranes. These processes are basically two: *mitosis* (membrane division) and *autopoiesis* (membrane creation). Both ways of generating new membranes have given rise to different variants of P systems: *P systems with active membranes*, where the new workspace is generated by membrane division, and *P systems with membrane creation*, where the new membranes are created from objects.

Both models are universal from a computational point of view, but technically, they are pretty different. In fact, nowadays there does not exist any theoretical result which proves that these models can simulate each other in polynomial time.

P systems with active membranes have been successfully used to design solutions to well-known **NP**-complete problems, as SAT [16], Subset Sum [13], Knapsack [14], Bin Packing [15] and Partition [3], but as Gh. Păun pointed in [10] *"membrane division was much more carefully investigated than membrane creation as a way to obtain tractable solutions to hard problems"*. Recently, the first results related to the power and design of algorithms to solve NP problems in these model have arisen (see [4, 17]).

P systems with active membranes were introduced in [7] with the membranes having polarizations, one of the "electrical charges" $0, -, +$, and several times the problem was formulated whether or not these polarizations are necessary in

[1] A layman-oriented introduction can be found in [9] and further bibliography at [20].

order to obtain polynomial solutions to **NP**–complete problems. The last current result is that from [1], where one proves that two polarizations suffice.

The present paper is both a contribution to this problem, and a contribution to another interesting problem in membrane computing, namely, of characterizing classic complexity classes, such as **P** and **NP**, by means of membrane computing complexity classes.

Specifically, we prove that **P** is equal to the family of problems which can be solved in a polynomial time by P systems with membrane division, without polarizations and *without dissolution*. At this moment, we do not know whether this last condition can be avoided, but either result would be of a great interest: if our result would remain true also when using membrane dissolution, then we would have the possitive answer to the problem of removing polarization; the other possibility would indicate a surprising role of the –apparently "innocent"– operation of membrane dissolution, as it will make the difference between efficiency and non–efficiency for P systems with membrane division and without polarization.

References

1. A. Alhazov, R. Freund, Gh. Păun: P systems with active membranes and two polarizations. *Proceedings of the Second Brainstorming Week on Membrane Computing* (Gh. Păun, A. Riscos, A. Romero, F. Sancho, eds.), Report RGNC 01/04, 2004, 20–35.
2. M.R. Garey, D.S. Johnson: *Computers and Intractability. A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, New York, 1979.
3. M.A. Gutiérrez-Naranjo, M.J. Pérez-Jiménez, A. Riscos-Núñez: A fast P system for finding a balanced 2-partition. *Soft Computing*, in press.
4. M.A. Gutiérrez-Naranjo, M.J. Pérez-Jiménez, F.J. Romero-Campero: Solving SAT with membrane creation. *CiE 2005: New Computational Paradigms, Amsterdan* (S. Barry Cooper, B. Lowe, L. Torenvliet, eds.), Report ILLC X-2005-01, University of Amsterdam, 82–91.
5. M.A. Gutiérrez-Naranjo, M.J. Pérez-Jiménez, F.J. Romero-Campero: A linear solution for QSAT with Membrane Creation. Submitted, 2005.
6. A. Păun, Gh. Păun: The power of communication: P systems with symport/antiport. *New Generation Computing*, 20, 3 (2002), 295–305
7. Gh. Păun, Computing with membranes: Attacking **NP**–complete problems. In *Unconventional Models of Computation* (I. Antoniou, C.S. Calude, M.J. Dinneen, eds.), 2000, 94–115.
8. Gh. Păun: Computing with membranes. *Journal of Computer and System Sciences*, 61, 1 (2000), 108–143.
9. Gh. Păun, M.J. Pérez-Jiménez: Recent computing models inspired from biology: DNA and membrane computing. *Theoria*, 18, 46 (2003), 72–84.
10. Gh. Păun: Further open problems in membrane computing. *Proceedings of the Second Brainstorming Week on Membrane Computing* (Gh. Păun, A. Riscos, A. Romero, F. Sancho, eds.), Report RGNC 01/04, University of Seville, 2004, 354–365.

11. M.J. Pérez–Jiménez: An approach to computational complexity in Membrane Computing. En G. Mauri, Gh. Păun, M. J. Pérez-Jiménez, Gr. Rozenberg, A. Salomaa (eds.) *Membrane Computing, 5th International Workshop, WMC5, Revised Selected and Invited Papers.* LNCS 3365 (2005), 85-109.
12. M.J. Pérez–Jiménez, A. Riscos–Núñez: A linear time solution to the Knapsack problem using active membranes. *Membrane Computing* (C. Martín-Vide, Gh. Păun, G. Rozenberg, A. Salomaa, eds.). LNCS 2933 (2004), 250–268.
13. M.J. Pérez-Jiménez, A. Riscos-Núñez: Solving the Subset-Sum problem by active membranes. *New Generation Computing*, in press.
14. M.J. Pérez-Jiménez, A. Riscos-Núñez: A linear solution for the Knapsack problem using active membranes. In *Membrane Computing* (C. Martín-Vide, G. Mauri, Gh. Păun, G. Rozenberg, A. Salomaa, eds.). LNCS 2933 (2004), 250–268.
15. M.J. Pérez-Jiménez, F.J. Romero-Campero: Solving the BIN PACKING problem by recognizer P systems with active membranes. *Proceedings of the Second Brainstorming Week on Membrane Computing* (Gh. Păun, A. Riscos, A. Romero and F. Sancho, eds.), Report RGNC 01/04, University of Seville, 2004, 414–430.
16. M.J. Pérez-Jiménez, A. Romero-Jiménez, F. Sancho-Caparrini: A polynomial complexity class in P systems using membrane division. *Proceedings of the 5th Workshop on Descriptional Complexity of Formal Systems, DCFS 2003* (E. Csuhaj-Varjú, C. Kintala, D. Wotschke, G. Vaszil, eds.), 2003, 284-294.
17. P. Sosik: The computational power of cell division. *Natural Computing*, 2, 3 (2003), 287–298.
18. W.M. Spears: *Evolutionary Algorithms. The role of mutation and recombination.* Natural Computing Series, Springer, 2000.
19. ISI web page `http://esi-topics.com/erf/october2003.html`
20. P systems web page `http://psystems.disco.unimib.it/`