PhD THESIS

Models of social behaviour based on game theory

Jelena Grujić

UNIVERSIDAD CARLOS III DE MADRID

PhD THESIS

Models of social behaviour based on game theory

Author

Jelena Grujić

Supervisors

José Antonio Cuesta Ruiz

Angel Sánchez Sánchez

DEPARTMENT OF MATHEMATICS

Leganés, September 2012

MODELS OF SOCIAL BEHAVIOUR BASED ON GAME THEORY

AUTOR: Jelena Grujić

DIRECTORES: José Antonio Cuesta Ruiz and Angel Sánchez Sánchez

Firma del Tribunal Calificador:

		Firma
PRESIDENTE:	Pablo Brañas-Garza	....
VOCAL:	Tom Lenaerts	....
SECRETARIO:	Yamir Moreno	....

Calificación:

Leganés, 21 de septiembre de 2012

Mojoj porodici

Acknowledgments

It would not have been possible to write this thesis without the help and support of many people, of which only a few I can mention here.

First and foremost, I would like to thank my supervisors José Cuesta and Anxo Sáchez for their help, support and patience. They encouraged and guided me, leaving me room to work in my own way and respecting my opinion. I will always appreciate the fact that they were always available for me, along with all the valuable discussions we had during this long journey.

I own my deepest gratitude to Arne Traulsen and all the members of his group in The Max Plank Institute for Evolutionary Biology in Plön, Germany, where I did my internship. The time I spent there was invaluable for my development as a scientist. The knowledge I acquired there in a very short time, both about my scientific subject and about being a scientist in general, was priceless for my progress. I am especially grateful to Arne, for all his support and his brilliant insight regarding my work.

I was fortunate enough to spend my formative years in not one but two amazing educational institutions which profoundly influenced my life and set me on the path I am following to this day. The first one is The Mathematical High School, Belgrade, Serbia, and the other is Petnica Science Center, Serbia. Apart from giving me a solid background in mathematics, physics, programming, etc., these two shining examples of what education should be like, taught me how to be analytical, resourceful, skeptical and curious, but above all how to love and appreciate science. For this, I am indebted to all the individuals whose selfless effort makes these institutions so unique and exceptional.

It was an honor for me to have such great colleagues who accompanied me during this amazing and also difficult time. To Luis, Giovanna, Carlos, Kenier, Svetozar, Edoardo, Juanma, Daniele and many, many others who were there to listen to my problems, offer helpful advice, or just be there for me when I needed the support to move on, I offer my most sincere thanks.

And last but not least, a big thank you to my family, who always believed in me, supported me and helped me with anything I needed. I also cannot forget my friends who always encouraged me. I am especially grateful for your patience and understanding during my long absence while I was writing this thesis.

Jelena Grujić
Leganés, 2012

This work was supported in part by Ministerio de Ciencia e Innovación (Spain) through grants PRODIEVO, MOSAICO, FPI y Estancias Breves, and by Comunidad de Madrid (Spain) through grant MODELICO

Contents

1

Introduction

The emergence of cooperation created a puzzle for generations of scientists across several disciplines (Pennisi 2009). Why an individual would sacrifice herself for another when natural selection favors the survival of the fittest? Charles Darwin himself remarked the paradox of a worker bee that labors for the good of the colony, although its efforts do not lead to its own reproduction. He proposed that selection might favor families whose members were cooperative, and it is accepted today that kinship helps explain cooperation. But defectors, those who benefit without making a sacrifice, are likely to evolve because they will have an advantage over individuals who spend energy on helping others, therefore jeopardizing the stability of any cooperative effort. Yet cooperation and apparently even altruism have evolved and remained, on any level of biological organization. Without cooperation, genomes, cells or multicellular organisms would have never been formed (Maynard Smith and Szathmary 1995). There are numerous examples of cooperation in both animal and human kingdoms and to this date there is no widely acceptable explanation why.

The suitable theoretical framework to address this issue is evolutionary game theory (Axelrod and Hamilton 1981; Axelrod 1984), which has been intensively used for this research during recent years. Its main virtue consists in that it allows to pose the dilemmas involved in cooperation, and the mechanisms proposed to explain it, in a simple and rigorous manner. In what follows, we will introduce the basic concepts from game theory and then we will discuss some possible mechanisms which lead to the promotion of cooperation.

1.1 Game theory

Game theory formalizes mathematically how individuals or groups interact with each other. Each individual has a set of options she can choose from. Depending on their choice and the choices of others, the individuals obtain some benefit. In the game theory framework, we call individuals *players*, the decisions they make are *actions*, and the series of actions which fully describe the players' behavior are *strategies*. In the case of one shot games the actions and strategies are equivalent. The benefits players obtain in the game are referred to as *payoffs*.

If player a uses the same strategy i in every possible situation we call the strategy i a *pure strategy*. However, player a can use different strategies i from the set S_a of player a's strategies, each with a probability p_i. This is referred to as the *mixed strategy* $\mathbf{p}$. It is formally written as:

$$\mathbf{p} = (p_1, p_2, \ldots, p_{N_a}), \qquad p_i \geq 0, \sum_{i \in S_a} p_i = 1, \tag{1.1}$$

where $N_a = |S_a|$ is the number of strategies available to player a (likewise N_b to player b). In the special case when $p_i = 1$ for strategy i and 0 otherwise, we say that player a uses the pure strategy i.

A common situation is that where only two players, a and b, are confronted in the game. If player a uses strategy i and player b uses strategy j, let a_{ij} denote the payoff player a collects. The set of payoffs for all strategies within S_a when confronted to all strategies within S_b is usually gathered in matrix form:

$$A_a = \underset{\text{player a}}{} \overset{\text{player b}}{\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1N_b} \\ a_{21} & a_{22} & \cdots & a_{2N_b} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_a1} & a_{N_a2} & \cdots & a_{N_aN_b} \end{pmatrix}}. \tag{1.2}$$

This matrix is known as the *payoff matrix* of player a. Player b also has her own payoff matrix A_b, which needs not be the same as player a's. In case that player a is playing a pure strategy i and player b plays the mixed strategy $\mathbf{q} = (q_1, q_2, \ldots, q_{N_b})$, then the expected payoff of player a is

$$(A_a\mathbf{q})_i = \sum_{j \in S_b} a_{ij} q_j. \tag{1.3}$$

Finally, if player a plays mixed strategy $\mathbf{p}$ and player b plays mixed strategy $\mathbf{q}$, then the expected payoff of the player a is

$$\mathbf{p} \cdot A\mathbf{q} = \sum_{i \in S_a, j \in S_b} a_{ij} p_i q_j \tag{1.4}$$

Elementary ideas of game theory could be found all through history, from Talmud to the writings of Charles Darwin. Although there were some important papers before, it is widely accepted that modern game theory starts with the classical work of von Neumann and Morgenstern (1944). Building on their results, John Nash later gave the game theory its modern methodological framework (Nash 1950). He introduced the concept of Nash equilibrium. It is a combination of strategies in which none of the players has anything to gain by changing only her own strategy. In other words, if player A is taking the best decision she can, taking into account the action of player B, and if player B is taking the best decision, taking into account the decision of player A, then player A and player B are in a Nash equilibrium. Formally written, we say that the pair of strategies $\mathbf{p}, \mathbf{q}$ are a Nash equilibrium if the two inequalities

$$\mathbf{p}\cdot A_a\mathbf{q} \geq \mathbf{p}'\cdot A_a\mathbf{q} \qquad \mathbf{q}\cdot A_b\mathbf{p} \geq \mathbf{q}'\cdot A_b\mathbf{p} \tag{1.5}$$

had for all strategies $p' \neq p$, $q' \neq q$. If the inequality is strict, we speak about a strict Nash equilibrium.

Since it was introduced the Nash equilibrium is the most widely used "solution method". Nevertheless, just because the players are in a Nash equilibrium it does not mean they earn the highest possible payoff. In many cases, players could obtain a higher payoff if both simultaneously changed their strategy, as we will see later in the example of the Prisoner's Dilemma. The set of strategies which will give the highest possible payoff for the all players is called Pareto optimal. More precisely, the set of strategies is called Pareto optimal if there is no other set of strategies which improves the payoff of one or more players while maintaining the payoffs of others.

1.1.1 Game dynamics and replicator equation

The Nash equilibrium gives us the solution set of strategies, but tells us nothing about the dynamics of the game. We do not know how will the system come to that equilibrium. Furthermore, if there are more that one Nash equilibria in the game we do not know in which one the system will end up. Also, the Nash equilibrium is the solution of the game given two assumptions: rational players and perfect knowledge. Rational player are players whose only aim is maximizing their own payoff. Perfect knowledge means that all players are perfectly informed about the rules of the game and they can calculate their best option. They also know that their opponents know the rules and can calculate the best option. Furthermore, they know that their opponents know that they themselves can calculate the best option, etc. These two assumptions are not always applicable to the real situations, since people are not always able to calculate the dominant strategy or they assume that other people are not able to do it. Furthermore, they will just not always act rationally. For example, in experiments of the Ultimatum game with humans, the players rarely choose the rational strategy (Henrich et al. 2001). In the Ultimatum game, two players should split a certain amount of money. One players is the proposer and the other one is the responder. The proposer decides how to split the sum and the responder can accept or refuse the proposition. If the responder accepts the split, both players get the money; otherwise they do not get anything. The rational strategy for the proposer is to propose the smallest amount of money and for the responder to accept whatever is offered (because something is always better than nothing). However, in the experiments humans usually propose and accept fair splits. If an unfair split is proposed people will react emotionally and punish their opponent by refusing the offer, even at the expense of earning nothing.

Game theory was given a new framework by Maynard Smith and Price (1973); see also (Maynard Smith 1982), who analyzed the games in a population-dynamics setting. Unlike in classical game theory, instead of individuals we now have a population of players which interact randomly with each other. The idea is that the successful strategies spread by natural selection. Each individual in the population plays a fixed strategy (pure or mixed) and the payoff they obtain is related to the reproductive fitness: the higher their payoff the more offspring they have. Offspring will play the same strategy as their parent. The reproduction is asexual. We assume that the population is large and that players play an infinite number of rounds. Therefore, here no assumption of rationality or perfect knowledge is necessary. The notions of rationality and perfect knowledge are exchanged for the notion of fitness. Given these assumptions, the system will eventually evolve to an evolutionarily stable strategy (ESS). Once adopted by the whole population, the evolutionarily stable strategy cannot be invaded by a small number of individuals of a mutant strategy. Therefore, strategy $\mathbf{p}$ is an ESS if it performs strictly better in a mixed population whose huge majority of players play strategy $\mathbf{p}$ and the rest play the mutant strategy $\mathbf{q}$. Formally written,

$$\mathbf{p} \cdot A[(1-\epsilon)\mathbf{p} + \epsilon\mathbf{q}] > \mathbf{q} \cdot A[(1-\epsilon)\mathbf{p} + \epsilon\mathbf{q}]. \tag{1.6}$$

Consequently, $\mathbf{p}$ is an ESS, if two conditions are satisfied:

(i) $\mathbf{p}$ is a Nash equilibrium

$$\mathbf{p} \cdot A\mathbf{p} \geq \mathbf{q} \cdot A\mathbf{p}, \tag{1.7}$$

(ii) if $\mathbf{p} \neq \mathbf{q}$ and $\mathbf{p} \cdot A\mathbf{p} = \mathbf{q} \cdot A\mathbf{p}$, then

$$\mathbf{p} \cdot A\mathbf{q} > \mathbf{q} \cdot A\mathbf{q}. \tag{1.8}$$

Therefore, ESS is actually a refinement of the notion of Nash equilibrium, but they are not equivalent. Most notably not all Nash equilibria are evolutionarily stable.

The replicator equation (Taylor and Jonker 1978) incorporates all the previously stated assumptions into an evolutionary game dynamics. If there are N different types of strategists in the population, with frequencies given by $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ (where x_i is a frequency of strategy i) and A is the payoff matrix of the game, then we can write the law of motion $\mathbf{x}(t)$ in the following way:

$$\dot{\mathbf{x}}_\mathbf{i} = [(A\mathbf{x})_i - \mathbf{x}A\mathbf{x}]. \tag{1.9}$$

Here, the term $(A\mathbf{x})_i$ is the expected payoff of the individuals using strategy i and the term $\mathbf{x} \cdot A\mathbf{x}$ is the average payoff in the population in the state $\mathbf{x}$. Therefore if the payoff of the individuals playing the strategy i is larger that the average payoff of the population in the state $\mathbf{x}$, the frequency of these players will increase; otherwise it will decrease. If these two terms are the same for all the strategies i, the system will remain in that stationary state, also called a fixed point of the dynamics. Some stationary states will never be reached unless the system started in that state. These are unstable stationary states. However, some states act like an attractor and the system will end up in that state starting from a larger set of initial states. These are stable stationary states. All the initial states which end up in the same attractor form its basin of attraction. The attractors are often evolutionarily stable states, but it is not always like that. More precisely, an ESS is always an attractor, but an attractor is not always an ESS (Taylor and Jonker 1978). An example of a game where we have an attractor which is not an evolutionarily stables state is Zeeman's game (Zeeman 1980), where we have two attractors, but only one of them is an evolutionarily stable strategy. More about this in Chapter 3.

The replicator equation is not the only way to model evolutionary game dynamics, there are a few others (Nowak and Sigmund 2004; Hofbauer and Sigmund 2003). The main property of the replicator equation is that it allows the fitness to depend on the distribution of population types (the vector $\mathbf{x}$). When the fitness does not depend on $\mathbf{x}$ and the dynamics includes mutations, it is referred to as the quasi-species equation (Eigen and Schuster 1977). The replicator-mutator equation adds the mutation to a frequency-dependent selection. Adaptive dynamics describes how continuous traits or strategies are changing under mutations on a frequency dependent fitness landscape (Hofbauer and Sigmund 1990). Finally, the Price equation (Price 1970) provides a link between the replicator framework and adaptive dynamics (Page and Nowak 2002).

1.1.2 Symmetric 2×2 games

There are numerous examples of games, which can involve many players and many strategies, like the Public goods game, also known by the phenomenon it leads to, namely the Tragedy of Commons (Hardin 1968; Hardin 1971). Here we are going to focus just on the simplest case of symmetric 2×2 games (Rapoport et al. 1976). Those are the games between two players both having the same two strategies and the same payoff matrices. We call the strategies cooperate (C) and defect (D). The payoffs have illustrative names which come from the Prisoner Dilemma game, which we will explain later: reward for mutual cooperation (R), sucker's payoff (S), temptation to defect (T) and punishment for mutual defection (P). The payoff matrix of the game is:

$$\begin{array}{c} \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cc} C & D \end{array} \\ \begin{pmatrix} R & S \\ T & P \end{pmatrix} \end{array}. \tag{1.10}$$

For 2×2 games the notion of evolutionarily stable strategy and attractor are equivalent.

Let us solve these games. The replicator equation for 2×2 games is

$$\dot{x}_C = [(R-T)x_C - (P-S)(1-x_C)]x_C(1-x_C) \equiv F(x_C), \tag{1.11}$$

where x_C denotes the fraction of cooperators (consequently $1 - x_C$ is the fraction of defectors). We have three stationary states, $\hat{x}_C = 0$, $\hat{x}_C = 1$ and $\hat{x}_C = (P-S)/(R-T+P-S)$. Therefore, full defection and full cooperation will always be rest points, but that does not mean they are also stable. The stability of these strategies depends of the values of the payoffs. The last fixed point is

$$\hat{x}_C = \frac{P-S}{R-T+P-S}. \tag{1.12}$$

Depending on the values of the payoffs, $\hat{x}_C$ can have a meaningful value for frequency within $[0, 1]$ and can be an attractor or repeller. To determine the stability of the fixed point, we need to calculate the first derivative of the right hand side of the equation (1.11) in those fixed points. Consequently

$$F'(x)|_{x=\hat{x}_C} = \begin{cases} S-P, & \hat{x}_C = 0, \\ T-R, & \hat{x}_C = 1, \\ \frac{(R-T)(P-S)}{R-T+P-S}, & \hat{x}_C = \frac{P-S}{R-T+P-S} \end{cases} \tag{1.13}$$

Depending on the values of of R, T, P and S, we have 12 possible non equivalent games (Rapoport and Guyer 1966; Kilgour 1988). These games can be divided into 4 different categories with the following representatives: Harmony, Snowdrift, Stag Hunt and Prisoner's Dilemma.

The harmony game (Licht 1999) has payoff coefficients which satisfy $P < S < R$ and $P < T < R$ and as its name says, it represents the situation when the best possible outcome for both players is mutual cooperation. It is an example of the category where $R > T$ and $S > P$. For all the four games in this category there will be two fixed points, full defection and full cooperation, with full cooperation being the only one stable. Therefore, there is a unique strict Nash equilibrium in the pure strategy C, as illustrated in Figure 1.1.

The situation of the Snowdrift game (Sugden 2004) resembles the case of two drivers who are trapped on opposite sides of a snowdrift. Each of them has the option to shovel the snow (cooperate) or to seat in the car (defect). The best option is to stay in the car while your opponent is doing all the shoveling (T), the next best option is to do the shoveling together (R), then to do all the shoveling on you own (S) and finally the worst outcome is that both drivers stay in the car and nobody clears the road (P). Therefore the payoffs in the Snowdrift game satisfy $P < S < R < T$. In this game we have three Nash equilibria. Two equilibria are pure strategies, where one players plays C and the other D and the other way around. This is why this game is an anticoordination game, because the best response is to always do the opposite of your opponent. There is also a mixed strategy equilibrium, where the probability of cooperating is given by $\hat{x}_C$ in (1.12) and that of defecting by $1 - \hat{x}_C$. Let us now look at this game in the population-dynamic setting. The only ESS here is equivalent to the mixed strategy Nash equilibrium. Full cooperation and full defection will be unstable rest points of the dynamics (Figure 1.1). The Snowdrift game is a representative of the category $P < S$ and $R < T$. There are two more games equivalent to the Snowdrift game: Hawk-Dove (Maynard Smith and Price 1973), introduced to study conflicts between animal species, and Chicken (Russell 1959).

The Stag hunt game (Skyrms 2003) describes the situation where two persons go out to hunt. Each of them can decide individually to hunt a stag or a hare. The stag is a much better catch than a hare, but one person cannot catch it alone; on the other hand, if one goes for the hare, she can catch it alone, but the catch is much smaller. Therefore, the highest payoff is obtained when both players go for a stag (both cooperate) then if they both go for a hare (both defect), then if one goes for the hare and catches it alone (defects and the other cooperates) and finally the worst outcome is for the player who goes for the stag alone (cooperate but the other defects). Usually, it is $R > T > P > S$. It is better to go for hare alone and laugh at the guy who went for stag alone! In this game there are two Nash equilibria in pure strategies: if both players go for stag or if both players go for a hare. Therefore the best option is to play the same strategy as the other player (this game is a coordination game). However, if both players go for a stag they obtain a higher payoff, therefore this equilibrium is Pareto efficient. But if a player hunts a hare she faces a lower risk to lose, therefore this equilibrium is called risk dominant. There is also another Nash equilibrium in mixed strategies, with a probability of cooperating given by (1.12). In the population-dynamics setting this will result in three fixed points of the dynamics: two stable (always cooperate and always defect) and the third one unstable. Actually this unstable rest point separates the two basins of attraction. The two attractors will be also ESSs. This game is an example of the games in the category $S < P$ and $T < R$ (Figure 1.1).

The prototypical example of the last category of 2×2 games is the Prisoner's dilemma. Since in all the research presented here we have used this game, we will devote a separate section to it.

1.1.3 Prisoner's Dilemma

Prisoner's dilemma (PD) is a canonical example of a game analyzed in game theory. It was originally introduced by Merrill Flood and Melvin Dresher in 1950 as part of the Rand Corporation's research in game theory. Rand Corporation's interest in the subject was motivated by possible applications to the global nuclear strategy. The title, Prisoner's dilemma, and its most common interpretation today are due to Albert Tucker, who wanted to make Flood and Dresher's ideas more accessible to an audience of psychologists. In this interpretation of the game, police have arrested two suspects but do not possess enough information for a conviction. Therefore, they interrogate them in separate rooms. If nobody confesses then the police is left with circumstantial evidence, only enough to imprison them for a period of six months. In this case we say that prisoners cooperated. On the other hand, if both of them confess, now the police have enough evidence to imprison them for a period of 2 years. In this case

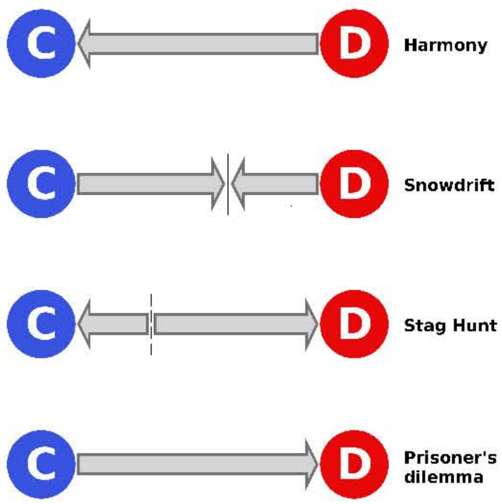


Figure 1.1: *Dynamics of different* 2×2 *games.*

we say that the prisoners defected. However, if one of them confesses (defects) and the other one does not confess (cooperates) the one who confesses goes free and the other one is imprisoned for 5 years. Therefore, no matter what the other prisoner does it is always better to defect, making defection a dominant strategy and the outcome of double defection a Nash equilibrium. However, if they both defect they will go to prison for 2 years whereas if they both cooperated they would serve a smaller sentence of 6 months each. Accordingly, this Nash equilibrium is not Pareto efficient. In the population-dynamics setting full defection will be an ESS, and full cooperation an unstable fixed point (Figure 1.1).

The names of the payoffs in 2×2 games come from the Prisoner's dilemma. They represent the sentence reductions. Accordingly a player obtains, the highest sentence reduction by defecting while her opponent cooperates (T - temptation to defect), the second best outcome is if both players cooperate (R - reward for mutual cooperation), then if both players defect (P - punishment for mutual defection) and finally the smallest payoff is for the players who cooperated while the other player defected (S - sucker's payoff). Hence, the following inequality holds: $T > R > P \geq S$.

The case when $P = S$ is known as weak Prisoner's dilemma. It has been of particular relevance in the study of the influence of spatial structure on social dilemmas since the pioneering work by Nowak and May (1992), and will be the case we will use along this work.

If two players play a Prisoner's dilemma more than once in succession, remember previous actions of their opponent, and change their actions accordingly, the game is called iterated Prisoner's dilemma. In addition to the general pattern of payoffs above, the iterative version also requires that $2R > T + S$ to prevent alternating cooperation and defection yielding a greater reward than mutual cooperation.

1.2 Promotion of cooperation

Without any mechanism for the evolution of cooperation in a PD, defection is the dominant strategy. However, several mechanisms have been proposed that can promote cooperation. The most prominent mechanisms are: kin selection, direct reciprocity, indirect reciprocity, group selection and network reciprocity (Nowak 2006). Here we will present the first four, and in the next section we will focus on network reciprocity which is the main focus our work. There are some other interesting mechanisms like the "green beard" effect (where players can recognize each other (Cohen et al. 2001)), making the game voluntary (Hauert et al. 2002), or introducing punishment (Yamagishi 1986; Fehr and Gächter 2000; Rockenbach and Milinski 2006).

1.2.1 Kin selection

The earliest mathematically formal treatments of kin selection were introduced by Fisher (1930) and later by Haldane (1932). Haldane is the one to whom the following sentence is attributed: "I will jump into the river to save two brothers or eight cousins". Later on, this work became the basis for what is known as Hamilton's rule (Hamilton 1964a; Hamilton 1964b). The idea behind this rule is that altruistic acts can be justified if the donor of the act is a genetic relative. That way the sacrifice will help individual's genes to survive. The rule is widely known as "kin selection", a term coined by Maynard Smith in 1964, or "inclusive fitness". Therefore, the higher is the relatedness between the two individuals the more justified is the altruistic act. Therefore the fitness of the behavior induced by a gene is not just determined by its ability to help the survival of the individual carrying it, it is also determined by the influence this behavior has on relatives which might carry the same gene. That way the cooperative behavior is the consequence of "selfish genes" (Dawkins 1976).

1.2.2 Direct reciprocity

Kin selection provided an explanation for some of the cooperation observed in human and animal societies. However, cooperation is not observed only between related individuals, often it can be seen between individuals who do not share any common genes or even belong to different species. A mechanism which could explain this kind of cooperation is proposed by Trivers (1971). The idea is that in an iterated Prisoner's Dilemma, since the players have repeated encounters it might pay off to cooperate in order to encourage your opponent to cooperate later. In the late 1970's Axelrod set out to find the best strategy in this framework (Axelrod 1984). He invited a number of well-known game theorists to submit their strategies in the form of a computer program, which would be played against each other. The payoffs used in the tournament were

$$\begin{array}{c c} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} 3 & 0 \\ 5 & 1 \end{pmatrix}, \end{array} \tag{1.14}$$

The winner of the tournament was a strategy called Tit for tat, submitted by Anatol Rapoport. In the strategy, the player's first actions is cooperation and in every other round it just repeats the action of her opponent from the previous round. Few years later Axelrod organized another tournament. A much higher number of strategies was submitted, many of which would have won the first tournament, but once again the winner was Tit for tat. However, it turned out that Tit for tat had a weak spot. Namely, in the presence of even very small noise in communication, Tit for tat's performance declines (Fudenberg and Maskin 1990; Selten and Hammerstein 1984). The problem with Tit for tat is that it cannot correct mistakes, therefore erroneous defections will lead to long chains of retaliations. To correct for this, Generous tit for tat was introduced (Nowak and Sigmund 1992). This strategy responds to cooperation with cooperation, but shows some level of forgiveness to defection. Namely, it will start with cooperation and copy unconditionally the previous action of the opponent if it is cooperation, just like Tit for tat, however if the opponent defected in the previous round it will still cooperate with some probability. For the payoffs in Axelrod tournament that probability is $1/3$. However, Generous tit for tat cannot exploit pure cooperators. Therefore it was eventually exchanged for the Win-Stay-Lose-Shift strategy (Nowak and Sigmund 1993). As the name says, in this strategy, if the player is satisfied with the benefit obtained she repeats the action from the previous round, otherwise she will switch the action to the opposite one. The satisfying payoffs are T and R, whereas P and S are unsatisfying. This strategy is capable to correct mistakes, but also exploits pure cooperators. A detailed study of one memory strategies for 2×2 game has been recently performed by Martínez-Vaquero et al. (2012).

1.2.3 Indirect reciprocity

Direct reciprocity can explain how cooperation evolves if we have repeated encounters between two individuals. However, in real life relationships are not always reciprocal. Many times we are in a position to do a favor to people although we do not need a favor from that person. We even help individuals we know we are never going to meet again. The proposed mechanism for the promotion of cooperation in this case is reputation. By helping somebody we earn a good reputation in the society and later people help us because of our good reputation. The process is called indirect reciprocity as you do not reciprocate directly to you benefiter, but indirectly through society. There are two flavors of indirect reciprocity: downstream and upstream. Downstream reciprocity (Nowak and Sigmund 1998; Wedekind and Milinski 2000; Milinski et al. 2002; Ohtsuki and Iwasa 2004; Brandt and Sigmund 2004; Nowak and Sigmund 2005; Milinski et al. 2001) occurs when a person who provided help in the past has higher chances of receiving help in the future. Upstream reciprocity is when a person receives help and because it feels good about it helps a third person. Although it is difficult to explain how would upstream reciprocity evolve (Boyd and Richerson 1989) it is observed in laboratory experiments with human subjects (van Damme et al. 2001; Nehring et al. 2001). Indirect reciprocity is almost exclusively a human phenomenon, although in some simpler form it is also noticed in animals. It is a very cognitively demanding task, it requires memory, observation and development of language to communicate one's experience. The calculations involving indirect reciprocity are very difficult and there is yet a lot to be discovered about its mechanisms.

1.2.4 Group selection

Although cooperative individuals will have lower fitness than non cooperative ones, groups of cooperative individuals will be able to perform tasks beyond the reach of individuals, and therefore might have higher fitness than non cooperative groups. This is called group selection. In a thorough study of the spatial dispersion of animal populations, Wynne-Edwards (1962) proposed that this principle applies to much of the animal kingdom. Maynard Smith can be credited with what has become known as the "haystack model" of group selection (Maynard Smith 1964). In this model, mice live in a field full of haystacks. Each haystack houses a group of mice. Cooperators benefit the group they live in, in such a way that the whole group reproduces faster, while the cooperators themselves reproduce less than defectors in the same group do. After a life cycle the haystacks are removed and all the mice mix up. The groups that contain many cooperators will contribute more to the metapopulation than groups with few cooperators when the haystacks are removed. Next year a new set of haystacks is set up and the process is repeated. After this model was proposed, there was some controversy about group selection. A few authors suggested that group selection is just a generalized kin selection, or that the effects of groups selection are negligible or both. Nevertheless, despite all this controversy, groups selection has been recently resurrected under the name of "multilevel selection" by Wilson and Sober (1994). Wilson, the developer of Multilevel Selection Theory, compares the many layers of competition and evolution to the Russian Matryoska Dolls within one another. The lowest level is the genes, next come the cells, and then the organism level and finally the groups. Multilevel Selection Theory does not lean towards individuals or groups selection but can be used to evaluate the balance between group selection and individual selection on a case-by-case scenario.

1.2.5 Network reciprocity

In the previous section we presented several mechanisms to explain how cooperation can emerge and be sustained under natural selection. The mechanism we are going to focus on here is the existence of a (social, spatial, geographical) structure that determines the interactions among individuals in the population (Axelrod 1984; Nowak and May 1992). Until now, we have assumed that the population is well mixed, meaning that everybody interacts with everybody. This is not a very realistic assumption. In everyday life people never have the opportunity to interact with everybody else on this planet. Their interactions are more often limited to their family, neighbors, co-workers, etc. Therefore, it is logical to introduce spatial structure into our models. In the spatial variant of the iterated PD players are placed on a network and they simultaneously play with each of their neighbors. They take only one action, either to cooperate (C) or to defect (D), the action being the same against all the opponents. The resulting payoff is calculated by adding interaction payoffs with all of their neighbors.

A pioneering model of network reciprocity was studied by Nowak and May (1992). They simulated a set of agents located on a square lattice, playing a PD with their Moore neighborhoods (all eight surrounding neighbors) and showed that cooperation could thrive even under very adverse choices of the payoffs. The authors explained this promotion of cooperation on the network as a result of the formation of clusters of cooperators. Namely, if a group of cooperators formed a cluster, where they mostly played with each other, then the cooperators in the cluster would earn more than the surrounding defectors. Therefore, cooperation, as the most successful strategy in the neighborhood, would spread and the cluster would grow until reaching the huge majority of the system.

Many papers were published after this seminal work (Szabó and Fáth 2007) with consistent results. Yet, in 2004 another import paper was published by Hauert and Doebeli (2004). In this paper they investigated a Snowdrift game and surprisingly got that promotion of cooperation is not nearly as large as in Nowak and May's paper. The change of the game cannot introduce such a difference, especially since the Snowdrift game allows more cooperation than the PD. Moreover it was previously shown that there is indeed promotion of cooperation in the Hawk-Dove game, which is equivalent to the Snowdrift game (Killingback and Doebeli 1996). The surprising result comes, not from the difference in the game, but from the difference in the update mechanism. In Nowak and May's paper the update mechanism used was unconditional imitation. Unconditional imitation is a deterministic rule where each player will imitate the action of the most successful neighbor in the previous round. However in Hauert and Doebeli's paper the update mechanism used is the replicator rule (Schlag 1998; Helbing 1992b). This rule, also known as proportional update (Helbing 1992a), is inspired by the replicator dynamics. If the players in the populations are labeled $i = 1 \dots N$, s_i is a strategy of player i, and W_i the payoff of that player, when we choose at random a player j from the neighborhood of player i, the probability that player i adopts the strategy of player j is given by

$$p_{ij}^t \equiv P\{s_j^t \to s_i^{t+1}\} = \begin{cases} (W_j^t - W_j^t)/\Phi, & W_j^t > W_i^t, \\ 0, & W_j^t \leq W_i^t, \end{cases} \tag{1.15}$$

where Φ is a constant appropriately chosen to keep the probabilities within the limits $[0, 1]$. These results shed light on the importance of the update mechanisms on the promotion of cooperation (Roca et al. 2009b). However, unconditional imitation and the replicator rule are not the only two update mechanisms available. Some of the most widely used ones are Moran's update (Moran 1962) and the Fermi rule (Szabó and

Töke 1998). Unlike unconditional imitation and the replicator rule, in the Moran and Fermi rules, the players also can, with a small probability, adopt the strategy of a less successful neighbor. Moran rule is inspired by the Moran process, which in biology is a prototypic dynamic update rule for asexual replication. At each time step, one individual is chosen for reproduction with a probability proportional to its payoff. An identical offspring is produced, which replaces another individual in the neighborhood. It can be regarded as a type of imitation, where the probability that player i will change the strategy to the strategy of her neighbor j is

$$p_{ij}^t \equiv P\{s_j^t \to s_i^{t+1}\} = \frac{W_j^t - \Psi}{\sum_{k \in N_i^*}(W_k^t - \Psi)} \tag{1.16}$$

where N_i^* is the neighborhood of player i including herself and Ψ is added to ensure that the probabilities are positive, since the payoffs can be negative. Therefore, the probability of imitating a neighbor is given by the relative success of that neighbor in the neighborhood. In the Fermi rule, based on the Fermi distribution function, the probability of imitating depends of the relative success of that neighbor compared to the focal player. Furthermore, it allows us to control the intensity of the influence of the payoff difference in the probability of imitation. Formally

$$p_{ij}^t \equiv P\{s_j^t \to s_i^{t+1}\} = \frac{1}{1 + e^{-\beta(W_j^t - W_i^t)}} \tag{1.17}$$

where β can be interpreted as an inverse temperature or noise and it tunes the intensity of selection.

To properly understand the influence of the spatial structure on the promotion of cooperation, apart from the update mechanism, we also need to consider different topologies. The main mechanism behind the promotion of the cooperation is cluster formation, which can be largely influenced by the network topology (especially the clustering coefficient) (Roca et al. 2009c). In the case of complex networks, where nodes with many connections appear, these nodes, called hubs, also play a key role in governing the emergence of cooperation (Gómez-Gardeñes et al. 2007; Santos et al. 2006b). Additionally the existence of hubs cannot be considered as the only reason for the promotion of cooperation. Other structural properties like node-node correlation should be taken into account in order to capture properly the mechanisms that help fixation of cooperation in real complex networks (Poncela et al. 2009; Poncela et al. 2010).

Very many models have explored analytically and by simulation the effects of topology and update mechanisms on cooperation (Szabó and Fáth 2007; Roca et al. 2009b). It turns out that the only rule that will significantly promotes cooperation in the spatial PD in a system without self interactions is unconditional imitation. The game in which more positive effects of the spatial structure are found is the Stag hunt. This effect is also robust against the update mechanism. Furthermore, it was found that the influence of the spatial structure on the promotion of cooperation is directly linked to the high clustering of the network.

Unfortunately, in spite of the large body of theoretical work devoted to this issue, it has not been possible to reach a general conclusion about whether the existence of structure in a population can promote cooperation. Since the survival of cooperative behavior depends so crucially on the details of the models (Roca et al. 2009b; Roca et al. 2009a), its applicability to real life situations is dubious, at best. Therefore, experimental work beyond the large body of results on the PD in unstructured populations (Ledyard 1995; Camerer 2003) is needed to ascertain both the relevance of the population structure and the types of dynamics that are actually at work in real situations. To progress towards answering these two questions, we have carried out a few experiments which were followed by theoretical analysis. These works are the core of this thesis.

In Chapter 2 we describe a laboratory experiment with human subjects playing a PD on a sizable network, with a setup as close as possible to the one studied by Nowak and May (1992). This experiment was intended to see whether cooperation emerges in a spatial setting and to analyze what kind of update mechanisms players actually use. Chapter 3 presents a theoretical study of the coexistence of the three strategies observed in the experiments reported in Chapter 2 by means of the replicator dynamics. In Chapter 4 we discuss the analysis of an experiment carried out on a smaller network by another group (Traulsen et al. 2010). Here we focus on the different update mechanisms and their influence on cooperation with and without spatial structure. Finally, in Chapter 5 we present the results of another experiment where players were not on a network, but rather they were playing in groups of different sizes, in which we intended to probe further the existence of moody conditional cooperation and the effect of the number of players involved. The last chapter presents the conclusions of our work and includes a preliminary comparison of our lattice experiment with that of Traulsen et al. (2010) and a larger, more recent one by Gracia-Lázaro et al. (2012). In the appendixes we include further details on the experiments, the manual and the technical instructions for the software developed, and specific details of different issues of the work which are too technical to include in the main text. At the end we present and the publications which resulted from the work described here.

2

Cooperation in a mesoscale experiment with human subjects

As established in the previous chapter, many models have explored analytically and by simulation the possibilities of network reciprocity on the promotion of cooperation. However, the results of these models largely depend on details such as the type of spatial structure or the evolutionary dynamics. In order to determine how population structure influences the promotion of cooperation and what type of dynamics players actually use, experimental work especially designed for structured population is needed.

To progress towards answering these two questions we have focused on the pioneering model studied by Nowak and May (1992). They simulated a set of agents located on a square lattice, playing a PD with their Moore neighborhood (i.e., playing a PD with each of their eight surrounding neighbors, but using the same strategy with all of them), and showed that when they imitated the behavior of the neighbor who obtained the largest payoff in the previous round (including themselves), cooperation could thrive even under very adverse choices of the payoffs. We set a laboratory experiment with human subjects, on a sizeble network, mimicking as close as possible the setup of Nowak and May's simulations. In this respect, it is important to note that in those simulations agents do not have memory and update their strategies with a specific, fixed rule, whereas we are implementing the same system with humans. It is clear that the rules used by humans are unknown a priori (they are not instructed to follow any particular rule). As a matter of fact, the goal of the experiment is precisely to unveil the way they behave.

Specifically, 169 volunteers were located on a (virtual) 13×13 square lattice with periodic boundary conditions, on which they were able to interact anonymously. At the time we carried out the experiment this was by far the largest experiment of this kind ever carried out. The organizational burden and the cost of the experiment increase geometrically with its size, so going to larger systems is increasingly higher. It is only recently that our record has been surpassed (Gracia-Lázaro et al. 2012), but at the time the only experimental work on this issue had been conducted on networks an order of magnitude smaller. In what follows we will review the most relevant ones.

Cassar (2007) studied an iterated Coordination game and a PD. She compared three structures: a local network (players were placed on a circle and interact with a few of their neighbors), a random network (players were randomly connected) and a small world network (built from the local networks by randomly rewiring some percentage of the connections). In the PD experiments, the percentage of cooperation started at around 50% and declined during the game. Therefore, there was no promotion of cooperation in any of the examined networks. Statistical analysis showed that individual decisions depend significantly on the number of cooperating neighbors in the previous round as well as on the focal player's own previous action. Kirchkamp and Nagel (2007) performed iterated PD experiments on two types of structure: circles and groups. The circle setting was the same as the local network in the experiments of Cassar (2007). On the other hand, in the group setting players were interacting with all members of their group and has no interactions with players of other groups. The experiment was repeated for two different groups sizes: small (5 partners in a group, as in the circle setting) and big (with groups sizes of 8 to 10 neighbors). As found by Cassar (2007), there was no promotion of cooperation in any of the settings. The initial level of cooperation was around 40% to 60% and then declined over time. The authors concluded that players do sometimes imitate their neighbors, but that they primary learn from their own actions. The only one experiment that has addressed the questions we are interested in here was carried out by Traulsen et al. (2010). In this experiment humans played an iterated PD on 4×4 lattice. Like in previous experiments, although the cooperation started at a high level (70%) it declined rapidly during the game. The authors found that players use the Fermi rule as an update mechanism. However, not all of the behavior can be explained with the Fermi rule. Sometimes the players would switch their action although they are surrounded only by players who use the same action. They refer to this as a spontaneous change of strategy and claim that it corresponds to mutations.

In all the experiments above the sizes of the network were very small (18 nodes the largest). However, the question of the size of the network is very important because the putative mechanism leading to the emergence of cooperation (Roca et al. 2009a; Langer et al. 2008) is the appearance of clusters of cooperators. Cooperator clustering can be difficult to observe in small systems, hence the necessity of studying larger ones and, in addition, for times as long as possible (Helbing and Yu 2010).

2.1 Experimental setup

In our experiment, volunteers played a 2×2 PD game with each of their eight neighbors (Moore neighborhood) taking only one action, either to cooperate (C) or to defect (D), the action being the same against all the opponents. The resulting payoff was calculated by adding all eight interaction payoffs. Payoffs of the PD game were set to be 7 cents of a euro for mutual cooperation, 10 cents for a defector facing a cooperator, and 0 cents for any player facing a defector (weak PD (Nowak and May 1992)). With this choice (a cooperator and a defector receive the same payoff against a defector) defection is not a risk dominant strategy, which enhances the possibility that cooperation emerges. In addition, to avoid framing effects (Levin and Gaeth 1988; Brañas-Garza 2007), the two actions were always referred to in terms of colors (blue for C and yellow for D), and the game was never referred to as PD in the material handed to the volunteers. This notwithstanding, players were properly informed of the consequences of choosing each action, and some examples were given to them in the introduction. After every round players were given the information of the actions taken by their neighbors and their corresponding payoffs in the form given in Fig. 2.1.

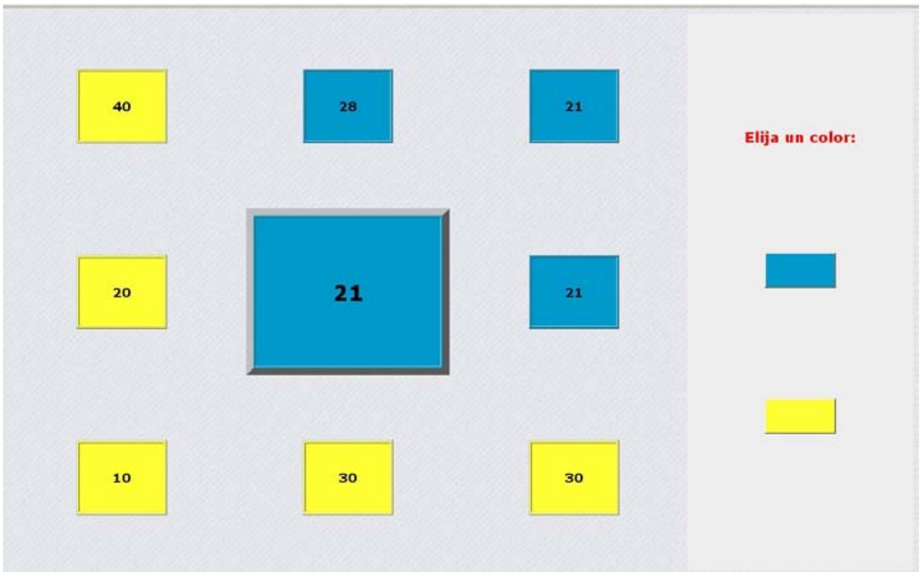


Figure 2.1: *Information given in the experimental setup.* After every round players saw the information in the screen as depicted here. Given are the strategies of the player's neighbors (color coded) and their earnings in the previous round (in cents of a euro). To the right the player had two clickable buttons with the two actions to choose from for the next round under the label "Elija un color:" ("Choose a color:").

The full experiment consisted of three parts: experiment 1, control, and experiment 2. In experiment 1 players remained at the same positions in the lattice with the same neighbors throughout the experiment. In the control part we removed the effect of the lattice by shuffling players every round. Finally, in experiment 2 players were again fixed on a lattice, albeit different from that of experiment 1. On the screen players saw the actions and payoffs of their neighbors from the previous round, who in the control part were different from their current neighbors with high probability. All three parts of the experiment were carried out in sequence with the same players. Players were also fully informed of the different setups they were going to run through. The number of rounds in each part was randomly chosen between 40 and 60 in order to avoid players knowing in advance when it was going to finish, resulting in 47, 60, and 58 rounds for experiment 1, control, and experiment 2, respectively.

