

# Membrane Computing — Current Results and Future Problems<sup>\*</sup>

Francesco Bernardini<sup>1</sup>, Marian Gheorghe<sup>1</sup>, Natalio Krasnogor<sup>2</sup>,  
and German Terrazas<sup>2</sup>

<sup>1</sup> Department of Computer Science, University of Sheffield, UK  
{F.Bernardini, M.Gheorghe}@dcs.shef.ac.uk

<sup>2</sup> ASAP Group, School of Computer Science and IT, University of Nottingham, UK  
Natalio.Krasnogor@Nottingham.ac.uk  
gzt@cs.nott.ac.uk

In the last decade and especially after Adleman's experiment [1] a number of computational paradigms, inspired or gleaned from biochemical phenomena, are becoming of growing interest building a wealth of models, called generically Molecular Computing. New advances in, on the one hand, molecular and theoretical biology, and on the other hand, mathematical and computational sciences promise to make it possible in the near future to have accurate systemic models of complex biological phenomena. Recent advances in cellular Biology led to new models, hierarchically organised, defining a new emergent research area called Cellular Computing.

P-systems represent a class of distributed and parallel computing devices of a biological type that was introduced in [14] which are included in the wider field of cellular computing. Several variants of this model have been investigated and the literature on the subject is now rapidly growing. The main results in this area show that P-systems are a very powerful and efficient computational model [15], [16], [13]. There are variants that might be classified according to different criteria. They may be regarded as language generators or acceptors, working with strings or multisets, developing synchronous or asynchronous computation. Two main classes of P-systems can be identified in the area of membrane computing [15]: *cell-like P-systems* and *tissue-like P-systems*. The former type is inspired by the internal organization of living cells with different compartments and membranes hierarchically arranged; formally this structure is associated with a tree. Tissue P-systems have been motivated by the structure and behaviour of multicellular organisms where they form a multitude of different tissues performing various functions [2]; the structure of the system is instead represented as a graph where nodes are associated with the cells which are allowed to communicate alongside the edges of the graph.

More recently, a notion of population P-systems has been introduced [3], [4] as a model for tissue P-systems where the structure of the underlying graph can be modified during a computation by varying the set of nodes and the

---

<sup>\*</sup> We are indebted to the anonymous referees for their comments that allowed us improving the readability of the text.

set of edges in the graph. Specifically, nodes are associated with cells, each of them representing a basic functional unit of the system, and edges model bonds among these cells that are dynamically created and destroyed. Although mainly inspired by the cell behaviour in living tissues, population P-systems may be also regarded as an abstraction of a population of bio-entities aggregated together in a more complex bio-unit (e.g. social insects like ants, bees, wasps etc, organized in colonies or bacteria of different types). This is the main reason why we use the term population instead of tissue albeit the term cell is retained to denoting an individual in the system. The concept also recalls other similar computational models: grammar systems [8], eco-grammar systems [9], or more recently, networks of parallel/evolutionary processors [10].

Universality results have been obtained [4] for a number of variants of population P-systems. The following different rules are considered: transformation rules for modifying the objects that are present inside the cells, communication rules for moving objects from a cell to another one, cell division rules for introducing new cells in the system, cell differentiation rules for changing the types of the cells, and cell death rules for removing cells from the system. As well as this, bond making rules are considered that are used to modify the links between the existing cells (i.e., the set of edges in the graph) at the end of each step of evolution performed by means of the aforementioned rules. In other words, a population P-system in [4] is basically defined as an evolution-communication P-system [7] but with the important difference that the structure of the system is not rigid and it is represented as an arbitrary graph. In particular, bond making rules are able to influence cell capability of moving objects from a place to another one by varying the set of edges in the underlying graph.

