An Infection-Based Mechanism for Self-Adaptation in Multi-Agent Complex Networks

Norman Salazar Juan A. Rodriguez-Aguilar Josep Ll. Arcos
IIIA, Artificial Intelligence Research Institute
CSIC, Spanish National Research Council
Campus UAB, Bellatera, Spain
{norman, jar, arcos}@iiia.csic.es

Abstract

Distributed mechanisms that regulate the behavior of autonomous agents in open multi-agent systems (MAS) are of high interest since we cannot employ centralized approaches relying on global knowledge. In actual-world societies, the balance between personal and social interests is self-regulated through social conventions that emerge in a decentralized manner. As such, a computational mechanism that allows to engineer the emergence of social conventions in MAS can become a highly promising tool to endow open MAS with self-regulating capabilities. To this end we propose a computational self-adapting *mechanism that facilitates agents to distributively* evolve *their social behavior to reach the best social conventions. Our approach borrows from the social contagion phenomenon: social conventions are akin to infectious diseases that spread themselves through members of the society. Furthermore, we experimentally show that our mechanism helps a MAS to regulate itself by searching and establishing (better) social conventions on a wide range of interaction topologies and dynamic environments.*

1 Introduction

Distributed mechanisms that regulate the behavior of autonomous agents in multi-agent systems (MAS) have become necessary because centralized approaches relying on global knowledge are not viable in open MAS. Furthermore, it is difficult for centralized approaches to cope with dynamic environments. We observe that in actual-world societies, social behavior is self-regulated through social conventions. These conventions emerge in a decentralized manner to balance personal interests with respect to the society's, in such a way that each member can pursue its individual goals without preventing other members to pursue theirs.

From a sociological point of view, conventions result when members of a population adhere to some behavior, which is neither dictated nor enforced by a central authority. They can be regarded as rules followed by most members of the society, which are created and self-perpetuated by such members. Lets compare this observation with the requirements for a system to be considered self-organizing. According to [7], self-organization is: "...a process in which pattern at global level of system emerges from numerous interactions among lower-level components of the system. Moreover, the rules specifying the interactions between the system's components are executed using only local information, without reference to the global pattern". Hence, the emergence of social conventions can be regarded as a self-organizing process. Therefore, we aim at creating a computational self-adapting mechanism that allows engineering the emergence of social conventions in MAS, using self-organization as a guiding principle. This mechanism can become a highly promising tool to endow open and dynamic MAS with self-regulating capabilities.

One of the trends of thought in social studies is that conventions emerge by propagation or contagion, where social facilitation and imitation are key factors [8, 6]. From the MAS point of view, the studies in [18] and [17] show that convention emergence is possible. However, these works limit to analyze propagation, leaving out innovation (the discovery of rules), which is a very important factor for the evolution of societies. When the aim is to help a MAS reach conventions in dynamic environments, propagation may not be enough since this assumes that at least some agent in the society knows the appropriate behavior, and this is not always the case. Additionally, the problem can become even more difficult when the aim is to not only to reach (any) convention(s), but the best convention(s).

Moreover, studies in complex networks have shown that the effectiveness of propagations (be it of rumors, diseases or computer viruses) in societies is closely influenced by the topology of social interactions [1, 15, 19] (henceforth referred to also as *interaction topology*). Along this direction, the so-called *scale-free* and *small-world* networks are the most relevant ones because they model the most common networks appearing in societies and nature (e.g. film-relations between actors, the protein-protein interaction and the Internet, just to name a few). Thus, they have been employed to model societies in MAS. In particular, the emergence of conventions over different network topologies has been already studied by Pujol et al. in [16]. Nonetheless, Pujol et al. do not consider societies completely dominated by the less efficient conventions.

In this paper we attempt at going beyond finding conventions. We propose an evolutionary computational mechanism that facilitates agents in a MAS to self-organize and self-adapt in such a manner that the best conventions dynamically emerge for a wide range of interaction topologies. At this aim, we take inspiration on the argument in the social sciences literature that social conventions arise from social contagion [6]. Along this line, our approach borrows from the social contagion phenomenon to exploit the notion of *positive infection*: agents with *good behaviors* become infectious to spread such behaviors in the agent society. Thus, good behaviors propagate as an infectious disease. Moreover, to prevent stagnation on less efficient conventions, we incorporate an innovation mechanism that allows agents to constantly explore new behaviors in hope of finding better ones. By combining infection and innovation our computational model helps MAS to establish better conventions even when less efficient conventions are fully settled in the society. Furthermore, we empirically show how our approach empowers agents to reach the best set of conventions for a wide range of interaction topologies.

Notice that further evolutionary approaches appear in the literature. Indeed, convention emergence has been described as an evolutionary process [3], and so evolutionary algorithms (EA) have been employed to find conventions in agent societies. Nevertheless, they are usually applied either: (i) as a centralized process [11]; or (ii) as an individual self-contained process for each agent [14]. Both approaches can be potentially slow and tend to be *off-line* processes. By off-line we mean they require complete information regarding the system (at least up to certain point in time). Moreover, an off-line process requires to be run separately from the system execution. This makes off-line mechanisms unsuitable for dynamically adapting conventions, which understandably require an *online* mechanism that promptly reacts to the changes in the system. The later case is our purpose.

To summarize, in this article we present an online, self-organizing and self adapting mechanism, based on the social contagion phenomenon, that allows agents in a MAS to find and establish good (if not the best) social conventions that regulate their behavior.

The paper is organized as follows. Section 2 formalizes the problem we tackle. In section 3 we present our evolutionary infection-based model. Section 4 empirically evaluates our model for the problem described in section 2 over a wide range of interaction topologies. Finally, section 5 draws some conclusions.

