

# MEMBRANE COMPUTING: A GENTLE INTRODUCTION

Matthew N. O. Sadiku, Adebowale E. Shadare, and Sarhan M. Musa

Roy G. Perry College of Engineering

Prairie View A&M University

Prairie View, TX 77446

---

## ABSTRACT

Membrane computing is an emerging branch of natural computing, i.e. computing which deals with what is going on in nature and inspired by nature. It is inspired from the functioning of living cells to generate models and simulations of cellular phenomena. It provides distributed parallel computational devices called membrane systems, which are inspired in biological features of living cells. The paper provides a brief introduction to membrane computing.

**Key words:** Membrane Computing, Natural Computing, Cellular Computing, P Systems.

---

## 1. INTRODUCTION

One may regard the whole history of computer science as a record of continuous attempts to discover, study, and implement computing ideas. Membrane computing (MC) is part of this intellectual journey [1]. It is a young branch of natural computing, which involves computer science, mathematics, biology and artificial intelligence. It uses cellular membrane systems (seen as computing devices), which are known as P systems after the Romanian scientists Gheorghe Păun who initiated the model in 1998. Since then, the literature of this area has grown very fast.

The basic element of a P system is its membrane structure, which can be a hierarchical arrangement of membranes, as in a cell or a net as in a tissue or a neural net. Life is directly related to cells. The cell is the smallest living thing. Cell means membranes [2]. Biological cell membranes are complex structures that are capable of carrying out diverse functions, functions which are controlled by the membrane's main building blocks (lipids and proteins). The interaction between proteins and lipids is responsible for defining cell membrane function [3].

## 2. CONCEPT OF MEMBRANE COMPUTING

Membrane computing deals with computing models abstracted from the structure and the functioning of living cells. Each living cell is being compartmentalized by membranes. A membrane system is shown in Figure 1 [2]. Each membrane determines a region; and the region outside the membrane system is called the environment. A membrane without any other membrane inside is said to be elementary. The skin membrane plays only the role of communication environment. The membranes can change their permeability, they can dissolve, and they can divide. They allow certain molecules to pass through (such as waste products to leave) and certain nutrients to enter. Each cell has multisets of objects which follow some evolution rules to decide the next state. The rules dictate how are created, removed, or migrate across regions.

Being a model inspired by biology, membrane computing uses objects as transporting mechanisms through membranes. The basic model of MC consists of a hierarchical structure composed by several membranes. All the variants of membrane computing (cell-like, tissue-like and neural-like) share the same simulation structure. Membrane computing is inherently a parallel computing model because communication between the multisets and objects within the regions of a membrane takes place concurrently [4].

The graphic processing unit (GPU) has been used as a parallel tool to implement membrane computing. Another hardware architecture for parallel implementation is the compute unified device architecture (CUDA). Membrane computing is an unconventional computing model just like quantum computing and DNA computing. It may also be regarded as a branch of molecular computing that aims to develop models and paradigms that are biologically motivated. The initial objective of membrane computing was to learn from cell biology something possibly useful to computer science. Membrane computing aims to abstract computational model from the structure and functioning of living cells. The computational model is known as membrane system or P system. Generally, there are three components in P system: (i) membrane structure, (ii) multi-sets and (iii) evolution rules [5].

### 3. APPLICATIONS

Like other natural computing techniques, MC aims at solving intractable (NP-complete) problems. Over the years, a variety of membrane system (or P systems) have been proposed. Most applications use cell-like P systems and tissue-like P systems.

Membrane computing is employed in global optimization method and is regarded as a optimizing program and strategy. Therefore, it is more like an evolutionary algorithm.

Evolutionary membrane computing explores the interactions between MC and evolutionary computation [6]. Membrane computing can be used to design an optimal infinite impulse response (IIR) filter, which is a challenging optimization problem.

The MC paradigm can also be used to enhance bee algorithm-based feature subset selection method in intrusion detection system. Other applications of MC are found in computer graphics, linguistics, cryptography, medicine and biology, modeling metapopulation, approximate optimization [7].

### 4. CONCLUSION

Membrane computing is a new unconventional, fascinating computing model that abstracts from the structure and functionality of biological entities like living cells and tissues. It studies the properties and applications of theoretical computing devices known as P systems, which are basically an abstraction of the functioning of a living cell. More information about membrane computing can be found in books in [2, 8-14] and the journal exclusively devoted to it: *Journal of Membrane Computing*.

### REFERENCES

- [1] G. Paun and M. J. Pérez-Jiménez, "Membrane computing: Brief introduction, recent results and applications," *BioSystems*, vol. 85, 2006, pp. 11-22.
- [2] G. Ciobanu, M. J. Pérez-Jiménez, and G. Păun (eds.), *Applications of Membrane Computing*. Springer, 2006, p. 8.
- [3] R. Askhar, "Unlocking the secrets of cell membranes," *ACM SIGCAS Computers & Society*, vol. 46, no. 2, August 2016, pp. 5-8.
- [4] K. I. Rufai, R. C. Muniyandi, and Z. A. Othman, "Improving bee algorithm based feature selection in intrusion detection system using membrane computing," *Journal of Networks*, vol. 9, no. 3, March 2014, pp. 523-529.
- [5] Z. Gao and C. Zhang, "MCIR: A multi-modal image registration algorithm based on membrane computing," *Proceedings of International Conference on Computing Intelligence and Information System*, 2017, pp. 263-269.
- [6] G. Zhang et al., "Evolutionary membrane computing: A comprehensive survey and new results," *Information Sciences*, vol. 279, 2014, pp. 528-551.
- [7] G. Paun, "A quick introduction to membrane computing," *The Journal of Logic and Algebraic Programming*, vol. 79, 2010, pp. 291-294.
- [8] G. Paun, *Membrane Computing: An Introduction*. Berlin: Springer, 2002. (Also available in Chinese.)
- [9] P. Frisco, M. Gheorghe, and M. J. Pérez-Jiménez, *Applications of Membrane Computing in Systems and Synthetic Biology*. Springer, 2014.
- [10] G. Paun, G. Rozenberg, and A. Salomaa, *The Oxford Handbook of Membrane Computing*. New York: Oxford University Press, 2010.
- [11] G. Rozenberg et al. (eds.), *Membrane Computing*. Springer, 2015.
- [12] A. Ionescu, *Membrane Computing: Traces, Neural Inspired Models, Controls*. OmniScriptum, 2009.
- [13] G. Zhang, M. J. Pérez-Jiménez, and M. Gheorghe, *Real-life Applications with Membrane Computing*. Springer, 2017.
- [14] P. Chinedu A *Study of the Application of Multiset to Membrane Computing*. LAP Lambert Academic Publishing, 2011.

### ABOUT THE AUTHORS

**Matthew N.O. Sadiku** is a professor at Prairie View A&M University, Texas. He is the author of several books and papers. He is an IEEE fellow. His research interests include computational electromagnetics and computer networks.

**Adebowale Shadare** is a doctoral student at Prairie View A&M University, Texas. He is the author of several papers.

**Sarhan M. Musa** is a professor in the Department of Engineering Technology at Prairie View A&M University, Texas. He has been the director of Prairie View Networking Academy, Texas, since 2004. He is an LTD Spring and Boeing Welliver Fellow.

