

An Evolutionary Game Theoretic perspective on Learning in Multi-Agent Systems

Karl Tuyls ^{*}, Ann Nowe, Tom Lenaerts and Bernard Manderick

Computational Modeling Lab, Department of Computer Science, Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussel

Abstract. In this paper we revise Reinforcement Learning and adaptiveness in Multi-Agent Systems from an Evolutionary Game Theoretic perspective. More precisely we show there is a triangular relation between the fields of Multi-Agent Systems, Reinforcement Learning and Evolutionary Game Theory. We illustrate how these new insights can contribute to a better understanding of learning in MAS and to new improved learning algorithms. All three fields are introduced in a self-contained manner. Each relation is discussed in detail with the necessary background information to understand it, along with major references to relevant work.

Keywords: Evolutionary Game Theory, Multi-Agent Systems, Reinforcement Learning

1. Introduction

Agent-Based computing is a new evolving paradigm in computer science. Nowadays, more and more technological challenges require distributed, dynamic systems. Traditional program paradigms make some assumptions which just do not hold for a great number of new applications. Often the environment for which a program is designed is neither static, nor completely known or deterministic (e.g. the internet). The characteristics of these environments imply that systems which need to interact in such an environment operate within rapidly changing circumstances, with an enormous growth of available information. These new requirements of today's applications suggest that alternative programming paradigms are necessary.

Since the early 90's agent-based systems or Multi-Agent Systems have emerged as an important active area of research to support these new requirements in Information Technology (Wooldridge, M., 2002; Luck et al., 2003; Weiss, G., 1999). In contrast with traditional methodologies the agent-based approach views a program as a set of one or more independent, rational agents. Typically, an agent is an autonomous computational entity with a flexible dynamic behaviour in an

^{*} Author funded by a doctoral grant of the institute for advancement of scientific technological research in Flanders (IWT)



unpredictable environment. The uncertainty of the environment implies that an agent needs to learn from, and adapt to, its environment to be successful. Indeed, it is impossible to foresee all situations an agent can encounter beforehand. Therefore, **learning** and **adaptiveness** become crucial for the successful application of Multi-agent systems to contemporary technological challenges. Robocup is a nice illustration of such a challenge. The global goal of the Robocup project is stated as, *by the year 2050, develop a team of fully autonomous humanoid robots that can win against the human world soccer champion team* (Robocup project, 2003). A team of robotic players, challenging human soccer teams, must be capable of learning how to communicate, cooperate and compete. If this team of robotic players, a standard multi-agent system, wants to be competitive, it must be able to coordinate their actions as a team. Hence learning and adaptiveness become crucial.

Reinforcement Learning (RL) is already an established and profound theoretical framework for learning in stand-alone or single-agent systems. Yet, extending RL to multi-agent systems (MAS) does not guarantee the same theoretical grounding. As long as the environment an agent is experiencing is Markov¹, and the agent can experiment enough, RL guarantees convergence to the optimal strategy. In a MAS however, the reinforcement an agent receives, may depend on the actions taken by the other agents present in the system. Hence, the markovian property no longer holds. Moreover, previous guarantees of convergence disappear.

Consider for instance the problem of finding the optimal way between two points in traffic. The cost measured in time it takes to get from point A to a point B using a particular route will be influenced by the current traffic conditions, i.e. how many other drivers decided to use the same route. Communication on these decisions is not always possible, moreover there is an associated cost and communication is subject to delays. Uncontrolled exploration in this situation can lead to policy oscillations, (Nowe et al., 1999). When everyone decides to take the alternative route, this one becomes less interesting than the original one. Most MAS belong to this last case of non-stationarity. Obviously in these environments, the convergence results of RL are lost.

¹ The Markov property states that only the present state gives any information of the future behaviour of the learning process. Knowledge of the history of the process does not add any new information.

In the light of the above problem it is important to fully understand the dynamics of reinforcement learning and the effect of exploration in MAS. For this aim we review Evolutionary Game Theory (EGT) as a solid basis for understanding learning and constructing new learning algorithms. The Replicator Equations will appear to be an interesting model to study learning in various settings. This model consists of a system of differential equations describing how a population of strategies evolves over time, and plays a central role in biological and economical models. Several authors have already noticed that the Replicator Dynamics (RD) can emerge from simple learning models, (Sarin et al., 1997; Redondo, F.V., 2001; Tuyls et al., June'03).

This article discusses the theoretical foundations of learning in multi-agent systems. For the moment, a theoretical framework in which learning and adaptiveness in agent-based systems can be understood profoundly is lacking. However, this paper reveals how Evolutionary Game Theory and Reinforcement Learning are connected and how insights from Evolutionary Game Theory provide a better understanding of learning in general in multi-agent systems. More precisely, this formal relation closes the triangle between the three fields and offers the necessary foundations for this missing formal framework. This comes down to an important triangular relation between the field of MAS, Reinforcement Learning and Evolutionary Game Theory expressed by figure 1. Each relation of this triangle will be discussed in detail in this paper.

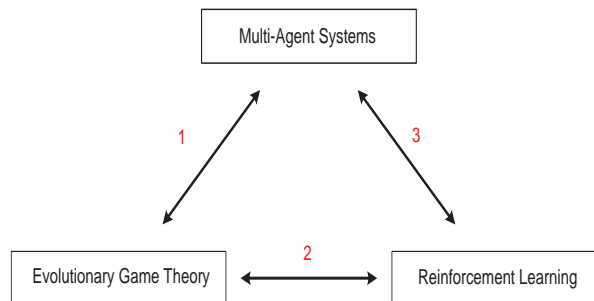


Figure 1. The triangular relation between RL, MAS and EGT.

The outline of the paper is as follows, in the second section we present an overview of the key concepts and key results from Game Theory (GT) and Evolutionary Game Theory. This is a necessary background for the further discussion and provides an evolutionary game theoretic perspective on learning in MAS. After this we continue with discussing the three relations in more detail. Section 3 discusses

the third link of figure 1 and makes explicit how both fields relate and what the current issues are in multi-agent learning algorithms. Section 4 discusses the second link of figure 1 and reveals how both fields relate mathematically and is an opening toward solving the issues of the previous section. Section 5 closes the circle and elaborates on the first link, i.e. reveals the interesting similarities between the fields and summarizes why EGT is an interesting framework to analyze and understand MAS. We end with a conclusion.

2. Evolutionary Game Theory

2.1. INTRODUCTION

When John Nash discovered the theory of games at Princeton, in the late 40's and early 50's, the impact was enormous. Originally, Game Theory was launched by John von Neumann and Oskar Morgenstern in 1944 in their book *Theory of Games and Economic Behavior* (von Neumann et al., 1944). The impact of the developments in Game Theory expressed itself especially in the field of economics, where its concepts played an important role in for instance the study of international trade, bargaining, the economics of information and the organization of corporations. But also in other disciplines in the social and natural sciences the importance of Game Theory became clear, as for instance studies of legislative institutions, of voting behavior, of warfare, of international conflicts, and of evolutionary biology.

However, von Neumann and Morgenstern had only managed to define an equilibrium concept for 2-person zero-sum games. Zero-sum games correspond to situations of pure competition, whatever one player wins, must be lost by another. John Nash addressed the case of competition with mutual gain by defining best-reply functions and using Kakutani's fixed point-theorem. The main results of his work expressed themselves in his development of the *Nash Equilibrium* and the *Nash Bargaining Solution* concept.

Despite the great usefulness of the Nash equilibrium concept, the assumptions traditional game theory make, like hyperrational players that correctly anticipate the other players in an equilibrium, made game theory stagnate for quite some time (Weibull, J.W., 1996; Gintis, 2000; Samuelson, L., 1997). A lot of refinements of Nash equilibria came along (for instance trembling hand perfection), which made it hard to choose the appropriate equilibrium in a particular situation. Almost

any Nash equilibrium could be justified in terms of some particular refinement. This made clear that the static Nash concept did not reflect the (dynamic) real world where people do not act hyperrational. This is where evolutionary game theory originated. More precisely, Maynard Smith adopted the idea of evolution from biology (Maynard-Smith et al., 1973; Maynard-Smith, J., 1982). This idea led Smith and Price to the concept of Evolutionary Stable Strategies (ESS), which in fact obeys a stricter condition than the Nash condition. In evolutionary game theory the game is no longer played exactly once by rational players who know all the details of the game. Details of the game include each others preferences over outcomes. Instead EGT assumes that the game is played repeatedly by players randomly drawn from large populations, uninformed of the preferences of the opponent players.

