Membrane Computing Aggregation (MCA): An Upgraded Framework for Transition P-Systems

Alberto Arteta[1](✉), Luis Fernando Mingo[2], Nuria Gomez[2],
and Yanjun Zhao[1]

[1] Computer Science, Troy University, Troy, USA
{aarteta,yjzhao}@troy.edu
[2] Computer Science, Polytechnic University of Madrid,
Crtra Valencia km 7, Madrid, Spain
{lfmingo,ngomez}@eui.upm.es

Abstract. MCA (Membrane computing aggregation is experimental computational frame. It is inspired by the inner properties of membrane cells (Bio-inspired system). It is capable of problem solving activities by maintaining a special, "meaningful" relationship with the internal/external environment, integrating its self-reproduction processes within the information flow of incoming and outgoing signals. Because these problem solving capabilities, MCA admits a crucial evolutionary tuning by mutations and recombination of theoretical genetic "bridges" in a so called "aggregation" process ruled by a hierarchical factor that enclosed those capabilities. Throughout the epigenetic capabilities and the cytoskeleton and cell adhesion functionalities, MCA model gain a complex population dynamics specifics and high scalability. Along its developmental process, it can differentiate into meaningful computational tissues and organs that respond to the conditions of the environment and therefore "solve" the morphogenetic/configurational problem. MCA, above all, represents the potential for a new computational paradigm inspired in the higher level processes of membrane cells, endowed with quasi universal processing capabilities beyond the possibilities of cellular automata of and agent processing models.

Keywords: Membrane system · Membrane computing · Natural computing

1 Introduction

In spite of all the recent emphasis and advancements in systems biology, synthetic biology, and network science about modelling of gene networks, protein networks, metabolic and signaling networks, etc. some of the most important computational properties of membrane cells have not been grappled and "abstracted" et: scalability, tissular differentiation, and morphogenesis - i.e., the capability to informationally transcend the cellular level and organize higher level information processes by means of heterogeneous populations of membrane cells organized as "computational tissues and organs".

A. Compagnoni et al. (Eds.): BICT 2019, LNICST 289, pp. 195–207, 2019.
https://doi.org/10.1007/978-3-030-24202-2_15

Synthetic biology has become extraordinarily active in the manufacture of very simple and robust models and simulations tailored to the realization problems of circuits and modules in vivo, mostly addressed to prokaryotic systems. In the first wave of these studies, very basic elements such as promoters, transcription factors, and repressors were combined to form small modules with specified behaviors. Currently modules include switches, cascades, pulse generators, oscillators, spatial patterns, and logic formulas [1]. The second wave of synthetic biology is integrating basic parts and modules to create systems-level circuitry. genomes and synthetic life organisms are envisioned, and application-oriented systems are contemplated. Different computational tools and programming abstractions are actively developed (the Registry of Standard Biological Parts; the Growing Point Language GLP; the Origami Shape Language OSL, the PROTO bio programming language, etc. See details at the Open Wetware site). Evolving cell models of prokaryotes have also been addressed [2, 3]. As some have put, "systems broaden the scope of synthetic biology designing synthetic circuits to operate in reliably in the context of differentiating and morphologically complex membrane cells present unique challenges and opportunities for progress in the field" [4]. However, very few synthetic biology researchers do contemplate using systems.

In systems biology, a plethora of modelling developments have been built around signaling pathways, cell cycle control, topologies of protein networks, transcriptional networks, etc. There is a relatively well consolidated thinking, in part due to traditional physiology and to systems science and control theory which were at the origins of this new field, of going "from genes to membrane cells to the whole organ" as D. Noble has done for heart models [5]. The integration of proteins to organs has also been promoted by bioinformatic-related projects such as the "Physiome Project" [6]. Important works have been done in the vicinity of "network science" in order to make sense of gene networks, protein networks, transcription networks, complexes formation, etc. For instance, about how is dynamically organized modularity in the yeast protein-protein interaction network [7], it was uncovered that two types of "hub" contribute to the organized modularity of the proteome: "party" hubs which interact with their partners simultaneously, and "date" hubs, which bind their different partners at different times and locations (we will see later on the importance of the discussion on "modularity" in the evo-devo field). Predictive models of mammalian membrane cells have been described using graph theory, assembling networks and integrative procedures [8]. Important systems biology compilations and far-reaching cellular models have been made by [9], Kitano [10–12] It has to be emphasized that concerning the views advocated in this proposal, most of systems biology works depart from the goal of "abstracting computational power out from systems" and focus instead on "applying computational power to analyze the organization of systems." Notwithstanding the foregoing, studies such as A. [13] on bacteria as computers making computers [14], on the operating system of bacteria could be considered as forerunners in the former direction.