2 Formalizing the Problem

In this article, we base the formalization of our self-adaptation problem in the agent and self-organizing system models presented by Biskupski, et al. in [5]. Firstly, we consider that a MAS is composed of a set of autonomous agents, Ag, and that no central authority exists to rule them. The relationships (neighborhoods) between these agents are given by an interaction topology, $E \subseteq Ag \times Ag$. If $(ag_i, ag_j) \in E$, then ag_i and ag_j are neighbors, and thus they can interact with each other. We also consider that the MAS state changes as agents act, and that all the possible system states are defined by the finite set S. The actions that agents can perform are defined by the finite set, Δ. Thus, the function $\Pi : S \times (\Delta \times \Delta)^{|E|} \to S$ models the MAS state-to-state transition, where $(\Delta \times \Delta)^{|E|}$ stands for the possible actions all the pairs of neighboring agents can perform when interacting with each other. Finally, we consider that not all states are equally valuable. Hence, the need for a (abstract) social welfare function $u : S \to \mathbb{R}$ that values the quality of some desired MAS property for a particular system state. Grouping together the components described above, we can formally define the our notion of MAS:

Definition 1 (Multi-agent System) *A multi-agent system is characterized as a six-tuple $\langle Ag, E, S, \Delta, \Pi, u\rangle$, where Ag stands for a finite set of agents, $E \subseteq Ag \times Ag$ represents the finite set of possible interaction topologies, S stands for a finite set of system states, Δ is a finite set of agent actions, $\Pi : S \times (\Delta \times \Delta)^{|E|} \to S$ stands for the system state transition function, and $u : S \to \mathbb{R}$ is the social welfare function.*

Now it is time to formally characterize the agents in a MAS. Agent-wise, we consider that agents within a MAS only work with partial (local) knowledge and that they are *social*. We label them as social because: (i) each agent can communicate and interact with other agents within the MAS; and (ii) agents have social rules that regulate their interactions with other agents. Therefore, notice that agents perform *communicative* actions: (i) to interact with each other, or (ii) to exchange information. On the one hand, when deciding the next action to take, agents depart from their individual internal state. This internal state encompasses the agent's *local* knowledge (obtained from local interactions) and the social rules it is aware of. The agent's set of social rules expresses the behavior that it currently believes is beneficial to itself and the society. We assume that agents can measure the utility of following their social rules after performing a particular action. On the other hand, agents must decide whether to share their information and with whom, as well as what to do with the information they receive from their peers. In this respect, we consider agents to have some *feedback* message exchange mechanism that allows them to share with their neighbors the social rules they are aware along with the associated valuations. Consequently, agents can validate their social rules through their peers.

Finally, each agent must present a mechanism that, based on the results after interacting with other agents along with their feedback, allows it to adapt its social rules and subsequently obtain better results through future interactions. Bundling all the agent features mentioned so far, we propose the following formal definition of *social-aware agent*.

Definition 2 (Social-aware Agent) *Given an agent, $ag_i \in Ag$, we say tat it is social-aware if it can be characterized as a tuple $\langle e_i^t, s_i^t, \pi, u_i, \mathcal{F}_i, \mathcal{A}_i \rangle$, where:*

- *e_i^t is the neighborhood of the agent at time t;*
- *$s_i^t : (M_i^t, R_i^t)$ is the agent's internal state at time t;*
  - *M_i^t: is the agent's memory at time t;*
  - *R_i^t: is the agent's set of social rules at time t;*
- *$\pi : s_i^t \rightarrow \Delta$ is the action selection mechanism.*
- *$u_i : s_i^t \rightarrow \mathbb{R}$ is the agent's utility function;*
- *$\mathcal{F}_i : (\mathcal{F}_I, \mathcal{F}_O)$ is the agent's feedback mechanism, where $\mathcal{F}_I$ stands for the input feedback received from other agents, and $\mathcal{F}_O$ models the output feedback sent to other agents;*
- *$\mathcal{A}_i$ is the agent's adaptation mechanism.*

Let Ag^t, stand for the agent population in the MAS at time t. When social-aware agents populate a MAS, we can define the system state at time t as $S^t = (s_i^t)_{i \in Ag^t} \in S$; and the interaction topology at time t as, $E^t = \cup_{i \in Ag^t} e_i^t \in E$. Furthermore, this MAS will present self-organizing and self-adapting properties if the appropriate feedback and adaptation mechanisms can be found, such that agents can distributively reach localized (if possible global) conventions of their social rules that allows the MAS to reach a social (global) goal. Where a (localized) convention stands for a group of neighboring agents that present a (localized) consensus [5] regarding their social rules.

Thus, we define the problem we aim at solving, the so-called *distributed self-adaptation problem* (DSAP), as follows:

Definition 3 (DSAP) *Given a multi-agent system, $\langle Ag, E, S, \Delta, \Pi, u \rangle$ where Ag is a finite set of social-aware agents, the distributed self-adaptation problem is that of finding for each agent, $ag_i \in Ag$, the feedback and adaptation mechanisms, $\mathcal{F}_i$ and $\mathcal{A}_i$ respectively, that allow each agent to dynamically assess the social rules at each time t, R_i^t, that maximize the social welfare u.*

3 An Evolutionary Infection-Based Mechanism

Next we propose a computational mechanism to solve the DSAP, namely to help agents in a MAS reach *social conventions* that maximize the social welfare. At this aim, we assume that we can accomplish our goal by maximizing agents' *individual welfares*. Thus, we stay in page with the distributed nature of the problem.