Evolutionary Game Theory offers a solid basis for rational decision making in an uncertain world, it describes how individuals make decisions and interact in complex environments in the real world. Modeling learning agents in the context of Multi-agent Systems requires insight in the type and form of interactions with the environment and other agents in the system. Usually, these agents are modelled similar to the different players in a standard game theoretical model. In other words, these agents assume complete knowledge of the environment, have the ability to correctly anticipate the opposing player (hyperrationality) and know that the optimal strategy in the environment is always the same (static Nash equilibrium). The intuition that in the real world people are not completely knowledgeable and hyperrational players and that an equilibrium can change dynamically led to the development of evolutionary game theory.

2.2. ELEMENTARY CONCEPTS

In this section we review the key concepts of EGT and its mutual relationships. This is important to understand the further discussion in later sections. We start by defining strategic games and concepts as Nash equilibrium, Pareto optimality and evolutionary stable strategies. Then we discuss the relationships between these concepts and provide some examples.

2.2.1. *Strategic games*

In this section we define n-player normal form games as a conflict situation involving gains and losses between n players. In such a game n players interact with each other by all choosing an action (or strategy) to play. All players choose their strategy at the same time. For

reasons of simplicity, we limit the pure strategy set of the players to 2 strategies. A strategy is defined as a probability distribution over all possible actions. In the 2-pure strategies case, we have: $s_1 = (1, 0)$ and $s_2 = (0, 1)$. A mixed strategy s_m is then defined by $s_m = (x_1, x_2)$ with $x_1, x_2 \neq 0$ and $x_1 + x_2 = 1$.

Defining a game more formally we restrict ourselves to the 2-player 2-action game. Nevertheless, an extension to n-players n-actions games is straightforward, but examples in the n-player case do not show the same illustrative strength as in the 2-player case. A game $G = (S_1, S_2, P_1, P_2)$ is defined by the payoff functions P_1, P_2 and their strategy sets S_1 for the first player and S_2 for the second player. In the 2-player 2-strategies case, the payoff functions $P_1 : S_1 \times S_2 \rightarrow \mathfrak{R}$ and $P_2 : S_1 \times S_2 \rightarrow \mathfrak{R}$ are defined by the payoff matrices, A for the first player and B for the second player, see Table I. The payoff tables A, B define the instantaneous rewards. Element a_{ij} is the reward the row-player (player 1) receives for choosing pure strategy s_i from set S_1 when the column-player (player 2) chooses the pure strategy s_j from set S_2 . Element b_{ij} is the reward for the column-player for choosing the pure strategy s_j from set S_2 when the row-player chooses pure strategy s_i from set S_1 .

The family of 2×2 games is usually classified in three subclasses, as follows (Redondo, F.V., 2001),

Table I. The left matrix (A) defines the payoff for the row player, the right matrix (B) defines the payoff for the column player

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \quad B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$$

Subclass 1: if $(a_{11} - a_{21})(a_{12} - a_{22}) > 0$ or $(b_{11} - b_{12})(b_{21} - b_{22}) > 0$, at least one of the 2 players has a dominant strategy, therefore there is just 1 strict equilibrium.

Subclass 2: if $(a_{11} - a_{21})(a_{12} - a_{22}) < 0, (b_{11} - b_{12})(b_{21} - b_{22}) < 0$, and $(a_{11} - a_{21})(b_{11} - b_{12}) > 0$, there are 2 pure equilibria and 1 mixed equilibrium.

Subclass 3: if $(a_{11} - a_{21})(a_{12} - a_{22}) < 0, (b_{11} - b_{12})(b_{21} - b_{22}) < 0$, and $(a_{11} - a_{21})(b_{11} - b_{12}) < 0$, there is just 1 mixed equilibrium.

The first subclass includes those type of games where each player has a dominant strategy², as for instance the prisoner's dilemma. However it includes a larger collection of games since only one of the players needs to have a dominant strategy. In the second subclass none of the players has a dominated strategy (e.g. battle of the sexes). But both players receive the highest payoff by both playing their first or second strategy. This is expressed in the condition $(a_{11} - a_{21})(b_{11} - b_{12}) > 0$. The third subclass only differs from the second in the fact that the players do not receive their highest payoff by both playing the first or the second strategy (e.g. matching pennies game). This is expressed by the condition $(a_{11} - a_{21})(b_{11} - b_{12}) < 0$. Section 2.2.6 provides an example of each subclass.

2.2.2. Nash equilibrium

In traditional game theory it is assumed that the players are hyperrational, meaning that every player will choose the action that is best for him, given his beliefs about the other players' actions. A basic definition of a Nash equilibrium is stated as follows. If there is a set of strategies for a game with the property that no player can increase its payoff by changing his strategy while the other players keep their strategies unchanged, then that set of strategies and the corresponding payoffs constitute a Nash equilibrium.

Formally, a Nash equilibrium is defined as follows. When 2 players play the strategy profile $s = (s_i, s_j)$ belonging to the product set $S_1 \times S_2$ then s is a Nash equilibrium if $P_1(s_i, s_j) \geq P_1(s_x, s_j) \forall x \in \{1, \dots, n\}$ and $P_2(s_i, s_j) \geq P_2(s_i, s_x) \forall x \in \{1, \dots, m\}$.

In section 2.2.6 some examples illustrate this definition.

2.2.3. Pareto optimality

Intuitively a Pareto optimal solution of a game can be defined as follows: a combination of actions of agents in a game is Pareto optimal if there is no other solution for which all players do at least as well and at least one agent is strictly better off.

More formally we have: a strategy combination $s = (s_1, \dots, s_n)$ for n agents in a game is Pareto optimal if there does not exist another strategy combination s' for which each player receives at least the same payoff P_i and at least one player j receives a strictly higher payoff than P_j .

² A strategy is dominant if it is always better than any other strategy, regardless of what the opponent may do.

2.2.4. *Evolutionary Stable Strategies*

The core equilibrium concept of Evolutionary Game Theory is that of an Evolutionary Stable Strategy (ESS). The idea of an evolutionarily stable strategy was introduced by John Maynard Smith and Price in 1973 (Maynard-Smith et al., 1973). Imagine a population of agents playing the same strategy. Assume that this population is invaded by a different strategy, which is initially played by a small number of the total population. If the reproductive success of the new strategy is smaller than the original one, it will not overrule the original strategy and will eventually disappear. In this case we say that the strategy is evolutionary stable against this new appearing strategy. More general, we say a strategy is an Evolutionary Stable strategy if it is robust against evolutionary pressure from any appearing mutant strategy.

Formally an ESS is defined as follows. Suppose that a large population of agents is programmed to play the (mixed) strategy s , and suppose that this population is invaded by a small number of agents playing strategy s' . The population share of agents playing this mutant strategy is $\epsilon \in]0, 1[$. When an individual is playing the game against a random chosen agent, chances that he is playing against a mutant are ϵ and against a non-mutant are $1 - \epsilon$. The payoff for the first player, being a non mutant is:

$$P(s, (1 - \epsilon)s + \epsilon s')$$

and being a mutant is,

$$P(s', (1 - \epsilon)s + \epsilon s')$$

Now we can state that a strategy s is an ESS if $\forall s' \neq s$ there exists some $\delta \in]0, 1[$ such that $\forall \epsilon : 0 < \epsilon < \delta$,

$$P(s, (1 - \epsilon)s + \epsilon s') > P(s', (1 - \epsilon)s + \epsilon s')$$

holds. The condition $\forall \epsilon : 0 < \epsilon < \delta$ expresses that the share of mutants needs to be sufficiently small.

2.2.5. *The relation between Nash equilibria and ESS*

This section explains how the core equilibria concepts from classical and evolutionary game theory relate to one another. The set of Evolutionary Stable Strategies for a particular game are contained in the set of Nash Equilibria for that same game,

$$\{ESS\} \subset \{NE\}$$

The conditions for an ESS are stricter than the Nash condition. Intuitively this can be understood as follows: as defined above a Nash equilibrium is a best reply against the strategies of the other players. Now if a strategy s_1 is an ESS then it is also a best reply against itself, or optimal. If it wasn't optimal against itself there would have been a strategy s_2 that would lead to a higher payoff against s_1 than s_1 itself. So, if the population share ϵ of mutant strategies s_2 is small enough then s_1 is not evolutionary stable because,

$$P(s_2, (1 - \epsilon)s_1 + \epsilon s_2) > P(s_1, (1 - \epsilon)s_1 + \epsilon s_2)$$

An important second property for an ESS is the following. If s_1 is ESS and s_2 is an alternative best reply to s_1 , then s_1 has to be a better reply to s_2 than s_2 to itself. This can easily be seen as follows, because s_1 is ESS, we have for all s_2

$$P(s_1, (1 - \epsilon)s_1 + \epsilon s_2) > P(s_2, (1 - \epsilon)s_1 + \epsilon s_2)$$

If s_2 does as well against itself as s_1 does, then s_2 earns at least as much against $(1 - \epsilon)s_1 + \epsilon s_2$ as s_1 and then s_1 is no longer evolutionary stable. To summarize we now have the following 2 properties for an ESS s_1 ,

1. $P(s_2, s_1) \leq P(s_1, s_1) \quad \forall s_2$
2. $P(s_2, s_1) = P(s_1, s_1) \implies P(s_2, s_2) < P(s_1, s_2) \quad \forall s_2 \neq s_1$

2.2.6. Examples

In this section we provide some examples of the classification of games (see section 2.2.1) and illustrate the Nash equilibrium concept and Evolutionary Stable Strategy concept as well as Pareto optimality.

