


Please cite the Published Version

García-Victoria, Pedro, Cavaliere, Matteo , Gutiérrez-Naranjo, Miguel A and Cárdenas-Montes, Miguel (2022) Evolutionary game theory in a cell: A membrane computing approach. Information Sciences, 589. pp. 580-594. ISSN 0020-0255

DOI: <https://doi.org/10.1016/j.ins.2021.12.109>

Publisher: Elsevier

Version: Accepted Version

Downloaded from: <https://e-space.mmu.ac.uk/628996/>

Usage rights:  [Creative Commons: Attribution-Noncommercial-No Derivative Works 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/)

Additional Information: This is an Author Accepted Manuscript of an article published in Information Sciences by Elsevier.

Enquiries:

If you have questions about this document, contact openresearch@mmu.ac.uk. Please include the URL of the record in e-space. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from <https://www.mmu.ac.uk/library/using-the-library/policies-and-guidelines>)

Evolutionary game theory in a cell: A membrane computing approach

Pedro García-Victoria^a, Matteo Cavaliere^{b,*}, Miguel A. Gutiérrez-Naranjo^a, Miguel Cárdenas-Montes^c

^aDepartment of Computer Sciences and Artificial Intelligence University of Seville, Spain

^bDepartment of Computing and Mathematics, Manchester Metropolitan University, UK

^cDepartment of Basic Research, Centro de Investigaciones Energéticas Medioambientales y Tecnológicas, Madrid, Spain

A B S T R A C T

Evolutionary Game Theory studies the spreading of strategies in populations. An important question of the area concerns the possibility that certain population structures can facilitate the spreading of more cooperative behaviours associated to the sustainability and resilience of many different systems ranging from ecological to socio-economic systems. In this paper, we propose a novel approach to study the spreading of behaviours in structured populations by combining Evolutionary Game Theory and membrane computing. We show that there is a general way to encode Evolutionary Game Theory into membrane computing, leading to a novel computational framework which can be used to study, analyze and simulate the spreading of behaviours in structured populations organized in communicating compartments. The proposed approach allows to extend the works on membrane systems, population and ecological dynamics, and, at the same time, suggests a novel bio-inspired framework, based on formal languages theory, to investigate the dynamics of evolving structured populations.

1. Introduction

Evolutionary Game Theory (EGT, for short) is a mathematical and computational framework which is used to study the spreading of behaviours (strategies) in evolving populations [1,2]. While classical game theory is used to describe the behaviour of completely rational players, in EGT, the individual strategies are not associated to rational choices, but they are assumed to be encoded into inherited programs that can be passed to the offsprings [2].

The main driving principle of EGT is that individuals that perform better (they get a higher payoff/fitness) will tend to replicate more often, so their encoded strategy will spread in the population [2]. The success (fitness) of an individual depends not only on its own adopted strategy but also on the strategies of the other individuals with whom is interacting in the population [2]. This means that there could not be an universal optimal strategy but the optimal choice may depend on the strategies that are adopted by the other components of the population (this makes EGT different from standard optimization and is technical referred as frequency-dependent selection in the area of evolutionary dynamics [2]).

* Corresponding author.

E-mail addresses: pedrogarciavictoria@gmail.com (P. García-Victoria), m.cavaliere@mmu.ac.uk (M. Cavaliere), magutier@us.es (M.A. Gutiérrez-Naranjo), miguel.cardenas@ciemat.es (M. Cárdenas-Montes).

An important problem studied using EGT is the resilience of cooperation and the conflict between cooperative and non-cooperative (cheating/defecting) individuals [3]. This issue is present in many systems, at different scales ranging from technological systems [4] to microbial systems [5] and human societies [6] and it is considered one of the most relevant problems in science [7]. The study of the resilience of cooperation in structured populations has received a strong attention [8,9,11–13], suggesting that the population structure (e.g., the social network) is a crucial part of the problem [10,12]. Therefore the search of appropriate, efficient and general mathematical and computational frameworks to study the interplay of population structure and spreading of certain strategies has become an important research area [8,9,14,15].

The dynamics of an EGT model can be studied analytically [1] but very often is simulated using ad hoc agent-based computational models [15]. In these models, the replication and death of individuals (agents) are explicitly simulated using a system updated by a series of discrete events [15].

In this paper, we propose a novel computational approach to study the spreading of behaviours in structured populations by combining EGT and membrane computing. Using the proposed approach, we show that there is an effective and general way to encode EGT into probabilistic membrane computing and P-Lingua [16,17]. This work therefore enhances the area of membrane computing, extending the line of research focused on the simulation and study of population/ecological dynamics [17,18] allowing the study of the spreading of strategies in populations structured in compartments using probabilistic simulators such as P-Lingua [17]. At the same time, the proposed combination enriches the EGT area with a novel cellular-inspired framework to study, analyze and simulate the spreading of behaviours in structured evolving populations organized in communicating nested compartments. Our proposal will make possible to tackle further systems, for instance those related to the cooperation concerns.

To sum up, the main contributions of this paper are the following:

- A novel approach to study the spreading of behaviours in structured populations by combining Evolutionary Game Theory and membrane computing is proposed.
- We show that there is a general way to encode Evolutionary Game Theory into membrane computing, leading to a computational framework which can be used to study, analyze and simulate the spreading of behaviours in structured populations organized in communicating compartments.
- The proposed approach allows to extend the works on membrane systems, population and ecological dynamics, and, at the same time, suggests a novel bio-inspired framework, based on formal languages theory, to investigate the dynamics of evolving structured populations.

The paper is organized as follows: Section 2 recall some basics of the P system model used along the paper. Among the possible models, our choice has been probabilistic P systems, since probability (in replication, deletion or migration) is one of the pillars of EGT. In Section 3, we present the design of a family of probabilistic P systems which simulates the behaviour of EGT and a short overview of the computation. It deserves to be remarked that all the simulation is performed by P system rules, without considering oracles or external functions. In this way, MeCoSim simulator [19] has been used for the experiments. Section 4 show some examples of EGT cases and how the compartmental structure of P systems can help in the development of the interaction among populations. Finally, the paper finishes with some conclusions and hints for future work.

2. Probabilistic P systems

Since Gh. Păun introduced in 1998 the first model of membrane computing [20] till now, many different variants of P systems have been presented. Among them, spiking neural P systems (SNPS) [21] is one of the most widely extended. From the starting model of SNPS, many other features have been added and explored. Among the most recent, we can cite *Homogeneous spiking neural P systems with structural plasticity* [22], *Delayed Spiking Neural P Systems with Scheduled Rules* [23] or *Spiking neural P systems with autapses*, [24]. Among the recent contributions we can cite the SN P systems with communication on requests [25,26] or SN P systems variant used in optimization and for building an arithmetic calculator [27–29].

Beyond SNPS, many other P system variants have proved their efficiency in order to model real life problems. In particular, probabilistic/stochastic models [16] have shown its efficiency in problems from biological processes.

In this paper, we consider the so-called *probabilistic P systems* [17] introduced by Cardona et al. in 2011. Next, we briefly summarize some of their main features.

Let us consider a working alphabet Γ and a membrane structure where different membranes have different labels. Membranes have electrical charges from the set $\{0, +, -\}$ and \mathbb{R} is a finite set of evolution rules of the form

$$u[v]_i^\alpha \rightarrow w[v]_j^\omega$$

where u, w, v and v' are multisets over Γ , i and j are labels¹ and $\alpha, \omega \in \{0, +, -\}$.

The representation of information as multisets placed on a membrane structure and the use of biologically inspired rules for performing the evolution of such multiset of objects is common to many P system variants. The main feature of the prob-

¹ In the original description of probabilistic P systems $i = j$.

abilistic P systems is that the rules are endowed with a computable function f_r , ($r \in \mathbb{R}$) such that $dom(f_r) \subseteq \{1, \dots, T\}$ (with T a natural number) and $range(f_r) \subseteq [0, 1]$ verifying that if r_1, \dots, r_z are the rules form \mathbb{R} with the same left-hand side (i.e., $u[v]_i^\alpha$) then $\sum_{j=1}^z f_{r_j}(a) = 1$ for $a = 1, \dots, T$. Intuitively, such $f_r(a)$ represents the probabilistic constant associated with rule r . In general, it is written as $r : u[v]_i^\alpha \xrightarrow{f_r(a)} w[v']_j^\alpha$. If $f_r(a) = 1$, then it is denoted by $r : u[v]_i^\alpha \rightarrow w[v']_j^\alpha$.

The key point is that in this P systems applicable rules are non-deterministic chosen as usual, but, in this case the probability of each rule can be fixed at the beginning of the computation.

The semantics of the P system follow the next principles:

- [I1] When an object cross a membrane, its polarization may change. In such way, the electrical charges can act as traffic lights, i.e., rules can only be applied if the polarization of the rule is the appropriate.
- [I2] If a rule can be applied inside a membrane and, in the same step a send-in or send-out rule which changes the polarization of the membrane can also be applied, both rules are applied. In a certain sense, we can consider that the change of the polarization is performed *after* the application of the evolution rules.