2.2 Global cooperative behavior

We begin the presentation of the results of our experiment by discussing the first issue, namely the global cooperation level. Figure 2.2 represents the total percentage of cooperative actions in every round of the three parts of the experiment. Experiment 1 begins with a very large percentage of cooperation, above 50%, that rapidly decays to reach a more or less constant level after some 25 rounds. Experiment 2 exhibits the same behavior, but the initial cooperation level is much lower, a 32%, and the transient shorter. On the contrary, the control part shows a constant fraction of cooperative actions, fluctuating around 20%. This is a clear indication that players did realize that the fact that neighbors changed after every round made it hopeless to try to achieve a mutually profitable environment, which they did attempt to establish at the beginning of experiment 1 (particularly so) and experiment 2. On the other hand, after the initial transient, the amounts of cooperation observed in the two experiments and in the control part coincide approximately, showing that the existence of a fixed lattice structure has little influence on the players' asymptotic behavior.

Our conclusion that the lattice has little influence for the global cooperation level and our observed results are in good agreement with those reported by Traulsen et al. (2010), although in their case they also observe high initial cooperation levels in the well-mixed case, most likely because in their setup these players were beginning their participation without prior experience. We note also that their experiment is shorter in time than ours, with a duration comparable to the length of our transient (they do not observe a stationary state, as we do, as noted also in Helbing and Yu (2010)). In spite of that, it appears that their asymptotic value for cooperation is compatible with the 20% value we found. On the other hand, the differences between the results of experiments 1 and 2 cannot be attributed to the different distributions of players on the lattice: A learning process has occurred that has led players to use a better defined strategy in experiment 2. This is not only evident in the shorter transient period and the lower starting level of cooperation in experiment 2 compared to experiment 1, but it

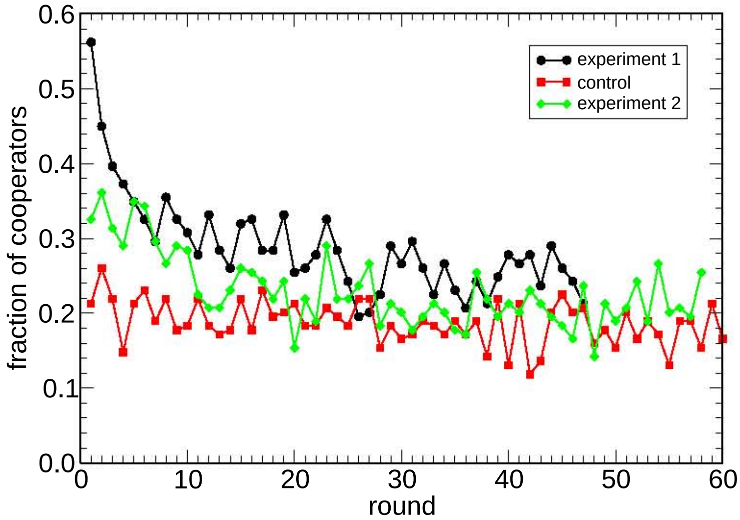


Figure 2.2: *The cooperation level declines to a low but non-zero level.* Fraction of cooperators in every round of the three parts of the experiment (in the first and the last ones players remain in the same node of the lattice along the whole experiment, whereas in the control part players are shuffled every round).

also shows up in many other features that we will be commenting on in the remaining of this article.

2.3 Testing the 'imitate-the-best' strategy

Our experiment has been set up to mimic Nowak and May's simulations as close as possible. As the system sizes considered in Nowak and May (1992) were larger than our experimental lattice, we have repeated their simulations on a 13×13 lattice with the payoffs of the experiment. We also used the same update rule, "imitate-the-best" —copying the action of the neighbor that performed the best provided it was better than their own—. The results show that the asymptotic level of cooperation is either 0 or a large value close to 1, depending on the initial condition, while an outcome with the level of cooperation observed in the experiment is never found. This suggests that either players do not update their actions with an imitate-the-best rule, or memory effects, absent in Nowak and May (1992), are important —or both. We will analyze the behavior of the players in terms of their previous actions and those of their neighbors in the next section. Presently, we will check to what extent imitation plays a role in our experiment. To that purpose we have computed the fraction of actions that can be interpreted as imitation of the best action in the neighborhood along the experiment, yielding the values 0.7149 for experiment 1 and 0.7687 for experiment 2. In spite of their being both above 70% one should bear in mind that there are only two actions to choose from and pure chance may be mistaken for imitation. To ascertain the statistical significance of these values we applied a non-parametric bootstrap (Efron and Tibshirani 1993) method, consisting of performing a thousand random shufflings of the positions of the players while keeping their sequences of actions during the experiments, and computing the corresponding fractions of imitation. This provides the empirical probability distributions of the null hypothesis "imitation is due to chance". The mean values of these distributions are 0.7145 ± 0.0014 for experiment 1 and 0.7678 ± 0.0013 for experiment 2, and values larger than the one we find can be obtained with probability $p = 0.425$ in experiment 1 and $p = 0.282$ in experiment 2 (see Appendix A, section A.2.2). This proves that the observed imitation is not significantly different from the apparent imitation yielded by pure chance. This result, which is consistent with the low level of cooperation observed (players using imitate-the-best should lead the system to higher cooperation) and with the responses to the questionnaires at the end of the experiment (no one claimed to have imitated the best neighbor), makes it plausible to conclude that imitate-the-best is not an appropriate explanation of players' behavior (although strictly speaking, this statistical analysis does not allow us to definitely rule out this strategy).

2.4 Analysis of players' strategies during the experiment

To make further progress towards clarifying the question of the dynamics of strategies, we considered as an alternative update strategy the possibility that players react to the number of cooperative neighbors ($k = 0, 1, \ldots, 8$) they observed in the previous round (henceforth *context*), i.e., we assume that they have one-step memory. This is a reasonable assumption in view that questionnaires suggest that players take into account what their neighbors do. Furthermore, Traulsen et al. (2010) briefly report that cooperative actions are more frequent in more cooperative environments. Therefore, we specifically computed from the experimental data the average frequency with which players cooperated, conditioned to both their previous action and their context, and made linear fits to these frequencies [Figures 2.3 (top) and 2.4 (top), and Table 2.1]. The first observation is that players' reactions to the context depend strongly on the past action of the focus player, something that to our knowledge has never been reported. The significance of this result can be assessed by comparing with the result obtained averaging over a thousand shufflings of the players in the lattice [Figures 2.3 (bottom) and 2.4 (bottom)], which show no dependence on the context. The parameters of the linear fits can be found in Table 2.1. The plots demonstrate that there is a strong dependence on the context for players that cooperated in the previous round (i.e., were in a "cooperative mood"), the cooperation probability increasing rapidly as a function of the number of cooperative neighbors in a way similar to the conditional cooperators found by Fischbacher et al. (2001). However, after having defected, players behave in a way that shares features of exploiting behavior, cooperating with equal or less

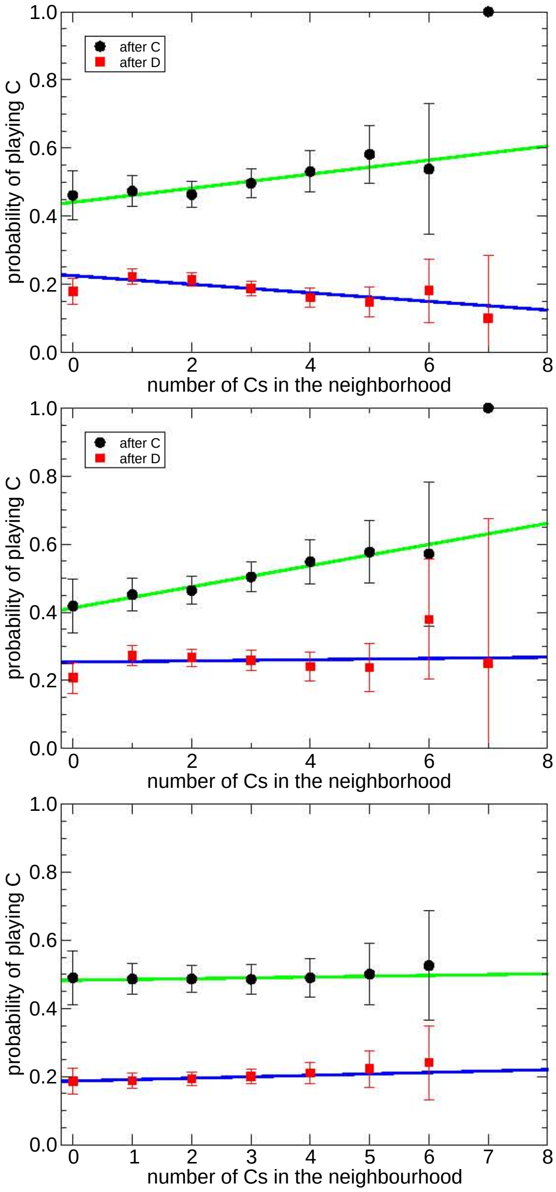


Figure 2.3: *Probabilities of cooperating after playing C or D, conditioned to the context for experiment 1*. Top panel shows results for all players, whereas the middle panel shows results for the group of players referred to as conditional cooperators. Bottom panel shows the probabilities of cooperating after playing C or D, conditioned to the context (number of cooperative neighbors in the previous round), averaged over 1000 random shufflings of players in the lattice.

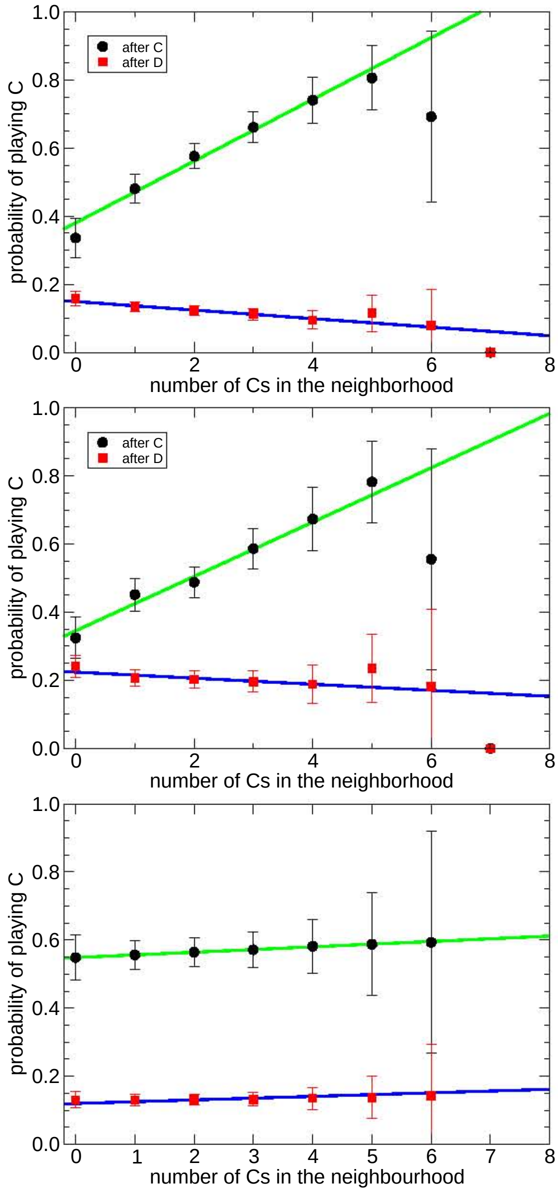


Figure 2.4: *Same as Figure 2.3 for experiment 2*

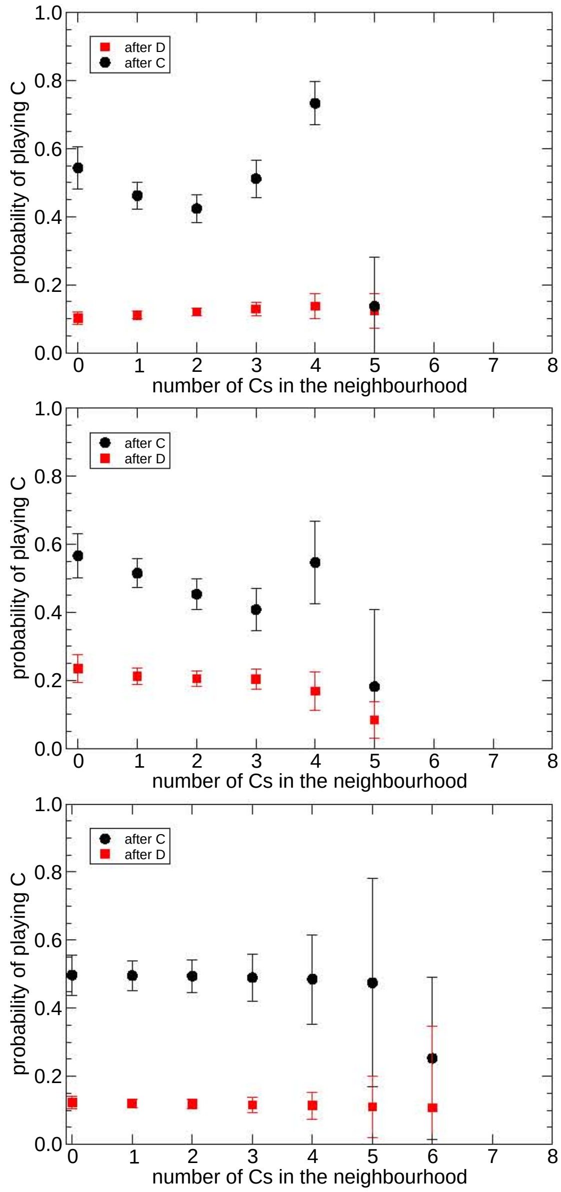


Figure 2.5: *Same as Figure 2.3 for control*

probability as the number of cooperators in their neighborhood increases. The very different behavior of players in the control experiment illustrates that conditional cooperation arises as a direct reciprocity effect —which is pointless if neighbors change every round. The conditioning of cooperation to the previous round is different in both experiments, which provides a strong indication that players learned to play as the experiment proceeded, moody conditional cooperation being more clearly observed in the plots corresponding to experiment 2. Finally, the bottom panels on Figures 2.3, 2.4 and 2.5 show the probabilities of cooperating after playing C or D, conditioned to the context, averaged over 1000 random shufflings of players in the lattice. The results show that there is no dependence on the context, proving that the dependence revealed in the top panels is statistically significant.

Table 2.1: **Linear fits to the probabilities of cooperating as a function of the context.**

type of data	$\alpha_{\rm C}$	$\beta_{\rm C}$	$\alpha_{\rm D}$	$\beta_{\rm D}$
exp. 1, all	0.021 ± 0.005	0.441 ± 0.015	-0.013 ± 0.005	0.225 ± 0.015
exp. 2, all	0.091 ± 0.009	0.381 ± 0.022	-0.013 ± 0.002	0.149 ± 0.005
exp. 1, c.c.	0.031 ± 0.003	0.413 ± 0.008	0.002 ± 0.007	0.254 ± 0.018
exp. 2, c.c.	0.080 ± 0.010	0.345 ± 0.022	-0.009 ± 0.004	0.224 ± 0.009

Fits are defined by $\Pr(C|X,k) = \alpha_{\rm X} k + \beta_{\rm X}$, where $\rm X = C, D$ is the player's action in the previous round and $k = 0, 1, \ldots, 8$ is the number of cooperators in the neighborhood in the previous round.

On the other hand, the differences observed in the fits of the two experiments provide another hint that players are using a better defined strategy in experiment 2, after having "learned" in the two previous phases of the experiment. Using these fits as a model (henceforth *homogeneous model),* we made simulations in a 13×13 lattice in which all players react according to these rules, with an initial condition similar to the one found in the experiments. This model is able to reproduce the observed asymptotic level of cooperation in both experiments, predicting an asymptotic value of 28% for experiment 1 and of 22% for experiment 2, but fails to reproduce other features. For instance, it leads to a histogram of total earnings much narrower than the experimental one, and the distribution of fractions of cooperative actions among players reveals that it does not capture a significant fraction of stubborn defectors and cooperators that appear in the experiment (see Figure 2.6).

We then tried to distinguish different kinds of behavior shown by players. First we found a sizeable number of *pure defectors*, as well as a few *pure cooperators*, in all three stages of the experiments —i.e., players who always defected/cooperated irrespective of the actions of their neighbors. Taking these individuals out, we still were able to classify the remaining players into three groups: *Mostly defectors* (people who defected more than 2/3 of the times in any context), *mostly cooperators* (cooperated more than 2/3 of the times in any context), and generalized conditional cooperators (players who seem to react to the context as before), which we hereafter refer to as *moody conditional cooperators* —indicating that their propensity to cooperate depends on their previous action, or "mood". Their amounts are listed in Table 2.3 and we have checked that this classification is consistent with the answers that players provided in their questionnaires. It is remarkable that the classification is very similar to the one reported by Fischbacher et al. (2001) in public goods experiments, and confirmed in subsequent papers (see, e.g., (Gächter 2007) for a review and (Ledyard 1995) for a general review about public goods experiments), even if they do not report the "moody" behavior of conditional cooperators. This is an important feature of their behavior because, as can be seen in top panels of Figures 2.3 and 2.4, the probability that a moody conditional cooperator cooperates after having defected in the previous round turns out to be slightly non-increasing as a function of the number of cooperators in the context.

It is worthwhile to compare the behavior of conditional cooperators in the two experiments [either top and middle panels of Figures 2.3 and 2.4] and in the control part [top and middle panels of Figure 2.5]. The different behavior that can be observed strongly suggests that this strategy arises as a result of direct reciprocity. Whereas in the two experiments conditional cooperators who cooperated in the previous action cooperate more the more neighbors cooperate, it is quite the opposite in the control experiment. Indeed, it makes no sense to reciprocate or retaliate in this control experiment because the recipients of your action are —with high probability— no more your previous opponents.

2.5 Cooperator clustering

Once we have a classification of the players, we are in a position to address another issue about the lack of global cooperative behavior, namely the assortment or clustering of cooperators. The low level of cooperation we observe is in agreement with the fact that cooperative players —i.e., players whose actions are always or almost always cooperative— do not cluster in space even if they are initially a majority, as in experiment 1. Interestingly though, the few cooperators in experiment 2 are somewhat clustered, and in both experiment 1 and 2, defectors show a slight anti-clustering trend: This can indeed be seen in Table 2.2, where we collect the average number of neighbors of the same type for the three types of players (pure and mostly cooperators, pure and mostly defectors, and conditional cooperators), as obtained from the experimental data. This average is computed, for each type of player, as the sum of pairs of neighbors of the given type divided by the number of players of that type. We resorted again to non-parametric bootstrapping to assign significance to those values, computing the average number of neighbors of the same type in a thousand random shufflings of players. The experimental values are always within the confidence interval of the null model, except for a few cases (in boldface in Table 2.2) that are particularly important because they suggest some cooperator clustering as well as some defector anti-clustering, precisely the cooperation fostering mechanism put forward by theoretical models. It would nevertheless be bold to speak about clustering of cooperators when the largest number of them we observe (that of experiment 2) is just nine.

Table 2.2: **Average number of neighbors of the same type.**

type	experiment 1			experiment 2		
	exper.	mean	SD	exper.	mean	SD
cooperator	0.0000	0.0946	0.2383	**1.3333**	0.3905	0.2740
cond. coop.	5.8560	5.9048	0.0819	4.1758	4.2855	0.1438
defector	**1.6585**	1.9163	0.2353	**2.9565**	3.2404	0.1823

The column *exper.* lists the average number of neighbors of the same type for the three types of players, computed, for each type of player, as the sum of pairs of neighbors of the given type divided by the number of players of that type. The columns *mean* and *SD* list the means and standard deviations of the values obtained in 1000 random shufflings of players.

2.6 Heterogeneous model

In order to assess the validity of our understanding of the players' behavior we designed a new model implementing heterogeneity by starting from the same amounts of each of the five types of players (the model is referred to as *heterogeneous model)*. In the simulations every player behaves according to her type, and for the generalized conditional cooperators we employed a model similar to the homogeneous one, but this time computing the average probabilities only for conditional cooperators [middle panels of Figures 2.3 and 2.4 and Table 2.1]. This heterogeneous model succeeds in reproducing even the features that the homogeneous model does not capture. To begin with, the global cooperation level is 28% for experiment 1 and 23% for experiment 2, in agreement with the experimental results. Furthermore, Figure 2.6(a) and (b) shows a comparison of the histogram of earnings, for all players aggregated and separated by types, as obtained from the two models (homogeneous and heterogeneous) and from the experiment. We can observe that experimental data are consistent with the simulations of the heterogeneous model, whereas the homogeneous model deviates from the experimental results (typically, as we already mentioned, it has a noticeably narrower distribution of earnings). This picture also shows that the distribution of earnings is the same for all kinds of players, clearly in the simulations but also in the experimental data, mainly in experiment 2. The slight advantage of defectors in experiment 1 is surely due to the longer cooperative transient. This advantage disappears in experiment 2, where players are supposed to have learned and to be using a more definite strategy. We note that the fact that payoffs are very similar for the different strategies supports their coexistence, as there is no real incentive (on average) to switch between them. Interestingly, a similar result was found in experiments on modified public goods games by Kurzban and Houser (2005). A further evidence in favor of the heterogeneous model is revealed by the histogram of cooperative actions occurred in both experiments [Figure 2.6(c) and (d)]. The homogeneous model shows a Gaussian-like peak, whereas the heterogeneous model shows a more widespread distribution, closer to the experimental one.

Table 2.3: *Evidence for heterogeneity in the behavior of the population.*

type of player	experiment 1	control	experiment 2
pure cooperators	1	1	6
mostly cooperators	2	2	3
conditional cooperators	125	92	91
mostly defectors	26	36	36
pure defectors	15	38	33

Frequency of the different types of players in the three parts of the experiment. Mostly defectors are people who defected more than 2/3 of the times in any context, mostly cooperators are those who cooperated more than 2/3 of the times in any context, and conditional cooperators follow the strategy described in the main text.

2.7 Alternative interpretations of players' strategies

The fact that Figures 2.3 and 2.4 reveal that the probability of cooperating after having defected in the previous round is both low and independent on the context, might suggest that the strategy actually employed by conditional cooperators is a version of GRIM. GRIM is a strategy of the so called "trigger" type, first introduced by Friedman (1971). This strategy amounts to cooperating until disappointment (by the lack of cooperation of the partners), and defecting from then on. Thus defined, GRIM plays an important role for proving theoretical results in game theory (see, e.g., (Hegselmann and Flache 2000; Buskens and Wessie 2000)). For our present purposes, let us note that if all or a majority of agents use this strategy, it is clear that as soon as one defects, a cascade of permanent retaliation is initiated until full defection dominates the system. This is the reason why in the famous experiments by Axelrod about the PD game GRIM did not perform very well (cf. (Axelrod 1984), where GRIM is referred to as FRIEDMAN). In our experiment we observe a background of cooperative actions near 20%, but perhaps players are using a weaker version of GRIM in which the final defection is 'noisy' in the same percentage. Alternatively, players could be progressively switching from an initial conditional cooperative strategy to a more defective strategy through some learning process (see, e.g., (Camerer 2003) for a review of the different

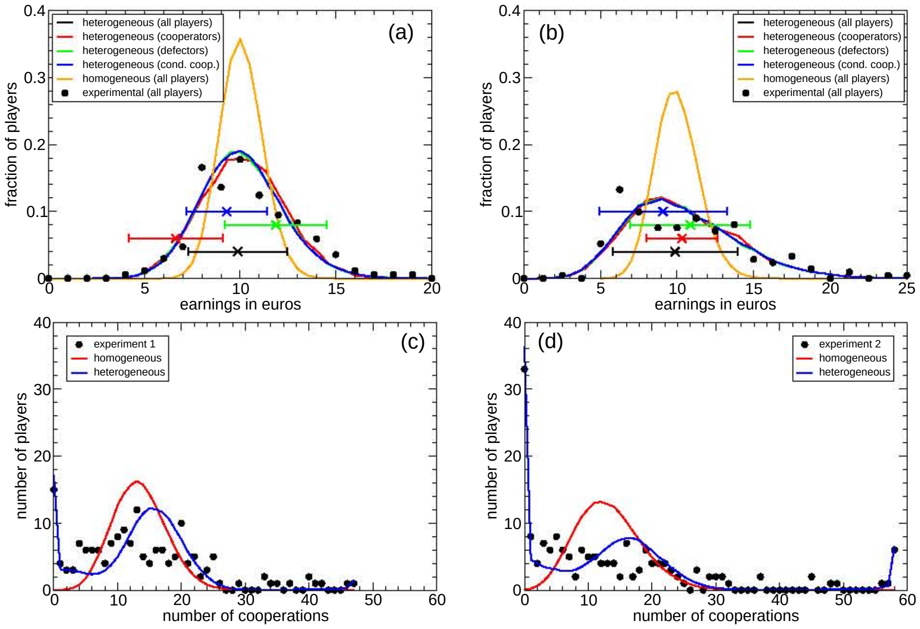


Figure 2.6: *The heterogeneous model reproduces the earning and cooperation histograms and supports the coexistence of different types of players.* Panels (a) and (b): Histograms of earnings in simulations of the heterogeneous model, for all players aggregated (black line, hidden by the blue line) as well as for the three basic types of players: pure and mostly cooperators (red line), pure and mostly defectors (green line), and conditional cooperators (blue line); histograms of earnings in simulations of the homogeneous model (orange line); and experimental histograms of earnings for all players aggregated (black dots). Results are presented for both experiment 1 (a) and experiment 2 (b). Simulations results are averages over 1000 runs. Crosses ($\times$) represent the mean earnings in the real experiments (their Y coordinate is arbitrary). Error bars span two standard deviations. Clearly, simulations of the homogeneous model do not fit the experimental data, thus supporting the introduction of the heterogeneous model. There is a reasonable consistency between experimental results and numerical simulations for the heterogeneous model, more so in experiment 2, where players are supposed to be playing with a better defined strategy. In experiment 1, the longer cooperative transient makes defection a more favorable strategy. The fact that the histograms for the different kinds of players are indistinguishable supports the coexistence of strategies, as there is no real incentive (on average) to switch from one strategy to any other. Panels (c) and (d): Number of players who cooperate a given number of rounds, both for experiment 1 (c) and experiment 2 (d). The experimental results are plotted together with the results of simulations with the homogeneous and the heterogeneous models, averaged over 1000 realizations. Once again, the homogeneous model is not able to reproduce the experimental results.

learning processes that could be at work). In both cases the result found in Figures 2.3, 2.4 and 2.5 for the probability of cooperating after having cooperated in the previous round would just be a consequence of the actions taken by these players during the transient, in the first rounds of both experiments, and the asymptotically surviving strategy would be noisy defection, regardless of the context.

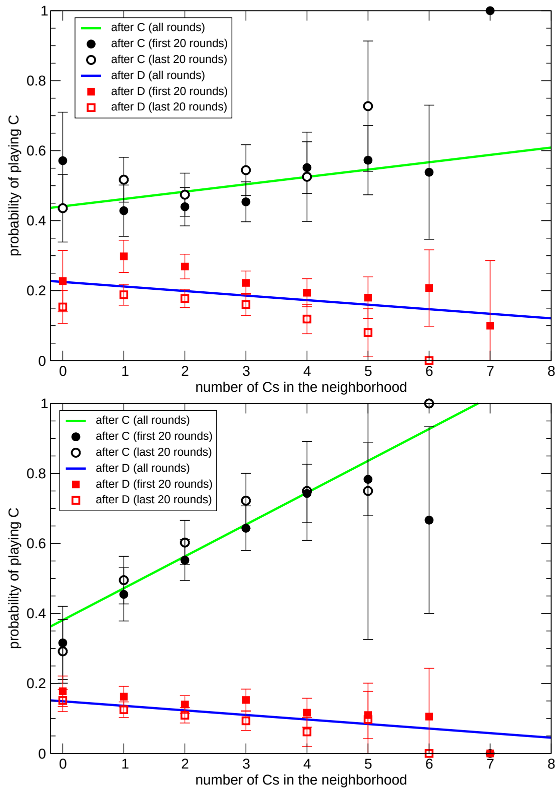


Figure 2.7: *Conditional cooperators' strategies (almost) do not change over time.* Same as the top panels of Figures 2.3 and 2.4. Straight lines are the fits appearing in Figures 2.3 and 2.4, whereas points are the strategies as obtained only from the first 20 rounds (full symbols) and only from the last 20 rounds (empty symbols). The strategies are statistically the same for experiment 2, and for the experiment 1 after having played C in the previous round. After having played D in the previous round in experiment 1, the probability of cooperating noticeably decreases over time down to a value compatible with that observed in experiment 2.

To test this alternative explanation, we have carried out an analysis of the conditional strategies at different times during the game. If any of these two strategies is at use, this analysis should reveal a change in the probabilities shown in Figures 2.3 and 2.4 over time; in particular, we should observe a decay of the probability of cooperating after having cooperated in the previous round. We do not have enough statistics to test the strategies every round of the game, but we can do it along two different intervals: in the first 20 rounds and in the last 20 rounds. Did the player use any GRIM-like strategy, either as such or through a learning process, the results of the analysis in these two periods should be different, as least as far as cooperative strategies are concerned. In Figure 2.7 we show the results of this analysis. We do not observe any significant change in the results for experiment 2, and for experiment 1 we only appreciate a small decay of the probability of cooperating after having defected. These results rule out the interpretation of players' strategies as GRIM or as 'learning-to-defect', with the exception, in this last case, of the small effect just pointed out. Our result is in agreement with recent experimental findings (Dal Bó and Frechette 2011) that in an infinitely repeated PD game GRIM explains some of the data, but its proportion is not statistically significant. It seems that during experiment 1 the probability that a player restores cooperation gets adjusted as time passes, decreasing towards values compatible with the stable value found in experiment 2. On the other hand, there is, of course, the difference in the cooperative strategy between both experiments, also attributable to some kind of learning. Particularly interesting is the stability of the strategies along experiment 2, consistent with the idea that players had a more precise idea of how to play in this second experiment than they had in the first one.

2.8 Discussion

The large size of our experimental setup and the data analysis presented above allow us to contribute to the two questions we wanted to address. First, we have observed that the existence of a lattice giving structure to a population playing PD does not lead to an increase of the cooperation level, even if as in our case the PD is weak. Thus, subjects behave as if they were playing a repeated public goods game, the fact that the game in which a player is involved overlaps with those of their neighbors having very little influence on the observed asymptotic level of cooperation. Second, regarding the manner in which people update their strategies, we have not found evidence in favor of imitate-the-best behavior, in agreement with the analysis in Kirchkamp and Nagel (2007, Traulsen et al. (2010). These two observations imply in turn that the model simulated in Nowak and May (1992) does not describe our experiment with human subjects —albeit it may of course be applicable in many other instances such as, e.g., experiments with bacteria. We then analyzed the way subjects behaved by considering that they might be influenced by the previous actions of their neighbors. This analysis has allowed us to make some progress in understanding human behavior, reaching two important conclusions about individual learning models. The first one is that there is a large degree of heterogeneity, with an important fraction (25–45%) of players sticking to a strategy of (almost) always defect or cooperate. This is a crucial observation because the experimental results are not recovered unless those individuals are included in the modeling. The analysis of the total earnings of players also suggests that this heterogeneity can be evolutionarily stable, in the sense that all strategies are (on average) equally profitable, as observed also in Kurzban and Houser (2005) (some theoretical support for the evolutionary stability of a simplified model of conditional cooperation in the presence of social norms has been already provided (Spichtig and Traxler 2009)). The second conclusion is that the rest of the players are well described as moody conditional cooperators, i.e., players whose probability to choose one action depends on the amount of cooperation they observe in the previous round and their own previous action. Our clearest results, those of experiment 2, show that players have a high chance to continue cooperation (larger than 50%) if 3 or more neighbors cooperated, whereas if they had defected in the previous round, their chances to cooperate in the current one are small and slightly decreasing with the number of cooperating neighbors. This is consistent with an exploitation strategy which tries to incentive cooperation in low cooperative environments and also with a mutualistic strategy aiming at achieving better global results, something that many players claim to have done in their responses to the questionnaire. Indeed, the small resumption of cooperation at the beginning of experiment 2 as compared to the lack of it in the control indicates that a number of players hope they can restart cooperation for either of those two reasons. Our observation that the probability to cooperate depends on the context agrees with the results in Traulsen et al. (2010), and improves them by identifying that this probability depends in turn on the focal individual's previous action. In addition, our values for the probabilities are also consistent with their observation of high levels of "mutation", albeit our results provide a more intentional interpretation of these probabilities. A more detailed comparison between these two experiments as well as some more recent ones, will be presented in Chapter 6.

The results of this experiment have implications that go beyond the specific case study of PD on networks. Thus, the dependence on the player's own previous action we have found may be relevant to deepen our understanding of the conditional cooperation observed in public goods games (Fischbacher et al. 2001; Gächter 2007). In addition, we have proposed a model that, in spite of its simplified description of heterogeneity, provides a more thorough picture of the way human subjects might behave in these experiments, as we show that apparent mutation can be also understood (at least partly) as conscious changes of behavior arising from cooperative or exploiting strategies. Indeed, for the first time to the best of our knowledge, a model is able to reproduce the observed features in the experiment, from the decline of cooperation through the earnings distributions to the coexistence of strategies. In this regard, it is worth noting that recent experiments by Fischbacher and Gächter (2010) led to an explanation of the decline of cooperation in public goods games in which heterogeneity seemed to matter only at the end of the experiment. This is similar to what we have observed, in so far as our homogeneous model could also explain how cooperation evolved in time, but other features crucially required the introduction of heterogeneity. On the other hand, our observations are not consistent with a vast majority of the theoretical models of evolutionary games on graphs studied and simulated so far (Szabó and Fáth 2007; Roca et al. 2009b). Our experiment should therefore be a reference for future, more accurate modeling of these important social systems, as they strongly indicate that heterogeneity, that only recently has been considered in theoretical models (Moyano and Sánchez 2009; Szabó et al. 2009; Szolnoki and Perc 2008), is a key ingredient to understand human behavior. This is crucial for the design of mechanisms that promote or at least support cooperation, one of the goals of this line of research. In this respect, our work points to avoiding early disappointment of the agents that leads them to a "defective mood" as an important aspect to act upon. Finally, the issue of finding an evolutionary explanation of this coexistence of strategies is a challenge which should also be addressed to understand human cooperative behavior.

3

Coexistence of cooperators, defectors and moody conditional cooperators

The analysis presented in Chapter 2 suggests an alternative way to understand the experimental observations by building upon the idea of reciprocity (Trivers 1971), i.e., the fact that individuals behave depending on the actions of their partners in the past. In iterated two-player games, this idea has been studied through the concept of reactive strategies (Nowak and Sigmund 1989a; Nowak and Sigmund 1989b; Nowak and Sigmund 1990; Nowak and Sigmund 1992) (see Sigmund (2010) for a comprehensive summary on this matter), the most famous of which is Tit-For-Tat (Axelrod and Hamilton 1981). Reactive strategies generalize this idea by considering that players choose their action among the available ones with probabilities that depend on the opponent's previous action. For the simple case of two strategies (say C and D), players choose C with probability p following a C from their partner and with probability q after a D from their partner. Subsequently this idea was further developed by considering memory-one reactive strategies (Nowak et al. 1995; Sigmund 2010), in which the probabilities depend on the previous action of both the focal player and her opponent —i.e., the focal player would choose C with some probability following a (C,C) outcome, with some other following (C,D) and so on.

In iterated multiplayer games, such as public goods games or multiplayer Prisoner's Dilemmas (IMPD), reciprocity arises in the form of conditional cooperation (Fischbacher et al. 2001; Gächter 2007): individuals are willing to contribute more to a public good the more others contribute. Conditional cooperation has been observed a number of times in public goods experiments (Croson 2006; Fischbacher and Gächter

2010), often along with a large percentage of free-riders. The experiment by Traulsen et al. (2010) showed also evidence for such a behavior in an spatial setup. In Chapter 2, we extended this idea to include the dependence of the focal player's previous action, introducing the so-called moody conditional cooperation (cf. Figures 2.3 and 2.4). In this strategy, players are more prone to cooperate after having cooperated than after having defected, and in the first case they are more cooperative the more cooperative neighbors they have. This behavior has not been reported before in spatial games and appears to be a natural extension of the reactive strategy idea to multiplayer games (among the very many other extensions one can conceive). On the other hand, and from an economic viewpoint, which is an important part of the analysis of human behavior, this type of strategy update scheme responds to the often raised questions on payoff-based rules. In economic interactions it is usually the case that agents perceive the others' actions but not how much do they benefit from them, and therefore the use of action updates depending, e.g., on the payoff differences, may be questionable. This seems to be the case even if this information is explicitly supplied to the players (see Chapter 2).

Compared to the other two experiments (Kirchkamp and Nagel 2007; Traulsen et al. 2010), we have a new feature in our conclusions, namely the heterogeneity of the population: aside from the already mentioned moody conditional cooperators, there was a large minority of defectors, i.e., players that defected all or almost all the time, and a few cooperators, who cooperated practically all rounds. This heterogeneity, also found to be very important in public goods experiments (Fischbacher and Gächter 2010) had also been observed in four-player experiments by Kurzban and Houser (2005), who reported that their subjects could be roughly classified in three main types, including defectors, cooperators and conditional cooperators (called reciprocators in the original work), albeit they did not check for dependence on the past actions of the focal players either. In our experiment as well as in Kurzban and Houser (2005) the payoffs obtained by every type of player were more or less the same, thus suggesting that the population in the lattice experiment might be at an evolutionary equilibrium.

In this chapter we address the question of the existence and stability of such a heterogeneous or mixed equilibrium in the multiplayer iterated Prisoner's Dilemma. It is important to understand that we are not addressing the issue of the evolutionary explanation of moody conditional cooperation. This is a very interesting but also very difficult task, and in fact we do not even have an intuition as to how can one address this problem in a tractable manner. Our goal is then to understand whether or not the coexistence of moody conditional cooperators, defectors, and a small percentage of cooperators, as observed in the experiment, is theoretically possible. In so doing, we will shed light on experimental and theoretical issues at the same time. On the experimental side, our results show that there is coexistence for groups of 2 or 3 players for parameters reasonably close to those found in the experiment, but not for larger groups. As we will see in the discussion section, this prediction has important consequences related to the adequacy of replicator dynamics to describe the experimental result or to the cognitive capabilities of human subjects in dealing with large groups. We will also discuss there the ways in which our theoretical approach and the experiment may differ, something that can also have implications of its own. On the theoretical side, we present an analysis of a population of players interacting through a multiplayer Prisoner's Dilemma including strategies that generalize the ideas behind reactive strategies, as mentioned above. To our knowledge, this has not been carried out before, at least to the extent we are doing it here, in which we are able to show how this coexistence depends on the size of the groups considered. We believe that the approach we are presenting may be useful for other researchers working on related problems.

With the above goals in mind, we introduce below a model in which populations consisting of the three types of individuals discussed above, namely cooperators, defectors, and moody conditional cooperators, play a multiplayer iterated Prisoner's Dilemma with populations evolving according to the replicator dynamics. We have considered different group sizes, from $n = 2$ through $n = 5$ players, a size whose outcome is well described by the limit $n \to \infty$, which we analyze separately.

3.1 Game, strategies and payoffs

Let us consider a well-mixed population of players who interact via IMPDs. In these games, players interact in groups of n players. Every round each player adopts an action, either cooperate (C) or defect (D), and receives a payoff from every other player in the group according to a standard prisoner's dilemma payoff matrix (a cooperator receives R from another cooperator and S from a defector; a defector receives T from a cooperator and P from another defector; payoffs satisfy $T > R > P > S$). We note that this is a generalized version of a public goods game: In the latter, if there are k cooperators, a defector receives bk whereas a cooperator receives $b'(k-1) - c$ ($b' = b$ in a standard public goods game). In an multiplayer PD, a defector receives $(T - P)k + P(n - 1)$ whereas a cooperator receives $(R - S)(k - 1) + S(n - 1)$, and hence choosing $b = T - P$, $b' = R - S$ and $c = (P - S)(n - 1)$ the IMPD becomes a generalized public goods game. Notice an important difference with respect to the standard public goods game: in this generalized version ($b \neq b'$) the difference between the payoff received by a cooperator and a defector depends on the number of cooperators. Only when $T + S = R + P$ the standard public goods game is recovered.

Inspired by the experimental results from Chapter 2 but keeping at the same time as few parameters as possible, we will classify players' strategies into the three stereotypical behaviors that mimic those found in the experiment: mostly cooperators, who cooperate with probability p (assumed relatively close to one) and defect with probability $1 - p$; mostly defectors, who cooperate with probability $1 - p'$ and defect with probability p' (for simplicity we will assume $p' = p$); and moody conditional cooperators, who play depending on theirs and their opponents' actions in the previous round. Specifically, if they defected in the previous round they will cooperate with probability q, whereas if they cooperated in the previous round they will cooperate again with a probability

$$p_C(x) = (1 - x)p_0 + xp_1 \tag{3.1}$$

where x is the fraction of cooperative actions among the opponents in the previous round, and $p_0 < p_1$.

To complete the definition of the model, we need to specify how the populations of the different strategies are going to evolve in time. Players interact infinitely often in an IMPD, so payoffs both increase in time and depend on the whole history of play. It thus make sense to use the (time) average payoffs to study the evolution of the game in terms of the abundance of the three strategies considered. As these strategies are defined depending on players' actions in the round immediately before, a multiplayer game with n players and given populations of each type of player can be described as a finite state Markov chain whose states are defined by the actions taken by the n players. Of course the chain is different for different compositions of strategies in the group. In any case, given that all outcomes have non-zero probability, the chain is ergodic and therefore there is a well defined steady state (Karlin and Taylor 1975). Average payoffs are readily obtained once the probability vector in the steady state is known, and subsequent evolution is described through imitation via replicator dynamics (Hofbauer and Sigmund 1998).

3.2 Two-persons game (iterated PD)

3.2.1 General scheme of the approach

In the case $n = 2$ the states of the Markov chain are described as CC, CD, DC, and DD, where the first action is the focal player's and the second one is the opponent's. The transition probability matrix will be denoted as

$$M = \begin{array}{c} \\ \text{CC} \\ \text{CD} \\ \text{DC} \\ \text{DD} \end{array} \begin{array}{c} \begin{array}{cccc} \text{CC} & \text{CD} & \text{DC} & \text{DD} \end{array} \\ \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{pmatrix} \end{array}, \tag{3.2}$$

where m_{ij} gives the probability that players who played i in the previous round play j in the present round ($i, j \in$ {CC, CD, DC, DD}). The matrix M will of course depend on the nature of the two players involved, so there will be nine different matrices. Denoting 'mostly cooperators' by C, 'mostly defectors' by D and 'moody conditional cooperators' by X, the six combinations are CC, CD, CX, DD, DX, XX. As we stated above, the Markov chains so defined are always ergodic; consequently, the corresponding stationary probability vector, which we will term $\pi = (\pi_{\text{CC}}, \pi_{\text{CD}}, \pi_{\text{DC}}, \pi_{\text{DD}})$, is obtained by solving the equation $\pi = \pi M$ (Karlin and Taylor 1975). Note that there is such a stationary probability distribution π for each of the six combinations of two players, as we will see below. Now, once the probability distribution is known, the payoff matrix $W = (w_{ij})$, providing the average payoff that a player of type i gets when confronted to a player of type j ($i, j \in$ {C, D, X}) in an IMPD (in this Section,

$n = 2$, an iterated PD) can be computed as

$$w_{ij} = R\pi_{\text{CC}} + S\pi_{\text{CD}} + T\pi_{\text{DC}} + P\pi_{\text{DD}}. \tag{3.3}$$

These payoffs can then be used in the replicator dynamics to finally find the evolution of the three strategy population.

