

Evolution of Behaviour in the Prisoner's Dilemma

Kristian Lindgren

Institute of Physical Resource Theory
Chalmers University of Technology and
Göteborg University
S-412 96 Göteborg, Sweden

(e-mail: frtkl@fy.chalmers.se)

The term 'evolutionary dynamics' often refers to systems that exhibit a time evolution in which the character of the dynamics may change due to internal mechanisms. Such models are of course interesting for studying systems in which variation and selection are important components. A growing interest in these models does not only come from evolutionary biology, but also from other scientific disciplines. For example, in economic theory technological evolution as well as evolution of behaviour in markets may call for models with evolutionary mechanisms, see, e.g., [Andersen, 1994].

In this paper we focus on dynamical systems described by equations of motion that may change in time according to certain rules which can be interpreted as mutation operations. In models of this type, the variables typically correspond to the number or the mass of certain components (individuals/agents/organisms), or in more mechanistic (or spatial) models the position of certain components.

In the models that we shall discuss here, the equations of motion for the different variables (individuals) are usually coupled, which means that we have coevolutionary systems. The success or failure for a certain type of individual (species) depends on which other individuals are present. In this case, there is not a fixed fitness landscape in which the evolutionary dynamics climbs towards increasing elevation, but a position that at one time is a peak may turn into a valley. This ever changing character of the world determining the evolutionary path allows for complex dynamic phenomena.

In this paper, we make a distinction between evolutionary systems and adaptive systems. The equations of motion in an evolutionary system reflect the basic mechanisms of biological evolution, i.e., inheritance, mutation, and selection. In an adaptive system, other mechanisms are allowed as well, e.g., modifications of strategies based on individual forecasts on the future state of the system. But,

increasing the possibilities for individualistic rational behaviour does not necessarily improve the outcome for the species to which the individual belongs in the long run. An example of this difference is illustrated and discussed for one of the evolutionary lattice models.

The main part of this paper is devoted to a discussion of a number of evolutionary models based on the iterated Prisoner's Dilemma game as the interaction between individuals, see, e.g., [Axelrod, 1984, 1987]. Different aspects will be illustrated (i) by varying the trickiness of the game (iterated game, mistakes, misunderstandings, choice of payoff matrix), (ii) by introducing spatial dimensions, and (iii) by modifying the strategy space and the representation of strategies.

This paper is based on a review presented in [Lindgren, 1995]. The results in this paper concerning the finite memory strategies have to a large extent been reported before; for the mean-field model, see [Lindgren, 1992], and for the lattice model, see [Lindgren and Nordahl, 1994a]; see also the review article [Lindgren and Nordahl, 1994b]. The models based on the finitely iterated game as well as the models using finite automata strategies were first presented in [Lindgren, 1995].

Modelling evolutionary processes

As a basis for the models presented in the following sections, we use the three necessary mechanisms usually associated with biological evolution:

- inheritance of genetic information
- variation (mutations)
- the feedback effect of natural (or artificial) selection

Furthermore, to get a potential for evolution, the individuals need to be provided with a difficult "task" to solve, i.e., non-trivial interactions with an environment, which in these models means the rest of the individuals in the population. The Iterated Prisoner's Dilemma game (IPD) offers, in its many variations, a problem of choosing a successful strategy for interactions between the participating agents.

The models constructed all consist of a population of agents interacting in pairs (either locally or globally) in the IPD game. Each agent uses a strategy defined by the genetic code of that agent. After the game those who have been successful in the game will get more offspring. In this reproduction mutations may alter the genetic code, and new strategies may thus enter the next generation.

The models have been constructed with the mechanisms of biological evolution in mind, but one could also consider the resulting dynamics as an effect of a 'learning process'. This perspective will fit better with an economic perspective of these models, but, as the dynamics below is defined, it is still a very limited kind of 'learning'.

Game-theoretic models and dynamics

The Prisoner's Dilemma game, described below, is a suitable model for the interaction between individuals in an evolving population. There are many variations of the game that offers a non-trivial problem for the individuals, and still this type of interaction can often be solved analytically given the genetic code for the strategies.

In the Prisoner's Dilemma (PD) game, two players simultaneously choose to cooperate (C) or to defect (D), without knowing the opponent's choice. If they both cooperate they share the highest total payoff and get R points each, but there is a temptation to defect because defection scores $T > R$ against a cooperating opponent who in that case gets the lowest score S . If both defect they share the lowest total payoff and get P points each. If $T > R > P > S$ and $2R > T + S$, the dilemma is that, in an isolated game, rational players choose to defect and then share the lowest total payoff, see Table 1. Cooperative behaviour is possible in the iterated game when the same players meet in a series of rounds. In that case it has been shown that tit-for-tat (TFT), a strategy that mimics the opponent's last action (and starts with C), is capable of establishing cooperation [Axelrod 1984].

	player 2 cooperates	player 2 defects
player 1 cooperates	(R , R)	(S , T)
player 1 defects	(T , S)	(P , P)

Table 1. The payoff matrix of the Prisoner's Dilemma game, showing the scores given to player 1 and 2, respectively.

In all models described in this paper, we have used variations of the Prisoner's Dilemma that can be solved analytically. This reduces the computational power required and allows for long simulations (in terms of generations) of the models.

Dynamics based on the iterated Prisoner's Dilemma with mistakes

The iterated game is easily modified to a more complicated one by introducing noise. In this way the game is made more 'tricky', and very simple cooperative strategies, like TFT, are not very successful. Noise could either disturb the actions so that the

performed action is opposite to the intended one (trembling hand), or the information telling a player about the opponent's action is disturbed (noisy channels) [Axelrod and Dion 1988]. In any case, the complication is sufficient to give rise to interesting evolutionary behaviour. (Another variation, that has been tried in evolutionary models, is the possibility for the players to choose whether or not to cooperate, see, e.g., [Stanley et al 1993, Batali and Kitcher 1995]; see also [Lindgren and Nordahl 1993].)

In the first interaction model, there is no noise, but a dilemma is introduced by iterating the game a fixed (known) number of rounds. In that case, the final round is an ordinary Prisoner's Dilemma, and then the next to last round will get a similar status, and so on.

In the second and third models, noise is introduced in the form of mistakes, i.e., the action performed differs from the intended one (given by the strategy), with a certain probability. On the other hand, the problem with the dilemma of the final round is removed and the game is made infinitely iterated. In the second model the individuals are equipped with finite memory strategies, while in the third model the more powerful representation of finite automata is used. In all cases the average score per round is calculated, for the finitely iterated game and for the infinitely iterated game, respectively, and used in the dynamics to determine the next generation of the population.

We shall demonstrate various evolutionary phenomena, for all three models by studying the behaviour in two completely different 'worlds', the mean-field model (all interact with all, i.e., no spatial dimensions), and the cellular automaton (CA) model (local interaction on a lattice with synchronous updating).

In the mean-field model the dynamics is based on the simple equations,

$$x_k' = x_k + d x_k (\sum_i s_{ki} x_i - \sum_{ij} s_{ij} x_i x_j), \quad k = 1, \dots, N \quad (1)$$

where x_k is the fraction of the population occupied by strategy k , and s_{ij} is the score for strategy i against j . The prime on the left hand side denotes the next time step, i.e., we are assuming discrete time. Since the double sum ($\sum s_{ij} x_i x_j$) equals the average score in the population, the total population size is conserved. The number of different strategies N may change due to mutations, which are randomly generated after each step of the population dynamics equations (1). We have used continuous variables x_k , with a cut-off value corresponding to a single individual, below which the strategy is considered extinct and removed from the population. The equations reflect a change of abundance for a species that is proportional to its own score minus the average

score in the population. Each time step is viewed as one generation, and in addition to the equations mutations are added. This means that generations are non-overlapping and that reproduction is asexual, i.e., no combination of genetic code from two parents.

In the CA model, it is assumed that each individual interact with its four nearest neighbours on a square lattice. The average score for each individual (or lattice site) is calculated, and this determines which individuals that will be allowed to reproduce. In each neighbourhood (consisting of a cell with its four nearest neighbours), the individual with the highest score reproduces in the middle cell, see Figure 1. (Ties are broken by adding a small random number to the each player's score.) The offspring inherits the parent's strategy, possibly altered by mutations to be described below in detail for the different models.

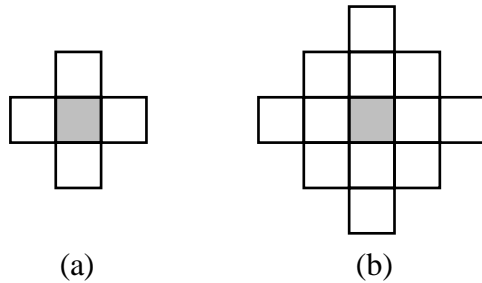


Figure 1 (a) In the cellular automaton (CA) model, each individual play the iterated Prisoner's Dilemma with each of its four nearest neighbours. Simultaneously, for each neighbourhood, the player with the highest average score per round reproduces in the "grey" cell. (b) Since the score of the nearest neighbours depends on the strategies present in the next nearest cells, we get a CA with interaction radius 2.