3. Evolutionary Game Theory in P systems

In this section, the description of a P system which computes the evolution of a population according to Game Theory is provided. The main contribution of this approach is that different sub-populations can be encapsulated in P system membranes. Individuals inside such membranes evolve according Game Theory principles:

- Each individual gets a *fitness value* obtained by meetings with other individuals according to a chosen strategy.
- One of the individuals is replicated. The probability of replication of an individual is proportional to its fitness value.
- In order to keep constant the number of individuals, after replication one individual is randomly chosen to be deleted.

Each sub-population evolves by these principles in a compartment. After each cycle one individual in each compartment migrates to other compartment. These migration between compartments is possible due to the intrinsic nature of P systems and it represents a big chance for exploring the behaviour of these populations. Next, we describe the technical details of the proposed model.

Let us consider a population of $n \times Q$ individuals distributed on Q compartments. In each compartment, there are n individuals with identifiers $1, \dots, n$. Due to technical reasons, in our approach, n must be even. Each individual can follow two different strategies: *cooperator* or *defector*. When a cooperator meets another cooperator, they both get R (Reward). If a cooperator meets a defector, the cooperator gets S (the Sucker's payoff) and the defector T (the Temptation). If two defectors meet, they both get P (the Punishment). This can be encoded in a payoff matrix.

	Cooperate	Defect
Cooperate	R	S
Defect	T	P

or in a 4-tuple $\langle R, S, T, P \rangle$. In our design, each compartment (which is represented by an elementary membrane) has a different payoff matrix and individuals can move from one into other compartment. In this way, we can study the evolution of the different populations of individuals according to the different payoff matrices and the influence of the communication among compartments. The number of individuals in each compartment keeps constant along the computation, but the proportion of defectors and cooperators in each compartment can change due to the processes of replication, deletion and migration.

For each compartment $k \in \{1, \dots, Q\}$, we have a payoff matrix represented by $\langle R_k, S_k, T_k, P_k \rangle$. In our approach, each individual have exactly M meetings per cycle. In such way, a cooperator in the compartment k will have, after A meetings, an accumulated payoff which can be calculated as the sum of A values taken from R_k and S_k in all the possible ways, i.e., the possible accumulated payoff of a cooperator after A meetings in the membrane k is one of the values of

$$V_{c,k,A} = \{(r \times S_k) + ((N - r) \times R_k) | r \in \{0, \dots, A\}\}$$

Analogously, the possible accumulated payoff of a defector after A meetings in the membrane k is one of the values of

$$V_{d,k,A} = \{(r \times T_k) + ((N - r) \times P_k) | r \in \{0, \dots, A\}\}$$

In EGT, the *fitness* F of an individual depends on: (1) The accumulated payoff Ac ; (2) The number of meetings of the individual M and; (3) A parameter w which we call *influence*. The fitness is computed by the formula

$$F = (1 - w) + w \times \frac{Ac}{M} \tag{1}$$

In this way, each individual will obtain a fitness depending on Ac , M and w and its probability of replication will be proportional to its fitness. In our model, the number of meetings M is fixed and also the influence w and therefore, the fitness F_i of an individual i with accumulated payoff Ac_i is computed as

$$F_i = (1 - w) + w \times \frac{Ac_i}{M} \quad (2)$$

From this equation we have

$$M \times F_i = M \times (1 - w) + w \times Ac_i \quad (3)$$

We can consider that $New_F_i = M \times F_i$ is the fitness F_i scaled by a constant M and then

$$New_F_i = M + w \times (Ac_i - M) \quad (4)$$

Since P systems deal with integer numbers, we can approximate Eq. 4 by considering

$$New_F_i = M + \lfloor w \times (Ac_i - M) \rfloor \quad (5)$$

where $\lfloor x \rfloor$ is greatest integer less than or equal to x . We define the following set $LF_{c,k}$ of pairs (Ac, New_F_i) , where Ac is the accumulated payoff obtained by a cooperator after M meetings in the membrane k and New_F_i the fitness value obtained from Ac according to Eq. 5,

$$LF_{c,k} = \{(Ac, New_F) \mid Ac \in V_{c,k,M}\}$$

We will denote by $MaxF_c$ the maximum value of the fitness that can be reached by a cooperator in any membrane, i.e.,

$$MaxF_c = \max\{B \mid (A, B) \in \cup_{k=1}^{k=Q} LF_{c,k}\}$$

Analogously, $LF_{d,k}$ of pairs (Ac, New_F_i) , where Ac is the accumulated payoff obtained by a defector after M meetings in the membrane k and New_F_i the fitness value obtained from Ac according to Eq. 5,

$$LF_{d,k} = \{(Ac, New_F) \mid Ac \in V_{d,k,M}\}$$

and $MaxF_d$ is the maximum value of the fitness that can be reached by a defector in any membrane, i.e.,

$$MaxF_d = \max\{B \mid (A, B) \in \cup_{k=1}^{k=Q} LF_{d,k}\}$$

In order to define formally a P system which simulates the evolution of the Game Theory, we will consider different type of objects:

- An individual will be represented by a tuple $\langle Id, t, Am, Ap \rangle$ where
 - $Id \in \{1, \dots, n\}$ is the identifier of the individual;
 - $t \in \{c, d\}$ is the strategy of the individual: cooperator (c) or defector (d);
 - Am is the number of meetings in the current configuration;
 - Ap is the accumulated payoff in the current configuration.

At the beginning of the computation, each compartment has n individuals with identifiers $1, \dots, n$ and each individual has a strategy. After each cycle, each compartment has again n individuals with identifiers $1, \dots, n$, but the proportion of defectors and cooperators can change due to the processes of deletion, replication and migration.

- Two special kind of objects are $p_{Id,j}$ and $q_{Id,j}$. Each of these objects represent a payoff unit of the individual with identifier Id : $p_{Id,j}$ if it is a cooperator and $q_{Id,j}$ if it is a defector.
- Other auxiliary objects will be also used: $z_i, f_i, h_i, K, L, L_a, L_b, V, def, coop \dots$

Broadly speaking, the computation is performed according to the following stages:

1. **Meetings stage.** Each individual have M meetings with other individuals randomly chosen in each cycle. It can have more than one meeting with the same individual. Each meeting modifies the accumulated payoffs of the individuals according to their strategies and increases by 1 the number of meetings of the individuals. From a P system point of view, these meetings will be performed along M steps. In order to be sure that all the individuals can be paired and have a meeting in each step, we will consider that the number of individuals n in each compartment is an even number. The end of this stage will be controlled by changing the polarization of the membrane where the meetings occur. This stage is performed in parallel along the Q elementary membranes.
2. **Replication and killing stage.** After M steps, the meetings between individual stop and the reproduction stage begins. In each elementary membranes, only one of the individuals is chosen to replicate and the probability of an individual to be chosen is proportional to its fitness. As pointed out above, the correspondence between the accumulated payoff and the fitness is stored in the sets $LF_{c,k}$ and $LF_{d,k}$. Each of the pairs (A, B) in these sets will produce rules of type

$$\lfloor \langle Id, c, M, A \rangle \rightarrow p_{Id,1} \dots p_{Id,B} \langle Id, c, 0, 0 \rangle \rfloor_k^+$$

$$[\langle Id, d, M, A \rangle \rightarrow q_{Id,1} \dots q_{Id,B} \langle Id, d, 0, 0 \rangle]_k^+$$

Rule $[\langle Id, c, M, A \rangle \rightarrow p_{Id,1} \dots p_{Id,B} \langle Id, c, 0, 0 \rangle]_k^+$ takes a cooperator with identifier Id such that, after M meetings, has obtained an accumulated payoff A and produces B different objects $p_{Id,1} \dots p_{Id,B}$ where B is the fitness associated² to the accumulated payoff A according to the corresponding $LF_{c,k}$. This rule also reset to 0 the accumulated payoff and the number of meetings and produces $\langle Id, c, 0, 0 \rangle$. The case of a defector is analogous.

In parallel with the production of objects $p_{Id,j}$ and $q_{Id,j}$, a set of rules is applied in order to compute the *mutation* process. A unique object L (L for *life*) is produced in each elementary membrane. In the next step of computation, one and only one of the following rules is triggered:

$[L \rightarrow L_a]_k^+$ with probability U (the probability of mutation).

$[L \rightarrow L_b]_k^+$ with probability V (the probability of *no* mutation).

Let us notice that both rules have the same left-hand-side and the sum of the probabilities is $U + V = 1$. If L_a is produced the replication process inside the membrane k will occur with mutation. If L_b is produced, the replication will be without mutation.