3.2.2 Payoff computation

Of the six combinations of players, three yield a trivial stationary vector π because they do not depend on the previous actions, namely those which do not involve the strategy X. The corresponding payoffs are therefore straightforward to compute, and we have (recall the focal player is denoted by the first subindex):

$$w_{\text{CC}} = p^2R + p(1-p)S + (1-p)pT + (1-p)^2P, \tag{3.4}$$
$$w_{\text{CD}} = p(1-p)R + p^2S + (1-p)^2T + (1-p)pP, \tag{3.5}$$
$$w_{\text{DC}} = (1-p)pR + (1-p)^2S + p^2T + p(1-p)P, \tag{3.6}$$
$$w_{\text{DD}} = (1-p)^2R + (1-p)pS + p(1-p)T + p^2P. \tag{3.7}$$

The payoffs for the cases where the moody conditional cooperators, X, play, require computing the corresponding stationary probability. Let us begin with the Markov matrix (3.2) for a mostly cooperator (C) and a conditional cooperator (X), given by

$$M = \begin{pmatrix} pp_1 & p\,(1-p_1) & (1-p)p_1 & (1-p)\,(1-p_1) \\ pq & p\,(1-q) & (1-p)q & (1-p)\,(1-q) \\ pp_0 & p\,(1-p_0) & (1-p)p_0 & (1-p)\,(1-p_0) \\ pq & p\,(1-q) & (1-p)q & (1-p)\,(1-q) \end{pmatrix}, \tag{3.8}$$

from which the stationary probability vector is given by

$$\pi = \frac{\big(pq, p[1-p_C(p)], (1-p)q, (1-p)[1-p_C(p)]\big)}{1+q-p_C(p)}, \tag{3.9}$$

where $p_C(x)$ is given by (3.1) (notice that it represents the average probability for a conditional cooperator to cooperate, given that she cooperated in the previous round, whereas her mostly cooperator opponent cooperates with probability p). Therefore, inserting (3.9) in (3.3) and having in mind who the focal player is, we arrive at

$$\begin{aligned} w_{\text{CX}} =& [1+q-p_C(p)]^{-1}\Big\{pqR + p[1-p_C(p)]S + (1-p)qT \\ &+ (1-p)[1-p_C(p)]P\Big\} \end{aligned} \tag{3.10}$$

$$\begin{aligned} w_{\text{XC}} =& [1+q-p_C(p)]^{-1}\Big\{pqR + (1-p)qS + p[1-p_C(p)]T \\ &+ (1-p)[1-p_C(p)]P\Big\}. \end{aligned} \tag{3.11}$$

The case for a mostly defector facing a moody conditional cooperator can be obtained immediately by realizing that the defector behaves as a mostly cooperator whose probability of cooperating is $1-p$ instead of p, hence we find trivially

$$\begin{aligned} w_{\rm DX} =&[1+q-p_C(1-p)]^{-1}\Big\{(1-p)qR+pqT \\ &+(1-p)[1-p_C(1-p)]S+p[1-p_C(1-p)]P\Big\}, \end{aligned} \tag{3.12}$$

$$\begin{aligned} w_{\rm XD} =&[1+q-p_C(1-p)]^{-1}\Big\{(1-p)qR+pqS \\ &+(1-p)[1-p_C(1-p)]T+p[1-p_C(1-p)]P\Big\}. \end{aligned} \tag{3.13}$$

Finally, if two conditional cooperators confront each other, the Markov matrix becomes

$$M=\begin{pmatrix} p_1^2 & p_1\,(1-p_1) & (1-p_1)\,p_1 & (1-p_1)^{\,2} \\ p_0q & p_0\,(1-q) & (1-p_0)\,q & (1-p_0)\,(1-q) \\ qp_0 & q\,(1-p_0) & (1-q)\,p_0 & (1-q)\,(1-p_0) \\ q^2 & q\,(1-q) & (1-q)\,q & (1-q)^{\,2} \end{pmatrix}, \tag{3.14}$$

and has a stationary vector π which, up to normalization, is proportional to a vector α with components

$$\begin{aligned} \alpha_{\rm CC} &= q^2(1+p_0-q), \\ \alpha_{\rm CD} &= q(1-p_1)(1+p_1-q), \\ \alpha_{\rm DC} &= q(1-p_1)(1+p_1-q), \\ \alpha_{\rm DD} &= (1-p_1^2)(1-p_0-q)+2qp_0(1-p_1). \end{aligned} \tag{3.15}$$

From this result one can compute $w_{\rm XX}$ as in the other eight cases. With the payoffs we have computed, we are now in a position to proceed to the dynamical study.

3.2.3 Replicator dynamics

Denoting $x=(x_{\rm C},x_{\rm D},x_{\rm X})$ (with $x_{\rm C}+x_{\rm D}+x_{\rm X}=1$) the vector with the population fractions of the three types of players, the dynamics of x_i is described by the replicator equation

$$\dot{x_i} = x_i[(Wx)_i - x\cdot Wx], \tag{3.16}$$

where W is the payoff matrix obtained above.

In order to use this dynamics in connection with the experiment from Chapter 2, we need to recall the payoffs used in that work, namely $T=10$, $R=7$, $P=S=0$ [i.e., a weak prisoner's dilemma as in Nowak and May (1992)]. Two consecutive experiments were carried out, leading to two different sets of parameters for the behavior of the players. Figure 3.1 shows the dynamics resulting for both sets of parameters, whose specific values are listed in the caption. As we may see, there are no interior points, which would indicate equilibria in which the three strategies coexist, as observed in the experiment. The only equilibria we find for these parameters are in the corners of

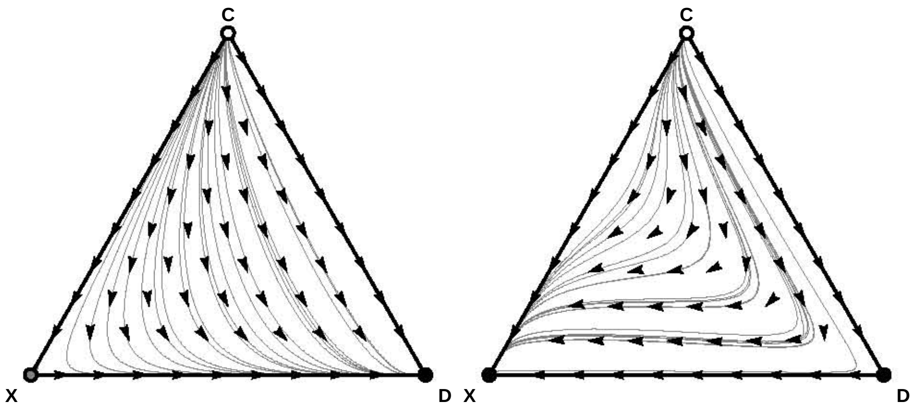


Figure 3.1: *There is no coexistence for the parameters of the experiment.* Phase portraits of the replicator dynamics for 2-players IMPD games with three strategies (C, D, and X) for the parameters inferred from experiment 1 (left; $p = 0.83$, $q = 0.26$, $p_0 = 0.44$, $p_1 = 0.60$) and experiment 2 (right; $p = 0.83$, $q = 0.21$, $p_0 = 0.34$, $p_1 = 0.98$). Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black).

the simplex, C being always a repeller, D an attractor and X being a saddle point or an attractor depending on the parameters. In the case where D and X are both attractors it is X that has the largest basin of attraction (almost the entire simplex), Therefore, the results for this model do not match what is observed in the experiment. However, it is important to keep in mind that in the experiment players played with their eight neighbors, this being the reason why we will later address the dynamics of IMPDs with larger groups.

Notwithstanding this first result, as we will now see it is very interesting to dwell into the $n = 2$ case in more detail. For the purpose of illustrating our results, let us choose the behavioral parameters to be $p = 0.83$, $q = 0.20$, $p_0 = 0.40$, and $p_1 = 0.80$, which are values we could consider representative of both experiments. Inserting these parameters into the calculations above, we find that the payoff matrix is given by

$$\begin{pmatrix} 0 & -0.3366 & 0.4367 \\ 1.6434 & 0 & -0.1800 \\ 1.0026 & -0.0526 & 0 \end{pmatrix}. \tag{3.17}$$

This type of matrix belongs to a class of games studied by Zeeman (1980). He analyzed the evolutionary dynamics of three strategies games. Apart from the well known rock-paper-scissors (Hofbauer and Sigmund 1998) he identified a game with the canonical payoff matrix for the strategies C, D and X, given by

$$\begin{pmatrix} 0 & -a_2 & b_1 \\ b_2 & 0 & -a_3 \\ a_1 & -b_3 & 0 \end{pmatrix}, \tag{3.18}$$

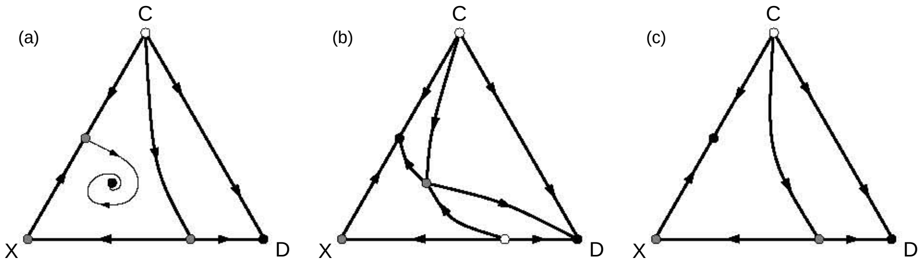


Figure 3.2: *Three different phase maps that can emerge for the Zeeman game*. There may (panels (a) and (b)) or may not be (panel (c)) an interior point, and it may be an attractor (panel (a)) or a saddle point (panel (b)). Circles mark the rest points and arrows indicate the direction of the flux. White circle denote unstable rest points, gray circles denote saddle points, and black circles denote stable points.

where all coefficients are positive. Any 3×3 payoff matrix can be transformed into a zero diagonal one because the replicator equation remains invariant if the same constant is subtracted from every element of one of its columns (Hofbauer and Sigmund 1998). The coefficients of the payoff matrix (3.18) represent the payoff an invader gets when it invades a homogeneous population. Thus a D or X individual invading a homogeneous C population will get b_2 or a_1, respectively. As both are positive a homogeneous C population is unstable. Similarly a C or X individual invading a homogeneous D population will get $-a_2$ or $-b_3$, respectively. Therefore a homogeneous D population is uninvadable (hence stable). As for a C or a D individual invading a homogeneous X population, it will obtain b_1 or $-a_3$, respectively. It is therefore a saddle point because it cannot be invaded by D individuals but it can be invaded by C individuals.

This simple analysis fixes the flux of the dynamics at the boundary of the simplex (Figure 3.2). It also implies the existence of two rest points on the boundary of the simplex: one on the D–X edge and another one on the C–X edge. These points are given by

$$\left(0, \frac{a_3}{a_3 + b_3}, \frac{b_3}{a_3 + b_3}\right), \qquad \left(\frac{b_1}{a_1 + b_1}, 0, \frac{a_1}{a_1 + b_1}\right). \tag{3.19}$$

Besides, an interior rest point $(y_\text{C}, y_\text{D}, y_\text{X})/(y_\text{C} + y_\text{D} + y_\text{X})$, with coordinates

$$\begin{aligned} y_\text{C} &= b_3(a_3 + b_1) - a_2 a_3, \\ y_\text{D} &= b_1 b_2 - a_1(b_1 + a_3), \\ y_\text{X} &= a_1 a_2 + b_2 b_3 - a_2 b_2, \end{aligned} \tag{3.20}$$

appears provided all three components have the same sign (Figure 3.2(a)). The y_D component is proportional to the difference between the payoff of the population at the C–X mixed equilibrium and the payoff of a D invader. When it is negative the C–X rest point becomes a saddle and the interior point is an attractor (the situation depicted in Figure 3.2(a)). When it is positive a D individual cannot invade the C–X equilibrium, which then becomes an attractor and the interior point becomes a repeller (this is illustrated in Figure 3.2(b)). If no interior point exists the behavior will be as plotted in Figure 3.2(c) (Zeeman 1980). In the region of parameters near those that can be inferred from the experiments from Chapter 2 the game behaves as in the first case (Figure 3.2(a)), known as a Zeeman game. This game has five rest points: an unstable one at the C corner, a stable one at the D corner, a saddle point at the X corner, and two mixed equilibria on the C–X and on the D–X edges of the simplex. Besides, under certain constraints (c.f. (3.20)) there is also an interior point.

Turning now to our example matrix (3.17), its non-trivial rest points turn out to be $(0, 0.7739, 0.2261)$, $(0.3034, 0, 0.6966)$, and $(0.1093, 0.3876, 0.5031)$. The stability of these mixed, interior equilibria depends on the parameters. For the present case, the situation is similar to that shown in Figure 3.2(a). Thus the evolution of this system is governed by the presence of two attractors: the interior point and the D corner, each with a certain basin of attraction. A key feature of the class of problems we are considering is that the precise location of the interior rest point is very sensitive to the values of the parameters. Figures 3.3–3.6 illustrate what happens to it when each of the four probabilities that define the strategies are changed around the values given above. Generally speaking, the figures show that the interior point approaches either one of the rest points on the edges C–X and D–X, while these in turn move along their edges. The specific details depend on the parameter one is considering as can be seen from the plots. We have also found that larger changes in the parameters can make the interior point coalesce with the mixed equilibrium on the C–X edge —thus transforming the dynamics into the one sketched in Figure 3.2(c)— or even change the Zeeman structure of the payoff matrix yielding different stable equilibria (generally at the corners). Notice that —particularly so in experiment 2— the values of the parameters are not far from those producing the plots of Figures 3.3–3.6. This indicates that, while we would not expect a two-person theory to describe quantitatively the experiments, the existence of an interior point with the same kind of mixed population as observed is possible with minor modifications of the parameters.

3.3 Games involving more than two players

Having discussed in depth the replicator dynamics for the IPD with mostly cooperators, mostly defectors and moody conditional cooperators, with the result that an interior point with a sizable basin of attraction exists for a wide range of parameters, we now increase the number of players to check whether the theory is a valid description of the experimental results. The mathematical approach for the case when more that two players are involved is similar to that for two players, only computationally more involved. The Markov transition matrix (3.2) now describes a chain containing 2^n states, n being the number of players. These are described as all combinations of C or D actions adopted by each of the n interacting agents. On the other hand, there will be $(n+2)(n+1)/2$ such matrices displaying all possible combinations of the three strategies (C, D, X). Obtaining the expressions for them is of course straightforward,

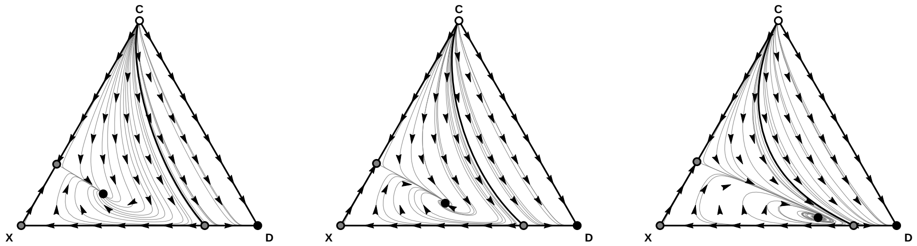


Figure 3.3: *Phase portraits for 2-players for different values of p.* The replicator dynamics for IMPD games with three strategies (C, D, and X) for the following values of p: 0.80 (left), 0.83 (middle), and 0.90 (right). Other parameters: $q = 0.2$, $p_0 = 0.4$, $p_1 = 0.8$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In all three cases the inner point as well as the $x_D = 1$ point are the only attractors of the system.

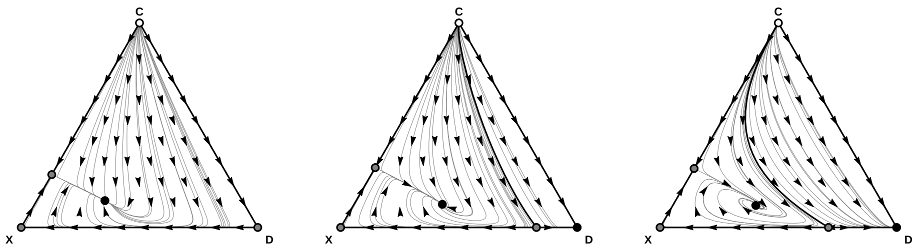


Figure 3.4: *Phase portraits for 2-players for different values of q* The replicator dynamics for IMPD games with three strategies (C, D, and X) for the following values of q: 0.10 (left), 0.15 (middle), and 0.30 (right). Other parameters: $p = 0.83$, $p_0 = 0.4$, $p_1 = 0.8$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In the last two cases the inner point as well as the $x_D = 1$ point are the only attractors of the system. In the latter case the point $x_D = 1$ has merged with the saddle in the edge $x_C = 0$ becoming a saddle point. Correspondingly, the basin of attraction of $x_D = 1$ has disappeared.

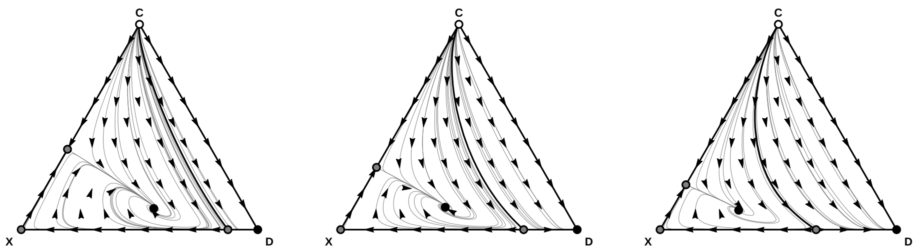


Figure 3.5: *Phase portraits for 2-players for different values of p_0* The replicator dynamics for IMPD games with three strategies (C, D, and X) for the following values of p_0: 0.20 (left), 0.40 (middle), and 0.50 (right). Other parameters: $p = 0.83$, $q = 0.2$, $p_1 = 0.8$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In all three cases the inner point as well as the $x_D = 1$ point are the only attractors of the system.

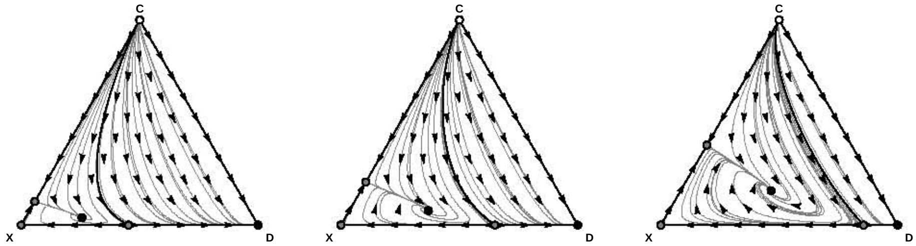


Figure 3.6: *Phase portraits for 2-players for different values of* p_1. The replicator dynamics for IMPD games with three strategies (C, D, and X) for different values of p_1: 0.70 (left), 0.75 (middle), and 0.85 (right). Other parameters: $p = 0.83$, $q = 0.2$, $p_0 = 0.4$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In all three cases the inner point as well as the $x_D = 1$ point are the only attractors of the system.

but doing it analytically for $n > 2$ is out of question. Once the matrices are obtained computing the vector π containing the 2^n stationary probabilities for each of the states simply amounts again to solving the linear system $\pi = \pi M$, readily providing the payoffs for any strategy i when confronted with any set $i_1, \ldots, i_n$ of strategies of the $n-1$ opponents. The result can be cast in a tensor $W = (W_{i,i_1,\ldots,i_{n-1}})$. For a population composition x the payoff received by an individual of strategy i will thus be

$$W_i(x) = \sum_{i_1,\ldots,i_{n-1}=\text{C, D, X}} W_{i,i_1,\ldots,i_{n-1}} x_{i_1} \cdots x_{i_{n-1}}, \tag{3.21}$$

and the average payoff of the population will be

$$\overline{W}(x) = \sum_{i=\text{C, D, X}} x_i W_i(x). \tag{3.22}$$

Finally, the replicator dynamics is then given by

$$\dot{x}_i = x_i[W_i(x) - \overline{W}(x)]. \tag{3.23}$$

Expression (3.21) can be further simplified if we exploit the symmetry implicit in public goods games, where the identity of the players is not at all relevant, only the number of them using a given strategy. This means that many payoffs are equal because

$$W_{i,i_1,\ldots,i_{n-1}} = W_i(n_\text{C}, n_\text{D}, n_\text{X}), \tag{3.24}$$

i.e., the payoff obtained by an i strategist only depends on the number n_C of cooperators, n_D of defectors, and n_X of conditional cooperators ($n_\text{C} + n_\text{D} + n_\text{X} = n - 1$) she is confronted to. Then

$$W_i(x) = \sum_{\substack{n_\text{C}+n_\text{D}+n_\text{X}=n-1 \\ n_\text{C},n_\text{D},n_\text{X}\geq 0}} \frac{(n-1)!}{n_\text{C}! n_\text{D}! n_\text{X}!} W_i(n_\text{C}, n_\text{D}, n_\text{X}) x_\text{C}^{n_\text{C}} x_\text{D}^{n_\text{D}} x_\text{X}^{n_\text{X}}. \tag{3.25}$$

As in Section 3.2, for the parameters obtained from the experiments there is no interior point that describes the coexistence of the three strategies. We subsequently proceeded as in the previous case and tried to find ranges of parameters for which such an interior point exists. It turns out that for groups of $n = 3$ players sets of parameters can also be found where the dynamics is similar to that for $n = 2$ (see Figure 3.7 for an example), albeit the parameters for which this happens are a bit different —but still reasonably close to those of the experiments from Chapter 2. As in the two player case, the structure displayed in this figures turns out to be extremely sensitive to variations in the parameters. Although we will not go into the details of those modifications here, we find it interesting to note that Figure 3.7 shows an evolution of the interior point with increasing p very similar to that for $n = 2$ (cf. Figure 3.3), albeit with more drastic changes, indicating that the existence of an interior point is less generic. For IMPDs with larger groups we find that, although for groups of $n = 4$ players it is still possible to find a Zeeman-like phase map, one has to choose values for p very close to one (meaning that cooperators and defectors are nearly pure strategies) and on top of that the region where this behavior can be obtained is extremely narrow. It can be clearly observed in Figure 3.8, where several of these maps are shown for different values of p_1, that variations of about 1% noticeably displace the location of the interior point. Importantly, it can be also observed from Figure 3.7 and Figure 3.8 that the basin of attraction of the interior point, when it exists, shrinks upon increasing the number of players, i.e., for $n = 4$ the fraction of trajectories that end up in the D attractor is larger than those ending in the interior point. Finally, for the largest group size we could handle computationally, $n = 5$ players, we have not been able to find an interior point for any choice of parameters. It turns out that the outcome of this game for $n \geq 5$ is well represented by the large group limit $n \to \infty$, which unlike the case of arbitrary but finite n, is amenable to analysis —as we show in the next section.

3.4 Infinitely large groups

Obtaining the payoffs (3.24) amounts to finding the stationary state of $(n+2)(n+1)/2$ Markov chains, each made of $(n_\mathrm{C}+1)\times(n_\mathrm{D}+1)\times(n_\mathrm{X}+1)$ states, where $n_\mathrm{C}+n_\mathrm{D}+n_\mathrm{X} = n$ defines the composition of the n-player group. The size of the corresponding Markov matrices grows as n^3, which makes it feasible to study groups even larger than $n = 5$ players. This will not be necessary though, because the resulting chain can be studied analytically in the limit $n \to \infty$, which characterizes well the behavior of large groups.

To determine how a group with $n_\mathrm{C} + n_\mathrm{D} + n_\mathrm{X} = n$ players of each type will respond in a given iteration of the prisoner's dilemma we only need to record the vector $(k_\mathrm{C}, k_\mathrm{D}, k_\mathrm{X})$ whose components count how many players of each strategy cooperate in a given round. Then the probability to observe the Markov chain in a certain state given

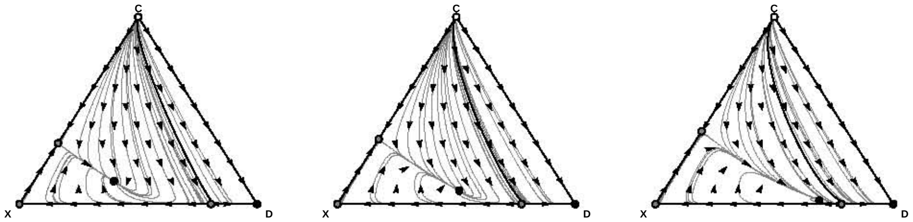


Figure 3.7: *Phase portraits for 3-players dynamics for different values of p.* The replicator dynamics for IMPD games with three strategies (C, D, and X) for the following values of p: 0.90 (left), 0.92 (middle), and 0.95 (right). Other parameters: $q = 0.10$, $p_0 = 0.20$, $p_1 = 0.95$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In all three cases the inner point as well as the $x_D = 1$ point are the only attractors of the system.

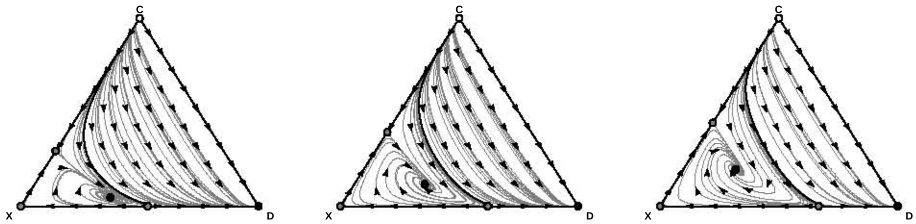


Figure 3.8: *Phase portraits for 4-players dynamics for different values of p_1.* Phase portraits of the replicator dynamics for IMPD games with three strategies (C, D, and X) for different values of p_1: 0.95 (left), 0.97 (middle), and 0.98 (right). Other parameters: $p = 0.95$, $q = 0.20$, $p_0 = 0.30$. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). In all three cases the inner point as well as the $x_D = 1$ point are the only attractors of the system.

that in the previous round the state was $(l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}})$ is

$$\begin{aligned}\Pr\left\{k_{\mathrm{C}}, k_{\mathrm{D}}, k_{\mathrm{X}} | l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right\} = &\binom{n_{\mathrm{C}}}{k_{\mathrm{C}}}\binom{n_{\mathrm{D}}}{k_{\mathrm{D}}} p^{n_{\mathrm{D}}-k_{\mathrm{D}}+k_{\mathrm{C}}}(1-p)^{n_{\mathrm{C}}-k_{\mathrm{C}}+k_{\mathrm{D}}} \\ &\times \sum_{j=0}^{k_{\mathrm{X}}} \binom{l_{\mathrm{X}}}{j}\binom{n_{\mathrm{X}}-l_{\mathrm{X}}}{k_{\mathrm{X}}-j} p_C(x)^j [1-p_C(x)]^{l_{\mathrm{X}}-j} \\ &\times q^{k_{\mathrm{X}}-j}(1-q)^{n_{\mathrm{X}}-l_{\mathrm{X}}-k_{\mathrm{X}}+j},\end{aligned} \tag{3.26}$$

with the usual convention that $\binom{a}{b} = 0$ for $b > a$ and where we have introduced the short-hand notation

$$x \equiv \frac{l_{\mathrm{C}} + l_{\mathrm{D}} + l_{\mathrm{X}} - 1}{n-1}.$$

Extracting analytical information for finite n from this matrix is not an easy task. However, let us focus on the limit $n \to \infty$. It is straightforward to show that

$$\begin{aligned} \mathrm{E}\left(\frac{k_{\mathrm{C}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) &= p\frac{n_{\mathrm{C}}}{n}, \\ \mathrm{E}\left(\frac{k_{\mathrm{D}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) &= (1-p)\frac{n_{\mathrm{D}}}{n}, \\ \mathrm{E}\left(\frac{k_{\mathrm{X}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) &= p_C(x)\frac{l_{\mathrm{X}}}{n} + q\frac{n_{\mathrm{X}} - l_{\mathrm{X}}}{n}, \end{aligned} \tag{3.27}$$

and

$$\begin{aligned} \mathrm{Var}\left(\frac{k_{\mathrm{C}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) =& p(1-p)\frac{n_{\mathrm{C}}}{n^2}, \\ \mathrm{Var}\left(\frac{k_{\mathrm{D}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) =& p(1-p)\frac{n_{\mathrm{D}}}{n^2}, \\ \mathrm{Var}\left(\frac{k_{\mathrm{X}}}{n}\middle| l_{\mathrm{C}}, l_{\mathrm{D}}, l_{\mathrm{X}}\right) =& p_C(x)[1-p_C(x)]\frac{l_{\mathrm{X}}}{n^2} \\ &+ q(1-q)\frac{n_{\mathrm{X}} - l_{\mathrm{X}}}{n^2}. \end{aligned} \tag{3.28}$$

Hence, introducing the random variable $r_i \equiv k_i/n_i$ and denoting $x_i \equiv n_i/n$, in the limit $n \to \infty$ the probability density of r_i becomes a delta function around $r_{\mathrm{C}} = p$, $r_{\mathrm{D}} = 1 - p$ and r_{X}, this last quantity arising from the solution to the equation

$$r_{\mathrm{X}} = \big\{p_0 + (p_1 - p_0)[px_{\mathrm{C}} + (1-p)x_{\mathrm{D}} + r_{\mathrm{X}}]\big\}r_{\mathrm{X}} + q(1 - r_{\mathrm{X}}). \tag{3.29}$$

If $p_0 = p_1$ this is a linear equation with solution $r_{\mathrm{X}} = q/(1 - p_0 + q)$. If $p_0 \neq p_1$ it is a quadratic equation with two solutions. The one that reduces to the solution found for $p_0 = p_1$ is

$$r_{\mathrm{X}} = \frac{2q}{\Delta + \sqrt{\Delta^2 - 4q(p_1 - p_0)x_{\mathrm{X}}}}, \tag{3.30}$$

$$\Delta \equiv 1 - p_0 + q - (p_1 - p_0)[px_{\mathrm{C}} + (1-p)x_{\mathrm{D}}]. \tag{3.31}$$

Notice that $\Delta > 0$ as long as $p_1 > p_0 > q$, as required.

Factors r_i yield the asymptotic, stationary fraction of cooperative actions among players of type i in the group. Hence the stationary level of cooperation is given by

$$\kappa \equiv px_{\mathrm{C}} + (1-p)x_{\mathrm{D}} + r_{\mathrm{X}}x_{\mathrm{X}}, \tag{3.32}$$

and the corresponding payoffs of the three type of players are

$$W_{\mathrm{C}}(x) = p\kappa R + p(1-\kappa)S + (1-p)\kappa T + (1-p)(1-\kappa)P, \tag{3.33}$$

$$W_{\mathrm{D}}(x) = (1-p)\kappa R + (1-p)(1-\kappa)S + p\kappa T + p(1-\kappa)P, \tag{3.34}$$

$$W_{\mathrm{X}}(x) = r_{\mathrm{X}}\kappa R + r_{\mathrm{X}}(1-\kappa)S + (1-r_{\mathrm{X}})\kappa T + (1-r_{\mathrm{X}})(1-\kappa)P. \tag{3.35}$$

Notice that

$$W_{\mathrm{D}}(x) - W_{\mathrm{C}}(x) = (2p-1)[\kappa(T-R) + (1-\kappa)(P-S)], \tag{3.36}$$

so as long as $p > 1/2$ we have $W_{\mathrm{D}}(x) > W_{\mathrm{C}}(x)$ i.e., cooperators are always dominated by defector irrespective of the composition of the population (provided $x_{\mathrm{D}} > 0$). This implies that no interior point exists in the limit $n \to \infty$, a property that suggests that the fact that we have not been able to locate an interior point for $n = 5$ is generic for larger values of n.

On the other hand,

$$W_{\mathrm{C}}(x) - W_{\mathrm{X}}(x) = (r_{\mathrm{X}} - p)[\kappa(T-R) + (1-\kappa)(P-S)], \tag{3.37}$$

$$W_{\mathrm{D}}(x) - W_{\mathrm{X}}(x) = (r_{\mathrm{X}} + p - 1)[\kappa(T-R) + (1-\kappa)(P-S)], \tag{3.38}$$

so any solution to $r_{\mathrm{X}} = p$ ($r_{\mathrm{X}} = 1-p$) determines a rest point on the $x_{\mathrm{D}} = 0$ ($x_{\mathrm{C}} = 0$) edge of the simplex. Taking the first equation and assuming $x_{\mathrm{D}} = 0$ we obtain

$$\left(\frac{2q}{p} - \Delta\right)^2 = \Delta^2 - 4q(p_1 - p_0)x_{\mathrm{X}}.$$

Upon simplification this equation becomes

$$q + p^2(x_{\mathrm{C}} + x_{\mathrm{X}}) = p(1 - p_0 + q).$$

Given that $x_{\mathrm{C}} + x_{\mathrm{X}} = 1$ on the $x_{\mathrm{D}} = 0$ edge of the simplex, it turns out that $r_{\mathrm{X}} = p$ does not hold for any point of this edge. A similar argument yields the same result for $r_{\mathrm{X}} = 1-p$ on the $x_{\mathrm{C}} = 0$ edge of the simplex (the equations are the same just replacing p by $1-p$ and x_{C} by x_{D}).

We have thus established that, depending on the parameters $p_1 > p_0 > q$ and $p > 1/2$, on the $x_{\mathrm{D}} = 0$ edge of the simplex either $W_{\mathrm{C}}(x) > W_{\mathrm{X}}(x)$ or $W_{\mathrm{C}}(x) < W_{\mathrm{X}}(x)$ irrespective of the composition, and on the $x_{\mathrm{C}} = 0$ edge of the simplex either $W_{\mathrm{D}}(x) > W_{\mathrm{X}}(x)$ or $W_{\mathrm{D}}(x) < W_{\mathrm{X}}(x)$ irrespective of the composition. In order to decide which one of the inequalities holds on each edge we can set an arbitrary composition, namely $x_{\mathrm{X}} = 1$. At this corner of the simplex

$$\begin{aligned} r_{\mathrm{X}} &= \frac{2q}{1 - p_0 + q + \sqrt{(1-p_0+q)^2 - 4q(p_1-p_0)}} \\ &= \frac{1 - p_0 + q - \sqrt{(1-p_0-q)^2 + 4q(1-p_1)}}{2(p_1-p_0)}. \end{aligned} \tag{3.39}$$

Then $W_{\mathrm{C}}(x) > W_{\mathrm{X}}(x)$ on $x_{\mathrm{D}} = 0$ if, and only if,

$$\frac{1 - p_0 + q - \sqrt{(1-p_0-q)^2 + 4q(1-p_1)}}{2(p_1-p_0)} > p > \frac{1}{2}, \tag{3.40}$$

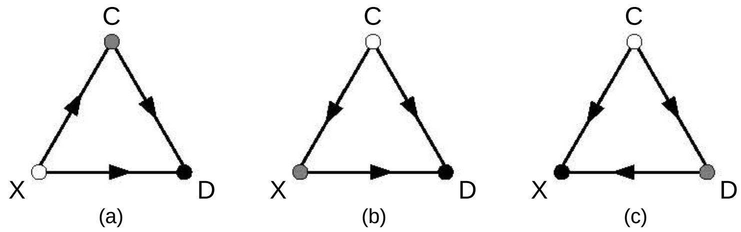


Figure 3.9: *Phase diagrams for infinitely large groups.* The only three phase portraits of the replicator dynamics for IMPD games with three strategies (C, D, and X) played in infinitely large groups. Rest points marked in the plot can be repeller (white), saddle points (gray) or attractors (black). Map (a) appears if inequality (3.40) holds (a necessary condition for this is $q + (p_1 + p_0)/2 > 1$); map (b) appears if inequality (3.40) does not hold but inequality (3.41) does; map (c) appears if neither (3.40) nor (3.41) hold (a sufficient condition for maps (b) and (c) to appear is $q + (p_1 + p_0)/2 < 1$).

and $W_{\mathrm{D}}(x) > W_{\mathrm{X}}(x)$ on $x_{\mathrm{C}} = 0$ if, and only if,

$$\frac{1 - p_0 + q - \sqrt{(1 - p_0 - q)^2 + 4q(1 - p_1)}}{2(p_1 - p_0)} > 1 - p. \tag{3.41}$$

Notice that if (3.40) is true so is (3.41) (but the converse does not hold).

For (3.40) to hold a necessary condition is that the left-hand side is larger than $1/2$, a condition that boils down to

$$\begin{aligned} 1 - p_1 + q &> \sqrt{(1 - p_0 - q)^2 + 4q(1 - p_1)} \\ &= \sqrt{(1 - p_1 + q)^2 + 2(p_1 - p_0)\left(1 - q - \frac{p_1 + p_0}{2}\right)}. \end{aligned}$$

As $p_1 > p_0$, the only way that this can hold is if $q + (p_1 + p_0)/2 > 1$. When inequality (3.40) is satisfied, D is an attractor, X is a repellor, and C a saddle point. Otherwise C is a repeller (obviously, a sufficient condition for this to happen is $q + (p_1 + p_0)/2 < 1$). In this case D is an attractor and X a saddle point if (3.41) holds and vice versa if it does not.

A summary of our results for $n \to \infty$ is shown in the sketch of Figure 3.9. As we can see from the plot, the main results are that there never exists an interior point, that homogeneous C populations are not stable, and that in two out of three cases the final result of the dynamics is a homogeneous D population. Therefore, although there is a region of parameters in which a homogeneous population of moody conditional cooperators is actually stable, we never observe coexistence even of pairs of strategies.

3.5 Discussion

Motivated by the experimental work from Chapter 2, where conditional cooperation depending on the player's previous action was observed in a spatial prisoner's dilemma coexisting with cooperation and defection, we have studied the replicator dynamics of the IMPD with these three strategies. The fact that the experimental results indicated that all three strategies were getting on average the same payoff suggested that they were in equilibrium; on the other hand, as the presence of a lattice had no significant consequences on the level of cooperation, it seemed likely that the spatial game could be understood in terms of separate multiplayer games.

Assuming a stylized version of the behaviors mentioned above, we have focused on the problem of their coexistence in well-mixed populations, when they interact in groups of $n \geq 2$ players through an IMPD. For $n = 2$, in a region of parameters compatible with those of the experiment we do find a mixed equilibrium in which all three types of players coexist, and they do it in a proportion similar to that found in the experiments. The phase portrait of the replicator dynamics reproduces that of a three-strategies game introduced by Zeeman (1980). However, upon increasing n, the region of parameters of this Zeeman-like dynamics shrinks, and for $n = 5$, the maximum size we could analyze with our analytical approach, we could not find a mixed equilibrium anymore.

Given that our Markov chain technique becomes computationally untraceable for larger sizes, we have carried out a rigorous analysis of the replicator dynamics for this game in the limit $n \to \infty$. The analysis reveals that in this limit, all rest points other than the three corners of the simplex —that can be found for small n— disappear. The dynamics in this limit is determined by who beats who, depending on the parameters. Cooperators are always defeated by defectors, but depending on the parameters, conditional cooperators are displaced by any other strategy, or only by defectors, or they can displace the other two strategies.

Putting together our numerical results for small n and our analytical calculations for large n, we can conclude that an imitative evolution like the one represented by replicator dynamics cannot account for the coexistence of strategies observed in the experiments, at least in groups as large as $n = 9$ (the case of the experiment). The reasons for this can be many. The most obvious one is that replicator dynamics might not be what describes the evolution of strategies in human subjects. In this regard, we have to make it clear that we are not studying the evolution of the players during the experiment, as it was shown in Chapter 2 that there is no learning. Our evolutionary approach would apply to much longer time scales, i.e., these strategies would have arisen from interactions of human groups through history. It may then well be the case that this slower evolution of human behavior requires another approach to its dynamics. By the same token, it might also occur that the typical number of iterations of the game is not very large, so the stationary probability density obtained from the Markov chains is not a good approximation to the observed behavior. All in all, it is clear that our analytical model might not be the most appropriate one to describe human behavior on IMPDs.

Nevertheless, another possible explanation for the discrepancy between our predictions and the coexistence of moody conditional cooperators with the cooperator and defector strategists might come from bounded rationality considerations. Thus, people may behave in a IMPD as though they were playing a (two-person) IPD with some kind of an “average” opponent, something that can be reinforced by the computer interface of the experiment that isolates the subjects from the other ones with whom they interact. Such a heuristic decision making process might be the result of cognitive biases or limitations, among which the inability to deal with large numbers may be of relevance here (Kahneman et al. 1982), or else it could arise as an adaptation itself (Gigerenzer and Selten 2001). Whatever the underlying reason, the fact that for $n = 2$ and $n = 3$ players we can easily find wide ranges of parameters for which the three strategies coexist and, furthermore, this coexistence have a large basin of attraction, suggests that the idea that people may be extrapolating their behavior to larger groups should at least be considered, and tested by suitably designed experiments.

On the other hand, it should be borne in mind that the strategies reported in Chapter 2 are aggregate behaviors, as they attempted to classify the actions of the player in a few archetypal types. Therefore, there may actually be very many different moody conditional cooperators, defined by different p_0, p_1 and q parameters and different propensities to cooperate (parameter p) among cooperators and defectors. Alternatively players who were classified as conditional cooperators might be using a totally different strategy, different for every player, which aggregated would look like the conditional cooperation detected in the experiment. This is not included anywhere in our replicator dynamics. It is certainly possible that considering several different subclasses of the strategy X in the replicator dynamics might actually provide an explanation for coexistence in larger groups. However, the corresponding calculations become very much involved, and whether this variability can sustain mixed equilibria is an interesting question that remains out of the scope of this work.

As a final remark, we would like to stress that, notwithstanding the issue that the agreement between our results and the experiments is problematic, this study proves that, under replicator dynamics, even for $n \to \infty$ our work predicts the dominance of moody conditional cooperators for certain regions of parameters. It is important to realize that this type of strategy had not been considered prior to the experimental observation, and as we now see it can successfully take over the entire population even from defection when playing an IMPD. This suggests that this or similar strategies may actually be more widespread than this simple case as they might also be the best ones in related games, such as the public goods game. It would be worth widening the scope of this work by analyzing the possible appearance of this conditional cooperators who are influenced by their own mood in other contexts, both theoretically and experimentally. In this regard, an explanation of the evolutionary origin of moody conditional cooperators would be a particularly important, albeit rather difficult goal.

4

Strategy updating in spatial and nonspatial settings

In Chapter 2 we presented the analysis of a laboratory experiment with human subjects playing an Iterated PD on a square lattice. Here we analyze the data of a similar laboratory experiment performed by a different group (Traulsen et al. 2010) in order to ascertain the strategy used by the players to update their actions. One important question from the perspective of a theoretician is whether human subjects condition their decision making on the population structure, i.e. whether they use the same strategy updating in spatial and non-spatial experiments. Traulsen et al. (2010) conducted two type of experimental treatments, one on the spatial structure, analogous to the experiment treatment of Chapter 2 and controls, analogous to the control treatment in the experiment of Chapter 2. Unlike the experiment from Chapter 2, here the two treatments were always conducted with different groups of volunteers who had no previous experience from playing the other type of treatment. This makes the experiment suitable for a detailed comparison of the way players behave in different settings. Previously, these data have only been used to infer the strategy updating in the spatial system, but no systematic comparison between the two treatments had been provided.

4.1 Experimental setup

The experimental setup in Traulsen et al (2010) is similar (but not identical) to the experiment described in Chapter 2. We will now describe the way this experiment was carried out and then we will discuss similarities and differences in Chapter 6.

