Vrije Universiteit Brussel

Faculty of Science and
Bio-Engineering Sciences
Department of Computer Science

Adaptive Heuristics

Master thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in Applied Sciences and Engineering: Computer Science

Stijn Doeraene

Promotor: Prof. Dr. Bernard Manderick
Advisors: David Catteeuw

2011-2012

Vrije Universiteit Brussel

Faculteit Wetenschappen en
Bio-Ingenieurswetenschappen
Departement Computerwetenschappen

Adaptieve Heuristieken

Proefschrift ingediend met het oog op het behalen van de graad van
Master in de Ingenieurswetenschappen: Computerwetenschappen

Stijn Doeraene

Promoter: Prof. Dr. Bernard Manderick
Begeleider: David Catteeuw

2011-2012

Summary

In this thesis, a selection of adaptive heuristics, simplistic learning rules which in contrast with more sophisticated learning models like Bayesian learning are not fully rational, is reviewed with regard to their empirical convergence behavior towards certain solution concepts of which the Nash equilibrium is the most prominent. Although being not fully rational, because of their simple nature, limited use of history and restricted information requirements, these adaptive heuristics have shown promise in a number of practical applications.

The set of learning rules has been grouped into two main sets, based on their underlying strategy. The first one, those using reinforcement learning consisting of models on cumulative payoff matching and Q-learning, reinforce well performing actions based on past play. Another group of models, on the notion of regret, aim to end up in a situation without regret. Especially for this last group of models, several theoretical properties have already been proven with regard to normal form games while this is not the case for the first set. However, whether these properties are also empirically desirable was often not known.

To assess this empirical desirability, along with other aspects like time and memory efficiency, sensitivity to initial conditions and convergence time, the final selection of learning rules is applied to a set of sample games, representing all classes and subclasses of the 2x2 normal form game space and this using a data gathering and visualization tool developed for this purpose.

Although a preference for the Nash equilibrium has been observed regularly, not all learning rules have shown the same consistent behavior when applied to all classes of games. While the models on cumulative payoff matching show a consistent Nash convergence on those games with more straightforward solutions, on other classes of games, especially those without pure Nash equilibria, this Nash convergence is only observed occasionally.

The basic Q-learning models exhibit a similar convergence behavior in which, for most games, depending on initial conditions, convergence is observed towards the Nash equilibrium or towards another state. While the models on FAQ-learning offer a solution, guaranteeing a Nash convergence and this for all games, these models also introduce a long convergence duration, which might be practically undesirable. The addition of leniency has been shown to only improve convergence results minimally.

As for the models on regret minimization, although in origins requiring additional information, using a regret estimation technique, models like RMe can be designed with the same practical characteristics of lacking any opponent information requirements, identical to the models of the previous groups. Although only being estimated, these regret values have shown to be aiding in a consisting convergence towards the Nash equilibrium and this for all tested games in the 2x2 game field.

Despite their limited set of information and use of history, these adaptive heuristics have ultimately given rise to well-performing results with convergence often either to Nash equilibria or towards outcomes with higher payoffs. Nevertheless these general results, only two types of models, FAQ-learning and the regret minimization models like RMe have shown this convergence consistently independent of the game class and this in accordance with their theoretical work. For all others, without knowledge on the game, Nash convergence can not be guaranteed.

Abstract

Adaptive heuristics, simplistic learning rules which in contrast with more sophisticated learning models like Bayesian learning are not fully rational, have shown promise in a number of practical applications because of their simple nature, limited use of history and restricted information requirements. In fact, for some of these rules, no information on the opponent is even required. In this thesis, the convergence characteristics of a selection of adaptive heuristics have been analyzed on the field of 2x2 normal form games with respect to the Nash equilibrium. Using a test framework developed for this purpose, it has been shown that all the considered learning rules do in fact show Nash convergence for at least part of the game space. However, while for some game classes, Nash convergence is frequently observed, other classes show a more irregular behavior. Ultimately, only a few learning rules show a consistent Nash convergence for all game classes in the 2x2 game space.

Acknowledgments

After a year of researching, developing, analyzing results and writing, I would first of all like to thank my advisor David Catteeuw for his numerous valuable suggestions and comments on earlier versions of this document allowing me to improve the quality of my research and this document. Secondly, I would also like to thank my promotor Prof. Dr. Bernard Manderick for the guidance during my research, his review of previous versions and providing me with a multitude of literature which has been shown to be invaluable. Finally I also thank Prof. Dr. Ann Nowé for her suggestions regarding FAQ-learning and the game classification scheme.

Contents

Chapter 1

Introduction

1.1 Introduction

In single-agent environments modeled as a Markov decision process (MDP) (Bellman, 1957), rational agents, or the individual decision makers optimizing their expected reward, operate in a stationary environment with no interaction with or influences of other agents. In these environments, optimization problems thus solely depend on the characteristics of the environment with possibly some stochastic fluctuations. Because of the stationary nature, these problems are relatively easy to solve (Papadimitriou and Tsitsiklis, 1987) and several learning models have in fact been shown to be guaranteed to converge to optimality (Sutton, 1988; Watkins and Dayan, 1992; Tsitsiklis, 1994).

However, when multiple agents start to operate strategically in the same environment, i.e. actions undertaken by one agent have consequences for the others, the environment turns to become more dynamic and the previous theoretical guarantees are no longer valid (Singh et al., 1994). Additionally, these strategic interactions introduce a much more complex notion of rationality. While in single-agent environments, rational behavior is highly limited to the search for the highest rewarding action, in multi-agent environments, rational agents also have to consider the other agents. Because of this, the concept of rational behavior can often not be determined unambiguously. In fact in some situations, rational behavior in these multi-agent environments can even lead to some paradoxical situations (Basu, 1994).

As an example of such a multi-agent environment with strategic interactions and such a paradoxical outcome, often denoted as a game, consider a real life example by Nisan (2007), called the ISP routing game, about two ISPs sending

Internet traffic to each other. In this game (Figure 1.1), both ISPs have their own network while being able to exchange traffic at exactly two points, being C and S. If now ISP1 needs to send traffic from his starting point s_1 to ISP2's destination point t_1, he basically has two options. In the first case, he opts for the shortest route and uses his own network as long as possible, corresponding with the top route, after which, considering most of his network is used, he receives a zero profit while the other ISP has no costs and ends up with a profit of 5. The second option however corresponds to a more selfish perspective. In this case, he drops his traffic to ISP2 as soon as possible through point C after which the traffic follows a longer path and the situation changes resulting in a profit of 5 for ISP1 and a zero profit for ISP2. It is clear that a rational agent will in this case opt for path 2 as this incurs the least cost for him. However, if now ISP2 at the same time symmetrically has to send traffic from his point s_2 to ISP1's destination point t_2, the exact same reasoning as before applies. Because of the higher load on the network however and since the traffic has to cross each other at point C, both only end up with a profit of 1. This in contrast with the most efficient use of the network which would lead to a total profit of 3 units each. Rational behavior by both ISPs in this case thus does not lead to the most optimal outcome.

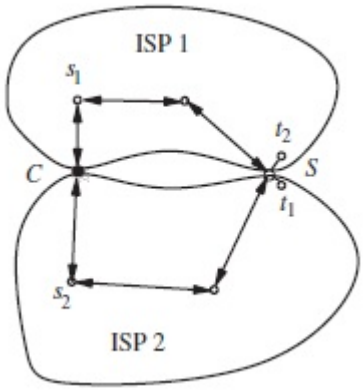


Figure 1.1: The ISP routing game

In this example, we assume the game is only played once, as a one-shot game (Scodel et al., 1959). However while these one-shot events are exemplary, the real interest lies in repeated games as these introduce the notion of learning. Repeatedly playing the exact same game with the same agents, and allowing these agents to use past information can ultimately lead to more informed choices. In the case of the ISP routing game for example, both ISPs might over time start to learn that acting selfish is not the most rewarding option.

Additionally, in these repeated games, these informed choices often lead the players to some sort of stable situation, i.e. an equilibrium. Although several different equilibrium definitions exist, the main concept remains identical, i.e. in an equilibrium, agents have no incentive to change their decision. An everyday example of this repeated play and the existence of equilibria can be found in the price setting strategies of competing companies (Kopalle and Shumsky, 2010). Consider two companies, each selling a virtually identical item at an initial high price. It can then be seen that continuous price updates, driven by the fact that a lower price than the competitor is rewarded with the largest portion of the market, ultimately lead to some stable equilibrium price of which no deviations are possible. After all, higher prices will push customers to the competition while even lower prices will lead to a lower revenue. The difficulty with these equilibria is however that they are sometimes neither unique nor optimal. In this price setting context for example, price fixing, an agreement between two companies, can actually lead to another but higher stable price level.

How agents are able to learn about the game they are playing can differ depending on the set of information there is available and the level of rationality. At one side, we can consider learning rules containing relatively intuitive and simple rules, called adaptive heuristics (Hart, 2005). Although exhibiting rational behavior in the long run, these rules do not perform fully rational. To achieve full rationality, more advanced models like Bayesian learning (Subsection 3.4.3) have to be adopted, explicitly reasoning about prior and posterior probabilities.

However in order for these advanced models to perform fully rational, a great amount of information is needed (Koulovatianos and Wieland, 2011). As in a lot of practical situations, information is often limited, adaptive heuristics are often much more suitable. In fact, several of these adaptive heuristics are even able to make rational choices completely independent of the opponent. Because the limited amount of information requirements and the lack of dependencies on other agents, these fully distributed algorithms (Debbah et al., 2011) are specifically qualified for several real life applications.

As an example of a situation that benefits from fully distributed algorithms, we can consider sensor networks as they are used in habitat monitoring (Heidemann et al., 2006). These networks, consisting of many low costing nodes and corresponding limited memory and power supplies, are often established in remote locations like underwater environments and therefore usually rely on each other to transmit their findings to a central base. In these networks, fully distributed algorithms can aid the different nodes in several ways. Not only in learning the

best neighbor to transmit to but also, taking the sleeping time of the transmitters into account, the best time for this purpose and this despite of their limitations. Additionally, the adaptive nature of these networks, the introduction of new nodes or the breakdowns of older ones, can be captured by these adaptive algorithms.

Adaptive heuristics, although not being able to achieve full rationality, thus have some significant advantages compared to other more advanced models like Bayesian learning, specifically concerning information availability, and this while still being able to perform rational in the long run (Hart, 2005).

1.2 Problem statement

While optimization problems in stationary single-agent environments have been proven to be solvable by rational agents using reinforcement learning techniques like Q-learning (Watkins and Dayan, 1992; Tsitsiklis, 1994), these theoretical proofs have not been successfully extended to more dynamic environments. Particularly in the case of repeated strategic interaction between multiple agents, the concept of rational behavior can become hard to define. To encapture this complex notion of rationality, diverse solution concepts have been identified of which with no doubt the most well known is the Nash equilibrium (Nash, 1951).

However, while some advanced fully rational learning models like Bayesian learning (Subsection 3.4.3) have shown consistent convergence to these Nash equilibria as long as some preconditions are fulfilled (Jordan, 1995), these models are often not applicable as some of the preconditions involve prior knowledge about the opponents. Adaptive heuristics on the other hand, also exhibiting rational behavior in the long run, often have less severe information requirements. However in contrast with Bayesian learning, these models have not theoretically been shown to consistently converge to Nash equilibria.

In this thesis, an empirical analysis will be made of a selection of adaptive heuristics in the context of repeated self-play in 2-player games. The focus will be on the convergence of those learning models with respect to the Nash equilibria of those games. If theoretical results regarding convergence are known, their empirical suitability will be assessed taking, among others, the convergence time into account.

1.3 Methodology

1.3.1 Overview

This section elaborates on the research approach adopted in this work. Following a literature review resulting in a selection of learning rules and normal form games, a test framework has been designed and developed in order to gather and visualize data. This data is analyzed with respect to convergence to solution concepts, speed of convergence and sensitivity to initial conditions. The main focus of this work lies in the short and medium term time periods, periods of at most a few thousand time steps, contrasting most theoretical work which focuses on the long term. However these long term results are also taken into account and are tested for their empirical usefulness.

1.3.2 Theory and conceptualizations

The learning rules and games used in this work are the result of a thorough literature review. The consulted literature, a number of academic articles and books on the matter, can roughly be divided in primary and secondary literature. While the first category gave rise to the ultimate selection of rules and games and their theoretical backgrounds, the secondary literature was mainly used as general background information on learning rules and other aspects of game theory.

Because of the large amount of different rules in the adaptive heuristic segment, a selection of algorithms was made so the final set contained both basic and promising extended versions. These extended versions are all built around a basic version and were originally proposed to enhance the performance of the basic rule. Some of these extensions were added because of conceptual reasons, others were meant to solve weaknesses as is the case for some Q learning variants. To incorporate diversity, every learning rule in the final selection differs from the others, including in underlying strategy.

1.3.3 Test framework

Due to the absence of an existing framework for the testing and comparing of learning models on normal form games and the visualization of the corresponding results, a test framework was developed to satisfy these requirements. The main purpose of this was the ability of data gathering and analyzing. Built using an extendable design to allow for the addition of other learning models and games, the framework offers data management incorporating the three C's, complete, correct and comparable. While complete refers to the availability of a complete data

overview, correct refers to the correctness of the data. To ensure this vital characteristic, several validation steps have been adopted ranging from extended code reviews and test cases over empirical comparisons against expected behavior to deep manual confirmation checks using the generated game data. Comparable then finally refers to the fact that several graphing options have been embedded in the framework to allow for graphical comparisons and overviews along with the ability to save all meaningful experimental data to files in order to allow processing by external tools and applications.

To ensure the randomness of probability choices, a pseudo random generator has been applied. The choice of generator was made in favor of the Mersenne twister (Matsumoto and Nishimura, 1998) because of the high level of statistical randomness and its high performance. This generator has proven its randomness quality in many research experiments over a variety of fields including computational finance (Singla et al., 2008) and cellular biology (Takahashi et al., 2004).

1.3.4 Data gathering, analysis and conclusions

As previous research (Tuyls et al., 2003; Wunder et al., 2010) has shown that convergence behavior of learning models can highly differ depending on the class of the game, our analysis has been made in close relation to the game classification scheme. Using the gathered data, different aspects of the learning rule applications on specific games have been analyzed, most importantly regarding the convergence direction and speed along with specific rule-game combination aspects. In these analyses both graphical representations as the raw data saved in data files have been consulted.

One of the most useful performance metrics in this context is the average outcome convergence rate, averaged over several different runs. Defined, over time, as the percentage of runs that are at that time period at that specific outcome, or otherwise stated as the probability of reaching that specific outcome at that time, this metric can provide a strong indication of equilibrium convergence and speed. Depending on the nature of the strategic behavior, this metric has been combined with several other data evolution plots among which are the internal data values (e.g. Q values in case of Q learning) and the empirical action distributions per player over time.

As for the sensitivity to initial conditions, a selection of parameters was made reflecting who were most likely to inhibit this behavior. Both theoretical work and empirical results were used in this determination process. The effects of these

parameter changes were then analyzed by setting the remaining learning rule parameters to constant values.

In all these processes, also a quantitative aspect has been applied. This aspect, reflected in the number of independent iterations, has been used because of random fluctuations causing different behavior. No analysis nor conclusion has been made or drawn based on a single run. Over the set of experiments, the minimum number of runs has been set to 30. However, because of computational possibilities, for experiments under 5000 time periods, this number has been increased to 1000 independent runs. To remove any statistical dependency between the different players, each player has been assigned a randomly generated initial seed to use in their own random generator instance.

1.4 Thesis structure

In Chapter 2, the essential game theoretic background related to this research is introduced. Covering the concepts of player strategies, game representations, the notion of repeated play and solution concepts, the chapter ends with a game classification scheme along with a set of sample games.

Chapter 3 then elaborates on the different learning rules. Together with an introduction of the set of rules used in the analysis phase, a short overview is also supplied about other related learning rules.

In Chapter 4, the results of the experiments are introduced and analyzed. While focusing on convergence behavior and sensitivity to initial conditions, attention is also payed to memory requirement differences and to which extent the theoretical properties are to be considered empirically desirable. For these experiments, the sample games introduced in Section 2.5 are used.

Chapter 5 completes this document with conclusions related to the aforementioned analysis.

Chapter 2

Game theory

This chapter introduces the fundamentals of game theory. In the first section, the basic concepts of the game considering players and their actions are introduced, followed by an elaboration on the difference between one-shot and repeated play in which the concept of learning is introduced. Subsequently, after an overview of the two commonly used game notation forms and how they relate, a section on the most important solution concepts is provided. The chapter ends with a section on the classification of the game space into a number of distinct classes.

2.1 Rules of the game

Game theory is concerned with strategic interactions, situations where two or more rational players, or decision makers, are interacting and their actions are not only affecting themselves but also the others. Considering the ISP routing game, each ISP's choice affects the use of the other's network thereby inducing additional costs for that other ISP. Both can thus be considered to be in a strategic interaction.

Definition (Players)**.** *Players, or agents, are the individual decision makers. The aim of a player is to maximize his expected payoff. N denotes the set of all players.*

Definition (Action)**.** *A player's action a_j is a possible choice he can make. Payer i's action set A_i is the set of all his possible actions.*

Definition (Payoff)**.** *Player i's payoff $u_i(a_1, \ldots, a_n)$ is the reward he receives after each of the n players, including player i, have made their choice a_j.*

Interactions as these can formally be modeled as games with certain characteristics regarding players, actions, payoffs and information. These four concepts are sometimes collectively called the rules of the game (Fudenberg and Levine, 1993). In these games, the players, the individual decision makers who's main aim is to maximize their own payoff, each have a set of options to choose from, denoted as actions. When each player has selected an action, the set of all these actions is then used to determine for each player his resulting payoff, also denoted as reward or utility (von Neumann and Morgenstern, 1944). How a player makes his decision out of the possible actions depends on the strategy, this strategy shows what to do at any point in the game. The set of the strategies of all players is called the strategy profile.

Definition (Pure strategy)**.** *Player i's pure strategy s_i is a rule that defines which action is to be taken at the next decision point in the game.*

Definition (Mixed strategy)**.** *A mixed strategy s_i is a probability distribution over all pure strategies. The set of all possible strategies is denoted by S_i.*

Definition (Strategy profile)**.** *A strategy profile $s = (s_1, \ldots, s_n)$ is an ordered set consisting of the individual strategies of all players in the game.*

Definition (Strategy payoff)**.** *Player i's payoff of profile s can be calculated as the expected values of the corresponding pure strategy profiles taking the probability distribution into account. Player i's payoff can be denoted as $u_i(s_1, \ldots, s_n)$ or shorter as $u_i(s_i, s_{-i})$, where s_{-i} refers to the other players' strategies.*

The strategy will thus comprise the strategic behavior of a player. Depending on the number of actions a player considers, two types of strategies can be distinguished. In pure strategies, only one action is considered in the final decision, this action is thus always chosen with 100% certainty. In mixed strategies (von Neumann and Morgenstern, 1944) on the other hand, several actions are to be considered. For this purpose, to each of the considered actions, a probability is assigned and the final choice is made randomly proportionally to these probabilities. Because of this, a pure strategy can be considered a special case of a mixed strategy in which one action is assigned a 100% probability and the others a zero probability. Similarly, a mixed strategy can be seen as a probability distribution over pure strategies. In the earlier mentioned ISP routing game for example, a sample pure strategy could be to always use the own network as long as possible. A mixed strategy on the other hand is then for example to use the own network (action A1) 75% of the time and send the traffic to the other ISP in 25% of the time (action A2).

To denote strategies, the combination of probabilities and actions can be used. The earlier mixed strategy example can for example be denoted by $(0.75A1 + 0.25A2)$. Similarly, the strategy profile is denoted as the combination of strategies for each player. For this example, if both ISPs use the same strategy, the strategy profile becomes $(0.75A1 + 0.25A2, 0.75A1 + 0.25A2)$

2.2 One-shot vs Repeated play

While playing a game once can already reveal certain characteristics, concepts as learning, teaching and leniency only come into play in case of repeated play. In these repeated games, or iterated games, the same game is played several times after each other using the exact same players. Additionally these players have access to information about the game history allowing them to learn from this history.

A fundamental difference between one-shot and repeated play is the need for an immediate well rewarding outcome versus the striving for long term results. As in one-shot play, the game is only played once, the entire goal is to maximize the resulting payoff of that one iteration. Since each player has this same goal, high levels of rationality and information play a very important rule in order to pick the optimal action. Players will try to use as much information as possible and take the potential strategies of the opponents into account in order to outsmart them when making a decision.

As an example of a possible one-shot game is what is known as the Microsoft-Google game (Figure 2.1 due to Gintis, 2009). This game is also known as the Microsoft-Netscape game (Gintis, 2000) using the same story but different actors and illustrates a possible decision choice both Microsoft and in this case Google have to make in the development process of their Web browsers. During this development, both companies have to choose a supporting platform being either ActiveX (being Microsoft's preference) or Java (Google's preference). However since both Microsoft and Google also prefer to be fully compatible with each other, they will both strive for an identical choice, making the acceptable outcomes either (A,A) or (J,J). As the stakes are high and no corrections are possible afterwards, both companies will do everything to make sure they make the correct and best rewarding decision.

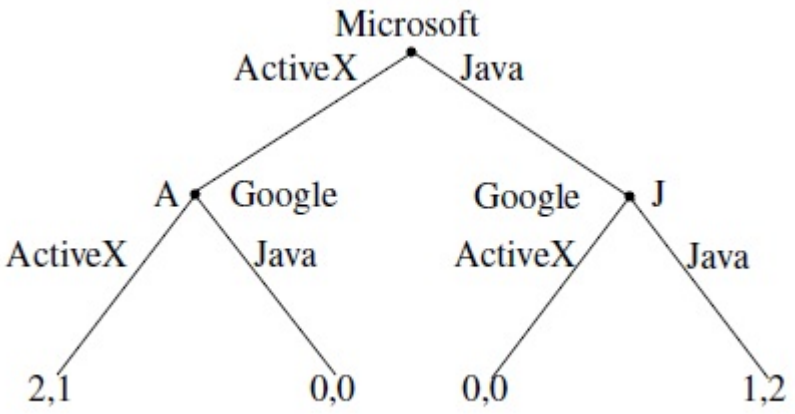


Figure 2.1: The Microsoft-Google game

Another situation arises in repeated play where the main goal is a high average reward over time. As players here have access to information about the past, the concept of learning comes into play. Players will be able to learn about the well rewarding actions and the opponent's behavior in order to continuously improve their strategy. Additionally besides learning, the notion of teaching can sometimes be applied. In certain games, a player can use this to teach the opponent that one action might be mutually beneficial and to encourage him to adopt that strategy. An often used strategy to achieve this is to repeatedly play that mutually beneficial action until the opponent adopts it too. Initial other actions of the opponent are then forgiven in favor of the superior average reward over time. This behavioral strategy covering forgiveness is called leniency.

A situation where these concepts can aid can be found in the Prisoner's dilemma (Figure 2.2, Hardin, 1787). In this game, two prisoners, Frank and Jesse, have been caught in a burglary and they are given the chance to confess or deny. If both confess, they both get 5 years, if they both deny, they only get 2 years due to lack of evidence and otherwise the denier gets 10 year of prison and the confessor goes free because of his collaboration. While the strategy 'Confess' is in fact always a better choice than 'Don't confess', being a dominant strategy (Section 2.4), resulting in the (Confess, Confess) outcome, the situation in which they both don't confess is ultimately better. If this game is thus played repeatedly, the best outcome is that both players eventually learn to play 'Don't confess' instead of the strategy 'Confess'. By applying the notion of teaching, i.e. if Frank decides to play the 'Don't confess' action repeatedly in order to persuade Jesse to do the

same, both might ultimately end up in this superior better rewarding state.

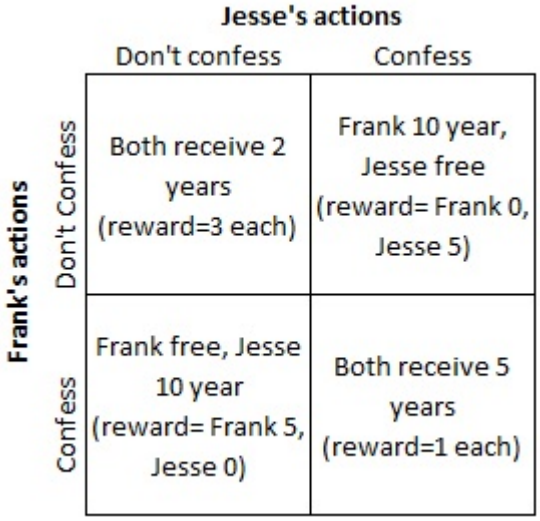


Figure 2.2: Prisoner's dilemma

This example also introduces the notion of outcome. As visible in Figure 2.2, the game contains four possible states depending on what Frank and Jesse choose to do. Given their decisions, one of these states is the outcome of the game. In the case both decide to confess, the outcome becomes (Confess, Confess) and both receive 5 years in jail.

Definition (Outcome)**.** *The combination of each player i's chosen action a_i results in an outcome of the game. Sometimes also denoted as state.*

2.3 Game representation

As has already been demonstrated in Figures 2.1 and 2.2, different game representation methods can be used. The most familiar one is the normal form representation (von Neumann and Morgenstern, 1944), also denoted as the strategic form. In this form, a game with n players is represented using an n-dimensional matrix notation where each row and column corresponds with an action and the internal cells contain the payoffs for the different players. As most other representations are completely equivalent to this notation and can be reduced to it, the normal form notation has been argued to be the most fundamental of all.

While the normal form is generally the most dominantly used, the extensive form has also proven its usefulness. The most significant difference is the fact that the extensive form explicitly embeds the notion of action sequence. Being represented as a tree with the nodes being choices and the edges being actions, an entire action sequence can be tracked throughout the tree in order to ultimately arrive at the resulting payoff. As the extensive form can be viewed as a generalization of a decision tree (Fudenberg and Tirole, 1991), each payoff node uniquely corresponds to a specific action sequence.

While all games in extensive form representation have been known to be transformable into normal form representation, the transformation might cause an exponential growth of the number of rows and/or columns in the payoff matrix. Because of this, while normal form representation is most suitable in general, the extensive form notation might be the preferred choice for games where the action sequence plays an important role. As an example of both representations, Figure 2.3 shows a game in extensive form and an equivalent normal form representation.

	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3, 8	3, 8	8, 3	8, 3
(A,H)	3, 8	3, 8	8, 3	8, 3
(B,G)	5, 5	2, 10	5, 5	2, 10
(B,H)	5, 5	1, 0	5, 5	1, 0

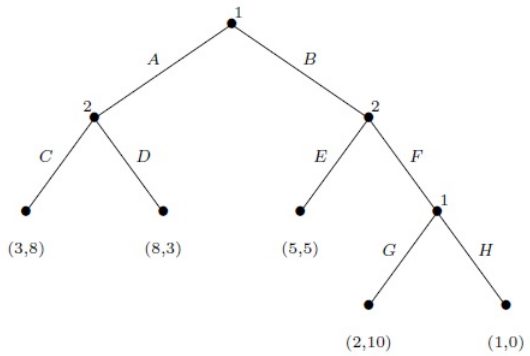


Figure 2.3: A 2 player game, both in normal (l.) and in extensive (r.) form

The normal form representation is also commonly used to define solution concepts (Section 2.4) in, including the well known Nash equilibrium (Subsection 2.4.3). However, because of the inherent notion of action sequence, some Nash equilibria are found to be irrational when represented in the extensive form (Gintis, 2009, p57).

2.4 Solution Concepts

2.4.1 General concept

Solution concepts are formal rules which allow to identify interesting subsets of game solutions. These subsets correspond to game outcomes that are desirable in one way or another. As many different outcomes can be classified as desirable, depending on the point of view, several types of solution concepts are currently in existence. The most widely used and well known equilibrium concept is the Nash equilibrium (Nash, 1951).

As described by popular dictionaries, an equilibrium is a state of rest and balance. Reflected on the field of game theory, a game equilibrium concerns an outcome in the game where no player has an incentive to change strategy. The study of equilibrium concepts has close relations with the field of economics (Myerson, 1999) due to the many applications in economic situations where the aim is to come to a stable but payoff-maximizing state.

In this section, three of these equilibrium concepts are introduced, selected on their general importance and use in further research. These are the Nash equilibrium (NE), correlated equilibrium (CE) and coarse correlated equilibrium (CCE). Other existing equilibrium concepts include the sequential equilibrium (Kreps and Wilson (1982)), the (trembling hand) perfect equilibrium (Selten, 1975) and the proper equilibrium (Myerson, 1978). Several of these others can be seen as refinements of the Nash equilibrium while others follow a different strategy. They are however not as widely used and not always applicable to all games.

Besides equilibrium concepts there are also some other solution concepts. The most well known of these might very well be the dominant strategy solution and iterated strict dominance which are both built upon the existence of dominant strategies and the belief that rational players ignore strategies that are always inferior compared to some other strategy. However, although still interesting for learning, because of the limited applicability due to specific requirements to the payoff structure of games, their use in practice is limited.

Finally it should be mentioned that the application of most of the solution concepts results in a set of several solutions making the exact end result uncertain. Additionally, equilibrium concepts do not guarantee that the end state is the state with the highest payoff, i.e. that there is no state that dominates it, in fact they are only guaranteed to conform to some definition of stability.

2.4.2 Dominant strategies

As a player can have multiple possible strategies, concepts have been developed to compare and eliminate strategies. One of the most popular is the one of best reply (Nash, 1953), also called best response. The best reply strategy is the strategy that, given the opponents' strategies, yields the highest payoff. Such a strategy is thus the best option for a certain player if he knows the behavior of the opponents.

Definition (Best reply)**.** *Player i's strategy s_i^* is a best reply strategy if, given the opponents' strategies s_{-i}, the resulting payoff is higher than or equal to any of his other strategies s_i'. Or mathematically:*
$u_i(s_i^*, s_{-i}) \geq u_i(s_i', s_{-i}), \forall s_i' \neq s_i^*$
If the inequality is strict, the strategy can be called a strongly best reply.

A stronger form of best reply is the notion of dominant strategy. This strategy is one that is superior to every other strategy, no matter what the opponents play. It will thus always yield a higher payoff, even when the behavior of the opponents is unknown. Those strategies that are strictly inferior to some other strategy are subsequently called dominated strategies.