Let us suppose that L_b is generated, i.e., the replication of the randomly chosen individual will be performed without mutation. In such case one and only one of the following rules is applied.

$$[p_{Id,j} L_b \rightarrow coop K]_k^+ [q_{Id,j} L_b \rightarrow def K]_k^+$$

Since the amount of copies of $p_{Id,j}$ and $q_{Id,j}$ represents the fitness of the individual Id (cooperator and defector, respectively), the probability of triggering one of these rules is proportional to B , i.e., proportional to the fitness of the individual. Since there is only one object L_b , only one of these rules is applied. Since L_b denotes no mutation, the object $p_{Id,j}$ produces a new object *coop* and $q_{Id,j}$ produces a new object *def*. Such object *coop* and *def* can be considered flags in order to recall that the individual which will be produced at the end of the replication process will be a cooperator or a defector (respectively).

If L_a is generated, the replication will be performed with mutation. It means that L_a together with a copy of $p_{Id_1,j}$ (i.e., an object which represents a payoff unit of a cooperator) will produce a defector or, analogously, L_a together with a copy of $q_{Id_1,j}$ (i.e., an object which represents a payoff unit of a defector) will produce a cooperator. The rules are the following:

$$[p_{Id,j} L_a \rightarrow def K]_k^+ [q_{Id,j} L_a \rightarrow coop K]_k^+$$

In any case, the application of one of these replication rules produce an object K (K for *killer*). In the next step, K chooses randomly one of the individuals and delete it by application of one of the rules

$$[K \langle Id, c, 0, 0 \rangle \rightarrow K_{Id}]_k^+$$

$$[K \langle Id, d, 0, 0 \rangle \rightarrow K_{Id}]_k^+$$

The application of such rule produces an object K_{Id} which can be considered a flag for recalling the identifier of the object which has been deleted. Finally we combine the flag K_{Id} with the flag *coop* or *def* and produce a new individual with the strategy obtained by the probabilistic process of mutation and with the identifier of the deleted individual. The rules are

$$[K_{Id} coop \rightarrow \langle Id, c, 0, 0 \rangle]_k^+$$

$$[K_{Id} def \rightarrow \langle Id, d, 0, 0 \rangle]_k^+$$

In any case, after the replication and killing stage, in each compartment k there are n individuals with identifiers $1, \dots, n$.

3. **Migration stage.** At the end of the replication and killing stage, a new object V (V stands for *voyager*) appears in each elementary membrane k . In parallel, each of these objects V chooses randomly one of the individuals of the compartment and sends it out of the elementary membrane, by the application of one of the rules

$$[V \langle Id, c, 0, 0 \rangle]_k^+ \rightarrow V_{k,Id} \langle Id, c, 0, 0 \rangle []_k^-$$

$$[V \langle Id, d, 0, 0 \rangle]_k^+ \rightarrow V_{k,Id} \langle Id, d, 0, 0 \rangle []_k^-$$

One of these rules is applied in parallel in each of the Q membranes and therefore, after the application, there are Q individuals in the membrane surrounding the Q elementary membranes. Objects $V_{k,Id}$ can be considered as flags for remembering that the individual sent out from membrane k had identifier Id . In the next step of computation, each of these individuals is randomly paired to an object V_{k,Id_2} and one of the following rules is randomly applied

² If $B = 0$, then these rules are simply $[\langle Id, c, M, A \rangle \langle Id, c, 0, 0 \rangle]_k^+$ and $[\langle Id, d, M, A \rangle \langle Id, d, 0, 0 \rangle]_k^+$.

$$\langle Id_1, c, 0, 0 \rangle V_{k,Id_2} \square_k^+ \rightarrow [\langle Id_2, c, 0, 0 \rangle]_k^-$$

$$\langle Id_1, d, 0, 0 \rangle V_{k,Id_2} \square_k^+ \rightarrow [\langle Id_2, d, 0, 0 \rangle]_k^-$$

Object V_{k,Id_2} brings the individual $\langle Id_1, c, 0, 0 \rangle$ inside the membrane k and modifies the identifier in order to keep the unicity of the identifiers in the membrane. Let us notice that sending $\langle Id_1, c, 0, 0 \rangle$ inside the membrane k changes its identifier to Id_2 , but the individual keeps its strategy. With this step, the migration process finishes.

After computing these stages Meeting, Replication/Killing and Migration, the P system is again ready for a new Meeting stage.

4. **Stopping stage.** The cycle of the previous three stages is performed C times. After these C cycles, the computation stops.

These stages are summarized in Algorithm 1.

Algorithm 1: Summary

The following cycle of three stages is performed C times

Stage 1 (Meeting stage):

- It is performed in parallel in the Q elementary membranes;
- It takes M steps of the P system;
- Each meeting modifies the accumulated payoff of the involved individuals.

Stage 2 (Replication and killing stage):

- In each elementary membrane:
 - ☆ One individual is chosen with probability proportional to its fitness;
 - ☆ The individual is replicated (maybe with mutation in its strategy).
 - ☆ An individual is removed (killed).

Stage 3 (Migration stage):

- In each elementary membrane, one individual is randomly chosen and sent into other membrane;
- The algorithm finishes with a **Stopping stage**.
-

Definition of the P system

Let us consider a EGT system with the following parameters³:

For each compartment i :

$$\begin{aligned} R_i &: \text{Reward} & S_i &: \text{Sucker's payoff} \\ T_i &: \text{Temptation} & R_i &: \text{Punishment} \end{aligned}$$

Other parameters:

- n : Number of individuals in each compartment
- M : Number of meetings per cycle
- w : Influence
- U : Prob. of mutation
- V : Prob. of no mutation
- C : Number of cycles
- Q : Number of compartments

Let us also consider the finite sets of tuples $LF_{c,k}$ and $LF_{d,k}$. In these conditions, let us consider the following P system

$$\Pi = \langle \Gamma, H, EC, \mu, w_1, \dots, w_Q, w_g, w_s, \mathbb{R} \rangle$$

where

- The alphabet of objects is

³ Let us remark that the choice of these parameters (number of compartments Q , number of individuals in each compartment n , etc. depends on the concrete instance of the problem to be simulated. Once fixed these parameters, the P system Π is completely defined and its evolution, and hence the simulation of the EGT problem, follows the membrane computing principles.

$$\begin{aligned} \Gamma &= \{ \langle Id, t, Am, Ap \rangle \mid Id \in \{1, \dots, n\}, t \in \{c, d\}, Am \in \{0, \dots, M\}, Ap \in \cup_{i=1}^Q \} \\ &\cup \{ p_{Id,j} \mid Id \in \{1, \dots, n\}, j \in \{1, \dots, MaxF_c\} \} \\ &\cup \{ q_{Id,j} \mid Id \in \{1, \dots, n\}, j \in \{1, \dots, MaxF_d\} \} \\ &\cup \{ z_k \mid k \in \{0, \dots, M+3\} \} \\ &\cup \{ L, L_a, L_b, K, V, coop, def \} \\ &\cup \{ f_{i,k} \mid i \in \{1, \dots, 7\}, k \in \{1, \dots, Q\} \} \\ &\cup \{ V_{k,Id} \mid Id \in \{1, \dots, n\}, k \in \{1, \dots, Q\} \} \\ &\cup \{ h_s \mid s \in \{0, \dots, CM+7C-3\} \} \end{aligned}$$

- $H = \{1, \dots, Q\} \cup \{g, s\}$ is the set of labels;
- $EC = \{0, +, -\}$ is the set of electrical charges;
- The membrane structure has Q elementary membranes with labels $1, \dots, Q$; the skin with label s ; an intermediary membrane with label g ;

$$\mu = [[[[]_1^0 \dots []_Q^0]_g^0]_s^0]$$

- The initial multisets are $w_j = z_0 \langle Id, t_{Id}^j, 0, 0 \rangle$ for $j \in \{1, \dots, Q\}$, $Id \in \{1, \dots, n\}$ and $t_{Id}^j \in \{c, d\}$ is the strategy followed by the Id -th individual in the j -th elementary membrane in the initial configuration. We also have $w_g = \emptyset$ and $w_s = h_0$.

We will also consider the following sets of rules \mathbb{R} (λ represents the empty multiset).