For the first subclass we consider the prisoner's dilemma game (Gintis, 2000; Weibull, J.W., 1996). In this game 2 prisoners, who committed a crime together, have a choice to either cooperate with the police (to defect) or work together and deny everything (to cooperate). If the first criminal (row player) defects and the second one cooperates, the first one gets off the hook (expressed by a maximum reward of 5) and the second one gets the most severe punishment. If they both defect, they get the second most severe punishment one can get (expressed by a payoff of 1). If both cooperate, they both get a minimum sentence.

The payoffs of the game are defined in table II. As one can see both players have one dominant strategy, more precisely *defect*. For both

Table II. Prisoner's dilemma: The left matrix (A) defines the payoff for the row player, the right one (B) for the column player.

$$A = \begin{pmatrix} 1 & 5 \\ 0 & 3 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 5 & 3 \end{pmatrix}$$

players, defecting is the dominant strategy and therefore always the best reply toward any strategy of the opponent. So the Nash equilibrium in this game is for both players to defect. Let's now determine whether this equilibrium is also an evolutionary stable strategy. Suppose $\epsilon \in [0, 1]$ is the number of cooperators in the population. The expected payoff of a cooperator is $3\epsilon + (1 - \epsilon)$ and that of a defector is $5\epsilon + (1 - \epsilon)$. Since for all ϵ ,

$$5\epsilon + 1(1 - \epsilon) > 3\epsilon + 0(1 - \epsilon)$$

defect is an ESS. So the number of defectors will always increase and the population will eventually only consist of defectors. In section 2.3 this dynamical process will be illustrated by the replicator equations. This equilibrium which is both Nash and ESS, is not a Pareto optimal solution. This can be easily seen if we look at the payoff tables. The combination (*defect, defect*) yields a payoff of (1, 1), which is a smaller payoff for both players than the combination (*cooperate, cooperate*) which yields a payoff of (3, 3). Moreover the combination (*cooperate, cooperate*) is a Pareto optimal solution.

For the second subclass we considered the battle of the sexes game (Gintis, 2000; Weibull, J.W., 1996). In this game a married couple loves each other so much they want to do everything together. One night the husband wants to see a movie and the wife wants to go to the opera. This situation is described by the payoff matrices of Table III. If they both do their activities separately they receive the lowest payoff. In this game there are 2 pure strategy Nash equilibria, i.e. (*movie, movie*) and (*opera, opera*), which both are also evolutionary stable (as demonstrated in section 2.3.4). There is also 1 mixed Nash equilibrium, i.e. where the row player (the husband) plays *movie* with 2/3 probability and *opera* with 1/3 probability and the column player (the wife) plays *opera* with 2/3 probability and *movie* with 1/3 probability. However, this equilibrium is not an evolutionary stable one (as demonstrated in section 2.3.4).

The third class consists of the games with a unique mixed equilibrium. For this category we used the game defined by the matrices in

Table III. Battle of the sexes: The left matrix (A) defines the payoff for the row player, the right one (B) for the column player.

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

Table IV. This equilibrium is not an evolutionary stable one (see section 2.3.4). Typical for this class of games is that the interior trajectories define closed orbits around the equilibrium point.

Table IV. The left matrix (A) defines the payoff for the row player, the right one (B) for the column player.

$$A = \begin{pmatrix} 2 & 3 \\ 4 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 3 & 1 \\ 2 & 4 \end{pmatrix}$$

2.3. POPULATION DYNAMICS

In this section we discuss the Replicator Dynamics in a single and a multi population setting. We discuss the relation with concepts as Nash equilibrium and ESS and illustrate the described ideas with some examples.

2.3.1. *Single Population Replicator Dynamics*

The basic concepts and techniques developed in EGT were initially formulated in the context of evolutionary biology. (Maynard-Smith, J., 1982; Weibull, J.W., 1996; Samuelson, L., 1997). In this context, the strategies of all the players are genetically encoded (called genotype). Each genotype refers to a particular behavior which is used to calculate the payoff of the player. The payoff of each player's genotype is determined by the frequency of other player types in the environment.

One way in which EGT proceeds is by constructing a dynamic process in which the proportions of various strategies in a population evolve. Examining the expected value of this process gives an approximation which is called the RD. An abstraction of an evolutionary process usually combines two basic elements: **selection** and **mutation**. Selection favors some varieties over others, while mutation provides variety in the population. The replicator dynamics highlight the role

of selection, it describes how systems consisting of different strategies change over time. They are formalized as a system of differential equations. Each replicator (or genotype) represents one (pure) strategy s_i . This strategy is inherited by all the offspring of the replicator. The general form of a replicator dynamic is the following:

$$\frac{dx_i}{dt} = [(A\mathbf{x})_i - \mathbf{x} \cdot A\mathbf{x}]x_i \quad (1)$$

In equation (1), x_i represents the density of strategy s_i in the population, A is the payoff matrix which describes the different payoff values each individual replicator receives when interacting with other replicators in the population. The state of the population (\mathbf{x}) can be described as a probability vector $\mathbf{x} = (x_1, x_2, \dots, x_J)$ which expresses the different densities of all the different types of replicators in the population. Hence $(A\mathbf{x})_i$ is the payoff which replicator s_i receives in a population with state x and $\mathbf{x} \cdot A\mathbf{x}$ describes the average payoff in the population. The growth rate $\frac{\frac{dx_i}{dt}}{x_i}$ of the population share using strategy s_i equals the difference between the strategy's current payoff and the average payoff in the population. For further information we refer the reader to (Weibull, J.W., 1996; Hofbauer et al., 1998).

2.3.2. Multi-Population Replicator Dynamics

So far the study of population dynamics was limited to a single population. However in many situations interaction takes place between 2 or more individuals from different populations. In this section we study this situation in the 2-player multi-population case for reasons of simplicity. Games played by individuals of different populations are commonly called **evolutionary asymmetric games**. Here we consider a game to be played between the members of two different populations. As a result, we need two systems of differential equations: one for the row player (R) and one for the column player (C). This setup corresponds to a RD for asymmetric games. If $A = B^t$ (the transpose of B), equation (1) would emerge again. Player R has a probability vector p over its possible strategies and player C a probability vector q over its strategies.

This translates into the following replicator equations for the two populations:

$$\frac{dp_i}{dt} = [(A\mathbf{q})_i - \mathbf{p} \cdot A\mathbf{q}]p_i \quad (2)$$

$$\frac{dq_i}{dt} = [(B\mathbf{p})_i - \mathbf{q} \cdot B\mathbf{p}]q_i \quad (3)$$

As can be seen in equation (2) and (3), the growth rate of the types in each population is now determined by the composition of the other population. Note that, when calculating the rate of change using these systems of differential equations, two different payoff matrices (A and B) are used for the two different players.

2.3.3. *Relating Nash, ESS and the RD*

As being a system of differential equations, the RD have some rest points or equilibria. An interesting question is how these RD-equilibria relate to the concepts of Nash equilibria and ESS. We briefly summarize some known results from the EGT literature (Weibull, J.W., 1996; Gintis, 2000; Osborne et al., 1994; Hofbauer et al., 1998; Redondo, F.V., 2001). An important result is that every Nash equilibrium is an equilibrium of the RD. But the opposite is not true. This can be easily understood as follows. Let us consider the vector space or simplex of mixed strategies determined by all pure strategies. Formally the unit simplex is defined by,

$$\Delta = \{x \in \mathfrak{R}_+^m : \sum_{i=1}^m x_i = 1\}$$

where x is a mixed strategy in m -dimensional space (there are m pure strategies), and x_i is the probability with which strategy s_i is played. Calculating the RD for the unit vectors of this space (putting all the weight on a particular pure strategy), yields zero. This is simply due to the properties of the simplex Δ , where the sum of all population shares remains equal to 1 and no population share can ever turn negative. So, if all pure strategies are present in the population at any time, then they always have been and always will be present, and if a pure strategy is absent from the population at any time, then it always has been and always will be absent³. So, this means that the pure strategies are rest points of the RD, but depending on which game is played these pure strategies do not need to be a Nash equilibrium. Hence not every rest point of the RD is a Nash equilibrium. So dynamic equilibrium or stationarity alone is not enough to have a better understanding of the RD.

