Simultaneous Evolution of Learning Rules and Strategies

Oliver Kirchkamp [1]

University of Mannheim, SFB 504, L 13, 15, D-68131 Mannheim, email

oliver@kirchkamp.de

Abstract

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

Previous work on local evolution assumes fixed learning rules while we study learning rules that are determined endogenously.

Simulations indicate that endogenous learning rules put more weight on a learning player's own experience than on the experience of an observed neighbour. Nevertheless stage game behaviour 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.

[1] 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 anony-

1 Introduction

In this paper we intend to study how strategies and learning[2] rules evolve simultaneously in a local environment. Models of local evolution of strategies with fixed learning rules have 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 (1998), and Kirchkamp (1995). In these models players either imitate the strategy of the most successful neighbour or the strategy with the highest average success in their neighbourhood respectively. Both rules are plausible and lead to the survival of cooperation in prisoners' dilemmas.

However, this phenomenon depends on the learning rule assumed. Other learning rules, for instance players that imitate agents with probabilities proportional to the success of observed strategies, do not lead to cooperation. Proportional imitation rules are arguably plausible since they are optimal when all members of a population are equally likely to interact with each other (a finding obtained by Börgers and Sarin (1995) and Schlag (1993, 1994)).

Since local evolutionary models are more sensitive than global models to the choice of learning rules we have to take special care. In contrast to Börgers, Sarin or Schlag we will not look for optimal rules. Instead we will use an evolutionary process not only to select strategies but to select learning rules as well. A result of our study will be that local evolution does not converge to the optimal global solution found by Börgers, Sarin or Schlag.

mous referees, and several seminar participants for helpful and stimulating comments.

[2] We have in mind here a descriptive definition of learning in the sense of a (persistent) 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.

We shall study a model where players have fixed positions on a network and who change their strategy using a rule that is based on imitation. Hence, our model has elements in common with Axelrod (1984); Nowak and May (1992, 1993), Nowak, Bonhoeffer and May (1993), Eshel, Samuelson, Shaked (1998), Kirchkamp (1995), and Lindgren and Nordahl (1994). However, while these writers assume players to use fixed learning rules we shall allow players to change their learning rule using a process that is based on imitation. We will neither allow players to travel[3] nor to optimise[4]. We will find that while endogenous learning rules are different from those assumed by the above authors stage game behaviour is not affected too much.

In section 2 we describe a model of local endogenous evolution of learning rules. In section 3 we discuss properties of these learning rules. Section 4 analyses the dependence of our results on parameters of the model. In section 5 we then study the implication of endogenous learning for the stage game behaviour. Section 6 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.

[3] See Sakoda (1971), Schelling (1971), Hegselmann (1994), Ely (1995).

[4] See Ellison (1993), Sakoda (1971) and Schelling (1971).

2.2 Stage Games

Players play games within neighbourhoods which have one of the shapes shown in figure 1.

[Fig. 1 about here.]

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.[5] In each period interactions among two neighbouring players take place independently from all others with probability p_{i}. Hence, in a given period players may have different numbers of interactions. A typical value for p_{i} is $1/2$. 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.[6] Changing games force players to adapt and, hence, learning rules to improve. We present results of simulations 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 with the following form:

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

When new games are selected parameters g and h are chosen randomly following a uniform 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 2).

[5] The subscript i will be used in the context of individual interactions.

[6] The subscript G will be used to denote changes in the underlying game.

[Fig. 2 about here.]

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}[7] in figure 2), all games with $g > -1$ and $h < 0$ are coordination games. In figure 2 equilibrium strategies are designated with *CC*, *CD*, *DC* and *DD*. The symbol $\overset{\text{risk}}{>}$ denotes risk dominance for games that have several equilibria.

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[8]. For many simulations we limit the number of states to less than three.

[7] Games with $g \in (-1,0)$ and $h > 1$ will not be called prisoners' dilemmas since *CC* is Pareto dominated by alternating between *CD* and *DC*.

[8] Each 'state' of a Moore automaton is described with the help of 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 initially plays *C* and remains in this state whatever the opponent does; the other always plays *D*).

There are 26 automata with one or two states. For example 'grim' is a two-state automaton. In the initial state it plays *C*. It stays there unless the opponent plays *D*. Then the automaton switches to the second state, where it plays *D* and stays there forever. Other popular two-state automata include 'tit-for-tat', 'tat-for-tit', etc.

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

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. Learning probabilities will be denoted $1/t_{\mathrm{L}}$ and range from $1/6$ to $1/120$. t_{L} then denotes the average time between two learning events.[9] Hence, learning occurs less frequently than interaction. Still, learning occurs more frequently than changes of the stage game and updates of the learning rule itself (see below).

When a player updates the repeated game strategy he or she randomly samples one member of the neighbourhood and then applies the individual learning rule. In one respect this approach is simpler than the rules discussed in the literature[10] which simultaneously use information on all neighbours from the learning neighbourhood. To facilitate comparison with the literature and as an extension of the base model we study endogenous multi-player rules in section 4.1. We find that endogenous multi-player rules have properties that are very similar to our endogenous single-player rules. To further facilitate comparison with the literature we study fixed single player rules in section 5 and find that they are very similar to the corresponding fixed multi-player rules from the literature.

Learning occurs in neighbourhoods which have the same shape as neighbourhoods for interaction (see again figure 1). We denote the size of the neighbourhood for

[9] The subscript L will be used in the context of changes of strategies (either through learning or through mutation).

[10] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonnhoeffer, and May (1993), Eshel, Samuelson, and Shaked (1998), Kirchkamp (1995).

learning with the symbol r_{L}.