Meeting rules: Let $\langle R_k, S_k, T_k, P_k \rangle$ is the encoding of the payoff matrix inside the compartment $k \in \{1, \dots, Q\}$.

$$RS_1 \equiv [\langle Id_1, c, A, B \rangle \langle Id_2, c, A, C \rangle \rightarrow \langle Id_1, c, A+1, B+R_k \rangle \langle Id_2, c, A+1, C+R_k \rangle]_k^0$$

for $k \in \{1, \dots, Q\}$, $Id_1, Id_2 \in \{1, \dots, n\}$, $A \in \{0, \dots, M-1\}$, $B, C \in V_{c,kA}$

$$RS_2 \equiv [\langle Id_1, c, A, B \rangle \langle Id_2, d, A, C \rangle \rightarrow \langle Id_1, c, A+1, B+S_k \rangle \langle Id_2, c, A+1, C+T_k \rangle]_k^0$$

for $k \in \{1, \dots, Q\}$, $Id_1, Id_2 \in \{1, \dots, n\}$, $A \in \{0, \dots, M-1\}$,

$$B \in V_{c,kA}, C \in V_{d,kA}$$

$$RS_3 \equiv [\langle Id_1, d, A, B \rangle \langle Id_2, d, A, C \rangle \rightarrow \langle Id_1, d, A+1, B+P_k \rangle \langle Id_2, d, A+1, C+P_k \rangle]_k^0$$

for $k \in \{1, \dots, Q\}$, $Id_1, Id_2 \in \{1, \dots, n\}$, $A \in \{0, \dots, M-1\}$, $B, C \in V_{d,kA}$

Replication rules:

$$RS_4 \equiv [\langle Id, c, M, A \rangle \rightarrow p_{Id,1} \dots p_{Id,B} \langle Id, c, 0, 0 \rangle]_k^+ \text{ for } k \in \{1, \dots, Q\}, Id \in \{1, \dots, n\},$$

$A \in V_{c,kM}$ and $(A, B) \in LF_{c,k}$

$$RS_5 \equiv [\langle Id, d, M, A \rangle \rightarrow q_{Id,1} \dots q_{Id,B} \langle Id, d, 0, 0 \rangle]_k^+ \text{ for } k \in \{1, \dots, Q\}, Id \in \{1, \dots, n\},$$

$A \in V_{d,kM}$ and $(A, B) \in LF_{d,k}$

$$RS_6 \equiv [L \xrightarrow{U} L_a]_k^+, \text{ i.e., } [L \rightarrow L_a]_k^+ \text{ with probability } U \text{ (for } k \in \{1, \dots, Q\} \text{)}.$$

$$RS_7 \equiv [L \xrightarrow{V} L_b]_k^+, \text{ i.e., } [L \rightarrow L_b]_k^+ \text{ with probability } V \text{ (for } k \in \{1, \dots, Q\} \text{)}.$$

$$RS_8 \equiv [p_{Id,j} L_a \rightarrow defK]_k^+ \text{ for } k \in \{1, \dots, Q\},$$

$Id \in \{1, \dots, n\}$ and $j \in \{1, \dots, MaxF_c\}$

$$RS_9 \equiv [q_{Id,j} L_a \rightarrow coopK]_k^+ \text{ for } k \in \{1, \dots, Q\},$$

$Id \in \{1, \dots, n\}$ and $j \in \{1, \dots, MaxF_d\}$

$$RS_{10} \equiv [p_{Id,j} L_b \rightarrow coopK]_k^+ \text{ for } k \in \{1, \dots, Q\},$$

$Id \in \{1, \dots, n\}$ and $j \in \{1, \dots, MaxF_c\}$

$$RS_{11} \equiv [q_{Id,j} L_b \rightarrow defK]_k^+ \text{ for } k \in \{1, \dots, Q\},$$

$Id \in \{1, \dots, n\}$ and $j \in \{1, \dots, MaxF_d\}$

$$RS_{12} \equiv [K_{Id} coop \rightarrow \langle Id, c, 0, 0 \rangle]_k^+ \text{ for } k \in \{1, \dots, Q\} \text{ and } Id \in \{1, \dots, n\}.$$

$$RS_{13} \equiv [K_{Id} def \rightarrow \langle Id, d, 0, 0 \rangle]_k^+ \text{ for } k \in \{1, \dots, Q\} \text{ and } Id \in \{1, \dots, n\}.$$

Killing rules:

$$RS_{14} \equiv [K \langle Id, t, 0, 0 \rangle \rightarrow K_{Id}]_k^+ \text{ for } k \in \{1, \dots, Q\}, t \in \{c, d\} \text{ and } Id \in \{1, \dots, n\}$$

Migration rules:

$$RS_{15} \equiv [V \langle Id, t, 0, 0 \rangle]_k^+ \rightarrow V_{k,Id} \langle Id, t, 0, 0 \rangle []_k^-$$

for $k \in \{1, \dots, Q\}$, $Id \in \{1, \dots, n\}$, $t \in \{c, d\}$

$$RS_{16} \equiv \langle Id_1, t, 0, 0 \rangle V_{k,Id_2} []_k^- \rightarrow [\langle Id_2, t, 0, 0 \rangle]_k^-$$

for $k \in \{1, \dots, Q\}$, $Id_1, Id_2 \in \{1, \dots, n\}$, $t \in \{c, d\}$

Control rules:

$$\text{For } k \in \{1, \dots, Q\}.$$

$$RS_{17} \equiv [z_i \rightarrow z_{i+1}]_k^0 \text{ for } i \in \{0, \dots, M-3\} \cup \{M, M+1, M+2\}.$$

$$\begin{aligned}
 RS_{18} &\equiv [z_{M-2} \rightarrow z_{M-1} f_{0,k}]_k^0 \\
 RS_{19} &\equiv [z_{M-1} \rightarrow z_M L_k^0]_k^0 \\
 RS_{20} &\equiv [z_{M+3} \rightarrow V_k^+]_k^+ \\
 RS_{21} &\equiv [f_{0,k}]_k^0 \rightarrow [f_{1,k}]_k^+ \\
 RS_{22} &\equiv [f_{i,k} \rightarrow f_{i+1,k}]_g^0 \text{ for } i \in \{1, \dots, 6\}. \\
 RS_{23} &\equiv [f_{7,k}]_k^- \rightarrow [z_0]_k^0
 \end{aligned}$$

Stopping rules:

$$\begin{aligned}
 RS_{24} &\equiv [h_i \rightarrow h_{i+1}]_s^0 \text{ for } i \in \{0, \dots, CM + 7C - 4\}. \\
 RS_{25} &\equiv [h_{CM+7C-3}]_g^0 \rightarrow [h_{CM+7C-3}]_g^+
 \end{aligned}$$

Cleaning rules:

$$\begin{aligned}
 RS_{26} &\equiv [p_{ldj} \rightarrow \lambda]_k^0 \text{ for } k \in \{1, \dots, Q\}, \\
 Id &\in \{1, \dots, n\} \text{ and } j \in \{1, \dots, \max\{B \mid (A, B) \in LF_{c,k}\}\} \\
 RS_{27} &\equiv [q_{ldj} \rightarrow \lambda]_k^0 \text{ for } k \in \{1, \dots, Q\}, \\
 i &\in \{1, \dots, n\} \text{ and } j \in \{1, \dots, \max\{B \mid (A, B) \in LF_{d,k}\}\} \\
 RS_{28} &\equiv [L_a \rightarrow \lambda]_k^0 \text{ for } k \in \{1, \dots, Q\}. \\
 RS_{29} &\equiv [L_b \rightarrow \lambda]_k^0 \text{ for } k \in \{1, \dots, Q\}.
 \end{aligned}$$

Overview of the Computation

We start with the initial configuration. It consists of Q elementary membranes, the skin (with label s) and the intermediate one (with label g). All of them have polarization 0. Each elementary membrane contains the object z_0 , plus n individuals with identifiers $1, \dots, n$. All the individuals have accumulated meetings and accumulated payoff equals to 0. Each of them has also a strategy c or d (t_i^j denotes the strategy of the individual with identifier i in the elementary membrane j). The skin contains an object h_0 .

$$C_0 = \left[\left[\left[\langle Id_1, t_1^1, 0, 0 \rangle \dots \langle Id_n, t_n^1, 0, 0 \rangle z_0 \right]_1^0 \right. \right. \\ \left. \dots \right. \\ \left. \left[\langle Id_1, t_1^Q, 0, 0 \rangle \dots \langle Id_n, t_n^Q, 0, 0 \rangle z_0 \right]_Q^0 \right]_g^0 \right]_s^0 h_0$$

From this starting configuration only meeting rules can be applied (together with rules $[z_0 \rightarrow z_1]_k^0$ from the set of rules RS_{17} and the rule $[h_0 \rightarrow h_1]_s^0$ from the set of rules RS_{24}). Each individual has a meeting with another individual and their accumulated payoff change according to their strategies. In order to fix ideas let us consider a concrete example in the elementary membrane 1 with the payoff matrix settled to $R_1 = 5, S_1 = 1, T_1 = 10, P_1 = 3$. Let us also consider that there are six individuals, four cooperators with identifiers $1, \dots, 4$ and two defectors with identifiers 5 and 6. In this case, the initial configuration in the elementary membrane 1 is:

$$\left[\langle 1, c, 0, 0 \rangle \langle 2, c, 0, 0 \rangle \langle 3, c, 0, 0 \rangle \right. \\ \left. \langle 4, c, 0, 0 \rangle \langle 5, d, 0, 0 \rangle \langle 6, d, 0, 0 \rangle z_0 \right]_1^0$$