Another interesting variant of population P-systems is obtained by considering the general mechanism of cell communication based on signal molecules as a mechanism for triggering particular transformations inside of a cell once a particular signal-object has been received from some other cell in the system [3]. This leads to a notion of population P-systems where the sets of rules associated with the cell can vary according to the presence of particular objects inside and outside the cells. Yet again, the introduction of this mechanism is motivated by the features shared by biological systems at various levels where the behaviour of an individual is affected both by its internal state and by the external stimuli received. Some results concerning the power of population P-systems with a rule activating mechanism have been obtained [5].

Further developments of the area of population P-systems are expected to cover alternative ways of defining the result of a computation and the use of string objects. Population P-systems in fact attempt to model aspects of biological systems formed by many different individual components cooperating in a coherent way for the benefit of the system as a whole; a more appropriate notion of computation is therefore necessary in order to characterise the emergent behaviour of the system. Existing approaches in the area of grammar system such parallel communicating grammar systems [8] or eco-grammar systems [9], rely on the use of a single sentential form that is rewritten in parallel by differ-

ent interacting/cooperating grammar components. In particular, in the case of eco-grammar systems, this sentential form is associated with the environment and it can be rewritten both by rules corresponding to action taken from the individual components in the system and by dedicated rules associated with the environment. In a similar way, we can consider string-processing population P-systems where the result of a computation is given by a string (or a language) produced in the environment at the end of a computation. However, with respect to grammar systems, population P-systems present some other interesting features like the possibility of moving objects from a place to another one, the possibility of forming bonds among the cells, the possibility of introducing new cells in the system by means of cell division, which need to be formalised for the particular case of string objects. In this respect, we aim to present some reasonable variants of population P-systems with string objects.

Apart from being a very interesting research area in theoretical computer science P-systems have been used in modelling different biological systems. One of the most exciting biological system is represented by the quorum sensing phenomenon occurring in bacteria.

Recent advances in analytical biotechnology, computational biology, bioinformatics and computational modeling promise ever deeper understanding of the complexity of biological systems, particularly the computations they perform in order to survive in dynamic and hostile environments. These insights will ultimately enable researchers to harness the living cell as a computational device with its own sensors, internal states, transition functions, actuators, etc, and to program them as "nano-bots" for particular tasks such as targeted drug delivery, chemical factories, nano-structures repairs, bio-film scaffolding and self-assembling, to name but a few.

Quorum sensing (QS) have been described as "the most consequential molecular microbiology story of the last decade" [20, 6]. It relies on the activation of a sensor kinase or response regulator protein by a diffusible, low molecular weight, signal molecule (a "pheromone" or "autoinducer") [18]. In QS, the concentration of the signal molecule reflects the number of bacterial cells in a particular niche and perception of a threshold concentration of that signal molecule indicates that the population is "quorated" i.e. ready to make a behavioral decision [19].

An overview on Quorum Sensing in *P. aeruginosa* with comments on some of the techniques that have been used to model this phenomenon as well as a more "computationally flavoured" approach for QS and some research tracks which could benefit from an in-depth understanding of QS are presented in [11]

This perspective on modelling biological systems at the level mentioned before is investigated by describing various bio-components as agents. An agent is a fairly complex computer system that is situated in some environment and is capable of flexible, autonomous actions in order to meet its design objectives [12]. The extreme complexity of agent systems is due to substantial differences between the attributes of their components, high computational power required by the processes running within these components, huge volume of data manipulated by these processes and finally possibly extensive amount of communication

in order to achieve coordination and collaboration. The use of a computational framework that is capable of modelling both the dynamic aspects (i.e. the continuous change of agents' states together with their communication) and the static aspects (i.e. the amount of knowledge and information available), will facilitate modelling and simulation of such complex systems.

## Acknowledgments

The research of FB and MG has been supported by the Engineering and Physical Science Research Council (EPSRC) of United Kingdom, Grant GR/R84221/01. NK acknowledges the EPSRC (GR/T07534/01) and BBSRC (BB/C511764/1) for (partially) funding his research in bioinformatics.

