Simultaneous Evolution of Learning Rules and Strategies

Oliver Kirchkamp*

First Version: June 1996, Revised: April 1998

Abstract

We study a model of local evolution. Agents are located on a network and interact strategically with their neighbours. Strategies are chosen with the help of learning rules that are based on the success of strategies observed in the neighbourhood.

The standard literature on local evolution assumes learning rules to be *exogenous* and fixed. In this paper we consider a specific evolutionary dynamics that determines learning rules *endogenously*.

We find with the help of simulations that in the long run learning rules behave rather deterministically but are asymmetric in the sense that while learning they put more weight on the learning players' experience than on the observed players' one. Nevertheless stage game behaviour under these learning rules is similar to behaviour with symmetric learning rules.

Keywords: Evolutionary Game Theory, Learning, Local Interaction, Networks. JEL-Code: C63, C72, D62, D63, D73, D83, R12, R13.

*University of Mannheim, SFB 504, L 13, 15, D-68131 Mannheim, email kirchkamp@sfb504.uni-mannheim.de

I am grateful for financial support from the Deutsche Forschungsgemeinschaft through SFB 303 and SFB 504. Parts of this paper were written at the European University Institute (EUI), Florence. I am very grateful for the hospitality and the support that I received from the EUI.

I thank Georg Nöldeke, Karl Schlag, Avner Shaked, Fernando Vega-Redondo, two anonymous referees, and several seminar participants for helpful and stimulating comments.

Contents

1 Introduction **1**

2 The Model **3**

2.1 Overview 3

2.2 Stage Games 3

2.3 Repeated Game Strategies 5

2.4 Learning Rules 5

2.5 Exogenous Dynamics that Select Learning Rules 8

2.6 Initial Configuration 10

3 Results **10**

3.1 Distribution over Learning Parameters 10

3.2 Probabilities to Switch 12

3.3 Comparison with other Learning Rules 13

3.4 Dependence on Parameters 14

3.4.1 The Selection Rule: Sampling Randomly or Selectively 14

3.4.2 Other Parameter Changes 15

3.5 Stage Game Behaviour 19

4 Conclusions **22**

1 Introduction

In this paper we want to study how *strategies* evolve simultaneously with *learning*[1] *rules* in a local environment. We regard this paper as a modification of models of local evolution where only strategies may evolve, but learning rules are kept fix. The latter kind of models has been studied by Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer and May (1993), Eshel, Samuelson and Shaked (1996), and Kirchkamp (1995). In these models players play games against neighbours, using a strategy that they may change from time to time. When changing this strategy they use a fixed rule; normally they either imitate the strategy of the most successful neighbour or the strategy with the highest average success in their neighbourhood respectively. Both rules seem to be rather plausible, and both rules lead to a nice explanation for the survival of cooperation in prisoners' dilemmas: A cluster of mutually cooperating players may seem to be more successful than a (neighbouring) cluster of mutually defecting players. Given myopic imitation the idea of cooperation spreads through a network.

However, this property depends on the assumed learning rule. Other learning rules, e.g. players that imitate with probabilities that are strictly proportional to the success of the observed strategies, do not give this explanation for the survival of cooperation. Such proportional imitation rules may actually be viewed as particularly plausible since proportional rules turn out to be optimal at least in a *global* setting where all members of a population are equally likely to interact with each other (see Börgers and Sarin (1995), Schlag (1993, 1994)).

Notice that local evolution is, thus, much more sensitive to seemingly innocuous changes of the learning rule than global evolution. This sensitivity makes the local setup particularly attractive to study selection of learning rules. To perform this task we have to decide whether we want to extend Börgers and Sarin or Schlag and search learning rules that are *optimal* in a *local* environment or whether we want to extend *local evolution* of strategies to include also learning rules. In this paper we take the second approach and study a model with evolution both on the level of strategies and on the level of learning rules.

Within the (wide) range of models of local evolution some (Sakoda 1971, Schelling 1971) assume players' states to be fixed and concentrate on the movements of players. Others (Axelrod 1984, May and Nowak 1992, May and Nowak 1993, Bonhoeffer, May, and Nowak 1993, Ellison 1993, Eshel, Samuelson, and Shaked 1996, Lindgreen and Nordahl 1994, Kirchkamp 1995) take players' positions as fixed but allow players to change their states.

[1] Given that there are lots of definitions for 'learning' let us start with a clarifying remark: When we talk about learning in the following we have in mind a *descriptive* definition of learning in the sense of a relative permanent change of behavioural potentiality (see G. Kimble (Hilgard, Marquis, and Kimble 1961)). We restrict our attention to very simple learning rules. In particular we do not aim to provide a model of learning as a cognitive process.

Furthermore there are models where players are allowed to move *and* to change their state (Hegselmann 1994, Ely 1995). Another distinction is that some authors (like Sakoda, Schelling and Ellison) assume myopically optimising players while others (Axelrod; Nowak, Bonhoeffer, May; Eshel, Samuelson, Shaked; Kirchkamp; Lindgren, Nordahl) assume that players learn through imitation.

In the following we restrict ourselves to players who have fixed positions and who change their strategy using a rule that is based on imitation. Thus, our model has more elements in common with Axelrod; Nowak, Bonhoeffer, May; Eshel, Samuelson, Shaked; Kirchkamp; and Lindgren and Nordahl. This literature assumes all players to use a *fixed* learning rule that either copies from the current neighbourhood the strategy of the neighbour that is most successful[2] or the strategy that is on average (over all neighbours who use it) most successful. In contrast we want to allow players to *change* their own learning rule using a process that is based on imitation.

Such a dynamics yields a set of learning rules that we can compare with the exogenously given learning rules from the literature. Further we can compare the stage game behaviour of a population using endogenous learning rules with the stage game behaviour of a population with fixed learning rules.

Regarding the literature on *local evolution* we want to ask two questions: First we want to know whether the *learning rules* discussed in the above literature are likely to be selected by evolution. Second we want to know whether the *behaviour* of a society with endogenous learning rules is different from the behaviour of one with a fixed learning rule. In this paper we present simulation results to give an answer to these questions.

Another useful benchmark is the literature that studies properties of *optimal* learning rules in a *global* environment, i.e. an environment where *all* players may interact with each other and learn from each other. This kind of problem is studied by Binmore and Samuelson (1994), Börgers and Sarin (1995), Schlag (1993, 1994). Binmore and Samuelson already require symmetry and study an aspiration level that is subject to a noisy evolutionary process. Börgers and Sarin as well as Schlag look for an optimal learning rule and find that such a learning rule (in a global context) turns out to be symmetric, in the sense that learning players put equal weight on their own as well as on other players' experience. Further, global learning rules that are optimal or that survive in the long run turn out to be linear in the sense that optimal learning rules require learning players to switch to an observed strategy with a probability that is a linear function of the player's own and the observed payoff.

In the next section we describe a model of local endogenous evolution of learning rules. In

[2]We understand here 'success' as 'average payoff per interaction'.

section 3 we discuss properties of these learning rules. Section 3.4 analyses the dependence of our results on parameters of the model. In section 3.5 we then study the implication of endogenous learning on the stage game behaviour. Section 4 concludes.

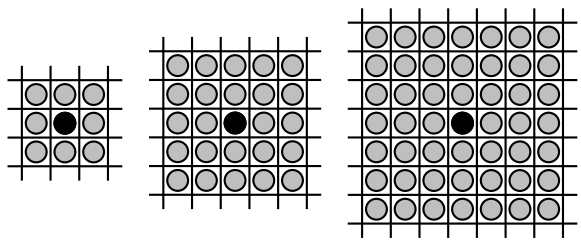
2 The Model

2.1 Overview

In the following we study a population of players each occupying one cell of a torus of size $n \times n$ where n is between 5 and 200. Players play games with their neighbours on this network, learn repeated game strategies from their neighbours and update their learning rule using information from their neighbours. In the remainder of this section we describe the kind of games players play, and strategies and learning rules they use.