In the science of development (the "evo-devo" discipline) most of the emphasis has been on modularity. What it exactly means in developmental terms is still a matter of controversy [15–17]; but undoubtedly modularity refers to the capability of cellular networks to dissociate networked processes at a lower level and to recombine or redeploy them at the higher level of the multicellular organism. Thanks to the cellular signaling system, the genetic switches, the cytoskeleton, and some other topobiological mechanisms [18], the unitary network of cellular processes integrated into the cell-cycle may be broken down into coherent modules and be performed separately in different membrane cells within differently specialized tissues [19]. This implies a flexible organization for the deployment of biomolecular processing modules, which actually are "cut" differently in each tissue along the developmental process, due also to chromatin remodelling during development [20]. Interestingly, not only differentiation but also morphology becomes an instance of the scalable "modular" processing, throughout the "tensegrity" emergent property and the ontogenetic arrangement of symmetry breakings in a force field. The emergence of cellular bauplans where signaling, force fields, and cytoskeletal mechanical modes conspire together to create but a few basic morphologies for membrane cells, depending also on the populations present, seems to be another important consequence [21]. Interestingly, complex morphologies obtained out from Turing diffusion model have been cogently discussed as a result of cell-to-cell developmental interactions [22]. Currently, the evo-devo field accumulates a considerable mass of biomolecular-organization-facts, poorly conceptualized yet, to be computationally "abstracted" in the perspective of MCA advancement.

In the fields closer to computer science and Biocomputing, it has been important the introduction of the agent based approach (as pioneered by W. Fontana and others), which uses sets of rules to define relationships between cellular components substituting for the simple Boolean networks and differential equations used up to now. Proteins and other biomolecules become molecular "automata" and the aggregate behavior that emerges out from these models is the combinatorial expression of all those automata doing their specific micro-functions [23]. This approach shows promise for "evolvable" advancement of network models endowed with the flexible modularity property. It is somehow close to the already mentioned predictive models of mammalian membrane cells that are using graph theory, assembling networks and integrative procedures [24]. New generations of cellular models (of "automata") have been developed too, with powerful data content and with potential for modelling multicellular systems in a general way, supporting user-friendly in silicon experimentation and discovery of emergent properties [25]. Under the approach of Artificial Embryology, a developmental system has been obtained by means of cellular automata systems capable of following "rewriting rules" procedures, emulating elementary morphologies and multicellular distributions [26].

As for the developments in molecular Biocomputing, the idea that bio-molecules (DNA, RNA, proteins) might be used for computing already emerged in the fifties and was reconsidered periodically with more and more arguments which made it more viable. But the definitive confirmation came in 1994 [27] when in [27] successfully accomplished the first experimental close connection between molecular biology and computer science. He described how a small instance of a computationally intractable problem might be solved via a massively parallel random search using molecular biology methods. An important part of this project is focusing on bio-inspired models of computation abstracted from the very complex networks in living systems. Its goal is to investigate several aspects of these models particularly focused on connections between theoretical models and natural (biological) networks.

The main topics are: Computational aspects (computational power, structural and description complexity).

Several new directions of research have been initiated in the last decade: computing devices inspired from the genome evolution [28–30] with an explosive development, evolutionary systems based on the behavior of cell populations [31]) computing models simulating the process of gene assembly in ciliates Swarm computation is mainly based on the same idea: a swarm is a group of mobile biological organisms wherein each individual communicates with others by acting on its local environment [32]. Regarding applicative models there are many attempts to update Cells computing paradigm in [33–36] among others.