A player's learning rule uses the following information:

(1) The learning player's repeated game strategy.

(2) The own payoff u_{own} obtained with this player's repeated game strategy, i.e. the average payoff per interaction 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 obtained while using this repeated game strategy.

Learning rules are characterised by a vector of three parameters $(\hat{a}_0, \hat{a}_1, \hat{a}_2) \in \Re^3$. 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

$$p(u_{\mathrm{own}}, u_{\mathrm{samp}}) = \langle \hat{a}_0 + \hat{a}_1 u_{\mathrm{own}} + \hat{a}_2 u_{\mathrm{samp}} \rangle \tag{1}$$

where $\langle \cdot \rangle$ is defined as

$$\langle x \rangle := \begin{cases} 1 & \text{if } x > 1 \\ 0 & \text{if } x < 0 \\ x & \text{otherwise} \end{cases} . \tag{2}$$

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

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. This can be interpreted as a higher or lower inclination to try something new. Choosing $(\hat{a}_0, \hat{a}_1, \hat{a}_2)$ a player determines three ranges of payoffs: One where switching will never occur (i.e. $p(u_{\mathrm{own}}, u_{\mathrm{samp}}) =$

0), a second one where behaviour is stochastic (i.e. $p(u_{\text{own}}, u_{\text{samp}}) \in (0,1)$), and a third one where switching always occurs (i.e. $p(u_{\text{own}}, u_{\text{samp}}) = 1$).

Of course, one or even two of these ranges may be empty. Our specification, indeed, allows for three types of rules: Rules which are (almost) always deterministic, rules which always behave stochastically, and rules which react determinstically for some payoffs and stochastically for others. An example of a deterministic rule (‘switch if better’) is $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (0, -\overline{a}, \overline{a})$ with $\overline{a} \to \infty$. An example of a rule that always implies stochastic behaviour for the game given above is $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (1/2, -\overline{a}, \overline{a})$ with $1/(4\overline{a}) > \max(|g|, |h|, 1)$.

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. In cases where this noise term becomes small our rule approximates Binmore and Samuelson (1994) with $(\hat{a}_0, \hat{a}_1, \hat{a}_2) := (\Delta, -\overline{a}, \overline{a})$ and $\overline{a} \to \infty$.

2.4.1 Normalisation

We map parameters $(\hat{a}_0, \hat{a}_1, \hat{a}_2) \in \Re^3$ into $(a_0, a_1, a_2) \in [0,1]^3$ using the following condition (see also figure 3):

$$\hat{a}_i \equiv \tan\left(\pi a_i - \frac{\pi}{2}\right) \qquad \forall i \in \{0,1,2\}\,. \tag{3}$$

[Fig. 3 about here.]

Evolution operates on the normalised values $(a_0, a_1, a_2) \in [0,1]^3$ in order not to rule out deterministic rules: Learning rules from the literature are often deterministic. They can be represented as rules whose parameter values $\hat{a}_i$ are infinitely large or small. Since our evolutionary process is modeled in finite time it can hardly converge to infinite values. However, it could converge to finite values of our normalisation and still denote determinstic rules.

2.4.2 Mutations of Repeated Game Strategies

We introduce mutations of repeated game strategies to show that simulations are particularly robust. However, as we see in section 4, we do not need mutations for our results.

When a player learns a repeated game strategy, sometimes the learning process described above 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. Such a 'mutation' occurs with a fixed probability m_{L}. We consider mutation rates between 0 and 0.7.

2.5 Exogenous Dynamics that Select Learning Rules

From time to time players revise their learning rules. We assume that players update learning rules independently with probability $1/t_{\mathrm{u}}$ in each period. t_{u}, hence, denotes the average time between two updates.[11] We consider learning rates $1/t_{\mathrm{u}}$ among $1/40\,000$ to $1/400$. If not mentioned otherwise $1/t_{\mathrm{u}} = 1/4000$. In particular, updates of learning rules occur more rarely than updates of strategies or changes of games. Given that updates of learning rules do not occur very often it is reasonable to assume that players make a larger effort to update learning rules: Firstly, all neighbours are sampled (and not only a single neighbour as for updates of

[11] The subscript u will be used to denote changes of learning rules (either through learning or through mutation).

strategies). Secondly, the sampled data is evaluated more efficiently, now using a quadratic approximation.

The shapes of the neighbourhoods that are used to update learning rules are similar to those used for interaction and learning (see figure 1). We denote the size of the neighbourhood for updates of learning rules with the symbol r_{u}.

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

(1) The (normalised) parameters of their current learning rules a_0, a_1, a_2.

(2) The average payoffs per interaction that the players obtained while their current 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 use the estimated model to choose an optimal learning rule. To describe this decision process we assume that players approximate a quadratic function of the learning parameters to explain how successful learning rules are. Formally the quadratic function can be written as follows:

$$u(a_0, a_1, a_2) = c + (a_0, a_1, a_2) \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} +$$

$$+(a_0, a_1, a_2) \begin{pmatrix} q_{00} & q_{01} & q_{02} \\ q_{01} & q_{11} & q_{12} \\ q_{02} & q_{12} & q_{22} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} + \varepsilon \qquad (4)$$

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 behave as if they would maximise a quadratic model which is derived from an OLS-estimation.

Given the parameters $(c, b_0, b_1, b_2, q_{00}, \ldots, q_{22})$ of the estimated model, players choose the triple 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 less than 1% of all updates remaining the quadratic model might be unreliable. In this case players copy the most successful neighbour.

Figure 4 shows (only for one dimension) an example of a sample of several pairs of parameter a_i and payoff u (black dots) together with the estimation of the functional relationship (grey line) between a_i and u.

[Fig. 4 about here.]