Let us suppose that the non-deterministically chosen rules has been

$$\begin{aligned}
 &[\langle 1, c, 0, 0 \rangle \langle 3, c, 0, 0 \rangle \rightarrow \langle 1, c, 1, 5 \rangle \langle 3, c, 1, 5 \rangle]_k^0 \\
 &[\langle 2, c, 0, 0 \rangle \langle 5, d, 0, 0 \rangle \rightarrow \langle 2, c, 1, 1 \rangle \langle 5, d, 1, 10 \rangle]_k^0 \\
 &[\langle 4, c, 0, 0 \rangle \langle 6, d, 0, 0 \rangle \rightarrow \langle 4, c, 1, 1 \rangle \langle 6, d, 1, 10 \rangle]_k^0
 \end{aligned}$$

From a Game Theory point of view, cooperators 1 and 3 meet; cooperator 2 and defector 5 meet; and cooperator 4 with defector 6. In this way, the configuration at time 1 in this elementary membrane is

$$\left[\langle 1, c, 1, 5 \rangle \langle 2, c, 1, 1 \rangle \langle 3, c, 1, 5 \rangle \right. \\ \left. \langle 4, c, 1, 1 \rangle \langle 5, d, 1, 10 \rangle \langle 6, d, 1, 10 \rangle z_1 \right]_1^0$$

These meetings occur in parallel in the Q elementary membranes. Since the polarization of these elementary membranes has not changed, the next step is similar to this one: only rules from the meeting set can be applied together with the corresponding rules from RS_{17} and RS_{24} . The P system goes on with the evolution till reaching the configuration $M - 2$. The configuration at time $M - 2$ is (in the general case) is

$$C_{M-2} = \left[\left[\left[\langle Id_1, t_1^1, M-2, Y_{M-2}^{1,1} \rangle \dots \langle Id_n, t_n^1, M-2, Y_{M-2}^{n,1} \rangle z_{M-2} \right]_1^0 \right. \right. \\ \left. \dots \right. \\ \left. \left[\langle Id_1, t_1^Q, M-2, Y_{M-2}^{1,Q} \rangle \dots \langle Id_n, t_n^Q, M-2, Y_{M-2}^{n,Q} \rangle z_{M-2} \right]_Q^0 \right]_g^0 \right]_s^0 h_{M-2}$$

In the next step of computation, the meeting rules are still applied. We also applied the rules RS_{18} and a rule of the set RS_{24} .

$$\mathbb{C}_{M-1} = \left[\left[\begin{array}{c} \left[\dots \langle Id_j, t_j^1, M-1, Y_{M-1}^{j,1} \rangle \dots Z_{M-1} f_{0,1} \right]_1^0 \\ \dots \\ \left[\dots \langle Id_j, t_j^Q, M-1, Y_{M-1}^{j,Q} \rangle \dots Z_{M-1} f_{0,Q} \right]_Q^0 \end{array} \right]_g^0 \right]_s^0 h_{M-1}$$

In each elementary membrane k in the configuration \mathbb{C}_{M-1} a new object $f_{0,k}$ appears. From this configuration, meeting rules can be applied together with RS_{19} , rules from RS_{21} and the corresponding rule from the set RS_{24} . The application of rules from R_{21} changes the polarization of the elementary membranes, but according to the principle **I2**, we consider that the evolution inside this membrane can be performed in the same computation step. In this way, the configuration a time M is

$$\mathbb{C}_M = \left[\left[\begin{array}{c} \left[\dots \langle Id_j, t_j^1, M, Y_M^{j,1} \rangle \dots Z_M L \right]_1^+ f_{1,1} \\ \dots \\ \left[\dots \langle Id_j, t_j^Q, M, Y_M^{j,Q} \rangle \dots Z_M L \right]_Q^+ f_{1,Q} \end{array} \right]_g^0 \right]_s^0 h_M$$

Since the polarization of the elementary membranes has changed to $+$, now the meeting rules cannot be applied and the replication and killing stage starts. Each individual $\langle Id, c, M, A \rangle$ in the membrane k evolves to an individual $\langle Id, c, 0, 0 \rangle$ together with B objects $p_{Id,1}, \dots, p_{Id,B}$ where (A, B) is a pair in $LF_{c,k}$. This evolution is performed by the application of the corresponding rule from RS_4 . As pointed out above, the fitness B corresponding to the accumulated payoff A is expressed by the object $p_{Id,1} \dots p_{Id,B}$. The application of the rule also resets the accumulated meetings and payoff of the individual to 0. The case of the defectors and the application of the rules from the set RS_5 is analogous.

In each elementary membrane, a new objects L has appeared. L starts the replication process. In each elementary membrane, one and only one of the rules RS_6 or RS_7 is triggered according to the probability of mutations. If L_a is produced, then there is mutation in the replication process; otherwise, if L_b is produced, there is no mutation. Rules RS_{17} and the corresponding from the sets RS_{22} and RS_{24} are also applied.

$$\mathbb{C}_{M+1} = \left[\left[\begin{array}{c} \left[\dots \langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots Z_{M+1} L_{a/b} \right]_1^+ f_{2,1} \\ \dots \\ \left[\dots \langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots Z_{M+1} L_{a/b} \right]_1^+ f_{2,Q} \end{array} \right]_g^0 \right]_s^0 h_{M+1}$$

In the next step the replication is starts. Let us consider that L_b was produced in the previous step. The object L_b is interpreted as the *no-mutation* object. This object is involved in the application of rules from RS_{10} and RS_{11} . Since p_{Id_j} is a fitness unit of a cooperator and L_b denotes that there is no mutation, the application of the rule $RS_{10} \equiv [p_{Id_j} L_b \rightarrow coopK]_k^+$ will produce a flag *coop* which will be used to recall the strategy of the offspring. Rules for 8 to RS_{11} consider all the possibilities of producing a flag *coop* or *def* depending on the strategy of the individual chosen to be replicated (denoted by p_{Id_j} or q_{Id_j} and the probability of mutation. Let us notice that, in any case, a new object K is produced. Rules from RS_{17} , RS_{22} and RS_{24} are also triggered.

$$\mathbb{C}_{M+2} = \left[\left[\begin{array}{c} \left[\dots \langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots Z_{M+2} coop/def K \right]_1^+ f_{3,1} \\ \dots \\ \left[\dots \langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots Z_{M+2} coop/def K \right]_1^+ f_{3,Q} \end{array} \right]_g^0 \right]_s^0 h_{M+2}$$

In any case, regardless which of the rules from RS_8 to RS_{11} is triggered, an object K will appear in the elementary membrane. As we will see below, such object K will be used to delete one of the individuals of the membrane randomly chosen. Let us notice the special case where all the individuals in the membrane are defectors all of them and the *punishment* P_k of the payoff matrix associated to the membrane is $P_k = 0$. In such case, all the possible meetings couple two defectors and their accumulated payoffs are constant (increased by $P_k = 0$). In such case, objects $q_{i,j}$ are never produced by RS_5 and therefore, RS_9 nor RS_{11} are applied. This means that in such membrane there is no replication. As a secondary effect, since RS_9 and RS_{11} are not applied, then the object K is not produced, a none of the individuals of the membrane is deleted. There is no replication a no deletion in this elementary membrane, but it is still possible the migration and then, the possibility of introducing cooperators in the membrane.

In the next step, one of the killing rules from RS_{14} is applied in each elementary membrane. The object K randomly chooses an individual and removes it. The result of the application of the rule is a new object K_{Id} which recalls the identifier of the deleted individual. Rules from RS_{17} , RS_{22} and RS_{24} are also triggered.

$$\mathbb{C}_{M+3} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots z_{M+3} \text{coop/def } K_{Id_1} \right]_1^+ f_{4,1} \right]_g^0 \\ \dots \\ \left[\left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots z_{M+3} \text{coop/def } K_{Id_Q} \right]_1^+ f_{4,Q} \right]_g^0 \\ h_{M+3} \end{array} \right]_s^0$$

In the next step the process of replication and killing finished with the application of one rule from RS_{12} or RS_{13} . The flag which recalls the strategy of the offspring together with the flag which recalls the identifier of the deleted individual and combined in order to produce a new individual with the obtained strategy and the same identifier as the deleted individual. In this case, after the replication and killing stage, the set of identifiers $1, \dots, n$ is kept in each elementary membrane. Rules from RS_{20} , RS_{21} and RS_{24} are also applied. Rules from RS_{20} produce a new object V inside each elementary membrane and the migration stage starts.

$$\mathbb{C}_{M+4} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots V \right]_1^+ f_{5,1} \right]_g^0 \\ \dots \\ \left[\left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots V \right]_1^+ f_{5,Q} \right]_g^0 \\ h_{M+4} \end{array} \right]_s^0$$

In the next step, rules form RS_{15} are applied. The object V chooses randomly an individual and sends it out of the elementary membrane. This happens in parallel in all the Q elementary membranes, so in the next configuration there are Q individuals in the membrane g which surrounds the elementary membranes. Each rule from RS_{15} also produce a flag $V_{k,Id}$ which recalls the identifier Id of the individual which has been sent out from the elementary membrane k . Rules from RS_{22} and RS_{24} are also applied.