2.2 Stage Games

Players play games within a neighbourhood. In the simulations that we discuss in the following such a neighbourhood has one of the following shapes:



$r = 1$ $r = 2$ $r = 3$ (1)

A player (marked as a black circle) may only interact with those neighbours (gray) which live no more than r_i cells horizontally or vertically apart. In each period a random draw decides for each neighbour independently whether an interaction takes place. Thus, in a given period a player may sometimes interact with all neighbours, sometimes with only some, sometimes even with no neighbour at all. Each possible interaction with a given neighbour takes place in each period independently from all other interactions with probability p_i. A typical value for p_i is $1/2$. This probability is low enough to avoid synchronisation among neighbours, it is still high enough to make simulations sufficiently fast. We consider values for p_i ranging from $1/100$ to 1 to test the influence of this parameter.

We assume that games which are played among neighbours change every t_g periods. We change games to create the necessity to adapt to a changing environment and, thus, induce evolutionary pressure on learning rules. In the following we present results of simulations

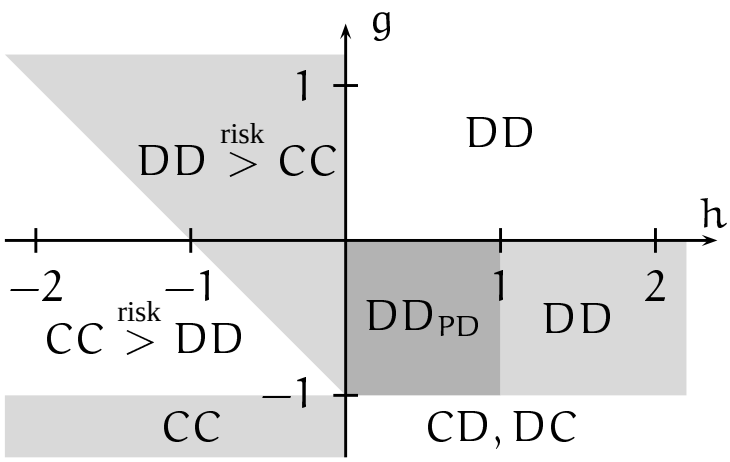


Figure 1: The space of considered games.

where t_g ranges from 200 to 20 000 periods. Once a new game is selected, all neighbours in our population play the same symmetric 2×2 game of the following form:

		Player II	
		D	C
Player I	D	g, g	h, -1
	C	-1, h	0, 0

(2)

When a new game is selected the parameters g and h in the above game are chosen randomly following an equal distribution over the intervals $-1 < g < 1$ and $-2 < h < 2$. We can visualise the space of games in a two-dimensional graph (see figure 1).

The range of games described by $-1 < g < 1$ and $-2 < h < 2$ includes both prisoners' dilemmas and coordination games. All games with $g \in (-1, 0)$ and $h \in (0, 1)$ are prisoners' dilemmas (DD_{PD} in figure 1), all games with $g > -1$ and $h < 0$ are coordination games. In figure 1 equilibrium strategies are denoted with CC, CD, DC and DD respectively. The symbol $\overset{risk}{>}$ denotes risk dominance for games that have several equilibria.

We already know from the literature on local evolution[3] that with the learning rule 'copy best player' players cooperate at least in some prisoners' dilemmas. Kirchkamp (1995) points further out that when playing coordination games using this learning rule players do not always coordinate on the risk dominant equilibrium but follow a criterion which puts also some weight

[3]See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer and May (1993), Eshel, Samuelson and Shaked (1996), Kirchkamp (1995).

on Pareto dominance. We will see in the remainder of this paper that this behaviour persists at least to some degree also with endogenous learning rules.

2.3 Repeated Game Strategies

We assume that each player uses a single repeated game strategy against all neighbours. Repeated game strategies are represented as (Moore) automata with a maximal number of states of one, two, three, or four[4]. For many simulations we limit the number of states to less than three.

2.4 Learning Rules

From time to time a player has the opportunity to revise his or her repeated game strategy. We assume that this opportunity is a random event that occurs for each player independently at the end of each period with a certain probability. Probabilities to learn will be denoted $1/t_l$ and range from $1/6$ to $1/120$. t_l denotes then the average time between two learning events of a player. Learning is a relatively rare event, as compared to interaction. Still, learning occurs more frequently than changes of the stage game and updates of the learning rule itself (see below).

If a player updates the repeated game strategy the player samples randomly one member of the neighbourhood and then applies the individual learning rule.

Notice, that this learning rule uses information on a *single* sampled player. The learning rules discussed in the literature[5] use information on *all* neighbours from the learning neighbourhood simultaneously.

We assume here that only a single player is sampled to simplify our learning rule in the sense that only a single alternative to the player's current repeated game strategy is considered. It is hard to specify a space of learning rules that learn from several neighbours but can still be described with a small number of parameters.

[4]Each 'state' of a Moore automaton is described by a stage-game strategy and a transition function to either the same or any other of the automaton's states. This transition depends on the opponent's stage-game strategy. Each automaton has one 'initial state' that the automaton enters when it is used for the first time.

There are 2 automata with only one state (one of them plays initially C and remains in this state whatever the opponent does, the other plays always D.

There are 26 automata with one or two states. E.g. 'grim' is a two-state automaton. The initial state plays C. The automaton stays there unless the opponent plays D. Then the automaton switches to the second state that plays D and stays there forever. Other popular two-state automata include 'tit-for-tat', 'tat-for-tit', etc.

The set of automata with less than four states contains 1752 different repeated game strategies. The set of automata with less than five states has already size 190646.

[5]See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonnhoeffer and May (1993), Eshel, Samuelson and Shaked (1996), Kirchkamp (1995).

We briefly analyse two *alternative* setups in order to test sensitivity with respect to the type of the learning rule.

First, and in order to be comparable with the above mentioned literature, we study in section 3.5 a *fixed* learning rule that samples a *single* player and that aims to be similar to the fixed multi-player rules in the literature. With the help of simulations we show that stage game behaviour under the fixed single-player rule turns out to be very similar to fixed multi-player rules. E.g. cooperation in prisoners' dilemmas occurs with the learning rule 'copy best player' for almost the same range of games, regardless whether only one or all neighbours are sampled.

Second, and in order to show that choosing the single-player setup is not crucial for the properties of the learning rules that we derive, we compare in section 3.4.1 the single-player rule with the following multi-player rule: All neighbours are sampled, the most successful neighbour is determined, and then imitated with a probability that is again a linear function of the most successful neighbour's and the learning player's success. It turns out that the learning rules that evolve under this regime are very similar to those that emerge under the single-player regime that we describe in the next paragraph.

Learning, as well as interaction, occurs in neighbourhoods of similar shape (see graph 1 on page 3). We denote the size of the neighbourhood for *learning* with the symbol r_l.

Learning rules use the following information:

1. The learning player's repeated game strategy.
2. The payoff u_{own} of the player's repeated game strategy, i.e. the average payoff per interaction that the player received while using this repeated game strategy.
3. A sampled player's repeated game strategy.
4. The sampled player's repeated game strategy payoff u_{samp}, i.e. the average payoff per interaction that the player received while using this repeated game strategy.

Learning rules are characterised by a vector of three parameters $(\hat{a}_0, \hat{a}_1, \hat{a}_2) \in \mathbb{R}^3$. Given a learning rule $(\hat{a}_0, \hat{a}_1, \hat{a}_2)$ a learning player samples one neighbours' strategy and payoff and then switches to the sampled strategy with probability

$$p(u_{\text{own}}, u_{\text{samp}}) = \langle \hat{a}_0 + \hat{a}_1 u_{\text{own}} + \hat{a}_2 u_{\text{samp}} \rangle \tag{3}$$

where

$$\langle x \rangle := \begin{cases} 1 & \text{if } x > 1 \\ 0 & \text{if } x < 0 \\ x & \text{otherwise} \end{cases} . \tag{4}$$

u_{own} and u_{samp} denote the player's and the neighbour's payoff respectively.