2.5.1 Mutations of Learning Rules

We introduce mutations of learning rules to show that our simulation results are robust. However, as we show in section 4, we do not need mutations. Results without mutations are very simular to results with a small amount of mutations.

When a player updates his or her learning rule, with a small probability m_{L} it is not the above described update scheme which is used, but the player learns a random learning rule. This rule is chosen following a uniform distribution 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.

2.6 Initial Configuration

At the beginning of each simulation each player starts with a random learning rule that is chosen following a uniform distribution over $(a_0, a_1, a_2) \in [0,1]^3$. Thus, initially, the parameters $\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$ are distributed independently following a Cauchy distribution. Hence, in the first period expected values of the parameters of the learning rules are $\overline{\hat{a}}_1 = \overline{\hat{a}}_2 = \overline{\hat{a}}_3 = 0$. Each player starts with a random repeated game strategy, following a uniform distribution over the available strategies.

3 Results with Endogenous Learning Rules

3.1 Distribution over Learning Parameters

Figure 5 displays averages over 53 simulations on a 50×50 grid, each lasting for 400 000 periods.

[Fig. 5 about here.]

Figure 5 shows two different projections of (a_0, a_1, a_2). The left part displays the distribution over (a_1, a_2), the right part over $(a_0, a_1 + a_2)$. 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 normalised values but $\hat{a}_0$, $\hat{a}_1$, $\hat{a}_2$ instead which range from $-\infty$ to $+\infty$.[12]

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

Density plot: Different densities of the distribution are represented by different shades of grey. The highest density is represented by the darkest grey.[13]

Table of relative frequencies: The pictures in figure 5 also contain 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 proportion of players that use a learning rule with parameters in the particular range.

The left part of figure 5 shows that, while players are sensitive to their own payoff, they are substantially less sensitive to observed payoffs.

[12] The scaling of all axes follows the normalisation given in equation 3. Hence, the value "$\hat{a}_1 + \hat{a}_2$" actually represents $2 \cdot \tan(\pi \cdot (a_1 + a_2)/2 - \pi/2)$ and not $\hat{a}_1 + \hat{a}_2$. In the current context this difference is negligible.

[13] Densities are derived from a table of frequencies with 30×30 cells for each picture. We map logs of densities into different shades of grey. 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 grey. Thus, the white area represents densities smaller than 1% of the maximal density while areas with darker shades of grey represent densities larger than 1.9%, 3.7% 7.2%, 14%, 27% and 52% of the maximal density.

Sensitivity to own payoffs: Given that we start with a uniform distribution of a_1 and a_2, figure 5 initially would look like a smooth grey surface without any mountains or valleys. During the course of the simulations learning parameters change substantially. Similar to many learning rules discussed in the literature [14] $\hat{a}_1$ is often close to $-\infty$.

Insensitivity to sampled payoffs: The left part of figure 5 shows that 96.3% of all players use learning rules with $|\hat{a}_2| < |\hat{a}_1|$, i.e. rules that are more sensitive to own payoff than to sampled payoff. Notice that the initial distribution over parameters of the learning rule implies that 50% of all rules fulfil $|\hat{a}_2| < |\hat{a}_1|$.

We describe this kind of behaviour as ‘suspicious’ [15] in the following sense. The success of a neighbour’s rule may depend on players who are neighbours of the neighbour, but not neighbours of the player. A ‘suspicious’ player behaves like somebody who ‘fears’ that the sampled neighbour’s experience can not be generalised to apply to the player’s own case.

3.2 Probabilities of switching to a sampled learning rule

The learning rules as specified in equation 1 determine probabilities of switching to the observed repeated game strategy. Figure 6 shows the cumulative distribution of these probabilities. [16]

[Fig. 6 about here.]

[14] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonnhoeffer, and May (1993), Eshel, Samuelson, and Shaked (1998), Kirchkamp (1995).

[15] Of course, our agents can only behave *as if* they had feelings like suspicion.

[16] This figure is derived from a table of frequencies with 30 cells. The scaling of the horizontal axis follows the normalisation given in equation 3.

The horizontal axis represents $\hat{a}_0 + \hat{a}_1 u_{\text{own}} + \hat{a}_2 u_{\text{samp}}$. Following the learning rule 1 a player switches *stochastically* with probability $\hat{a}_0 + \hat{a}_1 u_{\text{own}} + \hat{a}_2 u_{\text{samp}}$ if this expression is between zero and one. Otherwise the player either switches *with certainty* or *not at all*.

Figure 6 shows that only in about 12% of all learning events $0 < \hat{a}_0 + \hat{a}_1 u_{\text{own}} + \hat{a}_2 u_{\text{samp}} < 1$, this means that in only 12% of all learning events a player's decision is a stochastic one. Our endogenous rules are neither fully stochastic (as those from Börgers, Sarin (1995) and Schlag (1993)) nor fully deterministic (as those from Axelrod (1984, p. 158ff), Nowak and May (1992), etc.).

3.3 Comparison with other Learning Rules

Above we mentioned two reference points for learning rules: Those learning rules that are assumed to be *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 of adopting 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 of adopting a new strategy.

Let us next compare our endogenous rules with those rules that are seen to be optimal in a global setting[17]. As we have seen, our rules differ in two respects from those that are optimal in a global model. Firstly, as discussed in section 3.1,

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

they are more sensitive to changes in a learning player's own payoff than to changes in an observed neighbour's payoff. Secondly, as mentioned in section 3.2, players following endogenous rules quite often switch with certainty. Why is the outcome of our *evolutionary process* so different from *optimal* rules?