$$\mathbb{C}_{M+5} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \right]_1^- \langle Id_j, c, 0, 0 \rangle \right]_g^0 \\ \dots \\ \left[\left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \right]_1^- \langle Id_j, c, 0, 0 \rangle \right]_g^0 \\ f_{6,1} \dots f_{6,Q} V_{1,Id_1} \dots V_{Q,Id_Q} \end{array} \right]_s^0$$

Next, rules from RS_{16} are applied. Each object $V_{k,Id}$ chooses randomly an individual from the membrane g and sends it into the membrane k . Since there are Q objects different $V_{k,Id}$ and Q individuals, only one individual is sent into each elementary membrane k . The application of the rule also changes the identifier of the chosen individual and takes the identifier associated to $V_{k,Id}$. In this way, the individual which arrives to membrane k will take the same identifier as the individual which was sent out in the previous step. In this way, all the individuals inside each membrane will have the identifiers $1, \dots, n$ again. Rules from RS_{22} and RS_{24} are also applied.

$$\mathbb{C}_{M+6} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots \right]_1^- f_{7,1} \right]_g^0 \\ \dots \\ \left[\left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots \right]_1^- f_{7,Q} \right]_g^0 \\ h_{M+6} \end{array} \right]_s^0$$

In the next step, rules from RS_{23} and the corresponding from RS_{24} are also applied. Rules from RS_{23} send an object z_0 inside each elementary membrane and also changes the polarization, so the meeting stage can start again.

Let us notice that the configuration \mathbb{C}_{M+7} is similar to \mathbb{C}_0 . The elementary membranes have n individuals with identifiers $1, \dots, n$ and accumulated payoffs and meetings equal to 0. All the elementary membranes also have an object z_0 and its electrical charge is 0. With this configuration the first cycle has concluded. The differences between \mathbb{C}_{M+7} and \mathbb{C}_0 are:

- The number of the individuals in each elementary membrane is n , but it may be changed the proportion of cooperators and defectors (due to the deletions, replications and migrations).
- The counter h has reached h_{M+7} .
- In \mathbb{C}_{M+7} several objects p_{Id_j} and q_{Id_j} appear. These objects can be considered garbage and will be deleted in the next step by the set of rules RS_{26} and RS_{27} . As pointed above, in the case of that L_a or L_b has been produced but it has not been consumed, such object can also be deleted by rules RS_{28} and RS_{29} .

After other $M + 7$ steps, the configuration $\mathbb{C}_{2 \times (M+7)}$ is reached and this is the end of the second cycle. Analogously, the $(C - 1)$ -th cycle is finished after $(C - 1) \times (M + 7)$ steps. Let us consider the configuration at the step $M + 4 + ((C - 1) \times (M + 7)) = CM + 7C - 3$ which is analogous to the configuration at step $M + 4$

$$\mathbb{C}_{CM+7C-3} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots V \right]_1^+ f_{5,1} \right. \\ \dots \\ \left. \left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots V \right]_1^+ f_{5,Q} \right]_g \\ h_{CM+7C-3} \end{array} \right]_s^0$$

The transition from this configuration to the next one is similar to the transition from \mathbb{C}_{M+4} to \mathbb{C}_{M+5} . The difference is that no more rules from the set RS_{24} are applied and now rule RS_{25} is triggered. The application of this rule changes the polarization of the membrane with label g from 0 to +.

$$\mathbb{C}_{CM+7C-2} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \right]_1^- \langle Id_j, c, 0, 0 \rangle \right. \\ \dots \\ \left. \left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \right]_1^- \langle Id_j, c, 0, 0 \rangle \right]_g \\ f_{6,1} \dots f_{6,Q} V_{1,Id_1} \dots V_{Q,Id_Q} \end{array} \right]_s^0$$

Since the polarization of the membrane g is now positive, rules from RS_{22} are no longer applied and objects $f_{4,k}$ do not evolve any more. Rules from RS_{16} are applied and the migration of the C -th cycle finishes.

$$\mathbb{C}_{CM+7C-1} = \left[\begin{array}{c} \left[\left[\langle Id_j, c, 0, 0 \rangle q_{Id_j,1} \dots p_{1,B_j^1} \dots \right]_1^- f_{6,1} \right. \\ \dots \\ \left. \left[\langle Id_j, d, 0, 0 \rangle q_{Id_j,1} \dots q_{1,B_j^1} \dots \right]_1^- f_{6,Q} \right]_g \\ h_{CM+7C-3} \end{array} \right]_s^0$$

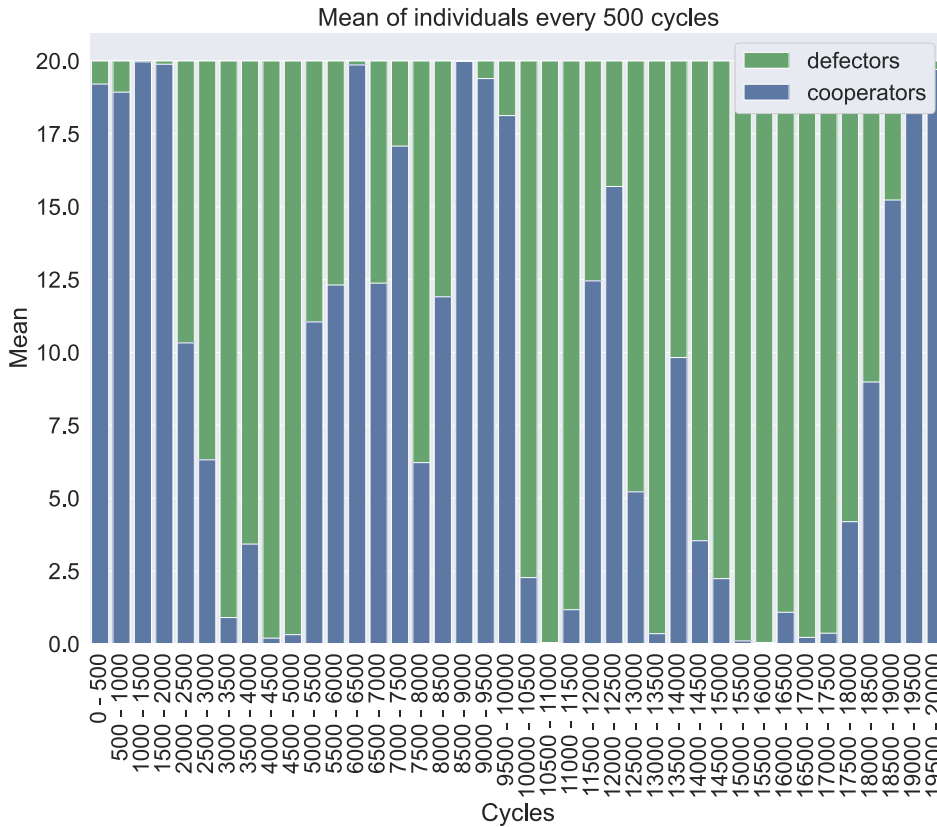


Fig. 1. Evolution of cooperators and defectors in Snowdrift game during 20000 cycles (mean every 500 cycles). We can observe that cooperators and defectors can co-exist, as we can also observe in the long-term average of cooperators (skipping first 1000 cycles, in order to avoid adding in the calculated long term mean the transient effects of the initial configuration) which is 8.73. Cooperators can invade a region full of defectors, while defectors can invade a region full of cooperators. Payoff matrix is composed of Reward = 5, Sucker's payoff = 4, Temptation = 6 and Punishment = 3. Initial individuals are 10 cooperators and 10 defectors. Mutation probability is 0.01 and parameter influence w is 0.5.