2 Membrane Computing

A Transition P System of degree n $n > 1$ is a construct $\prod = (V, \mu, \omega_1, .., \omega_n, (R_1, \rho_1), ..(R_n, \rho_n), i_0)$

Where:

V is an alphabet; its elements are called objects;

μ is a membrane structure of degree n, with the membranes and the regions labeled in a one-to-one manner with elements in a given set; in this section we always use the labels 1, 2, n;

$\omega_i \; 1 \leq i \leq n$, are strings from V^* representing multisets over V associated with the regions $1, 2, \ldots,$ n of μ

$R_i \; 1 \leq i \leq n$, are finite set of evolution rules over V associated with the regions $1, 2, \ldots,$ n of μ; ρ_i is a partial order over $R_i \; 1 \leq i \leq n$, specifying a priority relation among rules of R_i. An evolution rule is a pair (u, v) which we will usually write in the form $u \rightarrow v$ where u is a string over V and v = v′ or v = v′ δ where v′ is a string over $(V \times \{here, out\}) \cup \left(V \times \left\{in_j \; 1 \leq j \leq n\right\}\right)$, and δ is a special symbol not in. The length of u is called the radius of the rule $u \rightarrow v$

i_o is a number between 1 and n which specifies the output membrane of $\prod$.

Let U be a finite and not empty set of objects and N the set of natural numbers. A multiset of objects is defined as a mapping:

$$M : U \rightarrow N$$
$$a_i \rightarrow u_1$$

Where a_i is an object and u_i its multiplicity.

As it is well known, there are several representations for multisets of objects (Fig. 1).

$$M = \{(a_1, u_1), (a_2, u_2), (a_3, u_3)\ldots\} = a_1^{u_1} \cdot a_2^{u_2} \cdot a_n^{u_n} \ldots\ldots$$

Note: Initial Multiset is the multiset existing within a given region in where no application of evolution rules has occurred yet.

Definition Evolution rule with objects in U and targets in T.

Evolution rule with objects in U and targets in T is defined by $r = (m, c, \delta)$ where $m \in M(U), c \in M(U \times T) \; and \; \delta \in \{to\,dissolve, not\,to\,dissolve\}$

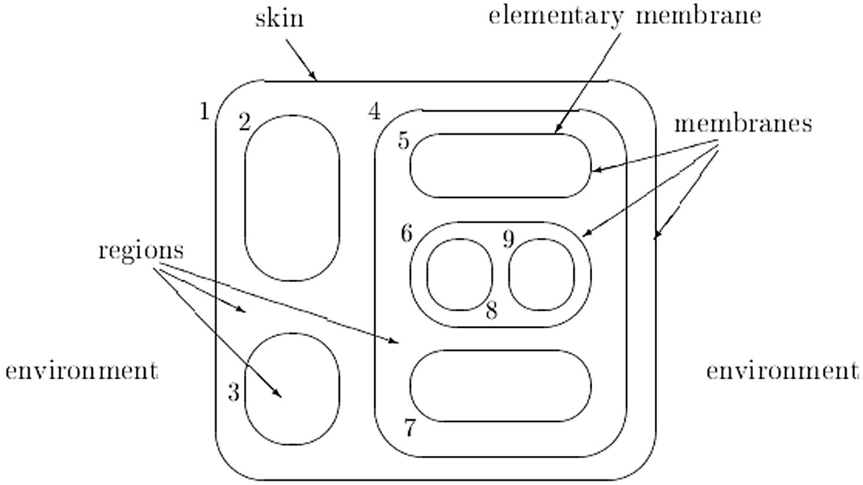

Fig. 1. P-system structure definition multiset of objects

From now on ‘c’ will be referred a s the consequent of the evolution rule ‘r’.

Note The set of evolution rules with objects in U and targets in T is represented by R (U, T).