For this reason the criterion of asymptotic stability came along, where you have some kind of local test of dynamic robustness. Local in the

³ Off course a solution orbit can evolve toward the boundary of the simplex as time goes to infinity, and thus in the limit, when the distance to the boundary goes to zero, a pure strategy can disappear from the population of strategies. For a more formal explanation, we refer the reader to (Weibull, J.W., 1996)

sense of minimal perturbations. For a formal definition of asymptotic stability, we refer to (Hirsch et al., 1974). Here we give an intuitive definition. An equilibrium is asymptotic stable if the following two conditions hold:

- Any solution path of the RD that starts sufficiently close to the equilibrium remains arbitrarily close to it. This condition is called **Liapunov stability**.
- Any solution path that starts close enough to the equilibrium, converges to the equilibrium.

Now, if an equilibrium of the RD is asymptotically stable (i.e. being robust to local perturbations) then it is a Nash equilibrium. For a proof, the reader is referred to (Redondo, F.V., 2001). An interesting result due to Sigmund and Hofbauer (Hofbauer et al., 1998) is the following : If s is an ESS, then the population state $x = s$ is asymptotically stable in the sense of the RD. For a proof see (Hofbauer et al., 1998; Redondo, F.V., 2001). So, by this result we have some kind of refinement of the asymptotic stable rest points of the RD and it provides a way of selecting equilibria from the RD that show dynamic robustness.

2.3.4. *Examples*

In this section we continue with the examples of section 2.2.6 and the classification of games of section 2.2.1. We start over with the Prisoner's Dilemma game (PD). In figure 2 we plotted the direction field of the replicator equations applied to the PD. A Direction field is a very elegant and excellent tool to understand and illustrate a system of differential equations. The direction fields presented here consist of a grid of arrows tangential to the solution curves of the system. Its a graphical illustration of the vector field indicating the direction of the movement at every point of the grid in the state space. Filling in the parameters for each game in equations 2 and 3, allowed us to plot this field.

The x-axis represents the probability with which the first player will play defect and the y-axis represents the probability with which the second player will play defect. So the Nash equilibrium and the ESS lie at coordinates (1, 1). As you can see from the field plot all the movement goes toward this equilibrium.

Figure 3 illustrates the direction field diagram for the battle of the sexes game. As you may recall from section 2.2.6 this game has 2 pure Nash equilibria and 1 mixed Nash equilibrium. This equilibria can be

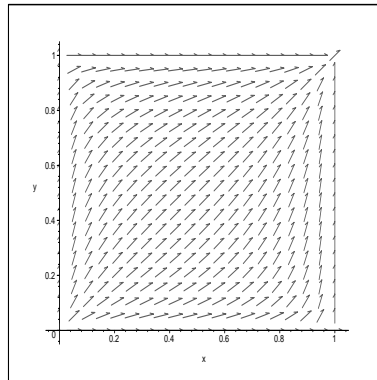


Figure 2. The direction field of the RD of the prisoner's dilemma using payoff Table II.

seen in the figure at coordinates $(0, 0)$, $(1, 1)$, $(2/3, 1/3)$. The 2 pure equilibria are ESS as well. This is also easy to verify from the plot, more precisely, any small perturbation away from the equilibrium would lead the dynamics back to the equilibrium.

The mixed equilibrium, which is Nash, is not an asymptotic stable strategy, which is obvious from the plot. From section 2.2.6, we can now also conclude that this equilibrium is not evolutionary stable either.

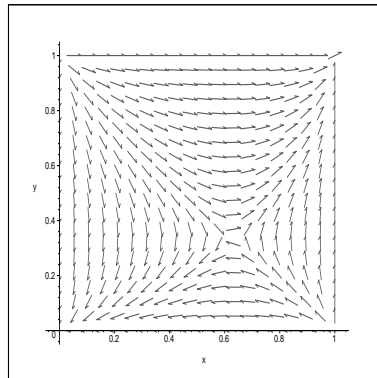


Figure 3. The direction field of the RD of the Battle of the sexes game using payoff Table III.

Figure 4 illustrates the last class of games (subclass 3). Typical for this class of games is that the interior trajectories define closed orbits around the equilibrium point, as you can see in the plot. This Nash equilibrium is not asymptotically stable, because its second condition is not met, which stated that any solution path that starts close enough to the equilibrium, converges to the equilibrium. However, the first condition,

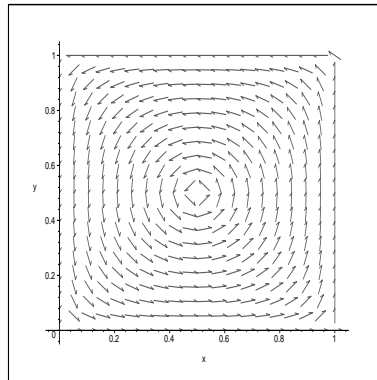


Figure 4. The direction field of the RD of the third category using payoff Table IV.

i.e. Liapunov stability, is met, stating that any solution path of the RD that starts sufficiently close to the equilibrium remains arbitrarily close to it. This can be intuitively understood from the plot.

3. Reinforcement Learning and Multi-Agent Systems

In this part we discuss the relation between learning and MAS (see figure 1). Recall from section 1 that learning and adaptiveness is crucial for the successful application of Multi-Agent Systems to challenging domains as for instance Robotic Soccer (Stone, P., 2000). In a first section we start with the already established theory of Single-Agent learning. We continue with the more challenging issues of Multi-Agent learning and discuss the different possible approaches.

3.1. SINGLE AGENT REINFORCEMENT LEARNING

RL is the problem faced by an agent that learns behavior through trial-and-error interactions with a dynamic environment. A reinforcement learning model consists of:

1. a discrete set of environment states
2. a discrete set of agent actions
3. a set of scalar reinforcement signals.

On each step of interaction the agent receives a reinforcement, possibly zero, and some indication of the current state of the environment, and chooses an action. The agent's job is to find a policy mapping states to

actions, that maximizes some long-run measure of reinforcement. Very often this measure is the discounted cumulative reward.

In its most general form, the RL problem is a problem of an agent located in an environment δ trying to maximize a long-term reward by taking actions a from different situations in δ . Figure 5 illustrates this problem statement in more detail. At time step t the agent finds itself

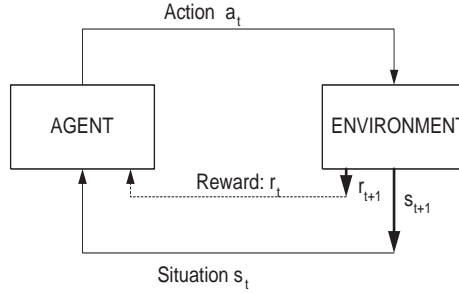


Figure 5. The reinforcement learning model.

in situation (or state) s_t . From s_t it takes action a_t . The environment reacts and places the agent in situation s_{t+1} . By performing action a_t the agent receives an immediate reward r_t . The immediate reward depends on either or both the action taken, and the next state. To choose an action a_t from a particular state s_t at time step t the agent uses a policy π_t , with $\pi_t(s, a)$ the probability that in state s at time step t action a_t will be performed.

Common reinforcement learning methods, which can be found in (Sutton et al., 2000) are structured around estimating value functions. A value of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state. More formally we have:

$$V^\pi(s) = E_\pi\{R_t | s_t = s\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s\right\}$$

Rewards further away in the future are discounted by γ with $0 < \gamma < 1$. One way to find the optimal policy is to find the optimal value function. If a perfect model of the environment as a Markov decision process is known, the optimal value function can be calculated using Dynamic Programming (DP) techniques. In DP two major approaches exist, i.e. **value-iteration** and **policy-iteration**. Both approaches have their

counter parts in RL, which can be considered as model-free⁴ stochastic approximation methods of the DP techniques.

Q-learning is a well-known RL technique that belongs to the value-iteration class. It learns an evaluation function for each situation- action pair. This function Q is defined by

$$Q^\pi(s, a) = E_\pi\{R_t | s_t = s, a_t = a\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a\right\}$$

Thus, the Q-function expresses the expected reward if an agent takes action a in state s and then continues with policy π . Based on his experience, the agent can iteratively improve this evaluation function and as such adapt the policy π to the ideal policy π^* , which maximizes the long term reward. The relation between $V^\pi(s)$ and $Q^\pi(s, a)$ is:

$$V^\pi(s) = \max_a Q^\pi(s, a)$$

The update rule used in standard Q-learning is given below

$$Q_{t+1}(s, a) \leftarrow (1 - \alpha)Q_t(s, a) + \alpha(r + \gamma \max_{a'} Q_t(s', a')) \quad (4)$$

Where $Q_t(s', a')$ and $Q_{t+1}(s, a)$ are the estimation of the state-action value at time step t and $t + 1$ respectively, and s' the state where the agent arrives after taking action a at time step $t + 1$ in situation s .