Depending on the choice of payoff matrix the evolutionary paths may look very different. In the CA model, since the dynamics is determined only by sign of score differences, there are only two independent parameters in the payoff matrix [Lindgren and Nordahl 1994]. Therefore, we have chosen to study the parameter region given by $R = 1$, $S = 0$, $1 < T < 2$, and $0 < P < 1$. In the mean-field model, the *value* of the score difference is also important, since the dynamics depends on how fast a score difference changes the composition of the population. Therefore, as a third parameter we can use the growth constant d in Eqs. (1), while we keep R and S fixed and vary T and P as in the CA model. In the mean-field simulations we have assumed $d = 0.1$. (This results in a growth that is slow enough to view Eqs. (1) as rough approximations to a set of ordinary differential equations, even though this is not the intention.) In all models, where noise is present, we have used an error probability of $e = 0.01$.

The two types of evolutionary dynamics described above could also be interpreted as a 'learning process'. In the mean-field model, we could say that each individual checks

if another randomly picked individual in the population did get a higher score, and, if so, switches to that strategy with a probability proportional to the score difference. The growth constant d then reflects the conservatism in the system. In the lattice model each strategy chooses the strategy of the most successful individual in its immediate neighbourhood. This may not, though, be the most successful strategy in the position of this individual, since neighbouring cells may see different opponent strategies. If the individual chooses the strategy that would be best in the present neighbourhood, the dynamics may change drastically.

Finite number of rounds

If the iterated game is finite and the number of rounds is known, the dilemma of the single round PD game remains. Assume that we have two players cooperating, for example, by using the Tit-for-tat strategy. When the two players enter the final round of their iterated game, they face the single round dilemma and profit maximizing behaviour calls for the defect action. But, if we know that our opponent is using a rational strategy, the next to last round will be a dilemma of the same type, and we should defect in that round too. Then the third round from the end will be under consideration, and so on. This implies that the only Nash equilibrium is given by both players defecting all rounds. This does not mean, though, that defecting all the time is the best strategy. Which strategy is the best depends on which other strategies are present in the population. If the population is dominated by, for example, strategies that play TFT for all rounds except the last two when they defect, it is of course a bad strategy to start with defection from the beginning. Instead, if one would know the opponent's strategy, one should start with TFT, i.e., cooperate, but switch to ALLD (always defect) the round before the opponent is to defect.

There are various ways to construct an evolutionary model of the PD game iterated n rounds. Here, we shall pick a simple example and look at a strategy set containing strategies playing TFT for k rounds and playing ALLD for the remaining $n-k$ rounds. We denote these strategies F_k , with $k = 0, \dots, n$. The score s_{jk} , for a strategy F_j against F_k , is then given by

$$s_{jk} = \begin{cases} j R + T + (n-j-1) P, & \text{if } j < k, \\ j R + (n-j) P, & \text{if } j = k, \\ k R + S + (n-k-1) P, & \text{if } j > k. \end{cases} \quad (2)$$

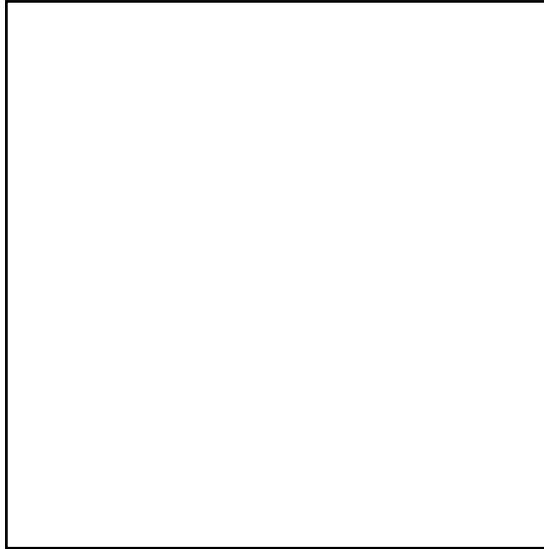


Figure 2. The interaction is given by the PD game iterated $n = 7$ rounds. Initially, the population (of 1000 individuals) consists only of the strategy F_7 , or TFT, but due to the dilemma of the fixed number of rounds, the evolution leads to strategies starting to defect earlier in the iterated game. The final state is then a fixed point, in which all play F_0 , or ALLD.

The population dynamics for a fully interacting population (mean-field model) can then be described by Eqs. (1), with mutations at a rate of 10^{-5} , so that a strategy F_i is replaced by a randomly chosen F_j . In Figure 2, the evolutionary path, starting with the strategy F_n , or TFT, shows that cooperative behaviour is exploited and that the evolution ends in the ALLD state.

Cooperation cannot be maintained since there will always appear exploiting strategies that grow at the expense of the more cooperative ones. It is well known that the introduction of spatial dimensions may increase the possibility for cooperative behaviour, see e.g., [Boerlijst and Hoegeweg 1991, 1992, Nowak and May 1992, Lindgren and Nordahl 1994].

In the CA model, the individuals are put on a 128×128 square lattice with periodic boundary conditions. Contrary to what we saw in the mean-field case, cooperation is maintained for a large part of parameter space defining the PD game. This is made possible either by the presence of spatio-temporal patterns, like waves of strategies sweeping over the system, see Figure 3 b and c, or by formation of stable islands of more cooperative strategies in a background of slightly exploiting strategies, see Figure 3 a and d.