## References

1. Adleman, L.M. 1994. Molecular computation of solutions to combinatorial problems. *Science*, 226, 1021-1024
2. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P. 2002. *The Molecular Biology of The Cell*. Fourth Edition. Garland Publ. Inc., London
3. Bernardini, F., Gheorghe, M. 2004. Cell Communication in Tissue P-systems and Cell Division in Population P-Systems. In [17], 74-91
4. Bernardini, F., Gheorghe, M. 2004. Population P-Systems. *Journal of Universal Computer Science*, 10, 509-539
5. Bernardini, F., Gheorghe, M. 2005. Cell Communication in Tissue P-Systems: Universality Results. *Soft Computing* (to appear)
6. Busby, S., de Lorenzo, V. 2001. Cell regulation - putting together pieces of the big puzzle. *Curr. Op. Microbiol.*, 4, 117-118.
7. Cavaliere, M. 2003. Evolution Communication P-Systems. In [16], 134-145 and in [17], 206-223.
8. Csuha-j-Varjú, E., Dassow, J., Kelemen, J., Păun, Gh. 1997. Grammar Systems. A Grammatical Approach to Distribution and Cooperation. Gordon and Breach, London
9. Csuha-j-Varjú, E., Kelemen, J., Kelemenova, A., Păun, Gh. 1997. Eco-Grammar Systems: A Grammatical Framework for Studying Life-Like Interactions. *Artificial Life*, 3, 1-28.
10. Csuha-j-Varjú, E., Salomaa, A., 1997. Networks of Parallel Language Processors. In *New Trends in Formal Languages. Control, Cooperation, and Combinatorics*. In Păun, Gh., Salomaa, A. (eds), *Lecture Notes in Computer Science*, 1218, Springer-Verlag, Berlin, Heidelberg, New York, 299-318
11. Krasnogor, N., Gheorghe, M., Terrazas, G., Diggle, S., Williams, P., Camara, M. 2005. *Bulletin of the EATCS* (to appear)
12. Jennings, N.R. 2000. On agent-based software engineering. *Artificial Intelligence*, 117, 277-296.
13. Martin-Vide, C., Mauri, G., Păun, Gh., Rozenberg, G., Salomaa, A. (eds) 2004. Membrane computing. International workshop, WMC 2003, Tarragona, Spain, July 2003. Revised papers. *Lecture Notes in Computer Science*, 2933, Springer, Berlin Heidelberg New York

14. Păun, Gh. 2000. Computing with Membranes. *Journal of Computer and System Sciences*, 61, 108-143
15. Păun, Gh. 2002. *Membrane computing. An introduction*. Springer, Berlin Heidelberg New York
16. Păun, Gh., Rozenberg, G., Salomaa, A., Zandron, C. (eds) 2003. *Membrane computing. International workshop, WMC-CdeA 02, Curtea de Arges, Romania, August 19-23, 2002. Revised papers*. *Lecture Notes in Computer Science*, 2597, Springer, Berlin Heidelberg New York
17. Păun, Gh., Riscos-Nunez, A., Romero-Jimenez, A., Sancho-Caparrini, F. (eds) 2004. *Second brainstorming week on membrane computing, Seville, 2-7 February 2004. Technical Report 01/2004*, Department of Computer Science and Artificial Intelligence, University of Seville, Spain
18. Swift, S., Downie, J.A., Whitehead, N.A., Barnard, A.M.L., Salmond, G.P.C., Williams, P. 2001. Quorum sensing as a population-density-dependent determinant of bacterial physiology. *Adv Micro Physiol*, 45, 199-270.
19. Williams, P., Camara, M., Hardman, A., Swift, S., Milton, D., Hope, V.J., Winzer, K., Middleton, B., Pritchard, D.I., Bycroft, B.W. 2000. Quorum sensing and the population dependent control of virulence. *Phil. Trans Roy Soc London B*, 355(1397), 667-680.
20. Winzer, K., Hardie, K.H., Williams, P. 2002. Bacterial cell-to-cell communication: sorry can't talk now – gone to lunch!. *Curr Op. Microbiol*, 5, 216-222.