It has been argued in the social sciences literature that behavior conventions in societies are reached through social contagion [6]. This phenomenon relates to the spreading of behaviors between individuals to an infectious disease. Hence, we chose to model the social contagion phenomenon into a MAS framework. However, as stated above, we target at beneficial conventions, and if possible we prefer the ones that tend to maximize the social welfare. Considering the social welfare as a composition of individual welfares, it makes sense to let the individual behaviors that impact positively on it, here named *good behaviors*, be more infectious. Nevertheless, with this positive infection we can achieve at most a total replication of the best-known behavior among agents. Obviously, this is not enough. We also require some mechanism to explore new behaviors. Moreover, the infectious spreading of behaviors needs to be interweaved with the constant and continuous search for new, better ones.

By this means we expect that a MAS can reach conventions that are dominant in the society so that no better ones can be found and no worst ones can upstage them. If so, we say that a MAS has reached a *stable state*. However, if some unaccounted factor(s) alter(s) the MAS in such a manner that the current (stabilizing) conventions become obsolete (the social welfare deteriorates), the infectious process will re-configure the MAS to drive it to another stable state. Thus, making it a self-adapting mechanism.

A contagion process that constantly evolves to create new and more powerful infections whenever the situation demands it can be regarded as an evolutionary process. Hence, an evolutionary algorithm (EA) [13] seems a suitable candidate for our implementation. Nevertheless, because of the nature of the problem, the self-organization principles must also stay present.

In our infection-based EA, outlined in algorithm 1, each agent has a set of genes that encodes its social rules. Agents can infect other agents with their genes following the *survival of the fittest* concept: the fittest the agent (the highest its individual welfare), the more infectious. Furthermore, our algorithm realizes innovation (exploration) by letting agents mutate their genes. Notice though that infections also contribute to innovation by recombining genes that may result in new genes. Importantly, our algorithm runs distributedly: each agent decides whether to infect or mutate based on local knowledge.

Thus, each agent is endowed with: i) an **evaluation function** (line 4) to assess its individual welfare; ii) a **selection process** (line 5) to choose a peer to infect, out of its local neighborhood, based on its fitness; iii) an **infection operator** (line 6) to inject some of its genes into the selected agent (with probability $p_{infection}$); and iv) an **innovation operator** (line 7) to mutate its genes (with probability $p_{mutation}$), thus creating new behaviors.

We implement the selection operator by adapting the roulette selection in the classic GA literature [4] to make it decentralized. We realize infection by using a classic crossover recombination. The classic crossover (*single-cut crossover*) randomly selects a cut point in the parents' gene sequences to exchange their genes and produce two new individuals. Consider a contagious agent and an agent to infect as two parents. Instead of creating child individuals, an infection operator combines the genes of both parents. Furthermore, there is no restriction on the number of agents each agent can infect (per iteration), but no agent can be infected twice. Therefore, the fittest agents enjoy more opportunities to spread. We realize innovation by having each agent randomly change its genes with a certain probability.

Agents in a MAS continuously interact until the *incubation time*, $t_{incubation}$, expires. Thereafter all agents locally start their evolutionary processes. Once this process finishes, agents resume their interaction. The incubation time bounds the time intervals used by agents to assess how the changes to their genes reflect on their behavior, analogously to viral infections requiring some time to influence an organism.

If we look a little bit in our algorithm we can see that it fulfills the required roles to solve the DSAP. On the one hand, the evaluation and selection take the part of the feedback mechanism. Through this methods each agent gets some understanding (feedback) on the performance of their social rules, since they can compare them with respect to other ones in their neighborhood. Thus, gaining knowledge of possible improvement and of the current localized social convention (consensus). On the other hand, infection allows agents to reach a consensus while at the same time proposing small improvements. Therefore, we introduce mutation, as a purely innovative component, to our algorithm, to finally realize an adaptation mechanism which empowers agents to continuously find and settle conventions of social rules with respect to a social goal.

1: **repeat**
2:     let agents interact for time $t_{incubation}$;
3:     **foreach** ag $\in$ MAS **do**
4:         $ag.evaluate()$
5:         $ag' \leftarrow ag.selection()$
6:         $ag.infection(ag', p_{infection})$
7:         $ag.mutation(p_{mutation})$
8:     **end for**
9: **until** MAS stops

Algorithm 1: Infection-based Algorithm.

4 Empirical Evaluation

We hypothesize that our infection-based computational mechanism can be applied to solve the DSAP. Given a multi-agent system, we shall consider the DSAP as solved if algorithm 1 can: (i) self-organize the agents in the MAS to reach the best social convention(s), (maximize the social welfare), for a wide range of initial social rule configurations and under the most common interaction topologies; and (ii) realize self-adaptation in the presence of dynamic (changing) conditions.

At the aim of validating these assumptions, two sets of experiments (self-organization and self-adaptation) were designed to empirically evaluate our mechanism under different conditions of a particular MAS.

Studies in the literature [12] have shown that games are well suited for testing *coordination* between individuals. Thus, agents in our MAS interact with each other by engaging in iterative games. Each game has multiple rounds. During a round, each agent plays a with a random neighbor agent. In a game both agents do some action in $\Delta = \{A, B\}$. The actions are constrained in each agent by its current social rules (R_i^t). Games are rewarded with a payoff, which is then accumulated. After a game each agent, ag_i, keeps on its memory, M_i, its opponent's action (overwriting its current content). In this work, for the sake of simplicity we only kept the opponent's last action. We leave for future work to consider a memory of larger size.