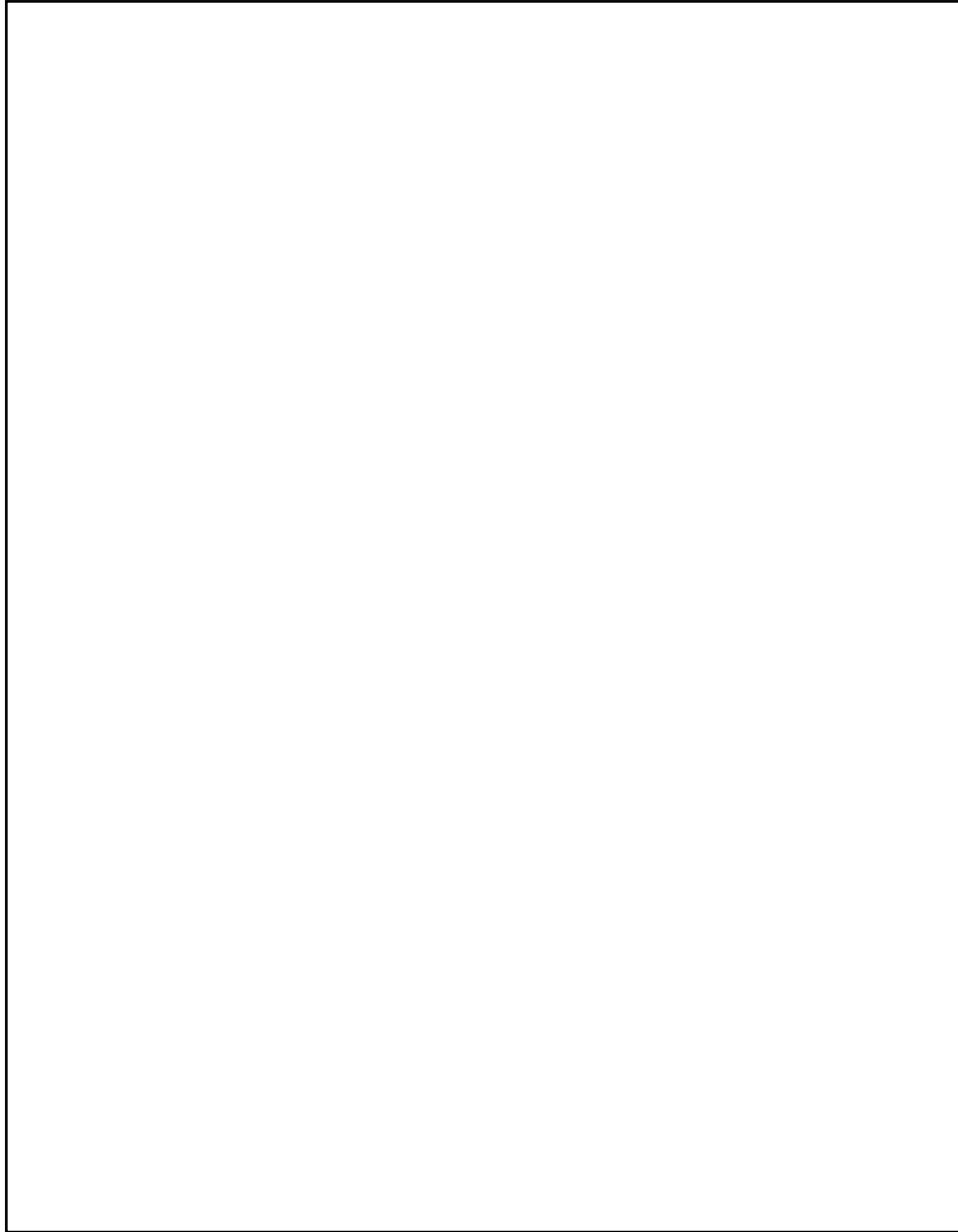


Figure 3. The evolution of F_k strategies playing the 7-round iterated Prisoner's Dilemma on a 128x128 lattice is shown for four different parameter values ($R = 1$, $S = 0$, T and P are varied). In all cases we see that the Nash equilibrium of a homogenous ALLD population is avoided. In (a), $T = 1.3$, $P = 0.3$, the state is frozen and dominated by F_7 (TFT), F_6 , and F_5 . In (b), $T = 1.667$, $P = 0.333$, and (c), $T = 1.3$, $P = 0.3$, there are waves running across the system constantly changing the pattern. In (d), $T = 1.7$, $P = 0.7$, there are frozen patterns that occasionally are modified by waves of TFT. Here, all strategies from F_0 (ALLD) to F_7 (TFT) are present, and they are represented in grey scale from white for ALLD to black for TFT.

Finite memory strategies

In the infinitely iterated Prisoner's Dilemma with mistakes, a simple type of strategy to consider is the deterministic finite memory strategy. This type of strategy may take into account the actions that have occurred in the game a finite number of rounds backwards, and deterministically choose a certain action.

A strategy can then be viewed as a look-up table, in which each entry corresponds to each of the possible finite-length histories that can be memorized by the strategy, see Figure 4. The symbols in the output column of the table denote the action to be performed, but there is a small chance (due to noise) that the action is altered. A different representation of finite memory strategies was used in an evolutionary model by Ikegami [1994].

The game between two finite memory strategies can be described as a stationary stochastic process, and the average score can therefore be calculated from a simple system of linear equations. The number of equations necessary equals 2^b , where b is the size of the history needed to determine the next pair of actions. For example, for two memory 1 strategies (as well as memory 2), both actions in the previous round are required and therefore the number of equations is 4.

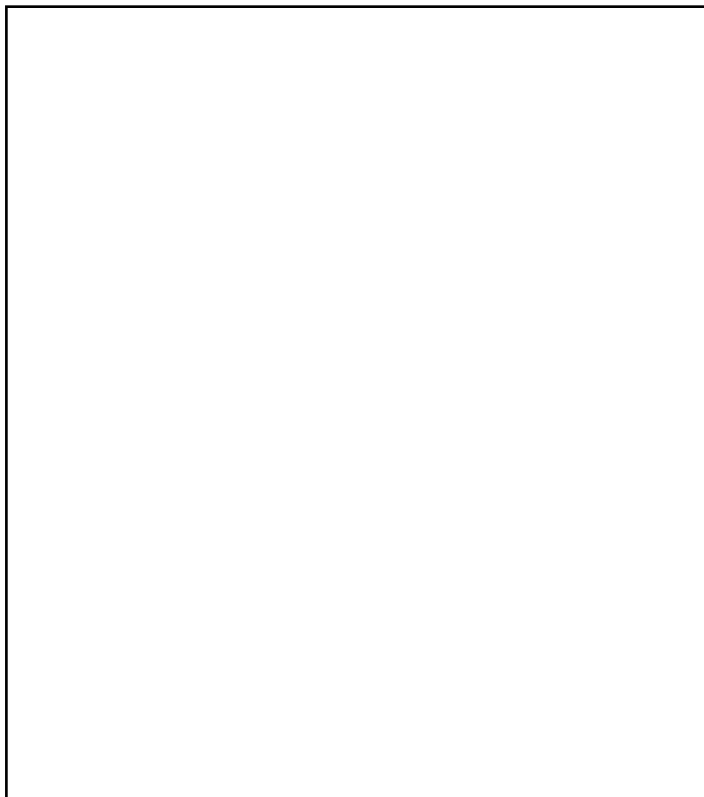


Figure 4. The genetic coding of finite memory strategies uses the look-up table. The length of the genome (size of the table) then carries the information on the memory capacity of the strategy, since the table double in size when memory increases by 1.

We use three types of mutations. The *point mutation* alters the symbol in the genome. The *gene duplication* attaches a copy of the genome to itself. In this way the memory capacity increases, but it should be noted that this mutation is neutral — the strategy is not changed, but a point mutation is required to make use of the increased capacity. The *split mutation* randomly removes the first or second half of the genome.

Mean-field model

In the mean-field model all interact with all, and the dynamics is given by Eqs (1). The number of individuals are 1000, the point mutation rate is $2 \cdot 10^{-5}$ and the other mutation rates are 10^{-5} .

In Figures 5 and 6, the evolutionary path is shown for two different choices of payoff matrices. The first one, $T = 1.6$ and $P = 0.3$, shows that the system finds a very stable state dominated by the strategy 1001. In the second simulation, at $T = 1.7$ and $P = 0.4$, this strategy does not succeed, but the evolution continues to strategies with longer memory. It turns out that the final state in Figure 6, consists of a large group of strategies having the same mechanism for correcting for mistakes in the iterated PD game. In both cases, the final population is cooperative and cannot be exploited by defecting strategies. For a more detailed discussion on simulations of the finite memory mean-field model, see [Lindgren 1992].

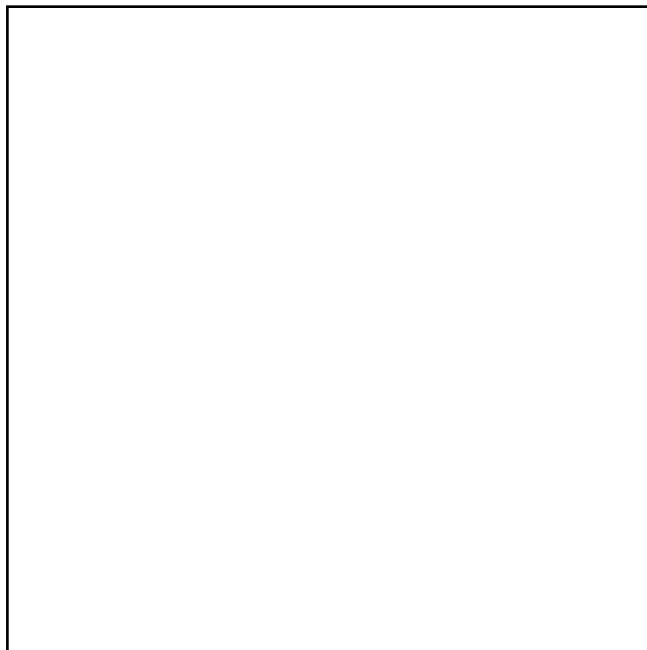


Figure 5. The evolution of finite memory strategies in the mean-field model is shown for 25000 generations, with the parameter values $T = 1.6$, $P = 0.3$. There is a meta-stable coexistence between 01 (TFT) and 10 (ATFT) before strategies of memory 2 takes over. The dominating 1001 strategy is actually evolutionarily stable for these parameter values.

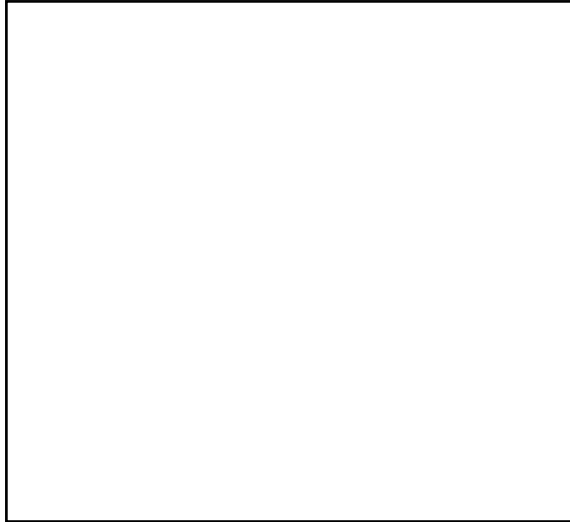


Figure 6. The evolution of finite memory strategies in the mean-field model is shown for 26000 generations, with the parameter values $T = 1.7$, $P = 0.4$. Again the coexistence between 01 (TFT) and 10 (ATFT) is present, but 1001 does not appear. For these parameter values, longer memories seem to be required to get cooperative and stable (or metastable) strategies. The simulation ends with a strategy that has found an unexploitable error correcting mechanism. The coexistence between the two memory three strategies 10010001 and 00011001 is an example of mutualism [Lindgren 1992].

