Update Dynamics, Strategy Exchanges and the Evolution of Cooperation in the Snowdrift Game

Carlos Grilo[1,2], Luís Correia[2]

[1] Dep. Eng. Informática, Escola Superior de Tecnologia e Gestão,
Instituto Politécnico de Leiria Morro do Lena, Portugal
[2] LabMag, Dep. Informática, Faculdade Ciências da Universidade de Lisboa, Portugal
grilo@estg.ipleiria.pt, luis.correia@di.fc.ul.pt

Abstract. We verify through numerical simulations that the influence of the update dynamics on the evolution of cooperation in the Snowdrift game is closely related to the number of strategy exchanges between agents. The results show that strategy exchanges contribute to the destruction of compact clusters favorable to cooperator agents. In general, strategy exchanges decrease as the synchrony rate decreases. This explains why smaller synchrony rates are beneficial to cooperators in situations where a large number of exchanges occur with synchronous updating. On the other hand, this is coherent with the fact that the Snowdrift game is completely insensitive to the synchrony rate when the replicator dynamics transition rule is used: there are almost no strategy exchanges when this rule is used.

Key words: evolution of cooperation, update dynamics, asynchronism.

1 Introduction

The existence of cooperation in nature has been challenging to explain since, from an evolutionary point of view, this type of behavior is apparently less advantageous than a selfish one [12]. This problem is also of central importance in social sciences [1] and especially on the development and maintenance of artificial societies [11], where it is relevant to study how cooperation may be promoted and sustained. Evolutionary games are models used to study these phenomena. In these models, a population of agents interacts during several time steps through a given *game* which is used as a metaphor for the type of interaction that is being studied. The underlying structure that defines who interacts with whom is called the *interaction topology*. After each interaction session, some or all the agents, depending on the *update method* used, have the possibility of changing their strategies. The *strategy update process* is modeled using a so called *transition rule* that emulates the fact that agents tend to adapt their behavior to the context in which they live by imitating the most successful agents they know. It can also be interpreted as the selection step of an evolutionary process in which the least successful strategies tend to be replaced by the most successful ones.

In the research areas of dynamical systems and evolution of cooperation, *synchronous updating* has been the most used update method: at each time step, all the elements of the system are updated at exactly the same time. This practice has been widely questioned, the argument being that perfect synchronism is absent from the real world [8][9]. The most common alternative to synchronous updating is *sequential updating*, which is an extreme case of asynchronism: at each time step, exactly one element is updated. It has been shown that the level of cooperation achieved and the dynamics of these models, can be significantly affected if such an asynchronous updating is used. Previous studies on the Prisoner's Dilemma game, played on regular lattices under the *best-neighbor* transition rule, reported that synchronous updating supports more cooperators than sequential updating [8][9]. The results are the same for the Snowdrift game played under the same conditions [13]. When this game is played using the *proportional rule* (see Section 2) it was found that sequential updating favors cooperation [13]. In our work with both games [2][3][5] we confirmed the results of previous works but also found that asynchronous updating is detrimental for cooperation for very small noise values only, especially for regular networks. We also showed that the influence of the update dynamics depends mainly on the noise present in the strategy update process [4]: asynchronism becomes increasingly beneficial to cooperators as the noise level grows up to a certain value. Finally, it was found that both games are insensitive to the update dynamics when the *replicator dynamics* rule is used [6][13]. However, we showed that this rule becomes sensitive to the synchrony rate if agents are allowed to imitate less successful agents, which is equivalent to raising the noise level [4].

Here, we verify the idea suggested by Tomassini et al. [13] that strategy exchanges between agents are the reason why sequential updating supports more cooperators than synchronous updating when a proportional transition rule is used. In Section 2, the model used in the simulations is described. In Section 3, we show and discuss the results of the simulations in order to verify this idea. Finally, in Section 4, some conclusions are drawn and future work is advanced.

2 The Model and Simulations Setup

We use a model very similar to [13]. The Snowdrift (SD) game, also known as Hawk-Dove, is a two-player game where there is a task which takes a cost c to be completed and which pays a benefit $b > c$ to each player, regardless of their participation in the task completion. If both cooperate (C), they divide the cost, which results in a payoff of $b - \frac{c}{2}$ to each; If only one cooperates, it receives a payoff of $b - c$, while the defecting agent (D) receives b. If none cooperates, both receive nothing. Given these conditions, it follows that the best action depends on the opponent's decision: the best thing to do is to take the opposite action the opponent takes. As is common practice, we set $c = 1$ which leads to a cost-to-benefit ratio of mutual cooperation $r = 1/(2b - 1), 0 \leq r \leq 1$.