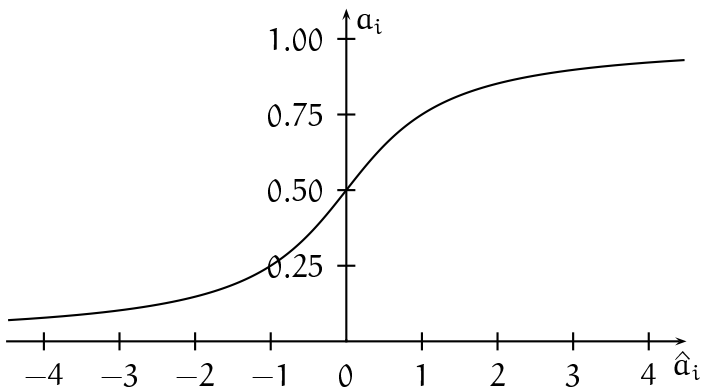
Thus, the two parameters $\hat{a}_1$ and $\hat{a}_2$ reflect sensitivities of the switching probability to changes in the player's and the neighbour's payoff. The parameter $\hat{a}_0$ reflects a general readiness to change to new strategies, which can be interpreted as a higher or lower inclination to make an experiment or to try something new. Choosing $(\hat{a}_0, \hat{a}_1, \hat{a}_2)$ a player determines a range of payoffs where to react probabilistically (i.e. $p(u_{\text{own}}, u_{\text{samp}}) \in (0, 1)$), a second range of payoffs where switching will never occur (i.e. $p(u_{\text{own}}, u_{\text{samp}}) = 0$) and finally a third range of payoffs where imitation always occurs (i.e. $p(u_{\text{own}}, u_{\text{samp}}) = 1$).

Note that one or even two of these ranges can vanish, i.e. we can specify stochastic as well as deterministic rules. An example for a deterministic rule ('switch if better') is $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (0, -\bar{a}, \bar{a})$ with $\bar{a} \to \infty$. An example for a rule that implies always stochastic behaviour for the game given in 2 is $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (1/2, -\bar{a}, \bar{a})$ with $1/(4\bar{a}) > \max(|g|, |h|, 1)$.

Notice also that our parameter $\hat{a}_0$ is similar to the aspiration level Δ from the global model studied in Binmore and Samuelson (1994). However, the learning rules studied in Binmore and Samuelson are not special cases of our learning rules, since their decisions are perturbed by exogenous noise. For cases where this noise term becomes small our rule approximates Binmore and Samuelson (1994) with $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (\Delta, -\bar{a}, \bar{a})$ and $\bar{a} \to \infty$.

Normalisation We map parameters $(\hat{a}_0, \hat{a}_1, \hat{a}_2) \in \mathbb{R}^3$ into $(a_0, a_1, a_2) \in [0, 1]^3$ using the following rule:

$$\hat{a}_i \equiv \tan\left(\pi a_i - \frac{\pi}{2}\right) \qquad \forall i \in \{0, 1, 2\}. \tag{5}$$



We let evolution operate on the normalised values $(a_0, a_1, a_2) \in [0, 1]^3$ for the following reason: The learning rules from the literature are often deterministic, thus, they can be represented as rules whose parameter values $\hat{a}_i$ are infinitely large or small respectively. We do not want to exclude these rules a priori. However, it might be a problem for a dynamics that selects learning rules to converge within the limits of a finite simulation to infinite values of the parameters. We

therefore map the unbounded space of parameters of our learning rules into a bounded space using the transformation given by equation 5 on the preceding page.

Mutations When a player learns a repeated game strategy, sometimes learning fails and a random strategy is learned instead. In this case, any repeated game strategy, as described in section 2.3, is selected with equal probability. These 'mutations' occur with a fixed probability m_l. We consider mutation rates between 0 and 0.7.

We introduce mutations in order to show that simulations are particularly robust. However, as we see in section 3.4, we do not need mutations for our results.

Mutations can also be seen as a way to compensate for the limited size of our population. Even if all members of one species die out the species still has a chance to enter the population again through a mutation.

2.5 Exogenous Dynamics that Select Learning Rules

From time to time a player has the opportunity to revise the learning rule. In our simulations we assume that this opportunity is a random event that occurs for each player independently with probability $1/t_u$. t_u, thus, denotes the average time between two updates of a player. We consider learning rates $1/t_u$ among $1/40\,000$ to $1/400$. If not mentioned otherwise $1/t_u = 1/4000$. In particular learning rules are updated much slower than updates of strategies or changes of games.

We want to model a situation where updates of learning rules occur very rarely. We find it justified that for these rare events players make a larger effort to select a new learning rule. For our model this has the following two implications: All neighbours are sampled when updating learning rules (and not only a single neighbour as for update of strategies) and the sampled data is evaluated more efficiently, using now a quadratic approximation.

The shape of the neighbourhoods that are used to update learning rules is similar to those used for interaction and learning (see graph 1 on page 3). We denote the size of the neighbourhood for *update of learning rules* with the symbol r_u.

A player who updates the learning rule has the following information for all neighbours individually (including him- or herself):

1. The (normalised) parameters of the respective learning rule a_0, a_1, a_2.

2. The average payoff per interaction that the respective player received while this learning rule was used, $u(a_0, a_1, a_2)$.

To evaluate this information we assume that players estimate a model that helps them explaining their environment, in particular their payoffs. Players then use this model to choose an optimal

learning rule. To model this decision process we assume that players approximate a quadratic function of the learning parameters to explain success of a learning rule. Formally the quadratic function can be written as follows:

$$u(a_0, a_1, a_2) = c + (a_0, a_1, a_2) \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} + \\ +(a_0, a_1, a_2) \begin{pmatrix} q_{00} & q_{01} & q_{02} \\ q_{01} & q_{11} & q_{12} \\ q_{02} & q_{12} & q_{22} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} + \epsilon \tag{6}$$

Players make an OLS-estimation to derive the parameters of this model (ϵ describes the noise). We choose a quadratic function because it is one of the simplest models which still has an optimum. Similarly we assume that players derive this model using an OLS-estimation because this is a simple and canonical way of aggregating the information players have. We do not want to be taken too literally: We want to model players that more or less behave *as if* they would maximise a quadratic model which is derived using an OLS-estimation.

The OLS-Regression determines the parameters $(c, b_0, b_1, b_2, q_{00}, \ldots, q_{22})$ of the above model. Given this model, the player determines the combination of a_0, a_1, a_2 that maximises $u(a_0, a_1, a_2)$ s.t. $(a_0, a_1, a_2) \in [0, 1]^3$. We find that in 99% of all updates the Hessian of $u(a_0, a_1, a_2)$ is negative definite, i.e. $u(a_0, a_1, a_2)$ has a unique local maximum. In the remaining less than 1% of all updates the quadratic model might be unreliable. In this case we therefore copy the most successful neighbour.

Figure 2 shows (only for one dimension) an example for a sample of several pairs of a parameter a_i and a payoff u (black dots) together with the respective estimation of the functional relationship (gray line) between a_i and u.

Mutations We also introduce mutations for players' learning rules. When a player updates the learning rule, with a small probability m_l not the above described update scheme is used but the player learns a random learning rule that is chosen following an equal distribution (for the normalised parameters) over $(a_0, a_1, a_2) \in [0, 1]^3$, which is equivalent to a random and independent selection of $\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$ following each a Cauchy distribution. We consider mutation rates for learning m_l between 0 and 0.7.