Both the strategy 1001 and the final strategies in Figure 6 belong to a class of strategies that can deal with mistakes. These strategies are based on retaliation and synchronization. An accidental defection is followed by a certain number of rounds of mutual defection. The rounds of defection both punish possible exploiting strategies and synchronize the strategies before cooperative behaviour is established again. The strategies can be described by the simple mechanism:

After a defection by either of the players, defect until there has been n consecutive rounds of mutual defection, and then start cooperating again.

We shall denote such a strategy by S_n . The first, S_1 , is Simpleton [Rapoport and Chamah 1965] (or Pavlov [Nowak and Sigmund 1993]), a strategy that cooperates if both players chose the same action in the previous round, and this one is identical to 1001 in our notation. This strategy exhibits the following pattern in case of an accidental defection \underline{D} : (... CC, CC, \underline{DC} , DD, CC, CC, ...). For the standard parameters used by Axelrod [1984], i.e., $R = 3$, $S = 0$, $T = 5$, and $P = 1$, a population of Simpletons can be invaded, since ALLD (always defects) can exploit it — a single round of mutual defections is not enough to punish an exploiting strategy. The second strategy in this class is S_2 , which answers a defection (by either of the players) by defecting twice in a row, and after that returns to cooperation. This behaviour leads to the pattern (... CC, CC, \underline{DC} , DD, DD, CC, CC ...), which is the mechanism we find in the population of strategies that dominate at the end of the simulation in Figure 6.

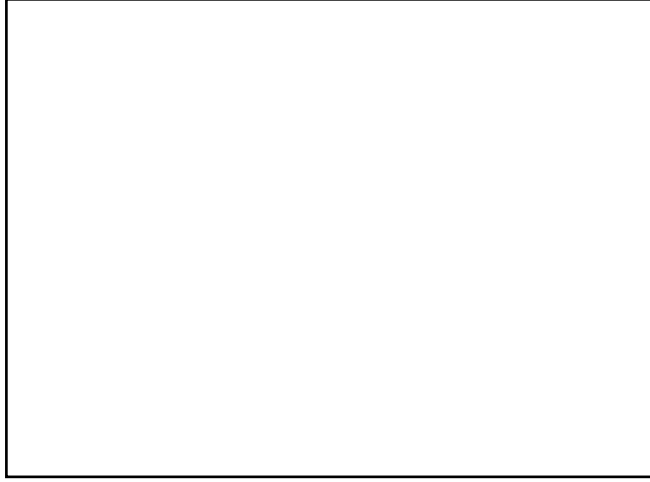


Figure 7. The parameter region, $1 < T < 2$ and $0 < P < 1$, is divided into subregions in which the smallest evolutionarily stable strategy of the S_n type is shown, according to the inequality requirement (4). The two parameter values corresponding to the simulations of Figures 5 and 6 are shown as crosses.

A population of S_n cannot be invaded by a strategy that tries to get back to cooperation faster than S_n does — the invador will only be exploited if it tries that. The only chance for a single invador (or mutant) to succeed is by exploiting the cooperative behaviour. Thus an invador will fail if the number n of defections after its exploiting defection is large enough to compensate for the temptation score T , i.e., $T + nP < (n + 1)R$, if terms of order e are neglected. This means that, in the limit of $e \rightarrow 0$, n must be chosen so that

$$n > \lfloor (T - R) / (R - P) \rfloor . \quad (3)$$

In other words, since $T > R > P$, there is always an error probability e and an integer k , for which the strategy S_k cannot be invaded. Note that $e > 0$ is needed, since it has been shown [Boyd and Lorberbaum 1987, Boyd 1989, Lorberbaum 1994] that the presence of mistakes is necessary to allow for evolutionary stability¹. With our choice of $T = 1$, the stability requirement for the strategy S_k , in the limit $e \rightarrow 0$, can be written

$$P < 1 + \lfloor (1; n) \rfloor - \lfloor (T; n) \rfloor . \quad (4)$$

These lines are drawn in Figure 7. The two examples discussed above (Figures 6 and 7) are on separate sides of the first line. Parameter choices towards the bottom left hand corner appear to make the situation easier for cooperative strategies, since the temptation to defect is relatively small. In, the upper right-hand corner, however, the

¹An evolutionarily stable strategy (ESS) cannot be invaded by any other strategy that initially is present at an arbitrarily small amount [Maynard Smith 1982].

score difference between cooperating and defecting is smaller, and a strategy that punishes defectors may need a longer memory to be able to avoid exploitation.

The CA model

The introduction of spatial dimensions, so that individuals only interact with those in their neighbourhood, may affect the dynamics of the system in various ways. It has been demonstrated, for example in the study by Boerlijst and Hoegeweg [1991, 1992], that parasitic species may be rejected by spiral waves consisting of several species. Other forms of cooperative behaviour are also possible, for example regions of cooperative strategies surrounded by exploiting cheaters. The possibility of spatio-temporal structures may allow for global stability where the mean-free model would be unstable. The presence of these various forms of spatio-temporal phenomena may therefore also alter the evolutionary path, compared with the mean-field case, and we may see other strategies evolve. A CA model based on the Prisoner's Dilemma was presented in [Axelrod 1984], and an investigation of the spatio-temporal dynamics of such a CA model as been done by Nowak and May [1992, 1993]. For a similar model, based on a local n -person Prisoner's Dilemma game, see [Matsuo 1995].

In [Lindgren and Nordahl 1994], we performed a detailed analysis of the dynamic behaviour of memory one strategies for the parameter region $1 < T < 2$ and $0 < P < 1$. Already at memory one, there are several spatio-temporal phenomena that may lead to global stability while the system is locally unstable. Spatial coexistence in frozen patterns is also possible. In Figure 8, four examples of spatial patterns from simulations of memory one strategies for different parameter choices are shown.

In Figure 9, the strategy abundance is shown for the first 800 generations of the CA model, with $T = 1.6$ and $P = 0.3$. (In the examples presented here we have used a 64×64 lattice, and the mutation rates are $p_{\text{point}} = 2 \cdot 10^{-3}$ and $p_{\text{dupl}} = p_{\text{split}} = 10^{-3}$.) It is clear that the strategy 1001 (or S_1) is successful, even though we are close to the line above which it can be exploited, cf. Figure 7. Before 1001 takes over, the system is dominated by 00 and 11, i.e., ALLD and ALLC (always cooperate). This is contrary to what we found in the mean field case, where 01 (TFT) and 10 (ATFT, does the opposite to TFT) dominate before memory two strategies appear. The reason is the possibility to form cooperating regions in the spatial model. Here we get islands of ALLC surrounded by ALLD, similar to the pattern in Figure 8 (d). The ALLC individuals at the edges of the islands are certainly exploited by the ALLD individuals, and in the reproduction step they are removed and replaced by the offspring of the most successful individual in their neighbourhood. Now, an ALLC individual that is not in contact with ALLD has the highest score and therefore the

edge individuals are replaced by an offspring of their own kind. This can be viewed as 'kin selection', in which the individual at the edge sacrifices herself since that is beneficial for her own strategy type.