At each time step, agents first play a one round game with all their neighbors. After this, each agent updates its strategy with probability α using a *transition*

rule (see below). The update is done synchronously by all the agents selected to engage in the update process. The α parameter represents the *synchrony rate* and is the same for all agents. It allows us to cover all the space between synchronous and sequential updating: $\alpha = 1$ models synchronism; as $\alpha \rightarrow \frac{1}{n}$, where n is the population size, the model approaches sequential updating.

Small-world networks (SWNs) [14] are used as interaction topologies as in [13]: first a toroidal regular 2D grid is built so that each node is linked to its 8 surrounding neighbors; then, with probability ϕ, each link is replaced by another one linking two randomly selected nodes. Self, repeated links or disconnected graphs are not allowed. These networks have the property that, even for very small ϕ values, the average path length is much smaller than in a regular network, maintaining a high clustering coefficient. Both these properties are commonly observed in real social systems. As $\phi \rightarrow 1$, we get random networks with both small average path lengths and clustering coefficients.

Two different transition rules are used to model the strategy update process: the *generalized proportional* (GP) [10] and the *replicator dynamics* (RD) [7]. Let G_x be the average payoff earned by agent x, N_x be the set of neighbors of x and c_x be equal to 1 if x's strategy is C and 0 otherwise. According with the GP rule, the probability that an agent x adopts C as its next strategy is

$$p_C(x, K) = \frac{\sum_{i \in N_x \cup x} c_i (G_i)^{\frac{1}{K}}}{\sum_{i \in N_x \cup x} (G_i)^{\frac{1}{K}}}, \qquad (1)$$

where $K \in\,]0, +\infty[$ is the noise present in the strategy update process. Noise is the possibility that an agent imitates strategies other than the one used by its most successful neighbor. $K \rightarrow 0$ corresponds to the *best-neighbor* rule in which x always adopts its best neighbor's strategy. With $K = 1$ we have a linear *proportional* rule. Finally, for $K \rightarrow +\infty$ we have random drift where payoffs play no role in the decision process. Usually, the interval $K \in\,]0, 1]$ is used.

According to the RD rule, the updating agent x imitates a randomly chosen neighbor y with probability $\frac{G_y - G_x}{b}$ if $G_y - G_x > 0$. Here, b is the largest possible payoff difference between two players in a one shot game. Notice that in this rule agents do not imitate neighbors with lower payoffs.

All the simulations were performed with populations of $50 \times 50 = 2500$ agents, randomly initialized with 50% of Cs and 50% of Ds. When the system is running synchronously, i.e., when $\alpha = 1$, we let it first run during a period of 900 iterations which, we confirmed, is enough to pass the transient period of the evolutionary process. After this, we let the system run for 100 more iterations and, at the end, we take as output the average proportion of cooperators, ρ_C, and the average number of strategy exchanges during this period. Simulations where $\alpha \neq 1$ are setup so that the number of individual updates is approximately the same as in the $\alpha = 1$ case. Each run is a combination of r (SD game), ϕ (SWNs), K (only for the GP rule) and α. For each tested combination, 30 runs were made and the average of these runs is taken as the output.

3 Strategy Exchanges and Cooperation

Tomassini et al. [13] suggested that, when the proportional rule ($K = 1$) is used, synchronous updating leads to less cooperation than sequential updating because in the former case agents may exchange their strategies, which is not possible in the last case. However, the authors did not verify this idea, which is the purpose of this paper. This idea raises some questions that are not answered by the authors. The first one is concerned with the transition rule: it is true that both the best-neighbor ($K = 0$) and the replicator dynamics rules do not allow direct strategy exchanges between two connected agents as the proportional rule does. That is, two agents x and y cannot infect each other simultaneously. However, it is possible that they exchange strategies indirectly: x can be infected by an agent a having the same strategy as y while y is infected by another agent b having the same strategy as x. It is not obvious also that strategy exchanges are disadvantageous for cooperators. After a strategy exchange, be it direct or not, the number of cooperators and defectors remains the same. This means that we must verify if there is a relation between the number of strategy exchanges and the final proportion of cooperators and, if it exists, we need to verify if there is a cause-effect relation between the two aspects. Finally, if we conclude that strategy exchanges negatively affect cooperation, one must try to explain why it is so.