Participants in the experiment treatment discussed here were virtually located on the nodes of a 4×4 square lattice with periodic boundary conditions (as if players would be located on a torus). They played a PD game with each of the four neighbors in their von Neumann neighborhood (the four cells orthogonally surrounding a central cell on the lattice). Players had to choose one action, the same for all four games with their four neighbors. The payoffs were calculated by adding the four payoffs of individual games with each neighbor. There were no self-interactions[1]. After each round, players were informed about their action and payoff, as well as the actions and payoffs of their four neighbors. Based on this information and their experience from previous interactions, they had to decide on their next action. The payoffs were chosen as $T = 0.40$ €, $R = 0.30$ €, $P = 0.10$ €and $S = 0.00$ €. Notice that this payoffs correspond to a strict PD game. This is an important difference with respect to the experiment reported on in Chapter 2 and we believe that some of the differences in the outcomes of the two experiments can be attributed to this fact.

The experiment had two different treatments. In the experiment treatment players had fixed neighbors, which stayed the same throughout the whole game. This treatment was repeated 15 times, each with 16 players and 25 rounds. In the control treatment (repeated 10 times with 16 players and 25 rounds), the players were assigned to a new, random location on the lattice after each round and consequently, the neighbors of each player changed in each round. In both treatments, players were informed every round about the actions and payoffs of the neighbors they played with. However, at the moment they had to make a decision about their next action, they were not informed about the previous actions or payoff of their new neighbors. In contrast, it was easy to remember the previous actions of the neighbors in the experimental setting. We emphasize that each player was identified by a letter ranging from *a* to *p* (e.g., *a* has the following neighbors: *b*, *d*, *e*, and *m*). Therefore, in the experiment treatment players could see that their neighbors were always the same, for example: f, d, e, and a. Subjects were told in the instructions that their neighbors would stay the same throughout. On the other hand, in the control treatment, players could see that in each round they had different neighbors. Subjects were told in the instructions that their neighbors would change after each round. Consequently, it is highly unlikely that the players misunderstood their specific rules of the game. A detailed explanation of the experiment can be found in Traulsen et al. (2010).

4.2 General observables in the two treatments

Let us compare the general outcomes of the experiment and control treatments.

We find no significant differences in the fraction of cooperative actions between the two treatments. Figure 4.1 illustrates that the errors bars of the treatments are overlapping to a large extend, which suggests that there are no large differences between the treatments. This can be backed up by several statistical tests. First, we fit the difference

[1]Although self-interactions were considered in the simulations of Nowak and May (1992) they make no sense in a social context, so no experiment includes them.

between the two treatments with a linear function. We find an intercept of 0.001 ± 0.017 and a slope 0.001128 ± 0.001153. Since both values are smaller than their errors, it suggests that the values are close to zero. Second, we constructed a nonlinear regression model with a dummy variable for the experiment and the control treatments. This and other regression models were done in The R Project for Statistical Computing (R Development Core Team 2011). In this model, the fraction of cooperative actions $C(t)$ in round t is given by

$$C(t) = (C(1) + s\Delta C(1))(\Gamma + s\Delta\Gamma)^{t-1} \tag{4.1}$$

Here, the parameters of the model are $C(1)$, measuring the fraction of cooperative actions in the first round of the control treatment, $\Delta C(1)$, measuring the difference in

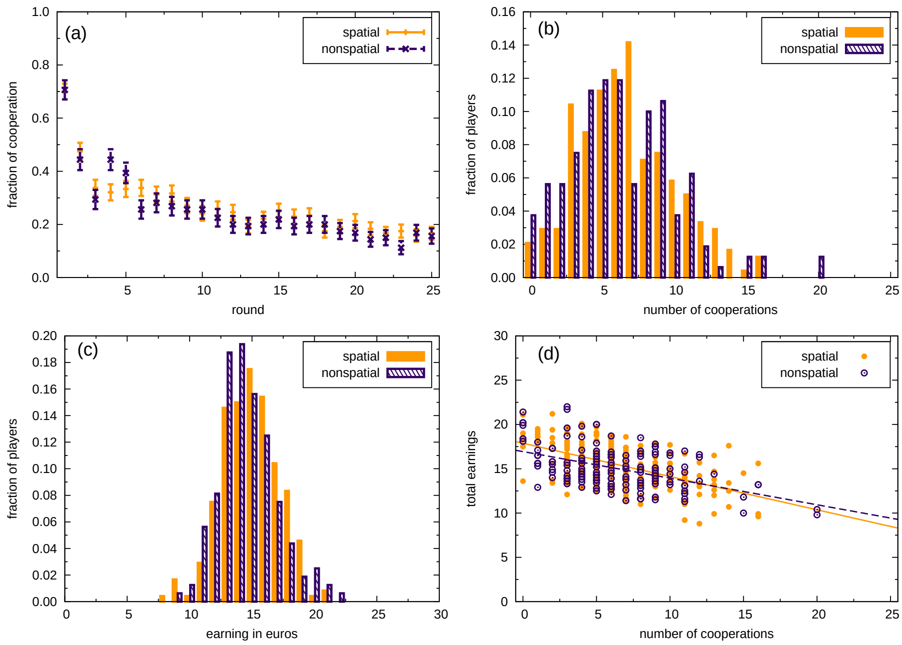


Figure 4.1: *Comparison of the experiment and control treatments.* (a) The fraction of players that have chosen to cooperate is decreasing over time, but remains substantial throughout the experiment (Traulsen et al. 2010). The error bars are the standard deviations of a binomial distribution, $\sqrt{C(1-C)/n}$, where n is the number of samples, and C is fraction of cooperation. (b) The distribution of cooperative acts per player. We do not observe unconditional cooperation, and very little unconditional defection (5 out of 240 players in the experiment treatment and 6 out of 160 players in the control treatment). (c) The distributions of cumulative payoffs are peaked with median of 15.4 € for the experiment and 15.0 € for the control treatment. The standard deviation is 2.3 € in both cases. (d) Correlation between the frequency of cooperation on the x-axis and the cumulative payoff on the y-axis. Each point is one player.

cooperative actions in the first round between the two treatments, Γ, measuring the decay on cooperative actions in the control treatment, and $\Delta\Gamma$, measuring the difference in this decay between the two treatments. In addition, we introduced the dummy variable s, which equals 0 for the control treatment and 1 for the experiment treatment. From the nonlinear regression model, we find $C(1) = 0.49$ ($p < 10^{-3}$) and $\Gamma = 0.94$ ($p < 10^{-3}$). For the differences, we obtain $p = 0.75$ for $\Delta C(1)$ and $p = 0.33$ for $\Delta\Gamma$, showing that the dependence on the dummy variable is not statistically significant. All this suggests that the difference between two treatments is not significant.

Next, we address the distribution of cooperative actions per player, the distribution of cumulative payoff per player, and the correlation between the two. Figure 4.1 illustrates that these two distributions are very similar in the two treatments. To compare the distributions between the treatments quantitatively, we performed a Kolmogorov-Smirnov test. We found $p = 0.69$ for the comparison of the two distributions of cooperative acts and $p = 0.13$ for the comparison of the two distributions of cumulative payoffs. These p-values indicate that we cannot accept the hypothesis that the two distributions arising from the two treatments are different. In order to compare the correlation between the cumulative payoffs and the number of cooperative acts, we developed a linear regression model,

$$E(N_C) = E_0 + s\Delta E_0 + \rho N_C + s\Delta\rho N_C, \tag{4.2}$$

where E is the cumulative payoff, N_C is the number of cooperative acts, E_0 is the intercept for the control treatment and ΔE_0 the difference between the intercepts of the two treatments. The slope in the control treatment is measured by ρ and $\Delta\beta$ measures the difference of the slope between the two treatments. Again, s is a dummy variable which is equal to 0 for the control treatment and 1 for the experiment treatment. We obtained $E_0 = 16.9 \pm 0.3$ ($p < 10^{-3}$), $\rho = -0.30 \pm 0.05$ ($p < 10^{-3}$), $\Delta E_0 = 0.9 \pm 0.5$ ($p = 0.022$), and $\Delta\rho = -0.7 \pm 0.6$ ($p = 0.17$). The large p-values for ΔE_0 and $\Delta\rho$ show that there is no significant difference between the two treatments.

4.3 Update strategies

In this section, we depart from the level of aggregate information on the system level and address the individual decisions of our players in more detail. Many theoretical models have shown that the kind of update strategy used by players can have a profound impact on the outcome in such models (Hauert 2002; Santos and Pacheco 2005; Szabó and Fáth 2007; Roca et al. 2009b).

In order to understand the dynamics of the system in more detail, we fitted two different update mechanisms that are popular in theoretical studies to the data of the experiment plus the kind of update strategy uncovered by the experiment of Chapter 2:

(i) *Unconditional imitation*, where each players switches to the action that performed best in the past in the neighborhood. In addition, we assume that some decisions are made at random and that this fraction changes over time.

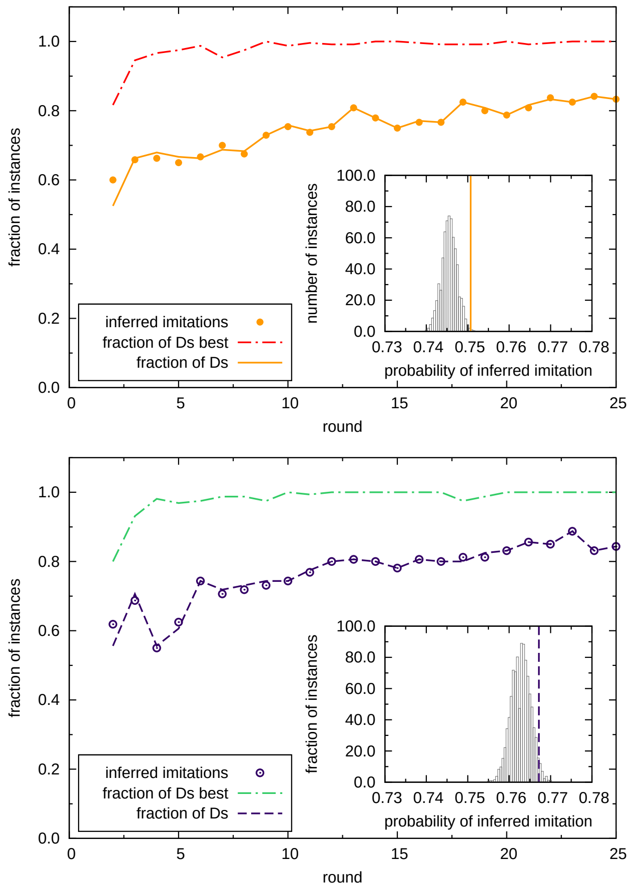


Figure 4.2: *Unconditional imitation test in the experiment treatment (top) and in the control treatment (bottom).* The main panels show three different type of data: the fraction of inferred imitations, the level of defection and the fraction of decisions in which defection was the best performing strategy in the neighborhood. The inferred level of imitation is the fraction of actions in which the players action coincided with the action of the best neighbor in the previous round. Since defection is almost always the best performing strategy, a defecting player seems to be imitating. Therefore, the level of defection is almost identical to the level of the inferred imitations. However, the randomization test illustrated in the inset shows that there is still more imitation than expected in a random setting. The vertical lines show the inferred imitation observed in the experiments and the gray bars show the distribution of the inferred imitation in the randomized sample.

(ii) *Fermi rule*, where action with higher payoffs are imitated with higher probability. In addition, sometimes a random decision is made.

(iii) *Moody conditional cooperation*, where cooperation is conditioned upon the own action in the previous round and the number of cooperators in the neighborhood.

In the pioneering studies of the promotion of cooperation on lattices, unconditional imitation has been assumed (Nowak and May 1992; Nowak et al. 1994b; Nowak et al. 1994a).

In this case, players update their strategies by imitating the previous action of the neighbor with the highest payoff. In Figure 4.2, we illustrate how often the player's action is the same as the action of the highest scoring neighbor in the previous round. The probability of this inferred imitation is around 75% and is growing during the game. However, before we conclude that the unconditional imitation is the update strategy players use frequently, we should notice that defection is almost always the most successful strategy in the neighborhood. Therefore, if a player defects it seems that she is imitating the best neighbor. Consequently, the level of defection is very similar to the level of inferred imitation (Figure 4.2). To further test the hypotheses of unconditional imitation we applied a non-parametric bootstrap (Efron and Tibshirani 1993) method. In this test, the action of the players is kept, but the neighborhood is randomized. This gives a reference model for imitation, because with randomized neighborhoods there can be no imitation. We repeated the randomization 10 000 times to compute the distribution of probabilities of inferred imitation from a random setting. The results are presented in the insets of Figure 4.2. We see that distributions are very narrow and that the value from the experiment is slightly higher than the randomized average. The p-value is $p = 0.001$ for the experiment treatment and $p = 0.0283$ for the control treatment, indicating that the difference between the observed imitation and the randomized one is fairly significant. The difference between this results and the same test on the data from Chapter 2 will be discussed in Chapter 6.

The second mechanism we tested is typically referred to as Fermi rule (Blume 1993; Szabó and Töke 1998; Traulsen et al. 2006). For this rule, the better the neighbor performs the higher the chances that she will be imitated, see Fig. 4.3. More precisely, the probability of switching action increases with the payoff differences between the focal player and the best player with the opposite strategy according to a Fermi function, $(1 + \exp[-\beta\Delta\pi])^{-1}$. Here, β measures the intensity of selection: for $\beta \to 0$ imitation is random and for $\beta \to \infty$, we recover unconditional imitation. Note that this is slightly different from the original Fermi update mechanism. In the original mechanism a random player is chosen and then imitated with the probability given above. However, the additional randomness would make it difficult to analyze the original rule in the experimental data. Therefore we measure the probability of imitating the most successful neighbor who played the opposite strategy instead. To analyze this dependence, we again fitted the data to a logistic regression model,

$$P_{C\leftrightarrow D}(\Delta\pi, s) = \frac{1}{1 + e^{-(\alpha + s\Delta\alpha + \beta\Delta\pi + s\Delta\beta\Delta\pi)}}. \tag{4.3}$$

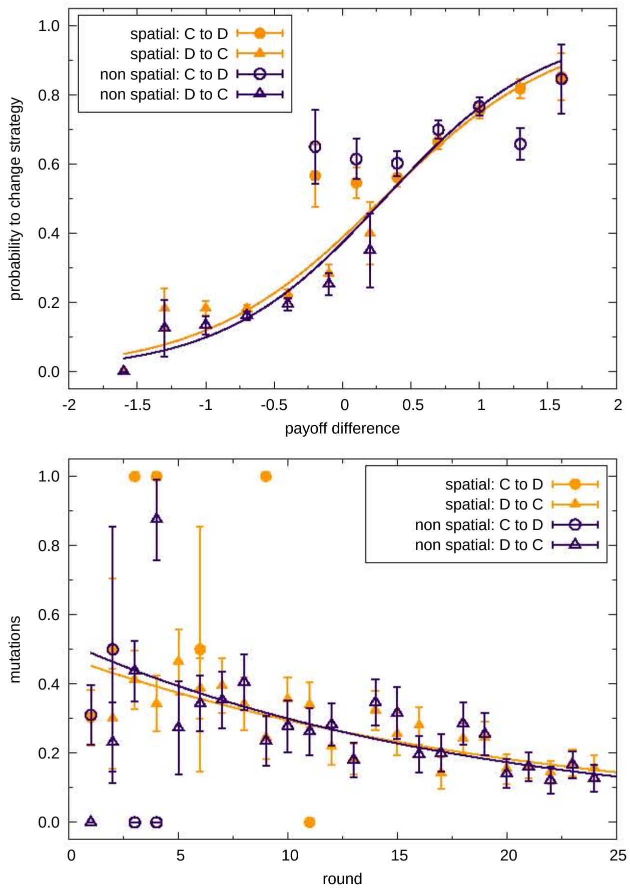


Figure 4.3: *Probability of imitating depending on the payoff difference.* Top: probability of switching to another strategy depending on the payoff differences for both the experiment and control treatment. The payoff difference is between the focal player and the best player of the opposite strategy. The results are consistent with imitating the neighbors with higher payoffs. However this imitation is not unconditional, but the higher the payoff difference the larger is the probability of imitation. In addition, players might spontaneously switch their strategies even if they have no neighbors playing the other strategy, resembling mutations. Error bars are the standard deviations of a binomial distribution, $\sqrt{P_{C\leftrightarrow D}(1-P_{C\leftrightarrow D})/n}$, where n is the number of samples and $P_{C\leftrightarrow D}$ is the probability to change the action. Bottom: Probability of mutations in time. Mutations are defined as the probability that a cooperator surrounded by four cooperators would change the strategy in the next round or that a defector surrounded by four defectors will change the strategy in the next round. We see a large number of mutations, which decreases over time, but always stays substantial. Again in both treatments the players show a similar pattern of behavior. Error bars are the standard deviations of a binomial distribution, $\sqrt{M(1-M)/n}$, where n is the number of samples and M is the probability of mutation.

Here, α measures the probability to switch strategy in the case of zero payoff differences and $\Delta\alpha$ measures the difference between this quantity in the two treatments. The parameter β measures the intensity of selection and $\Delta\beta$ is the difference in the intensity of selection between the treatments. As above, s is a dummy variable with $s = 0$ for the control and $s = 1$ for the experiment treatment. The p-values for $\Delta\alpha$ and $\Delta\beta$ are 0.7 and 0.6, respectively, so the dependence on the treatment is not significant.

However, the players can switch their strategies even if they are surrounded by players with the same strategy as theirs. This corresponds to mutations or exploratory behavior (Traulsen et al. 2009). This exploratory behavior decreases over time in a manner comparable to the decrease of the global cooperation level (Figure 4.2). To analyze the difference between the experiment and control treatments, we use the non linear regression model

$$M(t) = (\mu + s\Delta\mu)(\Gamma + s\Delta\Gamma)^{t-1}, \tag{4.4}$$

where $M(t)$ is the fraction of exploratory behavior in round t. The initial level of exploration is measured by μ and $\Delta\mu$, and its decay is measured by Γ and $\Delta\Gamma$. The p-values for the parameters $\Delta\mu$ and $\Delta\Gamma$ are both 0.48. Thus, the dependence on the treatment is statistically not significant. However, the behavior shows a significant compatibility with the Fermi rule.

The last update mechanism we analyzed is moody conditional cooperation, first proposed to explain the experiment of Chapter 2. This behavior is based on the own previous action and the number of cooperators in the neighborhood. In Figure 4.4, we show the probability of cooperating depending on the number of neighbors who cooperated in the previous round and the action of the focal players in the previous round. In the case that the focal player cooperated, the probability that she cooperate increases linearly with the number of cooperating neighbors, as in the experiment presented in Chapter 2. On the other hand, if the player defected, the probability that she cooperates decreases linearly with the number of cooperating neighbors. We developed a linear regression model with two dummy variables,

$$P_C(l) = \gamma_0 l + \gamma_1\lambda + \gamma_2 s + \gamma_3 l\lambda + \gamma_4 ls + \gamma_5\lambda s \tag{4.5}$$

where $P_C(l)$ is probability of cooperation after l of your neighbors cooperated in the previous round. The γ_i are the parameters of the model, etc. Again, s is a dummy variable which is equal to 0 for the control treatment and 1 for the experiment treatment. The second dummy variable λ is equal to 1 if the focal player cooperated herself in the previous round and 0 otherwise. $P(l)$ depends significantly only on γ_0 and γ_3 (p-values $< 10^{-4}$). Therefore, the probability of cooperating does not depend on the treatment. It does not depend either on the number of cooperators in the previous round if we do not control for the players own action. We could also consider a term $\gamma_6\lambda cs$; however, it turns out that dependence on that term is not significant, and therefore we did not include it in the model.

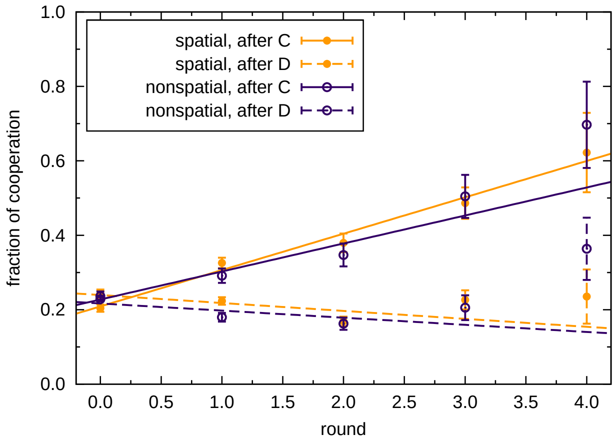


Figure 4.4: *Moody conditional cooperators.* Probability of cooperating depending on the previous action and the number of cooperators in the neighborhood in the previous round. We see that there is a clear difference between the behavior after the cooperating and defecting. After cooperating, the probability of cooperating increases with number of cooperating neighbors and after defecting the probability of cooperation decreases with number of cooperating neighbors. Again, in both the experiment and the control treatment the behavior is very similar.

4.4 Simulations

In the experiments, there is no hint for a significant difference between the treatments. In order to understand why this happens, we have performed simulations with the three update mechanisms fitted to the experimental data: unconditional imitation (with random strategy exploration), the Fermi rule (also with random strategy exploration) and moody conditional cooperation. Simulations were performed for an experiment and a control setting. In order to analyze the influence of the size of the lattice, we carried out simulations for lattice sizes 4×4 and 100×100. Figure 4.5 shows the levels on cooperation in these simulations.

The unconditional imitation with random strategy exploration obeys the equation

$$P_{A\to B}(\Delta\pi) = \mu\Gamma^{t-1} + (1 - 2\mu\Gamma^{t-1})\Theta(\Delta\pi) \qquad (4.6)$$

where $P_{A\to B}$ is the probability that a player with action A will change her action to B, provided B is the action of the best performing neighbor. The round of the game is t, $\Delta\pi = \pi_B - \pi_A$, where π_A is the payoff of player A and π_B is the payoff of her best performing neighbor playing B. $\Theta(x)$ is the Heaviside function, which is one for positive arguments and zero otherwise. From the experimental data, we found $\mu = 0.380 \pm 0.013$ and $\Gamma = 0.962 \pm 0.003$. For the simulations with imitation only we set the random strategy exploration parameter $\mu = 0$. In the first round, C is played with probability 70% and in the every other round the probability of imitating the best player is determined according to the probability given by Eq. (4.6).

If the player does not imitate she/he will play C or D with equal probability. We see that the simulations with $\mu > 0$ reproduce the cooperation level well, but as we saw before, the best performing neighbor will almost always be a defector. Therefore the above update mechanism is equivalent to the mechanism where the next action is determined only by the term $\mu\Gamma^{t-1}$ in Eq. (4.6). Promotion of cooperation can only occur through the formation of clusters of cooperators, which is prevented by the random strategy exploration. Therefore, since clusters of cooperators cannot be formed anyway, both experiment and control treatments show low levels of cooperation driven by $\mu > 0$ only. On the other hand, for $\mu = 0$, the two simulation setups display very different dynamics. In the experiment treatment, the level of cooperation drops at the beginning, until clusters start forming and expand in a sufficiently large system. In the control treatment, such clusters cannot form and the cooperation level drops to zero.

In the Fermi update rule, the probability of switching to the opposite strategy depends on the difference of the payoffs between the focal player its neighbors. The dependence is given by the Fermi function, see above. While conventionally a random neighbor is chosen for comparison, in the analysis of the experimental data we have focused on the neighbor with the opposite strategy and the highest payoff. In our simulations, we take the same approach. If there are no players with the opposite strategy in the neighborhood, players will still switch their strategy with some probability. We call this mutations or exploratory behavior. In contrast to Traulsen et al. (2010), here we assume that this quantity is time dependent. In the top panel of Figure 4.3, we present the probability of mutations over time. Summarizing this approach we find for the probability of changing strategy

$$P_{C\leftrightarrow D}(\Delta\pi) = \mu\Gamma^{t-1} + (1 - 2\mu\Gamma^{t-1})\frac{1}{1 + e^{-\beta\Delta\pi+\alpha}}. \tag{4.7}$$

Note that for $\beta \to \infty$ we recover unconditional imitation. For the simulations, we used the parameters obtained from fitting to the experiment, $\beta = 0.15 \pm 0.01$, $\alpha = 0.45 \pm 0.07$, $\mu = 0.45 \pm 0.05$, $\Gamma = 0.954 \pm 0.007$ for the experiment treatment and $\beta = 0.17 \pm 0.02$, $\alpha = 0.52 \pm 0.11$, $\mu = 0.49 \pm 0.07$, $\Gamma = 0.947 \pm 0.009$ for the control treatment. We can see that in both treatments the same kind of behavior emerges. It appears that for this update strategy the spatial structure is irrelevant.

The last model we simulated is moody conditional cooperation. The probability of cooperating is given by

$$P_{C|C}(l) = a_C + b_C l \qquad P_{C|D}(l) = n_D + k_D l \tag{4.8}$$

where $P_{C|C}(l)$ is the probability of cooperation after l neighbors cooperated and the focal player cooperated, $P_{C|D}(l)$ is the same probability after the focal player defected. The parameters are $a_C = 0.20$, $b_C = 0.35$, $a_D = 0.22$, $b_D = -0.08$. Independently of the spatial settings (whether experiment or control) the behavior stays the same. Like the Fermi rule, the moody conditional cooperation rule leads to the same level of cooperation in the experiment and the control treatments. This is in an agreement with

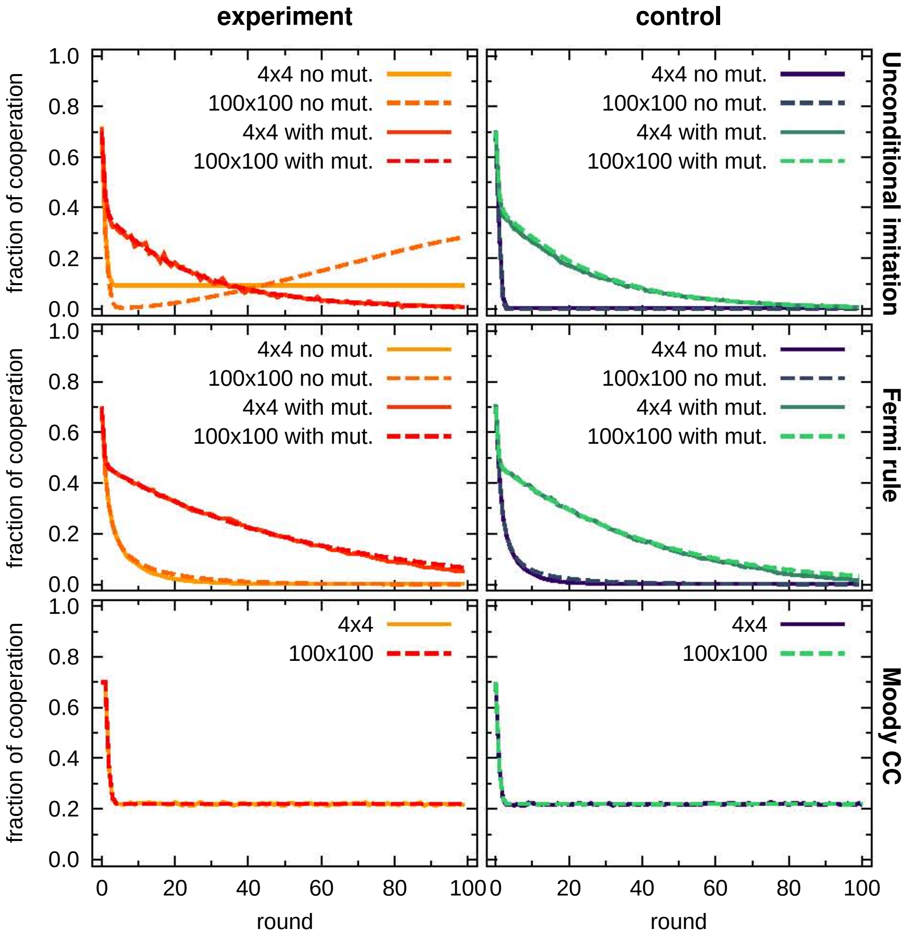


Figure 4.5: *Simulations for different update strategies.* Left figures are for the experimental treatment and the right ones are for the control treatment. Top to bottom: unconditional imitation, Fermi rule and moody conditional cooperation. Comparing the results for the experiment and control treatments we observe that the only update strategy for which the spatial structure is relevant is unconditional imitations without random strategy exploration.

results by Gracia-Lázaro et al. (2012), where they show that in a population of cooperators, defectors and moody conditional cooperators, the structure of the population does not promote or inhibit cooperation with respect to a well mixed population.

Summarizing, we find that these three update mechanisms will not promote cooperation on lattices and that for them the spatial structure does not make any difference, even for much larger systems.

4.5 Discussion

We have compared a spatial (experiment) and a non spatial (control) behavioral experiments with human subjects playing an iterated Prisoner's Dilemma. We have found no significant differences between the two treatments, neither in macroscopic properties such as the level of cooperation, nor in the way that players update their strategies. On the one hand, this is good news for theorists, because their assumption of consistent strategy updates in spatial and non spatial systems seems to be justified. On the other hand, our results strongly suggest that the idea that spatial structure promotes cooperation cannot be carried over to human experiments in a straightforward way. This result is in line with previous results from other experiments. Thus, Cassar found that cooperation was hard to reach on different, albeit small networks (Cassar 2007). Kirchkamp and Nagel performed an experiment on a one dimensional lattice which suggests that naive imitation may be negligible in such experiments (Kirchkamp and Nagel 2007). Suri and Watts performed an online experiment and found that the network topology has no significant effect on the level of cooperation (Suri and Watts 2011). The experiment presented in Chapter 2 did not report the promotion of cooperation either. There the experimental and control treatment do show different behavior, however the control treatment was not fully independent, since the same players were used for both treatments subsequently.

However, our results do not imply that the theoretical analysis of spatial games is not meaningful. In other biological or technological systems these considerations may be applicable directly. Moreover, the effect of spatial structure could be much more subtle than implied by many theoretical works. In particular, theories should consider the role of mutations (which may arise from mixed strategies, strategies that try to anticipate the future behavior of the neighbors, or from strategies which consider more than one past interaction) in structured populations, which is only rarely done (Allen et al. 2012).

Most importantly, theoretical considerations of fixed networks are a necessary first step to analyze dynamical networks, in which networks changes as part of the individual strategies. This may be a more realistic way to address human behavior. Recent experiments of such dynamical networks indicate that there is indeed a scope for the evolution of cooperation mediated by network structure (Fehl et al. 2011; Rand et al. 2011). In addition, a paper by Apicella et al. (2012) suggest that early humans may have formed ties based in part on their tendency to cooperate. There are also evidences that in the cooperativeness of the individuals is highly correlated with individual's social network position, e.g., the more central node are also the more cooperative one (Brañas-Garza et al. 2010; Kovářík et al. 2012). Therefore, the capability to form a population structure may have played a crucial role in our evolutionary past and potentially also in our present and future.

Finally, all the update strategies discussed in this chapter are compatible with the way our subjects made decisions. However, based only on this experiment it is difficult to conclude which decision process humans use. In order to answer that question, new experiments specially designed to answer this question are needed.

5

Promotion of cooperation, moody conditional cooperation, and group size: further experiments

In previous chapters we have seen that experiments provide a strong evidence for the existence of moody conditional cooperation, where players behavior (unlike for plain conditional cooperators) also depends on the players' own action, not just the actions of their neighbors. While conditional cooperation, be it plain or moody, provides a way to understand the experimental observations in small groups, it also poses new questions. To begin with, explaining the decay of cooperation in spatially structured populations by this means requires well-mixing of conditional cooperators and free-riders in the system: Indeed, if a set of reciprocal players happen by chance to be together, they could form a cluster capable to sustaining cooperation (Nowak and May 1992). This problem can be overcome by resorting to the fact that if the population contains enough free-riders, finding a cluster consisting only of conditional cooperators may be extremely rare. However, from the viewpoint of the ultimate origins of this behavior, conditional cooperation is a puzzle, as it has been shown (Boyd and Richerson 1988) that in an Iterative Multiplayer Prisoner's Dilemma (IMPD), the only conditionally cooperative, evolutionarily stable strategy prescribes cooperation only if all other group members cooperated in the previous period, which is not what is observed. Furthermore, for the case of moody conditional cooperation, the theoretical results based on a replicator dynamics approach showed that in groups with five or more people, the coexistence of moody conditional cooperators with free-riders (and possibly a few unconditional cooperators) is not possible (Chapter 3).

In view of these issues, we decided to investigate further this moody conditionally cooperative behavior by designing a series of experiments with human subjects playing an IMPD in groups of different sizes. Our starting research question is whether individuals actually behave in a moody conditionally cooperative manner or not, and whether the behavior of real subjects changes with the group size as suggested by the coexistence analysis presented in Chapter 3. Specifically, we looked at very long IMPDs on groups of 2, 3, 4 and 5 subjects.

5.1 Experimental setup

The experiment we study in this chapter was designed to match the game played in the network experiments, both ours and by other groups. Therefore, subjects played a multiplayer prisoner's dilemma in which they had to choose one action to interact with their opponents. For each opponent, they collected a payoff given by R (T) if their partner cooperated and they cooperated (defected). A total of 228 subjects participated in our experiments. Subjects were volunteers from the pool of the Economics Laboratory at the Department of Economics of Universidad Carlos III de Madrid. Participants interacted anonymously via computers at the Laboratory using software written with z-Tree (Fischbacher 2007). In all, 12 sessions were conducted in three consecutive days in April 2011. Each session lasted approximately 45 min on average. In each session, the subjects were paid a 10 euros show-up fee. Each subject's final score summed over all rounds was converted into dollars at an exchange rate that depended on the group size. The payoffs were set to $R = 7$ ECUs and $T = 10$ ECUs in all group sizes. The adjustment of the expected payoffs was then implemented through the conversion rate: The exchange rate was for 100 ECUs (Experimental Currency Units): 2 euros in the group of 2 players; 1.66 euros for 3-player groups; 1.33 euros for 4-player groups; and 1 euros for 5-player groups. Earnings in a typical session ranged from 5 to 15 euros. The instructions of the experiment, translated into English, are included in the Supporting Information. The Spanish original is also available upon request.

5.1.1 Computer intervention.

In half of the sessions, for all group sizes, there was a computer intervention in the decisions, in order to improve the statistics on the most cooperative contexts. The players were informed that:

> "Occasionally, and in completely random way, the computer can change your decision or that of the other player. The program does not report this change when it occurs. In such cases the payment is calculated as if the player concerned had actually taken the decision that the computer chose. The frequency with which this happens is low: your actions will remain unchanged for at least an 85% of the time."

Computer intervention was carried out in the following manner and only after the first 5 rounds took place unmodified: From round 5 to round 25, there is a 20% chance of having a computer intervention. In case there is such an intervention, the idea is to increase the number of highly cooperative contexts, and therefore every defection was turned to cooperation with probability $(N - N_{\rm coop})/(N - N_{\rm coop} + 1)$, N being the number of players in the group and $N_{\rm coop}$ being the number of players that cooperated in that round. After round 25, intervention was intended to increase the number of contexts that have appeared the smallest number of times up to that round. Let us call the number of cooperators in such context $N_{\rm wanted}$. If this number is higher than $N_{\rm coop}$, we change defection to cooperation with probability $(N_{\rm wanted} - N_{\rm coop})/(N - N_{\rm coop} + 1)$; otherwise, we change cooperation to defection with probability $(N_{\rm coop} - N_{\rm wanted})/(N_{\rm coop} + 1)$. This procedure allowed us to obtain better statistics for the highly cooperative contexts, and it was mild enough as not to influence the results, as the comparison of the results of the two treatments, with and without computer intervention, presented in Apendix D show.

5.2 Existence of moody conditional cooperation.

Let us begin reporting on the results of our experiment by looking at our first question, namely the existence of moody conditional cooperators and whether it depends on the group size or not. Figure 5.1 shows our results on this issue. We clearly observe that moody conditional cooperators are indeed present in all sizes, including groups of four and five players, at variance with the analysis in Chapter 3. However, this disagreement is not entirely surprising, since theoretical results for repeated games are notoriously sensitive to modeling assumptions: Thus, computational results on IMPD based on genetic algorithms (Yao and Darwen 1995) show that the evolution of cooperation in theoretical models depends very much on the implementation details. Therefore, the fact that our experimental observations do not agree with the predictions of a very specific model based on the replicator equation is something that can be expected. On the other hand, we observe only a few players using AllD (always defect) and even less players playing AllC (always cooperate), so what we are observing may be close to a homogeneous state consisting only of moody conditional cooperators, something that is possible even in large groups for certain parameters in Chapter 3. In any event, our results confirm beyond any doubt that in IMPD of sizes two through five the strategy of choice of players is moody conditional cooperation: Figure 5.1 shows very clearly the difference between the probability of cooperating after having cooperated or having defected, highlighting the importance of relating the current action with the one in the previous round. The plot also indicates that the probability of cooperation increases with the number of cooperators in the group in the previous round, for all group sizes. Cooperation when no one cooperated before is relatively large in groups of size 2 and lower for other group sizes (but similar among them). Interestingly, the increment in probability with increasing number of cooperators is similar for all groups.

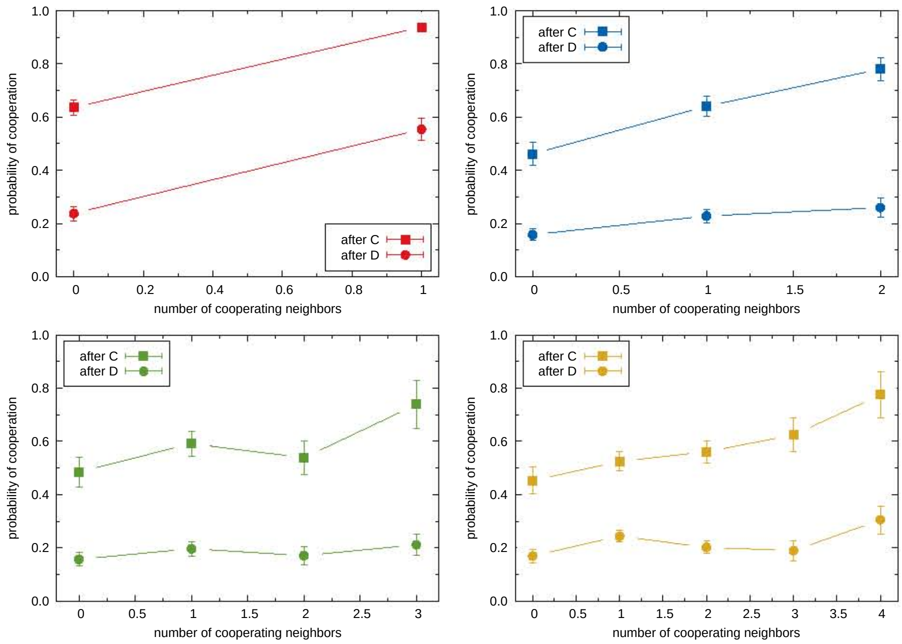


Figure 5.1: *Moody conditional cooperators.* Probability that an individual cooperates after having cooperated (squares) and after having defected (circles) in the previous round, for groups of 2 (top left), 3 (top right), 4 (bottom left) and 5 (bottom right) people. The error bars are the standard deviations of a binomial distribution, $\sqrt{p(1-p)/n)}$, where n is the number of samples, and p is probability of cooperation. Lines are only a guide to the eye.

5.3 Group size dependence of cooperation.

Let us now look at further insights provided by our experiments, beginning with the total cooperation level achieved in the different groups. The corresponding plot, showing the fraction of cooperative actions as a function of the iteration of the game, is presented in Figure 5.2. From these plots, it is immediately apparent that the results for groups of size two (i.e., pairwise interactions or usual 2×2 IPD) are very different for the observations on the rest of groups (sizes three and higher). Pairwise interactions show very high cooperation levels with an increasing trend, whereas for the rest of groups we find that cooperation decays from initially large values (around 60% or larger) much in the same way as in most Public Goods or networked IPD experiments. The fact that for groups with three subjects or more the cooperation level behaves in a similar manner is in agreement with earlier findings that the level of contributions to voluntary public goods does not depend significantly on the group size (Zelmer 2003). However, it is interesting to note in this regard that earlier results were obtained in public good games

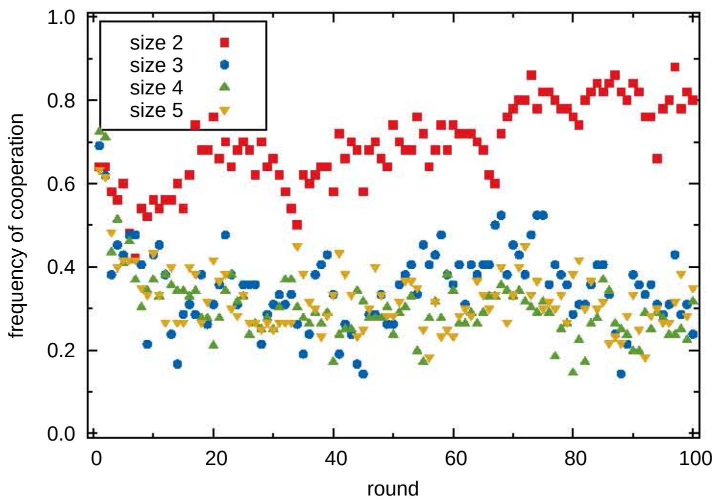


Figure 5.2: *Cooperation level in time.* Percentage of cooperation as a function of the round for groups of 2, 3, 4 and 5 people. Error bars correspond to the standard deviation of the observations.

with four or more subjects, and thus we are providing evidence that there is an abrupt change in behavior in going from a two-player IPD to IMPD or public goods games with three or more participants, i.e., we could say that *three is a crowd.*

5.4 The case of the two-player IPD.

The results for the pairwise IPD deserve a separate discussion as they offer several interesting insights. In our experiment, participants were not informed about the number of rounds of the game, although they were given an estimate of the time duration of the procedure, so they could realize that there would be a sizable number of rounds in any event. Therefore, the 'shadow of the future' effect is very present. As a consequence, pairwise IPD experiments show a large level of cooperation in agreement with the observations of (Dal Bó 2005), obtained for much shorter IMPDs (an expected length smaller than 6 rounds). Interestingly, the large length of our repeated game allows us to go beyond this observation: Indeed, if we compare our observations to those reported in (Kümmerli et al. 2007), who carried out experiments of length 12, we find an agreement for this initial part of the repeated game, as in both cases the cooperation level decreases with increasing round number. However, as the game continues in our experiment, we observe a clear trend towards increasing cooperation, punctuated by episodes of lower cooperation levels which are rapidly overcome.

To our knowledge, this increment in cooperation has never been reported before, and it is apparent from our results that it is specific of the pairwise PD, as for groups of size three and larger the level of cooperation is at least non-increasing. Note also that the cooperation at the first round is mostly independent of the group size (cf. Figure 5.2), so this difference among groups does not arise from the initial propensity to cooperate but is due to the behavior of the players as the repeated game progresses. This is in agreement with the type of moody conditionally cooperative strategy we found: the strategy parameters for pairwise PD, being clearly different from those of the larger groups, indicate that choosing cooperation is very likely if one cooperated in the previous step, while the probability to cooperate following a defection is relatively large, below but close to 0.5. It is important to realize at this point that this strategy is not the well known Tit-for-tat (TFT) (Axelrod and Hamilton 1981), as TFT does not depend on the player's own previous action, while Figure 5.1 strongly suggests that the previous action of the player affects her next choice. Our result is also in agreement with those reported by Fudenberg *et al.* (Fudenberg et al. 2012), who in their treatments without noise found that when a player has cooperated in all rounds, a defection by her partner is not immediately answered with defection in a 42% of the cases, a number that is roughly similar to the ones we obtain for our moody conditional cooperators (albeit the comparison must be taken with caution as the way to characterize the behavior in both experiments is not exactly the same).