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

Optimal learning rules always switch stochastically in order to evaluate information slowly but efficiently. Even small differences in payoffs are translated into different behaviour. Learning slowly is not harmful in Börgers and Sarin's or in Schlag's model. Their players do not care whether the optimal strategy is only reached after an infinite time. In our evolutionary process, however, discounting is involved through the regular update of learning rules. Learning rules that quickly achieve sufficiently good results outperform rules that slowly approach the optimum. Evolution of learning rules at a speed that is more than infinitesimal may lead to deterministic behaviour.

4 Dependence on Parameters

Our comments so far have been based on a particular parameter combination. Changing the parameters, however, does not affect our main results. We will consider an alternative rule to sample neighbours in section 4.1. We will study changes in the other parameters in section 4.2.

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 let us assume in this section that players sample the most successful neighbour, i.e. the neighbour with the highest payoff per interaction for the current repeated game strategy, measured over the lifetime of the repeated game strategy.

Figure 7 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 5.

[Fig. 7 about here.]

While the picture is more noisy than figure 5 the properties of learning rules are the same as in section 3.1. Players are quite sensitive to changes in their own payoff but less sensitive to changes in the sampled neighbours' payoff.

We suspect that the additional noise stems from the reduced evolutionary pressure on learning rules. Since only 'best' rules are available, identifying good rules is too easy. In figure 7 we 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. This might be reasonable, since 'whatever you see' is already the best available strategy in your neighbourhood under this selection rule.

4.2 Other Parameter Changes

Figure 8 shows the effect of various changes in the other parameters. We always start from the same combination of parameters as a reference point and then vary one of the parameters, keeping all the others fixed. The reference point is a simulation on a torus of size 50×50, where the interaction neighbourhood and learning neighbourhood both have the same size $r_{\mathrm{i}} = r_{\mathrm{L}} = 1$ while the neighbourhood that is used when updating the learning rule has size $r_{\mathrm{u}} = 2$. To learn a new repeated game strategy players randomly sample a neighbour. Learning occurs on average every $t_{\mathrm{L}} = 24$ periods [18] . The underlying game is changed every $t_{\mathrm{G}} = 2000$ periods. Players update their learning rule on average every $t_{\mathrm{u}} = 4000$ periods [19] . The mutation rate for learning as well as for update of learning rules is $m_{\mathrm{L}} = m_{\mathrm{u}} = 1/100$. Simulations last for $t_{\mathrm{s}} = 40\,000$ periods. [20] Thus, except for the simulation length, parameters are the same as those for figure 5.

[Fig. 8 about here.]

Figure 8 shows averages [21] of $\hat{a}_1$ and $\hat{a}_2$ for various changes in the parameters. Each dot represents a parameter combination that we simulated. To make the underlying pattern clearer, 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 grey.

The main result is that for *all* parameter combinations we find relative sensitivity to the player's 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.

[18] Remember that we assume learning to be an independent random event that occurs for each player with probability $1/t_{\mathrm{L}}$.

[19] We also assume update of learning rules to be an independent random event that occurs for each player with probability $1/t_{\mathrm{u}}$.

[20] The subscript s will be used to denote the length of the simulation.

[21] These are arithmetic averages of a_1 and a_2, taken over 20 different simulations runs, which are randomly initialised. The average values of a_1 and a_2 are then transformed into $\hat{a}_1$ and $\hat{a}_2$ using equation 3.

Notice that we do not *need* mutations for our results. However, the simulations are robust with respect to mutations. To show that we can dispense with both kinds of mutations simultaneously we ran a simulation where $m_{\mathrm{L}} = m_{\mathrm{u}} = 0$ and show the result in the graph 'mutation of learning rules' in table 1 with a small triangle. While learning on average leads to a smaller value of $\hat{a}_2$ we still have $\overline{\hat{a}}_2 < -\overline{\hat{a}}_1$. On the other hand, we can introduce quite large probabilities of mutations (up to 0.7) and still have $\overline{\hat{a}}_2 < -\overline{\hat{a}}_1$.

Let us now discuss the sensitivity of learning rules to parameters of the evolutionary process in more detail. We distinguish three different classes of parameters, those who influence relative *locality*, relative *speed*, and relative *noise* of the evolutionary process. We will see that parameters values that describe a *slow* or *noisy* process, cause the distribution over the parameters of the learning rule to remain close to the initial distribution. Parameters values that describe a less *local* situation (for example the interaction radius is large, such that almost everybody interacts with everybody else) cause 'suspicious' behaviour to disappear gradually.

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

[Table 1 about here.]

Let us briefly discuss some of the dependencies on parameters:

4.2.1 Locality

In the following paragraph we will explain in which way the parameters t_{G}, t_{L}, m_{L}, r_{L}, C, and n influence diversity of players, and hence, locality of the evolutionary process. More locality makes a player's own experience more important, hence, the player will be more 'suspicious'. Thus, the ratio $\overline{\hat{a}}_1/\overline{\hat{a}}_2$ will be smaller.

When games change rarely (i.e. t_G is large) or when players learn frequently (i.e. t_L is small) they have a better chance of finding the 'long-run' strategy for a given game. Thus, they become more similar which reduces locality. Diversity among players may also be reduced by 'background noise' m_L since noise makes players more similar. A larger learning radius (r_L), increases the effects of locality since players will sample more neighbours that could be in a different situation. This shows that being able to spot locality is actually one of its conditions. Likewise, situations become more diverse when the interaction neighbourhood r_i is small. Another source of heterogeneity is that complexity of strategies also diversifies players' behaviour. A larger heterogeneity is also provided by a larger population (n).

4.2.2 Speed

The parameters t_s, t_u, and r_u influence the speed of the evolutionary process since they affect the frequency or the scale of learning steps. Higher speed allows learning rules to move away from the initial distribution, thus, to move farther to the left in the diagram.