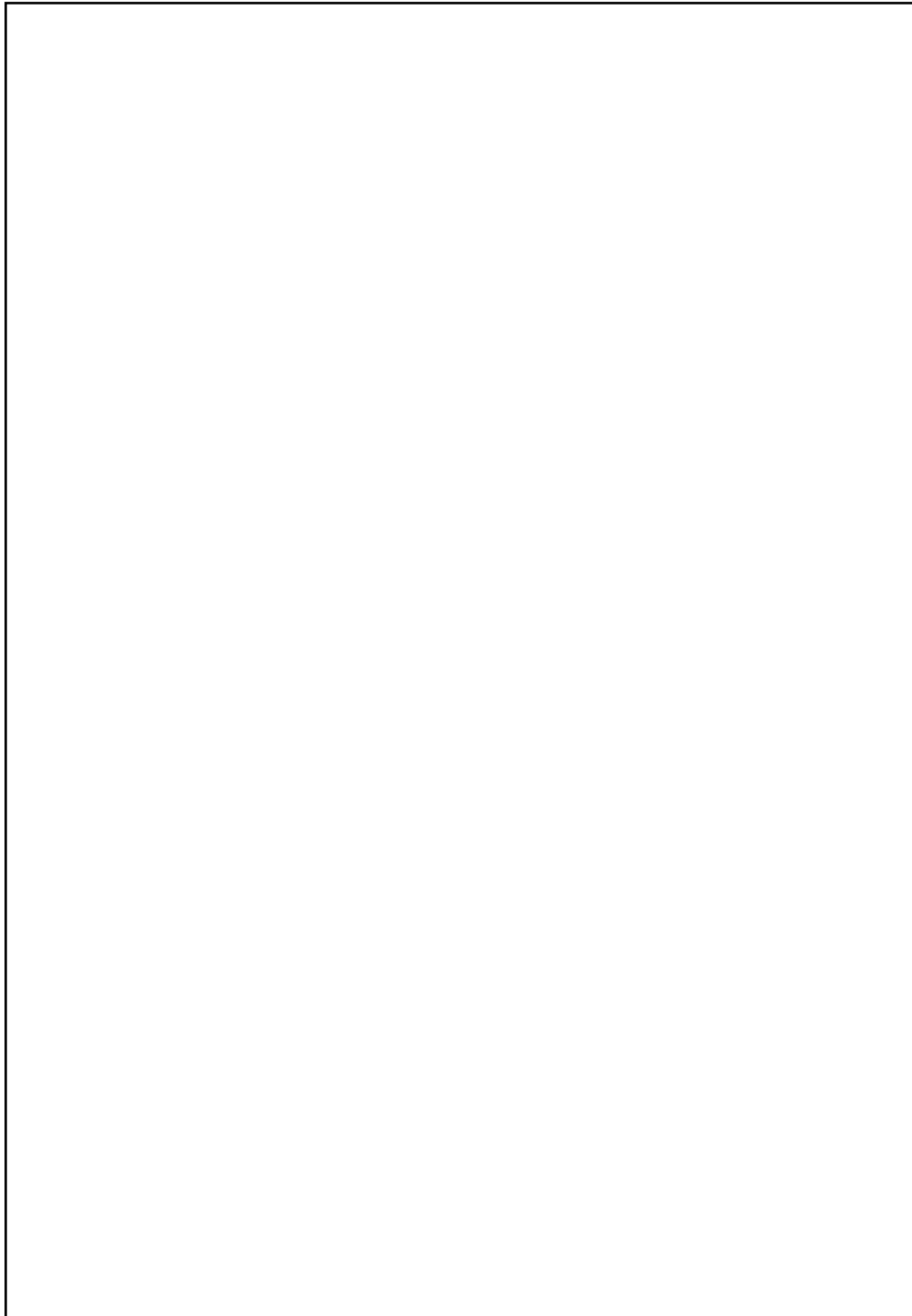


Figure 8. Examples of four different types of patterns from simulations of memory one strategies on the lattice. The payoffs are (a) $(T, P) = (1.4, 0.05)$, (b) $(T, P) = (1.9, 0.2)$, (b) $(T, P) = (1.9, 0.8)$, (b) $(T, P) = (1.4, 0.5)$. The strategies ALLD, TFT, ATFT, and ALLC are represented in grey-scale from white to black.



Figure 9. The evolution of finite memory strategies in the cellular automaton model is shown for 800 generations, with the parameter values $T = 1.6$, $P = 0.3$, cf. Figure 5. There is a meta-stable coexistence between memory one strategies has switched from 01 (TFT) and 10 (ATFT) to 00 (ALLD) and 11 (ALLC). The reason to this can be found in a spatial pattern of ALLC islands surrounded by ALLD, similar to what is seen in Figure 8 (d). Still the 1001 strategy dominates at the end.



Figure 10. The evolution of finite memory strategies in the cellular automaton model is shown for 2000 generations, with the parameter values $T = 1.7$, $P = 0.4$, cf. Figure 6. Initially, during the first 200 generations, there is a meta-stable coexistence between TFT, ALLC, and ALLD in the form of spiral waves, similar to the pattern in Figure 8 (b). The spatial dimension allows 1001 to dominate in a metastable state, due to the possibility to form cooperative regions. At the end the S_2 -like strategy from Figure 6 comes to dominate.

If we would choose a more 'rational' procedure for selecting the new strategy, e.g., by taking the strategy in the neighbourhood that maximizes the score in the particular cell in the next time step (generation), ALLD would invade the island and wipe out the ALLC strategy from the population. It would be interesting to investigate the

difference between 'rational' and 'natural' selection in this type of model, and to see how the evolutionary dynamics changes.

In Figure 10, the evolutionary path is shown for $T = 1.7$ and $P = 0.4$. Here, 1001 (S_1) should not be stable, according to (4), but instead strategies of type S_2 should be able to successfully deal with both mistakes and exploiting strategies. The simulation ends with one of the strategies using the S_2 pattern for error correction, i.e., two rounds of mutual defection after a mistake. There is a large number of strategies in this group, and longer simulations show that in the CA model there is usually a certain memory 5 strategy that takes over the whole population, since even very small score differences may lead to the total dominans of one of the strategies. In the CA model there are also examples of spatial coexistence between strategies of longer memory, that were not observed in the mean-field model. For some examples of these phenomena, see [Lindgren and Nordahl 1994].

Before 1001 dominates the system, in Figure 10, there is a period with three strategies coexisting. This is due to a spatio-temporal pattern of spiral waves involving the strategies TFT, ALLD, and ALLC, similar to Figure 8 (b). Again we see an example of a locally unstable composition of strategies that can be globally stable in the CA model.

Finite Automata strategies

As a final modification of the previous models, we shall look at a different representation of strategies for the game. A natural approach is to use finite automata, and this has been done before by Miller [1989] and by Stanley et al [1993], see also [Nowak et al 1995]. The novelty in the approach presented here, is the combination of an evolvable genome size, the analytically solved iterated Prisoner's Dilemma with mistakes, and the finite automata representation.

The reason why this representation could be interesting to investigate is that there are reasons to believe that more successful strategies may evolve. In fact, there is an extremely simple cooperative, uninvadeable and error-correcting strategy in this representation.

This strategy was suggested by Sugden [1986], as a robust strategy that can maintain cooperation in the presence of mistakes. We call the strategy "Fair", since the error-correcting mechanism is built on the idea of a fair distribution of the scores. The Fair strategy can be in one of three possible states: "satisfied" (cooperating), "apologizing" (cooperating), and "angry" (defecting). In the satisfied state, Fair cooperates, but if the

opponent defects Fair switches to the angry state and defects until the opponent cooperates, before returning to the satisfied state. If, on the other hand, Fair accidentally defects, the apologizing state is entered and Fair stays in this state until it succeeds in cooperating. Then it returns to the satisfied state again, regardless of the opponent's action. The strategy can be described as a finite automaton, see Figure 11.

This means that two players using the Fair strategy will quickly get back to cooperation after an accidental defection. The player making the mistake enters the apologizing state and the opponent the angry state, so that in the next round they will switch actions making the scores fair. Thus a sequence of rounds including an accidental defection D looks like (... CC, DC, CD, CC, CC, ...). (Only if the apologizing player fails to cooperate, for example by another mistake, they will both stay in the apologizing and angry state, respectively, until the apologizing player succeeds in a C action.)

Before we go on discussing the evolutionary simulations based on finite automata, we shall take a look at the game-theoretic properties of the Fair strategy. The stability properties of the Fair strategies were investigated by Sugden [1986], and the analysis was extended by Boyd [1989], who labeled the strategy "contrite-tit-for-tat". Wu and Axelrod [1995] showed the advantage of the Fair strategy in a population dynamics simulation, starting with a population mainly consisting of the 63 strategies submitted to the second tournament organized by Axelrod [1984].

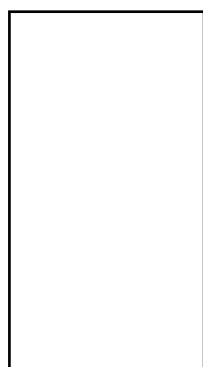


Figure 11. The Fair strategy can be represented as a finite automaton with three states. The symbols in the state circles denote the action that is intended, though a mistake may alter the action. After each round of the game, the performed actions are inspected and the strategy enters the state given by the appropriate transition. (On the transition arcs, the left and right symbols correspond to the action performed by the strategy itself and the opponent, respectively, where an X denotes an arbitrary action.) The anti-symmetry assures that when two Fair players meet they will either both be in the satisfied state, or they will be in the Apologizing and the Fair state, respectively, from where they simultaneously return to the satisfied state.

In the following analysis we are assuming the infinitely iterated PD, with a small probability e for mistakes, i.e., the performed action is not the intended one. In the game between two Fair strategies, the probability is $1/(1+2e)$ to be in the satisfied state and $e/(1+2e)$ for both the apologizing and the angry state². This leads to an average score of $R + 2e(S+T-2R)$, if terms of order e^2 or higher are neglected. (We are assuming small e , and in the following we shall make a first order analysis, omitting terms of order e^2 or higher.) For the standard parameter values used by Axelrod ($R=3$, $T=5$, $S=0$, and $P=1$) and $e = 0.01$, the score is 2.98. This is close to the score that two ALLC strategies get when they meet³, $R+e(S+T-2R)$, or, for the same parameters, 2.99.