Definition Multiplicity of an object in a multiset of objects M(U)

Let $a_i \in U$ be an object and let $m \in M(U)$ be a multiset of objects. The multiplicity of an object is defined over a multiset of objects such as:

$$||_{a_i}: U \times M(U) \to N$$
$$(a_i, m) \to |m|_{a_i} = n|(a_i, n) \in m$$

Definition Multiplicity of an object in an evolution rule r

Let $a_i \in U$ be an object and let $R(U, T)$ be a multiset of evolution rules. Let $r = (m, c, \delta) \in R(U, T)$ where $m \in M(U), c \in M(U \times T)$ *and* $\delta \in$ $\{$*to dissolve*, *not to dissolve*$\}$

The multiplicity of an object is defined over an evolution rules such as:

$$||_{a_i}: U \times R(U, T) \to N$$
$$(a_i, r) \to |m|_{a_i} = n|(a_i, n) \in m$$

P-system evolution

Let C_i be the consequent of the evolution rule r_i. Thus, the representation of the evolution rules is:

$$\begin{array}{l} r_1 : a_1^{u_{11}} a_2^{u_{12}} \ldots a_n^{u_{1n}} \rightarrow C_1 \\ r_2 : a_1^{u_{21}} a_2^{u_{22}} \ldots a_n^{u_{2n}} \rightarrow C_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots \rightarrow \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ r_m : a_1^{u_{m1}} a_2^{u_{m2}} \ldots a_n^{u_{mn}} \rightarrow C_m \end{array} \tag{1}$$

P-systems evolve, which makes it change upon time; therefore, it is a dynamic system. Every time that there is a change on the p-system we will say that the p-system is in a new transition. The step from one transition to another one will be referred to as an evolutionary step, and the set of all evolutionary steps will be named computation. Processes within the p-system will be acting in a massively parallel and non-deterministic manner. (Similar to the way the living cells process and combine information).

3 The Upgrade

The proposal is a new computational paradigm based on Membrane cells, scalable ones which are capable to produce "computational tissues and organs". The organization of such computational tissues and organs is inspired by the emerging informational properties of biomolecular networks and will be based on scalable "membrane cells" guided by functional rules similar to the biological ones (molecular recognition, self-assembly and topo biology-theory rules).

The direct inspiration from the membrane cells is precisely the breakthrough of the MCA project. By building computational tissues our proposal makes an evolutionary jump with respect of today research in this field, mainly focused on aggregates of unicellular organisms (e.g. bacteria). Far from modelling and simulating the cellular processes, our computational paradigm will be a clear abstraction of the basic mechanisms and computational capabilities of the membrane cells and tissues, in order to solve complex problems in a new (bioinspired) way.

Real tissues display far more complex properties (emergent properties) than the sum of the properties of the individual membrane cells they are made from. In the same way, the emergent properties and functions of our membrane cells and computational tissues will be used for the resolution of real problems, impossible to be appropriately solved by conventional methods: not only biological morphogenesis, but also evolution of economic systems and prediction of crisis, optimization of "industrial ecologies", analysis of the dynamics of social interactions and conflicts, ecosystem disturbances, etc., that are more complex than combinatorial optimization, as well as other classical NP-Complete ones.

Our "membrane cells" will be a species of "proto-membrane cells" and a far objective of the project is also the ex-novo synthesis of "membrane cells" and tissues performing as living computational biomolecular networks. The long-term vision that motivates this breakthrough is to build new information processing devices with evolving capabilities, which will adapt themselves to the complexity of the problems. In particular, we foresee a synthetic approach to build computational membrane cells and tissues, and to create computational bio-inspired devices of higher complexity (tissues-organs). A far future objective of the project goes beyond the mathematical, software and hardware tools. It is to obtain in lab synthesized "living" information processing systems based on artificial "membrane cells" and hybrid systems combining living components (our "synthesized membrane cells") and non-living elements (e.g. silicon-based).

MCA approach is the most appropriate to deal with extremely complex problems that will be crucial in the future. It shows potential to go beyond classical Biocomputing strategies such as self-reproducing machines, cellular automata, perceptron's & neural networks, genetic algorithms, adaptive computing, bacteria-based computation, artificial membrane cells, etc. Specifically, a new generation of natural computing could be built, based upon the scalable "membrane cells" with problem solving capacity in very different realms: biomaterials and bioengineering, non-linear parallel processing, design of bioinspired systems, modelling of economic, industrial and financial systems, optimization strategies in social settings, etc.