The longer the evolutionary process runs (t_s), the more time learning rules have to develop and to move away from the initial distribution. Also, the more frequently players update learning rules (i.e. the larger t_u), the faster learning rules evolve and move away from the initial distribution. The farther players see (r_u) when updating a learning rule, the faster successful learning rules spread through the population.

4.2.3 Noise

The parameters t_L, t_u, m_L, m_u influence the noise within the evolutionary process. This again keeps averages of the parameters of the learning rules closer to the initial distribution.

When learning rules are rarely used to select strategies (i.e. when t_L is large), then players remain inexperienced, and learning rules are not pushed into a particular direction. When learning rules are rarely updated (i.e. when t_u is large) then more reliable data is available when success of a learning rule is evaluated. Developement is, as a result, less noisy. The more strategies are perturbed when they are learned (m_L), the more difficult it becomes 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 some parameters may have conflicting effects. One example is the speed t_u in updating learning rules: For very small values of t_u learning rules are updated too often to accumulate a reasonable amount of data on the success of the rule. As a consequence, the evolutionary process is too noisy to diverge from the initial distribution. For very large values of t_u the data on the performance of learning rules becomes more reliable while learning becomes slower. Again learning rules do not move away from their initial distribution. In other words: Updating a learning rule is beneficial for the updating player who takes advantage of information that is provided by neighbours. At the same time neighours suffer since the updating player ceases to serve as a reliable source of information. These two conflicting effects explain the turn in the t_u-curve.

We can conclude that, whatever parameters are choosen, learning rules have similar properties in the end. The dependence of learning rules on parameters seems to be intuitive.

5 Stage Game Behaviour

Having dealt with the immediate properties of endogenous learning rules let us now analyse the impact that endogenous rules have on stage game behaviour.

[Fig. 9 about here.]

Figure 9 shows the proportions of stage game strategies for various games both for endogenous and for fixed learning rules. In simulations represented in figure 9 the underlying game changes every $t_{\mathrm{G}} = 2000$ periods. From other simulations we know that during these 2000 periods strategies have adapted to the new game[22]. Just before the game changes we determine the proportion of stage game strategies C and D. These proportions are represented in figure 9 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. Otherwise it is black.

Figure 9 compares two cases: An exogenously given learning rule of the type 'switch if better', 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. Two points are worth making:

- 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[23],

[22] See Kirchkamp (1995).

[23] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer, and May (1993), Eshel, Samuelson, and Shaked (1998), and Kirchkamp (1995).

leads to results very similar to those observed in this 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 between risk dominance and Pareto dominance [24] .

- Under endogenous learning the range of prisoners' dilemmas where most players cooperate shrinks to 10.2% of the 137 prisoners' dilemmas in the simulation. Behaviour in coordination games again does not follow risk dominance.

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

The second point shows that the 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.

6 Conclusions

In this paper we have studied properties of endogenously evolving learning rules and the stage game behaviour that is implied by these rules. We compared endogenously evolving learning rules with two types of other rules: 'Switch if better' rules that are assumed in standard models of local evolution [25] and rules that are optimal

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

[25] See Axelrod (1984, p. 158ff), Lindgren and Nordahl (1994), Nowak and May (1992, 1993), Nowak, Bonhoeffer, and May (1993), Eshel, Samuelson, and Shaked (1998), and

in a global context[26].

We find that our dynamics selects rules which are different from those commonly assumed in the literature on local evolution. In particular, the learning rules selected on the basis of 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 that of the learning player.

The rules that emerge from our local evolutionary process differ from rules that are optimal in a global model. Our rules are not symmetric and they often involve deterministic behaviour. The lack of symmetry in the learning rule is analogous to the lack of similarity between the situation of a learning player and that of the neighbours. The deterministic behaviour is a result of the lack of patience which is a consequence of the more than infinitesimal learning speed.

Stage game behavior of endogenous rules differs gradually from behaviour found with 'switch if better'. Cooperation for a range of prisoners' dilemmas and coordination not on risk dominant equilibria is present with 'switch if better' rules as well as with our endogenous learning rules, however, with endogenous rules to a more limited degree.

Besides the selection dynamics that we have presented 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. We have also analysed models where players do not estimate any model at all but copy successful neighbours instead. Both alternative specifi-

Kirchkamp (1995).

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

cations lead to similar learning rule properties. Probabilities of switching are less sensitive to changes in the neighbour's payoff and more sensitive to changes in the learning player's payoff. The properties of the induced stage game behaviour are also 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.

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 (1998): “Altruists, Egoists, and Hooligans in a Local Interaction Model,” *The American Economic Review*, 88, 157–179.

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.

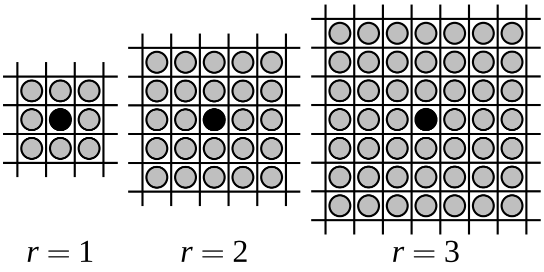
Sakoda, J. M., 1971, The Checkerboard Model of Social Interaction, *Journal of Mathematical Sociology,* 1, 119–132.

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.