For a small probability for mistakes (in the limit $e \rightarrow 0$), a new player entering a population of Fair players must adopt the same strategy not to be outcompeted: If Fair is in the satisfied state, a defect action (by the invader) scoring T must be compensated for by an apology scoring only S , before Fair enters the satisfied state again, or Fair will stay in the angry state giving a score of P . Since $S+T < 2R$ and $P < R$, the invader would have done better cooperating with Fair in the satisfied state. This also implies that when Fair expects an apology, the opponent had better to cooperate. In the apologizing state, Fair does not take any notice of the opponent's action, and therefore the opponent should defect⁴.

Representation

Each individual has a strategy represented by a finite automaton like the one in Figure 11. The size of the automaton is not fixed in the evolution, but nodes may be added by mutation increasing the number of internal states. Other mutations may alter the actions associated with the internal states, move transition arcs, and change start node.

The game between two FA strategies is a stochastic process on a finite automaton representing the game between the two strategies. This 'game' automaton has as its internal states the pairs of possible internal states for the two players, and the transitions are given by the transitions of the two strategies. The average score for the

²A detailed analysis will be presented elsewhere.

³Even if ALLC is more cooperative it cannot invade a population of Fair players, for small values of e , since the score received by an invading ALLC, $R+e(2S+T-3R)$, differs by $e(R-T) < 0$ compared to the score for Fair, if terms of order e^2 are neglected.

⁴The Fair strategy can only be invaded if the constants of the payoff matrix are close to the limits, e.g., the punishment score P for mutual defection is not much less than the reward for mutual cooperation R , $R \approx P$, or if the risk for mistake is high. Fixing $S = 0$, one finds that the mistake probability must be of the order $e \approx (1-P/R)/(4-3P/R)$ for ALLD (always defecting) to exploit the Fair strategy. The punishment score P must not be less than $0.99R$, for ALLD to invade with an e of 1%. For the standard parameter values, an error probability of $e \approx 2/9 \approx 22\%$ is required for ALLD to invade.

players is then calculated from probability distribution over the internal states of the game automaton.

Note that all finite memory strategies can be represented by FA, but that the FA representation results in a larger evolutionary search space. As in the finite memory models, we start with a population consisting of equal numbers of the memory one strategies, ALLD, ALLC, TFT, and ATFT, which can be represented as FA like in Figure 12.

The mutations we allow are the following. The *node mutation* changes the action associated with a certain node. The *transition mutation* changes the destination node of a transition. The *start mutation* changes start node of the FA. The *growth mutation* adds a node to the FA. This is done in a neutral way, by adding a copy of a node that has a transition to itself. After the mutation, this transition connects the original node with a new one that is identical to the original one, so that this mutation is neutral, see Figure 13. There is also a *delete mutation* that randomly removes a node of the FA. The mutation rate is p_{mut} per node for all mutations, except for the transition mutation that has a rate of $4p_{\text{mut}}$ per node, i.e., p_{mut} per transition. In the mean-field model we have used a rate of $p_{\text{mut}} = 10^{-5}$. In the lattice model, we have used the same mutation rate per individual for all types of mutations, $p_{\text{ind}} = 10^{-3}$.

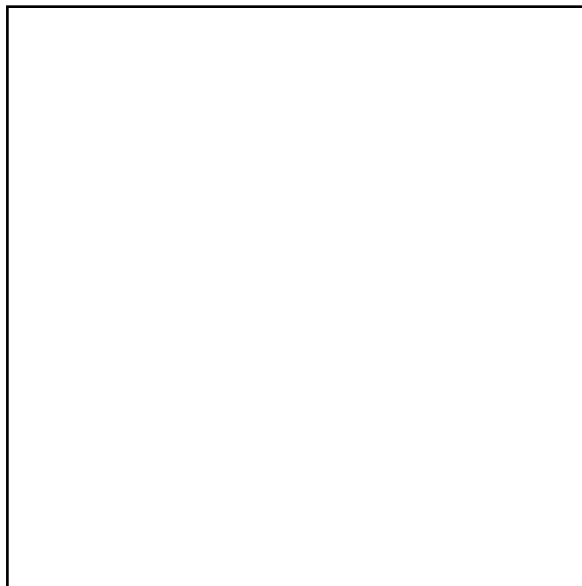


Figure 12. The initial state of the finite automata (FA) models consists of equal numbers of four FA that correspond to the memory one strategies: (a) ALLD, (b) ATFT, (c) TFT, and (d) ALLC.

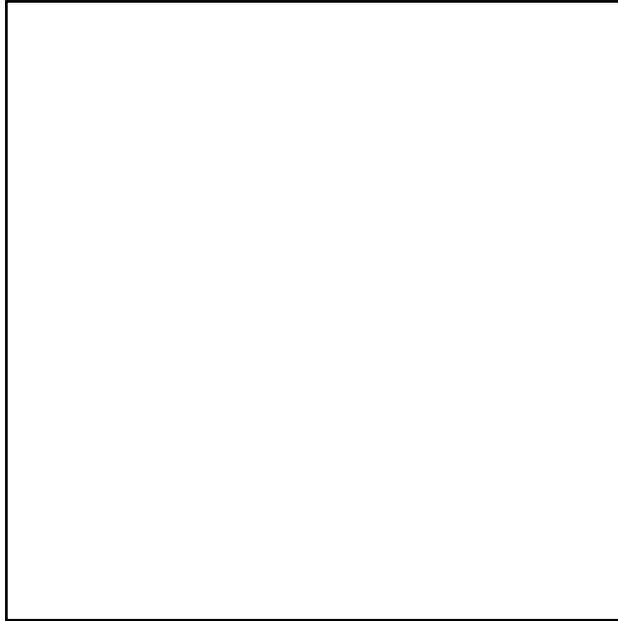


Figure 13. The growth mutation can only be applied if the parent FA has a node with a transition to itself. In that case the mutation creates a copy of the original node with transitions *from* the copied node identical to the original one. The self-transition of the original node is directed to the copy, while the self-transition of the copy remains. This results in a mutation that increases the number of internal states without changing the strategy — the mutation is neutral.

Mean-field model

Even if we start with the same strategies as in the finite memory representation, the evolutionary paths usually look very different since the evolutionary search space is dramatically increased and the points (in this space) corresponding to the finite memory strategies are not close to each other (in terms of mutations). It turns out that in the mean-field model with finite automata (FA) representation, the ALLD strategy more often takes over the whole system, compared with the finite memory representation. In the simulations shown below, we have chosen a smaller population size ($N = 500$) than in the previous case, which reduces the probability for ALLD to dominate.

In the first illustration of the evolution of FA, Figure 14, the payoff is $T = 1.6$ and $P = 0.3$. After the initial oscillations dominated by TFT, a strategy (not of finite memory type) that only differs from TFT by a transition mutation appears. This strategy is slightly better than TFT since it does not leave the cooperative state if both players defected last round, while in the defecting state it requires a cooperation from the opponent to get back to its cooperating state. The strategies that dominate the picture later on are similar to the finite memory strategy 1001, showing the same pattern for correcting for single mistakes. The slowly growing strategy at which the arrow is pointing in the bottom right-hand corner is the FA representation of 1001.

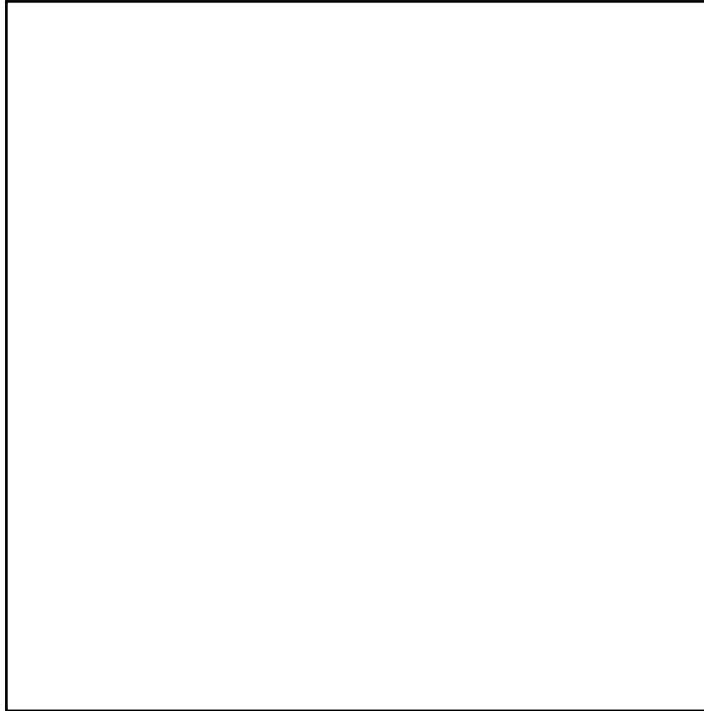


Figure 14. The evolution of finite automata (FA) is shown for 25000 generations in the mean-field model, with payoff parameters $(T, P) = (1.6, 0.3)$. Some of the automata that evolve are drawn on top of their abundance curves, respectively. The FA drawn at the bottom towards the end of the simulation is the FA representation of the finite memory strategy 1001. Also the dominating strategy at the the end has an error correcting mechanism similar to 1001.