Definition (Dominant strategy)**.** *Player i's strategy s_i^* is a dominant strategy if regardless of the opponents' strategies s_{-i}, the resulting payoff is higher than any of his other strategies s_i'. A dominant strategy is a best reply strategy regardless of the opponents' strategies. Or mathematically:*
$u_i(s_i^*, s_{-i}) > u_i(s_i', s_{-i}), \forall s_{-i}, \forall s_i' \neq s_i^*$

While the presence of dominant strategies is game dependent, best reply strategies can be found in every game with finite players and strategies (Nash, 1953; Gibbons, 1992). As an example consider a simple heads or tails game (Figure 2.4) called matching pennies (Green, 1859; von Neumann and Morgenstern, 1944). Two players each, at the same time, either select heads or tails. If both choose the same, player I gets a reward, in the other case player II gets a reward. In this game, the best response action of player I is always to choose the same as the opponent does, while the best response strategy of player II relies on choosing a different action than the opponent. As the best response strategy changes if the opponent changes strategy, no dominant strategy can be found here. For both players, a mixed strategy might for example be a randomized $50/50$ distribution for heads or tails. Similarly, a pure strategy for both might be to always choose heads although this would result in player B never winning. Since no strategy is always better than the other, the matching pennies game does not have a dominant strategy.

		II	
		Heads	Tails
I	Heads	1,0	0,1
	Tails	0,1	1,0

Figure 2.4: Matching pennies game, a game without dominant strategies

In games with dominant strategies though, like the prisoner's dilemma (Figure 2.5) where the strategy D is dominant, the technique of iterated dominance can be applied. This technique removes those strategies that are dominated by another strategy, in this case strategy C, in order to end up in the outcome (D, D). This outcome can then be seen as the solution of the game since no rational player will ever choose the dominated strategy C instead of the, always better rewarding, strategy D.

		II	
		C	D
I	C	3,3	0,5
	D	5,0	1,1

Figure 2.5: Prisoner's dilemma, a game with a dominant strategy D

2.4.3 Nash equilibrium (NE)

The most leading and well known solution concept is no doubt the Nash equilibrium (NE), named after John Forbes Nash who defined the mixed strategy Nash equilibrium for all non-cooperative games (Nash, 1951), i.e. games where the players make decisions independently of each other, although the basic ideas behind it date back to at least Cournot (1838). A Nash equilibrium can be defined as a strategy profile in which no player can improve his situation by changing his strategy. Otherwise stated, a Nash equilibrium is a situation in which each player plays his best reply strategy against the other player(s). Formally this can be defined as follows:

Definition (Nash equilibrium)**.** *A strategy profile s^* is a (nonstrict) Nash equilibrium if, for all players i:*
$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*)$*, for all* $s_i \in S_i$

A Nash equilibrium is pure if the underlying strategy profile s^* consists of pure strategies, otherwise it is called a mixed equilibrium. When a strict inequality applies, the strategy profile s^* is a strict Nash equilibrium.

An example of a Nash equilibrium, Figure 2.6 is provided. This game, called battle of the sexes (Subsection 2.5.1), contains three different Nash equilibria, 2 pure and 1 mixed. While in the figure the pure Nash equilibria are marked, the mixed equilibrium is located in $(1/3LW + 2/3WL)$ for the row player and $(2/3LW + 1/3WL)$ for the column player resulting in a payoff of $2/3$ each. It can easily be confirmed that these are in fact Nash equilibria as none of the players can unilaterally change their strategy to receive more payoff.

	LW	WL
LW	**2, 1**	0, 0
WL	0, 0	**1, 2**

Figure 2.6: Battle of the sexes (BoS), with two pure NE and one mixed NE

An element that contributes to the importance of the Nash equilibria is the fact that every non-cooperative game has been proven to have at least one (non-strict) NE, given that the number of players and strategies are limited (Nash, 1951). Additionally, von Neumann and Morgenstern (1944) have shown that if a game has a uniquely rational solution, then that solution is with no doubt also a Nash equilibrium (although Nash only formally defined the NE concept a few years later). Although the opposite of this last statement does not hold and the uniqueness of a Nash equilibrium solution is often not guaranteed, the Nash equilibrium does allow for a solid base of predicting optimal play. Finally, to relate with the technique of iterated dominance, for games in which each player has a dominant strategy as for example the prisoner's dilemma, the outcome reached by using iterated dominance is in fact the Nash equilibrium (NE) of the game.

2.4.4 Correlated equilibrium (CE)

While the two pure NE in the previous game (Figure 2.7) are highly acceptable solutions, they do require a form of coordination between both players. Surely,

if a player does not know which NE the opponent is going for, his own choice cannot be made with full certainty. Following the mixed NE proves to be a fair solution for this, however because of the lack of real coordination between the two players, the zero rewarding outcomes (WL,LW) and (LW,WL) are to be visited as well, thereby decreasing the average expected payoff to only $2/3$.

	LW	WL
LW	$\boxed{2,1}$	$0,0$
WL	$0,0$	$\boxed{1,2}$

Figure 2.7: Battle of the sexes (BoS), with two pure NE and one mixed NE

The correlated equilibrium (CE) (Aumann, 1974), a generalization of the NE and by some to be considered the most fundamental of all (Shoham and Leyton-Brown, 2009), offers a solution by introducing a form of correlation between the different players. In the example of Figure 2.7, imagine that both players are able to see a random coin flip. If they now base their strategies on the result of this flip, they are able to construct more advanced and coordinated strategies. A possible equilibrium strategy for both players could then be: "play LW if heads and play WL if tails". It can be verified that any deviation from this strategy yields a loss in utility. Additionally, using this strategy, the zero rewarding outcomes are avoided and the expected payoff becomes $1.5(50\% \times 2 + 50\% \times 1)$, higher than the mixed NE before.

Although in this case, the result of the random event (i.e. the coin flip) was made public to both players, this is in general not necessary. In general it suffices to have a random event with known probability to deliver private signals to all players about its value. A formal definition by Shoham and Leyton-Brown (2009) is to consider n random variables, one for each player, and a joint distribution over them. The random event then only informs each player the result of his variable thereby allowing him to condition his action on that value.

Definition (Correlated equilibrium)**.** *A correlated equilibrium is defined as a tuple* (v, π, σ) *where* v *is a tuple of random variables* $v = (v_1, \ldots, v_n)$ *with respective domains* $D = (D_1, \ldots, D_n)$*,* π *is a joint distribution over* v*,* $\sigma = (\sigma_1, \ldots, \sigma_n)$ *is*

a vector of mappings to actions $\sigma_i : D_i \rightarrow A_i$, and for each player i and every mapping $\sigma'_i : D_i \rightarrow A_i$, the following holds:

$$\sum_{d \in D} \pi(d) u_i(\sigma_1(d_1), \ldots, \sigma_i(d_i), \ldots, \sigma_n(d_n)) \geq \sum_{d \in D} \pi(d) u_i(\sigma_1(d_1), \ldots, \sigma'_i(d_i), \ldots, \sigma_n(d_n))$$

Or put in similar short notation as NE:

$$\sum_{d \in D} \pi(d) u_i(\sigma_i(d_i), \sigma_{-i}(d_{-i})) \geq \sum_{d \in D} \pi(d) u_i(\sigma'_i(d_i), \sigma_{-i}(d_{-i}))$$

In this definition, a weighted sum is taken of all possible outcomes $d \in D$ in order to compare the expected utility of selecting the suggested action $\sigma_i(d_i)$ (at the left) against the utility of adopting another action $\sigma'_i(d_i)$ (at the right). A certain tuple (v, π, σ) is thus a CE if no player has an incentive to deviate from the action suggested to him, assuming the others also follow their suggestion $\sigma_{-i}(d_{-i})$ (Osborne and Rubinstein, 1994).

Because of the suggestion, a correlated equilibrium is a strictly weaker notion of equilibrium compared to the Nash equilibrium. In fact, the set of correlated equilibria contains the set of Nash equilibria as each NE is automatically a CE in which the players' signals and choices are probabilistically independent (Hillas et al., 2007).

Although the correlated equilibrium is weaker than the Nash equilibrium, it does have the major advantage of being computationally easier to calculate. While computing Nash equilibria is still considerably hard, especially when the number of players increases (PPAD-complete, see Daskalakis et al., 2006), computing a correlated equilibrium has been shown to be solvable in polynomial time (Papadimitriou and Roughgarden, 2008). Additionally, many learning models are known to converge to correlated equilibria rather than the stricter class of Nash equilibria (e.g. ICRM at Chapter 3). CE thus also has empirical importance.

2.4.5 Coarse correlated equilibrium (CCE)

Another solution concept using the notion of correlation is the coarse correlated equilibrium (CCE) (Moulin and Vial, 1978). As with the correlated equilibrium, consider again a third party random event sending private signals to each player thereby suggesting actions to play. Before the signals are sent however, each player is given the chance to opt out, thereby allowing himself to play any action he prefers. In the case he does not opt out, he is obliged to play the action

suggested to him by the random event. A certain distribution is then a CCE if no player has an incentive to opt out, assuming the others don't opt out either. Formally this amounts to:

Definition (Coarse correlated equilibrium)**.** *A coarse correlated equilibrium is defined as a tuple (v, π, σ) where v is a tuple of random variables $v = (v_1, \ldots, v_n)$ with respective domains $D = (D_1, \ldots, D_n)$, π is a joint distribution over v, $\sigma = (\sigma_1, ...\sigma_n)$ is a vector of mappings to actions $\sigma_i : D_i \to A_i$. π_{-i} is the marginal probability that the (opponents') action tuple a_{-i} is played. Then for each player i and every action $a'_i \in A_i$, the following holds:*

$$\sum_{d \in D} \pi(d) * u_i(\sigma_i(d_i), \sigma_{-i}(d_{-i})) \geq \sum_{a_{-i} \in A_{-i}} \pi_{-i}(a_{-i}) * u_i(a'_i, a_{-i})$$

In which the marginal probability $\pi_{-i}(a_{-i})$ that the action tuple a_{-i} will be played is defined as:

$$\pi_{-i}(a_{-i}) = \sum_{a'_i \in A_i} \pi(a'_i, a_{-i})$$

In this adapted version of the CE definition, the expected utility of opting in (on the left) and that of opting out (on the right) are compared to each other. By comparing this definition to the CE one, because the left part of the CCE equation can become equal to the left part of the CE equation, it can be seen that every correlated equilibrium is actually also a coarse correlated equilibrium (Young, 2004). Additionally it should be noticed that in fact the amount of correlation does not differ between CE and CCE. The main difference lies in the moment players are allowed to opt out. In fact, if the focus is on games with just two actions, the set of correlated equilibria is found to be exactly the same as that of coarse correlated equilibria (Young, 2004).

Although CCE is a less widely used concept, its value lies in the fact that certain learning models on the notion of regret minimization have theoretically been proven to converge to the set of CCE (e.g. RM in Chapter 3). Although this form of convergence is rather weak in comparison with Nash convergence or even CE convergence, it can nevertheless be considered a theoretical advantage against learning models with no guaranteed convergence. Chapter 3 introduces models of both categories.

2.5 Game classification

This section introduces a game classification scheme to subdivide the set of 2x2 normal form games, games with two players each having two strategies, into groups with different Nash equilibrium characteristics. Besides this first classification, a second often used classification scheme is shortly presented along with the differences with the first scheme. The section ends with a classification of a number of selected normal form games along with their Nash characteristics. For the representation, the aforementioned normal form representation will be used (Figure 2.8).

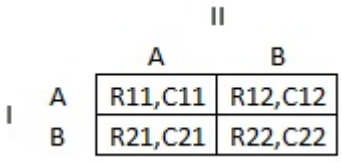


Figure 2.8: Normal form game representation

The classification restriction to 2x2 games instead of the general class of mxn games is motivated by the fact that these have a remarkable illustrative strength regarding diverse strategic interactions (Robinson and Goforth, 2005). Additionally, the class of 2x2 games has already undergone the most research, both from a theoretical and empirical point of view and has shown its strength in many problems and paradoxes in the field like economics and political science (Barany et al., 1992; Wang and Yang, 2003). Extending the classification to general mxn games would only contaminate the general overview. However the general ideas behind the classification should also be able to be used in a more broader class of games.

While the class of 2x2 games knows many, sometimes overlapping, subclasses like zero-sum games, grouping games with similar characteristics, only two classification schemes are generally used to strictly subdivide the entire class of 2x2 games. The first of these is a classification scheme that classifies games depending on their Nash equilibrium characteristics. The second one on the other hand focusses on the internal payoff relations. Both of these are introduced in this section.

2.5.1 Classification by Nash equilibria

The classification used in this work, relies on the number and type of Nash equilibria that can be found in a game. The base classification, introduced by Vega-Redondo (2003, p403) divides the field of games into three major classes, thereby assuming no payoff ties are possible to avoid the classes from overlapping. The main advantage of this classification is the fact that by dividing the games by their Nash characteristics, a learning model's convergence behavior towards these Nash equilibria might be the same for all games in the same class.

However, as discovered by Wunder et al. (2010), convergence behavior of games in the same class can with this base classification still differ. This is caused by the fact that Nash equilibria are not necessarily the most rewarding outcome regarding payoffs, i.e. there might be other outcomes that deliver higher payoffs to all players. To solve this, the initial classification is deepened with one more level with the introduction of two mutual exclusive subclasses (Wunder et al., 2010) depending on the existence of a better rewarding dominant non-Nash state. Formally, using the payoff distribution in Figure 2.8, this becomes:

Definition (Subclass A)**.** *Games in this subclass have no non-Nash outcome with higher payoffs for both players than their (lowest rewarding) Nash equilibrium. Every game not in subclass B is a subclass A game.*

Definition (Subclass B)**.** *Games in this subclass have a non-Nash outcome with higher payoffs for both players compared to the (lowest rewarding) Nash equilibrium. Mathematically, this relates to the following condition:* $\exists i, j : R_{ij} > R_N$ *and* $C_{ij} > C_N$ *where* R_N *and* C_N *either stand for the unique Nash payoffs for the row and column player or for the lowest Nash payoffs for the players if multiple exist. The non-Nash outcome can be referred to as the Nash dominating outcome.*

Games of subclass B contain a non-Nash state that has a higher payoff for both players than the (lowest rewarding) Nash equilibrium in that game, while this is not the case for subclass A games. For 2x2 games, the mention of the lowest rewarding Nash equilibrium only applies to games with multiple Nash equilibria (Class 2). In general, this subclass B condition states that there exists an non-Nash outcome which can be preferred to at least one Nash equilibrium regarding the payoff characteristics, causing for some learning models possibly a non-Nash convergence. Subclass A games on the other hand have no such outcome. For these games, no Nash outcome is dominated by another outcome resulting often in a more straightforward Nash convergence. Empirically, games of this B subclass are particularly interesting because it can be argued that the most rational choice

in an iterated game is to choose that better rewarding non-Nash state instead of the Nash equilibrium.

These two subclasses are applicable for all three subsequent game classes: dominance games, coordination and contribution games (Rasmusen, 2007) and discoordination games (Rasmusen, 2007).

Definition (Class 1: Dominance games)**.** *Games falling in this category have at least one player with a dominant strategy. Mathematically, this relates to the following condition:* $(R_{11}-R_{21})(R_{12}-R_{22}) > 0$ *and/or* $(C_{11}-C_{12})(C_{21}-C_{22}) > 0$*. Games in this category have one pure Nash equilibrium.*

In this categorization, the first class includes those games where each player has a dominant strategy. In this case, the Nash equilibrium can easily be derived by removing all the dominated strategies as no rational player could ever favor them. However, this class also incorporates a larger collection of games in which just one player has a dominant strategy (Tuyls et al., 2003). In these games, the Nash equilibrium is composed by the dominant strategy of the one player together with the best reply action of the other player. Disregarding any payoff equalities, this results again in a single pure Nash equilibrium.

As a subclass A example of this class, consider the Deadlock game (Figure 2.19). In this game, both players have a dominant strategy in the form of action D. Additionally, this dominant state (D, D) is also the mutually most beneficial state as there is no other state in which both players generate more payoff.

		II	
		C	D
I	C	1,1	0,3
	D	3,0	2,2

Figure 2.9: Example of class 1A classification: The Deadlock game

To illustrate the 1B class of game, the Prisoner's dilemma can be used. In this game (Figure 2.10), built around the story of two caught burglars and already introduced in Section 2.2, each player has an obvious dominant strategy in the form of the D(efect) action resulting in the single pure Nash equilibrium (D, D).

However, this Nash outcome is not the most rewarding outcome in the game. In fact, the non-Nash outcome (C, C) actually rewards both players with a higher payoff. The prisoner's dilemma can be seen as almost identical to the ISP routing game introduced in Section 1.1. In both cases, the most rewarding outcome $(3, 3)$ can only be reached if both work together (i.e. don't confess and use the own network), but where in both games the more 'greedy' action (i.e. confess and use the other network) dominates resulting in payoffs of only 1 each.

		II	
		C	D
I	C	3,3	0,5
	D	5,0	1,1

Figure 2.10: Example of class 1B classification: Prisoner's dilemma

Definition (Class 2: Coordination and Contribution games)**.** *Games falling in this class have no dominant strategies. Instead they have two pure NE at opposite sides of the payoff matrix. Mathematically, this relates to the following condition:* $(R_{11} - R_{21})(R_{12} - R_{22}) < 0$ *and* $(C_{11} - C_{12})(C_{21} - C_{22}) < 0$ *and* $(R_{11} - R_{21})(C_{11} - C_{12}) > 0$*. Games in this category have two pure and one mixed Nash equilibrium.*

In contrast with class 1 games, class 2 games do not have dominant strategies. Instead they have two pure Nash equilibria at opposite sides of the payoff matrix. However, as players do not know which equilibrium the opponent will play, the game also contains a mixed equilibrium, corresponding with the fact that both players will switch between both equilibria.

While coordination games have their equilibria in the top left and bottom right corner of the payoff matrix, reflecting the need for some coordination between both players to both select the same action, contribution games have their equilibria in the opposite two corners. This structure reflects to situations in which players have to contribute to some public good but they both like the other to contribute the most. However, although the structures of these games appear different, mathematically they are essentially the same as the locations of the equilibria can simply be changed by switching the order of an agent's actions in the payoff matrix. They are therefore seen as one class of games.

The most famous example to represent the subclass A class of these games is the ”Battle of the Sexes” game (BoS) (Luce and Raïffa, 1957). Consider a couple wanting to go out for the evening, both preferring each other's company. However there is no agreement on the location of the date. The husband prefers the football game while the wife prefers the opera. Put into normal form representation, we find the following payoff matrix (Figure 2.11). This game is sometimes also referred to as the ”Bach or Stravinsky” game (BoS) in which case the background story involves two different concerts by two different composers.

Still preferring each other's company, it can be seen that both husband and wife aim to choose the same action as the other. This results in two pure Nash equilibria (Opera-Opera and Football-Football) in which no one will try to change his action as that would mean going alone. Additionally also a mixed equilibrium can be found. In this case, this is the one in which the players choose their preferred choice 60% and the other choice 40% of the times, for both players resulting in an average payoff of $6/5$. It can be mathematically confirmed that any deviation from this strategy yields a decrease in payoff.

		Wife	
		Opera	Football
Husband	Opera	2,3	0,0
	Football	0,0	3,2

Figure 2.11: Example of class 2A classification: Battle of the sexes (BoS)

To illustrate the subclass B class of these games, an almost equally famous game as BoS can be used, the chicken game (Sermat and Gregovich, 1966). In this game, two drivers drive towards each other on a narrow road while they are given two choices, either dare and continue driving or chicken out and swerve to avoid a collision. If they both dare, a collision is unavoidable resulting in no payoff at all. However if only one of them chickens out, he loses the game and is called a chicken, hence the name of the game. It is only when both players chicken out, that they still receive acceptable payoffs. The corresponding payoff structure can be seen in Figure 2.12.

In this game, both combinations of C(hicken) and D(are) are pure Nash equilibria. Additionally, as with BoS also a mixed equilibrium is in place in which both

players differ their actions. In this case they will both play C and D respectively $2/3$ and $1/3$ of the time, each resulting in a payoff of $14/3$. However interesting about this game is the fact that the non-Nash outcome (C, C) actually delivers a higher payoff for both players than their mixed equilibrium and thereby differing from the BoS example in which there was no other outcome in which both players could receive more payoff.

		II	
		C	D
I	C	6,6	2,7
	D	7,2	0,0

Figure 2.12: Example of class 2B classification: Chicken game

Although the chicken game is also known as the hawk-dove game, the use of the terminology often depends on the field of application. Whereas the chicken game is prevalent in political science (Russell, 1959; Deutsch, 1968) and economics (Lipnowski and Maital, 1983), the hawk-dove game is usually used in biology (Wolf et al., 2007) and evolutionary game theory (Sigmund and Nowak, 1999).

Definition (Class 3: Discoordination games)**.** *Games falling in this class again have no dominant strategies. However they also lack pure Nash equilibria. Mathematically, this relates to the following condition:* $(R_{11} - R_{21})(R_{12} - R_{22}) < 0$ *and* $(C_{11} - C_{12})(C_{21} - C_{22}) < 0$ *and* $(R_{11} - R_{21})(C_{11} - C_{12}) < 0$*. Games in this category one mixed Nash equilibria.*

Discoordination games are characterized by their unstable structure. As there are no dominant strategies and no pure Nash equilibrium, there is not a single state in which both players have no incentive to deviate, creating an eternal changing behavior. As these games lack pure Nash equilibria, only a mixed equilibrium is in place (Vega-Redondo, 2003).

An example of the 3A class of games is the matching pennies game (Figure 2.13). In this game, two players each choose either heads or tails. If both choose the same, player 1 receives a coin, in the other case player 2 receives one. The game is a clear example of a 3A game as whatever happens, there will always be one player that is unhappy with the outcome. Additionally, the mixed equilibrium

of $50/50$ delivers a 0.5 payoff to both, which makes that there is no other state that delivers more payoff to both players.

		II	
		C	D
I	C	1,0	0,1
	D	0,1	1,0

Figure 2.13: Example of class 3A classification: Matching pennies game

A generalization of the matching pennies game can be found in Shapley's game (Figure 2.14, Shapley, 1964). This famous game , named after the mathematician and economist Lloyd Shapley is well known since Shapley (1964) showed that fictitious play does not converge to a Nash equilibrium for that game. Although this game has 3 actions for each player, it can still be classified as a 3A game because of the existence of just 1 mixed equilibrium without any pure equilibria.

		II		
		A1	A2	A3
I	A1	0,0	1,0	0,1
	A2	0,1	0,0	1,0
	A3	1,0	0,1	0,0

Figure 2.14: Example of class 3A classification: Shapley's game

While for these two games there is no outcome dominating the (mixed) Nash equilibrium, this not the case for the spoiled child game (Wunder et al., 2010), therefore being an example of the 3B classification. In this game, a father and his spoiled child are interacting with each other in a simple scenario. While the father can either go for P(unish) or S(poil), the child has two other options in the form of B(ehave) and M(isbehave). This leads to the payoff structure in Figure 2.15 containing four different payoff combinations but rather unstable behavior. In fact, in every state, either the parent or the child will want to change to his other action in order to receive more payoff.

This unstable behavior results in a mixed Nash equilibria, which is, in contrast with the previous games, not the same for both players. While the father will choose both actions with a 50% possibility, the child will choose his B option $2/3$ of the time and his M option $1/3$ of the time. This results in an average expected payoff of $2/3$ and $3/2$ for the father respectively for the child. As can be observed, this mixed Nash equilibria is actually dominated by the outcome (S, B), in which both parent and child receive more payoff.

		Child: II	
		B	M
Parent: I	S	1,2	0,3
	P	0,1	2,0

Figure 2.15: Example of class 3B classification: Spoiled Child game

2.5.2 Classification by payoff relations

Another classification, commonly used in evolutionary game theory (EGT) as for example in Nowak (2006), classifies the game space depending on the internal payoff relations. Given again the general normal form representation of a 2x2 game (Figure 2.16), one can use the relations between the payoffs to separate one class of games from another. Figure 2.16 for example shows the prisoner's dilemma game in which the following payoff relations are valid:

$$R21 > R11 > R22 > R12 \text{ and } C12 > C11 > C22 > C21$$

		II	
		A	B
I	A	R11,C11	R12,C12
	B	R21,C21	R22,C22

⇨

		II	
		A	B
I	A	3,3	0,5
	B	5,0	1,1

Figure 2.16: Normal form game representation of Prisoner's dilemma

While in traditional game theory, individual agents interact with each other, in EGT interactions are undertaken by entire populations and success of a population translates into reproductive success. Consider for example a generic population

game as in Figure 2.17. In this game, two populations A and B are interacting with each other in the way that each individual in the population either interacts with a member of his own population or of one of the opposite population, with a probability depending on the respective population sizes, thereby receiving a payoff depending on the opponent. The size of the population is then adjusted depending on the average of all the individual payoffs. Nowak (2006) shows that depending on how the payoffs are set in relation with each other, 5 different classes of scenarios (Table 2.1) can arise for this game, ranging from no change to the domination of one population causing the extinction of the others.

	A	B
A	R11,R11	R12,R21
B	R21,R12	R22,R22

Figure 2.17: A symmetric 2x2 game, a generic population game

Table 2.1: Results depending on payoff relations (Nowak, 2006)

Class	Condition	Result
1	$A11 > A21$ and $A12 > A22$	A dominates B
2	$A11 < A21$ and $A12 < A22$	B dominates A
3	$A11 > A21$ and $A12 < A22$	A or B dominates
4	$A11 < A21$ and $A12 > A22$	A and B coexist
5	$A11 = A21$ and $A12 = A22$	No population change

However, while this type of classification has been frequently used in symmetric 2x2 games, games where each player is identical with respect to the game rules, this classification becomes infeasible to use in the generic 2x2 game space because of the increase of payoff parameters (Figure 2.18). Additionally, while the number of different possible relations is still quite limited in the symmetric game structure, from a game theoretic point of view, several different payoff relations ultimately yield a similar convergence result. In Table2.1 for example, whether A dominates B or B dominates A is in this view in fact not important. Additionally, as Cheng et al. (2004) showed that every 2 player symmetric game has a pure Nash equilibrium, a restriction to 2x2 symmetric games eliminates all occurrences of games of the earlier discoordination class (2.5.1) where games only have

a mixed NE. Therefore, in situations where the differences in convergence behavior are more prominent, the first classification based on the Nash characteristics is often preferred.

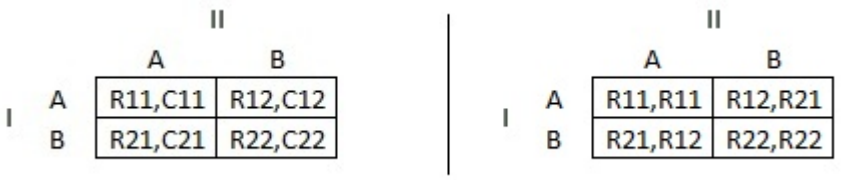


Figure 2.18: A generic (l.) and a symmetric (r.) 2x2 normal form game

2.5.3 Sample classification

As the classification based on Vega-Redondo (2003) and extended by Wunder et al. (2010) delivers the best subdivision for (Nash) convergence characteristics, this classification scheme is used in the rest of this work. Following is an overview of the result of this classification applied on a number of sample games. This overview serves as a short schematic repetition of the earlier more elaborate overview. The games in this overview with their specified payoff characteristics are used further in Section 4.

Class 1: Dominance games

Subclass A: Deadlock game (Figure 2.19)

NE1
- Type: Pure NE
- Strategy: (D, D)
- Payoffs: $(2, 2)$
- Dominated by non-Nash state: No

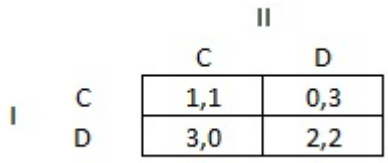


Figure 2.19: 1A: The Deadlock game

Subclass B: Prisoner's dilemma (Figure 2.20)

NE1
- Type: Pure NE
- Strategy: (D, D)
- Payoffs: $(1, 1)$
- Dominated by non-Nash state: Yes by (C, C) with payoffs $(3, 3)$

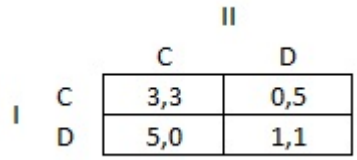


Figure 2.20: 1B: Prisoner's dilemma

Class 2: Coordination and Contribution games

Subclass A: Battle of the sexes (Figure 2.21)

NE1
- Type: Pure NE
- Strategy: (O, O)
- Payoffs: $(2, 3)$
- Dominated by non-Nash state: No

NE2
- Type: Pure NE
- Strategy: (F, F)
- Payoffs: $(3, 2)$
- Dominated by non-Nash state: No

NE3
- Type: Mixed NE
- Strategy: $(.4O + .6F, .6O + .4F)$
- Payoffs: $(1.2, 1.2)$
- Dominated by non-Nash state: No

		Wife	
		Opera	Football
Husband	Opera	2,3	0,0
	Football	0,0	3,2

Figure 2.21: 2A: Battle of the sexes

Subclass B: Chicken game (Figure 2.22)

NE1
- Type: Pure NE
- Strategy: (C, D)
- Payoffs: $(2, 7)$
- Dominated by non-Nash state: No

NE2
- Type: Pure NE
- Strategy:(D, C)
- Payoffs: $(7, 2)$
- Dominated by non-Nash state: No

NE3
- Type: Mixed NE
- Strategy: $(.67C + .33D, .67C + .33D)$
- Payoffs: $(14/3, 14/3)$
- Dominated by non-Nash state: Yes by (C, C) with payoffs $(6, 6)$

		II	
		C	D
I	C	6,6	2,7
	D	7,2	0,0

Figure 2.22: 2B: Chicken game

Class 3: Discoordination games

Subclass A: Matching pennies (Figure 2.23)

NE1
- Type: Mixed NE
- Strategy: $(.5C + .5D, .5C + .5D)$
- Payoffs: $(0.5, 0.5)$
- Dominated by non-Nash state: No

		II: C	II: D
I	C	1,0	0,1
	D	0,1	1,0

Figure 2.23: 3A: Matching pennies game

Subclass A: Shapley's game (Figure 2.24)

NE1
- Type: Mixed NE
- Strategy: $(.33A1 + .33A2 + .33A3, .33A1 + .33A2 + .33A3)$
- Payoffs: $(1/3, 1/3)$
- Dominated by non-Nash state: No

		II: A1	II: A2	II: A3
I	A1	0,0	1,0	0,1
	A2	0,1	0,0	1,0
	A3	1,0	0,1	0,0

Figure 2.24: 3A: Shapley's game

Subclass B: Spoiled child (Figure 2.25)

NE1
- Type: Mixed NE
- Strategy: $(.5S + .5P, .67B + .33M)$
- Payoffs: $(2/3, 3/2)$
- Dominated by non-Nash state: Yes by (S, B) with payoffs $(1, 2)$

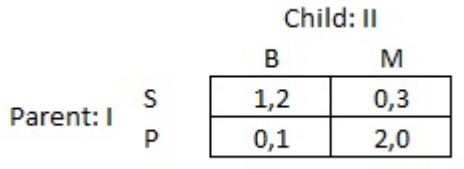


Figure 2.25: 3B: Spoiled Child game

Chapter 3

Learning under interaction

This chapter introduces the different learning models that will be considered in this thesis. The focus will be on simple unsophisticated learning rules called adaptive heuristics (Hart, 2005) and in particular those adaptive heuristics with no information requirements regarding the opponents' actions or payoff. This particular set of learning models is sometimes referred to as fully distributed algorithms (Debbah et al., 2011).