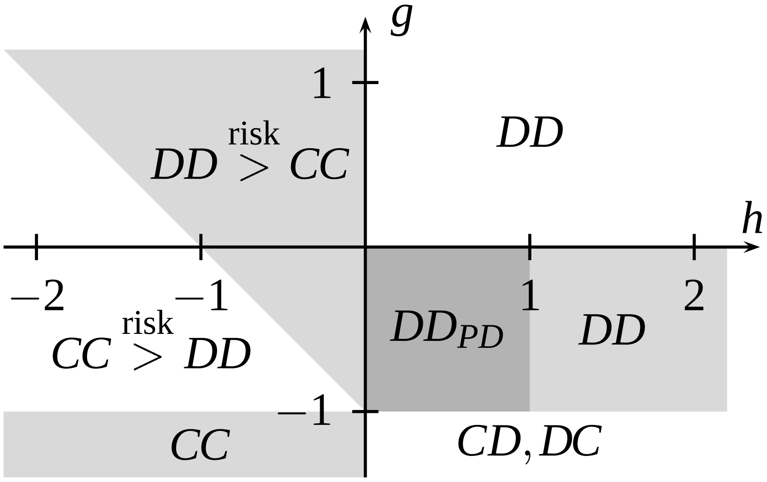
——— , 1994, Why Imitate, and if so, How? Exploring a Model of Social Evolution, Discussion Paper B–296, SFB 303, Rheinische Friedrich Wilhelms Universität Bonn.

List of Figures



$r = 1$ $r = 2$ $r = 3$

Fig. 1. Neighbourhoods of different sizes



g
1
$DD \overset{\text{risk}}{>} CC$
DD
h
-2
-1
1
2
DD_{PD}
DD
$CC \overset{\text{risk}}{>} DD$
-1
CC
CD, DC

Fig. 2. The space of considered games.

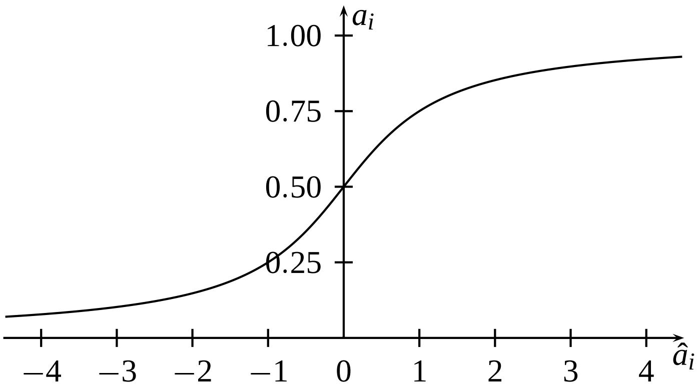


Fig. 3. Mapping from $(\hat{a}_0, \hat{a}_1, \hat{a}_2) \in \Re^3$ into $(a_0, a_1, a_2) \in [0.1]$

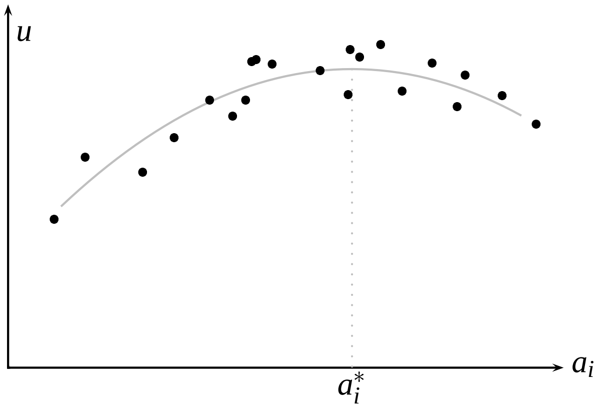


Fig. 4. An example of samples of pairs of parameters and payoffs (black) which are used to estimate a functional relationship (grey) between a_i and u. Given this relationship an optimal value a_i^* is determined.

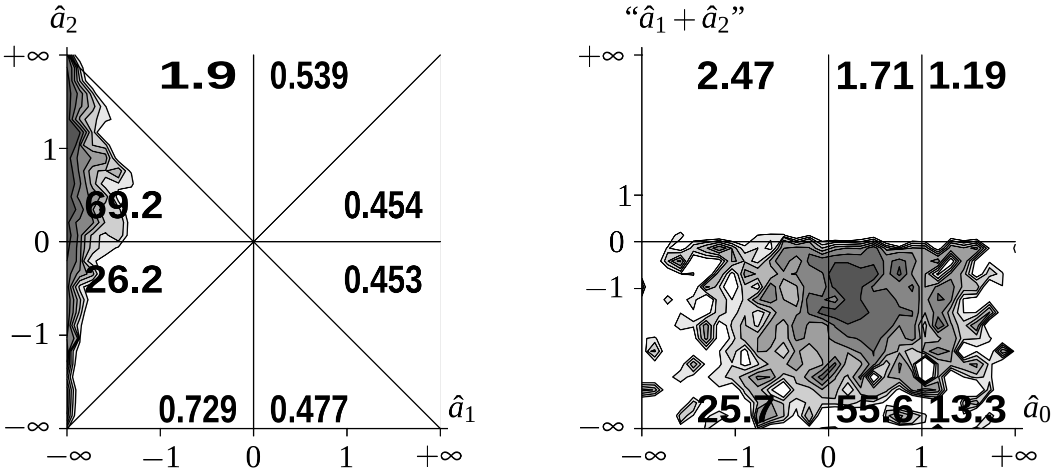


Fig. 5. Long run distribution over parameters of the learning rule (a_0, a_1, a_2). Average over 53 simulations runs on a torus of size 50×50 with 2-state automata. Neighbourhoods have sizes $r_\mathrm{i} = r_\mathrm{L} = 1$, $r_\mathrm{u} = 2$. Relative frequencies are given as percentages. Simulations last for $t_\mathrm{s} = 400\,000$ periods, interactions take place with probability $p_\mathrm{i} = 1/2$, repeated game strategies are learned from a randomly sampled player with probability $1/t_\mathrm{L} = 1/24$, learning rules are changed with probability $1/t_\mathrm{u} = 1/4000$, new games are introduced with probability $t_\mathrm{G} = 1/2000$, mutations both for repeated game strategies and for learning rules occur at a rate of $m_\mathrm{L} = m_\mathrm{u} = 1/100$.