The number of rounds for each iterative game is given by $t_{incubation}$. When a game finishes, each agent starts its own evolutionary process (algorithm 1) employing its accumulated payoff as its fitness value for the evaluation function. Notice that the mechanism for selecting agents for infection is independent of the opponent agents chosen to play games. Once the evolutionary process ends, each agent resets its accumulated payoff and clears its memory so that the next iterative game can begin. Thus, the individual-welfare u_i of each agent is given by the accumulated payoff between iterative games.

4.1 Interaction Topologies

As mentioned above, the propagation of infectious diseases has been studied by epidemiology and complex networks [9, 19, 15]. It is well known that its behavior its affected by the type of topology on which the population interacts. Since algorithm 1 is inspired on infections, it is reasonable to think that the interaction topology of a MAS may influence its performance. In order to empirically analyze our infection-based model we chose the following interaction topologies:

Small-world These networks present the small-world phenomenon, in which nodes have small neighborhoods, and yet it is possible to reach any other node in a small number of hops. This type of networks are *highly-clustered* (i.e. have a high clustering coefficient). Formally, we note them as $W_V^{k,p}$, where V is the number of nodes, k the average connectivity, i.e., the average size of the node's neighborhood, and p the re-wiring probability. We used the Watts & Strogatz model [19] to generate these networks.

Scale-free These networks are characterized by having a few nodes acting as highly-connected hubs, while the rest of them have a low connectivity degree. Scale-free networks are *low-clustered* networks. Formally we note them as $S_V^{k,-\gamma}$, where V is the number of nodes and its degree distribution is given by $P(k) \sim k^{-\gamma}$, i.e. the probability $P(k)$ that a node in the network connects with k other nodes is roughly proportional to $k^{-\gamma}$.

Random graphs are networks with a clustering coefficient that tends to zero. Although these networks do not appear in nature, we also study the behavior of our model over them for the sake of completeness. They are formally noted as $R_V^{<k>}$, where V is the number of nodes and k the average connectivity.

4.2 Coordination Game

An agent's interactions with its neighbors have the form of an *n*-player iterative game, where the game length is defined by the $t_{incubation}$ parameter. At each iteration of the game, an agent receives a payoff based on its current action and the action of the neighbor playing in that iteration according to the matrix in table 1.

Table 1. Game Payoff Matrix

		Agent j: A	Agent j: B
Agent i	A	(α,α)	(-1,-1)
	B	(-1,-1)	(1,1)

The pay off matrix can help capture pure coordination games [18][17] and coordination games with equilibrium differing in social efficiency [16]. Thus, if $\alpha = 1$ the matrix represents a pure coordination game; and when $\alpha > 1$ a game with different social efficiencies. Notice that regarding the latest use if both agents do B (coordination in B), a suboptimal payoff is achieved, since coordination in A offers a higher payoff to both agents.

Since agents are social-aware, each agent $ag_i \in Ag$ has two parameterized rules: one to help it decide what action to take based on the last opponent's (peer's) past action; and another to decide the action to take when no past action is known. To this end, each agent can record on its memory the last action performed by its last opponent without distinguishing who the opponent was. The parameterized rules are represented by the following templates:

RT_i^{start} : **if** [empty(M_i)] **then** do(X_i^0)

RT_i^{react} : **if** [M_i = X_i^0] **then** do(X_i^1) **else** do(X_i^2)

where M_i stands for the contents of ag_i memory and X_i^0, X_i^1, X_i^2 are variables over Δ.

Rule template RT_i^{start} constrains agent ag_i to perform the action assigned to the variable X_i^0 whenever the memory is empty, i.e. at the beginning of each iterative game. Rule template RT_i^{react} chooses X_i^1 or X_i^2 based on the action performed by the last opponent. Notice that agents learn their rules by finding values for $X_i = \{X_i^0, X_i^1, X_i^2\}$, thus obtaining instances of the rule templates above. For example, if at time t, agent ag_i has values $X_i = \{A, A, B\}$ then its rule templates resolve into rules: R_i^t =\{*if [empty(M_i)] then do(A) , if [M_i = A] then do(A) else do(B)* \}. Therefore, this instance of the DSAP is solved if algorithm 1 can stand for the feedback and adaptation mechanisms, $\mathcal{F}_i$ and $\mathcal{A}_i$ respectively, that allows agents in the MAS to learn the values of X_i that maximize the social-welfare, u.

This game is interesting because no global knowledge of the MAS is required by the agents, namely the identities of opponents are not needed at each iteration nor the payoff matrix of the game. However, the lack of knowledge of the identities makes the MAS more complex since it causes strong interdependencies between agents' actions. To illustrate the effect of interdependencies consider, for example, that agent ag_i rules are given by $X_i = \{A, A, B\}$ and its memory by $M_i = \{A\}$. Say that at iteration t, he interacts with agent ag_j, who does action B. Then ag_i memory changes to $M_i = \{B\}$. At iteration $t+1$, ag_i interacts with ag_k. By following R_i^{react}, ag_i does B because M_i is different from X_i^0. Thus, if ag_k does A then both agents suffer a payoff drop caused by ag_j previous action.

We know before-hand that four cooperative-only solutions exist for this game, and that they are the strongest attractors in the MAS. We call them as cooperative-only because they try to cooperate by repeatedly doing the same action regardless of the past interactions. Two of them always make agents do A ($X_i = \{A,A,A\}$ and $X_i = \{B,A,A\}$), and another two make agents always do B ($X_i = \{A,B,B\}$ and $X_i = \{B,B,B\}$). Henceforth, we shall refer to them as *A-Conventions* and *B-Conventions*. The A-Conventions are the best (give higher payoffs) whenever $\alpha > 1$.