For the achievement of our long-term objectives we need to:

analyze the wide amount of existing knowledge regarding one of the deepest sources of biocomputational power, the topological and flexible networking properties of biomolecular scalable modules in membrane cells,

realize an abstraction of the basic mechanisms and computational capabilities of the membrane cells both at sub cellular and networking level, and develop formal models to be used in new information processing technologies, basically based on combinatory processes of protein domains and genetic switches, together with cytoskeleton dynamics and topobiology-theory,

use the above proposed models to create scalable "/proto membrane cells" and abstract-formal "evolvable" cellular networks and computational tissues & organs endowed with these flexible modularity properties.

For our far final objective we need to obtain in lab proof that synthesis of new forms of living "membrane cells" in an inverse process: "membrane cells and tissues" => "theoretical abstract/formal models" => "artificial membrane cells and tissues" => "in lab synthesized living membrane cells" is possible. MCA breakthrough is an essential step towards the achievement of our long-term vision because it will set the theoretical basis and develop the experimental tools for the creation of the scalable membrane cells, computational tissues and organs (both abstract and living ones).

4 MCA System

A MCA is a set and a set of aggregation rules among membranes. The set of aggregation rules are not fully integrated with the evolution rules of a given p-System but establishes the correlation between 2 given membrane models by deciding the way 2 or mere P-systems are being aggregated. The rules can be defined as a Matrix relation

$$\varphi_1(k_1, k_2, .., k_m,) \equiv \begin{pmatrix} u_{11} & u_{21} & \ldots u_{m1} \\ u_{12} & u_{22} & \ldots u_{m2} \\ \ldots & \ldots & \ldots \\ u_{1n} & u_{2n} & \ldots u_{mn} \end{pmatrix} \begin{pmatrix} k_1 \\ k_2 \\ \ldots \\ k_m \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ \ldots \\ u_n \end{pmatrix} \tag{2}$$

Where $\varphi 1$(k) is the aggregation relation and is defined by the association of n P-systems, k determines the aggregation rules of each component in every p-system I and U are the component (objects).

Evolution rule application phase

This phase is the one that has been implemented following different techniques.

In every region within a p-system, the evolution rules application phase is described as follows:

Rules application to a multiset of object in a region is a transforming process of information which has input, output and conditions for making the transformation.

Given a region within a p-system, let U $= \{a_i | 1 \leq i \leq n\}$ be the alphabet of objects, m a multiset of objects over U and R(U, T) a multiset of evolution rules with antecedents in U and targets in T.

The input in the region is the initial multiset m.

The output is a maximal multiset m′.

The transformations have been made based on the application of the evolution rules over m until m′ is obtained.

Application of evolution rules in each region of P systems involves subtracting objects from the initial multiset by using rules antecedents. Rules used are chosen in a non-deterministic manner. This phase ends when no rule is applicable anymore.

The transformation only needs rules antecedents as the consequents are part of the communication phase.

Observation

Let $k_i \in N$ be the number of times that the rule r_i is applied. Therefore, the number of symbols a_j which have been consumed after applying the evolution rules a specific number of times will be:

$$\sum_{i=1}^{m} k_i \cdot u_{ij} \tag{3}$$

Definition

Given a region R and alphabet of objects U, and R (U, T) set of evolution rules over U and targets in T.

$$\begin{array}{c} r_1 : a_1^{u_{11}} a_2^{u_{12}} \ldots a_n^{u_{1n}} \rightarrow C_1 \\ r_2 : a_1^{u_{21}} a_2^{u_{22}} \ldots a_n^{u_{2n}} \rightarrow C_2 \\ \ldots \qquad \rightarrow \ldots . \\ r_m : a_1^{u_{m1}} a_2^{u_{m2}} \ldots a_n^{u_{mn}} \rightarrow C_m \end{array} \tag{4}$$

Maximal multiset is that one that complies with:

$$\bigcap_{l=1}^{m}\left[\bigcup_{i=1}^{n}\left(u_i - \sum_{j=1}^{m}(k_j \cdot u_{ji}) \leq u_{li}\right)\right] \tag{5}$$

5 Correction

The correction of the system fully relies in the correction of the internal P-system of the MCA. In order to prove the aggregation system is distributed then 2 processes need to be proven.