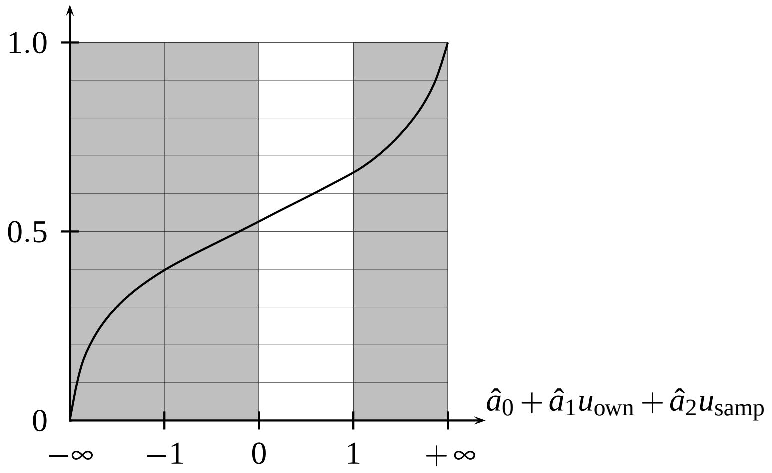


Fig. 6. Cumulative distribution of probabilities of switching, given the learning rules from figure 5.

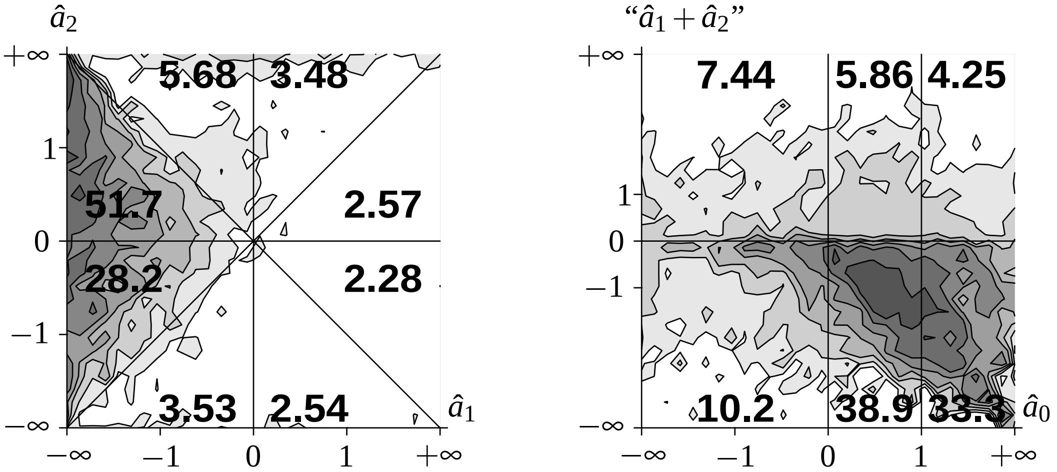


Fig. 7. Long run distribution over parameters of the learning rule (a_0, a_1, a_2). Average over 181 simulation runs, each lasting for 400 000 periods. Relative frequencies are given as percentages. Parameters are as in figure 5, except that learning players sample the most successful neighbour.