4.3 Self-Organization Empirical Results

As mentioned at the beginning of the section, one of our aims is to verify if our infection-based mechanism can self-organize the agents to reach the best social convention(s) for a wide range of configurations. Therefore, the experiments presented in this section were designed with that purpose.

To that extend, each experiment is defined by a combination of: i) an interaction topology model; ii) a payoff matrix; and iii) an initial social rules distribution, namely the agents' initial rule settings. We run **50 simulations** of each experiment. In all simulations agents interact and infect each other according to the game in section 4.2 during 20000 ticks. Our empirical settings were:

Topologies We generated each interaction topology as described in section 4.1. Their parameters as well as their average clustering coefficient were : $W_{1000}^{<10>,0.1} = 0.492$, $S_{1000}^{<10>,-3} = 0.056$ and $R_{1000}^{<10>} = 0.020$. Notice that we generated a new interaction topology per simulation.

Payoff matrix Using table 1 we defined three payoff matrices: a pure coordination game ($\alpha = 1$), and two games with different social efficiencies ($\alpha = 1.5$, $\alpha = 2$).

Initial rule distribution At simulation startup, we initialized the norms (X_i values) of every agent using five distributions: i) **Random** (rules are randomly set); ii) **Attractor-free** (rules set from the non-cooperative-only rules); iii) **Low sub-optimal** (rules of 25% of the agents set from the B-Conventions ; iv) **High sub-optimal** (75% of agents with rules from the B-Convention); and v) **Fully sub-optimal** (rules of all agents were set from the B-Conventions).

The parameters of algorithm 1 where set to: $p_{infection} = 0.10$, $p_{mutation} = 0.0003$, and $t_{incubation}$=10.

To measure if a convention is established, we counted the number of agents with the same values for X_i per tick, and the number of agents doing A or B per tick. The counts of each simulation in the experiment where then aggregated using the inter-quartile mean.

4.3.1 Pure coordination game

This games are represented by a payoff matrix with $\alpha = 1$. The experiments show that the population organizes either to an A-convention or a B-convention depending on the number of agents initially doing A or B. Figure 1(a) shows that if the initial rule distribution leads to more than 50% of the agents doing action A, then an A-convention is established; otherwise, a B-convention settles down. Importantly, a MAS self-organizes toward the cooperative-only social rules even though for this game other social conventions exists that can achieve the same result. For instance, if all agents adopt the rule given by $X_i = \{A, A, B\}$, then every agent will always do A. We hypothesize that the cooperative-only rules emerge as a way for the society to overcome the problem generated by actions' interdependencies.

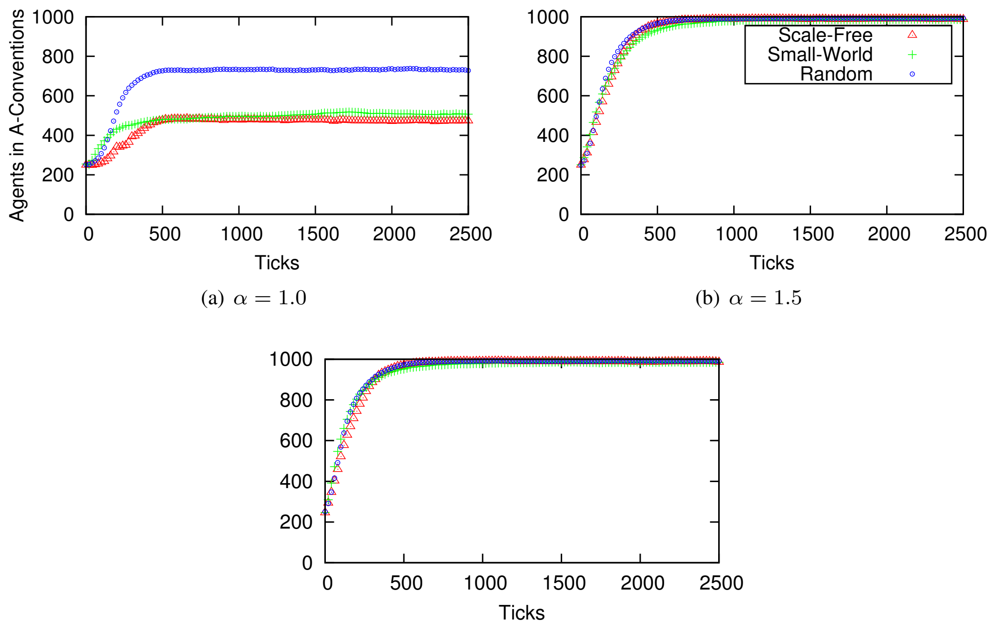
Since the A and B-conventions are equally valuable, we can say that for this case the MAS manages to organize itself to establish one of the best social conventions regardless of the initial rule distribution. Furthermore, this is accomplished independently of the interaction topology.

4.3.2 Different social efficiencies

This type of games are defined by matrices with $\alpha > 1$. When using **random** initial distribution, around 25% of the agents have rules from the A-conventions and around 50% of them do action A. Figures 1(b) and 1(c) show that in this case the agents in the MAS readily organize into an A-convention for both $\alpha = 1.5$ and $\alpha = 2.0$ independently of the interaction topology. Regarding the **Attractor-free** initialization, the agents promptly adopt an A-convention too, even though at startup no agent knew the best rules. As to the **Low sub-optimal** initialization, the same behavior can be observed.

Departing from a **High sub-optimal** distribution, agents in a MAS establishes a B-convention when $\alpha = 1.5$ for all interaction topologies. However, by setting α to 2.0, agents in the small-world networks are able to agree upon an A-convention. Thus, we conclude that agents will not consider a new convention if its benefit is not considerable enough. Moreover, it seems that for the scale-free case a greater benefit is needed.