3.1 Overview

In multi-agents environments where strategic interactions are common, different types of learning models can be distinguished from each other depending on their learning characteristics. In evolutionary game theory for example, learning is done through the use of groups of agents in which each agent contributes to the total utility of the group. In these groups, individual agents themselves don't have learning capabilities.

In traditional game theory however, games lack the presence of groups and strategic interactions only concern individual agents. In this setting, groups of learning models can be distinguished depending on their level of rationality. While there exist sophisticated learning models operating with a high level of rationality, these often require a great amount of information in order to converge consistently to Nash equilibria. Especially information on the opponent's payoff function has often been shown to be a important condition for guaranteed Nash convergence (Jordan, 1991, 1993; Nachbar, 2005; Foster and Young, 2006). In practice however, this information is often not available, making the application of these sophisticated learning models often infeasible.

Because of that, adaptive heuristics are often much more attractive. As the term heuristic refers to a simple unsophisticated rule, an adaptive heuristic is a simple, unsophisticated, simplistic rule that is able to adapt over time in order to reach better results (Hart, 2005). These rules usually have less strict information requirements and they, although not being fully rational, nevertheless yield rational behavior in the long run (Hart, 2005). Additionally, if we consider computational behavior, these algorithms usually require only a limited amount of memory and computational power as in most cases the action history is limited to the previous time step while the computations are very intuitive and follow simple rules.

For several of these rules, the information requirements are even restricted to information own to the agent: his action space, past selected actions and corresponding payoffs. These fully distributed algorithms, also referred to as completely uncoupled (Young, 2009), or radically uncoupled (Foster and Young, 2006) rules, are even more applicable in real life situations as they do not require any information about the opponent's actions or payoffs. In fact agents don't even have to know how many other players are in the game or even if there are other players involved. If rules do require information about the opponent's actions, they are simply called uncoupled rules (Foster and Young, 2003, 2006).

In the next two sections, a selection of uncoupled adaptive heuristics is introduced with a main focus on completely uncoupled rules. While the first one (Section 3.2) elaborates on models based on the principle of reinforcement learning (Bush and Mosteller, 1951), the following section (Section 3.3) introduces a number of algorithms all built around the notion of minimizing regret. Although in many ways quite similar and both conceptually easy to grasp, an important difference is the fact that the models based on minimizing regret all have important theoretical properties concerning their convergence behavior. This because of the built-in performance criterion. While reinforcement learning models aim to maximize their expected future reward, there is no specific goal limit. Regret minimization models on the other hand have as main goal to minimize the regret up to a point of zero regret. Although this situation cannot always be reached, this specific performance goal allows for some interesting performance proofs (Hart and Mas-Colell, 2000; Young, 2004).

To conclude this chapter a small overview will be given about some other classes of algorithms (Section 3.4). These models, often of a more sophisticated nature, all share the same characteristic, namely the requirement of more game and/or player information.

3.2 Reinforcement Learning

Reinforcement learning (RL) (Bush and Mosteller, 1951), inspired by the field of behavioral psychology, embraces a number of methods built upon the same principle that states that actions that already resulted in high payoffs are more probable to be taken again in the future. This principle is known as the "law of effect", a finding in origin dating back to at least the end of the 19th century with Thorndike's animal experiments (Thorndike, 1898). The algorithms which are the subject of this section will thus all share the same characteristic: the probability of each action to be selected depends on its past payoffs. The better an action performed in the past relative to the other actions, the more likely the action is to be selected again. Well performing actions are thus reinforced, hence the name. As probabilities will only depend on the own actions and payoffs, all these learning models can be classified as completely uncoupled rules.

In these reinforcement learning models, two classes can be distinguished. The first four models use the cumulative payoff matching (CPM) (Young, 2004) framework and all originate from the field of economics. While the first algorithm introduces the basic CPM framework, the subsequent algorithms introduce extensions and improvements including the addition of random trembles (Roth and Erev, 1995) and the incorporation of the recency effect (Ebbinghaus, 1913).

Following these CPM models, a selection of Q-learning (Watkins, 1989) models is introduced. Q-learning, in contrast with the previous models originating from the field of computer science, was originally designed for applications in single-agent environments. However, while in these environments convergence to the optimal outcome has been proven before (Watkins and Dayan, 1992; Tsitsiklis, 1994), these proofs cannot generally be extended to multi-agent environments because of the important influence of the other players' actions (Claus and Boutilier, 1998). Similar to the CPM models, a base version is followed by more extensive models. In the case of Q-learning, this involves the introduction of initial lenient behavior (Bloembergen et al., 2010) and an adaptation to the update rule (Kaisers and Tuyls, 2010).

3.2.1 Cumulative Payoff Matching

This algorithm, cumulative payoff matching (CPM), introduced by Roth and Erev (1995) demonstrates the basic framework of the CPM model. The main idea behind this model is to associate each possible action a with the total payoff that

action delivered in the past θ_a. The probability distributions $\boldsymbol{q}$ will then be formed proportional to these cumulative payoffs as observable in Algorithm 1.

The basic structure of this model can be structured in two different, but ultimately completely equal ways, namely in function of updating the cumulative payoff vector (Algorithm 1) or by directly manipulating the probability distribution (Algorithm 2). While Algorithm 1 might be the most intuitive to grasp, some extensions and improvements are more easily defined in function of manipulation of the probability distribution which is the reason that both are mentioned here. Young (2004) however showed that both are completely equivalent.

As can be observed in Algorithm 1, in order to exploit the past, the agent will at all times maintain a cumulative payoff vector $\boldsymbol{\theta}$, with for each action a notion of how well that action is, along with the previously chosen action a^{t-1} and corresponding payoff u^{t-1}. All other information, including the probability distribution $\boldsymbol{q}$, will be derived from this. The use of this method ensures that an agent does not have the requirement to maintain knowledge about the entire past as this is incorporated in the cumulative payoff.

Algorithm 1 Cumulative payoff matching (CPM)

$u \leftarrow$ Payoff function
$\boldsymbol{\theta}^t \leftarrow$ Cumulative payoff vector
$\boldsymbol{\theta}^0 \leftarrow$ Initial propensities (≥ 0)
$q_a^t \leftarrow$ Probability of playing a at period t
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

If $t > 1$: $\Delta\theta_a^t = \begin{cases} u^{t-1} & a = a^{t-1} \\ 0 & a \neq a^{t-1} \end{cases}$

$$\boldsymbol{\theta}^t = \begin{cases} \boldsymbol{\theta}^0 & t = 1 \\ \boldsymbol{\theta}^{t-1} + \boldsymbol{\Delta\theta}^t & t > 1 \end{cases}$$

$$\boldsymbol{q}^t = \boldsymbol{\theta}^t / \sum_{a' \in A} \theta_{a'}^t$$

$$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$$

A similar method can be seen in Algorithm 2 which, instead of the cumulative payoffs, requires the total realized payoff v and the previous probability distribution $\boldsymbol{q}$ to be known along with the previously selected action a^{t-1} and payoff

u^{t-1}. In this algorithm, a random unit vector $\boldsymbol{e}$ is updated which is used to calculate the change in probabilities $\boldsymbol{\Delta q}$. Because of the use of $\boldsymbol{e}$, the total sum of probabilities is guaranteed to remain exactly 1. Although the required information is maintained in different variables, the game history is again restricted to only 1 time step, identical to Algorithm 1.

Algorithm 2 CPM: structured in terms of probability distribution q

$u \leftarrow$ Payoff function
$\boldsymbol{e}^t \leftarrow$ Random unit vector
$q_a^t \leftarrow$ Probability of playing a at period t
$\boldsymbol{q}^0 \leftarrow$ Initial probabilities
$\boldsymbol{\theta}^0 \leftarrow$ Initial propensities (≥ 0)
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$v^t \leftarrow$ Value maintaining the total realized payoff
$\text{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

$$v^t = \begin{cases} \sum_{a' \in A} \theta_a^0 & t = 1 \\ v^{t-1} + u^{t-1} & t > 1 \end{cases}$$

$$\text{If } t > 1\text{: } e_a^t = \begin{cases} 1 & a = a^{t-1} \\ 0 & a \neq a^{t-1} \end{cases}$$

$$\text{If } t > 1\text{: } \boldsymbol{\Delta q}^t = [u^{t-1}/v^t] * [\boldsymbol{e}^t - \boldsymbol{q}^{t-1}]$$

$$\boldsymbol{q}^t = \begin{cases} \boldsymbol{q}^0 & t = 1 \\ \boldsymbol{q}^{t-1} + \boldsymbol{\Delta q}^t & t > 1 \end{cases}$$

$$a^t = \text{Choice}(\boldsymbol{q}^t)$$

While Algorithm 1 and 2 are both equivalent, Algorithm 2 introduces an important feature of CPM more clearly. This feature, also present in several other learning models, is what is called the "power law of practice", a psychological law first proposed by Newell et al. (1981). With ideas dating back to at least Snoddy (1926) and Blackburn (1936), the power law of practice covers the relation between practice and improved performance (Figure 3.1). As can be observed in the calculation of the change of probabilities $\boldsymbol{\Delta q}$, while the total realized payoff v^t is essentially unbounded, the payoff in each period u^{t-1} is not. Because of this contrast, the incremental impact of the payoff in every period t actually diminishes over time causing the agents to learn less and less over time.

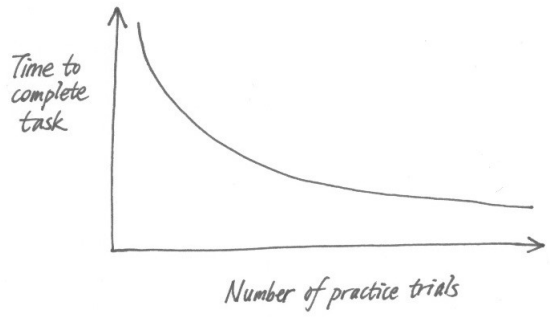


Figure 3.1: The power law of practice, depicting the relation between the number of trials and the improved performance.

3.2.2 Arthur's CPM model

Although the basic CPM algorithm already incorporates the power law of practice, the corresponding learning curve, depicting the changing rate of learning (Figure 3.1), cannot be manipulated in order to alter the rate of learning decline. For this purpose, Arthur (1990, 1991, 1993) introduced a related class of models with two additional variables to allow for a more extensive manipulation of this decrease.

While in the basic CPM model (Equation 3.1) the incremental impact of payoffs decreases over time with a fixed factor $1/t$ (Young, 2004), in Arthur's CPM model (CPM-A), this impact is assumed to decrease as a power of t (using estimations from data, Young, 2004) modifiable using two constants C and p (Equation 3.2).

$$\Delta q_a^t = [u^{t-1}/v^t] * [e_a^t - q_a^{t-1}] \qquad \forall a \in A \tag{3.1}$$

$$\Delta q_a^t = [u^{t-1}/(Ct^p + u^{t-1})] * [e_a^t - q_a^{t-1}] \qquad \forall a \in A \tag{3.2}$$

The way the power law of practice is implemented in this 2-parameter family of algorithms, where the speed is influenced using C and the deceleration using p, can be seen in Algorithm 3 showing an adapted Algorithm 2.

Algorithm 3 Arthur's CPM model (CPM-A)

$u \leftarrow$ Payoff function
$\boldsymbol{e}^t \leftarrow$ Random unit vector
$q_a^t \leftarrow$ Probability of playing a at period t
$\boldsymbol{q}^0 \leftarrow$ Initial probabilities
C and $p \leftarrow$ nonnegative constants
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

If $t > 1$: $e_a^t = \begin{cases} 1 & a = a^{t-1} \\ 0 & a \neq a^{t-1} \end{cases}$

If $t > 1$: $\boldsymbol{\Delta q}^t = [u^{t-1} / \left(C(t-1)^p + u^{t-1}\right)] * [\boldsymbol{e}^t - \boldsymbol{q}^{t-1}]$

$\boldsymbol{q}^t = \begin{cases} \boldsymbol{q}^0 & t = 1 \\ \boldsymbol{q}^{t-1} + \boldsymbol{\Delta q}^t & t > 1 \end{cases}$

$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$

Initially designed in order to mimic human learning in a multiple choice setting and for that reason only tested on a set of two-choice bandit experiments (see Bush and Mosteller, 1955), this algorithm shows promising convergence characteristics with, for these problems, a guaranteed convergence to the optimal outcome (when $p = 1$). Additionally, also with lower p values, an almost 100% convergence rate can be achieved in cases where the best action is at least 15% better than the second best option (Arthur, 1991, 1993). Whether these results can be extended to multi-agent environments is however unknown.

3.2.3 Roth and Erev's CPM model

Another model incorporating the power law of practice has been proposed by Roth and Erev (1995). However, while in CPM and CPM-A, the incremental impact of newly received utility keeps diminishing, and the learning curve keeps lowering, in Roth and Erev's CPM model (CPM-RE), a constant geometric rate $\lambda < 1$ determines how low the learning curve will ultimately become.

CPM-RE is based on Algorithm 1 but multiplies in each iteration the previous cumulative payoff with this geometric rate λ in order to limit its importance. Because of this, part of the past is "forgotten" and, in contrast with CPM and CPM-A, in CPM-RE a point emerges in which the "forgotten" part of the past compensates the newly received utility causing this declining effect to stop.

By using this mechanism, CPM-RE explicitly adds what is called the recency effect (Ebbinghaus, 1913). This psychological law states that the more recent the knowledge, the more important it will be considered. The main advantage of this effect lies in the fact that in the long run, the effects of the early-on exploration process are highly minimized and more emphasis is put on the later, more informed, decisions.

Another novelty compared to the previous CPM models, is the addition of random perturbations $\boldsymbol{w}^t$ in order to improve and somewhat randomize the exploration. These perturbations, or "trembles", can be seen as the introduction of small errors to compensate for particular uncertainties. Although these trembles might ultimately prevent a complete 100% convergence, in some situations they provide a way to react against changes in the environment (Karandikar et al., 1998). Especially in games without pure equilibria, as shown in chapter 4, these trembles can even be essential to achieve a proper performance.

Although no experiments on 2x2 repeated normal form games have been found, according to Roth and Erev (1995) who reviewed this algorithm for three other types of games containing strategic interaction, this algorithm can, depending on the characteristics, allow for a quick, consistent convergence to one Nash equilibrium state. However for one of the tested games, a simplified ultimatum game which can be transformed into a 2x9 normal form game, this convergence behavior was not observed. Although in most cases, the model for this game still did converge to one of the Nash equilibria, in 2 out of 10 simulations, convergence to a non-Nash outcome was also observed. Additionally, the results of this last game have been shown to be highly sensitive to the initial conditions. The full CPM-RE model can be seen at Algorithm 4.

Algorithm 4 Roth and Erev's CPM model (CPM-RE)

$u \leftarrow$ Payoff function
$\boldsymbol{\theta} \leftarrow$ Cumulative payoff vector
$\boldsymbol{\theta}^0 \leftarrow$ Initial propensities
$q_a^t \leftarrow$ Probability of playing a at period t
$\boldsymbol{w}^t \leftarrow$ Vector of random perturbations used in period t
$\lambda \leftarrow$ Constant geometric rate < 1
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\text{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

If $t > 1$: $\Delta\theta_a^t = \begin{cases} u^{t-1} & a = a^{t-1} \\ 0 & a \neq a^{t-1} \end{cases}$

$$\boldsymbol{\theta}^t = \begin{cases} \boldsymbol{\theta}^0 & t = 1 \\ \lambda\boldsymbol{\theta}^{t-1} + \boldsymbol{\Delta\theta}^t + \boldsymbol{w}^t & t > 1 \end{cases}$$

$$\boldsymbol{q}^t = \boldsymbol{\theta}^t / \sum_{a' \in A} \theta_{a'}^t$$

$$a^t = \text{Choice}(\boldsymbol{q}^t)$$

3.2.4 Borgers and Sarin's CPM model

Another model built on the CPM framework has been proposed by Borgers and Sarin (2000). In their work, they propose the introduction of an aspiration level. In each period t, an action will be either positively or negatively reinforced depending on the fact whether the corresponding payoff is either above or beneath this aspiration level, thus affecting its probability of getting selected again in the future.

Although the aspiration level could also be chosen exogenous and constant during the entire algorithm, Borgers and Sarin (1997) showed that such a fixed aspiration level causes persistent irrational behavior in certain situations. Therefore Borgers and Sarin's CPM model (CPM-BS) is fitted with an endogenous level based on the past realized payoffs which is adapted throughout the process in order to maintain a valid level. The idea behind the use of this endogenous aspiration level is mainly driven from an empirical point of view. As agents learn more and more about which payoffs to expect in a certain game, they will be able to adapt their aspiration level to reflect this knowledge (Borgers and Sarin, 2000).

In the experiments conducted by Borgers and Sarin (2000) on single agent decision problems, they found that the use of a correctly used aspiration level could indeed improve the agent's results. However it was also noted that in a large class of decision problems, the final results were persistently suboptimal caused by "probability matching" effects. These irrational effects cause the frequencies that the agents select an action to match to probabilities with which these actions are actually successful. Informally, it can be observed that an agent overreacts to disappointing payoffs causing the corresponding actions to be quite severely repositioned with regard to the aspiration level.

Although no experiments where conducted on normal form games, the analytical proofs by Karandikar et al. (1998) show that, given sufficiently accurate and slowly updated aspiration levels, both in class 1 (Dominance games: 2.5.1) and in class 2 games (Coordination and contribution games: 2.5.1), agents will opt for the most rewarding outcome most of the time, whether this is a Nash equilibrium or an outcome that dominates a Nash equilibrium. This even for games like the prisoner's dilemma where this means selecting the strictly dominated action. It is however unknown whether CPM-BS satisfies these conditions and the same results can be observed in practice.

As shown in Algorithm 5, the aspiration level will be set to represent a discounted sum of past payoffs d^t, using a constant discount factor λ. Since probabilities are calculated depending on the difference between the payoffs u^{t-1} and the aspiration level d^t, this difference cannot be allowed to grow beyond 1 as this would give rise to invalid probabilities. Therefore CPM-BS has been restricted to games with payoffs in the $[0, 1]$ interval. For this reason, all games on which the model is applied are normalized to accommodate this restriction. While this restriction puts serious limitations to practical applications as the payoffs have to be guaranteed to be in the specified interval, it is still applicable in situations where the opponent's realized payoffs are unknown as long as the range of own payoffs is known. CPM-BS is therefore still eligible to be called a completely uncoupled rule. In the rest of this work, this normalization is to be assumed in all applications of the model.

Algorithm 5 Borgers and Sarin's CPM model (CPM-BS)

$u \leftarrow$ Payoff function
$q_a^t \leftarrow$ Probability of playing a at period t
$\boldsymbol{q}^0 \leftarrow$ Initial probabilities
$\boldsymbol{e} \leftarrow$ Random unit vector
$\lambda \leftarrow$ Constant geometric rate < 1
$a^t \leftarrow$ Action chosen at period t
$d^t \leftarrow$ Averaged discounted sum of past payoffs t
$u^s = u(a^s) \leftarrow$ Payoff from time step s
$u^0 \leftarrow$ Value arbitrarily chosen, often 0
$\text{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

If $t > 1$: $e_a^t = \begin{cases} 1 & a = a^{t-1} \\ 0 & a \neq a^{t-1} \end{cases}$

If $t > 1$: $d^t = \dfrac{\sum\limits_{0 \leq s \leq (t-1)} \lambda^{t-1-s} u^s}{\sum\limits_{0 \leq s \leq (t-1)} \lambda^{t-1-s}}$

If $t > 1$: $\boldsymbol{\Delta q}^t = [u^{t-1} - d^t] * [\boldsymbol{e}^t - \boldsymbol{q}^{t-1}]$

$\boldsymbol{q}^t = \begin{cases} \boldsymbol{q}^0 & t = 1 \\ \boldsymbol{q}^{t-1} + \boldsymbol{\Delta q}^t & t > 1 \end{cases}$

$a^t = \text{Choice}(\boldsymbol{q}^t)$

3.2.5 Q-learning

Q-learning, first introduced by Watkins (1989) and initially designed for single-agent multi-state environments, has already been widely used in many fields of research like robotics (Yao et al., 2002) and packet routing in networks (Boyan and Littman, 1994). In this chapter, it is the first learning model that has its origins in the field of computer science rather than economics.

$$Q_{a^{t-1}} = Q_{a^{t-1}} + \alpha[u^{t-1} + \gamma \max_{a' \in A}(Q_{a'}) - Q_{a^{t-1}}] \tag{3.3}$$

In this single state Q-learning version, each player will associate a certain Q value to each of his actions, reflecting the expected utility of taking that action. In each time step, the player will then update one of these values depending on both the previous Q value, the action taken in the past time step a^{t-1} and the corresponding payoff u^{t-1} in order to either reinforce and deter the status of that action. This update rule can be seen in Equation 3.3.

As a possible initial Q value, it is common practice (Sutton and Barto, 1998) to choose the highest possible value to allow the learning model in the initial stage to explore all actions while actually acting greedy. However determining this highest possible value is often difficult. Especially in fully distributed algorithms, when the payoff matrix is unknown, this practice can be hard to accomplish. Therefore a neutral zero level might be a safe choice to avoid overly high initial values.

The Q update rule also knows two parameters whose values can influence the behavior of the algorithm and change the relation between exploration and exploitation. The first parameter is the discount factor γ, a value in the $[0, 1]$ interval which will determine the importance of future rewards. A small value will mainly consider the current payoffs, while a value close to 1 will make the algorithm strive for a long term result. The second parameter is the learning rate α. This value, also between 0 and 1, will determine how much the previous payoffs are reflected in the current Q values. A value of 1 will make the previous information obsolete by only considering the last payoff, a value of 0 on the other hand will make the learning model consider only the initial value and thus make the algorithm learn nothing.

Similar to basic CPM, after the update of the action related data, the action probabilities are calculated. However while in CPM this is commonly done using a basic proportional scheme, Q learning is often equipped with other action selection schemes. Most popular in these are the ϵ-greedy and the softmax method with Boltzmann distribution schemes. In the case of ϵ-greedy, where the epsilon refers to a predefined parameter, all actions are uniformly chosen with probability epsilon divided by the number of actions, except for the best action, which is chosen with the remaining, usually larger, probability. However, as this semi uniform probability prevents a complete stable convergence, a mechanism can be added to limit the importance of the ϵ value and ultimately remove or highly reduce the random exploration. Although several methods can be used ranging from abrupt changes at specific moments in time to continuous updating following a specific function, the overall behavior is identical. In the algorithm described in Algorithm 6, the perhaps most straightforward method is used in the form of a continuous linear decrease. Each time period, the influence of the ϵ parameter is reduced with proportion λ until a zero value is reached and a complete greedy behavior is experienced.

The second action selection rule is the softmax method using a Boltzmann distribution of which the corresponding function can be seen in Algorithm 6. This distribution has many similarities with the basic proportional scheme used in CPM

as it also favors the most promising actions. However how much the best actions will be favored can be tuned by adapting the temperature value τ. A value close to zero will significantly favor the best actions while larger numbers will make the distribution more equiprobable. In general, despite the large differences in behavior between these two action selection schemes, whether one of these action selection mechanism outperforms the other is hard to determine and usually depends on the problem and its parameter configuration.

While Watkins and Dayan (1992) and Tsitsiklis (1994) have shown that Q-learning in single-agent environments can be guaranteed to converge to the optimal outcome, these results cannot generally be extended to multi-agent environments because of the higher level of complexity. Additionally, Wunder et al. (2010) show in their empirical and theoretical analysis of Q-learning with ϵ-experimentation (Qϵ) that Nash convergence highly depends on the game it's applied to. In games where the Nash equilibrium is also the most optimal outcome (classification subclass A), convergence can be assumed to be guaranteed. However in games where the Nash equilibrium constitutes a suboptimal solution (i.e. there exists another outcome that dominates it), convergence behavior is more complicated ranging from no convergence for games without pure Nash equilibria to a convergence depending on the initial conditions. Kaisers and Tuyls (2010) finally showed that Q-learning with softmax action selection (Qs) can give rise to learning behavior different than expected, possibly even preventing Nash convergence from occurring. To solve this discrepancy, they have proposed an adjustment called frequency adjusted Q-learning as seen in the next subsection (Subsection 3.2.6).

Algorithm 6 Q-learning (Qϵ and Qs)

$u \leftarrow$ Payoff function
$Q_a \leftarrow$ Q values for each action a
$\alpha \leftarrow$ Learning rate $0 < \alpha < 1$
$\gamma \leftarrow$ Discount factor $0 < \gamma < 1$
$\epsilon \leftarrow \epsilon$-greedy action selection parameter
$\lambda \leftarrow \epsilon$-greedy experimentation reducing parameter
$\tau \leftarrow$ Softmax action selection temperature value
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$

Update q-values
If $t > 1$:$Q_{a^{t-1}} = Q_{a^{t-1}} + \alpha[u^{t-1} + \gamma \max_{a' \in A}(Q_{a'}) - Q_{a^{t-1}}]$
Option 1: Epsilon greedy

$$a^t = \begin{cases} a_{maxQ} & \text{With probability } 1 - \frac{\max(0,(\epsilon - t\lambda\epsilon))(|A|-1)}{|A|} \\ a_i & \text{With probability } \frac{\max(0,(\epsilon - t\lambda\epsilon))}{|A|}, \forall a_i \neq a_{maxQ} \end{cases}$$

Option 2: Softmax with Boltzmann

$$a^t = \begin{cases} a_i & \text{With probability} \frac{e^{(Q_{a_i}/\tau)}}{\sum_{a' \in A} e^{(Q_{a'}/\tau)}} \quad \forall a_i \in A \end{cases}$$

3.2.6 Frequency Adjusted Q-learning

One of the disadvantages of regular Q-learning is the fact that cases can occur where the empirically observed behavior does not comply with the expected behavior regarding convergence (Kaisers and Tuyls, 2010). And more importantly that this empirical behavior is actually less desirable than the expected behavior. More precisely, it has been shown that cases can arise when Q-learning actually behaves irrational in the sense that it moves away from dominant actions.

Research has shown that this is caused by the differences in update frequency of the different actions. Actions that are played more often will more often be updated and will quickly reach a reliable Q value while the less often chosen ones need more iterations to achieve this. To compensate for this behavior, Kaisers and Tuyls (2010) propose a modification called frequency adjusted Q-learning (FAQ).

This learning model is identical to the earlier seen regular Q learning but with an adjusted update rule (Equation 3.4). In this rule, the action probability q_a is taken into account. The factor β is introduced in order to eliminate the possibility

of the factor $\frac{\alpha}{q_a}$ becoming larger than 1 which would result in unreliable Q updates.

$$Q_a^t \leftarrow Q_a^{t-1} + \min\left(\frac{\beta}{q_{a^{t-1}}}, 1\right)\alpha\left[u^{t-1} + \gamma \max_{a' \in A} Q_{a'}^{t-1} - Q_a^{t-1}\right] \tag{3.4}$$

Using both theoretical and empirical confirmation, Kaisers and Tuyls (2010) have shown that this adaptation is indeed able to compensate for the different update ratio. Given a sufficiently small α and β value, FAQ with softmax action selection (FAQs) has in fact been shown to comply with the expected behavior and to exhibit a more rational behavior. For both the 1B prisoner's dilemma, 2A battle of the sexes and 3A matching pennies, convergence to the Nash equilibrium is observed and this in contrast with Qs almost independent of the initial conditions. If these α and β values however are initialized too high, this effect disappears and FAQ acts again like basic Q learning. The main disadvantage of this model is that, because of the change in the update rule, convergence is found to be much slower, with time periods exceeding the 100000 marker.

The full structure of FAQ can be found in Algorithm 7. Although Kaisers and Tuyls (2010) only introduced FAQs, FAQ can also be combined with an ϵ-greedy action selection scheme, resulting in FAQ with ϵ-experimentation (FAQϵ).

Algorithm 7 Frequency Adjusted Q-learning (FAQϵ and FAQs)

$u \leftarrow$ Payoff function
$Q_a \leftarrow$ Q values for each action a
$q_a^t \leftarrow$ Probability of playing a at period t
$\alpha \leftarrow$ Learning rate $0 < \alpha < 1$
$\gamma \leftarrow$ Discount factor $0 < \gamma < 1$
$\epsilon \leftarrow \epsilon$-greedy action selection parameter
$\lambda \leftarrow \epsilon$-greedy experimentation reducing parameter
$\tau \leftarrow$ Softmax action selection temperature value
$\beta \leftarrow$ FAQ parameter
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$

Update q-values

If $t > 1$:$Q_{a^{t-1}} = Q_{a^{t-1}} + \min\left(\frac{\beta}{q_{a^{t-1}}}, 1\right) \alpha \left[u^{t-1} + \gamma \max_{a' \in A}(Q_{a'}) - Q_{a^{t-1}}\right]$

Option 1: Epsilon greedy

$$a^t = \begin{cases} a_{maxQ} & \text{With probability } 1 - \frac{\max(0,(\epsilon - t\lambda\epsilon))(|A|-1)}{|A|} \\ a_i & \text{With probability } \frac{\max(0,(\epsilon - t\lambda\epsilon))}{|A|}, \forall a_i \neq a_{maxQ} \end{cases}$$

Option 2: Softmax with Boltzmann

$$a^t = \begin{cases} a_i & \text{With probability} \frac{e^{(Q_{a_i}/\tau)}}{\sum_{a' \in A} e^{(Q_{a'}/\tau)}} \quad \forall a_i \in A \end{cases}$$

3.2.7 Lenient (Frequency Adjusted) Q-learning

In reinforcement learning, avoiding convergence to less optimal solutions is often a prime challenge. Partially responsible for this phenomenon is the fact that agents might miscoordinate their actions in the initial stage of the game. To overcome this problem, the concept of lenient behavior has been proposed to be applied in multi-agent learning (Panait et al., 2006). This concept suggests that initial low rewarding actions taken by the other agents can be forgiven for the sake of a long term optimized result.

Bloembergen et al. (2010) propose a variant of FAQ incorporating this behavior called lenient frequency adjusted Q-learning (LFAQ). Although lenient behavior can also be implemented in regular Q-learning resulting in lenient Q-learning (LQ), Bloembergen et al. (2010) propose the combination of the lenient behavior with the earlier proposed FAQ in order to combine the advantages of both and to end up with a superior model compared to LQ.

The addition of lenient behavior could for example be usefully experienced in the prisoner's dilemma game (Figure 3.2). In this game, it could be a good idea to forgive initial D(efect) action in order to end up in the optimal (C, C) outcome of the game. This in favor of the less rewarding Nash equilibrium outcome (D, D) to which convergence might otherwise converge too.