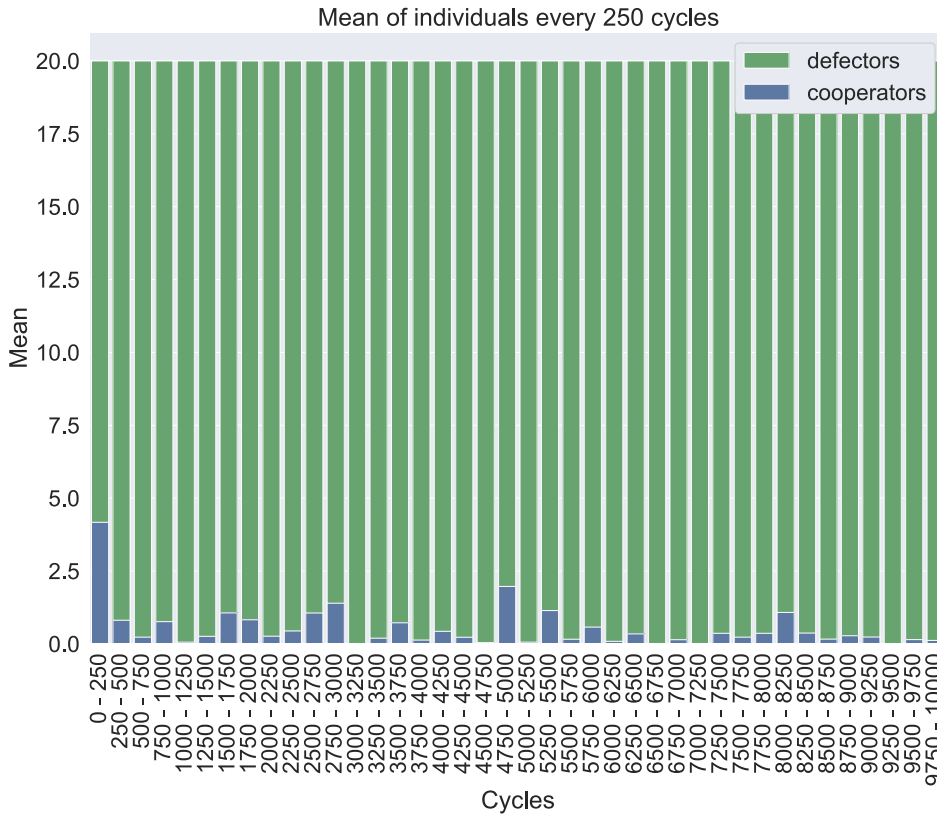


Fig. 2. Evolution of cooperators and defectors in Prisoner's dilemma game during 10000 cycles (mean every 250 cycles). We can observe that the population is composed by mostly defectors as we can also observe in the long-term average of cooperators (skipping first 1000 cycles) which is just 0.42. For cooperators it is difficult to invade a region full of defectors. Payoff matrix is composed of Reward = 3, Sucker's payoff = 1, Temptation = 7 and Punishment = 2. Initial individuals are 10 cooperators and 10 defectors. Mutation probability is 0.01 and parameter influence w is 0.5.

Since the polarization of the elementary membranes is now negative, the meeting and rules cannot be applied. No more rules in general can be applied and $C_{CM+7C-1}$ is a halting configuration. In each of the Q membranes there are n individuals which can be considered the output of the computation.

4. Running Evolutionary Game Theory in MeCoSim

We present the proposed encoding of the model by using the probabilistic P systems simulator MeCoSim [18] and testing the model for the classical games involving cooperation, Prisoner's dilemma and Snowdrift game⁴. The aim of this section is not to do a systematic analysis of these two games but to demonstrate a way to use a standard P systems simulator⁵ to run relevant examples of EGT, including the possibility of using compartments to simulate structured populations and migration, that can be used to enrich the study of cooperation and population dynamics [10,31–34], and, at the same time, enhance the study of P systems and ecological dynamics [18].

Simulations have been conducted using a computer with CPU Intel Core i7 and 16 Gb of RAM.

In Figs. 1 and 2 we compute the long-term average of the number of cooperators by calculating the average number of cooperators considering the entire simulation (skipping the first 1000 cycles). Intuitively, this provides the expected number of cooperators present in the population in the long term. As one would expect, the simulator correctly shows that cooperators and cheaters can co-exists only in the Snowdrift game (Fig. 1), while in the Prisoner's dilemma, the population is dominated by the defectors (Fig. 2).

Fig. 3 shows the evolution of a population of 40 individuals split into 2 membranes with 20 individuals each. In the first one (membrane 1) the individuals evolve with the Prisoner's dilemma matrix and the individuals in membrane 2 evolves according to the Snowdrift game. The system is evolved for 20000 cycles (mean every 500 cycles). In this case, the two previous games are encoded in different elemental membranes. Membrane 1 encodes the Snowdrift game whereas membrane 2 encodes Prisoner's dilemma. In this configuration, migration rules are applied in order to interchange individuals between

⁴ A detailed description of Prisoner's dilemma and Snowdrift game, among many other classical games involving cooperation, can be found in [30].

⁵ The code is available from the authors on request.

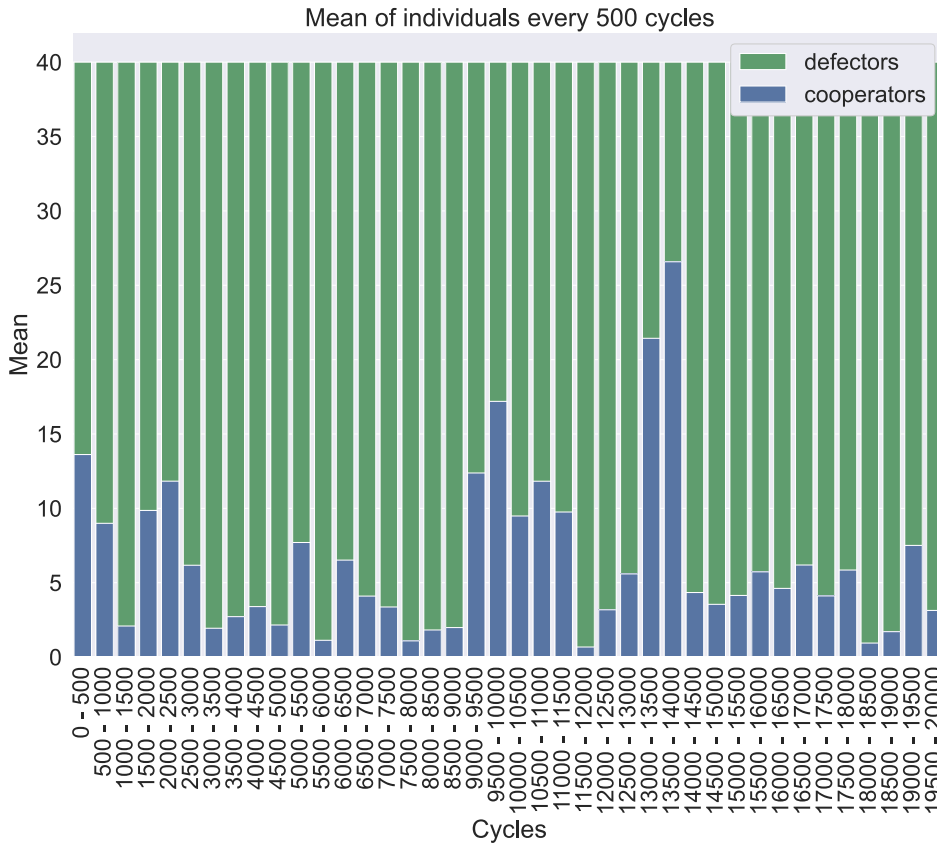


Fig. 3. This figure shows the evolution of a population of 40 individuals split into 2 membranes with 20 individuals each. In the first one (membrane 1) the individuals evolve with the Prisoner’s dilemma matrix and the individuals in membrane 2 evolves according to the Snowdrift game. The system has evolved during 20000 cycles (mean every 500 cycles). Long-term average of cooperators for the overall system (skipping first 1000 cycles) is 6.26. In this configuration, each membrane encode one of the games. Membrane 1 has a payoff matrix composed of Reward = 5, Sucker’s payoff = 4, Temptation = 6 and Punishment = 3 whereas membrane 2 has Reward = 3, Sucker’s payoff = 1, Temptation = 7 and Punishment = 2 (i.e., these payoffs correspond to the Prisoner’s dilemma and Snowdrift games). Both membranes has the same configuration for the rest of parameters: mutation probability is 0.01 and parameter influence w is 0.5.

membranes. In Fig. 3 we can see that the long-term average number of cooperators is distinct from the one obtained in the two scenarios (Prisoner’s dilemma and Snowdrift) when studied independently (Fig. 2 and 3) which highlights the relevance of combining compartments. More generally, the presented approach allows to study systems with complex structures where in different regions can be applied different games, with distinct payoff matrices, and individuals migrate across the compartments. In this way, the cellular-inspired structure of membrane computing can be used to study the spreading of behaviours in structured populations.

5. Conclusions

EGT is a mathematical and computational framework which is used to study the spreading of behaviours (often referred as strategies) in evolving populations. An important problem approached using EGT is the study of the resilience of cooperation in structured populations, i.e., organized according to specific structures. Very often, the problem is studied using ad hoc computational models.

In this paper, we propose a general, flexible, way to encode EGT in P systems. This allows us to provide a formal way to computationally study the dynamics of evolving populations, and, at the same time use well-known simulators and languages for P systems (such as MeCoSim and P-Lingua) to investigate, in silico, the spreading of strategies in structured populations, organized in compartments where individuals can replicate and migrate. To demonstrate the feasibility of the proposed approach we have encoded two well-known games to study cooperation (Prisoner’s dilemma and Snowdrift game) into P systems and simulated their dynamics using MeCoSim. As expected, cooperators and cheaters can co-exists in the Snowdrift game, while in the Prisoner’s dilemma, the population is mostly composed by defectors. The proposed approach allows also to simulate populations organized in compartments with migration across the different membranes, where different games can be associated to different membranes. Despite it is not the aim of this paper to focus on any specific sce-

nario, we believe that the proposed encoding of EGT into P systems will constitute an innovative framework to study and simulate the dynamics of ecological processes related to the spreading of new behaviours in structured populations, a key general research question in EGT. More generally, the proposed approach can provide an alternative general framework to study the evolutionary dynamics in structured populations, enriching the research area that studies ecological dynamics using P systems, and the area that studies the spreading of behaviours in structured populations.