The **Fully sub-optimal** distribution represents the worst case scenario because initially all agents share the convention of always doing B. In this case, innovation through mutation becomes a key factor. When the innovation probability is low, like the one we have been using in the experiments above, the agents are unable to converge to the best convention. This is because a low mutation equals to a low



(a) $\alpha = 1.0$

(b) $\alpha = 1.5$

(c) $\alpha = 2.0$

Figure 1. Results of experiments with random initial rule distribution.

number of agents changing their rules. This is a problem because the agents changing social rules will suffer from low accumulated payoffs. To illustrate this problem, say that a low mutation results in a very low number of agents adopting rules from an A-convention. In a small-world network, most likely all neighbors of each agent are constrained by their social rules to always do action B. Thus, an agent doing A will frequently lose at the iterative game, obtaining a negative payoff. This forces the agent to give in to peer pressure (neighbors doing B) and change its social rules to the B-Convention. On the other hand, infections on scale-free networks are known to be hard to eradicate once they have settled [10, 15]. This is equivalent to trying to overrun a settled infection with a new one. A known approach to accomplish that is for the new infection to surge from highly connected nodes (hubs), and from there start spreading. However, when only a small number of agents mutates, the chances of this spreading in one of the few hub nodes is very low, making the task of upstaging the current infection impossible.

At this aim, we increased the mutation probability. We empirically found that using $p_{mutation} = 0.055$ allows agents in scale-free networks with $\alpha = 2.0$ and small-world networks with $\alpha > 1$ to organize into an A-Convention (in small-world even a smaller probability suffices). The process through which an A-convention overcomes a B-Convention is interesting. First, a small group of agents playing *tit-for-tat* kind of rules (repeating the opponent's last action) starts to appear. Figure 2 shows that agents with this social rules appear almost from the start. Agents with this strategy can coexist with B-Convention agents with a small or non-negative effect to their accumulated payoffs because, even though they lose some games, they get the biggest payoff or at least the second one frequently. Thus, they become hard to infect by agents with B-convention rules (visible in the left-hand plots of figure 2 by the almost constant number of agents with the tit-for-tat-like rules). Moreover, when agents with A-convention rules appear, they have a higher chance of having neighbors that will cooperate with them. In other words, they can take advantage of this established group to get higher payoffs and so be able to start spreading their rules to other agents. However, a high mutation presents the disadvantage that a small part of the population will be constantly mutating. In our MAS this effect is translated to having around 80% of the agents converging to do action (right-hand plots of figure 2). Most of them with A-convention rules and a small fraction with tit-for-tat-like, sub-optimal rules (left-hand plots in figure 2). Finally, random networks require even a higher probability ($p_{mutation} = 0.6$) to establish the best convention.

Table 2 summarizes the innovation rates necessary to establish the best conventions under different conditions. Overall from these results we can observe that *highly-clustered* agent communities (e.g. small-world) are more open to positive infections, where as the *low-clustered* ones (e.g. scale-free) are harder to infect if a stable infection is already in place. This is similar to some results shown by the scenarios studied in [16]. However, our evolution-

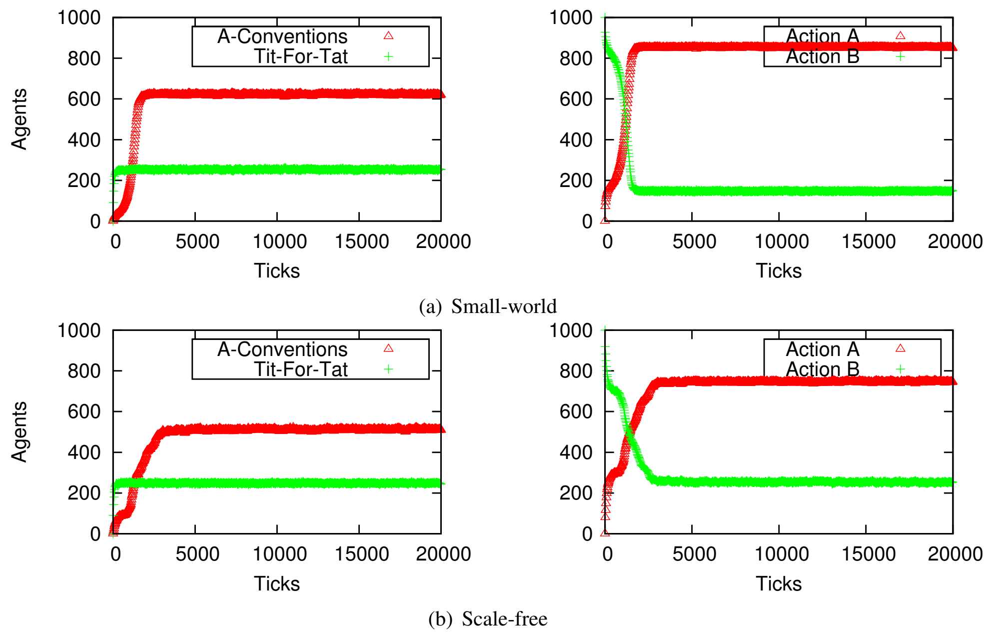


Figure 2. Results of experiments with full sub-optimal initialization. Graphs show on the left the number of agents per rule; on the right the number of agents doing each action

Table 2. Innovation rates to establish the best convention in a MAS

Initial Norm Distribution	Small-world		Scale-free		Random	
	1.5	2.0	1.5	2.0	1.5	2.0
Random	L	L	L	L	L	L
Attractor-free	L	L	L	L	L	L
Low sub-optimal	L	L	L	L	L	L
High sub-optimal	H	L	H	H	VH	VH
Full sub-optimal	H	H	H	H	VH	VH

L = Low(0.0003), H = High(0.055), VH = Very High(0.06)

ary model can overcome the difficulty of re-infecting *low-clustered* networks by using a high innovation through mutation rate. The high innovation is required because an innovating agent with high peer pressure is instantly affected by a decrease in its payoff. Nevertheless, there is an associated cost to this high innovation, in the form of a small subgroup of agents unable to settle on a set of rules

Finally, we can claim that i) a convention is always reached, and ii) under certain conditions this convention is the best one for all topologies. Moreover, when these conditions are not met, e.g. a suboptimal convention is fully established, our model can still reach the best social convention through innovation.