5.5 Discussion

Because of their very long duration and the (small) group sizes considered, our experiments on IMPD, shed light on a number of important questions regarding the onset cooperation and to contribute towards a consistent picture of human behavior in social dilemmas. First, we report an experimental confirmation that cooperation actually increases for pairwise interactions, while decaying as usual for groups of 3 or more individuals. This is a striking result in so far as this increasing of cooperation in pairwise iterated Prisoner's Dilemmas (IPD) has not been reported previously, in spite of the fact that many experiments looked into this game (Ledyard 1995; Dal Bó 2005; Dal Bó and Frechette 2011). The reason it has never been observed before is that we are running experiments several times longer than the longest experiments ever carried out. We also observe an initial decay of the level of cooperation is consistent with previous experiments, but at some point (after the first 20 rounds) this level rises steadily. It remains an open question whether full cooperation would be reached in much longer experiments.

Furthermore, we have shown that most subjects behave in a moody conditionally cooperative manner, reciprocating the observed cooperation after a cooperative choice while changing into a non-reactive, mostly defector strategy following their own defection. As described in previous chapters, this had been observed earlier in lattice and network PD experiments. Our results now show that this type behavior is characteristic of the social dilemma and not of the number of partners or their (spatial) arrangement.

In addition, using a Generalized Linear Mixed Model (GLMM), described in Appendix E, we have confirmed an independent analysis that in order to understand the behavior of subjects in the experiments, it is enough to consider the actions of the previous round, as the information on the precedent round turned out to be not significant. The agreement of these additional results with the analysis presented above, supports our conclusion that we are correctly analyzing and understanding the experimental result. An additional insight provided by the GLMM concerns the universality and heterogeneity of the moody conditionally cooperative strategy: Remarkably, heterogeneity arises through the initial predisposition to cooperate, which turns out to be quite idiosyncratic; in contrast, the probability to reciprocate cooperation after having cooperated has an approximately linear dependence whose slope shows a much smaller degree of variability.

6

Discussion, conclusions and open problems

The goal of this thesis has been to explore and assess one of the mechanisms that are thought to be important for promoting cooperation, namely population structure. Many models have explored analytically and by simulation the effects of a network structure on the promotion of cooperation, particularly in the framework of the Prisoner's Dilemma, but the results of these models largely depend on details such as the type of spatial structure or the evolutionary dynamics. Alas, experimental work suitably designed to address this question was lacking.

In view of this, we designed an experiment to test the emergence of cooperation when humans play an Iterated Prisoner's Dilemma on a lattice whose size is reasonably comparable to that of simulations. The experiment was set up so as to favor cooperation as much as possible if subjects played as guessed by the theoretical models. Surprisingly or not, we found that the cooperation level declines to an asymptotic state with low but nonzero cooperation (around 20%). We also observed that the population was heterogeneous, consisting of a high percentage of defectors, a small fraction of cooperators, and a large group that we termed moody conditional cooperators, as their probabilities of cooperating depend on the player's previous action as well as the previous actions of their neighbors. Our findings indicate that both heterogeneity and a moody conditional cooperation strategy are required to understand the outcome of the experiment.

Inspired by the results of the experiment, we turned to theoretical analysis. We studied the coexistence of the three strategies observed in the experiment: cooperators, defectors and moody conditional cooperators in the multiplayer iterated Prisoner's Dilemma by means of the replicator dynamics. We considered groups with $n = 2$, 3, 4 and 5 players and computed the payoffs for every type of player as the limit of a Markov chain where the transition probabilities between actions were found from the corresponding strategies. We showed that for group sizes up to $n = 4$ there exists an interior equilibrium in which the three strategies coexist, the corresponding basin of attraction decreasing upon increasing the number of players, whereas we were not able to locate such a point for $n = 5$, and proved that it cannot exist in the infinite n limit.

Thus, our experimental findings on the networked games suggest that conditional cooperation may also depend on the previous action of the player, but at the same time we theoretically predict that such a behavior cannot coexist with players that always free ride or cooperate in groups with more than 5 people. Therefore, we designed experiments meant to test our theoretical analysis. We confirmed the existence of moody conditional cooperation and low cooperation level in the groups of size larger than two. Remarkably, we showed that the behavior of subjects in pairwise dilemmas is qualitatively different from the cases with more players, although this outcome can also be explained by a moody conditional cooperative strategy. Our experiments lasted 100 rounds, which allowed us to probe the long run regime in all cases, showing that for the pairwise dilemma, after an initial decay, cooperation increases significantly reaching values above 80%.

In order to gain further insight on our main question, we analyzed the data of the experiment by Traulsen et al. (2010) where human subjects played an iterated Prisoner's dilemma with each of their 4 nearest neighbors in a 4×4 lattice. The experiment explored two treatments: spatial, where players had fixed positions on the virtual lattice, and non spatial, in which players were reassigned to a random position of the lattice after each round. We analyzed the statistics of individual decisions and inferred in which way they can be matched with the typical models of evolutionary game theory. We find no difference in updating strategies between the two treatments. However, none of the updating strategies coincide with the most popular models of evolutionary game theory and they do not lead to the promotion of cooperation on lattices, as our simulations showed. The update rules fitted to the experiment do not promote cooperation in the spatial structure analyzed, even if the system were substantially larger.

6.1 Comparison of experiments

Having presented in the previous chapters our work on the emergence of cooperation in the presence of a network structure we believe that the best way to inform our conclusions is to compare the results of different iterated PD experiments with human subjects, performed in the laboratory on virtual networks. As it has been already mentioned in previous chapters, to our knowledge there are only a few experiments of this kind: Cassar (2007) on small random, small world and local networks, Kirchkamp and Nagel (2007) on local networks and group settings; Traulsen et al. (2010) on small regular lattices; ours described in Chapter 2, on medium size regular lattice; Suri and Watts (2011) on small networks of different kinds; and, finally, a very recent experiment by Gracia-Lázaro et al. (2012) on a large lattice and a large heterogeneous (fat-tailed)

	Plön	Madrid	Zaragoza
Size of the network	4×4	13×13	25×25
Neighborhood	von Neumann's	Moore's	von Neumann's
Number of rounds	25	47	52
Number of sessions	15	1	1
	C D	C D	C D
Payoff matrix	C: 3 0	C: 7 0	C: 7 0
	D: 4 1	D: 10 0	D: 10 0
# of different players	240	169	625
# of actions	6000	7943	32500

Table 6.1: Comparison of the experimental settings for the three experiments on lattices. A von Neumann's neighborhood consists of the four nearest neighbors in a square lattice, whereas a Moore's neighborhood comprises all eight surrounding neighbors in the same lattice.

network. Although the setup of all experiments is fairly similar, their details are rather different and the effect of those details on the results is not totally clear, and might be important. Here we present a meta-analysis of the experiments in order to extract the properties underlying all of them and that can be considered independent of the details of the setups in order to give as much generality and support as possible to our conclusions..

In what follows, we focus on the three experiments on regular lattices: the Plön experiment performed by Traulsen et al. (2010) and described in Chapter 4, the Madrid experiment described in Chapter 2, and the Zaragoza experiment performed by Gracia-Lázaro et al. (2012). The Zaragoza experiment was performed on two different types of networks: a 25 regular lattice with degree $k = 4$ and periodic boundary conditions (625 subjects), and a heterogeneous network with a fat-tailed degree distribution (604 subjects, the number of neighbors varied between $k = 2$ and $k = 16$). For each type of network they performed two treatments, an experiment treatment where the network was fixed during the experiment and a control treatment where the network was shuffled after every round. The treatments are analogous to those of the Madrid experiment and, as in Madrid, the treatments were performed in sequence: first the experimental treatment and then the control treatment with the same players. There are some differences in the way this treatments were performed. To begin with, the control treatments were performed either with players who already had experience from the experiment treatment (Madrid and Zaragoza), or with inexperienced players (Plön). Whether this has a measurable effect still needs to be clarified.

Aside from the differences in the control treatments, the experiments themselves had also differences of their own, which are more relevant to their comparison. In Table 6.1 we summarize these differences in the experimental setups. The first difference between the experiments is in the size of the virtual networks, ranging from the smallest one (4×4) in Plön's experiment to the largest one (25×25) of Zaragoza's experiment. The size of the network could have a significant influence on the promotion of cooperation, because the formation of clusters of cooperators (which is the known mechanism by which cooperation can be promoted) only has a chance if the networks are large enough. Furthermore, the local structure of the networks is different. In the Plön and Zaragoza experiments players had four nearest neighbors; in the Madrid experiment players played with the eight surrounding neighbors. Notice that this introduces a crucial difference in the local structure of the network, because the clustering coefficient of the lattice with four neighbors is zero, whereas in the network with 8 neighbors it is $3/7$. Since the clustering can significantly influence the promotion of cooperation (Roca et al. 2009b), this difference might be important. Another significant difference is the payoff matrix. In the Plön experiment, players played a strict Prisoner's dilemma (PD) where $P > S$, but in Madrid and Zaragoza the game played was a weak PD, where $P = S$ which is more favorable to cooperation: namely, in the weak PD, cooperating in a situation where everybody is defecting is not costly, because players will earn the same no matter what they do. On the contrary, in the strict PD, players earn more by defecting against defectors, therefore it is to be expected that they cooperate less. Finally, in the Plön experiment, 15 different sessions were performed, whereas in the Madrid and Zaragoza experiment, because of the size of the networks, this was not possible. However this should not influence the results significantly, since the statistics in each round is of the same order of magnitude for all experiments: 240 actions per round (the Plön experiment), 169 actions per round (Madrid experiment) and 625 actions per round (Zaragoza experiment).

6.1.1 General observables

Let us start the analysis with, the global cooperation level. Figure 6.1 shows the fraction of cooperative players in each round of the experiments. We see that in all three experiments the cooperation starts at rather large levels (between 55% and 70%), and subsequently declines rapidly and settles on a small but non zero level (between 15% and 35%). It is interesting that in the Plön experiment the initial cooperation level was the largest one, but the decline was the fastest and the final level was the smallest. Although there are small differences in the levels of cooperation, each of them is significantly lower than predicted by the theoretical models (Nowak and May 1992; Szabó and Fáth 2007; Roca et al. 2009b).

In Figure 6.2 (top) we show the distribution of players by their fractions of cooperative actions during the game. Two differences are noticeable. First, in the Madrid experiment we have a high number of pure defectors, which is missing in the other two experiments. The detail in common of the Plön and Zaragoza experiment and different for the Madrid experiment, which could be responsible for large number of defectors, is the number of neighbors in the lattice. However, the exact mechanism of how this could lead a sizeable number of players to become defectors is unclear. Furthermore, in the experiment on the groups (Chapter 5), we observed that beyond the pairwise Pris-

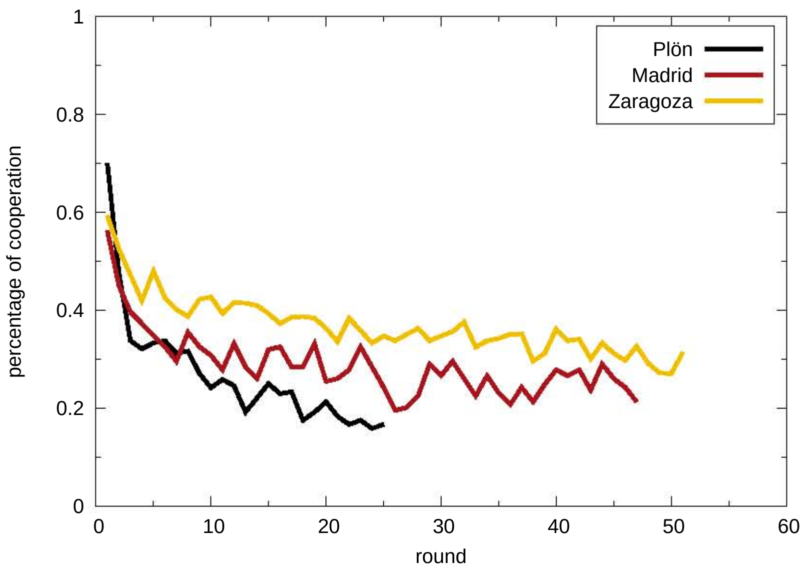


Figure 6.1: *Level of cooperation.* The fraction of players who cooperated in each round.

oner Dilemma the number of neighbors does not influence the behavior of the players, although the largest groups we tested there was of 5 players (4 neighbors). However, unlike the groups setting, on the lattice increasing the number of neighbors implies not merely increment of the number of individual you play with, it also implies an important change of the spatial structure, more specifically the clustering coefficient. Then again, all our results suggests that in the experiments spatial structure does not influence the global behavior of the players. Subsequently, the large number of pure defectors noticed in the Madrid experiment is still puzzling to us and further experiments need to address this issue. The other difference we can notice is that in the Plön experiment there are no players who cooperated more than 65% of the rounds, while the number of the players which cooperated more that 65% of the rounds is significant, albeit small, in the other two experiments. The reason for this could be that in Plön's experiment, the game was a strict PD, where cooperating is costly, and therefore players were less prone to cooperate.

Next, we present the distribution of players according to their earnings (Figure 6.2 (middle)). We notice that the distribution of earnings in Plön's experiment is slightly narrower. This could be the consequence of the size of the network or (more likely) the payoff matrix. Since there are not many players in the system, the earnings might be more correlated between themselves and therefore the distribution is narrower. However, the payoff matrix is also different in this experiment and as we have seen this could make players less prone to cooperate, which also narrows the earnings distribution. Finally, in Figure 6.2 (bottom) we present the correlation between the earnings

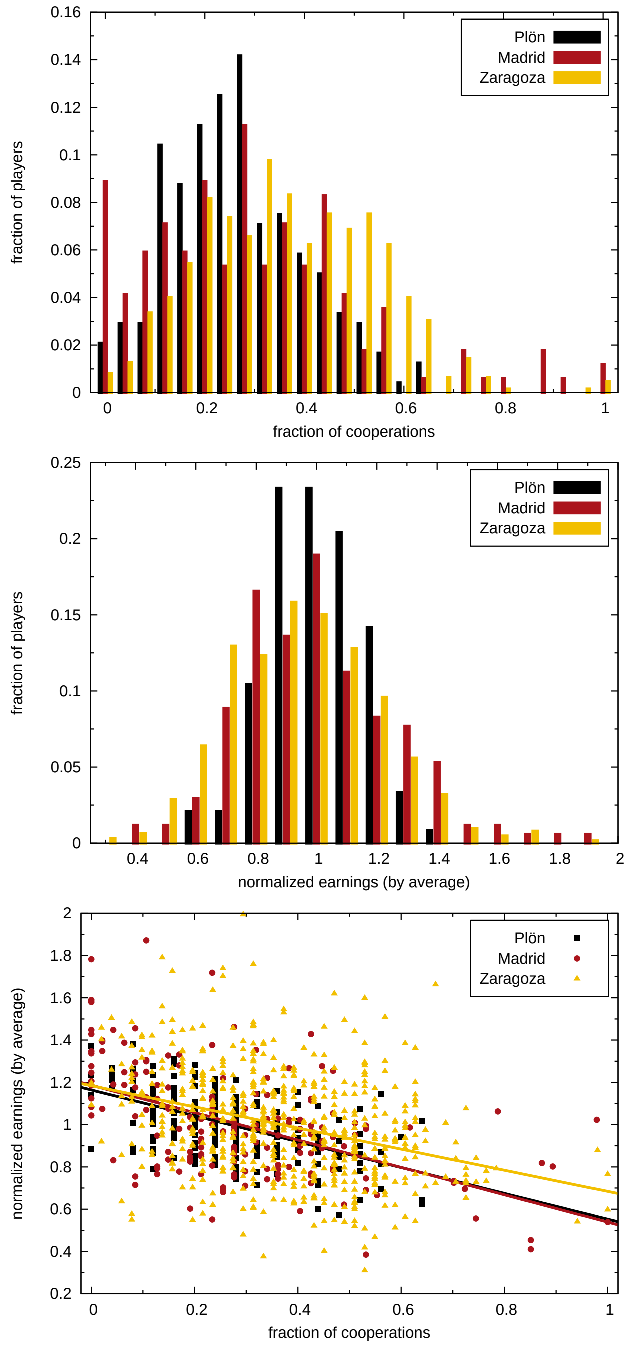


Figure 6.2: *Total earnings and cooperativeness.* (Top) The distribution of the players by the fraction of cooperation. (Middle) Distribution of player by their earnings normalized by the total earning of all players. (Bottom) Correlation of earnings and cooperativeness of the players. Each point represents one player. On the x-axis we plot the fraction of rounds the player make cooperates and on the y-axis we plot the earning normalized by the average earning of all the players. Slopes of the fits are following: Plön experiment $s = -0.61$, Madrid experiment $s = -0.64$ and Zaragoza experiment $s = -0.50$. In all three experiments the p-value is smaller than 0.001

and the cooperativeness of the players. In all three case there is a significant correlation (p-value < 0.001) and all three show the same trend: earnings and cooperation are anticorrelated.

6.1.2 Update strategies

Let us leave the level of global observables and analyze the data at the level of individual decisions. We will study the same three possible update strategies, beginning with unconditional imitation (Nowak and May 1992). First we calculate the probability that the player's action is the same as the action of the best player from the previous round. This number is rather high in all experiments (between 63% and 76%) and is presented on Figure 6.3 with vertical lines. However, as discussed in sections 2.3, 4.3 and A.2.2, since we have just two possible actions, more often than not the action of the focal player will coincide with the action of the best neighbor in the previous round just by chance. Therefore, we resorted again to a randomization test obtaining the distributions of inferred imitations presented on Figure 6.3. We notice that in the Madrid experiment the value of the inferred imitation from the experiment falls well within the distribution of the randomized samples (see also Figure A.6). Therefore, the observed level of imitation could simply be due to pure chance. In the Plön experiment, as we already saw, the experimental value is a bit off of the distribution. The p-value is 0.001, suggesting that there is significantly more imitation in the experiment than expected from the random neighborhood. In the Zaragoza experiment the experimental value is further off the distribution (p-value $< 10^{-4}$), showing even more clearly that the observed imitation is significantly different from the apparent imitation generated by chance. However, although statistically significant, the difference between the observed probability of imitation and random imitation is only 1% and the maximum value it reaches is around 75%. Now, this is an important result because it indicates that cooperation should not be sustained in any of the three cases. Indeed, for unconditional imitation to promote cooperation, as we found in the previous chapter, it has to be 100% imitation. If imitation is not unconditional then noise will prevent the formation of clusters and therefore the promotion of cooperation.

Let us now check if the players follow the Fermi rule (Szabó and Töke 1998). In Figure 6.4 we show the probability that the action changes depending of the payoff difference between the focal player and the best player who played the opposite action in the previous round. We see that in the Plön experiment the dependence is well fitted by the Fermi function. However in the other two experiments, although there is an increasing trend, the dependence is far from clear.

Finally, we studied the possibility that people behaved as moody conditional cooperators. In all the experiments there is a clear difference between the behavior after they cooperated and defected (Figure 6.5). After the player defected the probability of cooperating is slightly decreasing with the number of cooperators in the neighborhood, as after the player cooperated it increases. In the Plön experiment the behavior after the player cooperated is noticeably different than in the other two experiments, with the slope being considerably larger. The probability of cooperating if the player

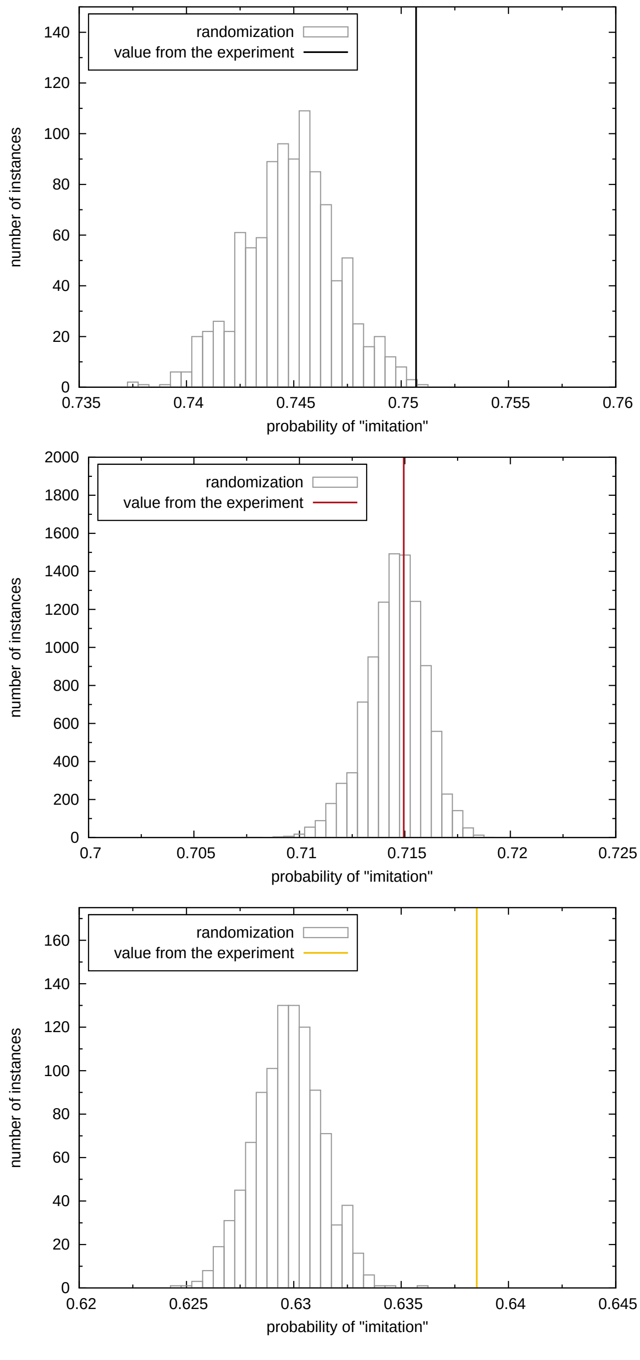


Figure 6.3: *Randomization test for unconditional imitation.* The distribution shows the results of randomizations and horizontal lines the value from the experiment. Top, the Plön experiment, middle Madrid experiment, bottom Zaragoza experiment payoff difference between the focal player and the best player of the opposite strategy. Notice the short range on the x-axes.

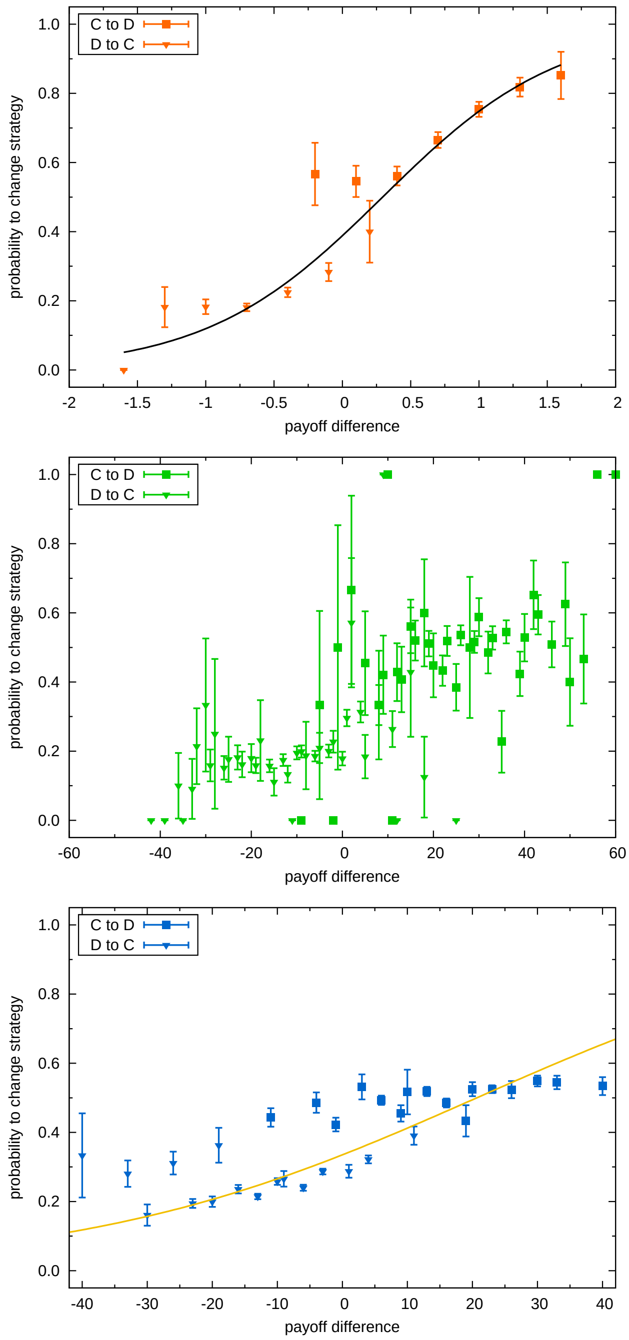


Figure 6.4: *Fermi rule.* On he x-axis is the difference of the payoffs between the focal player and the best of the players with the opposite action and on the y-axes is the probability to change the action in the next round. All error bars are the SDs of a binomial distribution, $\sqrt{p(1-p)/n}$, where n is the number of samples and p is the probability of changing the action. The results are presented separately for the players who changed from cooperation to defection and those who changed from defection to cooperation.

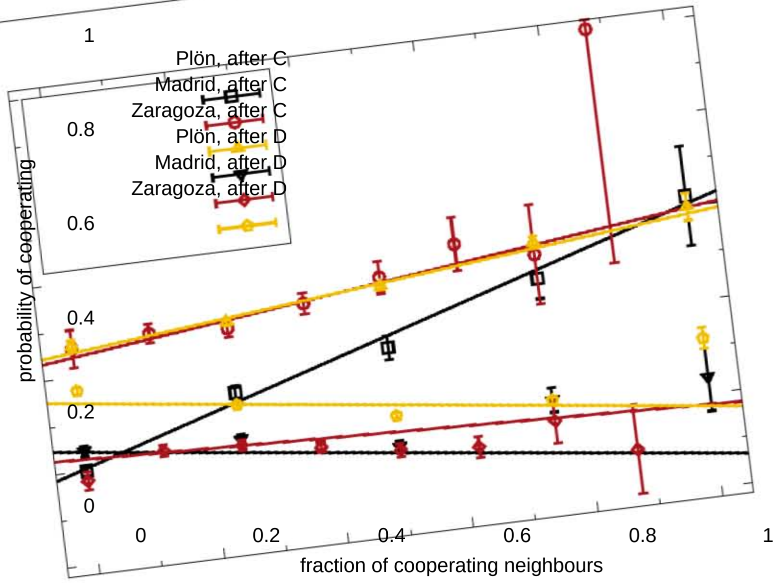


Figure 6.5: *Probability of cooperation in different contexts.* On the x-axis is the fraction of cooperating neighbors in the previous round and on the y-axis is the probability that the focal player will cooperate in the next round. We present separately the probabilities of cooperation after the focal player played C and he played D in the previous round. All error bars are the SDs of a binomial distribution, $\sqrt{p(1-p)/n}$, where n is the number of samples).

is surrounded by defectors is much lower in the Plön experiment than in the other two. This is probably a consequence of the different payoff matrices used in experiments. Since in Plön experiment players played a strict PD, cooperating while surrounded by defectors is costly. Therefore players in this experiment tend to cooperate less when they are surrounded with defectors than in the other two experiments.

6.2 Conclusions

Although there are some differences in the results between the three experiments, there are a few features that appear to be universal. The first one is the low but nonzero asymptotic level of cooperation. In spite of the fact that many theoretical models predict the promotion of cooperation by a mechanism of network reciprocity, it is clear that such a promotion was not observed in any of the experiments analyzed here. Additionally, in other known experiments (Cassar 2007; Kirchkamp and Nagel 2007) the level of cooperation is also low. The distributions of cooperation and earnings look similar, but details like the payoffs and the number of neighbors in the lattice do influence these distributions, most notably the percentage of cooperation. In all experiments there is a significant negative correlation between the number of cooperations and the earnings, the slopes being similar to each other.

The low level of cooperation is in line with the fact that we do not observe full unconditional imitation. Although imitation looks significant in some experiments, it is still low and far from being deterministic. However, it appears that the players are somehow influenced by the payoff difference, because there is a growing trend in the dependence of the probability to change the action with the payoff difference. This notwithstanding, the precise nature of the dependence and its correlation with the actions of the neighbors needs to be investigated more thoroughly. Moody conditional cooperators appear in all three experiments. It is clear that the behavior of the players depends on their own previous action in all of them, although the specific slopes vary from experiment to experiment and could be dependent on the payoff matrix. Interestingly, in the experiment by Cassar (2007) it was also shown that the behavior of the players is significantly correlated with their previous action.

Based on all the experiments performed by now we can safely state that a population structure given by a lattice does not promote cooperation. The results by Gracia-Lázaro et al. (2012) point to the fact that heterogeneous networks do not promote cooperation either, but being only one experiment (the evidence of Cassar (2007) is on system too small to be considered really complex) it may be too early to insist on this claim. Another result that arises from the the three experiments is that human player do not use the update mechanisms postulated by the theoretical models which lead to high cooperation (albeit there is some evidence in favor of unconditional imitation, only not purely deterministic). Consequently the cooperation level in the laboratory experiment with human subject on the networks is dramatically lower than predicted by theoretical models. Additionally we distinctly show that the behavior of the players depends on both their own previous action and the previous actions of their neighbors.

6.3 Open questions

The first question that our results and the ensuing discussion leaves open regards the update strategies. We have considered above a few update strategies that fit well with the experimental data we analyzed. However, which one people actually use is not absolutely clear. It is well established that people are influenced by the actions of other players as well as their own action in the previous round. On the other hand the influence of the payoff difference, although quite clear in the Plön experiment is not so pronounced in the other two. Additionally, the payoff differences and the actions of the players are not independent, and therefore the trend we see in the dependence with the payoff difference could be the consequence of the dependence on the actions. In addition, in this chapter we sketched some explanations for the differences noticed in different experiments. However, from only three experiments it is difficult to draw certain conclusions regarding all their aspects, more so in view of the differences we summarized above. Therefore, more experiments specially designed to clarify the influence of the payoff matrix, of the neighborhood, etc., are needed.

Another effect whose existence and importance is not clear in the experimental data is learning. In our lattice experiment the behavior of players is practically the same in the first and last 10 rounds, which led us to the conclusion that there was no learning during the single experimental treatment. Nonetheless, there is a small difference between the behavior or the players after they themselves defected, which was also noticed in the Zaragoza experiment (Gracia-Lázaro et al. 2012). This difference is attributed to learning in this reference, but future work should address properly this question in order to clarify the existence of learning in the experiments.

In none of the experiments on structured population a promotion of cooperation is observed. Notwithstanding, adding some other mechanism to the network structure might increase the cooperation. There are several mechanisms which could be added to the experimental setup in order to enhance it. Punishment has proved to be effective in enhancing the cooperation in pairwise Prisoners Dilemmas (Fehr and Gächter 2000). However, the influence of the punishment on a Multiplayer Prisoner's Dilemma in a network is still unknown. Also, players could be allowed to chose who they play with, breaking the connections with the players they do not wish to continue interacting with and forming new connections with other players. Numerous models (Aktipis 2004; Santos et al. 2006a; Helbing and Yu 2008; Meloni et al. 2009; Van Segbroeck et al. 2009; Van Segbroeck et al. 2010) show that this could be an mechanism to promote cooperation and some recent experiments suggest that there is indeed an effect on the global level of cooperation (Fehl et al. 2011; Rand et al. 2011). Furthermore, reputation could be added as a parameter of the decision making process. For example, it has been shown that in public goods experiments cooperation increases substantially if players can invest publicly or if the most or least cooperative individuals revealed , thus gaining or loosing social reputation (Milinski et al. 2006; Jacquet et al. 2011). The same mechanism should be tested in the spatial settings. Each player could be identified within the game (but still be anonymous outside of it) and this information could be provided to everybody. The question is, whether the fact that all player's actions are public knowledge will influence the behavior of the player and whether the other players will consider this reputation in their behavior toward them.

It would be interesting to see if different countries, age groups, genders, etc., have different behaviors in the experiments. The Ultimatum game experiment performed in different ethnic groups (Henrich et al. 2001), shows that there is a wide variety of behaviors depending on the group's culture. Another experiment showed that males and groups of mixed males and females act similarly, whereas groups consisting only of females show significantly higher cooperativeness (Kümmerli et al. 2007). Aguiar et al. (2009) show that women are expected to be more generous by women, which suggests that the behavior resulting from these expectations will also be different. In the Zaragoza experiment, females were observed to cooperate slightly more than males although this still requires proper statistical analysis. It would be good to check whether population structure has any influence on this effect. In the Madrid experiment there were many players that were relatively old, between 20 and 25 years and well advanced in their careers. This might be a reason that we observe so many pure defectors there.

Therefore, it would be interesting to do experiments by age and see what kind of influence it has on the results.

An important consequence of the experimental results presented here is that they should be used for developing new theoretical models. There is still no evolutionary explanation for the moody conditional cooperators we observe in our experiments. On the other hand, in Chapter 3 we showed that the coexistence between moody conditional cooperators, cooperators and defectors is only possible for the groups of very small size. One of the possible explanations is that players do not act rationally, but certainly this is still an open question that needs new ideas and further research before it can be given a satisfactory answer.

A

Additional material on the lattice experiment (Chapter 2)

A.1 Experimental setup.

A.1.1 Volunteer recruitment and treatment

The experiment was carried out with volunteers chosen among students of the engineering campus of Universidad Carlos III in Leganés (Madrid, Spain). Following a call for participation, we received about 500 applications, among which we selected 225, with preference for the youngest ones and keeping a fifty-fifty ratio of male to female. On the day of the experiment, 178 volunteers showed up, and we kept 169 so that we could arrange them in a square lattice with periodic boundary conditions by discarding the 9 latest arrivals —who were paid their 10 euros show-up fee and dismissed. The 169 volunteers were directed to 11 computer rooms in two adjacent buildings, previously prepared by setting up cardboard panels between posts so that no participant could look at her physical neighbors (which needed not be their actual neighbors in the game). They received directions in paper and also went through a tutorial on the screen, including questions to check their understanding of the game. When everybody had gone through the tutorial, the experiment began, lasting for approximately an hour and a half. At the end of the experiments volunteers were presented a small questionnaire to fill in, which will be discussed in Section A.4. Immediately after, all participants received their earnings and their 10 euros show-up fee. Total earnings in the experiment ranged from 18 to 45 euros.

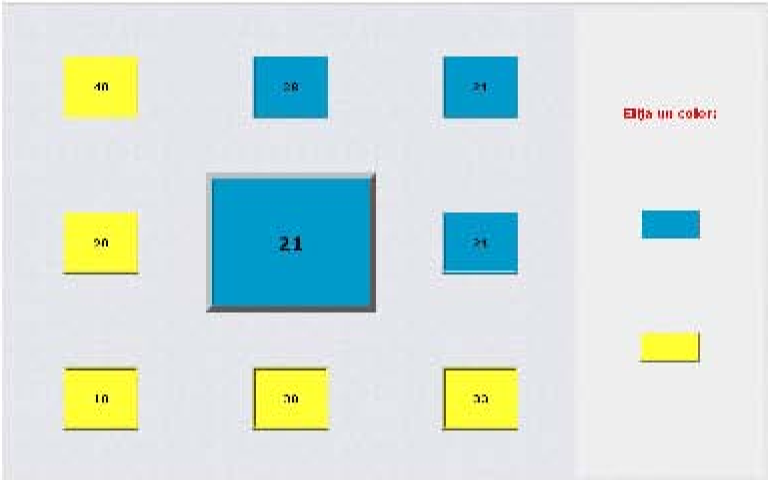


Figure A.1: *Information given in the experimental setup.* After every round players saw the information in the screen as depicted here. Given are the strategies of the player's neighbors (color coded) and their earnings in the previous round (in cents of a euro). To the right the player had two clickable buttons with the two actions to choose from for the next round. The label "Elija un color:" is Spanish for "Choose a color:".

A.1.2 On-line tutorial for players

The following is a translation of the Spanish original (available upon request).

Page 1: This is an experiment designed to study how individuals make decisions.
You are not expected to behave in any particular way.
Whatever you do will determine the amount of money you can earn.
You have a written version of this directions which you can check at any stage of the experiment.
Please keep in silence during the experiment. If you need help, raise your hand and wait to be attended.

Page 2: **DIRECTIONS TO PARTICIPATE IN THE EXPERIMENT**
This experiment consists of THREE (3) parts.
Each part consists of an undetermined number of ROUNDS (approximately between 40 and 60, by there might be more or less).
The experiment will last about 2 hours. In each part you will be able to earn different amounts of money, depending on the decisions that you and the rest of participants make every round.
Your total earning in this experiment is the accumulated earnings in all the three parts, plus a 10 euros showup fee.

Page 3: **A ROUND**

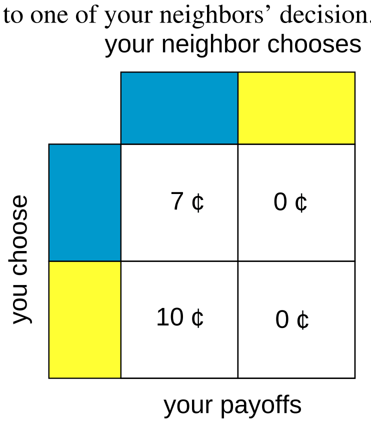
Each ROUND you will be placed in a nod of a virtual NETWORK.
In this network you will be linked to EIGHT (8) people, whom we shall refer to as “neighbors”.
All participants will be in the same situations, i.e., each of your 8 neighbors will have, in her turn, 8 neighbors, one of whom will be you.
You will never know who your neighbors are, and nobody will know if you are her neighbor either.
The network is virtual. People around you in the room are not necessarily your neighbors.

Page 4: **DECISION TO MAKE EVERY ROUND**

Every round, each of the participants must choose a color: BLUE or YELLOW.
To choose a color you just have to click a button appearing in the screen.
Each time you choose a color (either blue or yellow) you will earn an amount of money which will depend on yours and your 8 neighbors’ choices.
If you choose BLUE and your neighbor also chooses BLUE, you receive 7 cents each.
If you choose BLUE and your neighbor chooses YELLOW, you receive 0 cents and your neighbor 10 cents.
If you choose YELLOW and your neighbor also chooses YELLOW, you receive 0 cents each.
If you choose YELLOW and your neighbor chooses BLUE, you receive 10 cents and your neighbor 0 cents.
These rules are the same for all participants.

Page 5: **POSSIBLE PAYOFFS PER NEIGHBOR**

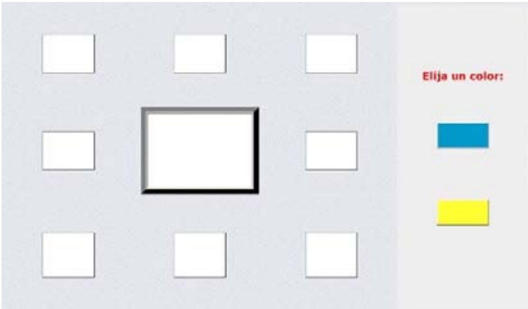
In the following table each row corresponds to the decision you can make and each column correspond to one of your neighbors’ decision.



Consider that:

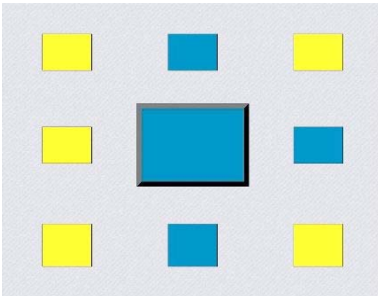
- you and each of your neighbors will globally earn more if you both choose BLUE (7 cents you / 7 cents your neighbor);
- you will earn more if you choose YELLOW and your neighbor chooses BLUE (10 cents you / 0 cents your neighbor);
- but if both you and your neighbor choose YELLOW you both will earn less (0 cents you / 0 cents your neighbor) than if you both chose BLUE.

Page 6: This is the screen you will be seeing during the experiment

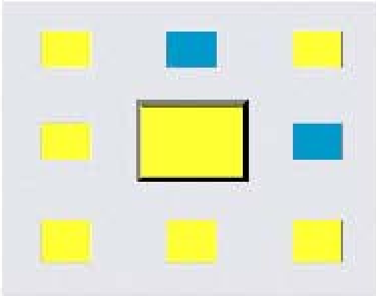


The central square represents you, and the surrounding squares represent your virtual neighbors in that round.
On the right of the screen you will see two buttons: BLUE and YELLOW.
Each round you must choose one of them clicking the corresponding button.

Page 7: These are some examples of what you could earn in a round:



Example 1: Imagine you choose BLUE, 3 of your neighbors choose BLUE and 5 choose YELLOW. In that round you will earn $3 \times 7 + 5 \times 0 = 21$ cents.



Example 2: In another round you choose YELLOW, 2 of your neighbors choose BLUE and 6 choose YELLOW. In that round you will earn $2 \times 10 + 6 \times 0 = 20$ cents.

Page 8: (Some multiple-choice tests are included in order to check whether the player has understood the rules of the game).

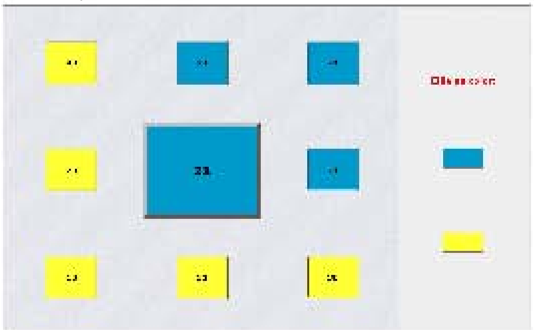
Page 9: (The correct answers to the tests are provided).

Page 10: **ROUND ITERATION**

Remember that each part will consists of an undetermined number of rounds. Each round you will have up to 30 seconds to choose a color. After these 30 seconds, if you didn't choose, the system will choose for you. Whatever happens will not affect the behavior of the system in the next rounds: you will be able to make your subsequent choices normally. (Don't worry: 30 seconds are more than enough to make a choice).

The round will not end until all participants have made their choice.

At the end of each round you will see a screen like this one:



The central square represents your choice and your earning in this round. The surrounding squares represent your 8 neighbors' choices and their respective earnings in that round.

Immediately after finishing a round there will be a new one, and then another one, and so on until you see a screen warning you about the end of that part of the experiment.

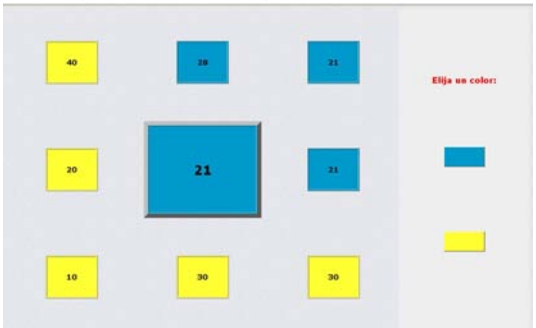
Page 11: **PART I OF THE EXPERIMENT**
In this part the system will randomly assign each participant to a given node of the virtual network.
This place will be kept fixed until this part ends.
This means that you will be interacting with **the same 8 neighbors** during all that part.
Remember that each round you must choose a color.
When this part finishes, you will be notified of it and will see the directions for the next part.
(Part I begins.)

Page 12: Part I of the experiment has finished.
Please, keep in silence.
Part II will start in a few seconds.

Page 13: **PART II OF THE EXPERIMENT**
In this part, **before each round begins,** every participant will be moved to a **new** random node of the virtual network. Therefore, in general **you will have 8 new neighbors every round.**
This means that **the node you are in will be changing along the experiment.**
Thus you will NOT be linked all rounds to the same 8 neighbors.

Page 14: The rules to make decisions every round are the same as in Part I.
The only thing that is different is that your neighbors will most likely not be the same every round.
Remember:

- Every round you have 30 seconds to make a choice.
- The round finishes only when all participants have made their decisions.
- At the end of each round you will be seeing a screen like this one:



The central square represents your choice and your earning in this round. The surrounding squares represent your 8 neighbors' choices and their respective earnings **in that round.**

(Part II begins.)

Page 15: Part II of the experiment has finished.
Please, keep in silence.
Part III will start in a few seconds.