		II	
		C	D
I	C	3,3	0,5
	D	5,0	1,1

Figure 3.2: Prisoner's dilemma

Different possible methods exist to embed leniency in Q-learning. The most straightforward, as also proposed by Bloembergen et al. (2010), is to wait to update the Q values until all actions have been played κ times. When this is done, each Q value will be updated using the maximum realized payoff resulting from that action. This way, initial mistakes from the other players are simply forgotten and the well rewarding situations are used for the initial Q value updating.

Experiments on two 2A coordination games and the 3A matching pennies game conducted by Bloembergen et al. (2010) for both lenient Q-learning with softmax action selection (LQs) and LFAQ with softmax action selection (LFAQs) show that the LFAQ version inherits the advantages of FAQ by allowing for a consistent convergence to the Nash equilibrium (NE). This in contrast with LQs who, similarly to Qs, has a more inconsistent convergence behavior depending on the initial conditions. Especially the initial Q value Q_0 has a large influence. Additionally, a large κ value can, for the tested 2A coordination game, allow for an almost complete convergence to the best rewarding NE. Although not tested, it can be expected that similar conclusions can be drawn with regard to LFAQ with ϵ-experimentation (LFAQϵ)

In Algorithm 8, both LQ (LQs, LQϵ) and LFAQ (LFAQs, LFAQϵ) have been implemented based on the implementations of basic Q-learning and FAQ (Algorithms 6 and 7). Although (L)FAQ might have the theoretical edge, it can also be pointed out that (L)FAQ has an important empirical disadvantage in the form of the slow convergence speed. Therefore it has been chosen to introduce both so

the potential added value of incorporating lenient behavior can be explored more deeply.

Algorithm 8 Lenient (frequency adjusted) Q-learning (L(FA)Q)

$u \leftarrow$ Payoff function
$Q_a \leftarrow$ Q values for each action a
$\boldsymbol{Q}^0 \leftarrow$ Initial Q-values
$q_a^t \leftarrow$ Probability of playing a at period t
$\alpha \leftarrow$ Learning rate $0 < \alpha < 1$
$\gamma \leftarrow$ Discount factor $0 < \gamma < 1$
$\epsilon \leftarrow \epsilon$-greedy action selection parameter
$\lambda \leftarrow \epsilon$-greedy experimentation reducing parameter
$\tau \leftarrow$ Softmax action selection temperature value
$\beta \leftarrow$ FAQ parameter
$\kappa \leftarrow$ The Q values are not updated untill all action have been chosen κ times
$a^t \leftarrow$ Action chosen at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$

Update q-values if all actions have been chosen κ times:
- Option 1: Lenient Q-learning
If t > 1:$Q_{a^{t-1}} = Q_{a^{t-1}} + \alpha \left[u^{t-1} + \gamma \max_{a' \in A}(Q_{a'}) - Q_{a^{t-1}}\right]$
- Option 2: Lenient Frequency Adjusted Q-learning
If t > 1:$Q_{a^{t-1}} = Q_{a^{t-1}} + \min\left(\frac{\beta}{q_{a^{t-1}}}, 1\right) \alpha \left[u^{t-1} + \gamma \max_{a' \in A}(Q_{a'}) - Q_{a^{t-1}}\right]$

Action selection:
- Option 1: Epsilon greedy
$$a^t = \begin{cases} a_{maxQ} & \text{With probability } 1 - \frac{\max(0,(\epsilon - t\lambda\epsilon))(|A|-1)}{|A|} \\ a_i & \text{With probability } \frac{\max(0,(\epsilon - t\lambda\epsilon))}{|A|}, \forall a_i \neq a_{maxQ} \end{cases}$$
- Option 2: Softmax with Boltzmann
$$a^t = \begin{cases} a_i & \text{With probability} \frac{e^{(Q_{a_i}/\tau)}}{\sum_{a' \in A} e^{(Q_{a'}/\tau)}} \quad \forall a_i \in A \end{cases}$$

3.3 Regret

While reinforcement learning makes decisions based on what has been received in the past and aims to maximize this, another group of models exists making decisions based on what could have been received if other actions were chosen. These models are based around the notion of regret, for repeated games first introduced by Hannan (1957), and sometimes also called opportunity gain or loss. Regret

can be seen as the amount that could have been gained or lost by taking another action in the past than the one that was actually chosen. Algorithms dealing with regret will aim to minimize this in order to come to outcomes without any regret. Situations without regret correspond to outcomes where no other action choice would have given rise to more payoff.

As a real life example due to Blocki et al. (2011), consider the situation of a hospital where several employees can access highly confidential patient records, either for professional purposes like treatments or for personal business like plain human curiosity. In this situation, the goal of the hospital will be to minimize the costs by balancing the cost of analyzing the system with the risk of inappropriate access. Surely, not only is the reputation of the hospital at risk, official regulations regarding confidentiality are also into play. The hospital's best strategy is then one that minimizes her regret.

Behavioral rules of this type can be largely classified into two classes, either unconditional regret learning or conditional regret learning. In unconditional regret learning, regret is defined as the amount that an agent would have gained or lost if he would have played a certain action in all previous time steps instead of the actual played actions. The term unconditional here refers to the fact that no condition is set to the previous action, the regret is calculated for all previously played actions. In the more sophisticated conditional regret learning however, conditions on the previously played actions are in place. Conditional regret can here be calculated for every action pair (x, y). The regret is then defined as the amount that an agent would have gained or lost if he would have played action x in all previous periods where he actually played action y. The difference with unconditional regret thus lies with the condition upon the action that was played before. Unconditional and conditional regret are in computer science literature also referred to as external and internal regret respectively (Greenwald and Jafari, 2003).

The advantage of conditional regret over unconditional regret is the availability of a higher level of information. To illustrate this consider the example (Dobson, 1999) of an agent who over time bought several stocks from a number of companies. After a while it is then possible that this agent has conditional regret if it would have been better that he had bought Microsoft at times where he actually bought IBM. However always buying Microsoft would not have yielded a higher reward. So while "buy Microsoft stock instead of IBM" corresponds with a positive conditional regret, "buying Microsoft" does not correspond with a positive unconditional regret.

Although there are differences in the handling and calculation of regret, the main strategy remains the same, i.e. the models will try to minimize the regret up to a point of zero regret. This is in contrast with the previous RI models, where the opposite direction is present, the aim there is to maximize some mathematical measure until it reaches an unknown maximum. For these regret minimization models, the inherent presence of a goal, i.e. zero, leads to some hard theoretical properties regarding firstly the elimination of regret in the long term and secondly the convergence to a set of equilibria. While these theoretical properties can be considered rather weak in practice, convergence in the long run might in practice not be visible for many iterations, they do provide a stronger base for empirical work compared to most reinforcement learning models who lack these properties.

Although the basic method of regret calculation, used in models as regret matching (RM) (Subsection 3.3.1), requires additional information in the form of the own payoff function and the opponents' actions, thereby not classifying it as a fully distributed algorithm, methods of regret estimations without opponent information have been invented to overcome this problem. However, while this is currently feasible for unconditional regret learning, for the more advanced conditional regret minimization models, no regret estimations without opponent information have yet been proposed. Therefore the proposed models of that type will only satisfy the criteria for uncoupled models and not for completely uncoupled models. This gives conditional regret minimization models a disadvantage compared to the reinforcement learning models and the unconditional regret minimization models like regret matching with ϵ-experimentation (RMϵ) (Subsection 3.3.2) that use a regret estimation technique.

3.3.1 Regret Matching

While since Hannan (1957) numerous rules have been invented embedding the regret property, one of the most simple and interesting rules incorporating regret has been introduced by Hart and Mas-Colell (2000). This rule, computationally almost as easy as the reinforcement learning models seen earlier, shall be called regret matching (RM).

In this model, each time period the average realized payoff $\overline{r}_a$ will be updated along with the different regret values r_a. Ultimately, the action probabilities will be derived in a proportional manner from the average regret values $\overline{r}_a$ for which only the nonnegative parts $[\overline{r}_a]_+$ are considered. Using only the non-negative parts guarantees that only the actions with positive regret and thus potential will be able to become selected and further explored. As in this algorithm the payoff function

u and the opponents' actions a^t_{-i} play an important role, it is clear that RM cannot be classified as a fully distributed algorithm.

As has been proven by Hart and Mas-Colell (2000), when all players choose according to this probability vector, the empirical distributions will converge to the set of coarse correlated equilibria. Additionally, independent of the other players' play, using RM yields no regret in the long run.

Algorithm 9 Regret matching (RM)

$u \leftarrow$ Payoff function
$q^t_a \leftarrow$ Probability of playing a at period t
$r^t_a \leftarrow$ Total regret up to t from not having played action a ($r^0_a = 0$)
$\overline{r}^t_a \leftarrow$ Average regret from not having played action a
$a^t \leftarrow$ Action chosen at period t
$a^t_{-i} \leftarrow$ Actions chosen by the opponent('s) at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\overline{u}^{t-1} \leftarrow$ Average payoff through to time $t-1$
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen
$[x]_+ \leftarrow$ The nonnegative part of x

---

if $t > 1$ **then**
$\overline{u}^{t-1} = \frac{1}{(t-1)}(\overline{u}^{t-2} * (t-2) + u^{t-1})$
$r^{t-1}_a = r^{t-2}_a + u(a, a^{t-1}_{-i})$
$\overline{r}^{t-1}_a = \frac{1}{(t-1)}(r^{t-1}_a) - \overline{u}^{t-1}$
$\boldsymbol{q}^t = [\overline{\boldsymbol{r}}^{t-1}]_+ / \sum_{a' \in A} [\overline{r}^{t-1}_{a'}]_+$
If denominator equal to zero, $\boldsymbol{q}^t$ is set to a uniform distribution over A
end if
$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$

3.3.2 Regret Matching with ϵ-experimentation

Among the more basic regret minimization algorithms classifiable as fully distributed algorithms is regret matching with ϵ-experimentation (RMϵ). This model, introduced by Young (2004), uses a regret estimation rule with no information requirements on the payoff function or the actions of the opponents. The general idea was initially proposed by Foster and Vohra (1993) in another context and through a suggestion of Dean Foster (Hart and Mas-Colell, 2000, fn30) first been related to regret minimization.

In this model, an agent will in each period experiment with a certain probability ϵ after which he plays all actions with equal probability. In the other case he will use RM (Subsection 3.3.1) with one modification. Instead of using the actual regret, an estimated version will be used, which for each action a can be calculated as the "average payoff in previous periods when he experimented and chose a, minus the average realized payoff over all actions in all previous periods" (Young, 2004).

It has been proven by Foster and Vohra (1998) that, given the experimentation value ϵ goes to zero at a sufficiently small rate ($O(t^{-1/3})$), the exact same convergence behavior can be expected as with RM and situations with no regret can be guaranteed. To enable this behavior, the same linear experimentation reducing mechanism has been applied as with Q learning (Algorithm 6, 7, 8). In each time period the experimentation coefficient ϵ is reduced with a proportion equal to $\lambda\epsilon$. The entire algorithm is shown in Algorithm 10.

Algorithm 10 Regret matching with ϵ-experimentation (RMϵ)

$u \leftarrow$ Payoff function
$\overline{r}_a^t \leftarrow$ Estimated regret from not having played action a
$\overline{u_{EXP}}_a^t \leftarrow$ Average payoff in all periods up to t when experimented + chose a
$q_a^t \leftarrow$ Probability of playing a at period t
$\epsilon \leftarrow$ Predefined probability $0 < \epsilon < 1$
$r \leftarrow$ Random number $0 < r < 1$
$\overline{u}^t \leftarrow$ Average realized payoff over all actions in all periods up to t
$a^t \leftarrow$ Action chosen at period t
$u^s = u(a^s) \leftarrow$ Realized payoff from time step s
$experimented = \textit{TRUE} \leftarrow$ Denote experimented or regret matching
$\mathrm{calcAvg}(A, B) \leftarrow$ The calculation of the new average using the previous average A and the new value B
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen
$[x]_+ \leftarrow$ The nonnegative part of x

---

if $t > 1$ **then**

$$\forall a \in A : \overline{u_{EXP}}_a^{t-1} = \begin{cases} \overline{u_{EXP}}_a^{t-2} & \text{If (not experimented)} \\ \overline{u_{EXP}}_a^{t-2} & \text{If (experimented \& } a^{t-1} \neq a) \\ \mathrm{calcAvg}(\overline{u_{EXP}}_a^{t-2}, u^{t-1}) & \text{If (experimented \& } a^{t-1} = a) \end{cases}$$

$\overline{u}^{t-1} = \frac{1}{(t-1)}(\overline{u}^{t-2}(t-2) + u^{t-1})$
if $r < \max(0, (\epsilon - t\lambda\epsilon))$ **then**
$experimented = \textit{TRUE}$
$q_a^t = 1/|A| \qquad \forall a \in A$
else
$experimented = FALSE$
$\forall a \in A : \overline{r}_a^t = \overline{u_{EXP}}_a^{t-1} - \overline{u}^{t-1}$
$\forall a \in A : q_a^t = [\overline{r}_a^t]_+ \,/ \sum_{a' \in A} \left[\overline{r}_{a'}^t\right]_+$
If denominator equal to zero, $\boldsymbol{q}^t$ is set to a uniform distribution over A
end if
end if
$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$

3.3.3 Incremental Conditional Regret Matching with weight λ

Incremental conditional regret matching with weight λ (ICRM) is a regret minimization model that incorporates the more sophisticated notion of conditional regret. As described earlier, a player has conditional regret if there exists an ac-

tion pair (x,y) so that he would have received a higher payoff when he would have played x in all periods where he actually played y. ICRM has been presented by Young (2004) as a computational efficient alternative to the version proposed by Foster and Vohra (1999) by removing the need of solving linear equations in each time period. Instead a form similar to some of the reinforcement learning models has been used (3.5), where a probability increment δ is used, calculated using the average conditional regret matrix $\overline{\boldsymbol{r}}$ along with the previous probabilities $\boldsymbol{q}^{t-1}$ (3.6). The complete outline of the algorithm can be seen at Algorithm 11.

$$\boldsymbol{q}^t = \boldsymbol{q}^{t-1} + \lambda \boldsymbol{\delta}^{t-1}, \quad \lambda > 0 \tag{3.5}$$

$$\delta_a^{t-1} = \sum_{h \in A} q_h^{t-1} \left[\overline{r}_{ha}^{t-1}\right]_+ - q_a^{t-1} \sum_{h \in A} \left[\overline{r}_{ah}^{t-1}\right]_+ \quad \forall a \in A \tag{3.6}$$

If the behavior of the algorithm is analyzed, similar conclusions as with the unconditional regret models can be reached. As proven by Young (2004), if the weight λ has been chosen appropriately small and all players use the learning model, the empirical action distributions will converge to the set of correlated equilibria. Additionally, using again a sufficiently small weight λ, it can be guaranteed that the conditional regrets become non-positive, even when other players are using other learning models. ICRM thus converges to a stronger set of equilibria compared to the previous unconditional regret minimization models. It does however depend on information about the actions chosen by the other opponents and his own payoff function thereby limiting his practical potential.

Algorithm 11 Incremental conditional regret matching with weight λ (ICRM)

$u \leftarrow$ Payoff function
$\boldsymbol{q} \leftarrow$ Probabilities
$\boldsymbol{r}^t \leftarrow$ Conditional regret matrix in period t
$r^t_{(a_x,a_j)} \leftarrow$ the gain (possibly negative) that player would have realized by playing action a_j instead of a_x
$\overline{\boldsymbol{r}}^t \leftarrow$ Average conditional regret matrix through to time t
$\boldsymbol{\delta}^t \leftarrow$ Vector analogous to the reinforcement increment
$a^t \leftarrow$ Action chosen at period t
$a^t_{-i} \leftarrow$ Actions chosen by the opponent('s) at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\lambda \leftarrow$ Constant geometric rate > 0
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

if $t > 1$ **then**
  $\forall a_x, a_j \in A : r^{t-1}_{(a_x,a_j)} = u(a_j, a^{t-1}_{-1}) - u(a_x, a^{t-1}_{-i})$
  $\overline{\boldsymbol{r}}^{t-1} = \frac{1}{t-1}(\overline{\boldsymbol{r}}^{t-2}(t-2) + \boldsymbol{r}^{t-1}) \in R^k$
  for all $a \in A$ **do**
    $\delta^{t-1}_a = \sum_{h \in A} q^{t-1}_h \left[\overline{r}^{t-1}_{ha}\right]_+ - q^{t-1}_a \sum_{h \in A} \left[\overline{r}^{t-1}_{ah}\right]_+$
  end for
  $\boldsymbol{q}^t = \boldsymbol{q}^{t-1} + \lambda \boldsymbol{\delta}^{t-1}$
end if
$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$

3.3.4 HM Conditional Regret Matching with inertia

Hart and Mas-Colell (2000) proposed another model in this category, called HM conditional regret matching with inertia (HMCRM). Inertia (Newton and Chittenden, 1848) here refers to the fact that in every time step the previously chosen action has more chance of being chosen again. The model thus embeds a resistance to a change in strategy.

More precisely, if an agent chose action j in the previous period, his probability distribution will be calculated as in Equations 3.7 and 3.8 where the latter defines the probability of choosing the same action and the first defines the probability of all other actions h, proportional to their nonnegative average conditional regret $\left[\overline{r}^{t-1}_{jh}\right]_+$ relative to the action j just played.

$$q_h^t = \epsilon \left[\overline{r}_{jh}^{t-1}\right]_+ \qquad \text{for all } h \neq j \tag{3.7}$$

$$q_h^t = 1 - \epsilon \sum_{h \neq j} \left[\overline{r}_{jh}^{t-1}\right]_+ \qquad h = j \tag{3.8}$$

Given a rather low ϵ value, this results in a high probability of choosing the same action with an occasional action change. A significant difference with all the previously seen models is the fact that instead of a smooth changing behavior, HMCRM will thus either show no change or a quite abrupt one, this because all the probabilities are calculated relative to the last chosen action j.

Using Blackwell's approachability theorem (Blackwell, 1956), Hart and Mas-Colell (2000) showed that like ICRM, HMCRM also converges to the set of correlated equilibria if all players play according to it. However unlike the previous regret minimization models, this last condition is also required for the regrets to become non-positive. Compared to the previous regret minimization models, this makes HMCRM less robust as non-positive regrets cannot be fully guaranteed, not even in the long term, if not all players use this same model (Young, 2004). Like ICRM (Subsection 3.3.3), HMCRM depends on information about the opponents' actions and the own payoff function thereby only classifying it as an uncoupled learning model.

Algorithm 12 HM conditional regret matching with inertia (HMCRM)

$u \leftarrow$ Payoff function
$\boldsymbol{q} \leftarrow$ Probabilities
$a^t \leftarrow$ Action chosen in period t
$\epsilon \leftarrow$ Predefined probability $0 < \epsilon < 1$
$\boldsymbol{r}^t \leftarrow$ Conditional regret matrix in period t
$r^t_{(a_x,a_j)} \leftarrow$ the gain (possibly negative) that player would have realized by playing action a_j instead of a_x
$\overline{\boldsymbol{r}}^t \leftarrow$ Average conditional regret matrix through time t
$a^t \leftarrow$ Action chosen at period t
$a^t_{-i} \leftarrow$ Actions chosen by the opponent('s) at period t
$u^{t-1} = u(a^{t-1}) \leftarrow$ Payoff from previous time step $t-1$
$\mathrm{Choice}(\boldsymbol{q}) \leftarrow$ Using probability distribution $\boldsymbol{q}$, an action is randomly chosen

---

if $t > 1$ **then**
$\forall a_x, a_j \in A : r^{t-1}_{(a_x,a_j)} = u(a_j, a^{t-1}_{-1}) - u(a_x, a^{t-1}_{-i})$
$\overline{\boldsymbol{r}}^{t-1} = \frac{1}{t-1}(\overline{\boldsymbol{r}}^{t-2}(t-2) + \boldsymbol{r}^{t-1}) \in R^k$

$$q^t_a = \begin{cases} \epsilon \left[\overline{r}^{t-1}_{(a^{t-1},a)}\right]_+ & \text{for all actions } a \neq a^{t-1} \\ 1 - \epsilon \sum_{h \neq a^{t-1}} \left[\overline{r}^{t}_{(a^{t-1},a)}\right]_+ & \text{for action } a = a^{t-1} \end{cases}$$

end if
$a^t = \mathrm{Choice}(\boldsymbol{q}^t)$

3.4 Others

This section introduces some other types of learning models covering a completely different underlying strategy. In these models, agents aim to predict the opponent's behavior in order to be able to reply with the best possible action. This is in severe contrast with the previous models in which the agents only try to assess how well their own actions are, whether they use cumulative payoffs (Algorithm 1), Q values (Algorithm 6) or the notion of regret (Algorithm 9) to achieve this. Because of this inherent dependency on the opponents, together with the often more complicated nature, these models are only introduced shortly and compared with the previously mentioned models.

3.4.1 Calibrated forecasting

While the previous models focus on the past and the agent's own actions and corresponding payoffs, forecasting rules are designed to make a forecast of the opponent's future actions. This forecast is then countered using the agent's best response. Because of the information needs, access to the opponents' actions and the payoff matrix is required in the process.

To verify the validity of the forecast, the concept of calibration is applied. This stipulates that if the distributions of a series of forecasts and the actual outcomes are close together, the forecast is calibrated and thus correct. The use of calibration is important in the sense that it is impossible to validate a single forecast. In fact in reality the same is true. A weather forecast predicting a 75% chance of sun is not necessarily wrong if no sun is observed. Similarly, it is not automatically correct if sun can indeed be observed.

The convergence potential of calibrated forecasting rules with regard to strategic learning has been explored by Foster and Vohra (1997). They showed that if all players use such rule, and given some preconditions, the empirical frequency distributions converge to the set of correlated equilibria. However given the convergence potential, Abernethy and Mannor (2011) recently argued that efficiency in these forecasts is still a serious drawback. Although rules with efficient calibration have recently been introduced, these are only applicable to binary situations and not generally extendable to other situations.

3.4.2 Fictitious play (FP)

Possibly the most well known and most researched of these other learning rules is fictitious play (FP). This adaptive heuristic, first proposed by Brown (1951) and studied by Robinson (1951), is also known as the Brown-Robinson learning process. Although the rule is conceptually as easy as CPM, fictitious play, just like calibrated forecasting, cannot be classified as a fully distributed learning model as it requires information about the opponents' chosen actions. However FP still maintains its status as an uncoupled learning model as the information requirements do not go as far as the opponents' received payoffs.

Informally, in each round when FP is used, the agents will play their best response against the empirical action distribution of the opponents. In this sense, the term fictitious play can be considered misguiding as an agent's play is still based on the actual played actions of the opponents. FP thus uses a simple extrapolation of the opponent action history to counter it with his best response.

The main drive behind this model was the initial believe that following the empirical distributions would ultimately lead to Nash equilibrium convergence. While this has been shown to be the case in 2x2 games (Miyazawa, 1961) and several other classes of games, games with no general convergence have also been discovered, among which the game introduced by Shapley (1964), since then known as Shapley's game (Subsection 2.5.1). An overview of work related to this research can be found at Berger (2007).

3.4.3 Bayesian Learning

In Bayesian learning, the approximation of the opponents' strategies is done using prior and posterior probabilities together with Bayes's theorem (Bayes and Price, 1763). In order for this to work, each player initially starts with a prior belief on the opponents' strategies. These priors are then updated after each round depending on the opponents' actions. When actions have to be chosen, these priors will be used to develop a strategy that maximizes the expected payoff in the future.

In order for Bayesian learning to converge to Nash equilibria however, prior knowledge on the opponents is required. Without this knowledge, Bayesian learning cannot be guaranteed to perform qualitatively well (Koulovatianos and Wieland, 2011). Because of the rigorous updating of beliefs using prior and posterior probabilities, Bayesian learning is argued to be among the most rational behaving learning models.

3.4.4 Hypothesis testing

As in Bayesian learning, hypothesis testing's main aim is to construct a valid representation of the opponents' strategies in order to be able to counter them. However, while Bayesian learning uses the Bayes's theorem for this purpose, hypothesis testing uses a method related to the concept of cognitive learning.

Schematically this relates to the following. At the start of the game, each player will adopt a provisional model of the opponents' strategies, called a hypothesis. This hypothesis is then tested against the perceived data and kept if the data complies. In the other case, the hypothesis is rejected and the process starts again.

In contrast with Bayesian learning, Foster and Young (2003) have shown that, given some conditions on the parameters, the players' responses converge with certainty to the set of Nash equilibria of the game.

3.5 Summary

This thesis focuses on those adaptive heuristics requiring as few information as possible about the environment of the agents. For most of the considered learning models, this means that knowledge on the opponents is not even required. These models, the fully distributed or completely uncoupled models, only base their learning behavior on their own chosen actions and corresponding payoffs. The lack of opponent dependencies and their inherent simplicity offers great possibilities with a wide spectrum of applications. However, in contrast with the more advanced models in Subsection 3.4, their convergence behavior cannot be fully predicted and theoretically guaranteed.

Two main classes of models have been introduces, being the reinforcement learning models and the models on the concept of minimizing regret. While they both share some characteristics, their main strategy is fundamentally different. The reinforcement learning models include CPM and Q-learning and different extensions and improvements and are all fully distributed. The regret minimization models can be subdivided in two subclasses depending on their use of respectively unconditional or conditional regret. While for the first subclass, regret estimation techniques have been invented allowing the creation of fully distributed regret minimization models like RMϵ, this is currently not the case for the latter subclass and might perhaps be even impossible because of the increased complexity of this type of regret. The regret minimization models using conditional regret thus are only uncoupled in the sense that they do require the observation of the opponents' actions.

After an overview of these two classes of learning models, some other more advanced learning models have also been introduced. For these models, the main strategy, in contrast with the previous models which evaluate the agent's own actions, is to try to predict the opponents' strategies in order to counter them with their best possible action. The selection of the best counter move is in that sense only of secondary importance. Because of this more sophisticated way of working, these models all have important theoretical characteristics regarding convergence. However, for some of these convergence properties to apply, information on the opponents is crucial and in practice not always feasible.

Chapter 4

Experimental results

This chapter elaborates on the results of the adaptive heuristics, introduced in the previous chapter (Sections 3.2 and 3.3), applied to a set of sample games as summarized in Subsection 2.5.3. While the emphasis is on empirical convergence behavior, other aspects as memory requirements, computational efficiency and sensitivity to initial conditions are also reviewed.

The chapter starts with a short section on the memory and computational efficiency of the different models, covering both a theoretical and empirical aspect. Following are sections regarding the behavior of the specific models. With regard to the review of the convergence behavior, the models are run with different variable settings and the results are related to the Nash equilibrium and other solution concepts introduced in chapter 2. If available, this empirical behavior is compared to the theoretical learning model characteristics and other known empirical results. To minimize the effect of random dependencies, each experiment consists of a number of independent runs. To avoid confusion, the learning models are treated in the same order as they were introduced in Chapter 3.

4.1 Memory and computational efficiency

This section shortly analyzes the different learning models with regard to memory and computational efficiency in order to come to an overall judgment of the value of the models.

In order to come to a clear overview, the most obvious method is to characterize the different models using big O notation (Bachmann, 1894) depending on the function characteristics in each time step for a given player. This way, the overview can be made with respect to the total number of actions a. As shown in

Table 4.1, this leads to the conclusion that in fact all models, with the exception of ICRM and HMCRM, have both a memory usage and runtime performance in the order of $O(a)$. This because of the inherent simplicity of these models, each containing only a sequence of simple functions, and the severely limited use of history.

Algorithm	Memory usage	Runtime performance
CPM	$O(a)$	$O(a)$
CPM-A	$O(a)$	$O(a)$
CPM-RE	$O(a)$	$O(a)$
CPM-BS	$O(a)$	$O(a)$
Qs	$O(a)$	$O(a)$
Qϵ	$O(a)$	$O(a)$
FAQs	$O(a)$	$O(a)$
FAQϵ	$O(a)$	$O(a)$
RM	$O(a)$	$O(a)$
RMϵ	$O(a)$	$O(a)$
ICRM	$O(a^2)$	$O(a^2)$
HMCRM	$O(a^2)$	$O(a^2)$

Table 4.1: Memory usage and computational specifications of the different learning models in big O notation with respect to the number of actions a.

While the $O(a)$ behavior serves as the lower limit for all these models, already caused by the manipulation of the player's probabilities in each time step, the two models using the concept of conditional regret, ICRM and HMCRM, increase this up to $O(a^2)$. In fact, the use of conditional regret requires these models to update an entire matrix of regrets, in contrast with the models using unconditional regret who only require an update of a vector of regrets, an operation similar to the update rule used by reinforcement models.

In practice however applied on games with just 2 actions like the 1B prisoner's dilemma, this difference is not that significant and in some cases even negligible. Figure 4.1 for example shows the average runtime of these models when applied to the 1B prisoner's dilemma game. The average runtime in this case is averaged over 100 experiments of which each experiment consists of 1000 independent runs of 1000 time steps long. Although these results give an indication of the more efficient models, models like CPM, CPM-A, Qϵ and RMϵ, and the slower models like CPM-BS and the two models with conditional regret, the differences remain rather small.

With a minimum standard deviation of 0.2 seconds, most differences cannot be claimed to be statistically relevant. In practice, most of the tested models can in this case be considered equally time efficient given an equal experiment duration. The lenient versions of these learning models have in these tests not been included as they are identical except for a short initial stage which, over 1000 time steps, is computationally not relevant.

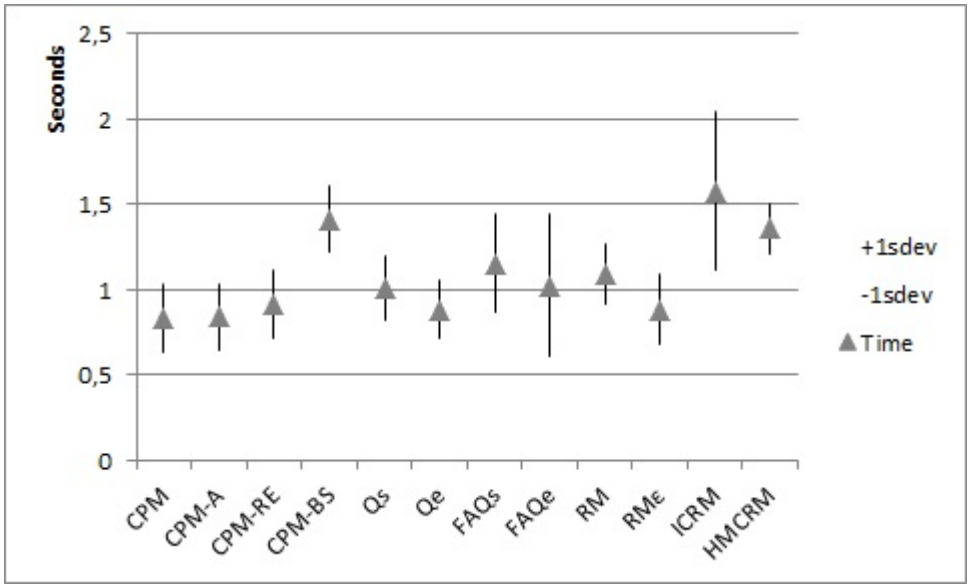


Figure 4.1: The average runtime of a learning model experiment applied to the 1B prisoner's dilemma game along with an indication of the standard deviation. The runtime data has been measured and averaged over 100 experiments. Each experiment covers 1000 independent runs of each 1000 time steps long.