4.4 Self-Adaptation Empirical Results

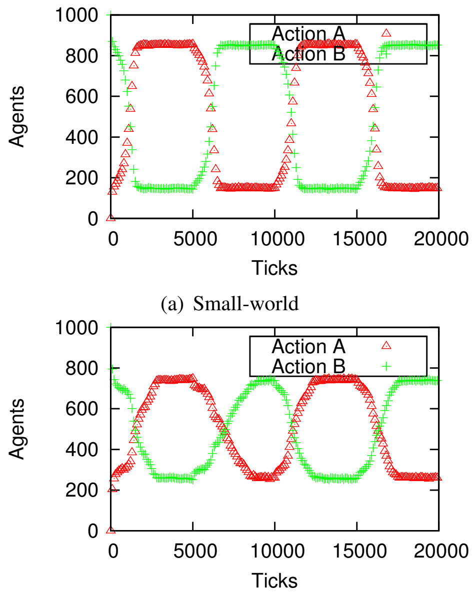
In the previous section, it was shown that our infection-based mechanism endows social agents in a MAS with self-organization capabilities. Next, we shall show through experiments that it also functions as a self-adaptation mechanism that allows agents to re-organize themselves in the presence of dynamic changes. Thus, proving that our proposed algorithm solves the DSAP.

The experiment definition is the same as the one used in section 4.3 with the addition of a dynamic component. This dynamic component can take the form of either run-time changes in the payoff matrix, *dynamic payoff matrix*, or an ever-changing agent population , *dynamic population and neighborhood*.

4.4.1 Dynamic payoff matrix

We simulate a dynamic environment by introducing changes into the payoff matrix at run time. The changes take the form of swapping the efficient action, which means that if A is the most efficient action then after the swap B will become the most efficient one (i.e the payoff values of cooperating in A are swapped with the values of cooperating in B) and vice versa. Notice that agents are not explicitly informed when the payoff matrix changes. Instead, they realize that the games they play lead to different results.

We performed experiments using: the matrices with different social efficiencies ($\alpha = 1.5$ and $\alpha = 2.0$); the scale-free ($S_{1000}^{<10>,-3}$) and small-world ($W_{1000}^{<10>,0.1}$) topologies; and the full sub-optimal social rules initialization (the worst case scenario). With respect to matrix changes, we introduced action swaps every 5000 ticks. Regarding the infection-base mechanism parameters, the same from the previous section where used with the innovation rate set to



(a) Small-world

(b) Scale-free

Figure 3. Results of the dynamic payoff matrix experiments

high.

Figure 3 shows the number of agents performing each of the possible actions. We observe that after each matrix change (at ticks 5000, 10000 and 15000) the agents quickly re-organize into conventions that result in performing the most efficient action. We also observe that in the small-world case, the *reaction time* is faster than the scale-free. By reaction time we mean the time elapsed between the dynamic change and the re-organization. This result was expected since the small-world is the most clustered one. The results from the experiments clearly show that our infection-based mechanism endows agents with self-adapting capabilities.

4.4.2 Dynamic population and neighborhood

In this environment, the number of agents in the MAS changes and so their neighborhoods. In practice these environment changes are achieved by dynamically changing the network topology. Specifically: 1) we create a scale-free network interaction topology up to certain number of agents; 2) we begin the social agent interactions using the after-mentioned topology; and 3) after every k number of simulation ticks, add new agents and define the new neighborhoods. We experimentally implemented this by interweaving the Barabasi-Albert (BA) scale-free network generation algorithm [1], and the MAS simulation. In other words, the MAS and the BA algorithm are executed at the same time.

The component combination used for the experiment was: a payoff matrix with $\alpha = 2.0$; scale-free topology that

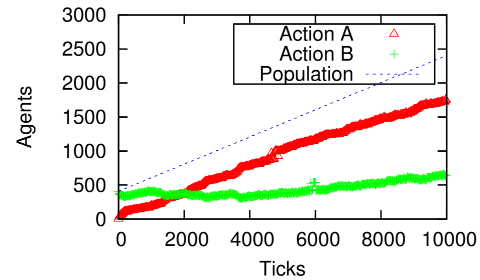


Figure 4. Results of the dynamic population experiments

started at $S_{400}^{<10>,-3}$ and ended at $S_{2400}^{<10>,-3}$; and a full sub-optimal social rules initialization for both the initial agents and the new ones. The new agents were added every 50 simulation ticks. The infection-based parameters were set to the same previously used.

The experiments show that, even in a MAS with a continuous influx of agents with less that optimal social rules, our mechanism is able not to only reach the best convention, but also to sway most of the incoming agents into performing the most efficient action. Figure 4 shows the number of agents performing each action per time step. We observe that even though at the beginning no agent performs action A, a convention over that particular action starts to emerge regardless of the continuously incoming agents performing action B. Thus, i) our mechanism allows agents to reach the best convention in dynamic populations; and ii) when the best social convention has emerged, it empowers incoming agents with pre-established (non-optimal) social rules to *adapt* its rules to conform to the best social convention.