Page 16: **PART III OF THE EXPERIMENT**
This part is the same as Part I.
Again, **before** it starts, the system will randomly place every participant in a given node of the virtual network.
That place will not change during the whole part.
This means that you will be interacting with the **same 8 neighbors** during all rounds.
Notice that your 8 neighbors in this part need NOT be the same as those of Part I.
(Part III begins.)

Page 17: Part III of the experiment has finished.
Please, keep in silence.
The experiment has not finished yet.
You have to answer the following questionnaire.
Please, answer ALL questions in the questionnaire that you will be shown immediately.
(The questionnaire was shown and afterward they were notified how much they had earned and were to go to get paid.)

A.1.3 Synchronous play and automatic actions

The experiment assumes synchronous play, thus we had to make sure that every round ended in a certain amount of time. This playing time was set to 30 seconds, which was checked during the testing phase of the programs to be enough to make a decision, while at the same time not too long to make the experiment boring to fast players. If a player did not choose an action within these 30 seconds, the computer made the decision instead. This automatic decision was randomly chosen to be the player's previous action 80% of the times and the opposite action 20% of the times. We chose this protocol after testing several ones in simulations. We run simulations in lattices of several sizes, including 13×13, using two different update rules: imitate-the-best and proportional updating. At the same time, we included a fraction of players (up to 15%) who played with a different update rule. We tested the one we finally chose, along with similar ones with different probabilities of copying the previous action. We also tested several other rules. Our finding was that a fraction below 10% of these "singular" players can hardly affect the results whichever their update rule. So we decided to choose the 80–20 rule as the one which could pass more unnoticed to other players when confronted to it. Anyway, for the reliability of the experiment it is important that a huge majority of actions were actually played by players, not by the computer. Figure A.2 shows the fraction of players who actually played in each round. We can see that in more than 90% of rounds no more than 4 of the choices were made by the

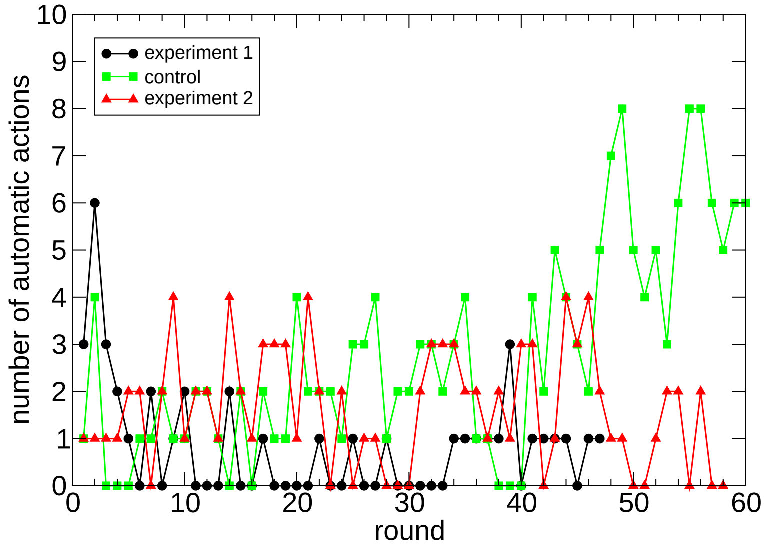


Figure A.2: *Number of automatic actions.* Number of players whose action was taken by the computer, not by themselves, in each round.

computer. The largest number of automatic actions occurred at the end of the control part, but even then their number never goes beyond 8.

A.2 Additional material about the experimental results

A.2.1 Earnings vs. cooperation

Although the average total earnings of players in all parts of the experiment are the same (around 8 euros), the distribution is clearly much narrower in the control part than in the first and third parts. In Figure A.3 each point represents one player's earnings (on the Y-axis) vs. the fraction of times she cooperated (on the X-axis). In general, we observe a slight decrease of earnings as cooperation increases, although the slope is so small that we can conclude that there is very little (if any) correlation between the fraction of times a player cooperated and how much she earned.

A.2.2 Imitate-the-best

Nowak and May's simulations on a lattice (Nowak and May 1992) employed a strategy updating known as imitate-the-best (or unconditional imitation) in the literature. This update rule makes each player copy the action of the most successful of her neighbors (including herself) in the previous round. Specifically, all players start with random actions; then each of them finds who among her 8 neighbors and herself had the highest payoff in the previous round and plays the action of this most successful player. This

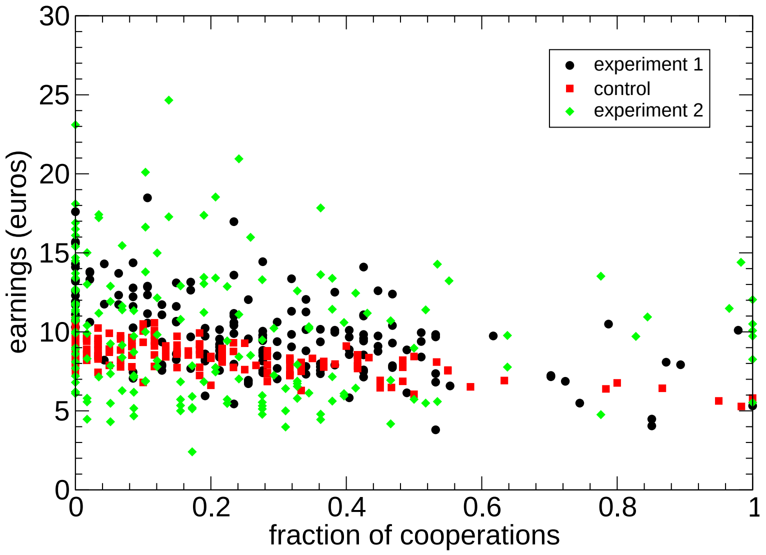


Figure A.3: *Players total earnings vs. fraction of cooperation.* Each point represents one player. On the X-axis we plot the fraction of times that the player cooperated and on the Y-axis the player's total earnings in that part.

update rule has been shown to be very efficient in enhancing cooperation in lattices (Roca et al. 2009b).

First we analyzed how Nowak and May's results changed when simulations were carried out on a lattice of the same size as our experiments. We reproduced their simulations on a 13×13 lattice, and the results are collected in Fig. A.4. We have computed the cooperation level as an unrestricted average (with values above 0.5) and also as an average that excludes the realizations that end up in full defection —which, as can be seen from the right panel of Fig. A.4, grow with the temptation parameter. For values similar to the ones we are using, around 80% of the simulations evolve to full defection, whereas the others yield an average cooperation level well above 60 %. None of these asymptotic behaviors has been observed in the experiment.

In order to check whether players use imitate-the-best for updating their choices we determined how many times players copied what the most successful player in their neighborhood did in the previous round. Figure A.5 shows that the number of times a player imitated the best action per round is almost the same as the number of Ds played in that round. In other words, since D is almost always the most successful action, a player playing D can be interpreted as if she is imitating the best action in the neighborhood, even though this may just be due to coincidence (there are only two actions). Compare this figure with Figure 1 of Traulsen et al. (2010).

In view of these results, we have devised the following null model. We generate 1000 shuffling of the players in each experiment. Players maintain always the same se-

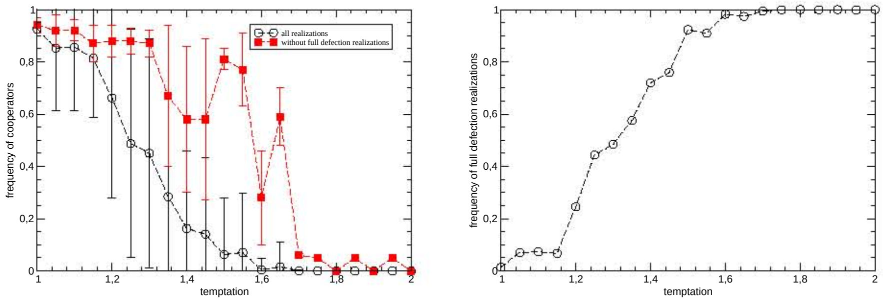


Figure A.4: *Simulations of a 13×13 lattice with imitate-the-best as update rule for the strategies.* Left: fraction of cooperators for a weak PD on a 13×13 square lattice with Moore neighborhood vs the value of the temptation parameter (the payoff to a defector facing a cooperator). Empty circles represent averages over 1000 realizations; full squares are averages restricted to those realizations that did not end up in a fully defective state. Right: frequency of realizations that finish in full defection.

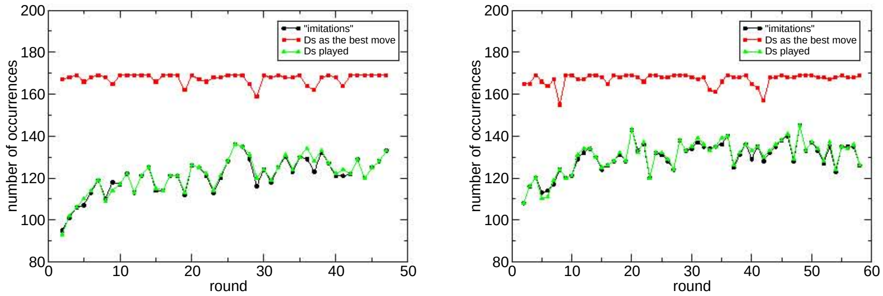


Figure A.5: *Test of imitate-the-best per round.* Results for experiment 1 are in the left panel, whereas those for experiment 2 appear in the right panel. In both cases we represent, for every round, the number of players playing D when D was the best action in the neighborhood in the past round (black line); the number of times D was the best action in the neighborhood in the past round (red line); and the number of Ds played in that round (green line). Since almost always D is the best past action in the neighborhood, the black and green lines nearly coincide, suggesting that "imitation" can be due just to coincidence. (Note the short range of the vertical axis.)

quence of actions, but every shuffling they play against different neighbors. Imitations now are purely due to coincidence because players are playing against people who were not their neighbors in the actual experiment. This way we generate a probability distribution of fractions of imitation purely due to chance, which we compare to the actual values of the fractions of imitation obtained in the true experiments. The results are shown in Figure A.6 and strongly support our conclusion that true imitation cannot be distinguished from accidental imitation in our experiment.

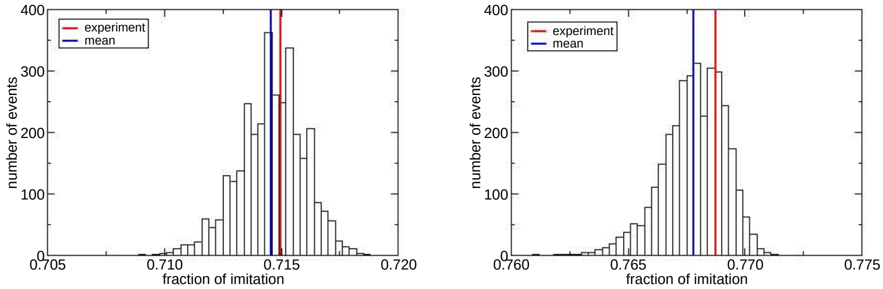


Figure A.6: *Null model for imitate-the-best.* Empirical histograms for the occurrence of a given fraction of imitation by pure chance. Distributions are obtained from 1000 random shuffling of the players, while keeping their sequences of actions. The blue vertical line represents the mean of the distribution and the red one is the result of the actual experiment. The experimental result is thus compatible with the null hypothesis —namely that imitation merely occurs by coincidence of actions. (Note the very short range of the horizontal axis.)

A.2.3 Analysis of strategies

The aim of this analysis is not literally to uncover which precise strategies players used during the game, but rather to elicit a plausible pattern of behavior that roughly explains the results we observe in the experiment. To achieve this we need to make a couple of simplifying assumptions. First of all, we assume that players' decisions were more influenced by the fraction of cooperators they observed than by their neighbors' payoffs. We did not ask explicitly for this in the questionnaires, but in their explanations of what strategies they had used they almost always speak about the actions of their neighbors and hardly ever mention their payoffs. Secondly, we assume that these decisions were based only on what occurred in the previous round. This is quite a drastic simplification because people have longer term memory and nothing precluded players from considering some kind of time-average of the full past history. However, it is plausible to assume that the last round has a much higher weight into players' decisions, and besides we cannot make more elaborated assumptions given the data we have. Already assuming that players' decisions depend only on what players did in the previous round and on how many cooperators they observed in their neighborhoods, leads us to devise a model with 18 parameters (the probabilities of playing C, if C or D was played in the previous round and there were $k = 0, 1, \ldots, 8$ cooperators in the neighborhood). There is not enough statistics to determine 18 parameters (some contexts, like $k = 8$ cooperators, never happened, and some others only happened once or twice in the experiments); certainly not to study the individual response of every player, but also to figure out the aggregated response.

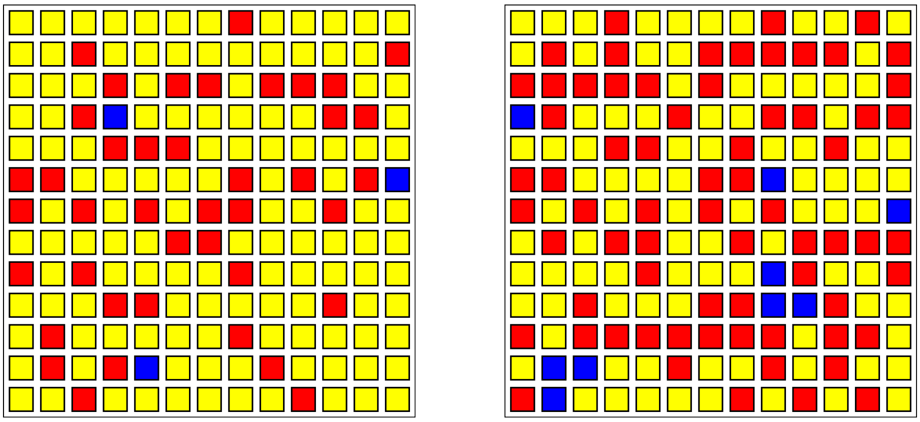


Figure A.7: *Position of players in the lattice according to the strategy they used.* Left panel: experiment 1; right panel: experiment 2. Color code: Red, pure or mostly defectors; blue, pure or mostly cooperators; yellow, conditional cooperators. Notice the two clusters of cooperators that were formed in experiment 2.

A.2.4 Spatial distribution of strategies

Figure A.7 shows the spatial location of players in the lattice according to the strategy they used (codified in color). In experiment 2 we can observe the formation of two clusters of three cooperators each. In order to check their statistical significance we computed the mean number of neighbors of the same kind for the different strategies. The results appear in Table 2.2. This table shows that indeed the clusters of cooperators in experiment 2 are very unlikely to have formed by chance in a random arrangement of players. It also reveals that defectors tend to anti-cluster in both experiments. The existence of this incipient clustering of cooperators is consistent with the fact that we do observe more cooperators in experiment 2 than in experiment 1, suggesting that clustering might be fostering cooperation in this experiment. The anti-clustering of defectors is also an indication of some spatial arrangement of strategies. Nevertheless experiments on much larger lattices would be needed in order to ascertain whether the spatial structure is yielding some ordering of the strategies.

A.3 Models

The classification of players' strategies led us to devise two models to describe the results obtained in the experiments. In the so-called homogeneous model all players play according to the conditional probabilities defined in Figure 2.7. In the heterogeneous model a mixture of five categories of players is made (pure and mostly cooperators, pure and mostly defectors, and conditional cooperators), taken in the same numbers as they appear in the experiment (see Table 2.1 in main text). In both cases the initial

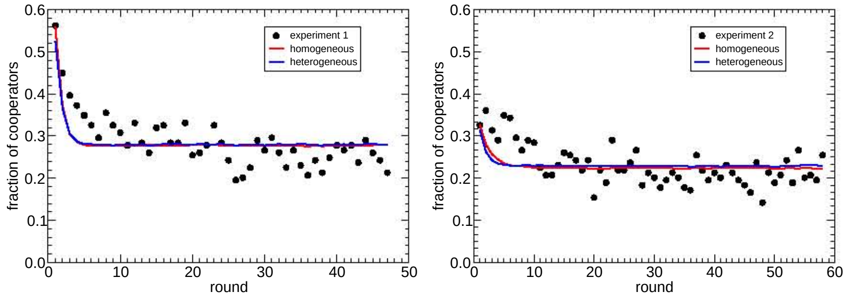


Figure A.8: *Cooperation levels.* Level of cooperation (fraction of players who are cooperating) in each round for experiment 1 (left) and experiment 2 (right). The experimental results are plotted together with the results of simulations with the homogeneous and the heterogeneous models, averaged over 1000 realizations.

probability of cooperating is taken so as to reproduce the initial cooperation level of the experiment.

A.3.1 Cooperation levels

Figure A.8 shows the cooperation levels reached by the simulations of the two models. Except for having a shorter transient —which indicates that players may take some time before they adjust their strategies, especially in experiment 1— the level of cooperation reached by both models is compatible with the experimental results.

A.4 Questionnaires

At the end of the experiments volunteers were presented a small questionnaire to fill in. The list of questions (translated into English) was the following:

1. Describe briefly how you made your decisions in part I (Experiment 1).
2. Describe briefly how you made your decisions in part II (Control).
3. Describe briefly how you made your decisions in part III (Experiment 2).
4. Did you take into account your neighbors' actions?
5. Is something in the experiment familiar to you? (yes/no).
6. If so, please point out what it reminds you of.
7. If you want to make any comment, please do so below.

The first three questions have a clear motivation, namely to see whether (possibly some) players did have a strategy to decide on their actions. Question 4 was intended to check whether players decided on their own or did look at their environment, because only in this last case imitative or conditionally cooperative strategies make any sense. Questions 5 and 6 focused on the possibility that some of the players recognized the game as a Prisoner's Dilemma because they had a prior knowledge of the basics of game theory. The final question just allowed them to enter any additional comment they would like to make. We did not carry out a more detailed questionnaire to avoid the risk of many players' leaving it blank (the whole experiment was already very long).

Regarding the first three questions, about the manner in which players made their decisions, we want to stress that the answers should not be taken too literally because sometimes people have a biased impression of what they really did. For instance, some people claim to have played randomly, however humans are known to be very poor random generators (Bains 2007). With this caveat in mind, it is nevertheless sensible, once we have a behavioral model for the players, to test its predictions against the players' responses in order to see if they are correlated. We considered their answers and compared with the results we obtained independently from analyzing their actions during the experiment. We found that, except in 7 cases, the perception the players had of their strategies agreed with our results. Not surprisingly, almost all the players which we identified as cooperators or defectors declared themselves to be so in their answers. Therefore, the really informative part arises from the answers of the players that did not fit clearly in these categories. In this respect, we found that most players defined themselves as either exploiters (cooperated to induce neighbors to cooperate only to defect on them and reap the benefit) or disappointed (wanted to achieve global cooperation but ended up doing as their defecting neighbors). Smaller groups included players who answered that they played at random, as we already mentioned, or that do not clarify what their strategy was, either because they did not answer or because their explanation was unclear. Keeping in mind that this is a very qualitative analysis of the results, and that not much more can be said quantitatively, the answers to the questionnaire support our general picture in terms of defectors, cooperators, and conditional cooperators. This is further confirmed by the answer to the fourth question, i.e. whether they took into account the actions of their neighbors: Only 25 out of 169 answered that they did not consider their neighbors' actions to make their own decisions. Interestingly, only 17 specify that they did not look at their neighbors' actions in the control part, where due to the shuffling after every round information about previous behavior became irrelevant. This opens the possibility that during the control part the players played according to something like a mean field approach, taking the information about their previous neighbors as a predictor for what they were going to find in their new location. Our data are certainly not enough to pursue further this question. Finally, as for the question about familiarity with the experiment, only 7 people answered affirmatively, mentioning the Prisoner's Dilemma or game theory in general.

B

User's manual for the PDexp software

PDexp is a software for conducting Prisoner's Dilemma (PD) experiments with human subjects on a large square lattice, as described in Chapter 2. The software has been developed by J. Grujić at Grupo Interdisciplinar de Sistemas Complejos at Universidad Carlos III de Madrid.

B.1 About

In the experiment, volunteers played a 2×2 PD game with each of their eight neighbors (Moore neighborhood) taking only one action, either to cooperate (C) or to defect (D), the action being the same against all the opponents. The resulting payoff was calculated by adding all eight interaction payoffs. Payoffs of the PD game were set to be 7 cents of a euro for mutual cooperation, 10 cents for a defector facing a cooperator, and 0 cents for any player facing a defector (weak PD). With this choice (a cooperator and a defector receive the same payoff against a defector) defection is not a risk dominant strategy, which enhances the possibility that cooperation emerges. The payoffs are given as parameters of the program when started and therefore the experiment can be performed with other payoffs. To avoid framing effects, the two actions were always referred to in terms of colors (blue for C and yellow for D), and the game was never referred to as PD in the material handed to the volunteers. This notwithstanding, players were properly informed of the consequences of choosing each action, and some examples were given to them in the introduction. After every round players were given the information of the actions taken by their neighbors and their corresponding payoffs.

The full experiment consisted of three parts: experiment 1, control, and experiment 2. In experiment 1 players remained at the same positions in the lattice with the same neighbors throughout the experiment. In the control part we removed the effect of the lattice by shuffling players every round. Finally, in experiment 2 players were again fixed on a lattice, albeit in different positions from those of experiment 1. On the screen players saw the actions and payoffs of their neighbors from the previous round, who in the control part were different from their current neighbors with high probability. All three parts of the experiment were carried out in sequence with the same players. Players were also fully informed of the different setups they were going to go through. The number of rounds in each part was randomly chosen in order to avoid players knowing in advance when that part was going to finish.

B.1.1 Experiment timeline

At the beginning of the experiment each player receives an envelope with a username and a password and is directed to a computer. On the computers the software is already running and there is a welcome message on the screen. By clicking on the button players are redirected to a login screen, where they can log in using the username and password they received. When everybody is logged in, an introductory tutorial starts. The rules are thoroughly explained with examples and a small test, to make sure that players understood the rules. Once all players have read the tutorial, the first part of the experiment starts. They see a screen showing them and their neighbors, as well as two buttons: blue and yellow, to choose from. After pressing the chosen button, they are presented with a new screen where they see their action and are asked to wait for everybody to play. When everybody has played, they see the actions and payoffs of their neighbors and themselves and they are asked to play again (choose one of the buttons on the right); see the technical instruction for the figure of a screen. This completes one round. After certain number of rounds, which is randomly chosen when the experiment is started, they are redirected to the tutorial for the second part, which is very brief and just explains the difference between part II and part I, and then they play again. After the second part is finished they get a brief tutorial for part III and then they play the third part. At the end of part III, they are asked some questions about the game, and after they answer they see how much money they earned and the good bye message. See the technical instruction for the sketch of the whole process. Therefore, the phases of the experiment are the following:

- Logging in
- Introductory tutorial
- Part I (experiment 1)
- Tutorial for part II
- Part II (control)

- Tutorial for part III
- Part III (experiment 2)
- Questionnaire about the game
- Final screen with information about their earnings

B.1.2 Software description

The software for the experiment was written in PHP 5, Javascript, and Python. In the original experiment there were 169 client computers running Opera in kiosk mode (to preclude players from doing anything else than playing according to the instructions) on Debian Linux. Clients communicated with the server through Javascript and PHP and on the server Python programs were running controlling the experiment, making calculations, and storing results. Another client was monitoring the whole experiment, displaying every player and their current status.

The basic purpose of the program was to present the information from the server to the clients, then receive the input from the clients, analyze it on the server and return new information, waiting for new input.

PHP is a general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. As such it was suitable for server side programming. However, client side programs cannot be done in PHP, and therefore all the programs on the client side (like pressing the buttons and the like) were done in Javascript. To avoid concurrency issues, we did the necessary analysis on the server through a background process which waited until all the clients finished certain segment. Then the process did the calculations and subsequently allowed PHP scripts to continue. We chose to write the background process in Python. Therefore, PHP is used to present information to the client, Javascript to receive input from the client and Python for the server analysis. The generation of auxiliary and data files is also done in Python. To automatically run all the Python code bash scripts were used.

A normal HTML website will not pass data from one page to another. In other words, all information is forgotten when a new page is loaded. To allow storing players' information on the server for later use (i.e. username, action, payoff, etc.) we used a PHP session. On the other hand, to pass variables between PHP and Python we stored them as text files on the server.

The software has two modes: normal mode and robot mode. Normal mode is used for the experiment itself and robot mode is used for testing the server. The difference is that in normal mode players are required to push the buttons, while in robot mode buttons are pressed automatically, therefore the robot mode is like having robots pressing the buttons instead of players. In this way, if we need to test the system, we do not need a large group of people pressing buttons, we just run the robot mode. The software is organized in the following directory structure:

```
PD
|- index.php                         Empty file, included for
|                                    security reasons
|- manual.txt                        Users manual
|- technical_instruction.pdf         Technical instructions
|----- PDexp                         Directory for normal mode of
|       |                            experiment
|       |- backup.sh                 Makes backup of the output
|       |                            files
|       |- clean.sh                  Resets data files to initial
|       |                            value
|       |- check_daemon.sh           Checks for background processes
|       |- makepass.py               Makes files with usernames and
|       |                            passwords
|       |- passwords.php             List of passwords
|       |- start_daemon.sh           Starts background processes
|       |- stop_daemon.sh            Stops background processes
|       |----- exp1                  Directory for Part I of the
|       |       |                    experiment
|       |       |- index.php         Redirects to current stage of
|       |       |                    experiment
|       |       |- first.php         Displays welcome screen
|       |       |- login.php         Displays login page
|       |       |- login2.php        Displays login page, if login
|       |       |                    fails the first time
|       |       |- waitlogin.php     Synchronizes all players
|       |       |                    for login
|       |       |- checkuser.php     Checks player's username
|       |       |                    and password
|       |       |- tutorial1.php     Display introductory
|       |       |                    tutorial pages
|       |       |- tutorial2.php
|       |       |- tutorial3.php
|       |       |- tutorial4.php
|       |       |- tutorial5.php
|       |       |- tutorial6.php
|       |       |- tutorial7.php
|       |       |- tutorial8.php
|       |       |- checktutorial8.php  Checks the tests on page 8 of
|       |       |                    the tutorial
|       |       |- tutorial9.php
|       |       |- part1.php         Announces that Part I is
|       |       |                    starting
|       |       |- ready.php         Informs that players are ready
|       |       |                    to play
|       |       |- wait.php          Synchronizes all players to
|       |       |                    start playing
|       |       |- main.php          Main program for playing
|       |       |                    the game
|       |       |- main_first.php    Displays the first playing
|       |       |                    screen
|       |       |- main_play.php     Screen prompting players to play
|       |       |- main_played.php   Screen after the players played
|       |       |- neighbor1.php
|       |       |- neighbor2.php
```

```
|        |        |- neighbor3.php
|        |        |- neighbor4.php       Presents actions and payoffs
|        |        |- neighbor5.php       in the last round for neighbors
|        |        |- neighbor6.php
|        |        |- neighbor7.php
|        |        |- neighbor8.php
|        |        |- you.php             Presents actions and payoffs
|        |        |                      for the player
|        |        |- buttonC.php         Makes button for
|        |        |                      cooperation (blue)
|        |        |- buttonD.php         Makes button for
|        |        |                      defection (yellow)
|        |        |- play.php            Writes actions to data files
|        |        |- play_first.php      Generates automatic action in
|        |        |                      the first round
|        |        |- play_auto.php       Generates automatic action after
|        |        |                      the first round
|        |        |- part1end.php        Announces that Part I has
|        |        |                      finished
|        |        |- sumain.php          Main program for monitoring user
|        |        |- logout.php          Logs the player out
|        |        |- styleproba.css      Style file for tutorial pages
|        |        |- stylemain1.css      Style file for playing screens
|        |        |- back_verlauf.jpg    Images for tutorial pages
|        |        |- banner.jpg
|        |        |- bgimage.gif
|        |        |- logo.gif
|        |        |- osmbanner1.png
|        |        |- payoff.gif
|        |        |- played.jpg
|        |        |------ data                   Stores data and
|        |                |                      background process
|        |                |- makefiles.py        Makes data files
|        |                |- makenet.py          Makes files with
|        |                |                      neighbors
|        |                |- calculate_exp1.py   Background process for
|        |                |                      part I
|        |                |- cleanfiles.py       Resets values of data
|        |                |                      files
|        |                |- erase.py            Removes old data files
|        |                |- erase_exp1.sh       Removes old data files
|        |
|        |------ control                         Directory for Part II of
|        |        |                              the experiment
|        |        |- unset_control.php           Redirects to index file
|        |        |                              in this folder
|        |        |- part2tutorial1.php          Tutorial files
|        |        |- part2tutorial2.php
|        |        |- main_control.php            Main program for playing
|        |        |                              in this part
|        |        |- part2end.php                Announces the end of
|        |        |                              Part II
|        |        |...(the same files as in exp1)
|        |        |------ data
|        |                | - makenet_control.php       Makes neighbor
```

```
|       |               |                            files for Part II
|       |               | - calculate_control.php     Background process
|       |               |                            for Part II
|       |               | - erase_control.php         Removes old data
|       |               |                            files
|       |               | - (the same files as in exp1)
|       |------ exp2                            Directory for Part II
|              |                                of the experiment
|              |- unset_exp2.php                Redirects to index
|              |                                file in this folder
|              |- part3.php                     Announces the
|              |                                beginning of Part III
|              |- part3_end.php                 Announces the end of
|              |                                Part III
|              |- questions1.php                Displays questionnaire
|              |- write.php                     Writes answers to data
|              |                                files
|              |- last.php                      Displays goodbye screen
|              |... (the same files as in exp1)
|              |---- data
|                      | - erase_exp2.php       Removes old data files
|                      | - calculate_exp2.php   Background process for
|                      |                        Part III
|                      |... (the same files as in exp1)
|
|------- PDrobot                                Programs for robot mode
|        | - exp1
|             | - makescripts.py                Makes scripts for
|             |                                 automatic login
|             |...the same structure as PDexp,
|                but some files are changed
|
|------- client_scripts                         Scripts which should be
|         |                                     on client
|         |- experiment.sh                      Starts Opera with
|         |                                     appropriate parameters
|         |- robot_make.sh                      Makes scripts for
|         |                                     starting the robots
|         |- robot.py                           Used by robot_make.sh
|         |                                     to make scripts
|         |- robot1.example                     Example of the scripts
|         |- IPaddresses.txt                    IP addresses of client
|         |                                     computers
|         |- setopera.sh                        Copies configuration
|         |                                     file for Opera
|         |- setopera.py                        Makes setopera.sh
|
|--------configuration_files                    Configuration files
|         |                                     for the server
          |- apache2.conf.experiment            For Apache
          |- php.ini.experiment                 For PHP
```

B.2 System requirements and server settings

The system consisted of the server and the clients. On the server Apache2+, PHP5+ and Python were installed and on the clients we needed Opera with Javascript enabled. Server and clients communicated through Internet.

During the experiment many clients are making many requests to the server. Therefore, the following directives on the server should be adjusted to allow for a large number of requests during the whole session.

B.2.1 Apache settings

In the file `/etc/apache2/apache2.conf`, the following settings should be changed from the default option:

- `MaxKeepAliveRequests 0 (default 100)`

The MaxKeepAliveRequests directive limits the number of requests allowed per connection when KeepAlive is on. If it is set to 0, unlimited requests will be allowed. We recommend that this setting is set to a high value allowing all the clients to connect to the server and play the game. In our experiment we set it to 0, allowing unlimited requests.

- `KeepAliveTimeout 120 (default 15)`

The number of seconds Apache will wait for a subsequent request before closing the connection. Once a request has been received, the timeout value specified by the Timeout directive applies. Since we do not want our connections to be closed at any moment, we recommend that this setting is set to a high value.

- `ServerLimit 5000` (add this line before MaxClients)
- `MaxClients 5000`

The MaxClients directive sets the limit on the number of simultaneous requests that will be served. Any connection attempts over the `MaxClients` limit will normally be queued. The default value is 256; to increase it, you must also raise `ServerLimit`. Since our server needs to serve 200 clients each with many requests, we raised the limit to 5000 and set the same for `Server Limit`.

B.2.2 PHP settings

To allow storing players' information on the server for later use (i.e. username, action, payoff, etc.) we used a PHP session. However, this session information is temporary. To prolong its lifetime, the following settings should be changed in the file `/etc/php5/apache2/php.ini`.

- `session.gc_maxlifetime = 36000 (default 1440)`

After this number of seconds, stored data will be seen as 'garbage' and cleaned up by the garbage collection process.

- `session.cache_expire = 660 (default 180)`

Document expires after n minutes.

- `session.cookie_lifetime = 0 (default 0)`

Lifetime of cookies in seconds or, if 0, until browser is restarted. This should be 0 by default; nevertheless, it should be checked.

- `session.gc_probability = 0 (default 1)`
- `session.gc_divisor = 100 (default 100)`

Define the probability that the 'garbage collection' process is started on every session initialization. The probability is calculated using `gc_probability/gc_divisor`, e.g. $1/100$ means there is a 1% chance that the GC process starts on each request. This is disabled in the Debian packages, due to the strict permissions on `/var/lib/php5`. If your server is running on Debian Linux, instead of setting this here, see the cronjob at `/etc/cron.d/php5`.

B.3 Tests with robots

Before running the experiment it is necessary to test the whole system. Instead of having many volunteers who are playing the game, it is easier to run the program in robot mode, where the buttons are pressed automatically.

B.3.1 Requirements for the test

A typical test consisted of 196 (the maximum number of players we would have) computer terminals which had 196 different accounts, where all accounts were on another server which was within the same network. Therefore on every terminal one can login with any of the accounts. The account names should be `experiment1`, `experiment2`,..., `experiment196`. There was one super user account which had permission to log into any other account without password. We call this "robot account". The folder called `client_scripts` should be copied in the home directory of the robot account. All the necessary scripts for running the robots are in this folder.

To start the test with robots, the same procedure as for starting the experiment should be applied (look below) except that instead of starting everything from the folder PDexp, it has to be done folder PDrobot.

Once the experiment is started, the following scripts `robot1.sh`, `robot2.sh`, ..., `robot8.sh` (which are in the folder `clients_scripts` on the robot account) should be started from the robot account. These scripts will start Opera through the

X-window System on 196 accounts on the different computers. As the Opera windows will all be on the computer where the script is started, it is recommended to start each of the scripts from a different computer. To generate these scripts for your system, you should use the script `robot_make.sh` which uses file IPadresses.txt with the IP address of all computers and as a parameter the web address on the server where the robot scripts are. This means that the script is started in the following way:

```
./robot_make.sh  http://path_on_the_server/PD/PDexp/exp1/
```

For running this script the file `robot.py` is also necessary. Once run, the robots should start to login and play automatically. If they have problems logging in or the server seems overloaded, adjust the parameters on the server again. For our system the above parameters were sufficient. Notice that if the test is working on one system, that does not mean it will work on other system (for example, when players are logging in from different locations outside of internal network).

B.4 Running an experiment

B.4.1 A day before the experiment

At the beginning of the experiment players need to login to the experiment with their unique login name and password. Login names are generically set to be `usuario1`, `usuario2`,...,`usuario196` but a list of password should be specifically generated for each experiment. Therefore, for each experiment file `password.php` should be generated. The file consists of the list of access words and the following three lines at the end of the file:

```
<script type="text/javascript">
top.location="index.php";
</script>
```

These lines are there for security reasons. Since this file, as well as the whole software, is in the public folder, it is possible that somebody guesses the web address and opens it by chance. However, if that happens it will be automatically redirected to the index file and they will not be able to see the passwords. All the passwords in the list should be one word with only ASCII characters. The number of passwords in the list should be equal to the maximal number of players. If number of players N is smaller than the number of passwords, the first N passwords from the list will be used. In the folder `/PD/PDexp/` there is an example file `passwords_example.php`. Afterwords, start `makepass.sh` with the following line:

1) `./makepass.sh N`,

where N is maximal number of players you want to participate in the experiment (in our case 196). This will make file `userlist.php` in `/PD/PDexp/exp1/data/` where users names are randomly associated to one of the passwords. There will be two columns in this file: the first one with the usernames and the second one with associated passwords. Like this:

```
username1 password1
username2 password2
username3 password3
...
```

Username and the corresponding password will be in the same line, `password1` will be the password for the player with `username1`, etc. At the end of the file there will be the same three lines as in file `password.php`, again, for the security reasons. After this file is created the pairs of usernames and passwords should be printed on the individual papers and placed into the envelopes, which will be distributed to the players just before the experiment.

At the end, file `experiment.sh`, which could be found in the folder `/PD/PDexp/clients_scripts`, should be copied to all the accounts that will be used by the players during experiment.

B.4.2 The day of the experiment

At the beginning we need to make sure that there are no previously started programs still running, and that all the files contain initial values. On the server, go to the directory `/PD/PDexp/` and start `check_daemon.sh` to check the presence of possible active background programs from the previous runs:

2) `./check_daemon.sh`.

If it shows some active background processes called `calculate` (with the exception of `grep calculate`), start `stop_daemon.sh` to stop them:

3) `./stop_daemon.sh`.

Then run `clean.sh` to reset all the files to their initial values:

4) `./clean.sh`

Now, one can proceed by starting the clients. Turn on the client computers and login each of them to a different account (the same accounts used for robots could be used). There should already be file `experiment.sh`, which was previously copied to the account. Start this script:

I) `./experiment.sh`

The script should start Opera in kiosk mode and present welcome screen of the experiment. Warning: The clients' computers are now ready for the experiment, but do not let the volunteers to enter the computer rooms yet! If some of them is logging in while the experiment is being started the program may get stuck.

B.4.3 Just before the experiment

According to the number of volunteers make the largest possible division $N \times M$, where N and M are approximately the same. Using these numbers start the background process:

6) `./start_daemon.sh N M Rmin Rmax mt_python backup`

where N and M are the dimensions of the network, R_{min} is the minimal number of rounds in each part of the experiment, R_{max} is the maximal number of rounds in each part of the experiment, mt_{python} is the time before the background process (daemon) plays instead of the player (recommended mt_{python}=40) and `backup` is the name of the back up file.

The whole experiment could be monitored from a separate computer. On the monitoring computer go to the site: `'servername'/PD/PDexp/exp1/` and login as `SuperUser` with password: `suclave` (password can be changed either in file `makepass.py`, as written in the comments of the file, before generating the password or in `userlist.php` after that, by changing the word next to `SuperUser`)

Now, the volunteers can enter the computer rooms. Give the players envelopes with usernames and passwords and let them enter the classrooms, login and start playing.

Follow what is going on on the monitor screen. You will see the table with all of the players. The color of the field tells what stage of experiment the player is in:

- grey: not logged in
- green: logged in, but did not read the tutorial
- white: read the tutorial, did not play
- yellow: played yellow
- blue: played blue

The meanings of the colors are explained in the legend on the top of the screen. Below is the table with the usernames of the players, their payoffs in the previous round and total earnings in all previous rounds in that part of experiment. In experiment 1 and experiment 2 the distribution of the players is exactly the same as in the experiment. This means that if two players are neighbors in the monitoring table, they are also neighbors in the experiment. However in the control part the distribution of the players in the monitoring table is arbitrary.

As mentioned before the length of the game is determined randomly at the beginning of the experiment. In case one wanted to change the number of rounds later during the game, it could be done from the monitoring screen. Above the table there are two buttons: "Stop now!" and "Plus 5 rounds" which are there if you want to stop the game or prolong it by 5 rounds, respectively. (Warning! "Plus 5 round" is not working properly)

B.5 Outputs

After the experiment is finished the earning of all the players (without the show up fee) will be in the file `/PD/PDexp/exp2/earnings`. In this file, the total earnings (without the show up fee) of all players are listed in the following way:

```
Earnings in_euros
usuario1 34.81
usuario5 23.86
```

```
usuario13 31.15
usuario12 25.51
usuario67 32.2
...
```

This information can be extracted from the `usuarioNhistory` files, but it is provided here so that the players could be paid immediately after the experiment.

Information about their actions and payoffs are in files:

```
exp1/usuario*history
control/usuario*history
exp2/usuario*history
```

In these files there will be 5 columns like this:

```
Round Move Score Time[ms] PlayedBy
1 C 21 11019.9999809 user
2 C 14 10210.0000381 user
3 D 40 5289.99996185 user
4 D 30 10319.9999332 user
5 C 14 9480.00001907 user
....
```

The first column is the round number, the second one the action of the player, the third one his/her payoff in the round, the fourth one how much time it took him/her to play, and the last one tells us who made the action. `User` means player took the action, `auto` means that the main.php module took the action, and `daemon` means that the background process was the one who played.

The answers to the small tests from the introductory tutorial and the questionnaire at the end of the experiment are in the folder `/PD/PDexp/exp2`. The names of these files are `usuarioNanswers` (where N is the ordinal number of the player) and they look like this:

```
Answers:
question1: #Here comes Correct or Incorrect and the answer they gave
question2: #Here comes Correct or Incorrect and the answer they gave
question3: #Here comes Correct or Incorrect and the answer they gave
question4: #Here comes Correct or Incorrect and the answer they gave

Describe briefly how you made your decisions in part I [Experiment 1]:
# Here comes the answer
Describe briefly how you made your decisions in part II [Control].
# Here comes the answer
Describe briefly how you made your decisions in part III [Experiment 2].
# Here comes the answer
Did you take into account your neighbors actions?
# Here comes the answer
Is something in the experiment familiar to you? (yes/no).
# Here comes the answer
```

```
If so, please point out what it reminds you of.
# Here comes the answer
If you want to make any comment, please do so below.
# Here comes the comment
```

The first 4 lines give us information about the tests from the tutorial. After question1, question2, etc., it says “Correct” if the player answered correctly or “Incorrect” otherwise, and the number they gave as their answer if they answered incorrectly. After that, we find the questions from the final questionnaire with the answers the player gave.

Wait for a while after the experiment and check if all the players finished answering the questionnaire at the end. Then run the backup script to backup all the results:

```
./backup.sh name_of_backup_file
```

C

Technical Instruction for the PDexp software

PDexp is a software for conducting Prisoner's Dilemma (PD) experiments with human subjects on a large square lattice, as described in Chapter 2. The software has been developed by J. Grujić at Grupo Interdisciplinar de Sistemas Complejos at Universidad Carlos III de Madrid.

In the experiment, volunteers played a 2×2 PD game with each of their eight neighbors (Moore neighborhood) taking only one action, either to cooperate (C) or to defect (D), the action being the same against all the opponents. The resulting payoff was calculated by adding all eight interaction payoffs. Payoffs of the PD game were set to be 7 cents of a euro for mutual cooperation, 10 cents for a defector facing a cooperator, and 0 cents for any player facing a defector (weak PD). With this choice (a cooperator and a defector receive the same payoff against a defector) defection is not a risk dominant strategy, which enhances the possibility that cooperation emerges. The payoffs are given as parameters of the program when started, therefore the experiment can be performed with other payoffs. To avoid framing effects, the two actions were always referred to in terms of colors (blue for C and yellow for D), and the game was never referred to as PD in the material handed to the volunteers. This notwithstanding, players were properly informed of the consequences of choosing each action, and some examples were given to them in the introduction. After every round players were given the information of the actions taken by their neighbors and their corresponding payoffs.

The full experiment consisted of three parts: experiment 1, control, and experiment 2. In experiment 1 players remained at the same positions in the lattice with the same neighbors throughout the experiment. In the control part we removed the effect of the lattice by shuffling players every round. Finally, in experiment 2 players were again fixed on a lattice, albeit in different positions from those of experiment 1. On the screen players saw the actions and payoffs of their neighbors from the previous round, who in the control part were different from their current neighbors with high probability. All three parts of the experiment were carried out in sequence with the same players. Players were also fully informed of the different setups they were going to go through. The number of rounds in each part was randomly chosen in order to avoid players knowing in advance when that part was going to finish.