Actor-critic methods are RL techniques that belong to the policy iteration class. These methods keep track of a current policy, the actor, and an estimate of corresponding state-value function. The critic is based on this estimated value function. It generates a Temporal Difference error, which is in its simplest form given by: $e_t = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$. When the error is positive the tendency of repeating the action taken in state s_t will be reinforced, otherwise it will be weakened. A learning cycle goes as follows. The agent is at time step t in a certain state s_t , for that state the current best action is chosen, with some degree of exploration. The selected action will bring the agent in a new state s_{t+1} . If the new state s_{t+1} looks better, i.e. $e_t > 0$, then the action a_t for state s_t will be strengthened, e.g. by increasing the probability to be selected, otherwise this probability will be decreased. (Sutton et al., 2000). Figure 6 illustrates the classification of RL-methods.

⁴ A model consists of knowledge of the state transition probability function $T(s, a, s')$, which is the probability of ending up in state s' after taking action a in state s , and the reinforcement function $R(s, a)$, which is the payoff for taking action a in state s .

		Model of the environment ?	
		Yes	No
Bootstrap ?	Yes	Dynamic programming	Temporal Difference (TD)
	No	—	Monte Carlo

Figure 6. Classification of RL-algorithms.

3.2. MULTI-AGENT REINFORCEMENT LEARNING

The original reinforcement learning algorithms as mentioned above, were designed to be used in a single agent setting. When applied to Markovian decision problems most RL techniques are equipped with a formal proof stating that under very acceptable conditions they are guaranteed to converge to the optimal policy, for instance for Q-learning see Tsitsiklis (Tsitsiklis, J.N., 1993). There has also been quite some effort to extend these RL techniques to Partially Observable Markovian Decision Problems and other non-Markovian settings (Loch et al., 1998; Pendrith et al., 1998; Kaelbling et al., 1996; Perkins et al., 2002).

The extension to multi-agent learning recently received more attention. It is clear that the actions taken by one agent might affect the response characteristics of the environment. So we can no longer assume the Markovian property holds. In the domain of Learning Automata, this is referred to as *state dependent non-stationarity* (Narendra, K. et al., 1989). When applying RL to a multi-agent case, two extreme approaches can be taken. The first one totally neglects the presence of the other agents, and agents are considered to be selfish reinforcement learners. The effects caused by the other agents also acting in that same environment are considered as noise. It is clear that for problems where agents have to coordinate in order to reach a preferable situation for both actions, this will not yield satisfactory results (Hu et al., 1999). The other extreme is the joint action space approach where the state and action space are respectively defined as the Cartesian product of the agent's individual state and action spaces. More formally, if S is the set of states and A_1, \dots, A_n the action sets of the different agents the learning will be performed in the product space

$$S \times A_1 \times \dots \times A_n \rightarrow \mathfrak{R} \quad (5)$$

This implies that the state information is shared amongst the agents and actions are taken and evaluated synchronously. It is obvious that this approach leads to very big state-action spaces, and assumes instant

communication between the agents. Clearly this approach is in contrast with the basic principles of multi-agent systems: distributed control, asynchronous actions, incomplete information, cost of communication. In between these approaches we can find examples which try to overcome the drawbacks of the joint action approach, some references are (Litmann et al., 1994; Claus et al., 1998; Jafari et al., 2001; Nowé et al., 1999). Below we describe cross-learning, which can be considered as multi-agent RL, and is important in clarifying the relationship between RL and EGT.

Cross learning is a less complex model than Q-learning and even Learning Automata (LA, see section 3.2.2) in the sense that it does not require an initialisation and that there are not so many parameters to fine tune as in LA and Q-learning. Cross learning does not consider a learning rate, a discount factor and an exploration strategy⁵ as Q-learning does, nor needs reward and penalty parameters⁶ as LA does.

3.2.1. Cross Learning

The cross learning model is a special case of the standard reinforcement learning model (Sarin et al., 1997). The model considers several agents playing the same normal form game repeatedly in discrete time. At each point in time, each player is characterized by a probability distribution over his strategy set which indicates how likely he is to play any of his strategies. The probabilities change over time in response to experience. At each time step (indexed by n), a player chooses one of its strategies based on the probabilities which are related to each isolated strategy. Positive payoffs represent *reinforcing* experiences, which induce a player to increase the probability of the strategy chosen. So, the larger the payoff, the larger the increase and thus the bigger the strength of reinforcement. As a result a player can be represented by a probability vector:

$$p(n) = (p_1(n), \dots, p_r(n))$$

In case of a 2-player game with payoff matrix U , player 1 gets payoff

Table V. A payoff table U

$$U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}$$

U_{ij} when he chooses strategy i and player 2 chooses strategy j . We

⁵ For instance in Boltzmann Q-learning the temperature determines the degree of exploration.

⁶ Parameters a and b in equations 10 and 11

assume that

$$0 < U_{ij} < 1 \quad (6)$$

In this case there is no deterrence. The iterations of the game are indexed by $n \in N$. Players do not observe each other's strategies and payoffs and play the game repeatedly. After making their observations, each stage they update their probability vector, according to,

$$p_i(n+1) = U_{ij} + (1 - U_{ij})p_i(n) \quad (7)$$

$$p_{i'}(n+1) = (1 - U_{ij})p_{i'}(n) \quad (8)$$

Equation (7) expresses how the probability of the selected strategy (i) is updated and equation (8) expresses how all the other strategies $i' \neq i$ are corrected. If this player p played strategy i in the n th repetition of the game, and if he received payoff U_{ij} , then he updates his state by taking a weighted average of the old state, and of the unit vector which puts all probability on strategy i .

The probability vector $q(n)$ (for the second player),

$$q(n) = (q_1(n), \dots, q_s(n))$$

is updated in an analogous manner. This entire system of equations defines a stochastic update process for the players $\{p(n), q(n)\}$. This process is called the "Cross learning process" in (Sarin et al., 1997). Börgers and Sarin showed that in an appropriately constructed continuous time limit, this model converges to the asymmetric, continuous time version of the replicator dynamics (see section 2.3).

3.2.2. Learning Automata

Learning Automata have their origins in mathematical psychology (Bush et al., 1955). Originally, Learning Automata were deterministic and based on complete knowledge of the environment. Later developments came up with uncertainties in the system and the environment and lead to the stochastic automaton. More precisely, the stochastic automaton tries to provide a solution of the learning problem without having any information on the optimal action initially. It starts with equal probabilities on all actions and during the learning process these probabilities are updated based on responses from the environment.

In Figure 7 a Learning Automaton is illustrated in its most general form. The environment is represented by a triple $\{\alpha, c, \beta\}$, where α represents a finite action set, β represents the response set of the environment, and c is a vector of penalty probabilities, where each component c_i corresponds to an action α_i .

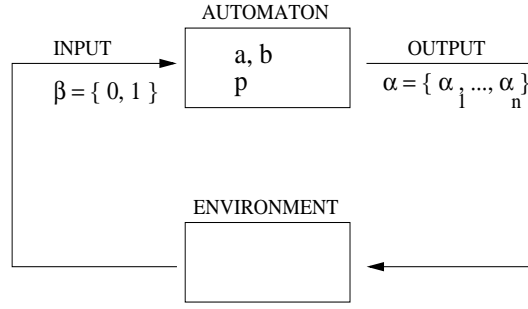


Figure 7. A Learning Automaton - Environment pair.

The response β from the environment can take on 2 values β_1 or β_2 . Often they are chosen to be 0 or 1, where 1 is associated with a penalty response and 0 with a reward. Now, the penalty probabilities can be defined as

$$c_i = P(\beta(n) = 1 | \alpha(n) = \alpha_i) \quad (9)$$

So c_i is the probability that action α_i will result in a penalty response. If these probabilities are constant, the environment is called stationary.