We first measure the average number of strategy exchanges, as a function of α. This average is taken over periods of $\frac{1}{\alpha}$ time steps so that the number of individual updates considered is approximately the same as for the synchronous case ($\alpha = 1$). The observed result is a strategy exchange decrease as α decreases (Fig. 1(a) shows an example). Exceptions to this result happen almost exclusively for small noise levels ($K = 0$ and $K = \frac{1}{100}$) when the GP rule is used: for some r values the number of strategy exchanges for $\alpha = 0.5$ is larger than for $\alpha = 1$ (Fig. 1(b)). In these situations, the smaller number of strategy exchanges for $\alpha = 1$ is due to cyclic dynamics in the asymptotic phase, resulting from the deterministic nature of the model (synchronous and best-neighbor rule). Even so, for the most part of these situations ρ_C is larger for $\alpha = 1$ than for $\alpha = 0.5$. For larger K values, the number of strategy exchanges is larger for $\alpha = 0.5$ than for $\alpha = 1$ only in rare cases. This happens when, for $\alpha = 1$, the system converges to uniform populations of Ds, where no exchanges can occur (ex: $r = 0.9$ in Fig. 1(a)).

Figs. 1(a) and (b) also exemplify how the number of strategy exchanges vary with K for the GP rule (K values used: $0, \frac{1}{100}, \frac{1}{10}, \frac{1}{8}, \frac{1}{6}, \frac{1}{4}, \frac{1}{2}, 1$): strategy exchanges grow with K, specially for $\alpha = 1$. Only results for $\phi = 0.05$ are shown but this pattern arises for all the tested topologies (ϕ values used: 0, 0.01, 0.05, 0.1, 1). On the other hand, we verified that there are almost no strategy exchanges when the RD rule is used: the maximum number of strategy exchanges never exceeds 4 during each $\frac{1}{\alpha}$ period no matter the α value used. We recall that, when the GP rule is used, smaller α values are more beneficial for cooperators as K grows (see Section 1). We also recall that the game is completely insensitive to α when the RD rule is used. This suggests that there
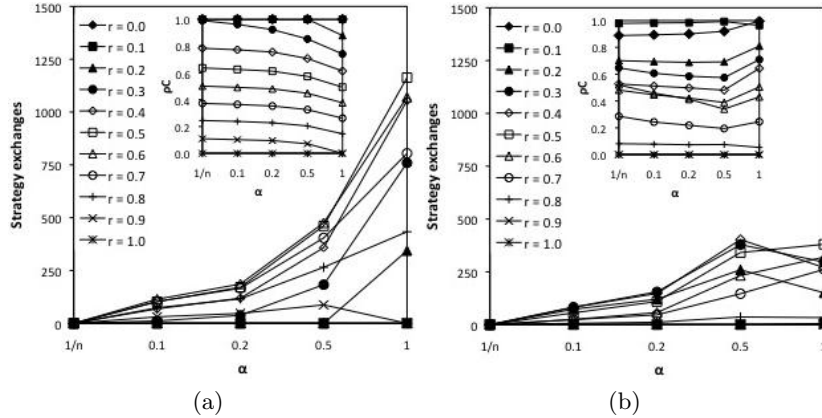
Fig. 1. Number of strategy exchanges and ρ_C (insert) as a function of α, when the game is played on SWNs ($\phi = 0.05$) using the GP rule. (a) $K = 1$, (b) $K = 0$.

is indeed a close relation between the number of strategy exchanges and how the model reacts to α changes. In other words, the larger the difference between the number of strategy exchanges occurring under synchronous and sequential updating, the larger is the difference between the level of cooperation achieved with synchronous updating (less cooperation) and sequential updating (more cooperation). However, this result does not allow us to establish a cause-effect relation between the number of strategy exchanges and the proportion of Cs since strategy exchanges are just a consequence of the input parameters.

In order to verify the effect of strategy exchanges on the level of cooperation, we did the following experiment: on each time step, either a randomly chosen agent is updated with probability $1-p$ using the transition rule or two randomly chosen neighbor agents exchange their strategies with probability p. We note that no strategy exchanges can occur as a result of the utilization of the transition rule since only one agent is selected. This means that, when $p = 0$, this is a sequential system. Fig. 2(a) and (b) exemplify the effect of strategy exchanges on the level of cooperation, respectively for the GP and RD rules, when the game is played on regular networks. For reasons of space only these two examples are shown but the results are similar no matter the noise level (only GP rule) and the interaction topology used. Results for $p = 1$ are not shown also because in this regime we have random drift, where payoffs do not influence the dynamics, and the cooperation level always converges around 0.5, which is the initial proportion of cooperators. The results are the following: when the game is played under the GP rule, the proportion, ρ_C, of Cs decreases as the probability of strategy exchanges increases. On the other hand, when the RD rule is used, the game is almost insensitive to p.