C.1 Experiment timeline

At the beginning of the experiment each player receives a closed envelope with a username and a password and is assigned to a computer. On the computers the software is already running and there is a welcome message on the screen. By clicking on the button players are redirected to a login screen, where they can log in using the username and password they received. When everybody is logged in, an introductory tutorial starts. The rules are thoroughly explained with examples and a small test, to make sure that players understood the rules. Once all players have read the tutorial, the first part of the experiment starts. They see a screen showing themselves and their neighbors, as well as two buttons: blue and yellow, to choose from. After pressing the chosen button, they are presented with a new screen where they see their action and are asked to wait for everybody to play. When everybody has played, they see the actions and payoffs of their neighbors and themselves and they are asked to play again (choose one of the buttons on the right) as seen on Fig. C.1. This completes one round. After certain number of rounds, which is randomly chosen when the experiment is started, they are redirected to the tutorial for the second part, which is very brief and just explains the difference between part II and part I, and then they play again. After the second part is finished they get a brief tutorial for part III and then they play the third part. At the end of part III, they are asked some questions about the game, and after they answer they see how much money they earned and the goodbye message. The whole process is sketched in Fig. C.2. Therefore, the phases of the experiment are the following:

- Logging in
- Introductory tutorial
- Part I (experiment 1)
- Tutorial for part II
- Part II (control)
- Tutorial for part III
- Part III (experiment 2)
- Questionnaire about the game
- Final screen with information about their earnings

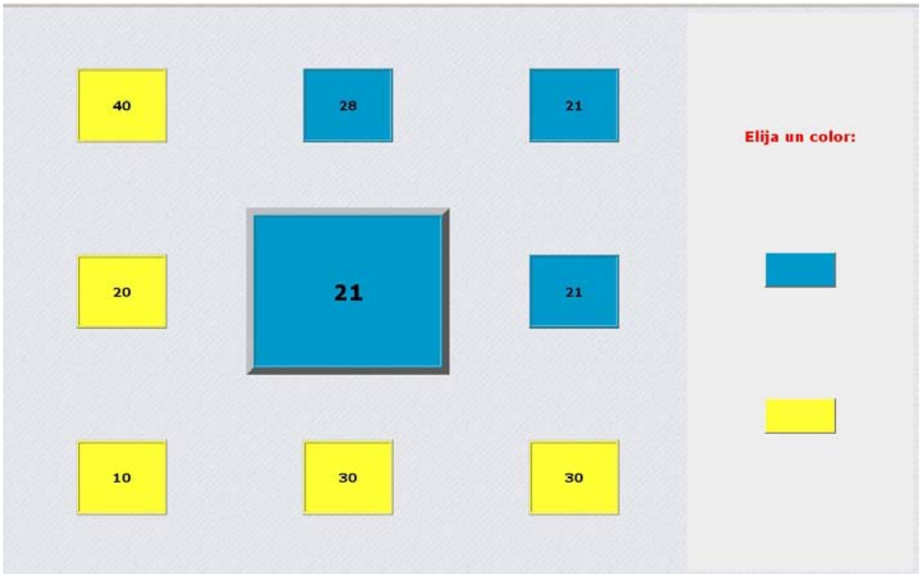


Figure C.1: *Screen displayed after every round of the game.* Numbers are the payoffs obtained in the most recent round. Colors represent actions (yellow: defect, blue: cooperate). The central square represents the player; the surrounding squares represent her eight neighbors. ("Elija un color" means "Choose a color" in Spanish.)

C.2 Software description

The software for the experiment was written in PHP 5, Javascript, and Python. In the original experiment there were 169 client computers running Opera in kiosk mode (to preclude players from doing anything else than playing according to the instructions) on Debian Linux. Clients communicated with the server through Javascript and PHP and on the server Python programs were running controlling the experiment, making calculations, and storing results. Another client was monitoring the whole experiment, displaying all players and their current status.

The basic purpose of the program was to present the information from the server to the clients, then receive the input from the clients, analyze it on the server and return new information, waiting for new input.

PHP is a general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. As such it was suitable for server side programming. However, client side programs cannot be done in PHP, and therefore all the programs on the client side (like pressing the buttons and the like) were done in Javascript. To avoid concurrency issues, we did the necessary analysis on the server through a background process which waited until all the clients finished certain segment. Then the process did the calculations and subsequently allowed PHP scripts to

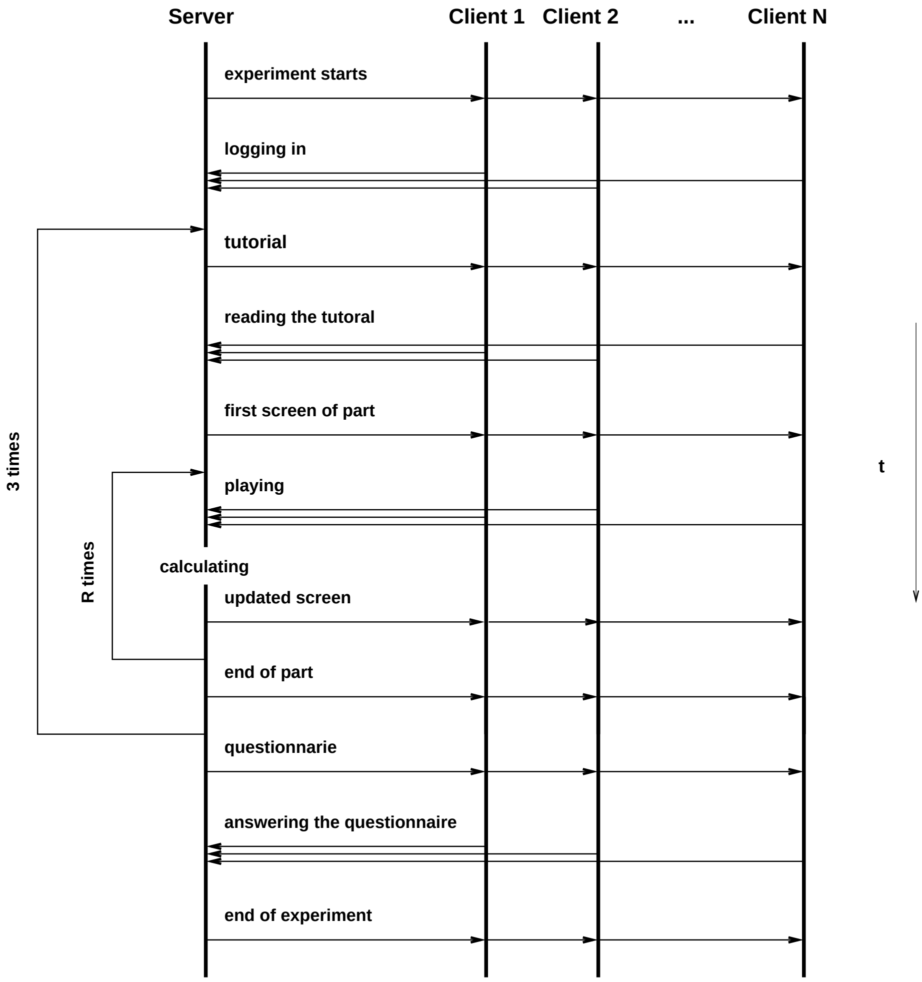


Figure C.2: *Experiment timeline.* After the server sends the starting signal to the clients, the clients log in, each at their own time. When everybody is logged in, the server sends the tutorial to all of them at the same time. Clients finish reading the tutorial each at their own time and afterward the server sends the first playing screen. Clients play and after everybody finishes they see the updated screen. The cycle of playing and updating the screen is repeated R times where R is the number of rounds. The part from sending the tutorial until the last playing cycle is repeated 3 times, for experiment 1, control and experiment 2. At the end players are asked to fill the questionnaire in.

continue. We chose to write the background process in Python. Therefore, PHP is used to present information to the client, Javascript to receive input from the client and Python for the server analysis. The generation of auxiliary and data files is also done in Python. To automatically run all the Python code bash scripts were used.

A normal HTML website will not pass data from one page to another. In other words, all information is forgotten when a new page is loaded. To allow storing players' information on the server for later use (i.e. username, action, payoff, etc.) we used a PHP session. On the other hand, to pass variables between PHP and Python we stored them as text files on the server.

The software has two modes: normal mode and robot mode. Normal mode is used for the experiment itself and robot mode is used for testing the server. The difference is that in normal mode, players are required to push the buttons, while in robot mode, buttons are pressed automatically, therefore the robot mode is like having robots pressing the buttons instead of players. In this way, if we need to test the system, we do not need a large group of people pressing buttons, we just run the robot mode.

C.3 Input and output files

The whole software consists of the programs run on the server and the program run on the clients. However, most of the software runs on the server. In normal mode, on the client we run only one module, `experiment.sh`, which is used to start Opera in kiosk mode. This module neither uses any input file nor produces any output file. On the other hand, in robot mode we use more modules to start. The only input file they use is a file with the IP addresses (see the example in `PD/client_scripts`) of the computers in the network, and again they do not produce any output file.

On the server side the input and output is the same in both normal and robot modes (although in the robot mode output is irrelevant). Apart from the input which the program gets from the players, the only input it uses is the file with passwords (`PD/'mode'/passwords.php`, where 'mode' can be either `PDexp` or `PDrobot`, depending on the mode the software is run in) and the parameters of the program. The `passwords.php` file consists of the list of access words and the following three lines at the end of the file:

```
<script type="text/javascript">
top.location="index.php";
</script>
```

These lines are there for security reasons. Since this file, like the whole software, is in the public folder, it is possible that somebody guesses the web address and opens it by chance. However, if that happens it will be automatically redirected to the index file and they will not be able to see the passwords. All the passwords in the list should be one word with only ASCII characters.

The most important parameters of the experiment are the size of the network, given as $N \times M$, and the minimum and maximum number of rounds (R_{min} and R_{max}) of each part. In the original experiment the size of the network was 13×13 and the number of rounds was randomly chosen between 40 and 60 (therefore $R_{min} = 40$ and $R_{max} = 60$). Afterward, we define the values of the parameters for the Prisoner's dilemma, originally set to sucker's payoff $S = 0$, punishment for mutual defection $P = 0$, reward for mutual cooperation $R = 7$ cents of euro, and temptation to defect $T = 10$ cents of euro. One should additionally give the time after which the background program will play instead of the player, mt_{python}. More precisely, if the player does not play, after 30 seconds the module `main.php` (specifically `play_auto.php` or `play_first` in the first round) will play. However, in case that a client is temporarily disconnected, the module `main.php` will not be able to play and the background process will then take action after mk_{python} seconds. In the original experiment mk_{python} was 40 seconds. The last parameter of the program is the name of the backup file for the data from the previous run. This is introduced to avoid losing important data when starting a new experiment. These parameters are introduced when running the `start_daemon` script in the following way:

```
./start_daemon.sh N M Rmin Rmax R S T P mt_python backup
```

There are three types of output files generated by the code. The most important ones are the files containing the actions of the players. The names of these files are: `usuario1history`, `usuario2history`, ..., `usuarioNhistory`. We have three different sets of these files, one for each part of experiment. Therefore each set is in its own folder: `PD/'mode'/'experiment'/data`, where 'experiment' can be exp1, control, exp2, depending on the part that is being executed. Mode can be 'PDexp' or 'PDrobot', although in robot mode these files are irrelevant. These files have 5 columns, like this:

```
Round Move Score Time[ms] PlayedBy
1 C 21 11019.9999809 user
2 C 14 10210.0000381 user
3 D 40 5289.99996185 user
4 D 30 10319.9999332 user
5 C 14 9480.00001907 user
....
```

The first column is the round number, the second one the action of the player, the third one his/her payoff in the round, the fourth one how much time it took him/her to play, and the last one tells us who made the action. `User` means the player performed the action, `auto` means that the `main.php` module made the action, and `daemon` means that the background process was the one that played.

The second type of output are the answers to the small tests from the introductory tutorial and the questionnaire at the end of the experiment. The names of these files are `usuarioNanswers` (where N is the ordinal number of the player) and they looked like this:

```
Answers:
question1: # Here comes Correct or Incorrect and the answer they gave
question2: # Here comes Correct or Incorrect and the answer they gave
question3: # Here comes Correct or Incorrect and the answer they gave
question4: # Here comes Correct or Incorrect and the answer they gave

Describe briefly how you made your decisions in part I [Experiment 1]:
# Here comes the answer
Describe briefly how you made your decisions in part II [Control].
# Here comes the answer
Describe briefly how you made your decisions in part III [Experiment 2].
# Here comes the answer
Did you take into account your neighbors actions?
# Here comes the answer
Is something in the experiment familiar to you? (yes/no).
# Here comes the answer
If so, please point out what it reminds you of.
# Here comes the answer
If you want to make any comment, please do so below.
# Here comes the comment
```

The first 4 lines give us information about the tests from the tutorial. After question1, question2, etc., it says "Correct" if the player answered correctly or "Incorrect" otherwise, and the number they gave as their answer if they answered incorrectly. After that, we find the questions from the final questionnaire with the answers the player gave.

The last type of output is the file `earnings`, which is in folder `PD/PDexp/exp2`. In this file, the total earnings (without the show up fee) of all players are listed in the following way:

```
Earnings in_euros
usuario1 34.81
usuario5 23.86
usuario13 31.15
usuario12 25.51
usuario67 32.2
...
```

This information can be extracted from `usuarioNhistory` files, but it is provided here so that the players could be paid immediately after the experiment.

In summary, the input and output files are the following:

Server side (same for normal and robot mode):

- Input:
  - players' actions
  - file with passwords
  - parameters of experiment (entered in command line)
- Output:

- history files for all players in three parts separately
- answer files for all players
- earnings file

Client side:

- Normal mode:
  - no input, no output
- Robot mode:
  - input: IPadresses.txt file
  - output: no output files

C.4 Structure of the program

In the main folder for the normal mode there are bash scripts for starting the experiment and three different folders `exp1`, `control` and `exp2`, which have a similar structure. In each of these three folders there is an `index.php` file. At the beginning, using the bash scripts we execute the Python scripts and start the background process. Afterward the clients can log into the experiment. The program starts the execution from `exp1`. The file `index.php` in the folder `exp1` redirects clients to the current phase of experiment. At the end of part I the clients are redirected to the folder `control`, where `unset_control.php` sets the current phase to the first tutorial in part II and then goes to the file `index.php` in the folder `control`, which redirects clients to the current stage of the experiment. At the end of Part II, clients are once again redirected to `exp2`, where the current phase is set in file `unset_exp2.php` and they are then redirected to the file `index.php` in folder `exp2`.

The folder structure is as follows:

```
Directory structure:

PD
|- index.php                        Empty file, included for
|                                   security reasons
|- manual.txt                       Users manual
|- technical_instruction.pdf        Technical instructions
|----- PDexp                        Directory for normal mode of
|       |                           experiment
|       |- backup.sh                Makes backup of the output
|       |                           files
|       |- clean.sh                 Resets data files to initial
|       |                           value
|       |- check_daemon.sh          Checks for background processes
|       |- makepass.py              Makes files with usernames and
|       |                           passwords
```

```
|       |- passwords.php              List of passwords
|       |- start_daemon.sh            Starts background processes
|       |- stop_daemon.sh             Stops background processes
|       |----- exp1                   Directory for Part I of the
|       |       |                     experiment
|       |       |- index.php          Redirects to current stage of
|       |       |                     experiment
|       |       |- first.php          Displays welcome screen
|       |       |- login.php          Displays login page
|       |       |- login2.php         Displays login page, if login
|       |       |                     fails the first time
|       |       |- waitlogin.php      Synchronizes all players
|       |       |                     for login
|       |       |- checkuser.php      Checks player's username
|       |       |                     and password
|       |       |- tutorial1.php      Display introductory
|       |       |                     tutorial pages
|       |       |- tutorial2.php
|       |       |- tutorial3.php
|       |       |- tutorial4.php
|       |       |- tutorial5.php
|       |       |- tutorial6.php
|       |       |- tutorial7.php
|       |       |- tutorial8.php
|       |       |- checktutorial8.php  Checks the tests on page 8 of
|       |       |                     the tutorial
|       |       |- tutorial9.php
|       |       |- part1.php          Announces that Part I is
|       |       |                     starting
|       |       |- ready.php          Informs that players are ready
|       |       |                     to play
|       |       |- wait.php           Synchronizes all players to
|       |       |                     start playing
|       |       |- main.php           Main program for playing
|       |       |                     the game
|       |       |- main_first.php     Displays the first playing
|       |       |                     screen
|       |       |- main_play.php      Screen prompting players to play
|       |       |- main_played.php    Screen after the players played
|       |       |- neighbor1.php
|       |       |- neighbor2.php
|       |       |- neighbor3.php
|       |       |- neighbor4.php      Presents actions and payoffs
|       |       |- neighbor5.php      in the last round for neighbors
|       |       |- neighbor6.php
|       |       |- neighbor7.php
|       |       |- neighbor8.php
|       |       |- you.php            Presents actions and payoffs
|       |       |                     for the player
|       |       |- buttonC.php        Makes button for
|       |       |                     cooperation (blue)
|       |       |- buttonD.php        Makes button for
|       |       |                     defection (yellow)
|       |       |- play.php           Writes actions to data files
|       |       |- play_first.php     Generates automatic action in
```

```
|       |       |                      the first round
|       |       |- play_auto.php       Generates automatic action after
|       |       |                      the first round
|       |       |- part1end.php        Announces that Part I has
|       |       |                      finished
|       |       |- sumain.php          Main program for monitoring user
|       |       |- logout.php          Logs the player out
|       |       |- styleproba.css      Style file for tutorial pages
|       |       |- stylemain1.css      Style file for playing screens
|       |       |- back_verlauf.jpg    Images for tutorial pages
|       |       |- banner.jpg
|       |       |- bgimage.gif
|       |       |- logo.gif
|       |       |- osmbanner1.png
|       |       |- payoff.gif
|       |       |- played.jpg
|       |       |------ data                   Stores data and
|       |               |                      background process
|       |               |- makefiles.py        Makes data files
|       |               |- makenet.py          Makes files with
|       |               |                      neighbors
|       |               |- calculate_exp1.py   Background process for
|       |               |                      part I
|       |               |- cleanfiles.py       Resets values of data
|       |               |                      files
|       |               |- erase.py            Removes old data files
|       |               |- erase_exp1.sh       Removes old data files
|       |
|       |------ control                        Directory for Part II of
|       |       |                              the experiment
|       |       |- unset_control.php           Redirects to index file
|       |       |                              in this folder
|       |       |- part2tutorial1.php          Tutorial files
|       |       |- part2tutorial2.php
|       |       |- main_control.php            Main program for playing
|       |       |                              in this part
|       |       |- part2end.php                Announces the end of
|       |       |                              Part II
|       |       |...(the same files as in exp1)
|       |       |------ data
|       |               | - makenet_control.php       Makes neighbor
|       |               |                             files for Part II
|       |               | - calculate_control.php     Background process
|       |               |                             for Part II
|       |               | - erase_control.php         Removes old data
|       |               |                             files
|       |               | - (the same files as in exp1)
|       |------ exp2                             Directory for Part II
|               |                                of the experiment
|               |- unset_exp2.php                Redirects to index
|               |                                file in this folder
|               |- part3.php                     Announces the
|               |                                beginning of Part III
|               |- part3_end.php                 Announces the end of
|               |                                Part III
```

```
|                |- questions1.php                Displays questionnaire
|                |- write.php                     Writes answers to data
|                |                                files
|                |- last.php                      Displays goodbye screen
|                |... (the same files as in exp1)
|                |---- data
|                        | - erase_exp2.php       Removes old data files
|                        | - calculate_exp2.php   Background process for
|                        |                        Part III
|                        |... (the same files as in exp1)
|
|------- PDrobot                                  Programs for robot mode
|        | - exp1
|             | - makescripts.py                  Makes scripts for
|             |                                   automatic login
|             |...the same structure as PDexp,
|                but some files are changed
|
|------- client_scripts                           Scripts which should be
|         |                                       on client
|         |- experiment.sh                        Starts Opera with
|         |                                       appropriate parameters
|         |- robot_make.sh                        Makes scripts for
|         |                                       starting the robots
|         |- robot.py                             Used by robot_make.sh
|         |                                       to make scripts
|         |- robot1.example                       Example of the scripts
|         |- IPaddresses.txt                      IP addresses of client
|         |                                       computers
|         |- setopera.sh                          Copies configuration
|         |                                       file for Opera
|         |- setopera.py                          Makes setopera.sh
|
|--------configuration_files                      Configuration files
|         |                                       for the server
          |- apache2.conf.experiment              For Apache
          |- php.ini.experiment                   For PHP
```

C.5 Modules

The experimental setup consists of four parts. The first part are the programs for starting the experiment. The corresponding files are in folder `PD/PDexp` and are mostly bash scripts, the only exception is a Python file for generating the file with passwords. The second part is the interaction part, which presents the playing screens and stores the data about the players actions. These files are in folders `/PD/PDexp/exp1`, `PD/PDexp/control` and `PD/PDexp/exp2`, and are written in PHP and Javascript, which are embedded in HTML. The third part are the files for the background processes. There are three of them, one for each part of the experiment:

`/PD/PDexp/exp1/data/calculate_exp1.py`

`/PD/PDexp/control/calculate_control.py`

`/PD/PDexp/exp2/calculate_exp2.py`

These files are written in Python. The last part of the setup are the files that generate the necessary data files for running it, once given the parameters for the experiment. These files are also written in Python.

C.5.1 Start up part

The scripts for running and stopping the experiment are in the folder `/PDexp`:

- `makepass.py` — uses `passwords.php` and produces `userlist.php`, which is necessary for logging in. The file `userlist.php` has two columns: the first one with the usernames of the players and the second one with the associated passwords. All the usernames have the same first part `usuario` and then a number from 1 to N, where N is the total number of players. Passwords are the words from the file `passwords.php`, which are randomly associated. The username for the monitoring client is `SuperUser` and the password `suclave`. At the end, this file has the lines which redirect a random visitor away from this site without seeing its content (the same as in `passwords.php`)

- `check_daemon.sh` — looks for remaining background processes. It lists all the processes which are called `calculate`. If the only process listed is "grep calculate*", then there are no active background processes

- `stop_daemon.sh` — stops remaining background processes by killing any process which has 'calculate' in its name.

- `start_daemon.sh` — starts background processes. This is the most important file in this module. It runs the scripts for generating data files and the background process. The parameters of this file are the parameters of experiment (as explained in section C.3). For its use, see `manual.txt`.

C.5.2 Interaction part

This part consists of all the programs necessary for the communication between the system and the players. It has the program that presents the working screens and stores raw data about the actions of the players. These raw data files will be later used by the background process. Most of the files in this module are the same in the three parts of the experiment. The difference is mainly in the tutorial files, although there are some differences also in the other files. The first program is `index.php` which redirects the player to the current stage of the experiment. The main part is the file `main.php`. This file comes in three flavors, depending of the current stage of the game. It first stores the time of the beginning of the action in the file `usuarioNtimestart`, where N is the ordinal number of the player. If it is the very beginning of the game it includes the

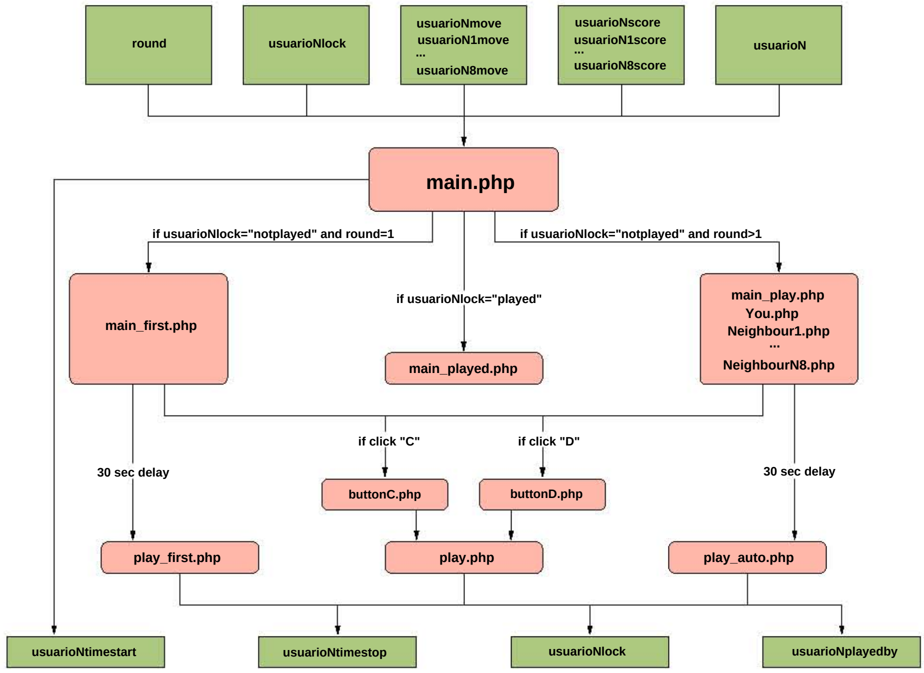


Figure C.3: *Main module of the interaction part.* Relationships with input and output data and other modules.

file `main_first.php`. After the player took the action and is waiting for others to take their action, it includes `main_played.php`. Finally, when the player is going to take a new action it includes `main_play.php`. The files `main_first.php` and `main_play.php` will further include the files `buttonC.php` and `buttonD.php`, which will trigger the file `play.php` that stores the information about the action in files `usuarioNlock`, `usuarioNtimestop`, `usuarioNplayedBy`. If the player does not play in 30 seconds the files `play_first.php` (for `main_first.php`) or `play_auto.php` (for `main_play.php`) are triggered and they store the data in the same files. Figure C.3 shows a scheme of this module.

Internal input files for this part of the program:

- `round` — contains the current round,
- `usuarioNlock` — the player's action in this round or "notplayed" if he/she has not played yet,
- `usuarioNmove` — the action in the previous round,
- `usuarioN1move`,..., `usuarioN8move` — neighbors' actions in the previous round,

- usuarioNscore — player's payoff in the previous round,
- usuarioN1score,..., usuarioN8score — neighbors' payoffs in the previous round,
- usuarioN — the list of player's neighbors.

In the first round players enter file main_first, and afterward are forwarded to file buttonC.php or buttonD.php, depending on their action, and to play.php which stores the data about the player's action. If a player does not play in 30 sec it is forwarded to play_first.php, where an automatic action is taken and the data are stored in the same files. If it is not the first round, main_play.php will be included. The difference between the two is that after the first round we also need to present the actions and payoffs of the player and his/her neighbors in the previous round. This is done through files you.php and neighbor1.php,..., neighbor8.php. The automatic action taken is also different. Therefore we use other file, play_auto.php, for the automatic action. Once the player plays, the content of his/her usuarioNlock file is changed and the part main_played.php is included. Notice that usuarioNlock is both input and output file for this part of the system. Therefore, while in main_played, the content of these files is checked once per second and the next round starts as soon as the contents of all of them are changed back to "notplayed" (by the background process) . All the input files are also updated by the background process at this moment and the part main_play.php is included. This way the new round starts.

Before the players can start the game they have to login and read the tutorial. There are a few synchronization points. First we wait for everybody to login, then we wait for everybody to read the tutorial before every part of experiment.

All parts

The files present in every part of the experiment are:

- index.php — the main file which redirects clients to the current stage of the experiment. If the player is not logged in and the background process is started, it redirects the client to the login module. If the client already tried to login it checks the password and afterward redirects it to the first page of the experiment. If the username is SuperUser it is redirected to the monitoring screen. If the player is already logged in, he/she is redirected to the current stage of the experiment, which is kept in the PHP session variable "step".
- main.php — is the main file throughout the stage of experiment when the game is played. It shows the playing screen, redirects to the main_first.php in the first round, to main_play.php if the player needs to make an action, and to main_played.php when the player has already played and is waiting for everybody to play. See Fig. C.3.

- neighbor1.php,..., neighbor8.php and you.php — are files used by main_play.php to show the action and payoff of the neighbors and you.php shows the payoff and action of the player herself.
- buttonC.php and buttonD.php — are used by main_first.php and main_play.php. They represent the buttons to choose from. Pressing one of them triggers the module play.php
- play.php — writes the information about the players action in the appropriate files. Its output are files: usuarioNlock (with the action of the player), usuarioNtimestop (with the time when the action is taken) and the file usuarioNplayedby (where it stores whether the player played herself)
- play_first,php, play_auto.php — are activated if the player does not play in 30 seconds. The modules automatically choose the action and then give the same output as the module play.php. The automatic action of play_first is randomly chosen between C and D with the same probability. The module play_auto.php plays the previous action of the player with 80% chance and the opposite one otherwise.
- ready.php — This file is used for synchronization. It is triggered when the player reads the tutorial. It puts in the file usuarioNready an indication that the player has read the tutorial and is now ready to play. This information is then processed by the background process.
- wait.php — The second function used for the synchronization. It waits for all the players to read the tutorial. When this happens the background process will change the contexts of the file /data/started, the file wait.php will read that and let the player proceed.
- sumain.php — The program for monitoring the experiment. It shows a table with the current status of all players. The color of the field tells in what stage of the experiment the player is:
  - grey: not logged in
  - green: logged in, but did not read the tutorial
  - white: read the tutorial, did not play
  - yellow: played yellow
  - blue: played blue

  The meanings of the colors are explained in the legend on the top of the screen. Below is the table with the usernames of the players, their payoffs in the previous round and total earnings in all previous rounds in that part of experiment. In experiment 1 and experiment 2 the distribution of the players is exactly the same as in the experiment. This means that if two players are neighbors in the monitoring table, they are also neighbors in the experiment. However in the control part the distribution of the players in the monitoring table is arbitrary.

- `stop.php`, `plus5rounds.php` — They are used by `sumain.php` to stop or extend the experiment.

Part I

The files exclusive of Part I are:

- `waitlogin.php` — It is used for synchronization at the very beginning when users are logging in. It checks the context of the file `allloggedin`. When all players are logged in, the background process will change the content of this file and `waitlogin.php` will let the players proceed.
- `tutorial1.php`, ..., `tutorial9.php`, `first.php`, `part1.php` and `checktutorial8.php` — are the tutorial programs which present the screens with the rules of the game. File `tutorial8.php` contains a small test to check the understanding of the rules by the players. File `checktutorial8.php` checks if the answers are correct and writes the obtained information in files `/exp2/data/usuarioNanswers`.
- `login.php`, `login2.php`, `logout.php`, `checkuser.php` — are files for logging in. The file `login.php` presents the log in screen, where the player should type his username and password. These are then forwarded to `checkuser.php`: if the username and password are correct the player can proceed to the tutorial. If not, the player is forwarded to `login2.php`, where he/she can try again.
- `part1end.php` — presents the screen after part I has finished. After pressing the button the player is redirected to the control part.

Part II

The files exclusive of the control part are:

- `unset_control.php` — is the first function in the control part. It sets the current step of the experiment to be `part2tutorial1.php`, then forwards this information to `index.php` in the control folder. This is the cause why the control of experiment is no longer in `exp1/index.php`.
- `part2tutorial1.php`, `part2tutorial2.php` — are tutorial programs for the second part. They give information about the differences between Part I and Part II.
- `main_control.php` — which is different from `main.php` from the exp1 only in lines 29 and 30. Unlike Part I and Part II, here it reads neighbors from different files every round.
- `part1end.php` — presents the screen after Part II has finished. After pressing the button the player is redirected to Part III.

Part III

The files exclusive of Part III are:

- unset_exp2 — like unset_control.php, this is the first function in Part III. It sets the current step of the experiment to be part3.php, then forwards this information to index.php, which is located in the exp2 folder. This is why the control of experiment is no longer in file control/index.php.
- part3.php, part3.end.php — display the screen with instructions.
- questions1.php, write.php — are the programs for the questionnaire. The programs questions1.php presents the screen with questions and forwards the answers to write.php, which writes the obtained answers to the files
/exp2/usuarioNanswers.
- waitdaemon.php — waits for the background process to finish and then lets the program proceed to the last screen. When the background process finishes it changes the content of file finished. Waitdaemon.php reads this file and if its content is "finished" it lets the player proceed, otherwise it asks him/her to wait.
- last.phpN and sulast.php present the last screens of the experiment. The program last.php presents the player's earnings to the players and a good bye message, and sulast.php presents the last screen to the monitoring client with the information about the total earnings of all players.

C.5.3 Background process

Calculate is the background program which controls the experiment, makes calculations and stores the data in permanently. There are three different background processes, one for each part of the experiment. In the first part, the background process is calculate_exp1.py and it is in the folder PD/PDexp/exp1/data/. At the beginning it changes the context of the file daemonstarted to allow users to log into the experiment. Afterward it waits for all players to login by checking the content of files usuario1loggedin,..., usuarioNloggedin. When the experiment is started the content of these files is "notloggedin". Once the player logs in the content of his file is changed to "loggedin". When the content of all these files is "loggedin" the background process changes the content of file allloggedin, allowing players to proceed to the introductory tutorial. Afterwards it waits for all players to read the tutorial, by checking the content of files usuario1ready,..., usuarioNready. When the content of all these files is "ready", it changes the content of file started, allowing players to start playing. It reads the current round from file round, then the total number of rounds from the file roundsnumber and the number of players from file numberofusers. Then it starts the main loop of the process. It waits for all players to play. Once they all have played, the content of all `usuario1lock`,..., `usuarioNlock` files is going to be “C” or “D”. Then `calculate` copies all files to `usuario1move`,..., `usuarioNmove`, reads the neighbors from the files `usuario1`,...,`usuarioN`, and calculates the payoffs which are written into the files `usuario1score`,..., `usuarioNscore`. Furthermore, it calculates the times it took the players to play (reads files `usuario1timestart`, ..., `usuarioNtimestart` for the starting time and `usuario1timestop`, ..., `usuarioNtimestop` for the end time), then reads files `usuario1playedby`, `usuario2playedby`, ..., `usuarioNplayedby` and writes the full information into files `usuario1history`, ..., `usuarioNhistory` (round, action, payoff, time of play and who played). At the end, it increases the number of rounds by one and resets the files `usuario1lock`,..., `usuarioNlock` to `notplayed`, therefore allowing the player to play again and looping back to the beginning of the cycle. In case that after some time ($ms_{python} > 30s$) there remain players who have not played, the program plays for them (choosing the previous move with 80% chance and the opposite one with 20%) and then does the same calculation. See the scheme of this module in Fig. C.4.

In Part III of the experiment (experiment 2), this file is the same except that it does not wait for everybody to login at the beginning, just to read the tutorial. The file in Part II (control) is also different when reading the neighbors files, since it reads neighbors from a different file every round. The names of these files are `usuarioN_rX`, where N is the ordinal number of the player and X is the ordinal number of the round.

C.5.4 Data file generation

These programs are located in `PD/PDexp/'experiment'/data`. There are three programs in each part of the experiment:

- `makefiles.py` — generates the data files for storing the information needed throughout the experiment. The data files it generates (with the value it puts in the file in brackets) are: `firstround` (first), `log` (), `log_php`(), `round` (1), `daemonstarted` (not), `allloggedin` (not), `totaltotal` (0), `started` (notstarted) and for each player: `usuarioNhistory` (Round Move Score Time[ms] PlayedBy), `usuarioNlock` (notplayed), `usuarioNscore` (0), `usuarioNtotalscore` (0), `usuarioNmove` (), `usuarioNplayedby` (), `usuarioNtimestarted` (1.5), `usuarioNtimeend` (2.3), `usuarioNready`(), `usuarioNloggedin` (no).
- `cleanfiles.py` — resets the content of the files to its value at the beginning of the experiment, so the experiment cannot be started again without deleting the data files produced. The files it resets (with the values it puts in brackets) are: `daemonstarted` (not), `started` (notstarted), `allogedin` (not) and for all the players, `usuarionNready` (notready), `usuarioNloggedin`(no).
- `makenet.py` — generates files with the lists of the neighbors of each player, the name of the file being the name of that player. This means that the file

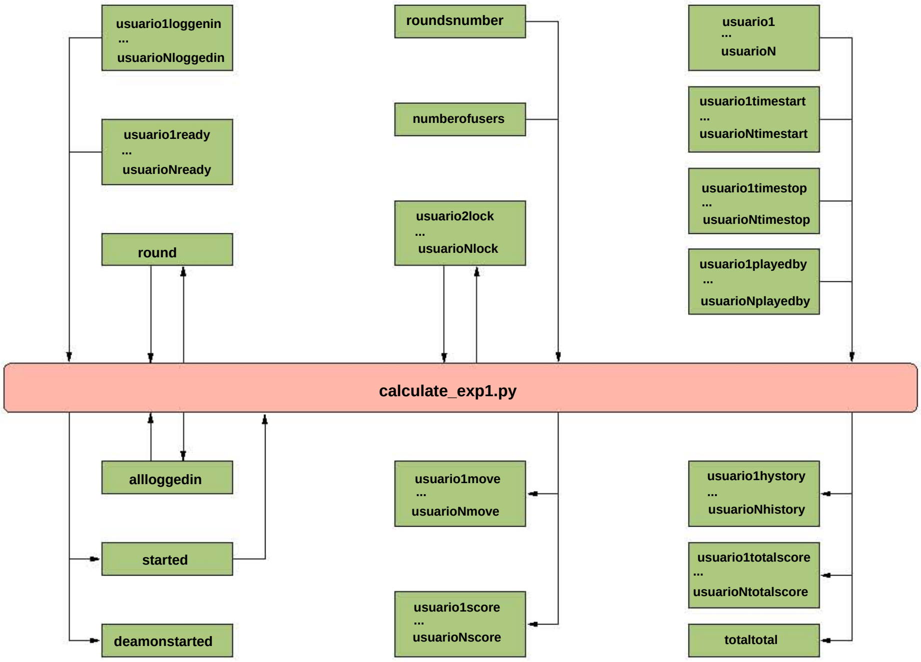


Figure C.4: *Scheme of the background process.* Input and output files. Notice that `round` and `usuario1lock`,..., `usuarioNlock` are both input and output files. At the beginning of the loop they are input files and at the end they are output. Although files `allloggedin` and `started` also appear as input and output files, they are essentially only output files. Their values are read only in case we need to rerun the background process in the middle of experiment.

`usuario34` contains the list of 8 players who are his/her neighbors. The file uses four parameters: two for the size of the network: N and M, and two for generating the number of rounds: minimal number of rounds R_{min} and maximal number of rounds R_{max}. At the beginning it generates the number of rounds as a uniform random number between R_{min} and R_{max}, then it reads the players usernames from the file `userlist.php`, shuffles and arranges them in the network of size $N \times M$. Furthermore it determines who are the neighbors of every player in this network and writes the list of neighbors for each player in the file with the name of that player. In the control part this function is called `makenet_control.py` and it generates a different set of neighbors for each round, which are stored in files `usuarioN_rY`, where N is the ordinal number of player and X is the ordinal number of round.

C.5.5 Client side scripts

There are a few scripts which should be run on the client side. These scripts are in folder PD/client_scripts. The whole folder should be copied to the clients' computers. In the normal mode of the experiment it is just the script experiment.sh. This file starts Opera in kiosk mode and excludes options to exit the kiosk mode and to use special keys.

The other files in this folder are used for the robot mode of experiment:

- robot_make.sh — creates the scripts for running the robots. As input it uses the web address of the starting page of the program. Usage:

  ```
  ./robot_make.sh http:/path_on_the_server/PD/PDrobot/exp1/
  ```

  As output, it produces the programs robot1.sh,..., robot8.sh, which are used to start the robots. For running this file, the file robot.py is also necessary.

- setopera.sh — creates the files necessary for setting the properties of Opera (this might not be necessary, since the only property it sets is a default web page, should be checked!) uses the file setopera.py and produces the file setopera_running.sh, which should be run to set Opera on the clients.

C.6 Robot mode

C.6.1 Tests with robots

A typical test consisted of 196 computer terminals which had 196 different accounts on the main computer and another server which was within the same network. The account names should be experiment1, experiment2,...,experiment196. There was one super user account which had permission to log into any other account without password. We call this "robot account". The folder called client_scripts should be copied in the home directory of the robot account. All the necessary scripts for running the robots are in this folder.

To start the test with the robots, the same procedure as for starting the experiment should be applied (look at the manual) except that instead of starting everything from folder PDexp, it has to be done from folder PDrobot.

Once the experiment is started, the following scripts: robot1.sh, robot2.sh, ..., robot8.sh (which are in folder clients_scripts on the robot account) should be started from the robot account. These scripts start Opera through the X-window System on 196 accounts on the different computer. As the Opera windows will all be on the computer where the script is started, it is recommended to start each of the scripts from a different computer. To generate these scripts for your system, you should use the script robot_make.sh which uses the file IPaddresses.txt with the IP address of all computers and as a parameter the web address on the server where the robot scripts are. This means that this script is started in the following way:

```
./robot_make.sh  http://path_on_the_server/PD/PDexp/exp1/
```

For running this script the file `robot.py` is also necessary. Once run, the robots should start to login and play automatically. If they have problems logging in or the server seems overloaded, adjust the parameters on the server again. For our system the above parameters worked. Notice that if the test is working on one system, that does not mean it will work on other system (for example, when players are logging in from different locations outside the internal network).

C.6.2 Creating robots

The difference between robot-mode and normal-mode files is just in the interaction part. Wherever the program is waiting for the action of the player, in robot mode an automatic action should be taken. Therefore, every button needs to be replaced with the redirection. This means that the following lines (for making the button) should be deleted:

```
 Pinche <input type="submit" value="Aqu&iacute;" class="btn"
name="submit"> para continuar.
```

And these lines (for automatic redirection) should be added at the end of the same files:

```
<script type="text/javascript">
setTimeout('document.form1.submit();',1000);
</script>
```

This change should be made in the following files:

- in `PD/PDrobot/exp1`
  - `login.php`, `tutorial1.php`, `tutorial2.php`, `tutorial3.php`, `tutorial4.php`, `tutorial5.php`, `tutorial6.php`, `tutorial7.php`, `tutorial8.php`, `checktutorial8.php`, `tutorial9.php`, `part1.php`, `part1end.php`
- in `PD/PDrobot/control`
  - `part2tutorial1.php`, `part2tutorial2.php`, `part2end.php`
- in `PD/PDrobot/exp2`
  - `part3.php`, `part3end.php`, `questions1.php`

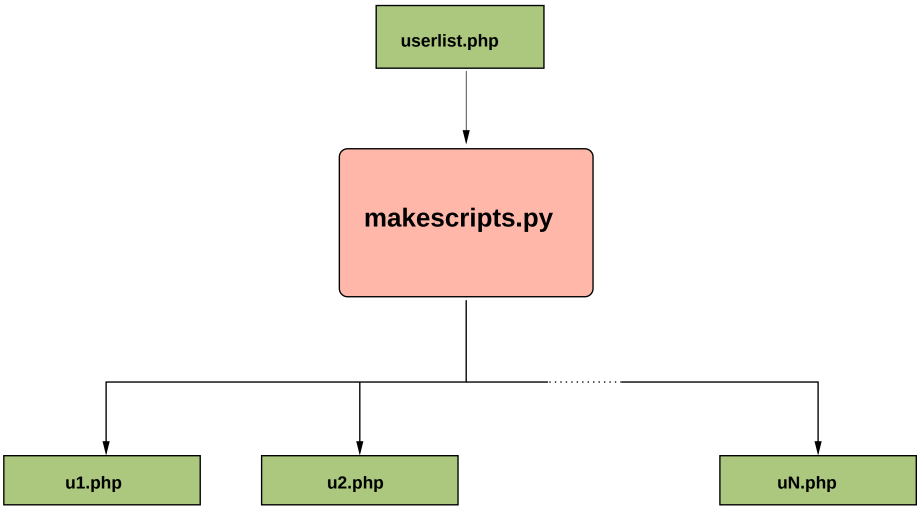


Figure C.5: *Graphic scheme of the file* `makescripts.py`. As input it uses the file with all usernames and passwords and generates the files for automatic logging in. Although the input and output files are PHP scripts, in this context they are not used as PHP scripts. The input file `userlist.php` is a PHP script for security reasons and the output files are going to be used as PHP scripts later.