Several models are recognized by the response set of the environment. Models in which the response β can only take 2 values are called P-models. Models which allow a finite number of values in a fixed interval are called Q-models. When β is a continuous random variable in a fixed interval, the model is called an S-model. In a variable structure stochastic automaton⁷ action probabilities are updated at every stage using a reinforcement scheme. The vector p is the action probability vector over the possible actions as with Cross Learning in the previous section. Important examples of update schemes are linear reward-penalty, linear reward-inaction and linear reward- ϵ -penalty. The philosophy of those schemes is essentially to increase the probability of an action when it results in a success and to decrease it when the response is a failure. The general algorithm at timestep $n + 1$ is given by:

$$p_i(n+1) = p_i(n) + a(1 - \beta(n))(1 - p_i(n)) - b\beta(n)p_i(n) \quad (10)$$

if α_i is the action taken at time n

$$p_j(n+1) = p_j(n) - a(1 - \beta(n))p_j(n) + b\beta(n)[(r - 1)^{-1} - p_j(n)] \quad (11)$$

if $\alpha_j \neq \alpha_i$

⁷ As opposed to fixed structure learning automata, where state transition probabilities are fixed and have to be chosen according to the response of the environment and to perform better than a pure-chance automaton in which every action is chosen with equal probability.

where equation 10 is the update rule for the performed action α_i and equation 11 for all the other actions. The constants a en b are the reward and penalty parameters respectively. When $a = b$ the algorithm is referred to as linear reward-penalty (L_{R-P}), when $b = 0$ it is referred to as linear reward-inaction (L_{R-I}) and when b is small compared to a it is called linear reward- ϵ -penalty ($L_{R-\epsilon P}$).

If the penalty probabilities c_i of the environment are constant, the probability $p(n+1)$ is completely determined by $p(n)$ and hence $p(n)_{n>0}$ is a discrete-time homogeneous Markov process. Convergence results for the different schemes are obtained under the assumptions of constant penalty probabilities, see (Narendra, K. et al., 1989).

Learning automata can also be connected in useful ways. A simple example of a multi-agent system modeled as an automata game is shown in figure 8.

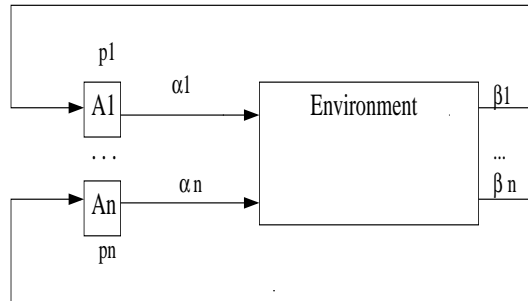


Figure 8. Automata Game representation.

A play $\alpha(t) = (\alpha^1(t) \dots \alpha^n(t))$ of n automata is a set of strategies chosen by the automata at stage t . Correspondingly the outcome is now a vector $\beta(t) = (\beta^1(t) \dots \beta^n(t))$. At every instance all automata update their probability distributions based on the responses of the environment. Each automaton participating in the game operates without information concerning payoff, the number of participants, their strategies or actions.

4. Reinforcement Learning and Evolutionary Game Theory

In this section we discuss the relation between Reinforcement Learning and EGT. We show how the 2 fields are formally related, with results from economics and computer science. Some examples will illustrate the strength of these results. More precisely, we show some examples of the dynamics of Q-learning and LA and show how EGT can be extended to be used as a formal foundation for the construction of new

RL algorithms for MAS. The first subsection summarizes some main results from economics, and the second will deal with extensions of these results. The third subsection will show how this formal relations between RL and EGT can be used as a foundation for modeling new RL algorithms for MAS, i.e. as an initial framework for RL in MAS.

4.1. THE FORMAL RELATION BETWEEN CROSS LEARNING AND EGT

In their paper, *Learning through Reinforcement and Replicator Dynamics*, Börgers and Sarin prove an interesting link between EGT and Reinforcement Learning (Sarin et al., 1997). More precisely they considered a version of R.R. Bush and F. Mosteller (Bush et al., 1955) stochastic learning theory in the context of games and proved that in a continuous time limit, the learning model converges to the asymmetric continuous time replicator equations⁸ of EGT. With this result they provided a formalization of the relation between learning at the individual level and biological evolution.

The version of the learning model of Bush and Mosteller is called Cross learning and has been thoroughly explained in section 3.2.1. It is important to note that each time interval has to see many iterations of the game, and that the adjustments which players make between two iterations of the game are very small. If the limit is constructed in this manner, a law of large numbers can be applied, and the learning process converges, in the limit, to the replicator dynamics. Important to understand is that this result refers to arbitrary, points in finite time. The result does not hold if infinite time is considered. The asymptotic behaviour for time tending to infinity of the discrete time learning can be quite different from the asymptotic behaviour of the continuous time RD. For the mathematical proof of this result we refer the interested reader to (Sarin et al., 1997).

Before continuing this discussion we illustrate this result with the prisoner's dilemma game. In Figure 9 we plotted the direction field of the RD and the Cross learning process for this game.

More precisely, the figure on the left illustrates the direction field of the replicator dynamics and the figure on the right shows the learning process of Cross. We plotted for both players the probability of choosing their first strategy (in this case defect). As you can see the sample paths of the reinforcement learning process approximates the paths of the RD.

⁸ Recall from section 2.3.2 the definition of the asymmetric RD

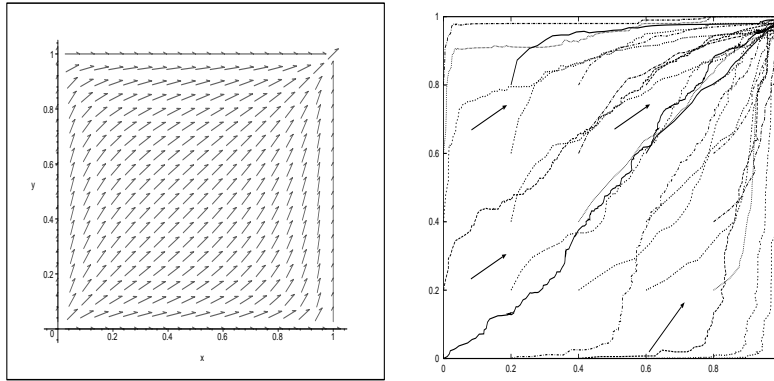


Figure 9. *Left:* The direction field of the RD of the prisoner's game. *Right:* The paths induced by the learning process.

As mentioned before, the result of Börgers and Sarin only holds for a point in time t with $t < \infty$. It doesn't apply however to the asymptotic behaviour for $t \rightarrow \infty$. Moreover, the asymptotic behaviour of the learning process may be very different from that of the continuous RD. To show this we demonstrate a result of Börgers and Sarin concerning the discrete time learning process. This result says that, with probability 1, the learning process will converge to a limit in which both players play some pure strategy. For a mathematical proof of this proposition we refer to (Sarin et al., 1997). Recall from section 2.3.4 that in the third category of games, the RD circle around the mixed Nash equilibrium. Figure 10 now clearly illustrates that in this type of game the asymptotic behaviour is different for both models.

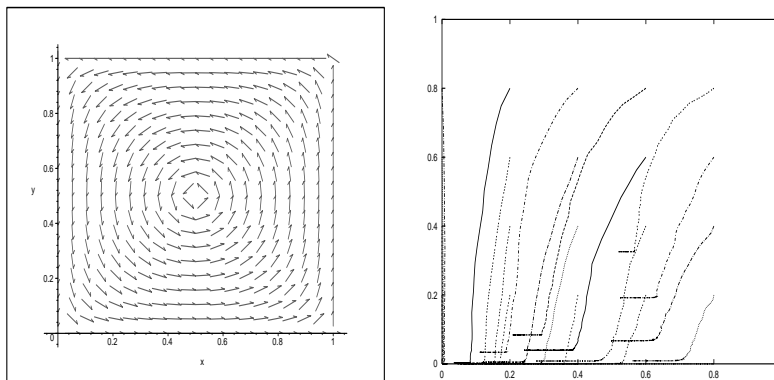


Figure 10. *Left:* The direction field of the RD. *Right:* The paths induced by the learning process.

4.2. EXTENDING THE FORMAL RELATION TO OTHER RL-MODELS

4.2.1. *Learning Automata and Evolutionary Dynamics*

In (Tuyls et al., October'02) it is shown by the authors that the Cross learning model is a Learning Automaton with a linear-reward-inaction updating scheme. To provide the reader with an intuition on this relation we briefly describe the mathematical relation between both learning models. More details and a variety of experiments can be found in (Tuyls et al., October'02). Showing that the Cross learning model is a special case of LA we need to relate the equations (7) and (8) with (10) and (11).

If it is assumed that $b = 0$ in equations (10) and (11), the relation between both models becomes apparent. We now see that the Cross learning model is in fact a special case of the reward-inaction update scheme. When the reward penalty term $a = 1$, the feedback from the environment $(1 - \beta(n))$ equals the game reward U_{jk}^R . Hence the equations become equivalent. The experiments in (Tuyls et al., October'02) have been conducted with both the linear reward-inaction update scheme and the reward- ϵ -penalty update scheme.