While a similar experiment could also be performed regarding memory usage, because of the design of the test framework as a data gathering and visualization tool and therefore not built for specific memory efficiency, this experiment would not allow for an appropriate memory efficiency comparison. It is however expected that the memory usage with respect to 2x2 normal formal games does not differ significantly because of the limited amount of history usage and the rather small number of actions.

4.2 Cumulative payoff matching

4.2.1 The basic framework: CPM

Because of the lack of adaptable parameters, the basic CPM model has the advantage of being both easy to set up and consistent in its convergence behavior. Using a simple proportional method of calculating probabilities, CPM can be considered one of the conceptually most simplistic of all considered learning models.

General convergence: class 1 and 2 games

However, this simple proportional method also has one major disadvantage in the sense that mistakes in the early exploration keep having effect even after a few hundred time periods. The consequences of this technique can be seen in the experiments of all subclass A games. While these games have a consistent convergence to the Nash equilibrium (Figure 4.2), a full 100% convergence can often not be reached as also the other actions keep a small probability of getting chosen causing a small ratio of exploration. Although this impact keeps decreasing, while CPM on 1A and 2A games already reach a 90% Nash convergence after respectively 29 and 19 time periods, even after 1000 time periods only convergence rates of 99.2% and 99.1% can be observed. For the 3A games, both matching pennies and Shapley's game, a sequence of action changes can be observed for every run causing the empirical action distribution to converge to the Nash equilibrium.

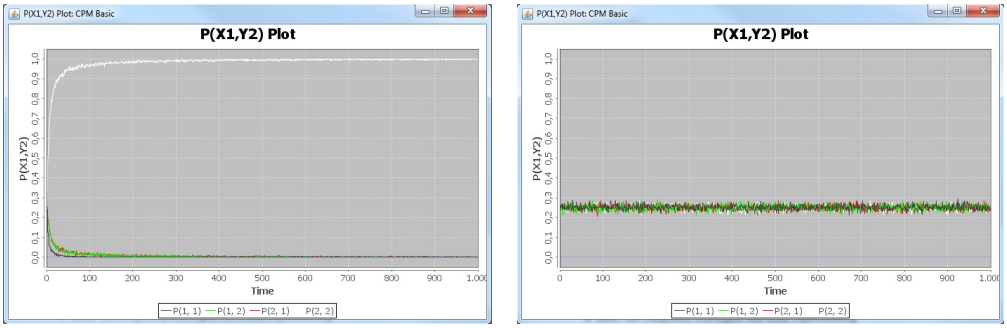


Figure 4.2: The rate of convergence towards Nash outcomes averaged over 1000 independent runs. CPM on the 1A deadlock game (NE=white, l.) and the 3A matching pennies game (r.)

For the subclass B games however, convergence is not that consistent. Even for the 1B prisoner's dilemma, a game having dominant strategies for both players, consistent convergence towards the NE can only be observed after more than

100000 time periods. While for this game, most independent runs immediately learn to play the NE (approximately 76.9% after 1000 time periods), another group of runs initially converges towards the Nash dominating outcome. However, since this outcome is not an equilibrium and therefore unstable, players are attracted to the other action. In a typical run as in Figure 4.3 (right), it can be observed that player 1 is the first to switch to action 2 (causing a CPM value crossover around time period 20000). From there on, player 1 has chosen to play his dominant action for the rest of the game causing his CPM value corresponding to action 2 to keep rising. However, although player 2 receives less payoff after the action switch of player 1, because of his previous large preference for action 1, only after several thousand time periods, this player switches to an action 2 preference causing Nash convergence. Similar behavior can be observed in other runs on the 1B prisoner's dilemma and can be confirmed using the textual game data.

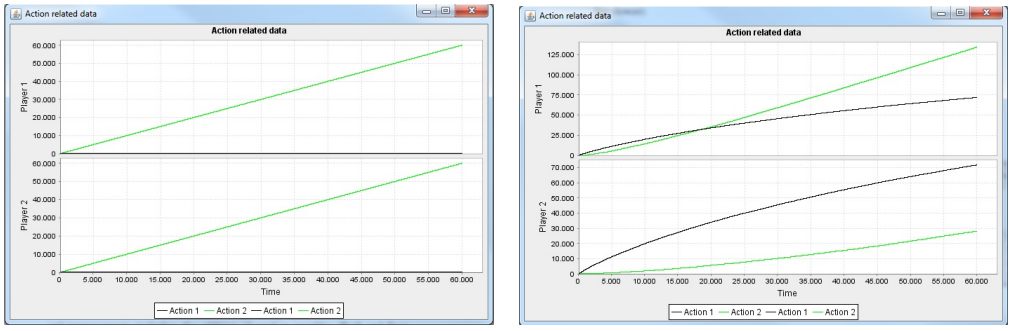


Figure 4.3: The differences in the evolution of the CPM values for two runs of CPM on the 1B prisoner's dilemma. An immediate convergence towards the NE (for both players green, l.) vs. an initial convergence towards the Nash dominating outcome followed by final convergence behavior towards NE (r.).

A similar slow convergence can also be observed for the 2B chicken game. However, because of the absence of dominant strategies, a relatively small percentage (approximately 20% in our experiments) is able to converge to the Nash dominating outcome, given that a sufficient preference was built up early in the run by random fluctuations (Figure 4.4 left). In the other case, when this preference was not built up early (Figure 4.4 right), convergence is turned towards one of the Nash outcomes.

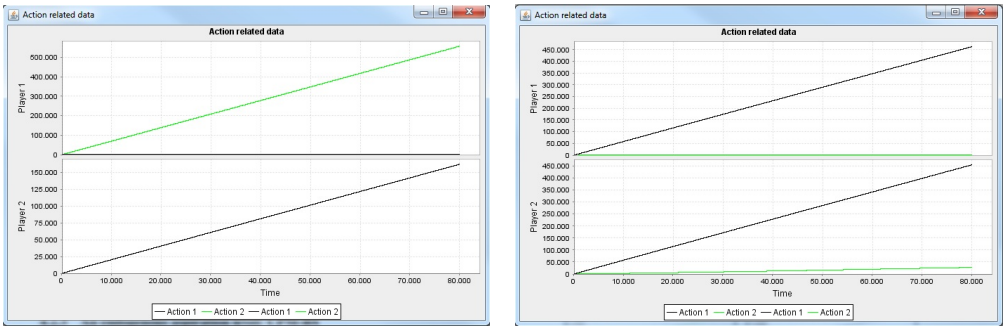


Figure 4.4: The differences in the evolution of the CPM values for two runs of CPM on the 2B chicken game. An ultimate convergence towards the NE (l.) vs. an ultimate convergence towards the Nash dominating outcome (r.)

Dynamics of class 3 games

However, when applied to games of the 3B class, a different convergence behavior occurs. While the previous subclass B games still have Nash convergence, this cannot immediately be observed with the 3B spoiled child game. In fact, while a minority of the runs (16% in our experiments) converge to the Nash dominating outcome (Figure 4.5 top left), the others are characterized by many action changes and slowly changing probability distributions and preferences (Figure 4.5) caused by the unstable nature of the game without any pure equilibria. Although their distributions might eventually converge to those of the NE, because of the slow changes in action preference (at most twice per million time periods), empirically these cannot be classified as NE converging.

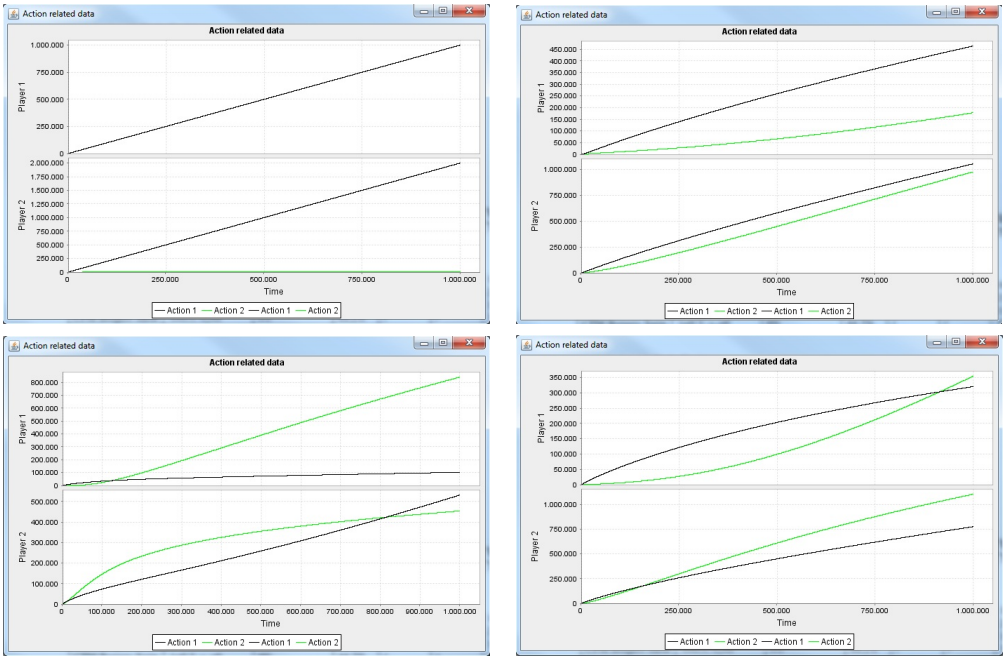


Figure 4.5: The differences in the evolution of the CPM values for two runs of CPM on the 3B spoiled child game. An ultimate convergence towards the Nash dominating outcome (top left) vs. an rather unstable result with action switches (others)

Although the chosen action distributions of CPM applied to the 3A games ultimately converges to the Nash equilibrium, a view on the evolution of the CPM values (Figure 4.6) shows that when an action change occurs, especially applicable for Shapley's game, the action CPM selects is in fact the best reply against the opponent's action. However, these action changes are preceded by long periods in which a single action has the preference. Although in these periods, exploration still occurs causing the other actions to be chosen as well, this is limited. Applying CPM on 3A games can thus lead to situations in which the game history is significantly in the favor of one player rather than equal as would be expected. In the long run however, the action distributions do converge to the Nash equilibrium, however the number of time periods required for this exceeds the one million mark and is thus in practice not desirable.

Figure 4.6: The differences in the evolution of the CPM values for two runs of CPM on the 3A games, matching pennies (l.) and Shapley's game (r.)

4.2.2 Manipulating the learning curve: CPM-A and CPM-RE

Building on the basic framework introduced by CPM, two other models, CPM-A and CPM-RE, allow for a more advanced manipulation of the learning curve. For this purpose, a number of parameters are introduced. For CPM-A, these are the parameters C and p, for CPM-RE, these are the geometric rate λ and the maximum random perturbation w_{max}.

Parameter settings

After examination of the two parameters of CPM-A, C and p, both having an influence on the learning curve, the conclusion can be drawn that for most games, the parameter value combination in fact can be called crucial for the convergence behavior. While for rather simple games as the 1A deadlock and the 2A battle of the sexes game, these differences are still small, for other games like the 1B prisoner's dilemma and 2B chicken game, different parameter value combination can actually yield different convergence (Figure 4.7). Especially adapting the C value can change the action preferences towards the Nash convergence as observable in the figure at the bottom two rows. However, while the p value does not significantly change the final convergence result, it does have an important influence on the time required for convergence. In combination with high C values, a different p value can even lead to a difference between empirical observable Nash convergence (Figure 4.7 third row) showing convergence after at most 5000 time periods and the experiments both high C and high p showing convergence after several hundred thousand time periods (Figure 4.7 fourth row).

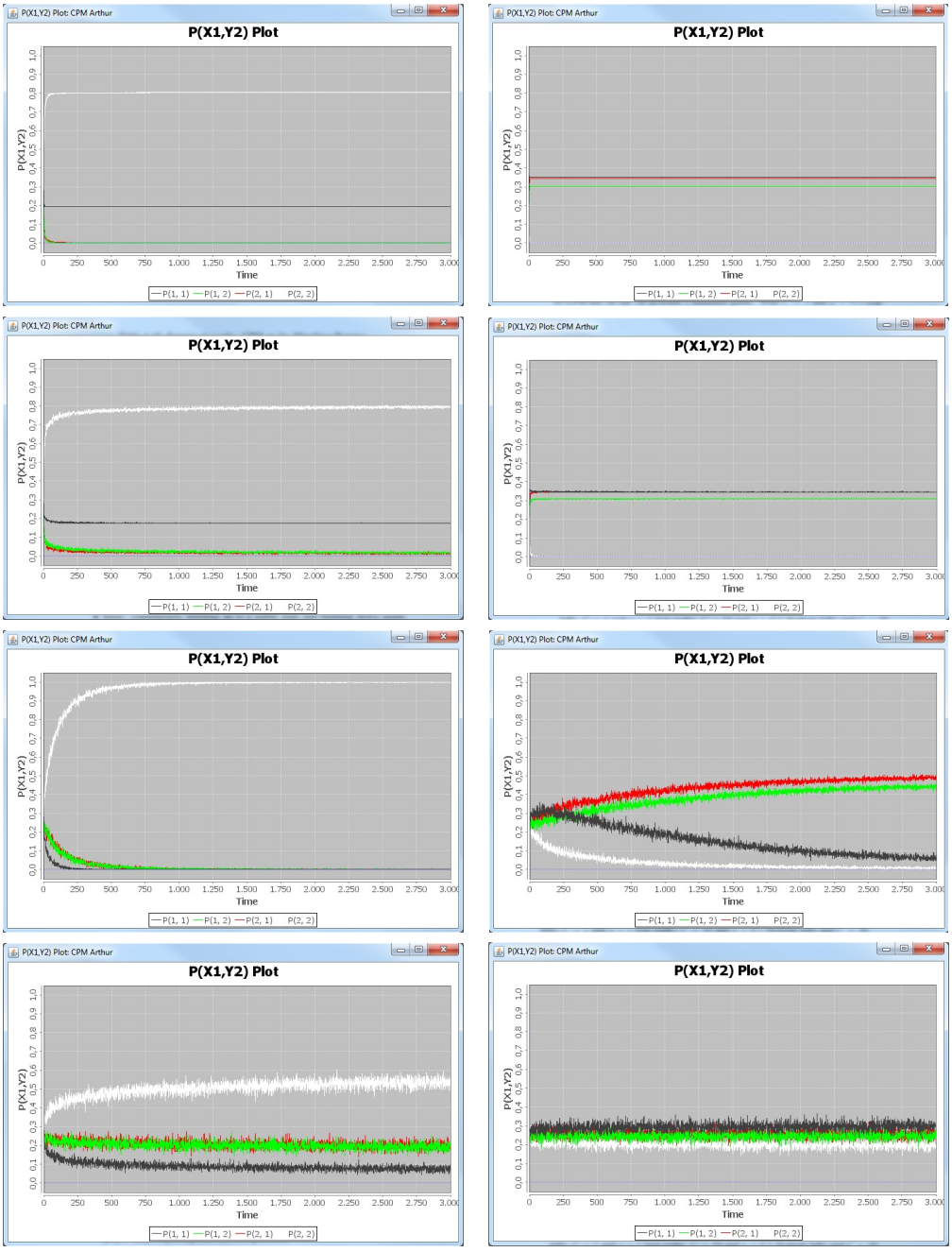


Figure 4.7: The rate of convergence towards the different outcomes of CPM-A on the 1B prisoner's dilemma game (left, Nash=white) and the 2B chicken game (right, Nash=red/green). With $C = 1$ and $p = 0.5$ (first row), $C = 1$ and $p = 1$ (second row), $C = 10(l.)$ or $20(r.)$ and $p = 0.5$ (third row) and $C = 10(l.)$ or $20(r.)$ and $p = 1$ (fourth row). Averaged over 500 independent runs of 3000 time periods.

For CPM-RE the parameters for the geometric rate λ and w_{max} are kept at respectively 0.9 and 0.05. By using this value of λ, CPM allows for both the possibility of learning from past play and maintaining a slight preference for recent play. Although for several games, all subclass A and the 1B prisoner's dilemma game, different values do not yield significantly different results, for games like the 2B chicken game and the 3B spoiled child game, lower λ values are found to steer convergence more towards the Nash dominating outcome (black curve at Figure 4.8).

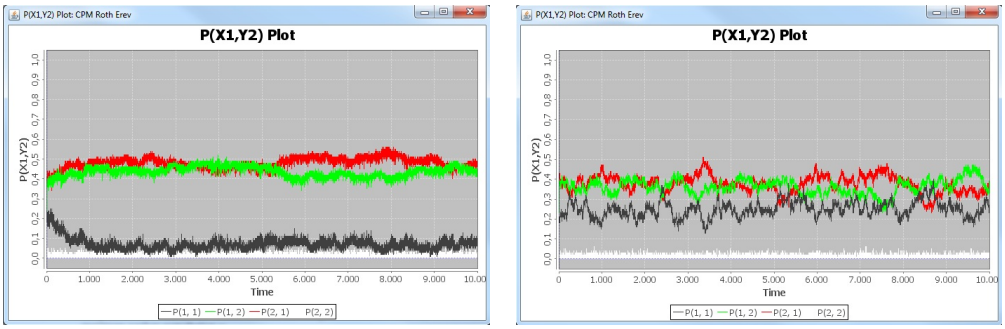


Figure 4.8: The rate of convergence towards the different outcomes (Nash=green and red) of CPM-RE on the 2B chicken game with $\lambda = 0.9$ (l.) and $\lambda = 0.5$ (r.). $w_{max} = 0.05$. Averaged over 100 independent runs.

As for the w_{max} value, our experiments have shown that even low values as 0.05 can introduce both the advantage of trembles while still maintaining a rather consistent convergence without too many exploration steps. Higher values like 0.5 for example can lead to behavior as shown in Figure 4.9 where each run only plays the Nash outcome about 80% of the time. Although each run still plays the NE, identical to runs with w_{max} values of 0.05, the increased random exploration lowers the average realized reward significantly.

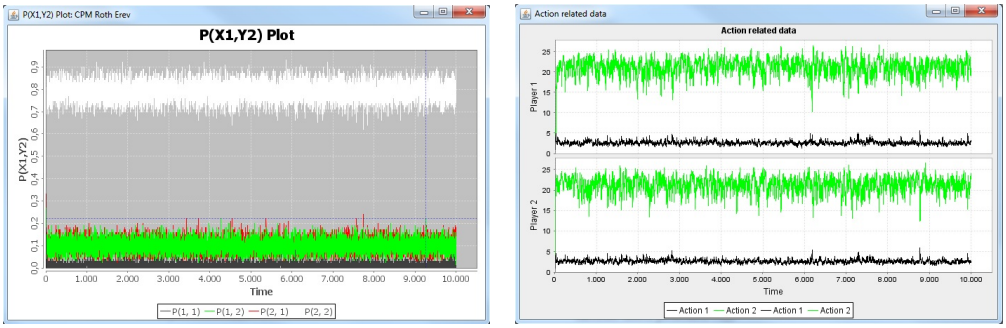


Figure 4.9: The rate of convergence towards the different outcomes (Nash=white) averaged over 100 independent runs. CPM-RE on the 1A deadlock game (l.) along with the evolution of the CPM values of a typical run (r.). Parameters: $\lambda = 0.9, w_{max} = 0.5$

General convergence: 1A and 2A

For those games with pure NE and no other outcome dominating this equilibrium, convergence, both for CPM-A and for CPM-RE, is almost consistently toward the NE. Only with parameter settings aiming for fast convergence, both low C and p for CPM-A and a zero w_{max}, a small percentage of runs gets stuck in the inferior $(1, 1)$ outcome of the game (Figure 4.10). For the runs of CPM-RE with positive w_{max} though, because of the random perturbations, a small percentage of experimentation is also retained.

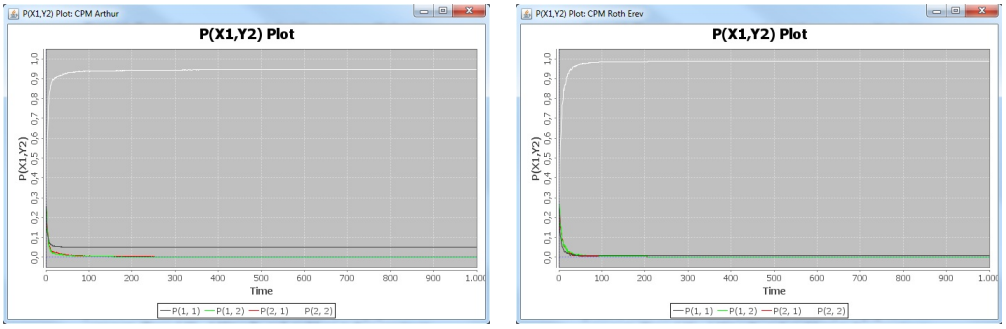


Figure 4.10: The rate of convergence towards the different outcomes (Nash=white) of CPM-A with $C = 1$ and $p = 0.5$ (left, 94.6% Nash) and CPM-RE with $\lambda = 0.9$ and $w_{max} = 0$ (right, 98.9% Nash) on the 1A deadlock game.

General convergence: 1B and 2B

While for CPM-A, the parameter settings have a great influence on the final convergence, either to the Nash outcome or the Nash dominating outcome (Figure 4.7), in case of CPM-RE, the differences between the different parameter settings, and in particular with or without random perturbations are rather small (Figure 4.11).

For this model, the presence of random perturbations is shown to push convergence more towards the NE, both for the 1B as for the 2B game, with respective Nash convergence, runs that converge towards the Nash outcome, of 100% and 96% . Comparing this to the runs without perturbations with only a Nash convergence of 96% and 83%, the presence of random perturbations thus has a positive effect on Nash convergence, however with the drawback of continuous exploration.

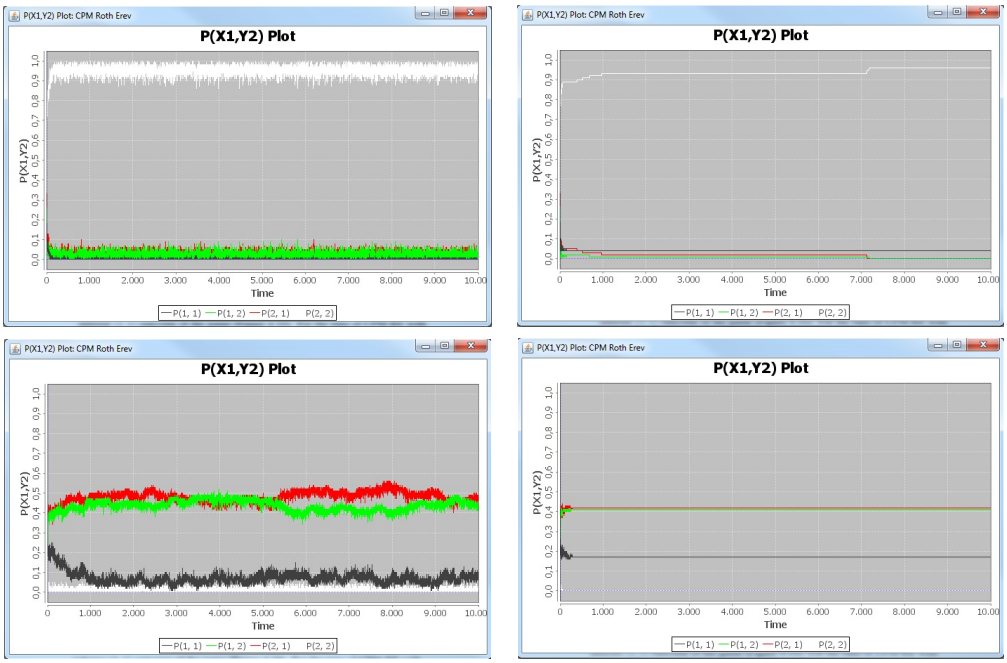


Figure 4.11: The rate of convergence towards the different outcomes of CPM-RE on the 1B prisoner's dilemma game (top, Nash=white) and the 2B chicken game (bottom, Nash=red/green) with $\lambda = 0.9$ and $w_{max} = 0.05$ (l.) or $w_{max} = 0$. Averaged over 100 independent runs.

General convergence: class 3 games

A similar behavior can be observed for the 3A matching pennies and Shapley's game. Although the empirical action distributions of CPM-A show to converge to the NE in the long run for all reasonable parameter settings, for runs of only a few thousand time periods, this behavior cannot always be observed because of the time between the different action changes. While low C values lead to degenerate distribution in favor of one player (Figure 4.12 left), higher C values allow for a more consistent action distribution towards the NE with faster action changes.

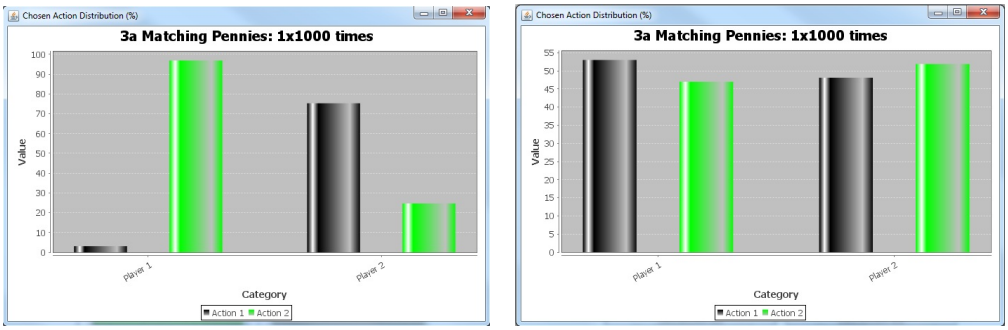


Figure 4.12: The empirical action distributions of one run of CPM-A on the 3A matching pennies game with $C = 1$ and $p = 0.5$ (l.) and with $C = 10$ and $p = 0.5$ (r.)

For CPM-RE, convergence similar to the earlier CPM can be observed. However, while for CPM, long periods of identical outcomes could be observed, this is not the case for CPM-RE. Because of the small random perturbations, exploration is maintained causing fast action changes and action distributions converging to the NE (Figure 4.13) can be observed and this for all tested 3A games.

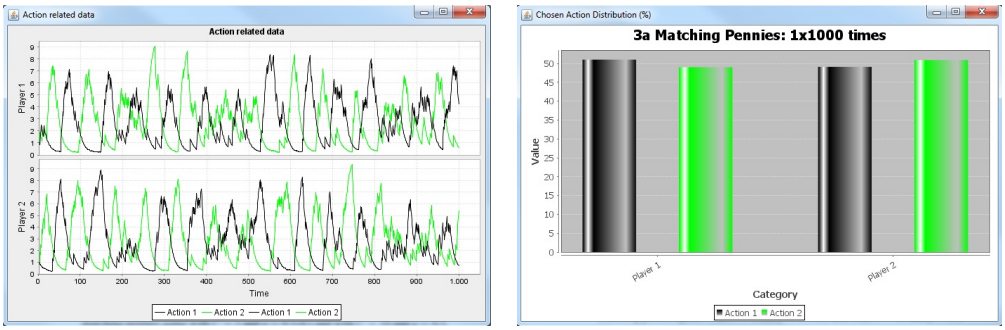


Figure 4.13: The evolution of the regret values of one run of CPM-RE on the 3A matching pennies game with $\lambda = 0.9$ and $w_{max} = 0.05$ (l.) and the corresponding action distribution (r.)

Additionally, although the empirical action distributions for both CPM-A, when applied with high C values, and CPM-RE for the 3A Shapley's game consistently converge towards the NE, a different period to period behavior can actually be observed. Similar to that of CPM, both CPM-A and CPM-RE aim to select their best-reply action against the opponent in case of an action switch. This leads to situations in which for the 3A Shapley's game the six rewarding outcomes are visited with a much higher probability than the other zero rewarding outcomes which are only visited occasionally (Figure 4.14 left).

Especially for CPM-RE, this effect is in place almost immediately. This particular situation with higher probabilities for the rewarding outcomes, actually constitutes a correlated equilibrium of the game (Young, 2004, p34). Although also observable for CPM-RE, because of the high C value, this is only after about 10000 time periods.

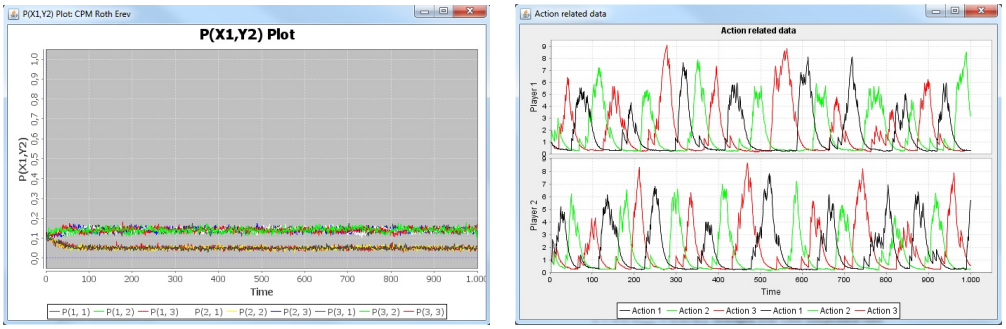


Figure 4.14: The rate of convergence towards the different outcomes on the 3A Shapley's game with CPM-RE (l.) and the evolution of the CPM values of one typical run (r.). Convergence rate averaged over 1000 independent runs.

When applied to the 3B spoiled child game, CPM-A shows a similar behavior with a certain sensitivity to the value of its parameters. When applied with a high C value, the empirical distributions converge again to those of the NE while lower C values force convergence towards the Nash dominating outcome (and in a few runs even towards one of the other three outcomes). CPM-RE however shows a different convergence steering runs towards the Nash dominating outcome. Because of the continuous random perturbations however, the other outcomes are visited as well. The lower the w_{max} parameter, the fewer those other outcomes are visited (Figure 4.15), although without random perturbations, some runs get stuck in the suboptimal outcome $(2, 2)$.