The reason to introduce mutations at this level is the same as given above for mutation of strategies: We want to show that our simulation results are robust. A population that due to its limited size gets stuck in some state may always escape through a mutation. Mutations are,

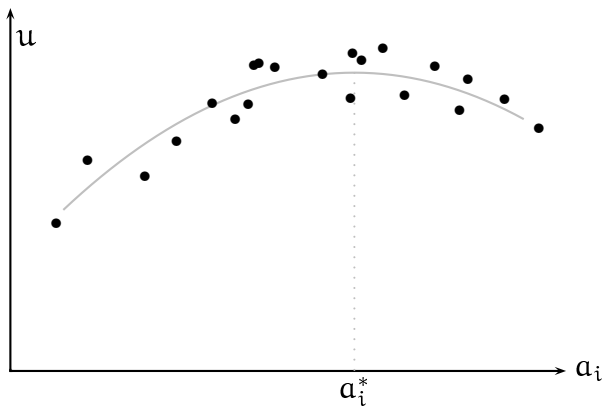


Figure 2: An example for samples of pairs of parameters and payoffs (black) which are used to estimate a functional relationship (gray) between a_i and u. Given this relationship an optimal value a_i^* is determined.

thus, a way to compensate for the fact that our simulations are done with a relatively small population (only 25 to 40 000 members).

However, as we show in section 3.4, we neither need mutations on the level of strategies nor on the level of players' learning rules. Results without mutations are very simular to the ones with a small amount of mutations.

2.6 Initial Configuration

At the beginning of each simulation each player starts with a random learning rule that is chosen following an equal distribution over $(a_0, a_1, a_2) \in [0, 1]^3$. Thus, the parameters $\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$ are distributed independently following a Cauchy distribution. Also each player starts with a random repeated game strategy, again following an equal distribution over the available strategies.

3 Results with Endogenous Learning Rules

3.1 Distribution over Learning Parameters

Figure 3 displays averages over 53 simulations on a 50×50 grid, lasting 400 000 periods each.

Since we can not display a distribution over the three-dimensional space (a_0, a_1, a_2) we analyse two different projections into subspaces. The left part of figure 3 displays the distribution over (a_1, a_2), the right part over $(a_0, a_1 + a_2)$ respectively. Axes range from 0 to 1 for a_0, a_1 and a_2 and from 0 to 2 in the case of $a_1 + a_2$. Labels on the axes do not represent the

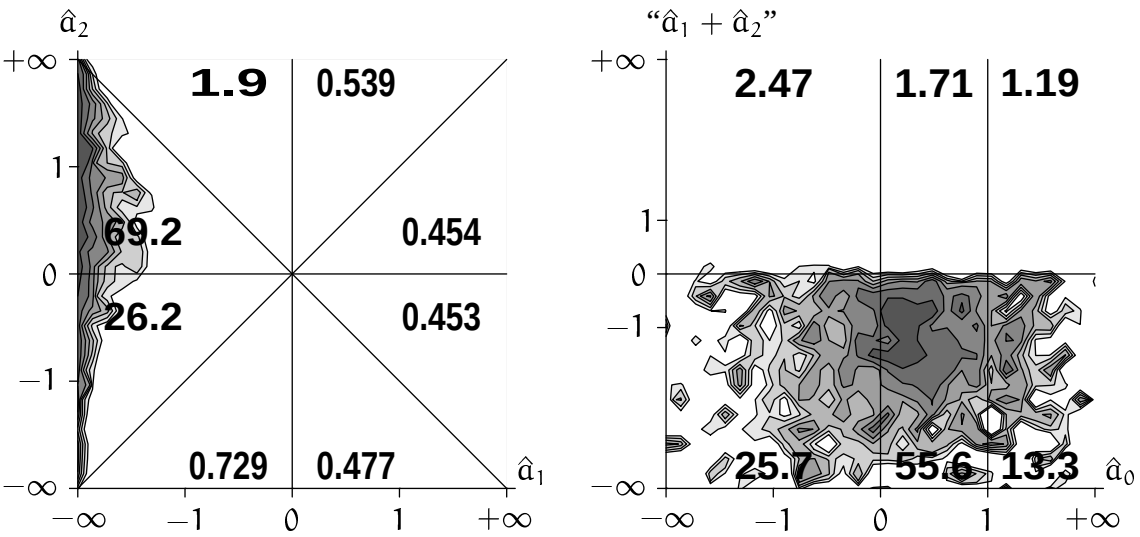


Figure 3: Long run distribution over parameters of the learning rule (a_0, a_1, a_2). Average over 53 simulations runs on a torus of size 50×50 with 2-state automata. Neighbourhoods have sizes $r_i = r_l = 1$, $r_u = 2$. Relative frequencies are given as percentages. Simulations last for $t_s = 400\,000$ periods, interactions take place with probability $p_i = 1/2$, repeated game strategies are learned from a randomly sampled player with probability $1/t_l = 1/24$, learning rules are changed with probability $1/t_u = 1/4000$, new games are introduced with probability $t_g = 1/2000$, mutations both for repeated game strategies and for learning rules occur at a rate of $m_l = m_u = 1/100$.

normalised values but instead $\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$ which range from $-\infty$ to $+\infty$.[6]

Both pictures are simultaneously a density plot and a table of relative frequencies:

Density plot: Different densities of the distribution are represented by different shades of gray. The highest density is represented by the darkest gray.[7]

Table of relative frequencies: The pictures in figure 3 also contains a table of relative frequencies. The left picture is divided into eight sectors, the right picture is divided into six rectangles. The percentages within each sector or rectangle represent the amount of players that use a learning rule with parameters in the respective range.

The left part of figure 3 shows two interesting properties of endogenous evolution: First, with endogenous evolution learning rules are sensitive to a player's own payoff. Second, they are

[6]The figure is derived from a table of frequencies with 30×30 cells. The scaling of all axes follows the normalisation given in equation 5 on page 7. To be precise, the value "$\hat{a}_1 + \hat{a}_2$" represents actually $2 \cdot \tan(\pi \cdot (a_1 + a_2)/2 - \pi/2)$ and not $\hat{a}_1 + \hat{a}_2$. In the current context this difference should be negligible.

[7]Densities are derived from a table of frequencies with a grid of size 30×30 for each picture. We actually map logs of densities into different shades of gray. The interval between the log of the highest density and the log of 1% of the highest density is split into seven ranges of even width. Densities with logs in the same interval have the same shade of gray. Thus, the white area represents densities smaller than 1% of the maximal density while areas with darker shades of gray represent densities larger than 1.9%, 3.7% 7.2%, 14%, 27% and 52% of the maximal density respectively.

substantially less sensitive to observed payoffs. We call this latter property *suspicion*. Of course, our agents can only behave *as if* they had feelings like suspicion. We still hope that the image helps the reader.

Sensitivity to own payoffs: Remember that the initial distribution over α_1 and α_2 is an equal distribution. Thus, would we draw the left part of figure 3 in period one, the result would be a smooth gray surface without any mountains or valleys. Starting from this initial distribution our learning parameters change substantially. Even if not in all cases $\hat{\alpha}_1$ becomes $-\infty$, the distribution over learning parameters puts most its weight on small values of α_1.

Insensitivity to sampled payoffs: In the left part of figure 3 we see that 96.3% of all players use a learning rule with $|\hat{\alpha}_2| < |\hat{\alpha}_1|$, i.e. a learning rule which puts more weight on the player's own payoff than on the sampled payoff.

If we restrict ourselves to 'reasonable' learning rules with $\hat{\alpha}_1 < 0$ and $\hat{\alpha}_2 > 0$ then 97.5% of all these rules have the property that $|\hat{\alpha}_2| < |\hat{\alpha}_1|$.

Notice that for both cases the *initial* distribution over parameters of the learning rule implies that 50% of all rules fulfil $|\hat{\alpha}_2| < |\hat{\alpha}_1|$.

We call this kind of behaviour 'suspicious' in the following sense: A sampling player may realise that an observed learning rule is successful for a neighbour. Nevertheless the player does not know whether the same rule is equally successful at the player's own location. Perhaps the success of a neighbour's rule depends on players which are neighbours of the player's neighbour, but not of the player. Thus, a 'suspicious' player behaves like somebody who 'fears' that the sampled neighbour's experience can not be generalised for the player's own case.