Correction of the formal definition of Transition P-System (Paun 1998)
Correction of the aggregation rules applying to 2 given P-systems.

The correction of the second point gets reduced to a deductive demonstration where the aggregation of 2 given P-systems is base case and the generic case of n-P-systems can be seen as the aggregation of n-1 P-systems (inductive case) with a correct aggregation to the last one.

Thus, the key is to prove that aggregation of 2 given P-system is a correct process and indeed reinforce the idea of full inherent parallelism and nondeterministic modelling that membrane models are after.

Aggregation rule. Let us use a short definition of a given P-System

$$\prod = \left(V, \mu, \omega_1, .., \omega_n, (R_{1,}\rho_1), ..(R_{n,}\rho_n), i_0\right)$$

Base case. Given 2 Transition P-system

Aggregation where P_1, P_2 are 2 given P-Systems, P_{12} is the aggregated P-system where is the aggregated alphabet of both P-systems, μ_{12} is the set of regions in the aggregated P-system and ω_{12}, R_{12} are the multiset of objects and set of evolution rules of the aggregated P-system.

Building the aggregated alphabet is obvious. The result is the Union of both. Correctness for this operation is also obvious.

The aggregation of the 2set of multisets is obvious. The result is the Union of both. Correctness for this operation is also obvious.

The aggregation of the 2set of the set of the evolution rules R_{12} is obvious. The result is the Union of both. Correctness for this operation is also obvious.

There are 2 factors in the aggregation that are not obvious which are the aggregated Set of regions μ_{12}. This set of regions is constructed in our proposal as supervised and directed by the factor λ that defines the capabilities previously mentioned. This λ is defined dynamically by the nature of problem the MCA is about to fix. i.e. in a problem of sum of squares is not necessary aggregation as 2 independent P-system could calculate their squares [Paun 2001] and send those outputs to a third (obvious) one that calculates the sum of both results. However, for didactic purposes and aggregated solution could be provided in where a MCA is created with 2 Input P-systems. The aggregated would assign equal λ (priority) to both of them, and then either of them could contain the other one. The container P-system process the output of the contained P-system by adding it to an another square number.

Other problems, especially those that requires sub solutions that are part of optimization techniques would be required to establish a clear hierarchy in the aggregation of MCA. Thus:

The aggregation of the regions of 2 P-systems would be determined by a priority or hierarchy described by λ. This is a dynamic factor that must be configured right before the problem is dealt with.

The aggregated P-system will have to work the communication phase after every evolutionary step. This communication phase also fully relies on the hierarchy establish by λ and will operate as normal when the aggregation is complete and the MCA is finished.

Inductive case:

Given a successful aggregation (MCA) of n P-systems MCA(n), is it correct to aggregate n+1 P-systems?

The inductive case is a direct consequence of the aggregated property.

MCA (n) system becomes a complex P-System with an aggregation of regions according to the λ factor. MCA (n) = let's call the aggregated P-system as. Once the aggregation is seen as a P-system, aggregating it with another P_1 is obvious by applying the base case.

Simulations and results:

We have been performing some simulations in simple problem solving in same traditional computing paradigm For small problems clearly aggregation is not necessary, although the advantage of this proposal shows up, when the complexity of the problem increases. Theoretically a fully and corrected aggregated Solution (A whole MCS) would overweight the cost of the calculation of λ and he redesign of the membrane system that can always occur during compiling time anyways (Table 1).

Table 1. Comparison traditional P-System with MCA (simulations).