Besides, the files for automatic logging in should be added in folder PD/PDrobot/exp1. The names of the files are: `u1.php`, `u2.php`,..., `u196.php` and they are generated by the script `makescripts.py`, which uses the file `userlist.php` where the usernames and passwords are (see Fig. 5).

The programs `main_first.php` and `main_play.php` require some more substantial changes. In these files there are two buttons from which one should be chosen at random and "pushed". For choosing randomly between the two buttons we add following line:

```
$rand_button=rand(0, 1);
```

in row 6, in the first PHP block, and instead of line:

```
<?php include_once("buttonC.php")?>
```

we include the following lines, that will mark cooperation button if the random choice was 0:

```
<?php  if($rand_button==0) include_once("buttonC.php")?>
```

Similarly, for the defection button:

```
<?php include_once("buttonD.php")?>
```

we include these lines, that will mark defection button if the random choice was 1:

```
<?php  if($rand_button==1) include_once("buttonD.php")?>
```

Once determined in `main_first.php` and `main_play.php` which button will be included, in files `buttonC.php` and `buttonD.php` the following button should be “clicked” at a random time within 30 seconds. This is achieved in `buttonC.php` by deleting:

```
<input type="submit" value="" style="background-color:#0099CC;
width:100; height:50">
```

And in `buttonD.php`, the following line should be deleted:

```
<input type="submit" value="" style="background-color:#FFFF33;
width:100; height:50">
```

On the other hand, the lines for automatic redirections:

```
<script type="text/javascript">
var ran_number=Math.floor(Math.random()*30);
setTimeout('document.login.submit();',ran_number*1000);
</script>
```

should be added at the end of the files `buttonC.php` and `buttonD.php`.

C.7 Some warnings

- In the files where it is needed, `session_start()` has to be the first line in a file, not even a comment can be written before that.
- When writing the robot, the control should be automatically redirected to other function, but the button has to be removed, otherwise the automatic redirection will not work.

D

Additional material on the group size experiment (Chapter 5)

D.1 English translation of the instructions

We include below an English translation of the instructions as they were read and distributed to the participants in the experiment. We present the version for the case in which there was computer intervention to increase highly cooperative contexts. The instructions for the sessions without random computer intervention are identical, except that the paragraph after **Random intervention** does not exist. The version provided is for the case of groups of five players. The instructions for the sessions with smaller number of players are identical except that the complete payoff tables are different and the exchange rate also varies as indicated in the main text.

Instructions

Thanks for participating in this experiment, which is part of a research project in economics in which we try to understand how decisions are made, but where a particular behavior is expected of you. From this moment the experiment begins. Please be quiet for its whole duration. turn of your cell phone, and remember that no material alien to the experiment is allowed (including pens, pencils and paper). Your earnings depend on your decisions and the decisions of other participants. In addition you receive as a payment 10 euros just for participating. From now and until the end of the experiment you are not allowed to communicate with other participants. If you have any questions, please raise your hand and an instructor will answer your questions in private.

Please do not ask questions out loud!

The experiment

The experiment consists of an undetermined number turns or rounds, and it will last around an hour and never more than two. The rules are the same for all participants and on every round. Throughout the experiment you will be part of the same group of 5 participants (5 including yourself). None of you will know who are the other 4 participants with whom you play, and in particular, they need not be the people close to you.

Rounds

In every round you will see two buttons on the screen, corresponding to the actions A and B, of which you must choose one by clicking the mouse on it. You have 10 seconds to do so, and you necessarily have choose one of two options (the experiment will be stopped if a participant does not press one of them). When all the players have chosen, you will see the information on the number of players who have chosen A, the number of those who have chosen B and your earnings on that round.

The earnings on each round are computed as follows:

- If you choose A: you receive 7 ECU for each player that chooses A A (excluding yourself) and nothing for each player choosing B.
- If you choose B: you receive 10 ECU for each player choosing B and nothing for each player choosing B.

The following table shows all the possibilities for your personal earnings:

	The others choose:				
	AAAA	AAAB	AABB	ABBB	BBBB
You choose A	28	21	14	7	0
You choose B	40	30	20	10	0

while the earnings for the group as a whole are:

Decisions	AAAAA	AAAAB	AAABB	AABBB	ABBBB	BBBBB
Group earn.	140	124	102	74	40	0

The screen with the information on what you and the other players have done and yours and theirs earnings will show itself for 20 seconds. You must press “OK” to go to the following round; the screen for the next round will be shown when all players press “OK”.

Random intervention

Occasionally, and in completely random way, the computer can change your decision or that of the other player. The program does not report this change when it occurs. In such cases the payment is calculated as if the player concerned had actually taken the decision that the computer chose. The frequency with which this happens is low: your actions will remain unchanged for at least an 85% of the time.

Payments

After the last round, the ECUs you obtained in each round will be added to obtain your total earnings, so you need to pay full attention until the end. The ECUs will be converted to euros so that 100 ECUs will be converted to one Euro. Additionally you will receive 10 Euros just for participating.

End of instructions

D.2 Comparison of the treatments with and without computer interventions

In order to see how computer interventions influenced the decision making process of the players, we compared the probabilities of cooperating in different context for the treatments with and without computer interventions (moody conditional cooperation strategy). In Figure D.1 we present this comparison. We see that the difference between the two treatments in decision making are minor and that the global trends are the same in both kind of treatments.

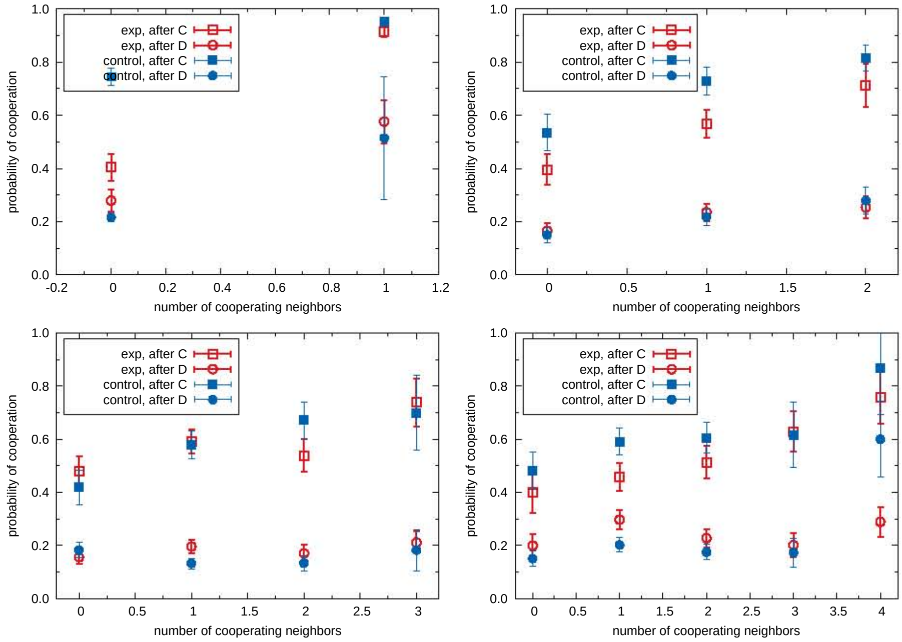


Figure D.1: *Comparison of moody conditional cooperators in experiments with and without computer interventions.* Probability that an individual cooperates after having cooperated (squares) and after having defected (circles) in the previous round, for groups of 2 (top left), 3 (top right), 4 (bottom left) and 5 (bottom right) people. We present the results separately for the experimental treatment with computer interventions (red) and the control treatment without computer interventions. The error bars show the 95% confidence intervals and are calculated as $1.96 * \sqrt{p(1-p)/n}$, where n is the number of samples, and p is probability of cooperation.

E

Additional model on the group size experiment (Chapter 5)

E.1 Model

The model presented here was developed by B. Eke.

In order to shed further light on our results, we resorted to the development a GLMM that helps understand our observations and identify the significant factors that influence them. To this end, it has to be taken into account that our data contains repeated measures on each subject of a binary variable. Let y_{ijt} be the response of the subject i in group j at time t. Let $y_{ijt} = 1$ if this subject cooperates at time t and 0 otherwise for all i, j and t. Then $y_{ijt} \sim$ Bernoulli(p_{ijt}). By the nature of the experiment, the subjects are nested in groups. Thus, a model needs to take into account the nested structure of the data, and the repeated measures on the subjects.

Our concern with respect to dependency is the repeated measures on the same subject. First, the observations on the same subject are correlated just because they are decisions of the same person. This is also known as within subject variability. Second, the observations close in time, on the same actor, are more likely to be highly correlated as oppose to the observations further apart. We interpret this as latent generosity with a time component. Third, another source of variation is the latent component of the individual reaction to the number of cooperative actions observed in the group in the previous round. We can interpret this as latent reciprocity. These latent effects then measure “between-subject” variability.

Before introducing the model we finally chose as the best for our data, let us point out that, in alternative specifications, we checked for effects of major and gender, without finding any significant effect. Most importantly, we tested the dependence on whether the group was manipulated by the computer or not, again finding no differences (see Materials and methods below). With these inputs, we finally proposed the following model:

$$\begin{aligned}\text{logit}(p_{it}) &= \sum_{l=2}^{5} \beta_l \chi(\text{size}_{il}) \\ &+ \sum_{l=2}^{5} \beta_l^{\text{C}} \text{LagCoop}_{it} \chi(\text{size}_{il}) \\ &+ \beta^{\text{A}} \text{LagAction}_{it} \\ &+ \alpha_i + \sum_{l=2}^{5} \gamma_{il} \text{LagCoop}_{it} \chi(\text{size}_{il}) + \xi_{it}, \end{aligned} \tag{E.1}$$

where p_{it} is the probability of cooperation of subject i at time t, and the factors that affect it are as follows: $\chi(\text{size}_{il})$ is the characteristic function corresponding to the group size of subject i, that is, $\chi(\text{size}_{il}) = 1$ if subject i played in a group of size l and 0 otherwise; LagCoop_{it} is the number of cooperative actions received by subject i at time $t-1$; LagAction_{it} is equal to 1 if the subject cooperated in the previous round and 0 otherwise, and β_l and β_l^C, $l = 2 \dots 5$, and β_l^A are the parameters of the fixed effects. On the other hand α_i is the latent cooperativeness of each subject, and γ_i is her latent reciprocity (the individual random variation in the response to perceived cooperation). Individual latent effects follow normal distributions: $\boldsymbol{\alpha} \sim N(\mathbf{0}, \Sigma)$, where, $\Sigma = \sigma_{\boldsymbol{\alpha}}^2 \mathbb{I}$, where $\mathbb{I}$ is the identity matrix, and analogously $\boldsymbol{\gamma} \sim N(\mathbf{0}, \Sigma_{\boldsymbol{\gamma}})$, where, $\Sigma_\gamma = \sigma_\gamma^2 \mathbb{I}$. In addition, we have the repeated measure structure modeled as AR(1) structure through the ξ_{it} term, where $\xi_{it} = \rho_R \xi_{i,t-1} + u_{it}$, where $\boldsymbol{u}$ is a vector of random variables with variance $\sigma_{\boldsymbol{u}}$. That is, there is a random component on the left hand side of the model which measures the "within subject" variability. The structure of the covariance matrix for this effect is given by a symmetric matrix, R, whose (ij)-entry is $\sigma_u \rho_R^{|i-j|}$.

E.2 Model results.

The model captures well the observations from the experiment, as can be seen from the comparison between the experimental data and the model predictions in Figs. 5.1 and 5.2. The agreement is particularly good for the cooperation level, as this magnitude can be obtained directly from the model, whereas there are small discrepancies in the slope of the conditional cooperation lines, mostly for the highest cooperative contexts. These discrepancies can be understood as the estimation of these lines is an indirect product of the model. Another feature that is confirmed is the clear dependence on the players' own previous action, their 'moodiness', an aspect to which we come back to below.

We first discuss the latent factors in the model. The corresponding variance components estimated within our model are represented in Table E.1. The corresponding $p-$values are obtained by applying the log-likelihood ratio significance test (LRT) on the boundary of variance parameter space as in

Turning now to the fixed effects, the predicted values for the corresponding parameters are presented in Table E.2. The estimates and their p-values give us the individual significance levels. The type 3 tests collect the information on overall significance of the effects. Based on the Table E.2, we have size, LagCoop and LagAction as highly significant covariates at 1% significance level). Other relevant results include, for instance, the fact that the size of the groups is important for cooperative attitudes. As Table E.2 shows, the parameter for the baseline cooperative attitude in a group of size 2 is larger and statistically different from all the others. In turn, the baseline cooperative attitude is not statistically different between sizes 3 and 5. The conditional cooperation declines monotonically with group size, although the differences become smaller as size increases, and the coefficient is still statistically different from zero even at the largest size. This is an interesting point that might be useful to understand why cooperation is more fragile in large groups, which could in turn explain why social groups often evolve punishment strategies directed solely at deviators, as in (Boyd and Richerson 1992). Finally, the result that LagAction is relevant points to the dependence of actions on what occurred at the previous round. In this respect, it is important to mention that we also tried other models in which dependence on two previous time steps was included, and we found that this was not significant. Therefore, the dependence on the player's own previous choice is enough to capture the results of the experiment, a finding that is in agreement with earlier work (Dal Bó and Frechette 2011; Fudenberg et al. 2012).

Table E.1: Results for the variance of the random effects. Shown are the estimates, their standard error and the log-likelihood ratio (LRT) p-value assessing their significance. From top to bottom, the table shows the results for the generosity, the reciprocity, and the two parameters of the AR(1) formalism.

	Estimate	SE	LRT $p-$value
$\sigma_{\boldsymbol{\alpha}}$	0.8590	0.1075	<0.0001
$\sigma_{\boldsymbol{\gamma}}$	0.3311	0.0394	<0.0001
ρ_R	-0.01971	0.0173	<0.0001
$\sigma_{\boldsymbol{u}}$	0.9021	0.0089	<0.0001

Table E.2: Results for the fixed effects. Shown are the estimates, their standard error and the p-value assessing their significance. The upper part of the table shows the estimates for β_i coefficients, $i = 2, \ldots, 10$. The second, third and forth parts of the table show significance test results for the different factors in the model. The tests are summarized in Methods.

Effect	Estimate	$p-$value
β_2	-1.4599	<0.0001
β_3	-1.6329	<0.0001
β_4	-1.7689	<0.0001
β_5	-1.5499	<0.0001
β_2^{C}	1.6310	<0.0001
β_3^{C}	0.4940	<0.0001
β_4^{C}	0.3762	<0.0001
β_5^{C}	0.2059	0.0143
β^{A}	0.5910	<0.0001

Type 3 Tests		
	F$-$value	$p-$value
Size	136.50	<0.0001
LagCoop × size	71.93	<0.0001
LagAction	195.68	<0.0001

Contrast Analysis		Bonferroni adj.
(Size)	t-value	p-value
2 vs. 3	0.61	0.4359
2 vs. 4	2.49	0.1162
2 vs. 5	0.19	0.6639
3 vs. 4	0.47	0.4918
3 vs. 5	0.16	0.6908
4 vs. 5	1.47	0.2272

Contrast Analysis		Bonferroni adj.
(Size*LagCoop)	t-value	p-value
2 vs. 3	61.50	<0.0001
2 vs. 4	98.35	<0.0001
2 vs. 5	117.69	<0.0001
3 vs. 4	0.85	0.3589
3 vs. 5	4.72	0.0311
4 vs. 5	2.32	0.1292

E.3 Statistical tests of significance.

The statistical tests used in the paper are the log-likelihood ratio (LRT) significance test of variance parameters, standard significance tests and type III test on fixed effects, and the contrast analysis for the levels of fixed effects. The log-likelihood ratio significance test is used for the variance parameters since the tested value, 0, in on the boundary of the parameter space of the variances. The theory and development behind this test is explained in (Self and Liang 1987). Basically, a low p-value indicates significant variance parameter, i.e., heterogeneity among the participants of the random effect. These results are presented in Table 1. The standard significance test and the type III tests test for the significance on the parameters, the former individually, and the latter jointly, for all levels of that variable. For example, consider the variable size. The first part of Table 1 presents the results of standard significance test results for β_2 , β_3, β_4 and β_5, which corresponds to sizes 2, 3, 4, and 5, respectively. The second part of the same table represents the joint significance of the size effect. i.e., $H_0 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0$ versus at least one is nonzero. The last test performed here is the contrast analysis. This procedure investigates the differences between the levels of the same variable. For example, again using size, in the previous tests we have considered differences from 0; now we are studying whether the effect of being in a group of size 2 is different than that of sizes 3, 4, and 5, respectively. Due to the multiple testing in this procedure the p-values are adjusted using Bonferroni adjustment, which adjusts the p-values by the number of tests performed. This adjustment is more on the conservative side, that is, we do not reject more often.

Publications

Part of the original content of this thesis appears in the following papers:

- Chapter 2:
  - *Social Experiments in the Mesoscale: Humans Playing a Spatial Prisoner's Dilemma*,
    Jelena Grujić, Constanza Fosco, Lourdes Araujo, Jose A. Cuesta, Angel Sánchez,
    PLoS ONE **5**, e13749 (2010).
- Chapter 3:
  - *On the coexistence of cooperators, defectors and conditional cooperators in the multiplayer iterated Prisoner's Dilemma*,
    Jelena Grujić,José A. Cuesta and Angel Sáchez
    Journal of Theoretical Biology, **300**, 299-308 (2012)
- Chapter 4:
  - *Consistent strategy updating in spatial an non-spatial behavioural experiments does not promote cooperation in social networks*,
    Jelena Grujić, Torsten Röhl, Dirk Semmann, Manfred Milinski and Arne Traulsen
    submitted
- Chapter 5:
  - *Reciprocity in multi-player prisoner's dilemma experiments leads to cooperation only in groups of two players*,
    Jelena Grujić, Burcu Eke, Antonio Cabrales, José A. Cuesta, Angel Sánchez,
    submitted

Resumen

Entender las interacciones entre las personas y sus contactos sociales es un problema clave para dilucidar la forma en la que funciona la sociedad y cómo ésta contribuye a la mejora del bienestar individual. El origen evolutivo de la cooperación entre individuos no emparentados es una cuestión sin resolver que afecta a varias disciplinas. Entre los distintos mecanismos propuestos para explicar cómo puede aparecer la cooperación destaca la existencia de una estructura en la población que determine las interacciones entre individuos. Muchos modelos han explorado analítica y computacionalmente los efectos de dicha estructura, sobre todo en el marco del Dilema del Prisionero, pero los resultados obtenidos dependen enormemente de muchos detalles, tales como el tipo de estructura considerada o la dinámica evolutiva. Por tanto, era preciso llevar a cabo trabajo experimental diseñado apropiadamente para identificar qué características de las que integran los modelos son las relevantes.

En esta tesis hemos investigado cómo la estructura espacial influye en la promoción de la cooperación. Para ello, diseñamos un experimento para estudiar la aparición de cooperación cuando las personas juegan al Dilema del Prisionero iterado. Los voluntarios que participaron en este experimento jugaron al Dilema del Prisionero en una red de tamaño considerable. Los paretros del experimento se escogieron para promover la cooperación en la mayor medida posible, partiendo de las predicciones de los modelos teóricos. Nuestros resultados indican que el nivel de cooperación no mejora por la existencia de una red, manteniéndose la fracción de cooperadores en un 20% aproximadamente. Estos resultados se pueden explicar a través de la existencia de heterogeineidad y de una estrategia de cooperación condicional generalizada, en la que la probabilidad de cooperar depende de la cooperación de los otros participantes en el juego y también de la acción previa del jugador. Nuestras conclusiones han tenido un gran impacto en la manera en la que la Teoría de Juegos en grafos se usa para modelar las interacciones humanas en grupos estructurados.

De hecho, nosotros mismos hemos propuesto un modelo basado en agentes en el que coexisten tres diferentes estrategias compatibles con las observaciones experimentales: cooperación, defección y cooperación condicionada generalizada. Consideramos grupos de $n = 2, 3, 4$ y 5 jugadores y calculamos los pagos para cada tipo de jugador en el equilibrio utilizanda cadenas de Markov. De esta manera, demostramos que para

los grupos de tamaño menor que $n = 4$ existe un punto interior en el cual las tres estrategias coexisten. La correspondente cuenca de atracción disminuye al aumentar el número de jugadores, mientras que para $n = 5$ no pudimos encontrar ningún punto de atracción interior. Finalmente, hemos visto que para el límite cuando n tiende a infinito, dicho atractor no existe.

Así pues, nuestros experimentos en red sugieren que la cooperación puede depender de la acción previa del jugador, pero al mismo tiempo hemos probado teóricamente que ese tipo de comportamiento no puede coexistir con jugadores que nunca cooperan y con cooperadores en grupos formados por más de 5 personas. Por ello, decidimos diseñar un experimento que reprodujese nuestro esquema teórico. As confirmamos la existencia de cooperadores condicionales y un nivel de cooperación bajo en grupos formados por más de dos miembros. Soprendentemente, hemos visto que el comportamiento de los jugadores en grupos de dos individuos es cualitativamente diferente a las situaciones donde este número es mayor. Nuestro experimento se prolongó durante 100 rondas, lo cual nos permitió estudiar el régimen a largo plazo. Cuando se juega al Dilema del Prisionero por parejas en esta situación, el nivel de cooperación, tras una caída inicial, se incrementa significativimente y llega a un nivel de más del 80 %.

Además, hemos reanalizado los datos del experimento de Traulsen et al. (2010), en el que los voluntarios jugaban al Dilema del Prisionero con sus cuatro vecinos más cercanos en una red de tamaño 4×4. El experimento tenía dos tratamientos: uno espacial, donde los jugadores tenían una posición fija en la red durante todo el experimento, y uno no espacial, en el cual los jugadores cambiaban sus posiciones en la red después de cada ronda. Analizamos estadísticamente las decisiones individuales y dedujimos con qué modelo o modelos de Teoría de Juegos evolutiva las podemos conectar. No encontramos ninguna diferencia entre ambos tratamientos. Sin embargo, las estrategias que usan los jugadores no corresponden con las que se suelen estudiar en Teoría de Juegos evolutiva. Finalmente, utilizando simulaciones numéricas, vimos cómo los mecanismos de actualización obtenidos en los experimentos no favorecen la cooperación en la estructura espacial.

Como apoyo a nuestras conclusiones, hemos comparado los resultado de experimentos diferentes. Aunque hay diferencias, ciertas características parecen ser universales. Así, el nivel de cooperacion se muestra bajo en todos los experimentos, a pesar de que muchos modelos teóricos predicen una promoción de la cooperacion, y la estructura de la población (la red) parece no tener ningun efecto sobre el nivel de cooperación. En todos los experimentos se observa cooperaci condicional generalizada, aunque tambi es posible describir el comportamiento observado con otras reglas, si bien de manera menos universal que con la anterior.

References

Aguiar F., Brañas-Garza P., Cobo-Reyes R., Jiménez N., and Miller L. (2009). Are women expected to be more generous? *Experimental Economics* **12**, 93–98.

Aktipis C. (2004). Know when to walk away: contingent movement and the evolution of cooperation. *JTB* **231**, 249–260.

Allen B., Traulsen A., Tarniţă C., and Nowak M. A. (2012). How mutation affects evolutionary games on graphs. *Journal of Theoretical Biology* **in press**.

Apicella C. L., Marlowe F. W., Fowler J. H., and Christakis N. A. (2012). Social networks and cooperation in hunter-gatherers. *Nature* **481**, 497–501.

Axelrod R. (1984). *The Evolution of Cooperation*. New York: Basic Books.

Axelrod R., and Hamilton W. D. (1981). The evolution of cooperation. *Science* **211**, 1390–1396.

Bains W. (2007). Random number generation and creativity. *Medical Hypotheses* **70**, 186–190.

Blume L. E. (1993). The Statistical Mechanics of Strategic Interaction. *Games Econ. Behav.* **5**, 387–424.

Boyd R., and Richerson P. J. (1988). The evolution of reciprocity in sizable groups. *J. Theor. Biol.* **132**, 337–356.

Boyd R., and Richerson P. J. (1989). The evolution of indirect reciprocity. *Soc. Netw.* **11**, 213–236.

Boyd R., and Richerson P. J. (1992). Punishment allows the evolution of cooperation (or anything else) in sizable groups. *Ethol. Sociobiol.* **13**, 171–195.

Brañas-Garza P. (2007). Promoting helping behavior with framing in dictator games. *Journal of Economic Psycholory* **28**, 477–486.

Brañas-Garza P., Cabo-Reyes R., Espinosa M. P., Jiménez N., Kovářík J., and Ponti G. (2010). Altruism and social integration. *Games and Economic Behavior* **69**, 249–257.

Brandt H., and Sigmund K. (2004). The logic of reprobation: assessment and action rules for indirect reciprocation. *J. Theor. Biol.* **231**, 475–486.

Buskens V., and Wessie J. (2000). Cooperation via Social Networks. *Analyse & Kritik* **22**, 44–74.

Camerer C. F. (2003). *Behavioral Game Theory*. Princeton: Princeton University Press.

Cassar A. (2007). Coordination and cooperation in local, random and small world networks: Experimental evidence. *Games Econ. Behav.* **58**, 209–230.

Cohen M. D., Riolo R. L., and Axelrod R. (2001). The role of social structure in the maintenance of cooperative regimes. *Rationality and Society* **13**, 5–32.

Croson R. T. A. (2006). Theories of commitment, altruism and reciprocity: evidence from linear public goods games. *Econ. Inquiry* **45**, 199–216.

Dal Bó P. (2005). Cooperation under the shadow of the future: Experimental evidence from infinitely repeated games. *Am. Econ. Rev.* **95**, 1591–1604.

Dal Bó P., and Frechette G. R. (2011). The Evolution of Cooperation in Infinitely Repeated Games: Experimental Evidence. *Am. Econ. Rev.* **101**, 411–429.

Dawkins R. (1976). *The Selfish Gene*. Oxford Univ. Pres.

Efron B., and Tibshirani R. (1993). *An Introduction to the Bootstrap*. New York: Chapman and Hall.

Eigen M., and Schuster P. (1977). A principle of natural self-organization. *Naturwissenschaften* **64**, 541–565.

Fehl K., van der Post D. J., and Semmann D. (2011). Co-evolution of behaviour and social network structure promotes human cooperation. *Ecol. Lett.* **14**, 546–551.

Fehr E., and Gächter S. (2000). Cooperation and Punishment in Public Goods Experiments. *Amer. Econ. Rev.* **90**, 980–994.

Fischbacher U. (2007). z-Tree: Zurich toolbox for ready-made economic experiments. *Exp. Econ.* **10**, 171–178.

Fischbacher U., and Gächter S. (2010). Social preferences, beliefs, and the dynamics of free riding in public goods experiments. *Am. Econ. Rev.* **100**, 541–556.

Fischbacher U., Gächter S., and Fehr E. (2001). Are people conditionally cooperative? Evidence from a public goods experiment. *Econ. Lett.* **71**, 397–404.

Fisher R. A. (1930). *The Genetical Theory of Natural Selection*. Oxford: Clarendon Press.

Friedman J. W. (1971). A non-cooperative equilibrium in supergames. *Rev. Econ. Stud.* **38**, 1–12.

Fudenberg D., and Maskin E. (1990). Evolution and cooperation in noisy repeated games. *Am. Econ. Rev.* **80**, 274–279.

Fudenberg D., Rand D., and Dreber A. (2012). Slow to Anger and Fast to Forgive: Cooperation in an Uncertain World. *Am. Econ. Rev.* **in press**.

Gächter S. (2007). Conditional cooperation: behavioral regularities from the lab and the field and their policy implications. In Frey B. S., and Stutzer A. (Eds.), *Economics and Psychology: A Promising New Cross-Disciplinary Field*, pp. 19–50. MIT Press.

Gigerenzer G., and Selten R. (Eds.) (2001). *Bounded Rationality: The Adaptive Toolbox*, Cambridge. MIT Press.

Gómez-Gardeñes J., Capillo M., Floría L. M., and Moreno Y. (2007). Dynamical organization of cooperation in complex networks. *Phys. Rev. Lett.* **98**, 108103.

Gracia-Lázaro C., Cuesta J., Sánchez A., and Moreno Y. (2012). Human behavior in Prisoner's Dilemma experiments suppresses network reciprocity. *Sci. Rep.* **2**, 325.

Gracia-Lázaro C., Ferrer A., Ruiz G., Tarancón A., Cuesta J., Sánchez A., and Moreno Y. (2012). Heterogeneous networks do not promote cooperation when humans play a Prisoners Dilemma. *PNAS* **forthcoming**.

Haldane J. B. S. (1932). *The Causes of Evolution*. London: Longmans, Green & Co.

Hamilton W. D. (1964a). The genetical evolution of social behaviour I. *J. Theor. Biol.* **7**, 1–16.

Hamilton W. D. (1964b). The genetical evolution of social behaviour II. *J. Theor. Biol.* **7**, 17–52.

Hardin G. (1968). The Tragedy of the Commons. *Science* **162**, 1243–1248.

Hardin R. (1971). Collective action as an agreeable n-prisoners' dilemma. *Behav. Sci.* **16**, 472–481.

Hauert C. (2002). Effects of space in 2×2 games. *Int. J. Bifurc. Chaos* **12**, 1531–1548.

Hauert C., de Monte S., Hofbauer J., and Sigmund K. (2002). Volunteering as Red Queen Mechanism for Cooperation in Public Goods Games. *Science* **296**, 1129–1132.

Hauert C., and Doebeli M. (2004). Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature* **428**, 643–646.

Hegselmann R., and Flache A. (2000). Rational and adaptive playing: A comparative analysis for all possible Prisoner's Dilemmas. *Analyse & Kritik* **22**, 75–97.

Helbing D. (1992a). Interrelations between Stochastic Equations for Systems with Pair Interactions. *Physica A* **181**, 29–52.

Helbing D. (1992b). A mathematical model for behavioral changes through pair interactions. In Hagg G., Mueller U., and Troitzsch K. G. (Eds.), *Economic evolution and demographic change*, Berlin, pp. 330–348. Springer–Verlag.

Helbing D., and Yu W. (2008). Migration as a mechanism to promote cooperation. *Advances in Complex Systems* **11**, 641–652.

Helbing D., and Yu W. (2010). The future of social experimenting. *Proc. Natl. Acad. Sci. USA* **107**, 5265–5266.

Henrich J., Boyd R., Bowls S., Camerer C., Fehr E., Gintis H., and McElreath R. (2001). In Search of Homo Economicus: Behavioral Experiments in 15 Small-Scale Societies. *American Economic Review* **91**, 73–78.

Hofbauer J., and Sigmund K. (1990). Adaptive Dynamics and Evolutionary Stability. *Applied Mathematics Letters* **3**, 75–79.

Hofbauer J., and Sigmund K. (1998). *Evolutionary Games and Population Dynamics*. Cambridge: Cambridge University Press.

Hofbauer J., and Sigmund K. (2003). Evolutionary Game Dynamics. *Bull. Am. Math. Soc.* **40**, 479–519.

Jacquet J., Hauert C., Traulsen A., and Milinski M. (2011). Shame and honour drive cooperation. *Biology Letters* **7**, 899–901.

Kahneman D., Slovic P., and Tversky A. (Eds.) (1982). *Judgment under uncertainty: Heuristics and biases*, New York. Cambridge University Press.

Karlin S., and Taylor H. M. (1975). *A First Course in Stochastic Processes* (Segunda ed.). New York: Academic Press.

Kilgour D. M. F. N. M. (1988). A Taxonomy of All Ordinal 2 x 2 Games. *Theory and Decision* **24**, 99–117.

Killingback T., and Doebeli M. (1996). Spatial Evolutionary Game Theory: Hawks and Doves Revisited. *Proc. R. Soc. Lond. B* **263**, 1135–1144.

Kirchkamp O., and Nagel R. (2007). Naive learning and cooperation in network experiments. *Games Econ. Behav.* **58**, 269–292.

Kovářík J., Brañas-Garza P., Cabo-Reyes R., Espinosa M. P., Jiménez N., and Ponti G. (2012). Prosocial behavior and degree heterogeinity social networks. *Physica A* **391**, 849–853.

Kümmerli R., Colliard C., Fiechter N., Petitpierre B., Russier F., and Keller L. (2007). Human cooperation in social dilemmas: comparing the Snowdrift game with the Prisoner's Dilemma. *Proc. R. Soc. B* **274**, 2965–2970.

Kurzban R., and Houser D. (2005). Experiments investigating cooperative types in humans: A complement to evolutionary theory and experiments. *Proc. Natl. Acad. Sci. USA* **102**, 1803–1807.

Langer P., Nowak M. A., and Hauert C. (2008). Spatial invasion of cooperation. *J. Theor. Biol.* **250**, 634.

Ledyard J. O. (1995). Public goods: A survey of experimental research. In Nagel J. H., and Roth A. E. (Eds.), *Handbook of experimental economics*, pp. 111–251. Princeton University Press.

Levin I. P., and Gaeth G. J. (1988). How consumers are affected by the framing of attribute information before and after consuming the product. *Journal of consumers research* **15**, 374–378.

Licht A. N. (1999). Games commissions play: 2x2 games of international securities regulation. *Yale J. Int. Law* **24**, 61–125.

Martínez-Vaquero L. A., Cuesta J. A., and Sánchez A. (2012). Generosity pays in the presence of direct reciprocity: A comprehensive study of 2 x 2 repeated games. *PLoS ONE* **7**, e35135.

Maynard Smith J. (1964). Group Selection and Kin Selection. *Nature* **201**, 1145–1147.

Maynard Smith J. (1982). *Evolution and the Theory of Games*. Cambridge: Cambridge University Press.

Maynard Smith J., and Price G. (1973). The logic of animal conflict. *Nature* **246**, 15–18.

Maynard Smith J., and Szathmary E. (1995). *The Major Transitions in Evolution*. Oxford: Freeman.

Meloni S., Buscarino A., Fortuna L., Frasca M., Gómez-Gardeñes J., Latora V., and Moreno Y. (2009). Effects of Mobility in a Population of Prisoners Dilemma Players. *PRE* **79**, 067101.

Milinski M., Semmann D., Bakker T. C. M., and Krambeck H. J. (2001). Cooperation through indirect reciprocity: image scoring or standing strategy? *Proc. R. Soc. Lond. B* **268**, 2495–2501.

Milinski M., Semmann D., and Krambeck H. J. (2002). Reputation helps solve the tragedy of the commons. *Nature* **415**, 424–426.

Milinski M., Semmann D., Krambeck H. J., and Marotzke J. (2006). Stabilizing the Earths climate is not a losing game: Supporting evidence from public goods experiments. *PNAS* **103**, 3994–3998.

Moran P. A. P. (1962). *The Statistical Processes of Evolutionary Theory*. Oxford: Clarendon Press.

Moyano L. G., and Sánchez A. (2009). Evolving learning rules and emergence of cooperation in spatial prisoner's dilemma. *J. Theor. Biol.* **259**, 84–95.

Nash J. (1950). Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences* **36**, 48–49.

Nehring W., Güth M., Königstein N., and Marchand K. (2001). Trust and reciprocity in the investment game with indirect reward. *Homo oeconomicus* **18**, 241–262.

Nowak M. A. (2006). Five rules for the evolution of cooperation. *Science* **314**, 1560–1563.

Nowak M. A., Bonhoeffer S., and May R. M. (1994a). More spatial games. *International Journal of Bifurcation and Chaos* **4**, 33–56.

Nowak M. A., Bonhoeffer S., and May R. M. (1994b). Spatial games and the maintenance of cooperation. *Proc. Natl. Acad. Sci. USA* **91**, 4877–4881.

Nowak M. A., and May R. M. (1992). Evolutionary games and spatial chaos. *Nature* **359**, 826–829.

Nowak M. A., and Sigmund K. (1989a). Game-dynamical aspects of the prisoners dilemma. *Appl. Math. Comp.* **30**, 191–213.

Nowak M. A., and Sigmund K. (1989b). Oscillations in the evolution of reciprocity. *J. Theor. Biol.* **137**, 21–26.

Nowak M. A., and Sigmund K. (1990). The evolution of stochastic strategies in the prisoner's dilemma. *Acta Appl. Math.* **20**, 247–265.

Nowak M. A., and Sigmund K. (1992). Tit for tat in heterogeneous populations. *Nature* **355**, 250–253.

Nowak M. A., and Sigmund K. (1993). A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature* **364**, 56–58.

Nowak M. A., and Sigmund K. (1998). Evolution of indirect reciprocity by image scoring. *Nature* **393**, 573–577.

Nowak M. A., and Sigmund K. (2004). Evolutionary Dynamics of Biological Games. *Science* **303**, 793–799.

Nowak M. A., and Sigmund K. (2005). Evolution of indirect reciprocity. *Nature* **437**, 1291–1298.

Nowak M. A., Sigmund K., and El-Sedy E. (1995). Automata, repeated games and noise. *J. Math. Biol.* **33**, 703–722.

Ohtsuki H., and Iwasa Y. (2004). How should we define goodness?—reputation dynamics in indirect reciprocity. *J. Theor. Biol.* **231**, 107–120.

Page K. M., and Nowak M. A. (2002). Unifying evolutionary dynamics. *J. Theor. Biol.* **219**, 93–98.

Pennisi E. (2009). On the origin of cooperation. *Science* **325**, 1196.

Poncela J., Gómez-Gardeñes J., Floría L. M., Moreno Y., and Sánchez A. (2009). Cooperative scale-free networks despite the presence of defector hubs. *EPL* **88**, 38003.

Poncela J., Gómez-Gardeñes J., Moreno Y., and Floría L. M. (2010). Cooperation in the Prisoners Dilemma Game in Random Scale-Free Graphs. *International Journal of Bifurcation and Chaos* **20**, 849–857.

Price G. R. (1970). Selection and covariance. *Nature* **227**, 520–521.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0.

Rand D. G., Arbesman S., and Christakis N. A. (2011). Dynamic social networks promote cooperation in experiments with humans. *Proc. Natl. Acad. Sci. USA* **108**, 19193–19198.

Rapoport A., Gordon D., and Guyer M. J. (1976). *The* 2×2 *game*. Ann Arbor: University of Michigan Press.

Rapoport A., and Guyer M. (1966). A taxonomy of 2×2 Games. *General Systems* **11**, 203–214.

Roca C. P., Cuesta J., and Sánchez A. (2009a). The effect of population structure on the evolution of cooperation. *Phys. Rev. E* **80**, 46106.

Roca C. P., Cuesta J., and Sánchez A. (2009b). Evolutionary game theory: temporal and spatial effects beyond replicator dynamics. *Phys. Life Rev.* **6**, 208.

Roca C. P., Cuesta J., and Sánchez A. (2009c). Imperfect imitation can enhance cooperation. *Europhys. Lett.* **87**, 47005–5.

Rockenbach B., and Milinski M. (2006). The efficient interaction of indirect reciprocity and costly punishment. *Nature* **444**, 718–723.

Russell B. (1959). *Common Sense and Nuclear Warfare*. London: George Allen and Unwin Ltd.

Santos F. C., and Pacheco J. M. (2005). Scale-Free Networks Provide a Unifying Framework for the Emergence of Cooperation. *Phys. Rev. Lett.* **95**, 98104.

Santos F. C., Pacheco J. M., and Lenaerts T. (2006a). Cooperation prevails when individuals adjust their social ties. *PLoS Comput. Biol.* **2**, e140.

Santos F. C., Pacheco J. M., and Lenaerts T. (2006b). Evolutionary dynamics of social dilemmas in structured heterogeneous populations. *Proc. Natl. Acad. Sci. USA* **103**, 3490–3494.

Schlag K. H. (1998). Why imitate, and if so, how? A boundedly rational approach to multi-armed bandits. *J. Econ. Theory* **78**, 130–156.

Self S., and Liang K. (1987). Asymptotic Properties of Maximum Likelihood Estimators and Likelihood Ratio Test under Nonstandard Conditions. *Jour. Amer. Stat. Asso.* **82**, 605–610.

Selten R., and Hammerstein P. (1984). Gaps in Harley's argument on evolutionarily stable learning rules and in the logic of tit for tat. *Behav. Brain Sci.* **7**, 115–116.

Sigmund K. (2010). *The Calculus of Selfishness*. Princeton: Princeton University Press.

Skyrms B. (2003). *The Stag Hunt and Evolution of Social Structure*. Cambridge: Cambridge University Press.

Spichtig M., and Traxler C. (2009). Social norms and the indirect evolution of conditional cooperation. *Munich Discussion Papers in Economics* **2007-12**.

Sugden R. (2004). *Economics of Rights, Co-operation and Welfare* (Segunda ed.). Hampshire: Palgrave Macmillan.

Suri S., and Watts D. J. (2011). Cooperation and Contagion in Web-Based, Networked Public Goods Experiments. *PLoS ONE* **6(3)**, e16836.

Szabó G., and Fáth G. (2007). Evolutionary games on graphs. *Phys. Rep.* **446**, 97–216.

Szabó G., Szolnoki A., and Vukov J. (2009). Selection of dynamical rules in spatial Prisoner's Dilemma games. *Europhys. Lett.* **87**, 18007.

Szabó G., and Töke C. (1998). Evolutionary prisoner's dilemma game on a square lattice. *Phys. Rev. E* **58**, 69–73.

Szolnoki A., and Perc M. (2008). Promoting cooperation in social dilemmas via simple coevolutionary rules. *Eur. Phys. J. B* **67**, 337–342.

Taylor P. D., and Jonker L. (1978). Evolutionarily stable strategies and game dynamics. *Math. Biosci.* **40**, 145–156.

Traulsen A., Hauert C., De Silva H., Nowak M. A., and Sigmund K. (2009). Exploration dynamics in evolutionary games. *Proc. Natl. Acad. Sci. USA* **106**, 709–712.

Traulsen A., Nowak M. A., and Pacheco J. M. (2006). Stochastic Dynamics of Invasion and Fixation. *Physical Review E* **74**, 011909.

Traulsen A., Semmann D., Sommerfeld R. D., Krambeck H.-J., and Milinski M. (2010). Human strategy updating in evolutionary games. *Proc. Natl. Acad. Sci. USA* **107**, 2962–2966.

Trivers R. L. (1971). The evolution of reciprocal altruism. *Q. Rev. Biol.* **46**, 35–57.

van Damme M., Dufwenberg M., Gneezy U., and Güth E. (2001). Direct vs indirect reciprocity: an experiment. *Homo oeconomicus* **18**, 19–30.

Van Segbroeck S., Santos F. C., Lenaerts T., and Pacheco J. M. (2009). Reacting differently to adverse ties promotes cooperation in social networks. *Phys. Rev. Lett.* **102**, 058105.

Van Segbroeck S., Santos F. C., Pacheco J. M., and Lenaerts T. (2010). Coevolution of cooperation, response to adverse social ties and network structure. *Games* **1**, 317–337.

von Neumann J., and Morgenstern O. (1944). *The Theory of Games and Economic Behavior*. Princeton: Princeton University Press.

Wedekind C., and Milinski M. (2000). Stabilizing the Earths climate is not a losing game: Supporting evidence from public goods experiments. *Science* **288**, 850–852.

Wilson D. S., and Sober E. (1994). Reintroducing group selection to the human behavioral sciences. *Behavioral and Brain Sciences* **17**, 585–654.

Wynne-Edwards V. C. (1962). *Animal Dispersion in relation to Social Behaviour*. Oliver & Boyd.

Yamagishi T. (1986). The provision of a sanctioning system as a public good. *Journal of Personality and Social Psychology* **51**, 110–116.

Yao X., and Darwen P. J. (1995). An experimental study of N-Person Iterated Prisoner's Dilemma games. *Lecture Notes in Comput. Sci.* **956**, 90–108.

Zeeman E. C. (1980). Population dynamics from game theory. In Nitecki Z., and Robinson C. (Eds.), *Global Theory of Dynamical Systems*, pp. 471–497. Springer.

Zelmer J. (2003). Linear public goods experiments: A meta-analysis. *Experim. Econ.* **6**, 299–310.