3.2 Probabilities to switch to a sampled learning rule

The learning rule as specified in equation 3 on page 6 determines for each learning player a probability to switch to the observed repeated game strategy. Figure 4 on the following page shows the cumulative distribution of these switching probabilities.[8]

The horizontal axis represents $\hat{\alpha}_0 + \hat{\alpha}_1 u_{\text{own}} + \hat{\alpha}_2 u_{\text{samp}}$. Following the learning rule 3 a player switches *stochastically* with probability $\hat{\alpha}_0 + \hat{\alpha}_1 u_{\text{own}} + \hat{\alpha}_2 u_{\text{samp}}$ if this expression is between zero and one. Otherwise the player either switches *with certainty* or *not at all*.

[8]This figure is derived from a table of frequencies with 30 cells. The scaling of the horizontal axis follows the normalisation given in equation 5 on page 7.

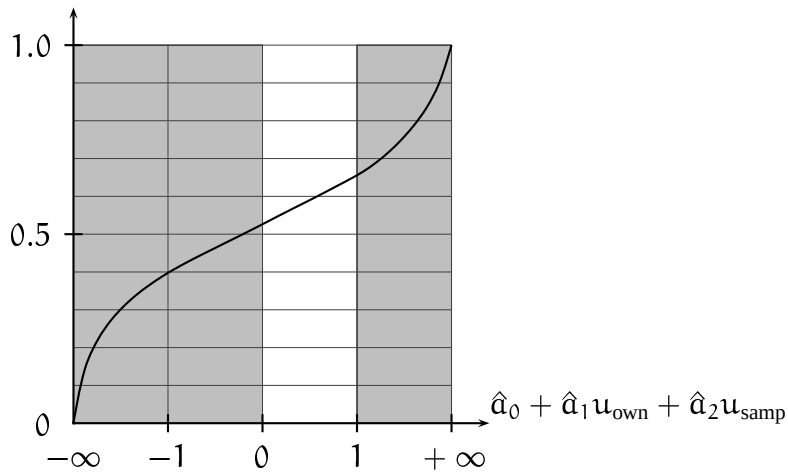


Figure 4: Cumulative distribution over switching probabilities, given the learning rules from figure 3.

Figure 4 shows that only in about 12% of all learning events $0 < \hat{a}_0 + \hat{a}_1 u_{own} + \hat{a}_2 u_{samp} < 1$, i.e. in only 12% of all learning events a player's decision is a stochastic one. Our endogenous rules seem to be neither fully stochastic[9] nor fully deterministic[10]

3.3 Comparison with other Learning Rules

Above we mentioned two reference points for learning rules: Those learning rules that are assumed as *exogenous and fixed* in the literature on *local* evolution and rules that turn out to be *optimal* in a *global* setting.

Let us start with the exogenous rules that are assumed in the literature on local evolution: We have seen that the *exogenous fixed* rules may be *similar* to the *endogenous* learning rules in the sense that small changes in the *player's own payoff* may lead to drastic changes in the probability to adopt a new strategy. Endogenous learning rules *differ* from those studied in parts of the literature on local evolution in the sense that changes in an *observed player's payoff* lead to smaller changes in the probability to adopt a new strategy.

Let us next compare our endogenous rules with those rules that turn out to be optimal in a global setting[11]. We may expect that the outcome of an *evolutionary process* that runs only for a finite time is at least close to any *optimal* rule. However, our rules differ in two respects from those that are optimal in a global model: First, as discussed in section 3.1, they are more sensitive to changes in a learning player's own payoff than to changes in an observed neighbours payoff. Second, as mentioned in section 3.2, players following endogenous rules quite often

[9] As those from Börgers, Sarin (1995) and Schlag (1993).
[10] As those from Axelrod (1984, p. 158ff), Nowak and May (1992), etc.
[11] See Börgers and Sarin (1995), Schlag (1993, 1994).

switch with certainty.

A higher sensitivity to a player's own payoff as compared to an observed neighbours payoff can be related to the local structure. A strategy that is successful in my neighbour's neighbourhood may be less successful in my own neighbourhood. Therefore my neighbour's payoff is a less reliable source of information than my own payoff.

The fact the (globally) optimal learning rules switch always stochastically results from the attempt to evaluate information efficiently. Even small differences in payoffs are translated into different behaviour. The price to pay for this efficient evaluation is time. Given that neither in Börgers and Sarin nor in Schlag players are impatient, they do not care whether the optimal strategy is reached only after infinite time.

While we do not have an explicit discount factor in our simulations discounting enters implicitly through the regular update of the learning rule. A learning rule that is efficient, but slow, compares (within finite time) badly to a learning rule that is not perfect in the long run, but achieves already good results in the short run. Therefore evolution of learning rules at a more than infinitesimal speed may lead to deterministic behaviour.

3.4 Dependence on Parameters

The discussion in the previous paragraphs was based on a particular parameter combination. In the following we want to show that parameter changes do not matter for our main results.

We will first consider in section 3.4.1 an alternative rule to sample neighbours that might be copied when learning. Section 3.4.2 then studies changes in the other parameters.

3.4.1 The Selection Rule: Sampling Randomly or Selectively

Above we assumed that players learn repeated game strategies from a randomly sampled player. One might, however, object that players could be more careful in selecting their samples. As a benchmark case we assume in this section that players sample the most successful neighbour available[12]. We show that this change in the selection rule has little influence on the endogenous learning rules.

Figure 5 on the following page shows a distribution over (a_0, a_1, a_2), projected into the a_1, a_2 and $a_0, a_1 + a_2$ space, similar to figure 3 on page 11. In contrast to figure 3 we assume here that players, when learning, sample the player with the highest payoff per interaction for the current repeated game strategy, measured over the lifetime of the respective repeated game strategy.

[12] I.e. the neighbour whose repeated game strategy yields the highest payoff on average (per interaction) since the point in time where the neighbour learned the respective strategy.

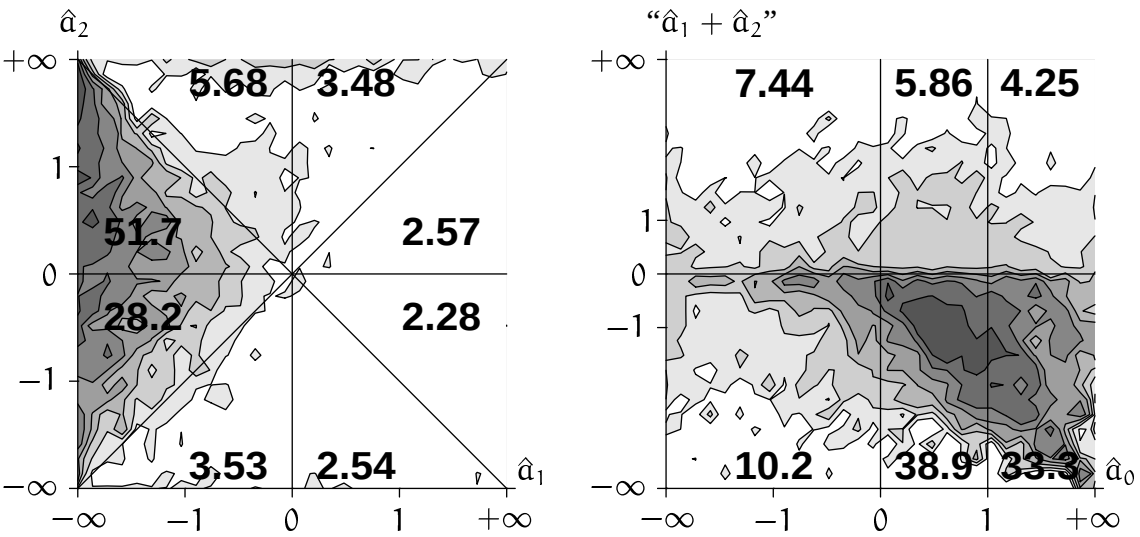


Figure 5: Long run distribution over parameters of the learning rule (a_0, a_1, a_2). Average over 181 simulations runs, each lasting for 400 000 periods. Relative frequencies are given as percentages. Parameters are as in figure 3 on page 11 except that learning players sample the most successful neighbour.