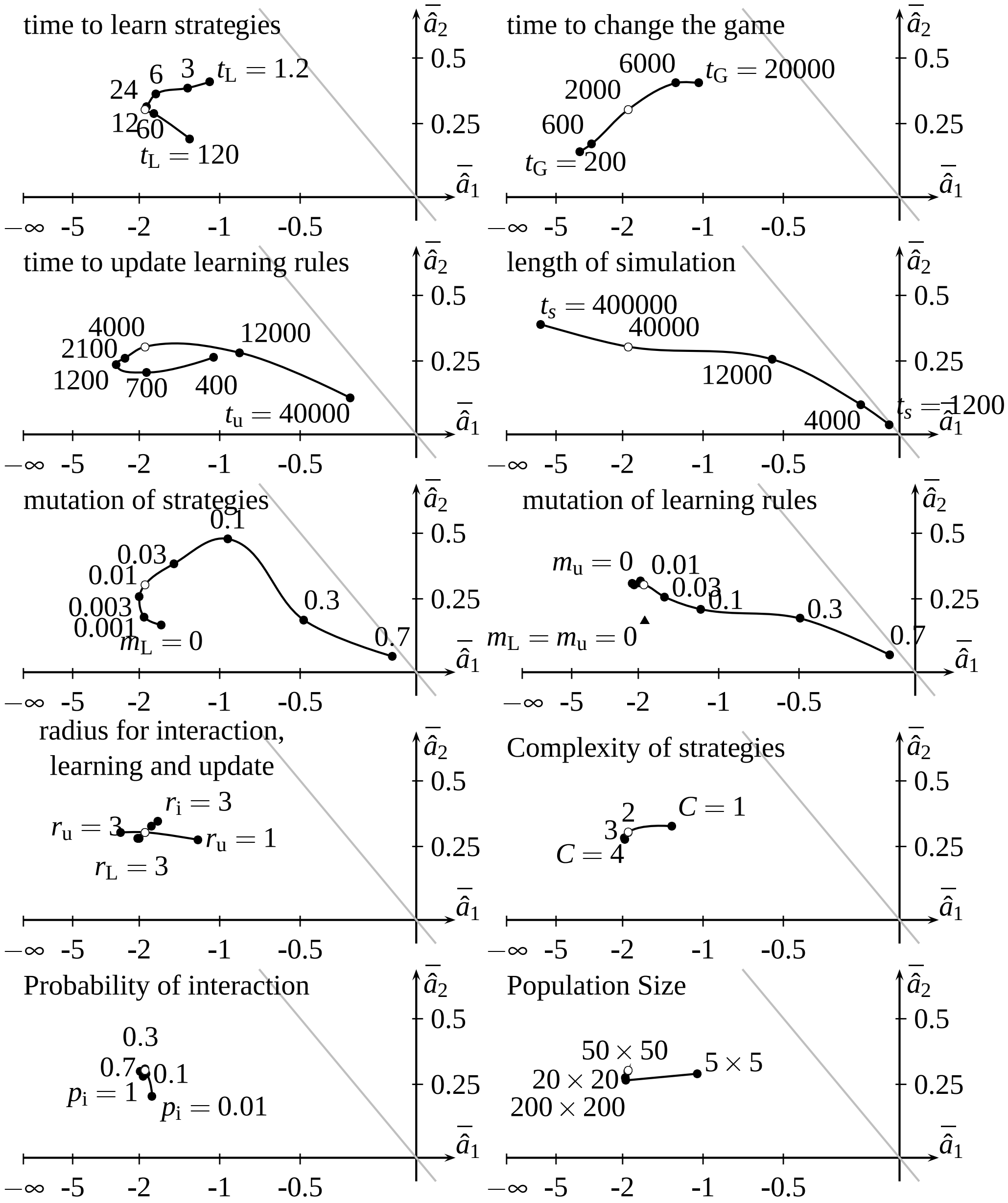
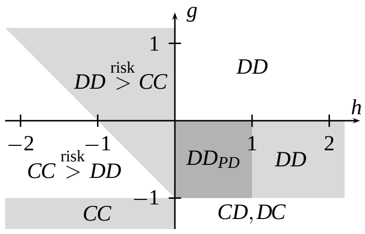


Fig. 8. Dependence of $\hat{a}_1$ and $\hat{a}_2$ on the parameters of the learning rule.

Dots represent averages over the last 20% of 20 simulation runs, 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, $t_L = 24$, $t_u = 4000$, $t_G = 2000$, $m_L = m_u = 1/100$, $t_s = 40\,000$, $r_i = r_L = 1$, $r_u = 2$, $C = 2$. The grey line shows $\overline{\hat{a}}_2 = -\overline{\hat{a}}_1$.

		Player *II*	
		D	*C*
Player *I*	*D*	g (II) g (I)	-1 (II) h (I)
	C	h (II) -1 (I)	0 (II) 0 (I)



g
1
$DD \overset{\text{risk}}{>} CC$
DD
h
−2
−1
1
2
$CC \overset{\text{risk}}{>} DD$
DD_{PD}
DD
−1
CC
CD, DC

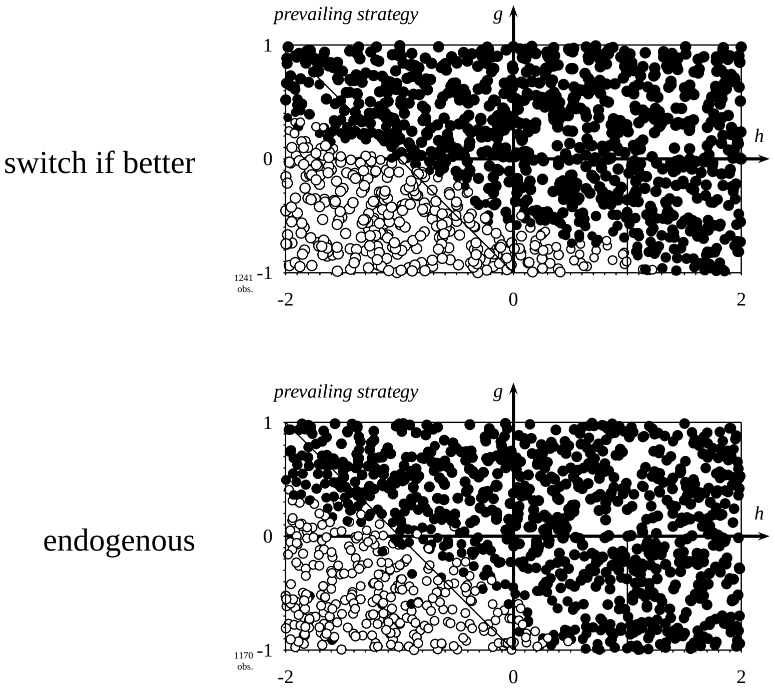


Fig. 9. Stage game behaviour depending on the game. (○=most players play *C*, ●=most players play *D*). Parameters: 50×50 torus, $r_{\mathrm{i}} = r_{\mathrm{L}} = 1$, $r_{\mathrm{u}} = 2$, sample a random player, $t_{\mathrm{L}} = 24$, $t_{\mathrm{u}} = 4000$, $t_{\mathrm{G}} = 2000$, $m_{\mathrm{L}} = 0.1$, $m_{\mathrm{u}} = 0.1$, $t_{\mathrm{s}} = 400\,000$.

List of Tables

	time to change			length of simulation	mutations		radius			complexity	prob. of interact.	population size
	strategy	game	learning rule		strategy	learning rule	interaction	learning	update learning rule			
	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 noise	$+$		$-$		$+$	$+$						

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.