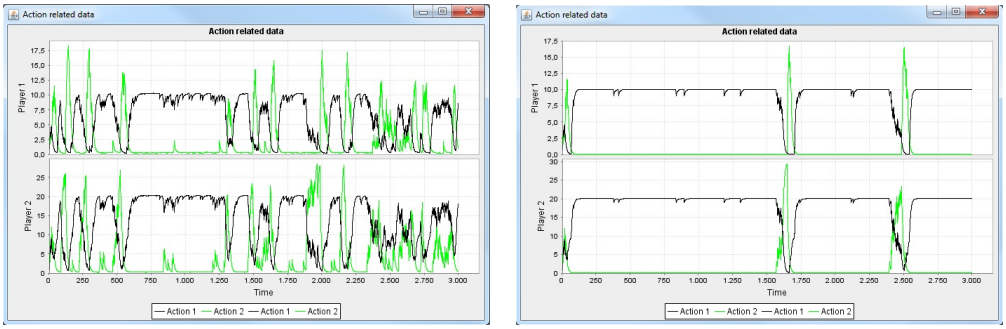


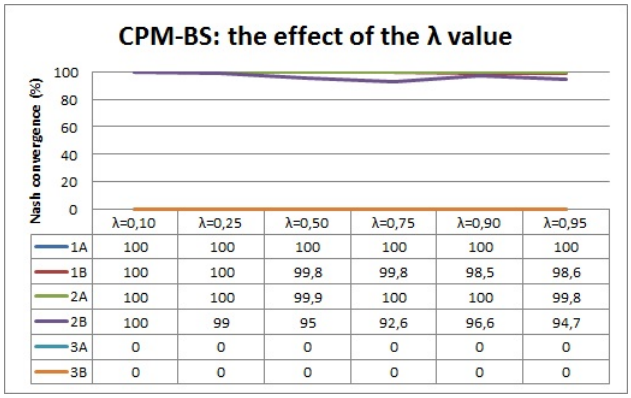
Figure 4.15: The differences in the evolution of the CPM-RE values for two runs of CPM-RE on the 3B spoiled child game, with $w_{max} = 0.05$ (l.) and $w_{max} = 0.005$ (r.). Remaining parameter: $\lambda = 0.9$

4.2.3 An endogenous aspiration level: CPM-BS

CPM-BS introduces another addition to the general CPM framework: the addition of an aspiration level with a value based on past play. Actions above the level are reinforced, action beneath it are impaired.

Parameter settings

For this purpose, CPM-BS maintains two parameters who's values can be set: the discount factor λ and a value to influence the initial height of the aspiration level u^0. Experiments with different parameter settings do not reveal any significant changes. Especially for the discount factor λ, the final convergence is quite resistant to changing values (Figure 4.16). Although smaller values like 0.1 allow for a slightly better convergence, they also lengthen the convergence behavior (Figure 4.17). Because of this increased duration and the only slight increase, the value of λ has been chosen to be 0.9. For the value of u_0 it has been observed that overly optimistic or pessimistic value like 5 or -5 have a dramatic influence on the convergence rate by almost randomizing it. A neutral value of 0 on the other hand shows consistent and well-performing results and is therefore used in all following experiments.



	λ=0,10	λ=0,25	λ=0,50	λ=0,75	λ=0,90	λ=0,95
1A	100	100	100	100	100	100
1B	100	100	99,8	99,8	98,5	98,6
2A	100	100	99,9	100	100	99,8
2B	100	99	95	92,6	96,6	94,7
3A	0	0	0	0	0	0
3B	0	0	0	0	0	0

Figure 4.16: The rate of convergence towards Nash outcomes averaged over 1000 independent runs of CPM-BS applied to games of the three main classes using a changing λ value. Other parameters of CPM-BS: $u^0 = 0$

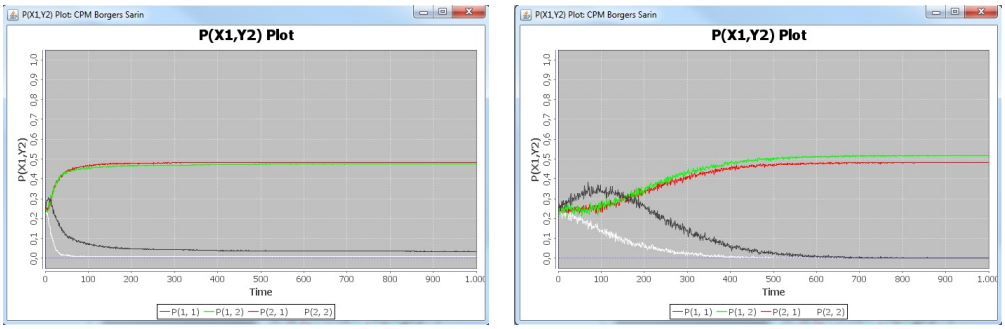


Figure 4.17: The rate of convergence towards the different outcomes on the 2B chicken game with CPM-BS with $\lambda = 0.9$ (l.) and $\lambda = 0.1$ (r.). Convergence rate averaged over 1000 independent runs.

General Nash convergence: class 1 and 2 games

In general, as visible in Figure 4.16, convergence for both class 1 and 2 games is highly aimed towards a full Nash convergence. However, when using higher λ values like 0.9, convergence is performed rather fast (as visible in Figure 4.17 left) causing in a limited amount of runs a suboptimal result. Although this can be the Nash dominating outcome for subclass B games, convergence is sometimes also observed towards less rewarding and even unstable outcomes like the (Cooperate, Defect) outcome for the 1B prisoner's dilemma. While this suboptimal convergence was not observed with the other CPM models, the general results still show

a high and desirable preference for the NE which was not always the case for the other CPM models.

Although given the analytical proofs by Karandikar et al. (1998), convergence to the Nash dominating outcome could have been expected, this is not the case. It can thus be assumed that the aspiration level of CPM-BS does not fulfill the requirements of a slowly and accurately adapted level.

Absence of mixed equilibria: class 3 games

However, this fast convergence to one single outcome is also present when applied to the class 3 games. For these games lacking pure equilibria, this means that a single run only converges to a single, non-Nash, outcome. However, although a single run does not exhibit Nash convergence, to which outcome the run converges is ultimately still dictated by the equilibrium characteristics. Similarly to CPM-A and CPM-RE, the probability to reach a certain outcome for the 3A matching pennies game follows the NE with a 25% probability for each outcome while the probability to reach a certain outcome for the 3A Shapley's game follows the CE in the sense that the six rewarding outcomes have a much higher probability than the three zero rewarding outcomes (Figure 4.18 left). Additionally, even for the 3A Shapley's game, the empirical action distributions collected over all 1000 runs converge to the NE as well (Figure 4.18 right). However, in contrast with the previous models, each run will only converge to one single outcome. A mixed equilibrium is thus never reached.

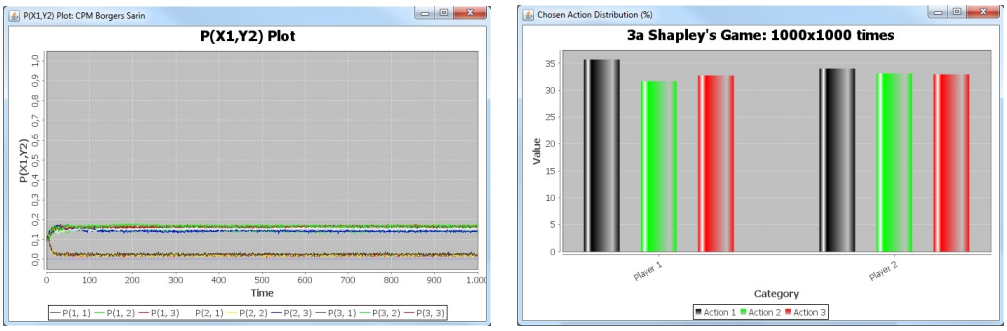


Figure 4.18: The rate of convergence towards the different outcomes on the 3A Shapley's game with CPM-BS and the corresponding empirical action distributions. Parameters: $\lambda = 0.9, u^0 = 0$. Convergence rate and action distribution averaged over 1000 independent runs.

Concerning the 3B spoiled child game, this phenomenon cannot be observed that clearly (Figure 4.19). Although the initial preference for the Nash dominating outcome $(1, 1)$ drops to a more reliable level after a normalization of the aspiration level, this outcome maintains its higher probability, followed by the outcome $(1, 2)$. Ultimately, identical to the 3A games, individual runs converge to only one of the four outcomes of the game.

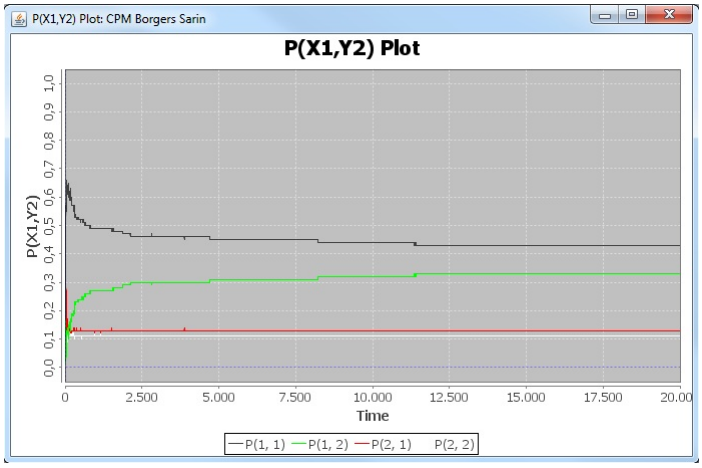


Figure 4.19: The rate of convergence towards the different outcomes on the 3B spoiled child with CPM-BS. Parameters: $\lambda = 0.9, u^0 = 0$. Convergence rate averaged over 100 independent runs.

4.2.4 Overview

This section reviewed four different models covering the concept of cumulative payoff matching. Although for all of these models, certain characteristics are identical, some of the more elaborate models have also shown to improve the efficiency of the basic CPM.

This basic CPM model, easy to set up because of the lack of parameters, has a rather consistent and fast Nash convergence ratio for subclass 1A and 2A games. However, because the influence of the initial exploration is not diminished, experimentation continues to occur causing the less desirable outcomes to be visited as well with a small probability, even after a few hundred time periods. For the remaining games, convergence behavior is more complicated. Because of the lack of a pure NE (3A games) or the existence of an outcome that dominates the NE (subclass B games), convergence can either take a long period (1B and 3A games) or turn towards other outcomes (2B and 3B games).

CPM-A tries to offer a solution for this inconsistent convergence behavior by allowing the manipulation of the learning curve. However, the introduction of the two new parameters to aid in this process introduces another complication. In fact, CPM-A has been shown to be highly sensitive to adapting these parameter values. Especially for the 1B and 2B games, this behavior is striking. However, CPM-A also allows for a more consistent convergence with regard to the class 3 games. Given a high C value, the empirical distribution of the class 3 games can be seen to converge to the NE. Additionally, for the 3A Shapley's game, after at least 10000 time periods, the final result actually constitutes a correlated equilibrium.

The addition of random perturbations in CPM-RE is another addition to CPM. Although for the subclass B games, runs can converge to the Nash dominating outcome, convergence is in general aimed towards the NE. With respect to the 3A games, the same conclusion can be drawn as with CPM-A. Even more, the duration between the action switches is much less compared to the previous models and therefore empirically more desirable. With respect to the 3B game class however, convergence is not observed towards the NE but rather to the Nash dominating outcome.

The fourth model, CPM-BS introduces the addition of an endogenous aspiration level, adapted throughout the run based on past play. Because of this, convergence is for all class 1 and 2 games aimed towards the NE. However, for a limited amount of runs, players can get stuck in other outcomes, both the Nash dominating outcomes and other less stable outcomes. Additionally, for class 3 games, each run has been shown to consistently converge to just one single outcome, so for individual runs, no Nash convergence can be perceived. An overview of the Nash characteristics can be found in Table 4.2.

Game classification						
	1		2		3	
	A	B	A	B	A	B
CPM	Yes	Yes/No	Yes	Yes/No	No	No
CPM-A	Yes	Yes/No	Yes	Yes/No	Yes/No	Yes/No
CPM-RE	Yes	Yes/No	Yes	Yes/No	Yes	No
CPM-BS	Yes	Yes/No	Yes/No	Yes/No	No	No

Table 4.2: Overview of the convergence characteristics of the different CPM models. For CPM-RE, only experiments with random perturbations ($w_{max} > 0$) are considered. For Nash convergence to be denoted by 'Yes', an upper limit of 50000 time steps is set.

4.3 Q-learning

In contrast with the previous models which can all be considered independent models, the models on Q-learning are connected through a more modular approach in which one model can be derived from another by changing one or two components (Figure 4.20). Additionally, each model can in these experiments be equipped with two different action selection mechanisms being the softmax method with Boltzmann distribution or ϵ-greedy. Because of this, each model will be discussed with regard to its added value compared to the more basic versions.

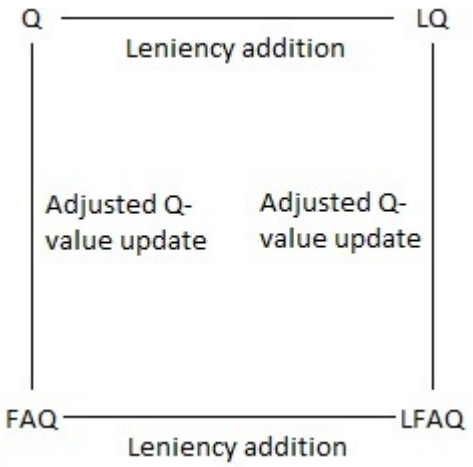


Figure 4.20: Relation between the Q-learning models

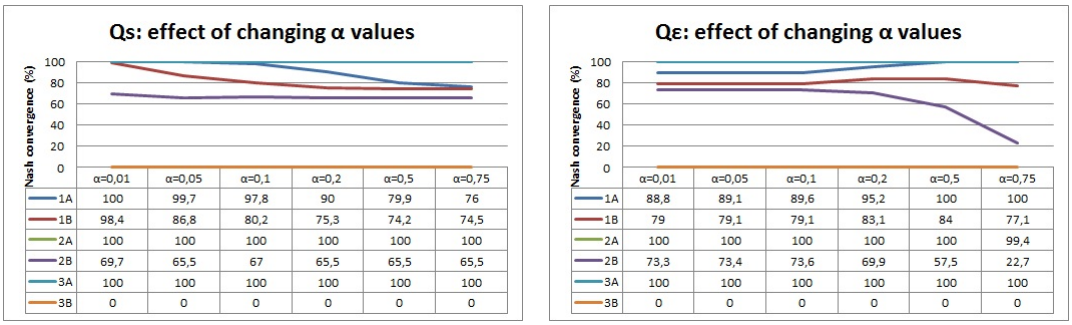
4.3.1 Basic Q-learning

Fixed α and γ parameters

While Qs and Qϵ contain a variety of parameters, two of the parameters they have in common are those belonging to the Q value update rule, the learning rate α and the discount factor γ, determining respectively the influence of the past results and the importance of future rewards.

As experiments using these models with different α and γ values (Figure 4.21 and 4.22) have shown no remarkable deviations, these parameters are kept constant in the following experiments at values of 0.1 for α and 0.9 for γ. Although in some games, other values have been found performing slightly better, these, often

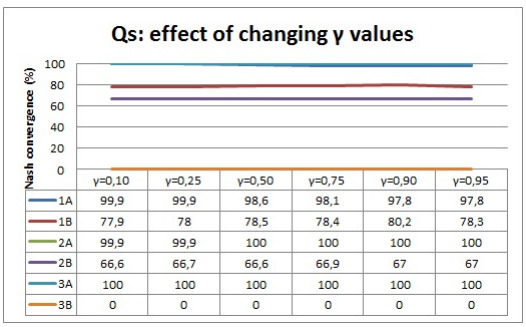
lower, values also induce a less desirable learning behavior by showing slower learning, in the case of α, or a lower importance of long term results in case of γ. Similarly, parameter values chosen excessively high can also result in less desirable behavior. For α, these high values can cause a significant drop in the Nash convergence rate caused by the higher focus on the most recent information and the neglecting of the past. For γ, the difference is not that striking, however a slight drop in Nash convergence rate can be observed with values approaching 1 closely as these values distort the balance between considering the current rewards and striving for a long term reward making the use of the discount factor γ ultimately obsolete. With the aforementioned constant chosen values however, the Q-learning models show acceptable results while the effect of past play is kept at a reasonable rate.



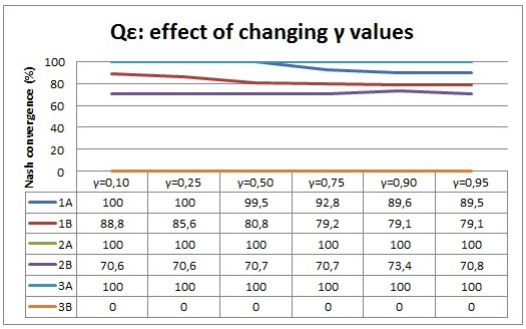
	α=0,01	α=0,05	α=0,1	α=0,2	α=0,5	α=0,75
1A	100	99,7	97,8	90	79,9	76
1B	98,4	86,8	80,2	75,3	74,2	74,5
2A	100	100	100	100	100	100
2B	69,7	65,5	67	65,5	65,5	65,5
3A	100	100	100	100	100	100
3B	0	0	0	0	0	0

	α=0,01	α=0,05	α=0,1	α=0,2	α=0,5	α=0,75
1A	88,8	89,1	89,6	95,2	100	100
1B	79	79,1	79,1	83,1	84	77,1
2A	100	100	100	100	100	99,4
2B	73,3	73,4	73,6	69,9	57,5	22,7
3A	100	100	100	100	100	100
3B	0	0	0	0	0	0

Figure 4.21: The rate of convergence towards Nash outcomes averaged over 1000 independent runs. Qs (l.) and Qϵ (r.) applied to games of the three main classes using a changing α value. Other parameters of Qs: $Q_0 = 0, \gamma = 0.9, \tau = 0.2$ and of Qϵ: $Q_0 = 0, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$



	γ=0,10	γ=0,25	γ=0,50	γ=0,75	γ=0,90	γ=0,95
1A	99,9	99,9	98,6	98,1	97,8	97,8
1B	77,9	78	78,5	78,4	80,2	78,3
2A	99,9	99,9	100	100	100	100
2B	66,6	66,7	66,6	66,9	67	67
3A	100	100	100	100	100	100
3B	0	0	0	0	0	0



	γ=0,10	γ=0,25	γ=0,50	γ=0,75	γ=0,90	γ=0,95
1A	100	100	99,5	92,8	89,6	89,5
1B	88,8	85,6	80,8	79,2	79,1	79,1
2A	100	100	100	100	100	100
2B	70,6	70,6	70,7	70,7	73,4	70,8
3A	100	100	100	100	100	100
3B	0	0	0	0	0	0

Figure 4.22: The rate of convergence towards Nash outcomes averaged over 1000 independent runs. Qs (l.) and Qϵ (r.) applied to games of the three main classes using a changing γ value. Other parameters of Qs: $Q_0 = 0, \alpha = 0.1, \tau = 0.2$ and of Qϵ: $Q_0 = 0, \alpha = 0.1, \epsilon = 0.2, \lambda = 0.005$

General convergence

If we only consider Qϵ and Qs, the basic Q-learning algorithms, we can immediately draw the conclusion that these Q-learning models do not have a truly consistent convergence behavior in the sense that for most games, independent runs can actually yield different final results (as observable in Figures 4.21 and 4.22). Especially for the 1B, prisoner's dilemma, and 2B, chicken, games this phenomenon is striking. For these games, where another (non-Nash) outcome exists that dominates the Nash outcome, it depends on the initial random fluctuations whether the model converges to the Nash outcome or the Nash dominating outcome. In the experiments as listed in Figures 4.21 and 4.22, the rate Nash convergence is often found lower than 80%. Remarkable is here also the convergence behavior of the 3B game, with a complete convergence to the Nash dominating outcome, both for Qs as for Qϵ.

When applied to subclass A games however, a clear preference for the Nash outcome can be observed, and this for all games of class 1, 2 and 3, with Nash convergence results generally in the neighborhood of 100%. For these subclass A games, changing parameter settings thus also has a much lower influence. In fact, changing parameter settings, unless they decrease exploration enough to remain stuck in suboptimal outcomes, only has an minor influence on the time required to converge to (one of) the Nash outcome(s). These results are consistent with the convergence results reported by Kaisers and Tuyls (2010) regarding 2A and 3A games for Qs and consistent with those by Wunder et al. (2010) for Qϵ (without λ factor) with regard to all tested games even for the 3B spoiled child game.

Increased exploration: the Q_0 value

This convergence behavior, especially for the subclass B games, is however not that robust against changing parameter values. In some games even, the wrong parameter settings can change the convergence behavior quite drastically. While in Q-learning there are several parameters that are able to influence its behavior, the most effective one of these is the initial Q value Q_0.

As mentioned earlier, a common practice is to initiate Q-learning with a rather high Q value. By increasing this value, the learning model is initially forced to act greedy thereby allowing for an initial exploration stage. This effect can be clearly observed when we consider the evolution of the Q values as in Figure 4.23. This figure shows the evolution of the Q values of a single run of Qϵ applied to the prisoner's dilemma with an initial very high Q value of 100. As can be observed, over time the Q values are decreased to a more stable level after which they stabilize. In this downward motion however, as shown in Figure 4.24, the Q values, corresponding to the two actions of the players, continuously switch position. This is caused by the greedy behavior of ϵ-greedy. This states that the action with the highest Q value has more chance of getting selected causing the Q value of this action to be decreased, lower than that of the other action. This dynamic thus causes an initial exploration stage in which all the actions are chosen and explored regularly.

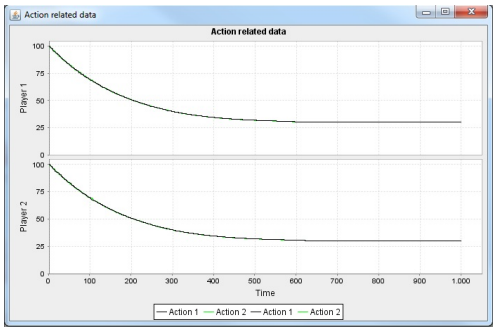


Figure 4.23: The evolution of Q values in a single run of Qϵ on the prisoner's dilemma with $Q_0 = 100$. Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$

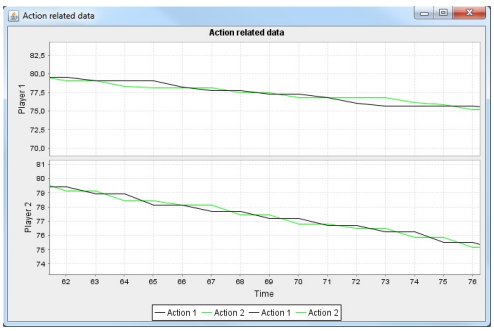


Figure 4.24: Partial enlarged image of Figure 4.23 showing the period-to-period Q value dynamics.

Which Q_0 values can be classified as high however depend on the particular game and its payoffs. While in Figure 4.23, the Q values converge to a value of

30, this is not the case for all games. In general, this final level s_f depends on the received payoffs (after convergence, this is payoff u_c) and the value of the λ factor as shown in Equation 4.1. It can be confirmed that for the 1B prisoner's dilemma with a run converging to $(1, 1)$ $(= (C, C))$, this final Q level is 30 as visible in Figure 4.23. For runs converging to the NE however, this level then only is 10 while for games like the 2B chicken game, this level turns out to be 60 or 70 depending on convergence to either NE or the Nash dominating outcome.

$$s_f = \sum_{k=0} u_c \gamma^k = \frac{u_c}{1-\gamma} \tag{4.1}$$

As expected, this initial exploration phase can in some cases have significant consequences on the convergence results. Considering Qϵ on both the 1B prisoner's dilemma and 2B chicken games, Figures 4.25 and 4.26 show how a higher Q value, respectively 30 and 50 (empirically verified to be high enough to induce this behavior), can drastically change the convergence towards the Nash dominating outcome. For the 1B prisoner's dilemma, this effect is even more prominent as with higher Q_0 values, convergence can be turned from 79% Nash convergence towards a full 100% to the Nash dominating outcome. Increasing the initial Q values can thus indeed change the convergence behavior and, by leading convergence to the, better rewarding, Nash dominating outcome, increase the average realized payoff.

Determining the ideal value for this purpose is however not always that straightforward. Ideally, the Q_0 value is chosen on or just above the final stable Q-level. This level however highly depends on the game itself. While Qϵ on the prisoner's dilemma suffices to have a Q_0 value of 30 (Figure 4.25), on the chicken game this should already be increased to 50 to induce the same behavior (Figure 4.26).

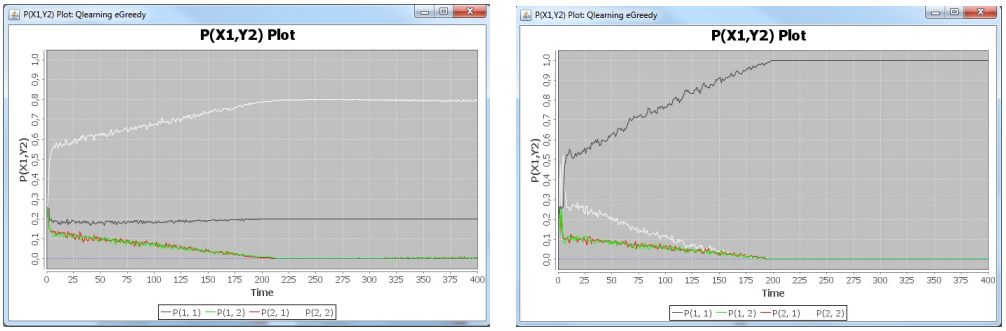


Figure 4.25: The differences in convergence behavior with Qϵ on the 1B prisoner's dilemma given different Q_0 values ($Q_0 = 0$ left and $Q_0 = 30$ right). Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$. Averaged over 1000 runs of 400 time periods with NE=white.

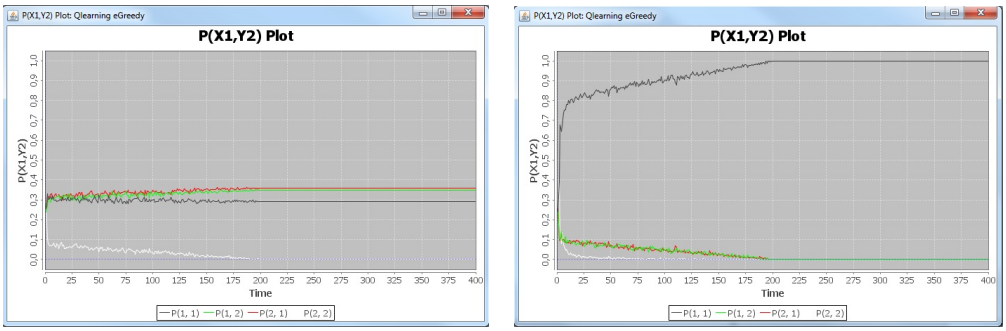


Figure 4.26: The differences in convergence behavior with Qϵ on the 2B chicken game given different Q_0 values ($Q_0 = 0$ left and $Q_0 = 50$ right). Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$. Averaged over 1000 runs of 400 time periods with NE=green/red.

Additionally, the presence of this sensitivity to the initial Q_0 value also depends on the particular action selection mechanism. Using Qs on the 1B prisoner's dilemma for example, a complete convergence to the Nash dominating outcome cannot be achieved, not even with a Q_0 of 100 which only achieves a convergence of approximately 50% to the Nash dominating outcome (Figure 4.27 left). This partial convergence is caused by the application of the softmax action selection mechanism along with the presence of dominant strategies. Since the softmax method in part allocates probabilities proportionally and not just greedy like ϵ-greedy, exploration towards both actions is maintained for a large number

of iterations. The final convergence then depends on random fluctuations steering the dynamics to either the Nash dominating outcome, the outcome with the highest rewards, or towards the NE, the dominant strategy. When Qs is applied on the 2B chicken game without dominant strategies on the other hand, the exact same behavior is observed as with Qϵ as again a full 100% convergence to the Nash dominating outcome is observed (Figure 4.27 right).

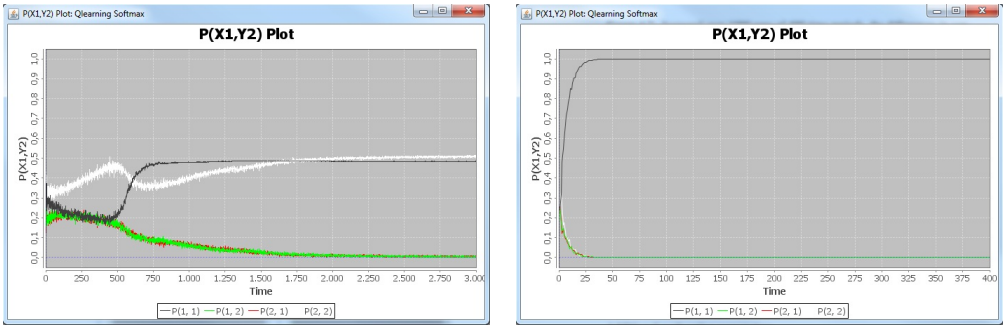


Figure 4.27: The differences in convergence behavior with Qs on the 1B prisoner's dilemma (l.) and 2B chicken game (r.) given different Q_0 values ($Q_0 = 100$ left and $Q_0 = 50$ right). Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \tau = 0.2$. Averaged over 1000 runs of respectively 400 and 3000 time periods with NE=white (l.) and green/red (r.).

Finally, while larger Q_0 values do in fact have a positive effect on the rate of convergence, although not always towards the Nash outcome, initial Q values that have been chosen too high also have a major disadvantage caused by the length of the normalization period, i.e. the time it takes for the Q values to reach more appropriate levels. As can be observed in any game from the 1A deadlock (Figure 4.28) game up to the 3B spoiled child game (Figure 4.29), higher Q_0 values increase the time it takes to reach a level of full convergence. For games of the 3B class, because of their unstable nature, a high Q_0 values additionally introduces a relatively long period of almost random behavior (Figure 4.29) before finally converging to the Nash dominating outcome.

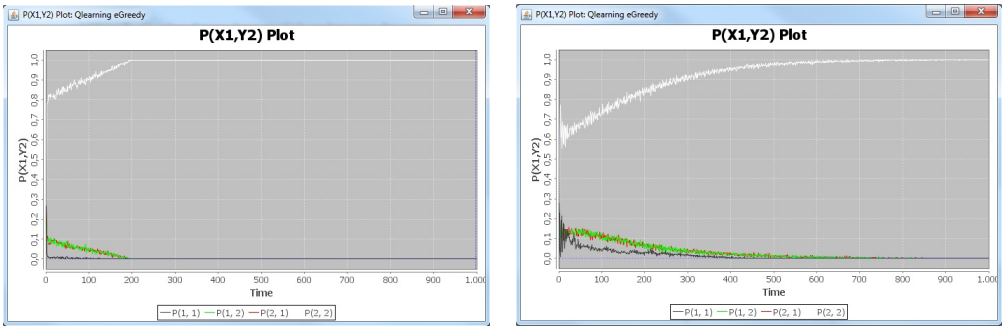


Figure 4.28: The frequencies to reach a certain outcome. Qϵ on the 1A deadlock game with $Q_0 = 10$ (l.) and $Q_0 = 30$ (r.). Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$. Averaged over 1000 runs of 1000 time periods.