While the picture is more noisy than figure 3 properties of learning rules are the same as already discussed in section 3.1 on page 10. Players are rather sensitive to changes in their own payoff and less sensitive to changes in the sampled neighbours payoff.

We suspect that the additional noise stems from the reduced evolutionary pressure on learning rules. Preselecting already 'best' learning rules for comparison makes the task of identifying good rules too easy for learning rules. We actually observe a cluster of learning rules with values of $\hat{a}_2$ close to $+\infty$. These rules apparently follow a 'just copy whatever you see' strategy, which might be reasonable, since 'whatever you see' is already the best available in your neighbourhood under this selection rule.

3.4.2 Other Parameter Changes

Figure 6 on page 17 shows the effect of various changes of the other parameters. We always start from the same parameter combination as a reference point and then vary one of the parameters keeping all the others fixed. The reference point is a simulation on torus of size 50×50, where the interaction neighbourhood and learning neighbourhood have both the same size $r_i = r_l = 1$ while the neighbourhood that is used when updating the learning rule has size $r_u = 2$. To learn a new repeated game strategy players sample a neighbour randomly. Learning occurs on average every $t_l = 24$ periods[13]. The underlying game is changed every $t_g = 2000$ periods. Players

[13]Remember that we assume learning to be an independent random event that occurs for each player with probability $1/t_l$.

update their learning rule on average every $t_u = 4000$ periods[14]. The mutation rate for learning as well as for update of learning rules is $m_l = m_u = 1/100$. Simulations last for $t_s = 40\,000$ periods. Thus, except for the simulation length, parameters are the same as those for figure 3.

Figure 6 on the following page shows averages[15] of $\hat{a}_1$ and $\hat{a}_2$ for various changes in the parameters. Each dot represents a parameter combination the we simulated. To ease the understanding of the underlying pattern, dots are connected through interpolated splines. The white dot in each diagram represents the average value ($\bar{\hat{a}}_1 = -1.89$, $\bar{\hat{a}}_2 = 0.30$) for the reference set of parameters described above. The line $\bar{\hat{a}}_2 = -\bar{\hat{a}}_1$ is marked in gray.

The main result is that *all* parameter combinations show again relative sensitivity to own payoffs, and insensitivity to observed payoffs. In particular the averages $\bar{\hat{a}_2} < -\bar{\hat{a}_1}$ for *all* parameter combinations that we simulated.

Notice that we do not *need* mutations for our results, however, the simulations are robust against mutations. To show that we can dispense with both kinds of mutations simultaneously we ran a simulation where $m_l = m_u = 0$ and show the result in the graph 'mutation of learning rules' with a small triangle. While learning on average to a smaller value of $\hat{a}_2$ we have still $\bar{\hat{a}}_2 < -\bar{\hat{a}}_1$. On the other hand we can introduce rather large probabilities of mutations (up to 0.7) and still have $\bar{\hat{a}}_2 < -\bar{\hat{a}}_1$.

In the remainder of this subsection we want to discuss the dependence on parameters in more detail. To do that we distinguish three dimensions how parameters influence learning rules: Relative *speed*, relative *efficiency*, and relative degree of *locality*.

If parameters of the simulation are chosen in a way that makes it *slow*, or *inefficient* (e.g. due to a lot of noise), the distribution over the parameters of the learning rule remains close to the initial distribution (which has averages $\bar{\hat{a}_1} = 0$, $\bar{\hat{a}_2} = 0$).

If the parameters of a simulation describe a situation that has less aspects of *locality* (e.g. the interaction radius is large, such that almost everybody interacts with everybody else) 'suspicious' behaviour disappears, and averages for $-\bar{\hat{a}_1}$ and $-\bar{\hat{a}_2}$ moves closer to symmetric values, i.e. moves closer to the gray line where $\bar{\hat{a}_2} = -\bar{\hat{a}_1}$.

Table 1 on page 18 summarises the effects of the parameters on these three dimensions.

Let us briefly discuss some of the dependencies on parameters:

Locality: The parameters t_g, t_l, m_l, r_l, C, and n influence 'locality' in the sense that players becomes either more similar or more diverse in the evolutionary process. As a consequence

[14]We also assume update of learning rules to be an independent random event that occurs for each player with probability $1/t_u$.

[15]These averages are arithmetic averages of a_1 and a_2 respectively, taken over 20 different simulations runs that are initialised randomly. The average values of a_1 and a_2 are then transformed into $\hat{a}_1$ and $\hat{a}_2$ using equation 5 on page 7.

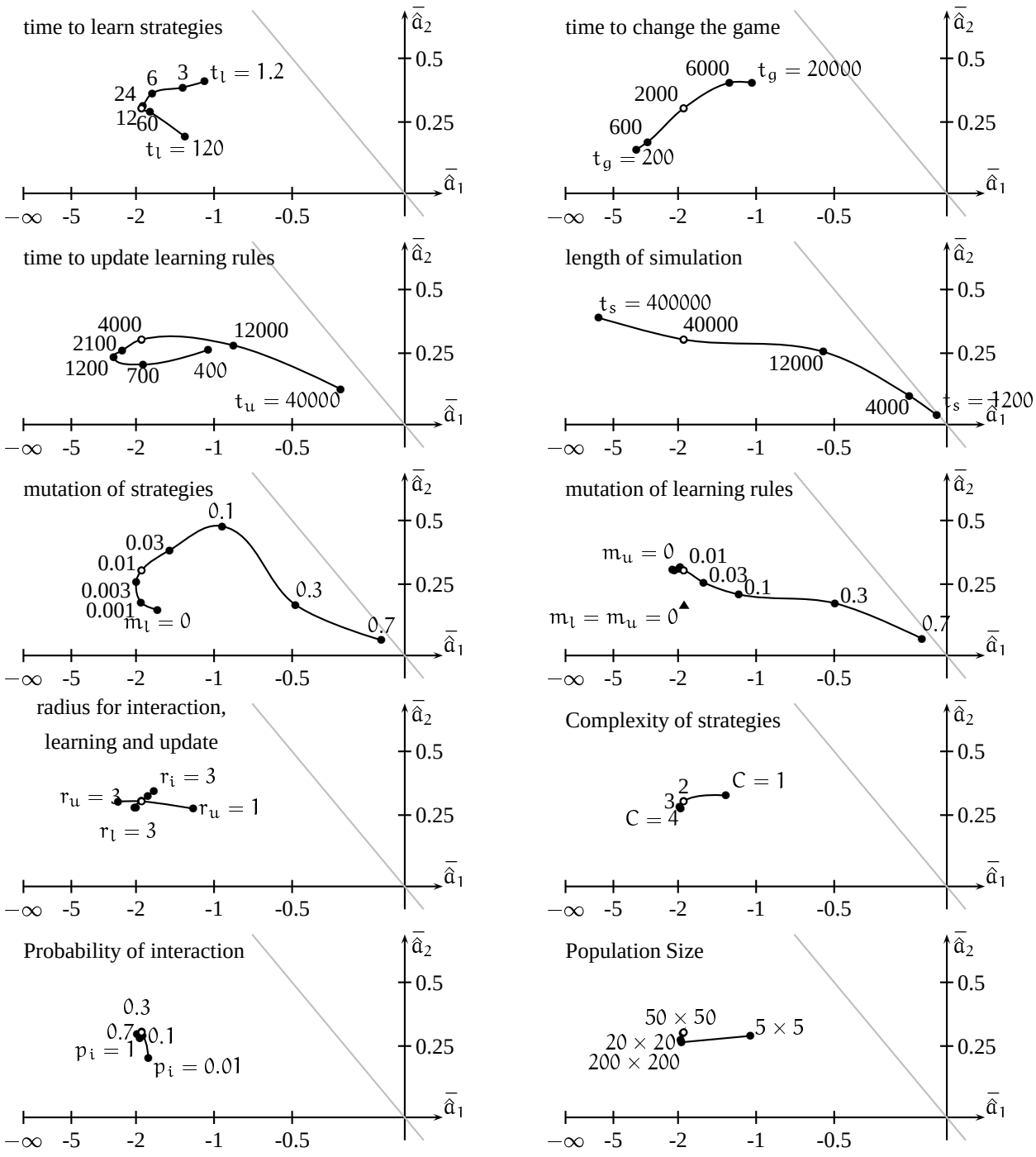


Figure 6: Dependence of $\hat{a}_1$ and $\hat{a}_2$ on the parameters of the learning rule.
Dots represent averages over the last 20% of 20 simulation runs respectively, each lasting for 40 000 periods. The white circle in each diagram represents averages of the reference parameters: 50×50 torus, sample a random player, $t_l = 24$, $t_u = 4000$, $t_g = 2000$, $m_l = m_u = 1/100$, $t_s = 40\,000$, $r_i = r_l = 1$, $r_u = 2$, $C = 2$.