The difference between these two behaviors can be understood if we look at the spatial patterns formed by the agents during the evolutionary process. Fig.
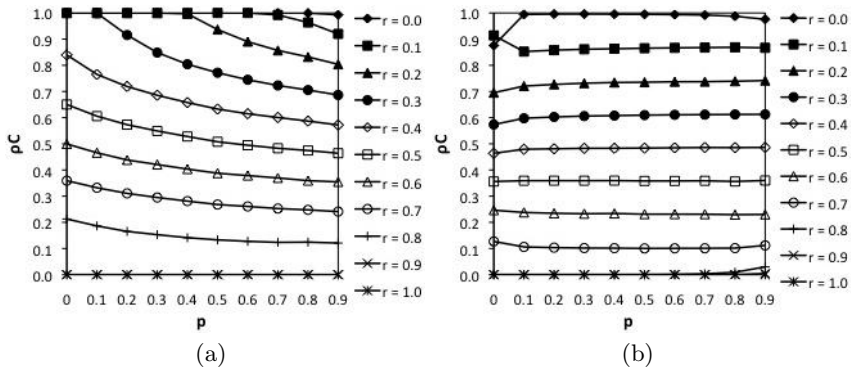
Fig. 2. Proportion of cooperators, ρ_C, as a function of p when the game is played on regular networks ($\phi = 0$) using (a) the GP rule ($K = 1$) and (b) the RD rule.

3 shows two examples of asymptotic spatial patterns achieved with the GP rule, for $p = 0$ and $p = 0.9$. When no strategy exchanges are allowed, C agents organize into more compact clusters. This is a well known phenomena: structured populations allow C agents to form clusters so that they interact mainly with each other, thus protecting themselves from exploration by D agents. This is important to understand how strategy exchanges influence the level of cooperation. A strategy exchange between a C and a D in the fringe of a cluster pushes the C away from the other Cs. At the same time, it introduces a D inside the cluster or, at least, it contributes to more irregular cluster frontiers, which is also detrimental for cooperators [10]. This can be seen in Fig. 3(b) where there are more isolated C agents and more filament-like clusters. The situation is different when the RD rule is used: C agents organize into filament-like clusters and this pattern does not change when strategy exchanges are introduced (Fig. 4). That is, when the RD rule is used, agents do not organize into compact clusters even when there are no strategy exchanges. This means that when strategy exchanges are introduced, there are no compact clusters to destroy and that is the reason why both the spatial patterns and ρ_C are not affected.

4 Conclusion and Future Work

We verified the idea from Tomassini et al. [13] that, when the Snowdrift game is played under the proportional transition rule ($K = 1$), sequential updating supports more cooperators than synchronous updating because in the last case strategy exchanges may occur, which is not possible in the former case. The results of the simulations, put together, are a strong evidence that this idea is correct. We saw that strategy exchanges are detrimental to the evolution of cooperation because they destroy compact clusters of agents when these exist, which is disadvantageous for cooperators. The results show that, when the *generalized*
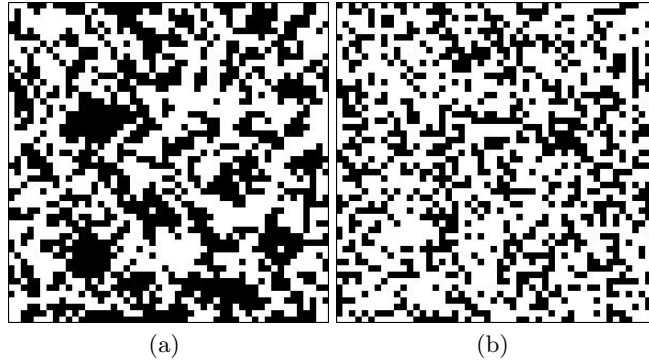
Fig. 3. Typical asymptotic patterns when the game ($r = 0.6$) is played on regular networks ($\phi = 0$) using the GP rule ($K = 1$). (a) $p = 0$, $\rho_C = 0.4692$; (b) $p = 0.9$, $\rho_C = 0.3636$. Colors: black for cooperators and white for defectors.
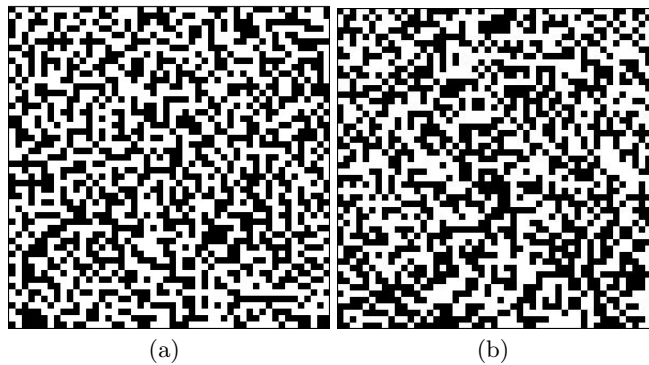
Fig. 4. As in Fig. 3 but for the RD rule and $r = 0.4$. (a) $\rho_C = 0.4668$; (b) $\rho_C = 0.4788$.