Algorithm	Membrane system	MCA
Sum of squares	1.9 µs	2.9 µs
Product of squares	2.3 µs	2.4 µs
Square + random	1.92 µs	2.92 µs
Cubic random	1.93 µs	3.93 µs
Square + random	1.92 µs	2.92 µs
NAND continuous	2.83 µs	2.87 µs
XOR continuous	2.72 µs	2.56 µs
Cubic random AND XOR	3.96 µs	4.01 µs
Square + random AND XOR	3.82 µs	3.52 µs
Cubic random CONTINUOS XOR	4.77 µs	3.99 µs

The analysis is very direct. The simulations are running in the same platform and just focuses in performance time based. All problems are considered simple problems due to the limitations of processing a complex problem with a complex set of aggregation rules which will jeopardize the accuracy of the analysis. Nevertheless, it is indicative to see that there is a variation in the performance when the level of complexity slightly increases which suggest that aggregation can be a good approach when the level of complexity increases.

6 Conclusions

Membrane computing has been growing since George Paun defined it in 1998. Since then new variations have been suggested to try to fit this model to new realities. The main goal for this unconventional paradigm is to improve the performance of the traditional algorithms due to the inherent limitation of the model. Simulations are still a big part of membrane computing and they are useful to extract right conclusions about the new model. In particular, this model is a great candidate to be applied to complex models that require an aggregated solution that is part of other sub solution whole super solutions as long as the defined rules in the MCA are followed. The aggregation factor that is linked to the minimal membrane cells is the component that complement the use membrane computing as a whole and as unite aggregated model. As the creation of this factor generates difficulties because it depends on the nature of the problem, it does not damage the performance during the execution as the factor is calculated in compiling time. New techniques to atomize the generation of λ as this could create a complete dynamic model that fully adjust to the problem and create the right MCA. The necessity of opening the line of research is out of question. The field is growing and new experiments are required. MCA systems are provided as a natural solution to upgrade the nature of membrane computing by not only taking advantage of the properties of the membrane cells but by the way these cells are aggregated. The future work will be involving complex problems in complex aggregated structures, so the analysis can be more relevant. Nevertheless, the evidence points out that aggregation is a natural solution to deal with complex problems that nowadays are being processed by conventional approaches such as backtracking or dynamic programming.

References

1. Adelman, L.M.: Molecular computation of solutions to combinatorial problems. Science **226**, 1021–1024 (1994)
2. Amir-Kroll, H., Sadot, A., Cohen, I.R., Harel, D.: GemCell: a generic platform for modelling multi-cellular. Theor. Comput. Sci. **391**, 276–290 (2008)
3. Arteta, A., Mingo, L.F., Castellanos, J.: An isomorphism based algorithm to solve complex problems. WSEAS Trans. Inf. Sci. Appl. **15**, 27–36 (2018)
4. Arteta, A., Mingo, L.F., Gomez, N.: New approach to optimize membrane systems. J. Bioinform. **1**(1), 1–6 (2014)
5. Arteta, A., Mingo, L.F., Gomez, N.: Membrane systems working with the P-factor: best strategy to solve complex problems. Adv. Sci. Lett. **19**(5), 1490–1495 (2012)