	time to change				mutations		radius					
	strategy	game	learning rule	length of simulation	strategy	learning rule	interaction	learning	update learning rule	complexity	prob. of interact.	population size
	t_l	t_g	t_u	t_s	m_l	m_u	r_i	r_l	r_u	C	p_i	n
degree of locality	$+$	$-$			$-$		$-$	$+$		$+$		$+$
relative speed			$-$	$+$					$+$			
relative efficiency	$-$		$+$		$-$	$-$						

Table 1: Effects of simulation parameters on properties of learning rules.
$+$ and $-$ denote the direction of the effect an increase of a parameter has on speed, efficiency or locality.

parameters that we attribute more 'locality' in the following also lead to more weight on a player's own experience, i.e. more 'suspicion', thus, the ratio $\overline{\hat{a}_1}/\overline{\hat{a}_2}$ is smaller.

When games change rarely (i.e. t_g is large) or when player learn frequently (i.e. t_l is small) players have a better chance to find the 'long-run' strategy for a given game and, thus, become more similar, which reduces locality. Diversity among players may also be reduced by 'background noise' m_l since noise makes players more similar. Farsightedness (r_l), when learning, increases the effects of locality since it exposes learning players to samples that are more likely to be in a different situation. (This shows that being able to spot locality is actually one of its prerequisites). Likewise, situations become more diverse when the interaction neighbourhood r_i is small. Another source of heterogeneity is complexity of strategies, since this determines diversity of players' capabilities. More heterogeneity can finally also be due to a larger population (n).

Speed: The parameters t_s, t_u, and r_u influence the speed of the evolutionary process in the sense that they affect the frequency or the size of evolutionary steps of the learning rules. More speed allows learning rules to move away from the initial distribution (which has averages of the parameters of the learning rule $\overline{\hat{a}_1} = 0$, $\overline{\hat{a}_2} = 0$), thus, move farther to the left in the diagram.

The longer our simulations run (t_s), the more time learning rules have to develop and to move away from the initial distribution. Also, the more frequently we update learning rules (i.e. the larger t_u), the faster learning rules evolve and move away from the initial distribution. The farther we see (r_u) when updating a learning rule, the faster successful learning rules spread

through the population.

Noise: The parameters t_l, t_u, m_l, m_u may make the evolutionary process more 'noisy' in the sense that the direction of moves of learning rules becomes more random. This again keeps averages of the parameters of the learning rules closer to the initial distribution.

The more rarely learning rules are *used* to select strategies (i.e. the larger t_l), the less they gain experience, and the more they remain close to the initial distribution. The more rarely we update learning rules (i.e. the larger t_u), the more data is available to evaluate a learning rule, thus, the less noisy its development. The more strategies are perturbed when they are learned (m_l), the less it is possible to evaluate a learning rule's impact on success. The more learning rules are perturbed during the update process (m_u), the more they are pushed back to the initial distribution.

Notice, however, that changes in a some parameter may have conflicting effects. One example is the speed to update learning rules t_u: For very *small* values of t_u learning rules are updated too often to accumulate a reasonable amount of data on the success the rule. As a consequence the evolutionary process it too noisy to move away from their initial distribution. For very *large* values of t_u the data concerning the performance of learning rules might be rather reliable, however, individual learning becomes slow. This again means that learning rules do not manage to move away from their initial distribution.

In other words: Updating a learning rule implies taking advantage of information that is provided by neighbours while simultaneously ceasing to provide the information that the respective individual has collected in the past. Of course in the long run also an updating player is a source of information again, but at least in the short run updates are bad for the neighbourhood but good for the individual. This effect explains the turn in the t_u-curve.

Still, the discussion of figure 6 shows that whatever parameters we choose, learning rules turn out to have similar properties. Further, the dependence of learning rules on parameters seems to be fairly intuitive.

3.5 Stage Game Behaviour

In the previous sections we were concerned with the immediate properties of endogenous learning rules. In the following we want to analyse the impact that endogenous rules have on stage game behaviour.

Figure 7 on the next page shows proportions of stage game strategies for various games both for endogenous and for fixed learning rules. In simulations represented in figure 7 the underlying game changes every $t_g = 2000$ periods. We know from other simulations that

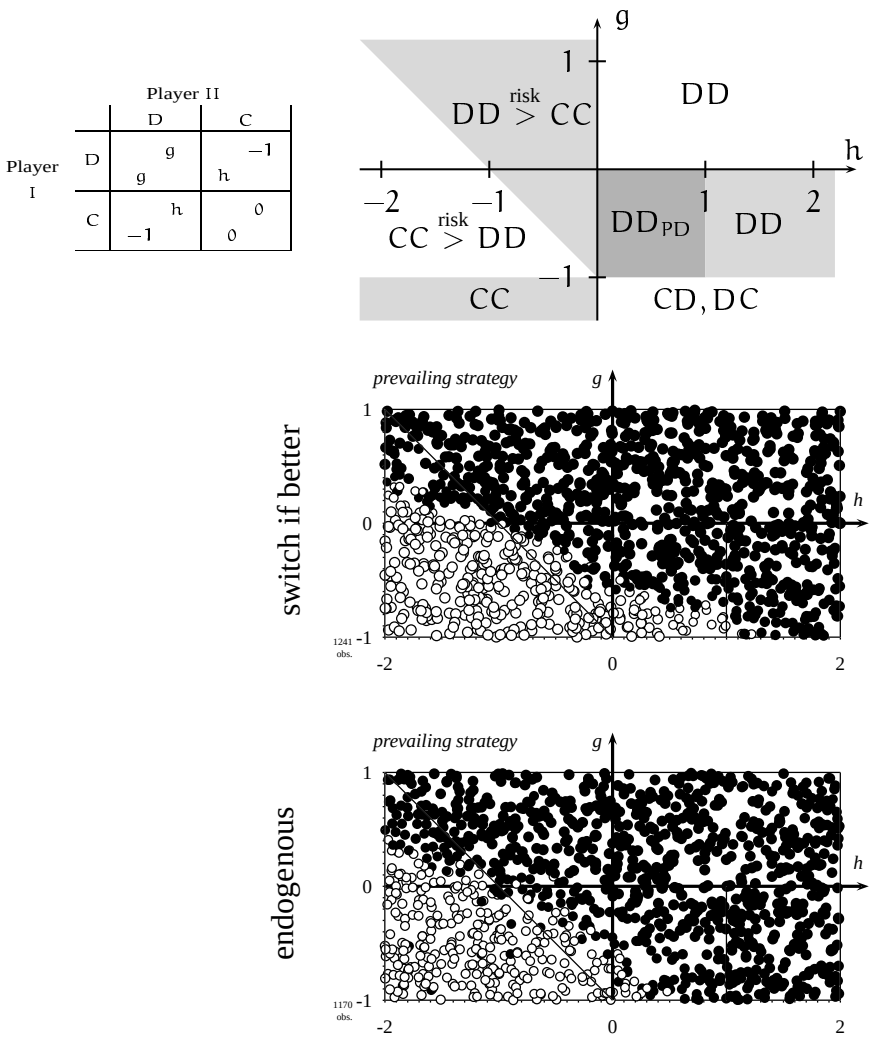


Figure 7: Stage game behaviour depending on the game. (○=most players play C, ●=most players play D). Parameters: 50×50 torus, $r_i = r_l = 1$, $r_u = 2$, sample a random player, $t_l = 24$, $t_u = 4000$, $t_g = 2000$, $m_l = 0.1$, $m_u = 0.1$, $t_s = 400\,000$.

during these 2000 periods strategies should have adapted to the new game[16]. Just before the game changes we determine the proportion of stage game strategies C and D. These proportions are represented in figure 7 as circles. The position of the circle is determined by the parameters of the game, g and h. The colour of the circle is white if the proportion of Cs is larger and black otherwise.