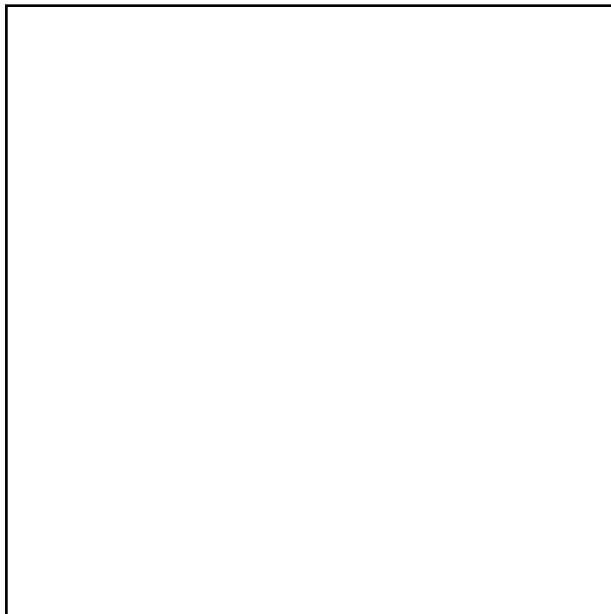


Figure 15. The evolution of finite automata (FA) is shown for 60000 generations in the mean-field model, with payoff parameters $(T, P) = (1.7, 0.4)$. Here, strategies corresponding to 1001 (S_1) should not be stable, cf. Figure 7. The automaton shown in the middle, is related to the Fair strategy in its mechanism for dealing with accidental defections, but it can be exploited. Instead an FA that defects twice after an isolated defect action dominates at the end, similar to the finite memory strategy S_2 .

In the second example, Figure 15, the payoff is $T = 1.7$ and $P = 0.4$, which means that the error correcting mechanism of 1001-like strategies can be exploited. The strategy shown in the middle of the figure, dominating for more than 10000 generations, resembles the Fair strategy, but it is too forgiving, which implies that it will be exploited. The strategy shown at the end, dominating for more than 15000 generations, has an error correction similar to S_2 , and it is possible (but not certain) that this simulation would end with a mixture of strategies of this type.

Lattice model

The spatial dimension again increases the chances for cooperative strategies to succeed in the population. Many simulations end with cooperative strategies like Fair dominating the whole lattice, for large parts of the parameter region. This is in accordance with the analysis of the Fair strategy above.

In the first example, Figure 16, we have the parameters $(T, P) = (1.7, 0.4)$ on a 64x64 lattice. After the initial oscillations we again see the S_1 -like strategy that dominated at the end of the mean-field simulation of Figure 14. This behaviour is not stable here, according to (4), and the simulation continues with a sequence of FA with three and four states, all having an S_2 -like mechanism for correcting for mistakes. There are many FAs that coincide with the pattern of two mutual defect actions after an isolated mistake, but that differ when it comes to second-order effects, i.e., when there is a mistake within the error-correcting pattern. These second-order effects may be sufficient to give a decisive advantage so that only one of them will dominate.

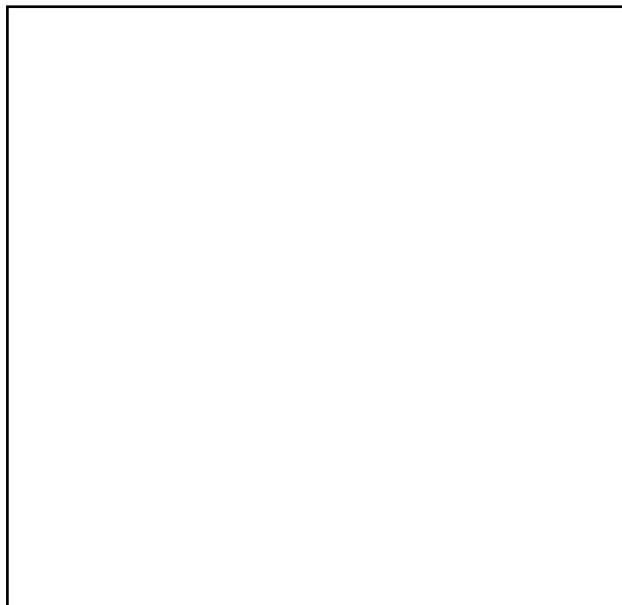


Figure 16. The evolution of finite automata for the cellular automaton model is shown for 1500 generations, with the parameter values $(T, P) = (1.7, 0.4)$. The automata of three and four states that evolve all show the S_2 -mechanism of the finite memory strategies for error correction.

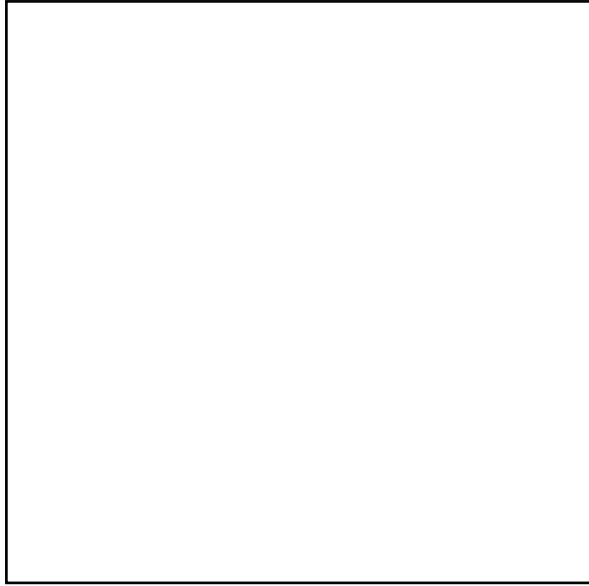


Figure 17. The evolution of finite automata for the cellular automaton model is shown for 1800 generations, with the parameter values $(T, P) = (1.9, 0.9)$. This is close to the upper right-hand corner of the payoff parameter region, see Figure 7. The chance that an S_n strategy would evolve here is small since the stability requirement (4) implies that n would be 10. In the simulation we see that a predecessor to Fair dominates for a long time until Fair appears, but at the end there is a 4-state variation of Fair that manages to take over.

In the second example, Figure 17, we are in the upper right-hand corner of the payoff matrix parameter region $(1.9, 0.9)$, see Figure 7. For these parameters, not even S_9 will be stable, but, according to (4), ten rounds of mutual defection are required to punish exploiting strategies. The simulation shows a meta-stable state dominated by a pair of strategies of which one is a three-state FA that resembles the Fair strategy. The Fair strategy is two transition mutations away, and the intermediate FA can be seen as one of the two narrow peaks before Fair appears, cf. Figure 11. However, there is a four-state variation of Fair that succeeds in outcompeting the original Fair. The difference lies in a fourth order effect, i.e., involving three mistakes in the apology pattern following one mistake. This strategy only differs from Fair by a (neutral) growth mutation and a transition mutation.

Other simulations from various parts of the parameter region of the payoff matrix in Figure 7, shows that Fair is successful and not very sensitive to the parameter choices. In most cases, the simulations end with one strategy dominating the whole lattice, a result that is not general in the CA simulations of the finite memory strategies. From the evolutionary point of view, these simulations may seem less interesting in the sense that the evolutionary transients appear to be short. On the other hand, they demonstrate the important difference between the spatial and the non-spatial models, as well as between different evolutionary search spaces, and that cooperative

behaviour may not be given the chance unless there is some compartmentalization possible, here in the form of spatial locality.

The examples presented here are of course preliminary, and a more detailed study of the evolution of finite automata strategies will be reported elsewhere.

Misunderstanding vs mistakes

As a possibility for future modifications of these models, one could consider a different kind of noise. Mistakes are a relatively simple form of noise to deal with in games, since both players have the same history of actions to look at. Misunderstanding on the other hand, leave the two players with different histories, and one cannot be sure whether the distribution of scores is fair or not even for a few rounds. This is of course critical for strategies like Fair, since a misunderstanding may lead them into situations where one player is in a satisfied state while the other one is in an angry state, which may result in sequences of alternating defect and cooperate actions, like for the TFT strategy.

The class of evolutionarily stable strategies (S_n) in the case of mistakes, discussed before, turn out to be error correcting also in the case of misunderstanding, at the same time as there is a possibility of avoiding both exploiting strategies and strategies that are more cooperative.

In the case of *misunderstanding*, two players using, e.g., the S_2 strategy, will show the same pattern as in the case of mistake, (... CC, CC*, DC, DD, DD, CC, CC, CC ...), where C^* denotes a C action that is perceived as a D action by the opponent. As before, the length of the 'quarrel' must be large enough to prevent potential exploiters to defect by purpose, which leads to the inequality requirement of (3).