In summary, we claim that our infection-based mechanism presents self-adaptation properties, since it allows agents (using only local information) to dynamically change their rules in response to dynamic changes in the system.

5 Conclusions and Future Work

In this paper we proposed an evolutionary computational mechanism, based on the concept of positive infection, that endows a MAS with self-organization and self-adaptation capabilities. At the macro-level this can be observed through the distributed emergence of social conventions. These social conventions regulate and balance the behaviors of the agents as individuals and as members of a society.

We ran separate experiments to validate our mechanism with respect to the self-organization and self-adaptation aspects. Furthermore, because it is well known from studies in biological and artificial viruses that the underlying topology of social interactions affects how infections spread, we designed our experiments to validate the behavior of our mechanism under different topologies and agent settings (initial social rules distributions).

On the one one hand, the self-organization results of our infection-based mechanism for highly-clustered topologies, are in line with the studies in [16]. However, our mechanism is able to reach the best social convention even in cases where the majority of the population is already dominated by a sub-optimal social convention. On the other hand, low-clustered societies are more sensitive to the initial distributions of social rules. With our algorithm, agents reach the sub-optimal convention if the population starts with a high number of agents with sub-optimal rules. Nevertheless, by introducing a high innovation rate it is possible to upstage sub-optimal conventions with the best one regardless of the topology. The experiments also showed that when a population has reached the best convention, it cannot be overthrown by a sub-optimal one.

The mutation as an innovation mechanism was shown to be a key factor in destabilizing established sub-optimal conventions, while still letting the best one to take over most of the population in a stable manner. However, this comes with a price, since having a high number of agents constantly innovating prevents the infection from taking over the whole population. Nevertheless, for a dynamic and complex MAS this is a small price to pay in exchange for a distributed behavior regulation mechanism. Furthermore, it is interesting to notice that on low-clustered topologies tit-for-tat-like strategies emerge in subpopulations and become stable allowing the more efficient social rules taking over sub-optimal dominated populations. This is congruent with [2], who argues that tit-for-tat is a robust strategy.

On the other hand, the self-adaptation experiments show that the infection-base mechanism is robust to different dynamic changes to the system. Agents adapt, if necessary, to reach new conventions only using local information. Moreover, agents can reach the best convention even when a continuous number of incoming agents try roots for a different convention. Besides that, after the best social convention has been reached, incoming agents quickly adapt their social rules to conform with this convention.

Finally, for future work we plan to analyze in more detail the rule infection spreading with the purpose of understanding the most basic underlying principles for convention emergence. Our motivation is that of learning how to direct the behavior of a MAS by deploying populations of agents aimed at distributedly regulating the MAS behavior.

Acknowledgments

The authors thank CONACyT for the scholarship of the first author. This work was funded by IEA (TIN2006-15662-C02-01), OK (IST-4-027253-STP), eREP(EC-FP6-CIT5-28575) and Agreement Technologies (CONSOLIDER CSD2007-0022, INGENIO 2010).

References

[1] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47, 2002.

[2] R. Axelrod. *The Evolution of Cooperation*. Basic Books, 1984.

[3] R. Axelrod. An evolutionary approach to norms. *The American Political Science Review*, 80(4):1095–1111, 1986.

[4] T. Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, Oxford, 1996.

[5] B. Biskupski, J. Dowling, and J. Sacha. Properties and mechanisms of self-organizing manet and p2p systems. *ACM Trans. Auton. Adapt. Syst.*, 2(1):1, 2007.

[6] R. Burt. Social contagion and innovation: Cohesion versus structural equivalence. *American J. of Sociology*, 92:1287–1335, 1987.

[7] S. Camazine, N. R. Franks, J. Sneyd, E. Bonabeau, J.-L. Deneubourg, and G. Theraula. *Self-Organization in Biological Systems*. Princeton University Press, Princeton, NJ, USA, 2001.

[8] R. Conte and M. Paolucci. Intelligent social learning. *Artificial Society and Social Simulation*, 4(1):1–23, 2001.

[9] J. Delgado. Emergence of social conventions in complex networks. *Artificial Intelligence*, 141(1):171–185, 2002.

[10] Z. Dezso and A.-L. Barabási. Halting viruses in scale-free networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 65(5):055103, 2002.

[11] I.-K. Jeong and J.-J. Lee. Evolving multi-agents using a self-organizing genetic algorithm. *Applied Mathematics and Computation*, 88:293–303, 1997.

[12] D. Lewis. *Convention*. Cambridge: Harvard University Press., 1969.

[13] Z. Michalewicz. *Genetic Algorithms+Data Structures=Evolution Programs*. Springer Verlag, 3rd edition, 1996.

[14] D. Moriarty, A. Schultz, and J. Grefenstette. Evolutionary algorithms for reinforcement learning. *Artificial Intelligence Research*, 11(1-1):241–276, 1999.

[15] R. Pastor-Satorras and A. Vespignani. Epidemic dynamics and endemic states in complex networks. *Physical Review E*, 63:066117, 2001.

[16] J. Pujol, J. Delgado, R. Sangüesa, and A. Flache. The role of clustering on the emergence of efficient social conventions. In *IJCAI 2005*, pages 965–970, 2005.

[17] Y. Shoham and M. Tennenholtz. On the emergence of social conventions: Modeling, analysis, and simulations. *Artificial Intelligence*, 94(1-2):139–166, 1997.

[18] A. Walker and M. Wooldridge. Understanding the emergence of conventions in multi- agent systems. In *ICMAS95*, pages 384–389, 1995.

[19] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.