4.2.2. *Q-learning and Evolutionary Dynamics*

In this paragraph we briefly⁹ describe the relation between Q-learning and the RD. More precisely we present the dynamical system of Q-learning. These equations are derived by constructing a continuous time limit of the Q-learning model, where Q-values are interpreted as Boltzmann probabilities for the action selection. Again we consider games between 2 players. The equations for the first player are,

$$\frac{dx_i}{dt} = x_i \alpha \tau ((A\mathbf{y})_i - \mathbf{x} \cdot A\mathbf{y}) + x_i \alpha \sum_j x_j \ln\left(\frac{x_j}{x_i}\right) \quad (12)$$

analogously for the second player, we have,

$$\frac{dy_i}{dt} = y_i \alpha \tau ((B\mathbf{x})_i - \mathbf{y} \cdot B\mathbf{x}) + y_i \alpha \sum_j y_j \ln\left(\frac{y_j}{y_i}\right) \quad (13)$$

Equations 12 and 13 express the dynamics of both Q-learners in terms of Boltzmann probabilities¹⁰. Each agent(or player) has a probability vector over his action set, more precisely x_1, \dots, x_n over action set

⁹ The reader who is interested in the complete derivation of the dynamics of Q-learning, we refer to (Tuyls et al., July'03)

¹⁰ Formally the Boltzmann distribution is described by,

$$x_i(k) = \frac{e^{\tau Q_{a_i}(k)}}{\sum_{j=1}^n e^{\tau Q_{a_j}(k)}}$$

a_1, \dots, a_n for the first player and y_1, \dots, y_m over b_1, \dots, b_m for the second player.

For a complete discussion on this equations we refer to (Tuyls et al., July'03). Comparing (12) or (13) with the RD in (1), we see that the first term of (12) or (13) is exactly the RD of EGT and thus takes care of the selection mechanism, see (Weibull, J.W., 1996). The second term turned out to be a mutation term, and can be rewritten as:

$$x_i \alpha \sum_j x_j \ln(x_j) - \ln(x_i) \tag{14}$$

In equation (14) we recognize 2 entropy terms, one over the entire probability distribution x , and one over strategy x_i . Relating entropy and mutation is not new. It is a well known fact (Schneider, T.D., 2000; Stauffer, D., 1999) that mutation increases entropy. In (Stauffer, D., 1999), it is stated that the concepts are familiar with thermodynamics in the following sense: the selection mechanism is analogous to *energy* and mutation to *entropy*. So generally speaking, mutations tend to increase entropy. Exploration can be considered as the mutation concept, as both concepts take care of providing variety.

Figure 4.2.2 illustrates the dynamics of Q-learning in the battle of the sexes game. The direction field is plotted for three values of the temperature τ .

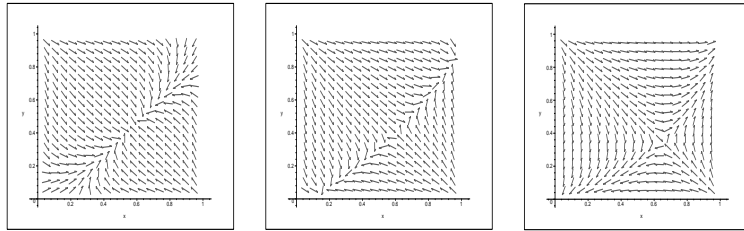


Figure 11. The direction field plots of the battle of the sexes (subclass 2) game with $\tau = 1, 2, 10$

4.3. EXTENDED REPLICATOR DYNAMICS (ERD): USING THE INITIAL FRAMEWORK

In (Tuyls et al., September'03) the authors changed the RD in a new kind of dynamics, i.e. the Extended Replicator Dynamics (ERD). The reasons for changing the RD become clear from section 4.1 and (Tuyls et

where $x_i(k)$ is the probability of playing strategy i at time step k and τ is the temperature.

al., June'03; Tuyls et al., July'03). In one-state games it is impossible for Cross learning and Learning Automata to guarantee convergence to a stable Nash equilibrium in all types of games. In Boltzmann Q-learning a Nash equilibrium can be attained, but there is no guarantee for stability.

For the development of an adapted selection dynamics, we took the replicator dynamics and its interpretation as a starting point. In RD, the probabilities a player has over its strategies are changed greedily with respect to payoff in the present. In this section a method is shown to change these probabilities over strategies not only with respect to payoff growth in the present but also to payoff growth in the future. We call those players that act so as to optimize future payoff extended Cross learners and the class of dynamics associated extended dynamics. There are of course different ways to build such extended players. The most obvious is to use a linear approximation of the evolution of fitness in time. This is the approach we use here.

For the ERD we compose the following equation f ,

$$f(x) = RD(x) + (dRD(x)/dt) * \eta \quad (15)$$

where $RD(x)$ is,

$$\frac{dx_i}{dt} = [(A\mathbf{x})_i - \mathbf{x} \cdot A\mathbf{x}]x_i \quad (16)$$

and η is the parameter that determines how far in the future we need to look.

The composition of equation 15 can best be understood as follows. When using the classical replicator equations (i.e. $RD(x)$), we act greedily toward payoff in the present. When adding our second term,

$$(dRD(x)/dt) * \eta \quad (17)$$

we act greedily toward payoff in the future. From an analytical point of view, the second term gives actions that are winning fitness (whether its fitness is negative or positive) a positive push toward a higher chance of getting selected. On the other hand, actions that are losing fitness (again whether its fitness is negative or positive) are given a negative push toward a lower chance of getting selected. This extends the traditional replicator equations. This extended evolutionary dynamics succeeds in converging to a stable Nash Equilibrium in all 3 categories of 2*2 games. In (Tuyls et al., September'03) we also constructed a model free RL algorithm which behaves as the ERD, based on Cross learning. Experiments confirming this can be found in (Tuyls et al., September'03). Here we show an experiment for the third category of

games. As you recall from section 4.1 this type of game shows an important difference with our ERD. ERD and the extended Cross learning algorithm will not circle but converge to the mixed Nash equilibrium. This is illustrated in figure 12. Moreover the equilibrium is stable, meaning that the learning process will not abandon it. The long-run learning dynamics are illustrated in the figure on the right.

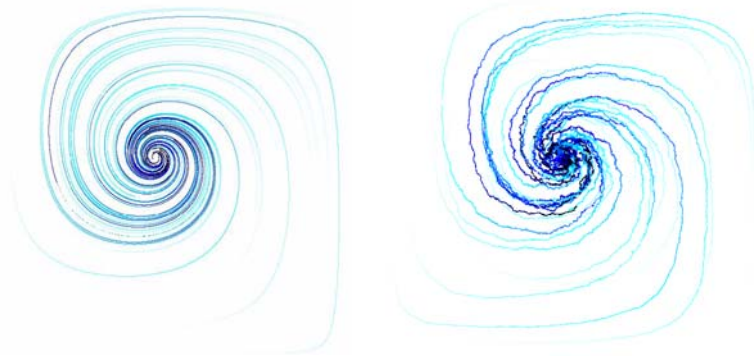


Figure 12. *Left:* The direction field of the RD. *Right:* The paths induced by the learning process.

5. Evolutionary Game Theory and Multi-Agent Systems

In this section we discuss the most interesting properties that link the fields of EGT and MAS.

Traditional Game theory is an economical theory that models interactions between rational agents as games of two or more players that can choose from a set of strategies and the corresponding preferences. It is the mathematical study of interactive decision making in the sense that the agents involved in the decisions take into account their own choices and those of others. Choices are determined by

1. stable preferences concerning the outcomes of their possible decisions,
2. agents act strategically, in other words, they take into account the relation between their own choices and the decisions of other agents.

Typical for the traditional game theoretic approach is to assume perfectly rational players who try to find the most rational strategy to play. These players have a perfect knowledge of the environment and the payoff tables and they try to maximize their individual payoff.

These assumptions made by classical game theory just do not apply to the real world and Multi-Agent settings in particular.

In contrast, EGT is descriptive and starts from more realistic views of the game and its players. A game is not played only once, but repeatedly with changing opponents that, moreover, are not completely informed, sometimes misinterpret each others' actions, and are not completely rational but also biologically and sociologically conditioned. Under these circumstances, it becomes impossible to judge what choices are the most rational ones. The question now becomes how a player can learn to optimize its behaviour and maximize its return. For this learning process, mathematical models are developed, e.g. replicator equations.

Summarizing the above we can say that EGT describes how **boundedly** rational agents can make decisions in complex environments, in which they interact with other agents. Bounded rationality means that agents are limited in their computational resources, in their ability to reason and have limited information. In such complex environments software agents must be able to learn from their environment and adapt to its non-stationarity.