proportional transition rule is used, the number of strategy exchanges grows considerably as the noise level gets larger, mainly for synchronous updating. This explains why smaller synchrony rates increasingly favor cooperation as the noise level grows. On the other hand, when the *replicator dynamics* rule is used, there are almost no strategy exchanges, which explains why this rule is completely insensitive to the synchrony rate of the system. The way how evolutionary games depend on the update dynamics has already been reported in other works, including ours, but this work, complementing the work by Tomassini et al., offers an explanation for how they react to changes in this parameter. The results presented in this paper are also in line with the one achieved in our previous work that the sensitivity of evolutionary games depends mainly on the noise level present in the strategy update process [4]. This is important because it means that, in order to build less sensitive artificial societies, special care should be

taken in the design of strategy update processes and agents' perception skills, namely by trying to avoid less successful agents to be imitated.

In future developments of this work we will verify this result with the Prisoner's Dilemma game. The way this game reacts to synchrony rate changes is similar to Snowdrift. It is, however, more sensitive than the Snowdrift when the GP rule is used, which turns it a good candidate for the task of verifying how general this result is.

5   Acknowledgements

References

1. Robert Axelrod. *The Evolution of Cooperation*. Penguin Books, 1984.
2. Carlos Grilo and Luís Correia. Asynchronous stochastic dynamics and the spatial prisoner's dilemma game. In *13th Portuguese Conference on Artificial Intelligence, EPIA 2007*, pages 235–246. Springer-Verlag, 2007.
3. Carlos Grilo and Luís Correia. The influence of asynchronous dynamics in the spatial prisoner's dilemma game. In *Proceedings of Animals to Animats - 10th Int. Conf. on the Simulation of Behavior (SAB'08)*, pages 362–371. Springer-Verlag, 2008.
4. Carlos Grilo and Luís Correia. What makes the spatial prisoner's dilemma game sensitive to asynchronism? In *Proceedings of the 11th Int. Conf. on the Simulation and Synthesis of Living Systems, Alife XI*, pages 212–219. MIT Press, 2008.
5. Carlos Grilo and Luís Correia. The influence of the update dynamics on the evolution of cooperation. *International Journal of Computational Intelligence Systems, Special Issue on Computational Intelligence Issues in Systems Experiencing Nonlinear Dynamics and Synchronization*, 2(2):104–114, 2009.
6. Christoph Hauert and Michael Doebeli. Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature*, 428:643–646, 2004.
7. Josef Hofbauer and Karl Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998.
8. Bernardo Huberman and Natalie Glance. Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences*, 90:7716–7718, 1993.
9. David Newth and David Cornforth. Asynchronous spatial evolutionary games: spatial patterns, diversity and chaos. In *Proceedings of the 2007 IEEE Congress on Evolutionary Computation*, pages 2463–2470, 2007.
10. Martin Nowak, Sebastian Bonhoeffer, and Robert M. May. More spatial games. *International Journal of Bifurcation and Chaos*, 4(1):33–56, 1994.
11. Jae C. Oh. Cooperating search agents explore more than defecting search agents in the internet information access. In *Proceedings of the 2001 Congress on Evolutionary Computation, CEC2001*, pages 1261–1268. IEEE Press, 2001.
12. John M. Smith. *Evolution and the Theory of Games*. Cambridge University Press, 1982.
13. Marco Tomassini, Leslie Luthi, and Mario Giacobini. Hawks and doves on small-world networks. *Physical Review E*, 73(1):016132, 2006.
14. Duncan Watts and Steven H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.