Moreover, our general way of encoding EGT into P systems opens also a completely new possibility - the study of the spreading of behaviours in terms of classical computational processes allowing to ask questions concerning the computability and the computational complexity of EGT in terms of formal languages and automata theory.

EGT has been applied to Social Dilemmas [35] and more recently to the behavioural dynamics of economic shutdowns and shield immunity in the COVID-19 pandemic [36]. This paper opens a door for linking membrane computing with these fields.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the support by David Orellana-Martín with the use of MeCoSim and P-Lingua. MCM is funded by the Spanish Ministry of Economy and Competitiveness (MINECO) for funding support through the grant FPA2017-82647-P, and “Unidad de Excelencia María de Maeztu”: CIEMAT - FÍSICA DE PARTÍCULAS through the grant MDM-2015-0509. MAGN is partially supported by the Ministry of Science and Innovation through the project PID2019-107339GB-I00 “Advances in Computational Topology and Applications”.

References

- [1] J. Hofbauer, K. Sigmund, *Evolutionary Games and Population Dynamics*, Cambridge University Press, 1998, <https://doi.org/10.1017/CBO9781139173179>.
- [2] M.A. Nowak, K. Sigmund, Evolutionary dynamics of biological games, *science* 303 (2004) 793–799.
- [3] M.A. Nowak, Five rules for the evolution of cooperation, *science* 314 (2006) 1560–1563.
- [4] K. Lai, M. Feldman, I. Stoica, J. Chuang, Incentives for cooperation in peer-to-peer networks, in: *Workshop on economics of peer-to-peer systems*, 2003, pp. 1243–1248.
- [5] M. Cavaliere, S. Feng, O.S. Soyer, J.I. Jiménez, Cooperation in microbial communities and their biotechnological applications, *Environmental microbiology* 19 (2017) 2949–2963.
- [6] S. Levin, Crossing scales, crossing disciplines: collective motion and collective action in the global commons, *Philosophical Transactions of the Royal Society B: Biological Sciences* 365 (2010) 13–18.
- [7] E. Pennisi, How did cooperative behavior evolve?, *Science* 309 (2005), 93–93
- [8] C.P. Roca, J.A. Cuesta, A. Sánchez, Evolutionary game theory: Temporal and spatial effects beyond replicator dynamics, *Physics of life reviews* 6 (2009) 208–249.
- [9] M. Perc, A. Szolnoki, Coevolutionary games—a mini review, *BioSystems* 99 (2010) 109–125.
- [10] E. Lieberman, C. Hauert, M.A. Nowak, Evolutionary dynamics on graphs, *Nature* 433 (2005) 312–316.
- [11] C.E. Tarnita, T. Antal, H. Ohtsuki, M.A. Nowak, Evolutionary dynamics in set structured populations, *Proceedings of the National Academy of Sciences* 106 (2009) 8601–8604.
- [12] M. Cavaliere, S. Sedwards, C.E. Tarnita, M.A. Nowak, A. Csikász-Nagy, Prosperity is associated with instability in dynamical networks, *Journal of theoretical biology* 299 (2012) 126–138.
- [13] M.A. Nowak, C.E. Tarnita, T. Antal, Evolutionary dynamics in structured populations, *Philosophical Transactions of the Royal Society B: Biological Sciences* 365 (2010) 19–30.
- [14] H. Ohtsuki, M.A. Nowak, J.M. Pacheco, Breaking the symmetry between interaction and replacement in evolutionary dynamics on graphs, *Physical review letters* 98 (2007) 108106.
- [15] C. Adami, J. Schossau, A. Hintze, Evolutionary game theory using agent-based methods, *Physics of life reviews* 19 (2016) 1–26.
- [16] P. Cazzaniga, M. Gheorghie, N. Krasnogor, G. Mauri, P. Pesciani, F.J. Romero-Campero, Probabilistic/stochastic models, in: Gh. Păun, G. Rozenberg, A. Salomaa (Eds.), *The Oxford Handbook of Membrane Computing*, Oxford University Press, Oxford, England, 2010, pp. 455–474.
- [17] M. Cardona, M.Á. Colomer, A. Margalida, A. Palau, I. Pérez-Hurtado, M.J. Pérez-Jiménez, D. Sanuy, A computational modeling for real ecosystems based on P systems, *Natural Computing* 10 (2011) 39–53.
- [18] M.Á. Colomer, A. Margalida, M.J. Pérez-Jiménez, Population dynamics p system (pdp) models: a standardized protocol for describing and applying novel bio-inspired computing tools, *PLoS one* 8 (2013) e60698.
- [19] I. Pérez-Hurtado, L. Valencia-Cabrera, M.J. Pérez-Jiménez, M.A. Colomer, A. Riscos-Núñez, Mecosim: A general purpose software tool for simulating biological phenomena by means of P systems, in: *Fifth International Conference on Bio-Inspired Computing: Theories and Applications, BIC-TA 2010, University of Hunan, Liverpool Hope University, Liverpool, United Kingdom/ Changsha, China, September 8–10 and September 23–26, 2010, IEEE*, 2010, pp. 637–643. <url:https://doi.org/10.1109/BICTA.2010.5645199>. doi:10.1109/BICTA.2010.5645199.
- [20] Gh. Păun, *Computing with Membranes*, Technical Report 208, Turku Centre for Computer Science, Turku, Finland, 1998.
- [21] M. Ionescu, Gh. Păun, T. Yokomori, Spiking neural P systems, *Fundamenta Informaticae* 71 (2006) 279–308.
- [22] R.T.A. de la Cruz, F.G.C. Cabarle, I.C.H. Macababayao, H.N. Adorna, X. Zeng, Homogeneous spiking neural P systems with structural plasticity, *J. Membr. Comput.* 3 (2021) 10–21.
- [23] Q. Ren, X. Liu, Delayed spiking neural p systems with scheduled rules, *Complexity* 2021 (2021) 1–13.
- [24] X. Song, L. Valencia-Cabrera, H. Peng, J. Wang, Spiking neural p systems with autapses, *Information Sciences* 570 (2021) 383–402.
- [25] T. Wu, F. Bilbie, A. Poun, L. Pan, F. Neri, Simplified and yet Turing universal spiking neural P systems with communication on request, *International Journal of Neural Systems* 28 1850013 (1–1850013) (2018) 19.
- [26] L. Pan, G. Paun, G. Zhang, F. Neri, Spiking neural P systems with communication on request, *International Journal of Neural Systems* 27 (2017) 1750042:1–1750042:13.

- [27] G. Zhang, H. Rong, F. Neri, M.J. Pérez-Jiménez, An optimization spiking neural P system for approximately solving combinatorial optimization problems, *International Journal of Neural Systems* 24 (2014).
- [28] M. Zhu, Q. Yang, J. Dong, G. Zhang, X. Gou, H. Rong, P. Paul, F. Neri, An adaptive optimization spiking neural P system for binary problems, *International Journal of Neural Systems* 31 (2021) 2050054:1–2050054:17..
- [29] G. Zhang, H. Rong, P. Paul, Y. He, F. Neri, M.J. Pérez-Jiménez, A complete arithmetic calculator constructed from spiking neural P systems and its application to information fusion, *International Journal of Neural Systems* 31 (2021) 2050055:1–2050055:17..
- [30] M.A. Nowak, *Evolutionary dynamics: exploring the equations of life*, Harvard University Press, 2006.
- [31] R. Cong, B. Wu, Y. Qiu, L. Wang, Evolution of cooperation driven by reputation-based migration, *PLoS One* 7 (2012) e35776.
- [32] L.-L. Jiang, W.-X. Wang, Y.-C. Lai, B.-H. Wang, Role of adaptive migration in promoting cooperation in spatial games, *Physical Review E* 81 (2010) 036108.
- [33] G. Ichinose, T. Arita, The role of migration and founder effect for the evolution of cooperation in a multilevel selection context, *Ecological Modelling* 210 (2008) 221–230.
- [34] D. Helbing, W. Yu, Migration as a mechanism to promote cooperation, *Advances in Complex Systems* 11 (2008) 641–652.
- [35] S. Iyer, J. Reyes, T. Killingback, An application of evolutionary game theory to social dilemmas: The traveler's dilemma and the minimum effort coordination game, *PLOS ONE* 9 (2014) 1–11.
- [36] K.M.A. Kabir, J. Tanimoto, Evolutionary game theory modelling to represent the behavioural dynamics of economic shutdowns and shield immunity in the COVID-19 pandemic, *R Soc Open Sci* 7 (2020) 201095.