The basic properties of a Multi-Agent System correspond exactly with that of EGT. First of all, a MAS is made up of interactions between two or more agents, with each trying to accomplish a certain (conflicting) goal. Not any agent has the guarantee to be completely informed about the other agents intentions or goals, nor has it the guarantee to be completely informed about the complete state of the environment. Of great importance is that EGT offers us a solid basis to understand dynamic iterative situations in the context of strategic games. A MAS has a typical dynamical character, which makes it hard to model and brings along a lot of uncertainty. At this stage EGT seems to offer us a helping hand in understanding this typical dynamical processes in a MAS and modeling them in simple settings as iterative games of two or more players.

6. Conclusions

By starting with a concise overview on key concepts from EGT and their mutual relationships, we provided an Evolutionary game theoretic point of view on Learning and MAS. By introducing the triangular relation between MAS, RL and EGT we formalized this perspective into one that results in new insights in MAS and more particularly for the indispensable concept of Learning in MAS. These new insights make it possible to overcome important daily occurrences and crucial

learning issues in MAS.

In this relation, EGT provides the necessary basic foundations for understanding and analyzing MAS. Because of the similarities between the two fields (recall section 5), EGT is vital to MAS as a framework for learning and today's applications. This is the first link of figure 1. The second link provided the mathematical formalization of the relation between RL and EGT. Again this relation offers a better understanding of learning in a MAS and provides basic mechanisms toward learning algorithms with more capabilities in the future (Tuyls et al., September'03). More precisely, in the case of 1-state games this means stable strategies which are possibly mixed. Obviously, this opens the door toward multi-state games, and MAS.

These two first links of figure 1 are the keys toward solving the issues in the third link between RL and MAS. These main difficulties in MAS were profoundly described in section 3. It is our belief that today's MAS applications require agents to be adaptive and hence will benefit from this evolutionary game theoretic perspective.

References

- Bazzan A. L. C., Klugl Franziska, Learning to Behave Socially and Avoid the Braess Paradox In a Commuting Scenario. *Proceedings of the first international workshop on Evolutionary Game Theory for Learning in MAS*, July 14 2003, Melbourne Australia.
- Bazzan A. L. C., A game-theoretic approach to coordination of traffic signal agents. *PhD thesis, Univ. of Karlsruhe*, 1997.
- Börgers, T., Sarin, R., Learning through Reinforcement and Replicator Dynamics. *Journal of Economic Theory, Volume 77, Number 1*, November 1997.
- Braess D., Über ein paradoxon aus der verkehrsplanung. *Unternehmensforschung 12 (1968), 258*.
- Bush, R. R., Mosteller, F., Stochastic Models for Learning, *Wiley*, New York, 1955.
- Claus, C., Boutilier, C., The Dynamics of Reinforcement Learning in Cooperative Multi-Agent Systems, *Proceedings of the 15th international conference on artificial intelligence*, p.746-752, 1998.
- Ghosh, A., Sen, S., Learning TOMs: Convergence to Non-Myopic Equilibria. *Proceedings of the first international workshop on Evolutionary Game Theory for Learning in MAS*, July 14 2003, Melbourne Australia.
- Gintis, C.M., Game Theory Evolving. *University Press*, Princeton, June 2000.
- Hirsch, M.W., and Smale, S., Differential Equations, Dynamical Systems and Linear Algebra. *Academic Press, Inc*, 1974.
- Hofbauer, J., Sigmund, K., Evolutionary Games and Population Dynamics. *Cambridge University Press*, November 1998.
- Hu, J., Wellman, M.P., Multiagent reinforcement learning in stochastic games. *Cambridge University Press*, November 1998.

- Jafari, C., Greenwald, A., Gondek, D. and Ercal, G., On no-regret learning, fictitious play, and nash equilibrium. *Proceedings of the Eighteenth International Conference on Machine Learning*, p 223 - 226, 2001.
- Kaelbling, L.P., Littman, M.L., Moore, A.W., Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 1996.
- Littman, M.L., Markov games as a framework for multi-agent reinforcement learning. *Proceedings of the Eleventh International Conference on Machine Learning*, p 157 - 163, 1994.
- Loch, J., Singh, S., Using eligibility traces to find the best memoryless policy in a partially observable markov process. *Proceedings of the fifteenth International Conference on Machine Learning, San Francisco*, 1998.
- Luck, M., McBurney, P., Preist, C., A Roadmap for Agent Based Computing. *AgentLink, network of excellence*, 2003.
- Maynard-Smith, J., Evolution and the Theory of Games. *Cambridge University Press*, December 1982.
- Maynard Smith, J., Price, G.R., The logic of animal conflict. *Nature*, 146: 15-18, 1973.
- Narendra, K., Thathachar, M., Learning Automata: An Introduction. *Prentice-Hall*, 1989.
- Nowé, A., Parent, J., Verbeeck, K., Social agents playing a periodical policy. *Proceedings of the 12th European Conference on Machine Learning*, p 382 - 393, 2001.
- Nowé A. and Verbeeck K., Distributed Reinforcement learning, Loadbased Routing a case study, *Notes of the Neural, Symbolic and Reinforcement Methods for sequence Learning Workshop at ijcai99*, 1999, Stockholm, Sweden.
- von Neumann, J., Morgenstern, O., Theory of Games and Economic Behaviour, *Princeton University Press*, 1944.
- Osborne J.O., Rubinstein A., A course in game theory. *Cambridge, MA: MIT Press*, 1994.
- Pendrith M.D., McGarity M.J., An analysis of direct reinforcement learning in non-Markovian domains. *Proceedings of the fifteenth International Conference on Machine Learning, San Francisco*, 1998.
- Perkins T.J., Pendrith M.D., On the Existence of Fixed Points for Q-learning and Sarsa in Partially Observable Domains. *Proceedings of the International Conference on Machine Learning (ICML02)*, 2002.
- Redondo, F.V., Game Theory and Economics, *Cambridge University Press*, 2001.
- Robocup project, The official robocup website at www.robocup.org, *Robocup*, 2003.
- Samuelson, L. Evolutionary Games and Equilibrium Selection, *MIT Press, Cambridge, MA*, 1997.
- Schneider, T.D., Evolution of biological information. *journal of Nucleic Acids Research*, volume 28, pages 2794 - 2799, 2000.
- Stauffer, D., Life, Love and Death: Models of Biological Reproduction and Aging. *Institute for Theoretical physics, Köln, Euroland*, 1999.
- Sutton, R.S., Barto, A.G., Reinforcement Learning: An introduction. *Cambridge, MA: MIT Press*, 1998.
- Stone P., Layered Learning in Multi-Agent Systems. *Cambridge, MA: MIT Press*, 2000.
- Tsitsiklis, J.N., Asynchronous stochastic approximation and Q-learning. *Internal Report from the laboratory for Information and Decision Systems and the Operation Research Center*, MIT 1993.

- Tuyls, K., Lenaerts, T., Verbeeck, K., Maes, S. and Manderick, B., Towards a Relation Between Learning Agents and Evolutionary Dynamics. *Proceedings of the Belgium-Netherlands Artificial Intelligence Conference 2002 (BNAIC)*. KU Leuven, Belgium.
- Tuyls, K., Verbeeck, K., and Maes, S. On a Dynamical Analysis of Reinforcement Learning in Games: Emergence of Occam's Razor. *Lecture Notes in Artificial Intelligence, Multi-Agent Systems and Applications III, Lecture Notes in AI 2691*, (Central and Eastern European conference on Multi-Agent Systems 2003). Prague, 16-18 june 2003, Czech Republic.
- Tuyls, K., Verbeeck, K., and Lenaerts, T. A Selection-Mutation model for Q-learning in Multi-Agent Systems. *The ACM International Conference Proceedings Series, Autonomous Agents and Multi-Agent Systems 2003*. Melbourne, 14-18 juli 2003, Australia.
- Tuyls, K., Heytens, D., Nowe, A., and Manderick, B., Extended Replicator Dynamics as a Key to Reinforcement Learning in Multi-Agent Systems. *Proceedings of the European Conference on Machine Learning'03, Lecture Notes in Artificial Intelligence*. Cavtat-Dubrovnik, 22-26 september 2003, Croatia.
- Weibull, J.W., *Evolutionary Game Theory*, MIT Press 1996.
- Weibull, J.W., What we have learned from Evolutionary Game Theory so far? *Stockholm School of Economics and I.U.I.* may 7, 1998.
- Weiss, G., *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence*. Edited by Gerard Weiss Cambridge, MA: MIT Press. 1999.
- Wooldridge, M., *An Introduction to MultiAgent Systems*. Published in February 2002 by John Wiley, Sons, Chichester, England, 2002.