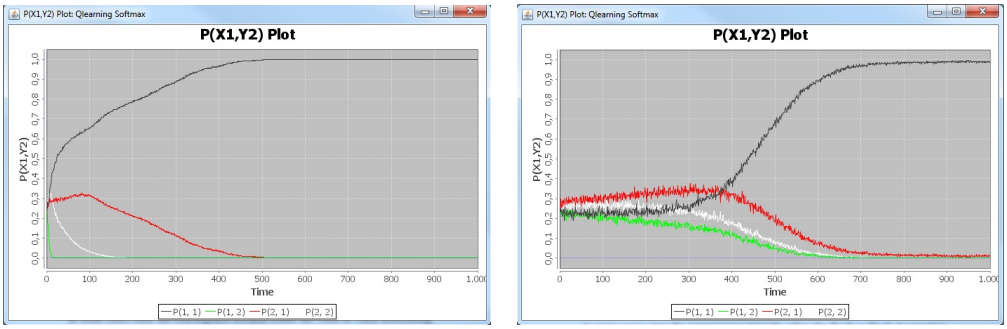
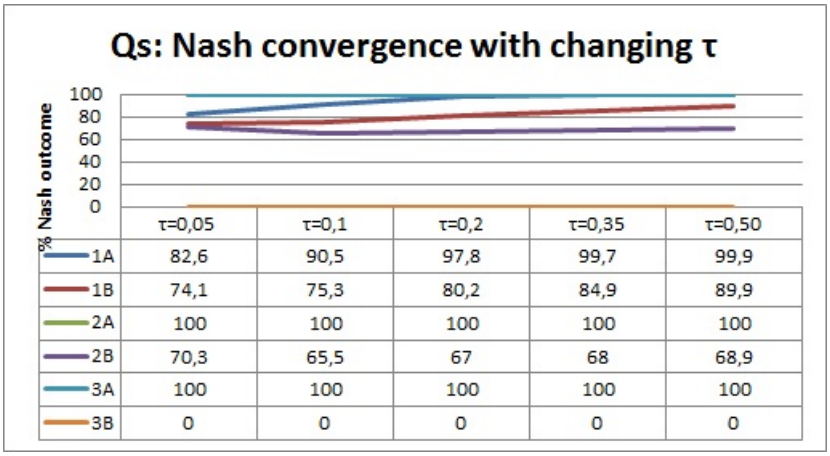


Figure 4.29: The frequencies to reach a certain outcome. Qs on the spoiled child with $Q_0 = 0$ (l.) and $Q_0 = 30$ (r.). Remaining parameters: $\alpha = 0.1, \gamma = 0.9, \tau = 0.2$. Averaged over 1000 runs of 1000 time periods.

Additional exploration parameters

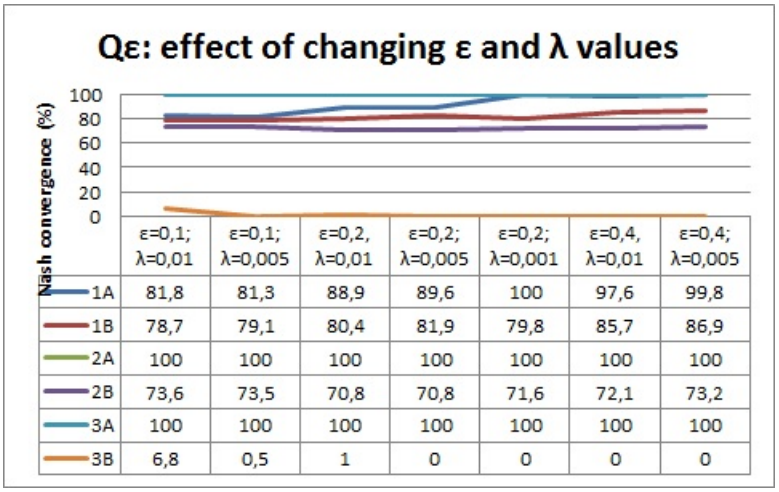
Besides the initial Q value there are some other parameters that can be linked with the degree of exploration. These are the parameters τ for Qs and for Qϵ ϵ and λ which determines the decay of ϵ. However, in contrast with the Q_0 value which in some cases steer the convergence towards either the Nash or Nash dominating outcome, the τ only has a minor influence on the convergence results as observable in Figure 4.30. While lower τ values as 0.05 make the model act more greedy, higher values like 0.5 make the model more equiprobable. In previous and further experiments therefore a value of 0.2 is found to both perform well and form a balance between greedy and equiprobable behavior.



	τ=0,05	τ=0,1	τ=0,2	τ=0,35	τ=0,50
1A	82,6	90,5	97,8	99,7	99,9
1B	74,1	75,3	80,2	84,9	89,9
2A	100	100	100	100	100
2B	70,3	65,5	67	68	68,9
3A	100	100	100	100	100
3B	0	0	0	0	0

Figure 4.30: Qs with different τ values. Plotted is the frequency that the Nash outcome is selected for both class 1, 2 and 3 games. Remaining parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9$. Averaged over 1000 runs of each 1000 time steps.

Similar to Qs, the parameters ϵ and λ only have a slight influence on the rate of convergence as can be observed in Figure 4.31 showing the convergence results with different parameter combination ranging from a low experimentation with $\epsilon = 0.1$ and $\lambda = 0.01$ up to configurations with a higher level of experimentation and ϵ values of 0.4. Although a slower decline of exploration as with $\epsilon = 0.2$ and $\lambda = 0.001$ in some cases shows a slight increase in Nash convergence (except for the 1B prisoner's dilemma), these settings introduce a long convergence time (1000 time periods because of $\lambda = 0.001$). As a fast level of convergence is in practice preferred, the standard parameter combination for Qϵ is $\epsilon = 0.2$ and $\lambda = 0.005$. This allows for a rather quick convergence, in which each time the ϵ value is lowered with 0.5% (reaching a zero level after 200 time periods) while still maintaining a well performing model. While an ϵ value of 0.4 is also a valid option with almost equal results except for the 1B prisoner's dilemma game, the choice was made in favor of 0.2 because of the superior results in the initial stage of the model (Figure 4.32). Ultimately however, the results of both parameter configurations do not outrun each other by much and a choice can be made depending on whether the initial stage or the long run is most crucial.



	ε=0,1; λ=0,01	ε=0,1; λ=0,005	ε=0,2, λ=0,01	ε=0,2; λ=0,005	ε=0,2; λ=0,001	ε=0,4, λ=0,01	ε=0,4; λ=0,005
1A	81,8	81,3	88,9	89,6	100	97,6	99,8
1B	78,7	79,1	80,4	81,9	79,8	85,7	86,9
2A	100	100	100	100	100	100	100
2B	73,6	73,5	70,8	70,8	71,6	72,1	73,2
3A	100	100	100	100	100	100	100
3B	6,8	0,5	1	0	0	0	0

Figure 4.31: Qϵ with different ϵ and λ values. Plotted is the frequency that the Nash outcome is selected for both class 1, 2 and 3 games. Remaining parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9$. Averaged over 1000 runs of each 1000 time steps.

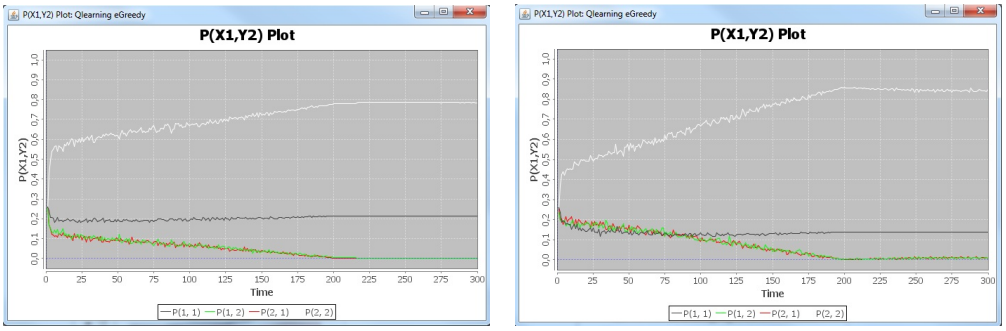


Figure 4.32: The frequencies to reach a Nash outcome. Qϵ on the 1B prisoner's dilemma game with $\epsilon = 0.2$ (l.) and $\epsilon = 0.4$ (r.). Remaining parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \lambda = 0.005$. Averaged over 1000 runs of 300 time periods.

Dynamics of 3A: matching pennies and Shapley's game

While all runs of the class 1, 2 and 3B games converge towards a pure outcome, either the Nash outcome or the Nash dominating outcome (for the subclass B games), the 3A games show a more unstable convergence. While their empirical distributions do in fact converge to the Nash distribution, for matching pennies a $50/50$ action distribution between their actions, because of the unstable nature, both agents continuously switch actions and thereby return different outcomes.

The underlying dynamics behind this switching behavior can be explained by using the evolution of the Q values.

On a typical run of Qϵ, the Q values on the 3A game matching pennies evolve according to Figure 4.33. In this figure it can be observed that for both players, the Q values corresponding to the different actions sequentially go up and down. Particularly interesting however are the switching moments as for example around time period 500 indicated with the red line. At that time period, player 1's Q value of action 2 has gone below the Q value of action 1 thereby triggering an action switch. Simultaneously, player 2's Q value of action 1 peeks and starts diminishing as the action change of player 1 suddenly makes this action rewarding zero payoff. This behavior is then continuously repeated and characterized with rather long periods (approximately 270 time periods) in which both players play the same action before one of them switches.

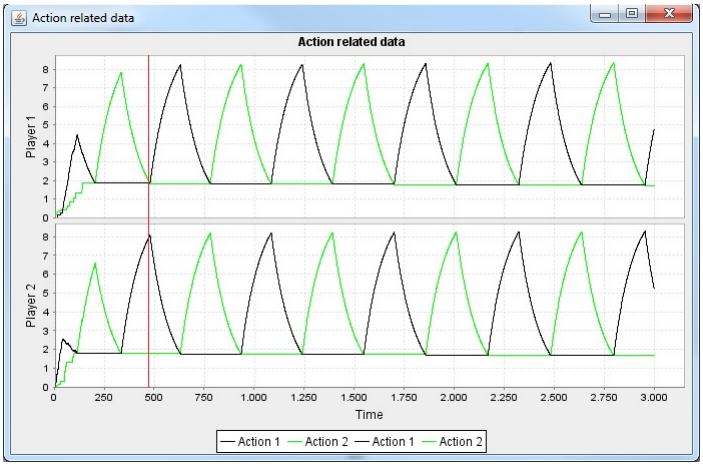


Figure 4.33: The evolution of Q values on a run of Qϵ on the 3A matching pennies game. Parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$.

When applied to the other 3A game, Shapley's game, the exact same behavior occurs. However, because of the presence of three actions, an interesting observation can be made. While the empirical distributions still converge to the Nash distribution, in this case each action is chosen about $1/3$ of the time, not all outcomes are visited equally in the process. As can be seen in Figure 4.34 (l.), when an action switch occurs, the switching agent selects his best response action against the opponent's action. Because of this, the zero-rewarding states in the game are hardly played in favor of the six rewarding outcomes as can be seen in Figure 4.34 (r.). This result, where each of the six rewarding outcomes is visited with a high probability and the remaining zero rewarding outcomes with a much

lower probability, in fact constitutes a correlated equilibrium (CE) (Young, 2004, p34). Because of this particular behavior, the realized payoffs are actually higher than expected given just the Nash characteristics (approximately 0.44 against the expected 0.33).

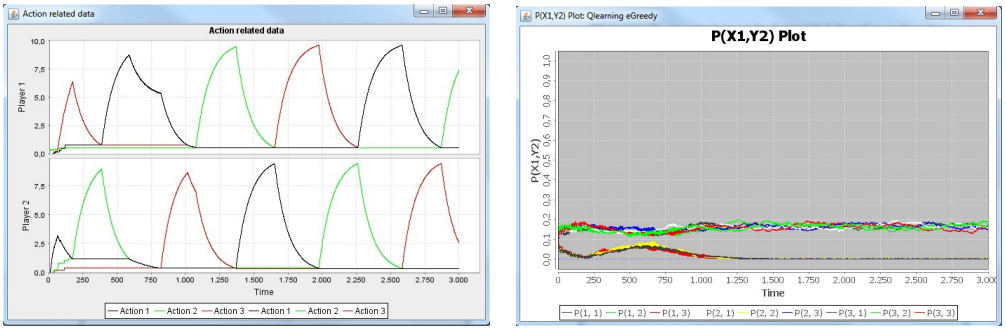


Figure 4.34: The evolution of Q values on a run of Qϵ on the 3A shapley's game (l.). The frequencies the different outcomes are chosen, averaged over 1000 runs. Parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$.

Similar results can be found when running the Qs model. However, where the action switches of Qϵ occur after a constant distance (since the exploration has diminished), this is not the case as can be observed in Figure 4.35. For these runs, because the softmax method keeps exploring, the Q values keep converging to a more stable level thereby decreasing the time between action switches. The same behavior can also be observed in runs of Qϵ when the λ factor is chosen zero in order to maintain exploration.

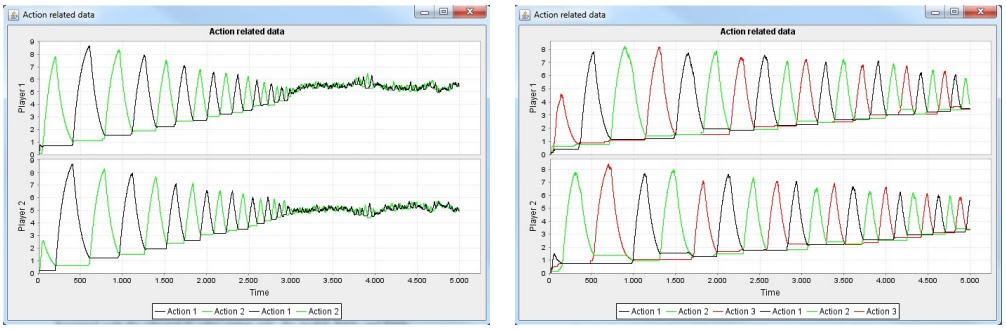


Figure 4.35: The evolution of Q values on a run of Qs on the 3A matching pennies game (l.) and shapley's game (r.). Parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \tau = 0.2$.

4.3.2 Frequency adjusted Q-learning

Equipped with the adjusted Q value update rule, the models FAQϵ and FAQs (and by extension also the variants with leniency) perform more consistently compared to their counterparts with the basic update rule. In fact, in all cases where the models are initialized with an initial Q value of zero and sufficiently low α and β parameters, the Nash equilibrium can be reached. Even for the rather unstable 3B spoiled child game, which previously consistently converged to the Nash dominating outcome, a full Nash convergence can be achieved. Additionally, where in the case of the 3A games, only the empirical action distributions of the basic Q-learning models converged to the NE, for both FAQs and FAQϵ, Nash convergence can also be observed in period to period behavior. For FAQϵ however, an extra condition is in place as the λ factor is to be chosen zero. With a positive λ value, experimentation terminates and FAQϵ again converges to the Nash dominating outcome.

Figure 4.36 for example shows a run of FAQs and FAQϵ on the 1B prisoner's dilemma using α and β parameters of 0.01. For both models, a convergence behavior towards the Nash outcome can be observed. However, even after 10000 time periods, this convergence is not terminated. Full 100% convergence takes an additional 30000 time periods for FAQs and 10000 time periods for FAQϵ. As the same behavior is also reported by Kaisers and Tuyls (2010) and Bloembergen et al. (2010), it can be concluded that FAQ consistently converges to the Nash equilibrium for all tested games, however with the disadvantage of being much slower compared then the regular Q-learning models.

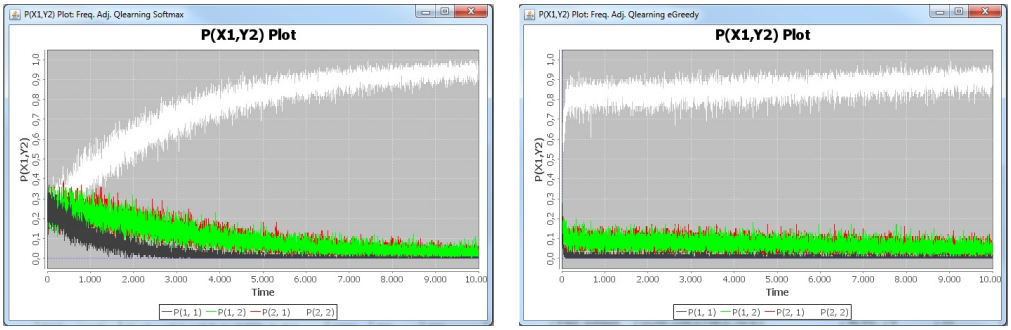


Figure 4.36: The frequencies to reach a Nash outcome. FAQs (l.) and FAQϵ (r.) on the 1B prisoner's dilemma game with $\alpha = \beta = 0.01$. Remaining parameters: $Q_0 = 0, \gamma = 0.9, \tau = 0.2, \epsilon = 0.2, \lambda = 0.00005$. Averaged over 100 runs of 10000 time periods.

Additionally, using even lower α and β values even augments this convergence period. As empirically observed, with values of 0.001 as used by Kaisers and Tuyls (2010) and Bloembergen et al. (2010) the convergence is prolonged to periods over 500000 time periods. As shown in Figure 4.36 though, higher values like 0.01 can often also lead to guaranteed convergence. However, as higher values give rise to behavior similar as basic Q-learning, how high these values can be depends on the particular game. While values of 0.05 allow for convergence in the 1B prisoner's dilemma, these values do not show complete convergence in the 2B chicken game (Figure 4.37). For the selection of games tested in these experiments, α and β values of 0.01 have shown to suffice to guarantee convergence to the Nash equilibrium given that the Q_0 values are initiated as zero.

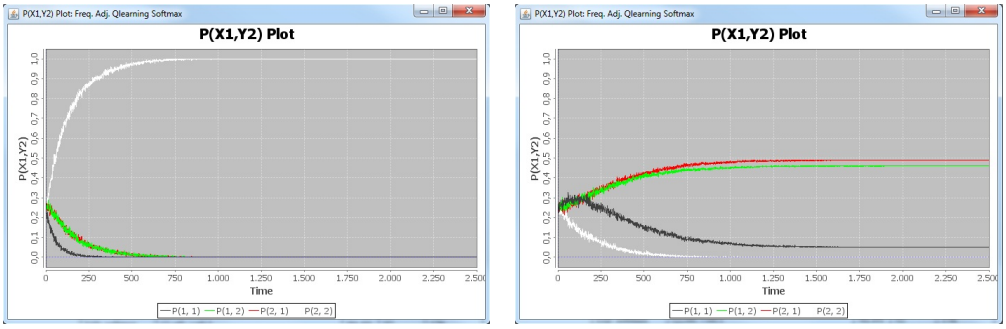


Figure 4.37: The frequencies to reach a Nash outcome. FAQs on the 1B prisoner's dilemma game (1.) and the 2B chicken game (r.) with $\alpha = \beta = 0.05$. Remaining parameters: $Q_0 = 0, \gamma = 0.9, \tau = 0.2$. Averaged over 1000 runs of 2500 time periods.

Sensitivity to high Q_0 values

Similar to basic Q-learning, when higher Q_0 values are used, the convergence behavior changes. For FAQs however these changes are highly limited. As depicted in Figure 4.38 for example showing an experiment of FAQs on the 1B prisoner's dilemma, the game suffering from this effect the most, overestimations of the Q value can prohibit the model from fully converging to the Nash equilibrium. Instead, because of the high Q values and the relatively small difference, a constant exploration factor is in place causing each player, from time to time, the choose the other actions as well.

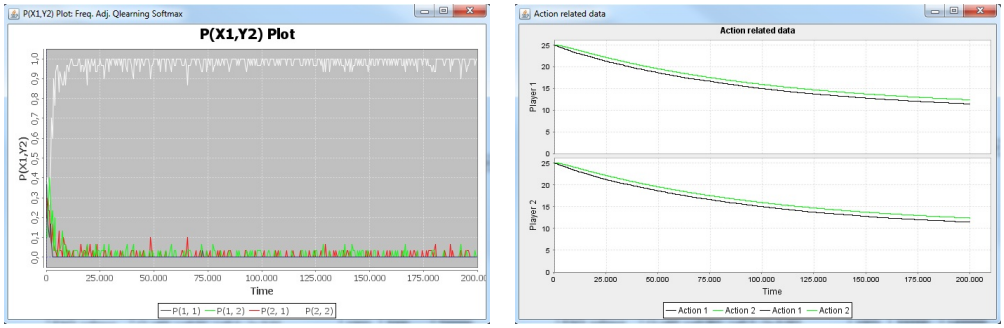


Figure 4.38: The frequencies to reach a Nash outcome (averaged over 30 runs of 200000 time periods) along with the typical evolution of the Q values of 1 run. FAQs on the 1B prisoner's dilemma game. Remaining parameters: $Q_0 = 25, \gamma = 0.9, \alpha = \beta = 0.01, \tau = 0.2$.

Considering FAQϵ however, matters are different. For this model, an overestimation of the Q values by the Q_0 value can have serious consequences for the convergence behavior. In fact, when experimentation is set to diminish using a positive λ factor, as soon as the experimentation stops, when the ϵ value has been lowered to zero by the λ factor, convergence drops, in this case to around 70%. With a λ factor of zero, this can be prevented by keeping a continuous exploration, convergence is then kept in the neighborhood of the Nash equilibrium as shown in Figure 4.39 on the left. For both these settings, the effect is quite similar to the one of FAQs as full convergence is prevented by a continuous exploration. Each agent thus selects his part of the Nash outcome most of the time while choosing other actions at other moments. However for FAQϵ, especially with positive λ factor, the convergence to the Nash outcome can be found to be much lower than with FAQs and highly depending on that λ factor. For both FAQs and FAQϵ, this behavior is found to be constant, even after the normalization process of the Q values to a more realistic level.

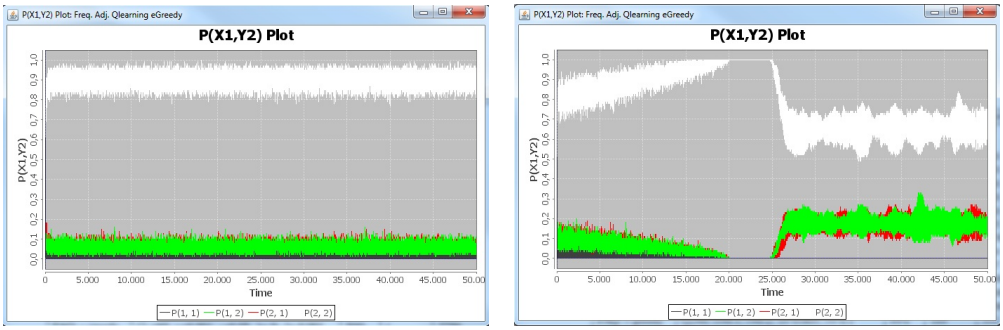


Figure 4.39: The frequencies to reach a Nash outcome (averaged over 50 runs of 50000 time periods). FAQϵ on the 1B prisoner's dilemma game with continuous experimentation $\epsilon = 0.1$ and $\lambda = 0$ (l.) or diminishing experimentation $\epsilon = 0.2$ and $\lambda = 0.00005$ (r.). Remaining parameters: $Q_0 = 25, \gamma = 0.9, \alpha = \beta = 0.01, \epsilon = 0.2$.

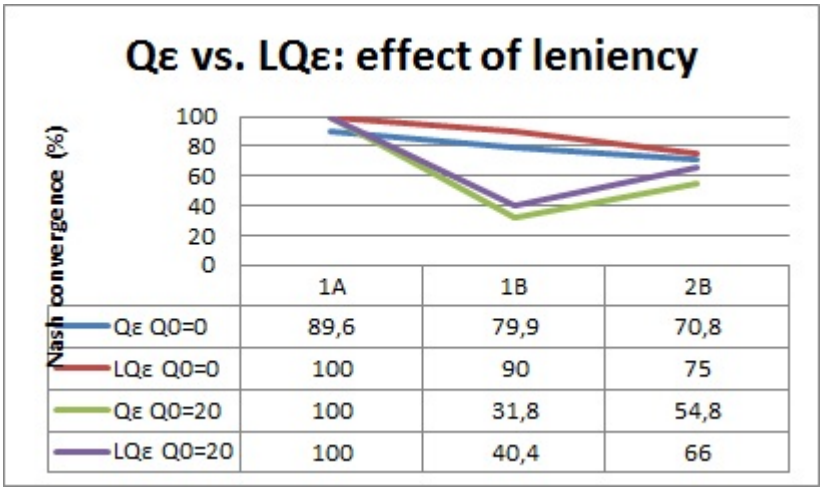
4.3.3 Leniency

The addition of leniency to both basic Q-learning and FAQ results in learning models like LQ and LFAQ. These models are identical to the aforementioned ones except for a short initial 'lenient' stage in which each action is chosen κ times before the Q values are updated for the first time. Because this stage is short, both LQ and LFAQ are computationally almost as efficient as their counterparts without leniency.

While incorporating leniency in these models mainly seems attractive for games of the subclass B like the prisoner's dilemma, experiments concerning these models actually show slightly different results. Overall, the addition of the initial lenient phase either amplifies the already existing convergence behavior towards the Nash equilibrium or maintains the already existing full convergence (for the 3B spoiled child game towards the Nash dominating outcome). This discrepancy can be attributed to the inherent increased exploration occurring in the initial lenient phase.

Consequences of this lenient phase can largely be divided in two types of situations. At first there are several situations in which the addition of leniency indeed improves the convergence behavior. This is for example the case for the 1A, 1B and 2B games, respectively deadlock, prisoner's dilemma and chicken game. As observable in Figure 4.40, the lenient models, independent of their action selection scheme, achieve a slightly higher convergence result. Although these kinds

of small differences might even be attributed to small random deviations, remarkable is that in case of the 1B game, convergence is not steered towards the Nash dominating outcome of cooperation as was expected. Improving cooperation in situations as the prisoner's dilemma was in fact one of the main goals of incorporating leniency in learning models.



	1A	1B	2B
Qε Q0=0	89,6	79,9	70,8
LQε Q0=0	100	90	75
Qε Q0=20	100	31,8	54,8
LQε Q0=20	100	40,4	66

Figure 4.40: Averaged over 1000 runs of each 1000 time steps, Qs vs. LQϵ with different Q_0 values. Plotted is the frequency that the Nash outcome is selected for both class 1A, 1B and 2B games. $\kappa = 5$.

In other situations however, improving convergence is not possible as a 100% convergence rate was already achieved using the models without leniency. Especially with FAQ which, when the parameters are set correctly, always reaches a full convergence. In these situations embedding leniency is often not worth it. In fact, in these situations where the rate of convergence itself cannot be changed, embedding leniency can even have some disadvantages. Especially observable with regular Q-learning, both Qs and Qϵ, the addition of the lenient phase can actually introduce an initial phase in which undesired behavior can be observed, before again yielding the same Nash convergence as without leniency. For Qs, this behavior is visible in the form of a random initial stage (Figure 4.41), for Qϵ, it can be observed that for some games, the initial lenient phase allows for the assigning of overoptimistic Q values to some actions. This is especially visible in the 3B spoiled child game (Figure 4.42), but to a lesser extent also visible in other games. This behavior can however not generally be observed with the FAQ models which show an almost identical consistent behavior with or without leniency, consistent with the results by Bloembergen et al. (2010).

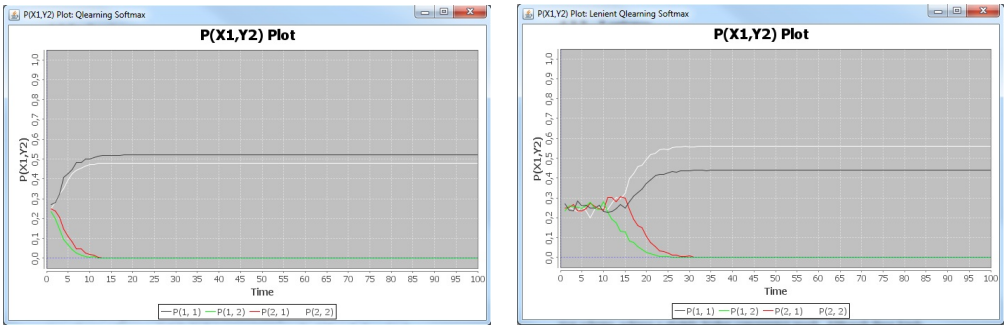


Figure 4.41: The frequencies to reach a Nash outcome (averaged over 1000 runs of 100 time periods). Qs (l.) and LQs with $\kappa = 5$ (r.) on the 2A matching pennies game. Remaining parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \tau = 0.2$.

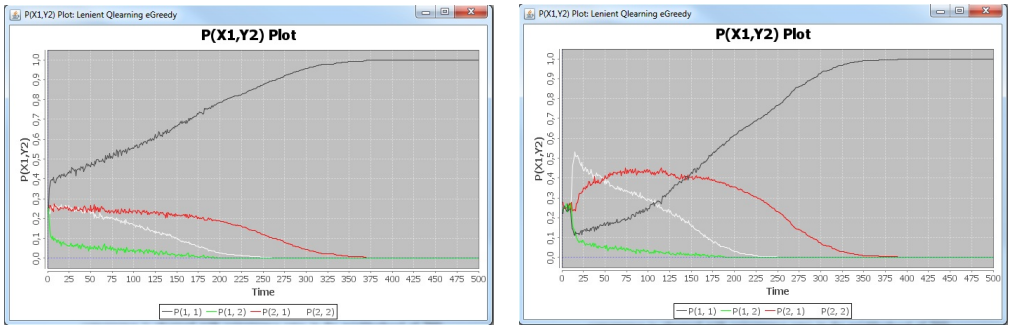


Figure 4.42: The frequencies to reach a Nash outcome (averaged over 1000 runs of 500 time periods). Qϵ (l.) and LQϵ with $\kappa = 5$ (r.) on the 3B spoiled child game. Remaining parameters: $Q_0 = 0, \alpha = 0.1, \gamma = 0.9, \epsilon = 0.2, \lambda = 0.005$.

4.3.4 Overview

When focusing on the basic Q-learning models, no consistent convergence behavior can be found. While for the subclass 1A and 2A games, convergence is in the neighborhood of 100% Nash convergence, for the 3A games, only the empirical distributions are found to converge to the NE while the final result actually constitutes a correlated equilibrium. For the subclass B games, besides 3B, matters are even more complicated. Using low initial Q values, a preference for Nash convergence is observed with convergence rates in the neighborhood of 70%. However using high initial Q values, this preference is turned towards the Nash dominating outcome. Depending on the game and the precise parameter settings, a 100% convergence to these Nash dominating outcomes can even be

found. Finally for the 3B game, convergence is always found to be towards the Nash dominating outcome.