Figure 7 compares two cases: An exogenously given learning rule of the 'switch if better' type, approximated as $(\hat{a}_0, \hat{a}_1, \hat{a}_2) = (0, -100\,000, 100\,000)$ and the case of endogenous learning rules.

In both pictures two areas can be distinguished. One area where most of the simulations lead to a majority of C and another one where most simulations lead to a majority of D. We make two observations:

- The fixed learning rule 'switch if better', which is an approximation of the learning rules studied in the literature on local evolution with fixed learning rules[17], leads to results that are very similar to those observed in the literature.
  - There is cooperation for a substantial range of prisoners' dilemmas. Actually 30.3% of the 142 prisoners' dilemmas in this simulation lead to a majority of cooperating players.
  - In coordination games players do not follow the principle of risk dominance but another principle which is between risk dominance and Pareto dominance[18].
- Under endogenous learning the range of prisoners' dilemmas where most players cooperate shrinks to 10.2% of the 137 prisoner's dilemmas in the respective simulation. Behaviour in coordination games again does not follow risk dominance.

The first point is interesting to note, because it shows that the model that we study in this paper is comparable with the models studied in the literature on local evolution with fixed learning rules.

The second point shows that properties of network evolution discussed in the literature on local evolution with fixed learning rules persist, at least to some smaller degree, even with endogenous learning rules.

[16] See Kirchkamp (1995).

[17] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer and May (1993), Eshel, Samuelson and Shaked (1996), and Kirchkamp (1995).

[18] A very similar behaviour is found for the fixed learning rule 'copy the best strategy found in the neighbourhood' in Kirchkamp (1995).

4 Conclusions

In this paper we studied properties of endogenously evolving learning rules and the stage game behaviour that is implied by these rules. We compared endogenously evolving learning rules both with rules that are assumed in standard models on local evolution[19] as well as with those that turn out to be optimal in a global context[20].

Regarding the first comparison we find that our dynamics selects rules which are different from the ones commonly assumed in the literature on local evolution. In particular the learning rules which are selected following our dynamics are much less sensitive to changes in a sampled player's payoff. This 'suspicion' can be related to the fact that the sampled player's environment is different from the learning player's one.

Comparing endogenous rules from local evolution with *optimal* rules from a global model we find two differences: Endogenous rules are not symmetric and they often imply deterministic behaviour. The lack of symmetry in the learning rule is analogous to the lack of symmetry in a learning player's and the respective neighbours situation. The deterministic behaviour is a result of the lack of patience which is a consequence of the more than infinitesimal learning speed.

As far as the stage game behaviour is concerned we find that important properties of stage game behaviour, like cooperation for some prisoners' dilemmas and coordination not on risk dominant equilibria, is present both with fixed learning rules specified in the literature and with our endogenous learning rules, however, with endogenous rules to a more limited degree.

Besides the selection dynamics that we present here we have also analysed other selection dynamics. In Kirchkamp and Schlag (1995) we study dynamics where players use less sophisticated update rules than the OLS-model used in this paper. We have analysed models where players move only in the direction of the maximum of the OLS model, but do not adopt the estimate of the optimal rule immediately. Further we have analysed models where players do not estimate any model at all but instead copy successful neighbours. Both alternative specifications lead to similar properties of learning rules: Switching probabilities are less sensitive to changes in payoff of the neighbour and more sensitive to changes in payoffs of the learning player. Also properties of the induced stage game behaviour are similar: Both alternative specifications lead to cooperation for some prisoners' dilemmas and coordination not on risk dominant equilibria. Thus, we can regard the above results as fairly robust.

[19] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer and May (1993), Eshel, Samuelson and Shaked (1996), and Kirchkamp (1995).

[20] See Börgers and Sarin (1995), Schlag (1993, 1994).

References

AXELROD, R. (1984): *The evolution of cooperation*. Basic Books, New York.

BINMORE, K., AND L. SAMUELSON (1994): "Muddling Through: Noisy Equilibrium Selection," Discussion Paper B–275, SFB 303, Rheinische Friedrich Wilhelms Universität Bonn.

BONHOEFFER, S., R. M. MAY, AND M. A. NOWAK (1993): "More Spatial Games," *International Journal of Bifurcation and Chaos*, 4, 33–56.

BÖRGERS, T., AND R. SARIN (1995): "Naive Reinforcement Learning With Endogenous Aspirations," Second international conference on economic theory: Learning in games, Universidad Carlos III de Madrid.

ELLISON, G. (1993): "Learning, Local Interaction, and Coordination," *Econometrica*, 61, 1047–1071.

ELY, J. (1995): "Local Conventions," Mimeo, University of California at Berkely, Economics Department.

ESHEL, I., L. SAMUELSON, AND A. SHAKED (1996): "Altruists, Egoists and Hooligans in a Local Interaction Model," Tel Aviv University and University of Bonn.

HEGSELMANN, R. (1994): "Zur Selbstorganisation von Solidarnetzwerken unter Ungleichen," in *Wirtschaftsethische Perspektiven I*, ed. by K. Homann, no. 228/I in Schriften des Vereins für Socialpolitik, Gesellschaft für Wirtschafts- und Sozialwissenschaften, Neue Folge, pp. 105–129. Duncker & Humblot, Berlin.

HILGARD, E. R., D. G. MARQUIS, AND G. A. KIMBLE (1961): *Conditioning and Learning*. Appleton-Century-Crofts, New York, 2nd edition revised by Gregory Adams Kimble.

KIRCHKAMP, O. (1995): "Spatial Evolution of Automata in the Prisoners' Dilemma," Discussion Paper B–330, SFB 303, Rheinische Friedrich Wilhelms Universität Bonn.

KIRCHKAMP, O., AND K. H. SCHLAG (1995): "Endogenous Learning Rules in Social Networks," Rheinische Friedrich Wilhelms Universität Bonn, Mimeo.

LINDGREEN, K., AND M. G. NORDAHL (1994): "Evolutionary dynamics of spatial games," *Physica D*, 75, 292–309.

MAY, R. M., AND M. A. NOWAK (1992): "Evolutionary Games and Spatial Chaos," *Nature*, 359, 826–829.

——— (1993): "The Spatial Dilemmas of Evolution," *International Journal of Bifurcation and Chaos*, 3, 35–78.

SAKODA, J. M. (1971): "The Checkerboard Model of Social Interaction," *Journal of Mathematical Sociology*, 1, 119–132.

SCHELLING, T. (1971): "Dynamic Models of Segregation," *Journal of Mathematical Sociology*, 1, 143–186.

SCHLAG, K. H. (1993): "Dynamic Stability in the Repeated Prisoners' Dilemma Played by Finite Automata," Mimeo, University of Bonn.

——— (1994): "Why Imitate, and if so, How? Exploring a Model of Social Evolution," Discussion Paper B–296, SFB 303, Rheinische Friedrich Wilhelms Universität Bonn.