But when misunderstandings are involved, there is another possibility to invade a population of S_n strategies, which leads to an additional requirement for the choice of n . If a player ignores a perceived D action from its opponent and still cooperates the score may increase, since the action could have been C and then the cost of quarrel could have been avoided, e.g. (... CC, CC*, CC, ...). On the other hand, if the action was truly D, then the opponent will defect again, and the return to cooperation will be delayed. For example, ignoring a defect action from S_2 leads to the pattern (... C*C, CD, CD, DD, DD, CC, CC, ...), where D is a defect action due to S_2 misunderstanding the previous action C^* . Since the risk for misunderstanding is equal for both players, the two patterns above, that can result from ignoring a perceived D action, occur with equal probability. The expected score for ignoring instead of

answering directly is $\sigma_0 = (n+2)R/2 + (2S+nP)/2$ to compare with the score for S_n , $\sigma_n = (S+T)/2 + nP + R$, for the rounds following a defect action. Thus, for S_n to avoid invasion, the inequality $\sigma_n > \sigma_0$ must be fulfilled. This can be rewritten as a requirement on n , $n < (T-S)/(R-P)$, or $n < (T-R)/(R-P) + 1 + (P-S)/(R-P)$. Since the third term is positive, there is always an integer n satisfying both conditions

$$\lfloor (T-R)/(R-P) \rfloor < n < \lfloor (T-R)/(R-P) \rfloor + 1 + \lfloor (P-S)/(R-P) \rfloor . \quad (5)$$

For the parameters used by Axelrod ($R = 1, S = 0, T = 5, P = 1$), we get $n = 2$, and the score is 2.92. Contrary to Fair, the error correction of these strategies work for both mistake and misunderstanding, but at a cost of a lower average score.

Discussion

By showing examples from various models based on the Prisoner's Dilemma as the interaction between individuals, I have illustrated some aspects of evolutionary models. The differences between the mean-field models and the lattice models are evident in all these cases. The possibility of spatio-temporal phenomena may give rise to a stable coexistence between strategies that would otherwise be outcompeted. These spatio-temporal structures may take the form of spiral waves, irregular waves, spatio-temporal chaos, frozen patchy patterns, and various geometrical configurations [Lindgren and Nordahl 1994]. This is very clear in the first model (finite game) where ALLD is the only fixed point in the mean-field model, but where ALLD seldom appears in the simulations of the CA model. Space also allows for increased cooperative behaviour by a mechanism that can be described as kin selection. For example, regions of ALLC can be stable in an ALLD environment, since the ALLC individuals at the border are replaced (in the next generation) by more successful ALLC individuals from the interior of the region.

The choice of representation is important as the difference between the simulations of the finite memory and FA strategies illustrate. Still, the existence of a stable fixed point, e.g., in the form of the Fair strategy, does not necessarily mean that the evolutionary dynamics always finds it. Of course, the regions of attraction of the fixed points are important to characterize the evolutionary search space. This may, however, be a very difficult task since the space of strategies is potentially infinite. The final example also illustrates that the region of attraction may increase dramatically if the spatial dimension is added. The increased probability for the Fair strategy to succeed probably depends on the advantage for cooperative strategies in

the spatial world. The Fair strategy may more easily survive in a local situation from where it may then spread.

The models presented here have all in common that they are very simple and can be easily simulated. Despite their simplicity, a number of complex phenomena in the form of evolutionary dynamics and spatio-temporal patterns can be observed. Models similar to these (in terms of simplicity) could be constructed in a more economic context to study economic phenomena like, e.g., trading and price formation. Some of the ingredients of the models presented here should be considered in that work, for example the advantage of starting small in combination with neutral mutations. This has been used in another context, where evolving artificial neural networks were selected for recognizing strings of a regular language [Lindgren et al, 1992].

Acknowledgement

This research is supported by the Swedish Natural Science Research Council.

References

- Axelrod, R. (1984). *The Evolution of Cooperation*, (Basic Books, New York).
- Axelrod, R. (1987). "The evolution of strategies in the iterated Prisoner's Dilemma," pp. 32-41 in L. Davis (Ed.) *Genetic Algorithms and Simulated Annealing* (Morgan Kaufmann, Los Altos, CA).
- Axelrod, R., and Dion, D. (1988). "The further evolution of cooperation," *Science* **242**, 1385-1390.
- Batali, J., and Kitcher, P. (1995). "Evolution of altruism in optional and compulsory games," *Journal of Theoretical Biology* **175**, 161-171.
- Boerlijst, M. C., and Hogeweg, P. (1991). "Spiral wave structure in prebiotic evolution: hypercycles stable against parasites," *Physica D* **48**, 17-28.
- Boerlijst, M. C., and Hogeweg, P. (1992). "Self-structuring and selection: spiral waves as a substrate for prebiotic evolution," pp. 255-276 in C. G. Langton et al (Eds.) *Artificial Life II* (Addison-Wesley, Redwood City, CA).
- Boyd, R. (1989). "Mistakes allow evolutionary stability in the repeated Prisoner's Dilemma game," *Journal of Theoretical Biology* **136**, 47-56.
- Boyd, R., and Lorberbaum, J. P. (1987). "No pure strategy is evolutionarily stable in the repeated Prisoner's Dilemma game," *Nature* **327**, 58-59.
- Ikegami, T. (1994). "From genetic evolution to emergence of game strategies," *Physica D* **75**, 310-327.
- Lindgren, K. (1992). "Evolutionary phenomena in simple dynamics," pp. 295-311 in C. G. Langton et al (Eds.) *Artificial Life II*, (Addison-Wesley, Redwood City, CA).

- Lindgren, K. (1995) "Evolutionary dynamics in game-theoretic models", Institute of Physical Resource Theory, preprint 95-17. Talk presented at the workshop "The economy as an evolving complex system II" at the Santa Fe Institute, NM, USA, August, 1995.
- Lindgren, K., Nilsson, A., Nordahl, M. G., and Råde, I. (1992). "Regular language inference using evolving recurrent neural networks," pp. 75-86 in *COGANN-92 International Workshop on Combinations of Genetic Algorithms and Neural Networks*, Whitley, L. D., and Schaffer, J. D. (eds.) (IEEE Computer Society Press, Los Alamitos, CA).
- Lindgren, K., and Nordahl, M.G. (1993). "Artificial food webs," pp. 73-103 in C. G. Langton (Ed.) *Artificial Life III*, (Addison-Wesley, Redwood City, CA).
- Lindgren, K., and Nordahl, M.G. (1994a). "Evolutionary dynamics of spatial games," *Physica D* **75**, 292-309.
- Lindgren, K., and Nordahl, M.G. (1994b). "Cooperation and community structure in artificial ecosystems," *Artificial Life Journal* **1**, 15-37.
- Lorberbaum, J. (1994). *J. Theor. Biol.* **168**, 117-130.
- Matsuo, K. (1985). "Ecological characteristics of strategic groups in 'dilemmatic world'," pp. 1071-1075 in *Proceedings of the IEEE International Conference on Systems and Cybernetics*.
- Maynard-Smith, J. (1982). *Evolution and the Theory of Games*, (Cambridge University Press, Cambridge).
- Miller, J. H. (1989). "The coevolution of automata in the repeated iterated Prisoner's Dilemma," (Santa Fe Institute working paper 89-003).
- Molander, P. (1985). "The optimal level of generosity in a selfish, uncertain environment," *Journal of Conflict Resolution* **29**, 611-618.
- Nowak, M. A., and May, R. M. (1992). "Evolutionary games and spatial chaos," *Nature* **359**, 826-829.
- Nowak, M. A., and May, R. M. (1993). "The spatial dilemmas of evolution," *International Journal of Bifurcation and Chaos* **3**, 35-78.
- Nowak, M. A., and Sigmund, K. (1992). "Tit-for-tat in heterogenous populations," *Nature* **355**, 250-253.
- Nowak, M. A., and Sigmund, K. (1993). "A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game," *Nature* **364**, 56-58.
- Nowak, M. A., Sigmund, K., and El-Sedy, E. (1995). "Automata, repeated games and noise," *Journal of Mathematical Biology* **33**, 703-722.
- Rapoport, A., and Chammah, A. M. (1965). *Prisoner's Dilemma*, (University of Michigan Press, Ann Arbor).
- Stanley, E. A., Ashlock, D., and Tesfatsion, L. (1993). "Iterated Prisoner's Dilemma with choice and refusal of partners," pp. 131-175 in C. G. Langton (Ed.) *Artificial Life III* (Addison-Wesley, Reading, MA).
- Sugden, R. (1986). *The Economics of Rights, Co-operation and Welfare* (Basil Blackwell, Oxford).
- Wu, J., and Axelrod, R. (1995). "How to cope with noise in the iterated Prisoner's Dilemma," *Journal of Conflict Resolution* **39**, 183-189.