6. Arteta, A.: MEIA systems: membrane encrypted information application systems. Nat. Inf. Technol. Madr. Int. J. Inf. Theor. Appl. **19**(2), 103–109 (2012)
7. Arteta, A., Gomez, N., Gonzalo, R.: Solving diophantine equations with a parallel membrane computing model. Int. J. Inf. Models Anal. **1**, 220–225 (2012)
8. Arteta, A., Mingo, L.F., Gomez, N.: Solving complex problems with a bio-inspired model. Eng. Appl. Artif. Intell. **24**(6), 919–927 (2011)
9. Arteta, A., Fernández, L., Arroyo, F.: P-systems: study of randomness when applying evolution rules. In: International Book Series "Information Science and Computing, pp. 15–24 (2009)
10. Arteta, A., Goñi, A., Castellanos, J.: Analysis of P-systems under multiagents perspective. In: International Book Series "Information Science and Computing", pp. 117–128 (2009)
11. Angeleska, A., et al.: RNA-guided DNA assembly. J. Theor. Biol. **248**, 706–720 (2007)
12. Ardelean, I., et al.: A computational model for cell differentiation. BioSystems **76**(1–3), 169–176 (2004)
13. Balazsi, G., Barabasi, A.-L., Oltvai, Z.N.: Topological units of environmental signal processing in the transcriptional regulatory network of Escherichia coli. PNAS **102**(22), 7841–7846 (2005)
14. Bashor, C.J., Horwitz, A.A., Peisajovich, S.J., Lim, W.A.: Rewiring cells: synthetic biology as a tool to interrogate the organizational principles of living systems. Annu. Rev. Biophys. **39**, 515–537 (2010)
15. Blow, N.: Systems biology: untangling the protein web. Nature **460**, 415–418 (2009)
16. Bray, D.: Wetware: A Computer in Every Living Cell: The Computer in Every Living Cell. Yale University Press, New Haven (2009)
17. Brooks, R.A.: The relationship between matter and life. Nature **409**, 409–411 (2001)
18. Cao, H., Romero-Campero, F.J., Heeb, S., Camara, M., Krasnogor, N.: Evolving cell models for systems and synthetic biology. Syst. Synth. Biol. **4**(1), 55–84 (2010)
19. Carroll, S.B.: Endless Forms Most Beautiful. W.W. Norton, Chicago (2005)
20. Danchin, A.: Bacteria as computers making computers. FEMS Microbiol. Rev. **33**(1), 3–26 (2009)
21. Dasgupta, D. (ed.): Artificial Immune Systems and Their Applications. Springer, Heidelberg (1998). https://doi.org/10.1007/978-3-642-59901-9
22. Dassow, J., Mitrana, V.: On some operations suggested by genome evolution. In: Proceedings of the Second Pacific Symposium on Biocomputing, pp. 97–108 (1997)
23. Dassow, J., Mitrana, V., Salomaa, A.: Context-free evolutionary grammars and the structural language of nucleic acids. BioSystems **43**, 169–177 (1997)
24. Dassow, J., Mitrana, V., Salomaa, A.: Operations and language generating devices suggested by the genome evolution. Theor. Comput. Sci. **270**(1–2), 701–731 (2002)
25. Edelman, G.M.: Topobiology: An introduction to molecular embriology. Basic Books, New York (1988)
26. Ehrenfeucht, A., Harju, T., Petre, I., Prescott, D.M., Rozenberg, G.: Computation in Living Cells: Gene Assembly in Ciliate. Natural Computing Series. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-662-06371-2
27. Engelbrecht, A.: Fundamentals of Computational Swarm Intelligence. Wiley, Chichester (2005)
28. Federici, D., Downing, K.: Evolution and development of a multicellular organism: scalability, resilience, and neutral complexification. Artif. Life **12**(3), 381–409 (2006)
29. Freund, R., Martin-Vide, C., Mitrana, V.: On some operations suggested by gene assembly in ciliates. New Gener. Comput. **20**, 279–293 (2002)
30. de Frutos, J.A., Fernández, L., Luengo, C., Arteta, A.: Improving active rules performance in new P system communication architectures. Inf. Technol. Knowl. **4**(1), 3–18 (2010)

31. Arteta, A., Castellanos, A., Martinez, A.: Membrane computing: non deterministic technique to calculate extinguished multisets of objects. Int. J. Inf. Technol. Knowl. **4**(1), 30–41 (2010)
32. de Frutos, J.A., Arroyo, F., Arteta, A.: Usefulness states in new P system communication architectures. In: Corne, D.W., Frisco, P., Păun, G., Rozenberg, G., Salomaa, A. (eds.) WMC 2008. LNCS, vol. 5391, pp. 169–186. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-540-95885-7_13
33. Han, J., et al.: Evidence for dynamically organized modularity in the yeast protein–protein interaction network. Nature **430**, 88–93 (2004)
34. Haynes, K.A., Silver, P.A.: Eukaryotic systems broaden the scope of synthetic biology. JCB **187**(5), 589–596 (2009)
35. Ho, L., Crabtree, G.R.: Chromatin remodelling during development. Nature **463**, 474–484 (2010)
36. Holcombe, M., Bell, A.: Computational models of immunological pathways. In: Holcombe, M., Paton, R. (eds.) Information Processing in Cells and Tissues. Plenum Press, New York (1998)