For these models, the parameter settings are thus often crucial to determine the convergence behavior. Especially the initial Q value Q_0 has been shown to be play an important role. However determining the optimal Q_0 value is often hard as information about the game is required. Additionally, if these values are chosen too high, the initial stage of the convergence is generally much slower. In some cases even, for the 3B spoiled child for example, these values can even lead to an almost random behavior in the first few hundred time periods before finally converging to the Nash outcome. Therefore often a more neutral Q_0 value of zero can be considered (although for games allowing negative payoffs this might lead to similar effects as high Q_0 values in our experiments).

This inconsistent convergence behavior is solved using the frequency adjusted Q-learning models. Given that the parameters α and β are chosen sufficiently low, a full convergence to the Nash outcome, even for the rather unstable 3B game, can always be observed. Only for the FAQϵ model, attention should be paid to the value of the λ parameter. To avoid convergence drops with high Q_0 values and to be able to guarantee a full convergence for the 3B game, exploration for this model should be maintained. The downside for this consistent behavior is however the length of the convergence. While the basic Q-learning models achieved a stable level after at most a few hundred time periods, the FAQ models take several thousands and sometimes even a few hundred thousand time periods.

While leniency can be added to both the basic Q-learning models and the FAQ models, it has only shown to improve results in the first case. The improvements are however quite limited and driven by the increased exploration in the 'lenient' initial phase. Remarkable in this is the fact that the models with leniency showed a higher convergence to the Nash outcome in the 1B prisoner's dilemma. This while the addition of leniency was expected to aid convergence to the more rewarding Nash dominating outcome. However, in some games where basic Q-learning already achieved full convergence, the addition of leniency can also have some disadvantages. When applied to FAQ however, no significant differences can be observed with regard to the versions without leniency. Again a consistent 100% convergence to the NE can here be observed. A schematic overview of the different learning models can be found in Table 4.3 and 4.4 showing the convergence rates and the average time periods required to reach convergence.

Game classification						
	1		2		3	
	A	B	A	B	A	B
Qs	97.8% Nash	80.2% Nash	100% Nash	67% Nash	100% CE/Nash	100% non-Nash
Qϵ	89.6% Nash	79% Nash	100% Nash	70.7% Nash	100% CE/Nash	100% non-Nash
FAQs	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash
FAQϵ	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash	If $\lambda > 0$: 100% non-Nash If $\lambda = 0$: 100% Nash
LQs	99.8% Nash	95.1% Nash	100% Nash	75.7% Nash	100% CE/Nash	100% non-Nash
LQϵ	100% Nash	88.8% Nash	100% Nash	75% Nash	100% CE/Nash	100% non-Nash
LFAQs	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash
LFAQϵ	100% Nash	100% Nash	100% Nash	100% Nash	100% Nash	If $\lambda > 0$: 100% non-Nash If $\lambda = 0$: 100% Nash

Table 4.3: Overview of the convergence characteristics of the different Q-learning models using the standard earlier presented parameter settings and $Q_0 = 0$. For convergence characteristics with higher Q_0 values, see earlier.

Game classification						
	1		2		3	
	A	B	A	B	A	B
Qs	109 (9)	99	21 (7)	24	/	565 (316)
Qϵ	200	200	200 (109)	197	/	472 (264)
FAQs	14482 (5113)	70150 (23141)	22848 (13433)	55776 (42988)	/	/
FAQϵ	19966 (18311)	19966 (18311)	19954 (18232)	19967 (16390)	/	If $\lambda > 0$: 76501 (150000+)
LQs	33 (18)	165 (35)	43 (21)	47	/	619 (349)
LQϵ	200 (108)	192	200 (107)	199	/	501 (284)
LFAQs	12820 (57496)	72498 (30039)	25486 (12205)	76678 (46973)	/	/
LFAQϵ	19977 (18974)	19977 (18974)	19952 (18233)	19967 (16445)	/	If $\lambda > 0$: 81001 (150000+)

Table 4.4: Overview of the average time periods required to reach a stable state of convergence for the different Q-learning models using the standard earlier presented parameter settings and $Q_0 = 0$. Between parenthesis, the time required to reach a 90% convergence if available. For convergence characteristics with higher Q_0 values, see earlier. When times are not listed, the reached outcome was not a stable outcome but rather a mixed equilibrium for which the time could not be accurately measured.

4.4 Regret minimization

This section elaborates on the results of the models based on the notion of minimizing regret. The analysis is divided in two parts based on the difference between unconditional and conditional regret.

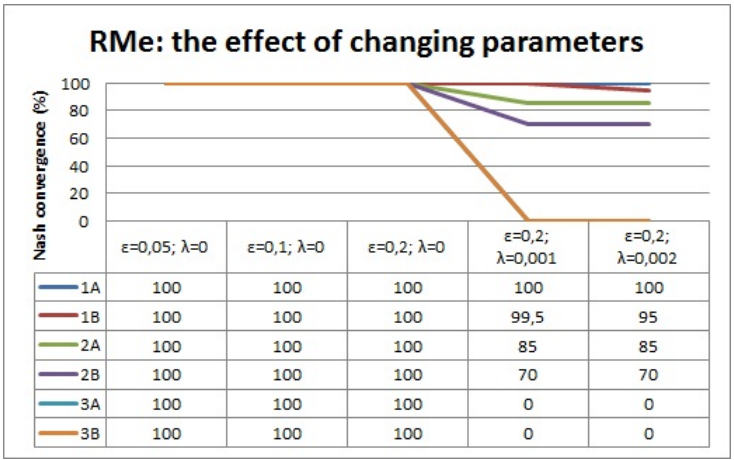
4.4.1 Unconditional regret matching

As for the models on unconditional regret, two models have been introduced earlier. The first one, RM, introduces the basic notion of regret along with the

traditional method of regret calculation. The second one, RMϵ, uses a regret estimation technique in order to arrive at a regret minimization model without information requirements on the opponents and thus equally powerful compared to the earlier CPM and Q-learning models.

Parameter settings

Although RM has no parameters that have to be initialized, this is not the case for RMϵ which uses a parameter ϵ defining how much random exploration is to be maintained. Additionally, a λ value determines the decay of the random exploration where a zero λ value causes a constant amount of exploration. As shown in Figure 4.43, these cases with continuous explorations yield consistent Nash convergence for games of all classes. Although the reported results in case of a zero λ do not take the random exploration into account (a result of 100% with $\epsilon = 0.1$ can thus be interpreted as 90% playing the best action), they still in general outperform the experiments with a relatively fast diminishing exploration, this in contrast with the earlier Qϵ models where these diminishing exploration values mostly increased performance.



	ε=0,05; λ=0	ε=0,1; λ=0	ε=0,2; λ=0	ε=0,2; λ=0,001	ε=0,2; λ=0,002
1A	100	100	100	100	100
1B	100	100	100	99,5	95
2A	100	100	100	85	85
2B	100	100	100	70	70
3A	100	100	100	0	0
3B	100	100	100	0	0

Figure 4.43: The rate of convergence towards Nash outcomes of RMϵ applied to games of the three main classes using a changing ϵ and λ value thereby affecting the exploration rate. For the 100% results with $\lambda = 0$, exploration is not taken into account.

Especially for the class 3 games, this diminishing experimentation can lead to degenerate results in which regret values end up incorrectly causing one action to

be favored above the other contrasting with the NE. Figure 4.44 gives an example of such a run in which, caused by a positive λ value, the regret values are, especially for player 2, fixed incorrectly leading to non-Nash action distributions.

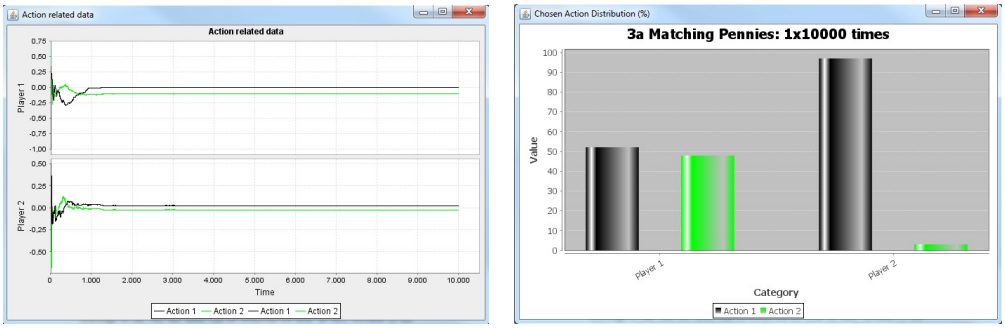


Figure 4.44: The evolution of regret values on a run of RMϵ on the 3A matching pennies game with fast diminishing experimentation ($\lambda = 0.002$, l.) and the empirical action distribution of that run. Parameters: $\epsilon = 0.2$.

The main reason for this phenomenon can be found in the specific method of diminishing experimentation. While positive λ values ultimately stop experimentation, the calculation of the estimated regret values is not stopped. Because of this along with the unstable nature of the class 3 games, these regret values ultimately take on incorrect values. In fact, it has been observed that, at the moment when experimentation stops, the player's probability distributions still represent Nash equilibrium characteristics. Because of this, fixing the probabilities at the same time when experimentation stops could ultimately also lead to NE results, given that the λ values are low enough.

For class 1 and 2 games, as already proven by Foster and Vohra (1998), small λ values can be found leading to full Nash convergence. However these values are game dependent. While the 1A deadlock game suffices to have a 0.002 value, the 1B prisoner's dilemma game requires a value in the neighborhood of 0.0001. Bigger values lead to final results in which a small portion of exploration is still maintained and a 100% convergence is never reached. Because of the game dependency of the ideal λ value, the particular convergence for class 3 games and the fact that for class 1 and 2 games the convergence itself is not influenced by changing λ values, all following experiments are conducted with a ϵ value of 0.2 and a λ of 0.

Calculated regret vs. estimated regret

While RMϵ with his regret estimation technique has been shown to consistently converge to the NE, the same can be observed for RM. In fact, because of the ability to calculate regret by using additional information, RM outperforms RMϵ both in speed (Figure 4.45) with full convergence always reached before 50 time periods and final result as RMϵ maintains a constant experimentation factor.

Figure 4.45: The frequencies to reach a certain outcome (NE=white) of RM (l.) and RMϵ with $\epsilon = 0.2$ and $\lambda = 0$ (r.) on the 1B prisoner's dilemma game. Averaged over 1000 independent runs of 200 time periods.

With respect to the class 3 games, it has been observed that the empirical action distributions, similar to the previous Q-learning and several CPM models again converge to the NE. Additionally, for the 3A Shapley's game, the final result in which the six rewarding outcomes are visited with high probability again constitutes a CE. For RM, the zero rewarding outcomes are, except for the initial stage, even never visited (Figure 4.46) while RMϵ maintains a small probability for those outcomes. However, for both RM and RMϵ, this NE convergence behavior is also observed for the 3B spoiled child game (Figure 4.47), for which the continuous adaption of the regret values causes a continuous action changing behavior where each player repeatedly plays his best reply action using his regret values.

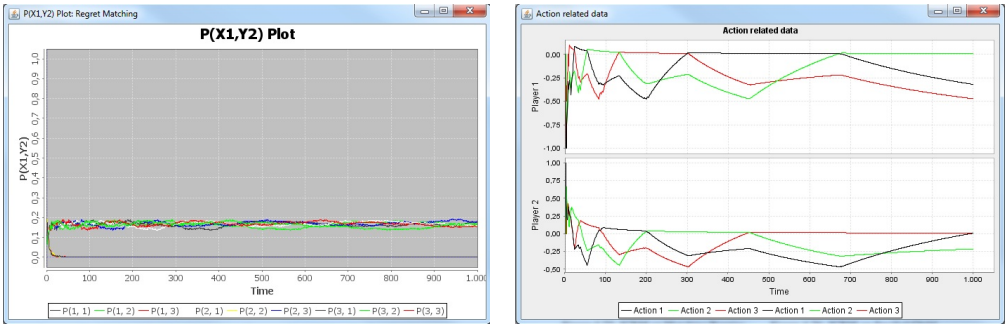


Figure 4.46: The frequencies to reach a certain outcome of RM on the 3A Shapley's game (l.) with the evolution of the regret values of a typical run (r.). Frequencies averaged over 1000 independent runs of 1000 time periods.

Figure 4.47: The frequencies to reach a certain outcome of RM (l.) and RMϵ with $\epsilon = 0.2$ and $\lambda = 0$ (r.) on the 3B spoiled child game. Averaged over 1000 independent runs of 1000 time periods.

However, while the empirical action distributions of both RM and RMϵ over time converge to the NE, this cannot always be observed in short runs. While for runs on the 3A matching pennies game, even short runs of 200 periods show action distributions towards the NE for both models, this is not the case for the 3A Shapley's game with his 3 actions. For this game, action distributions in favor of one action can be observed and this even for runs of 50000 time periods (Figure 4.48). For the 3B spoiled child on the other hand, while RM in general after only 200 time periods results in action distributions similar to the NE, for RMϵ, time periods around 2000 are recommended for the same result.

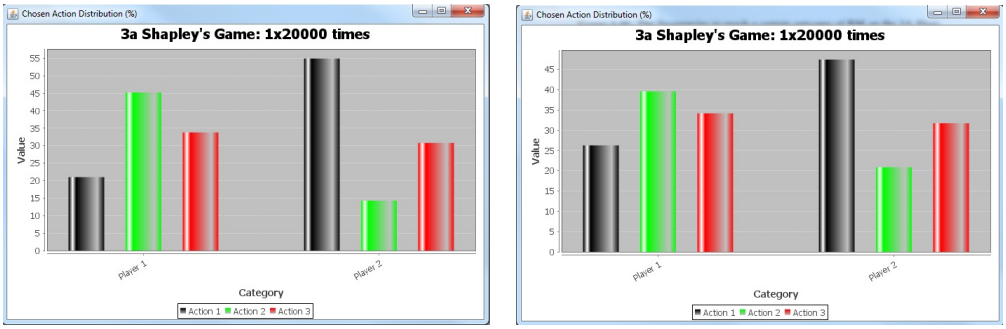


Figure 4.48: The empirical action distributions of a single run of RM (l.) and RMϵ with $\epsilon = 0.2$ and $\lambda = 0$ (r.) on the 3A Shapley's game.

Both RM and RMϵ thus consistently converge towards the NE and this for all tested games in all classes although for the 3A Shapley's game only in the long run. Using its regret calculation technique, RM shows to have a more reliable regret value with faster convergence and the ability to reach a full NE convergence where RMϵ is forced to maintain a continuous experimentation. However with positive λ values, as mentioned earlier, given that the λ values are low enough to allow sufficient experimentation, full Nash convergence can also be reached by RMϵ. This also for class 3 games, as long as the probabilities are fixed as soon as experimentation stops, instead of continuing adapting these.

4.4.2 Conditional regret matching

This subsection introduces the results of the two models covering conditional regret matching, ICRM and HMCRM. As no conditional regret estimation techniques have been found, both models compute their regret using information on the opponent's actions and the payoff matrix. These additional requirements form a disadvantage in practical usage. Whether it improves performance compared to models like RMϵ lacking these requirements is assessed in this subsection.

Parameter settings

Similar to both RM and RMϵ, both conditional regret models are able to achieve a full 100% Nash convergence and this for all reasonable parameter values. Because of this, the value of the parameters is chosen based on the time required to reach this result. As shown in Figure 4.49, larger values for both ICRM (the λ parameter) and HMCRM (the ϵ parameter) reduce the time periods needed for a full convergence. By using values of 0.9 for both, both models are able to both

comply with the Nash characteristics and be able to achieve this in a short time period. The results of the class 3 games are not listed in these figures as for those games, mixed equilibria are reached of which the convergence time cannot precisely be assessed. However, also for these games, the values of the parameter do not show significantly different results. Both ICRM and HMCRM prove to be rather insensitive to adapting their parameter settings.

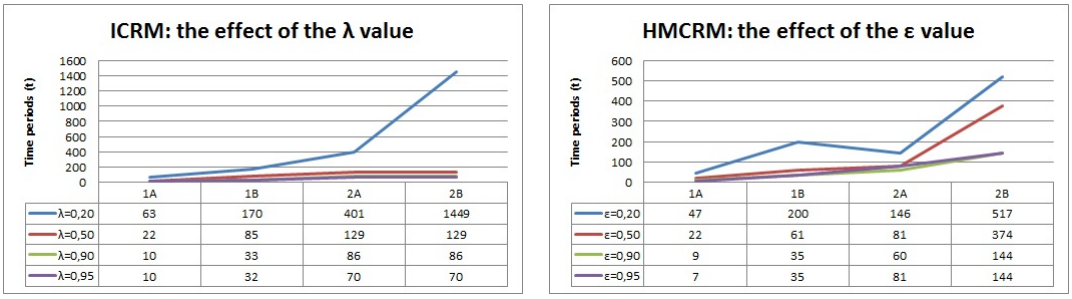


Figure 4.49: The number of time periods required to reach a full Nash convergence of ICRM (l.) and HMCRM (r.) applied to games of the three main classes using a changing λ and ϵ value. Data measured over 1000 independent runs

General convergence: class 1 and 2 games

As can be observed in Figure 4.49, for all class 1 and 2 games, both ICRM and HMCRM are able to reach a full Nash convergence and this quite fast. In fact, For several games, a full convergence is reached before 35 time periods, while, seen over 1000 runs, a 90% convergence never takes more than 50 time periods.

General convergence: class 3 games

For the class 3 games however, because of the absence of pure NE, convergence is forced towards the mixed NE. This process is however not as fast as with the other classes of games. Considering the 3A game class, although both ICRM and HMCRM have an identical convergence behavior with action distributions converging to the NE while the three zero rewarding outcomes are never visited constituting a CE (Figure 4.50 left), in short runs both models show a slightly degenerate result in which some actions are played considerable more than others. Similarly to the previous regret minimizing models, this effect is most present for the 3A Shapley's game (Figure 4.50 right) for which runs of even 100000 time periods yield non-Nash distributions. Although the effect is only to a lesser

extent also present for the 3A matching pennies game and the 3B spoiled child (for runs below respectively 1000 and 5000 time periods), these conditional regret minimization models are ultimately outperformed by both RM and RMϵ when applied to class 3 games on short runs.

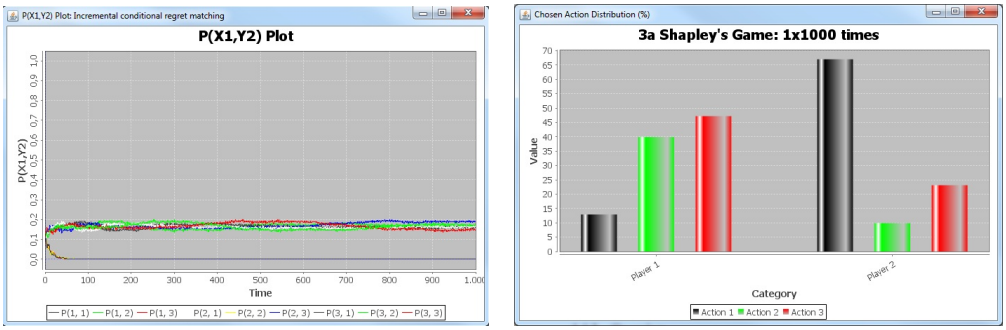


Figure 4.50: The frequencies to reach a certain outcome of ICRM on the 3A Shapley's game (l.) with the empirical action distributions of one single typical run (r.). Frequencies averaged over 1000 independent runs of 1000 time periods.

4.4.3 Overview

Four different models covering both unconditional and conditional regret minimization have been covered in this analysis. While for both class 1 and 2 games, convergence has been found to be consistent towards the NE, some differences concerning the class 3 games have also been noticed.

As for the unconditional regret minimization models, a basic regret minimization model, RM has been compared against RMϵ, a model using a regret estimation technique thereby eliminating the requirements for additional information. Because of this technique, RMϵ, is in practice as applicable as the earlier introduced CPM and Q-learning models. Despite of the information requirement differences, no significant differences in Nash convergence have in fact been observed. Although RMϵ was in these experiments forced to maintain a continuous experimentation, a clear preference for the Nash convergence can also be observed and this for all runs in all tested games, identical to RM. Only when applied to the 3A Shapley's game, both models do not exhibit Nash convergence for short runs. Focusing on the time differences, by using a regret calculation technique, RM has shown to be faster than RMϵ, although for some games this difference is minimal.

Concerning the conditional regret minimization models however, both using a regret calculation method, almost identical results as the unconditional regret minimization models have been observed: a clear convergence to the NE for both class 1 and 2 games while struggling with the 3A Shapley's game. Additionally, also for the other class 3 games, convergence of the empirical action distributions to the NE is found to be slower than the previous regret minimization models. On the other hand, for the class 1 and 2 games, convergence by the conditional regret minimization models is found to be slightly faster.

Similar to some CPM and Q-learning models, applications to the 3A Shapley's game of all regret based models show final results constituting a correlated equilibrium as the zero rewarding outcomes are hardly visited in favor of the rewarding outcomes. This result, and by extension the consistent convergence to the NE, confirms the theoretically expected convergence behavior. A schematical overview of the convergence characteristics and the times required to reach that convergence can be found at Table 4.5 and 4.6.

Game classification						
	1		2		3	
	A	B	A	B	A	B
RM	Yes	Yes	Yes	Yes	Yes/No	Yes
RMϵ	Yes	Yes	Yes	Yes	Yes/No	Yes
ICRM	Yes	Yes	Yes	Yes	Yes/No	Yes
HMCRM	Yes	Yes	Yes	Yes	Yes/No	Yes

Table 4.5: Overview of the empirical convergence characteristics of the different regret minimizing models. For RMϵ, only experiments with continuous experimentation are considered ($\epsilon = 0.2$). For Nash convergence to be denoted by 'Yes', an upper limit of 50000 time steps is set.

Game classification				
	1		2	
	A	B	A	B
RM	20	17	190 (11)	661 (12)
RMϵ	45	800	150	700
ICRM	10	33	86 (13)	86 (48)
HMCRM	9	35	60 (7)	144 (9)

Table 4.6: Overview of the durations required to reach a full 100% Nash convergence. For time periods above 50, if measurable, also the 90% time boundary is included between parenthesis. Considering RMϵ, only experiments with continous experimentation are considered ($\epsilon = 0.2$). Remaining parameters: $\lambda = \epsilon = 0.9$ for ICRM and HMCRM.

Chapter 5

Conclusion and future work

In this chapter, the main results and key observations of this thesis are summarized. To end, an overview is provided of other opportunities with regard to future research.

5.1 Conclusion

In this master thesis, two main classes of adaptive heuristics, simplistic learning rules with limited information requirements, have been reviewed, being the models on the concept of reinforcement learning and those of regret minimization. In the first group, the models on the economic notion of cumulative payoff matching and those using the computer science based Q-learning methods can also be distinguished.

The first group of reviewed models covers those on cumulative payoff matching. Although convergence towards the NE has been observed for several classes of games, other classes of games have a more inconsistent convergence behavior. Particularly when applied to the subclass B games, convergence towards the Nash dominating outcome is rather common. Covering the different models, while CPM already showed promising results, the improvements to the basic models do show an increased performance. Even for CPM-A which exhibits a sever sensitivity to its initial parameter values, convergence in general is observed more consistently. Despite of these results however, for class 3 games Nash convergence is for all methods still difficult to achieve.

The Q-learning based models are the second group of adaptive heuristics that were analyzed. Covering both basic and more extended versions incorporating leniency and an adjusted update rule, in most games, Q-learning has been shown

to have an almost consistent convergence towards the NE. Although the basic versions are not always able to reach a full convergence, the FAQ models guarantee a Nash convergence. Even for the 3B subclass, FAQ is able to reach a consistent Nash convergence and this almost insensitive to the initial parameter values. Although incorporating leniency did slightly improve Nash convergence for the basic models, for the FAQ models the influence is limited. Nevertheless, because of the short duration of the lenient phase, the performance of the models is hardly affected. The implementation of leniency can thus be seen as a useful addition, especially for the basic versions. Additionally, while the FAQ models applied to the 3A Shapley's game converge to the NE, the basic Q-learning models in this situation actually converge to the CE with higher average realized payoffs as a consequence.

The final group of models are those on the notion of minimizing regret. Although most of these models require additional information requirements, the RMϵ model does not by using a regret estimation technique. Nevertheless, even though RMϵ only estimates regret, a similar result as with the other regret minimization models can be achieved: consistent Nash convergence. In fact, although the convergence of RMϵ is consistently slower for both class 1 and 2 games, when applied to class 3 games, RMϵ together with RM, actually outperforms the more advanced conditional regret models ICRM and HMCRM. RMϵ thus shows to perform consistently towards the Nash outcome although a complete 100% convergence is harder to achieve.

To end, it can thus be concluded that Nash convergence for these adaptive heuristics is hard to achieve consistently for all classes of games. While most completely uncoupled models fail when applied to the 3B subclass, the FAQ models along with RMϵ (for the tested 2x2 games) are the only ones to consistently accomplish Nash convergence, however with the drawback of a long convergence period or a small portion of experimentation. When a full convergence is however not critical, other learning models have also shown to be promising. Especially the basic Q-learning models, possibly extended with the possibility of leniency, show both fast and frequent Nash convergence while also the CPM models generally achieve high average realized payoffs although less often together with Nash convergence.

5.2 Future work

- Games of incomplete information: In addition to the games covered in this thesis, the convergence behavior of other types of games could also be re-

viewed. One of these possibilities is games of incomplete information as for example signaling games (Cho and Kreps, 1987) in which players are not equally informed.

- An endogenous aspiration level for CPM models: Although the CPM-BS models showed an almost full Nash convergence, its contradiction with the theoretical work of Karandikar et al. (1998) could provide an indication that the aspiration level was not adequately set and/or adapted throughout the process. Further research might include the use of other aspiration level methods and an assessment whether convergence can in fact be turned towards the Nash dominating outcome in case of subclass B games.

- Leniency: While the addition of leniency to Q-learning did not provide the expected results, it can be suspected that this result is caused by the specific applied method. Future work could therefore include the combination of other implementation methods of lenient behavior aiding in convergence towards 'cooperation' outcomes in games like the prisoner's dilemma.

- Further use of the test framework: One of the contributions of this master thesis is the development of a test framework driven by the motivation that no other application previously allowed us to conduct experiments on normal form games while having the ability of visualizing the results and gathering important test data. Further use of this framework could include the addition of other learning models, other games and visualization possibilities.

Glossary

CCE coarse correlated equilibrium. 1, 16, 21, 22

CE correlated equilibrium. 1, 16, 19–22, 83, 85, 97, 103, 105, 109, 112, 114, 117

CPM cumulative payoff matching. 39–44, 48, 64, 66, 68, 70–74, 76, 77, 79–85, 107, 109, 113, 114, 117, 118

CPM-A Arthur's CPM model. 42, 43, 68, 74, 75, 77–80, 83, 85, 117

CPM-BS Borgers and Sarin's CPM model. 45–47, 68, 81–85, 118

CPM-RE Roth and Erev's CPM model. 43–45, 68, 74, 76–81, 83, 85

EGT evolutionary game theory. 30, 37

FAQ frequency adjusted Q-learning. 50, 51, 53, 98, 101, 102, 104, 117

FAQϵ FAQ with ϵ-experimentation. 51, 52, 68, 98, 100, 101, 104–106

FAQs FAQ with softmax action selection. 51, 52, 68, 98–100, 105, 106

FP fictitious play. 2, 64

HMCRM HM conditional regret matching with inertia. 61–63, 68, 111, 112, 114, 115, 117

ICRM incremental conditional regret matching with weight λ. 59–62, 68, 111–115, 117

L(FA)Q lenient (frequency adjusted) Q-learning. 54

LFAQ lenient frequency adjusted Q-learning. 52, 53, 101

LFAQϵ LFAQ with ϵ-experimentation. 53, 105, 106

LFAQs LFAQ with softmax action selection. 53, 105, 106

LQ lenient Q-learning. 52, 53, 101

LQϵ lenient Q-learning with ϵ-experimentation. 53, 102, 103, 105, 106

LQs lenient Q-learning with softmax action selection. 53, 103, 105, 106

MDP Markov decision process. 3

NE Nash equilibrium. 1, 16, 18–21, 32, 53, 67, 70–72, 76–80, 83–85, 90–92, 98, 103, 104, 108–114, 117

Qϵ Q-learning with ϵ-experimentation. 49, 50, 68, 86–97, 102, 103, 105–107

Qs Q-learning with softmax action selection. 49–51, 53, 68, 86–88, 91–94, 97, 102, 103, 105, 106

RL reinforcement learning. 38, 39, 54, 56

RM regret matching. 56–58, 68, 106, 107, 109–111, 113–115, 117

RMϵ regret matching with ϵ-experimentation. 2, 56, 57, 59, 66, 68, 107–111, 113–115, 117, 118

Bibliography

Abernethy, J. and Mannor, S. (2011). Does an efficient calibrated forecasting strategy exist? *Journal of Machine Learning Research - Proceedings Track*, 19:809–812.

Arthur, W. B. (1990). A learning algorithm that mimics human learning. *Santa Fe Institute Working Paper*.

Arthur, W. B. (1991). Designing economic agents that act like human agents: A behavioral approach to bounded rationality. *American Economic Review*, 81(2):353–59.

Arthur, W. B. (1993). On designing economic agents that behave like human agents. *Journal of Evolutionary Economics*, 3(1):1–22.

Aumann, R. J. (1974). Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1):67–96.

Bachmann, P. (1894). *Die analytische Zahlentheorie*, volume 2. Teubner.

Barany, I., Lee, J., and Shubik, M. (1992). Classification of two-person ordinal bimatrix games. *International Journal of Game Theory*, 21(3):267–90.

Basu, K. (1994). The traveler's dilemma: Paradoxes of rationality in game theory. *The American Economic Review*, 84(2):391–395.

Bayes, M. and Price, M. (1763). An Essay towards Solving a Problem in the Doctrine of Chances. By the Late Rev. Mr. Bayes, communicated by Mr. Price, in a letter to John Canton, M. A. and F. R. S. *Philosophical Transactions (1683-1775)*.

Bellman, R. (1957). A markovian decision process. Technical report, DTIC Document.

Berger, U. (2007). Brown's original fictitious play. *Journal of Economic Theory*, 135(1):572–578.

Blackburn, J. (1936). *The Acquisition of Skill: An Analysis of Learning Curves*. Reports. H.M. Stationery Office.

Blackwell, D. (1956). An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1–8.

Blocki, J., Christin, N., Datta, A., and Sinha, A. (2011). Adaptive regret minimization in bounded-memory games. *Computing Research Repository (CoRR)*.

Bloembergen, D., Kaisers, M., and Tuyls, K. (2010). Lenient frequency adjusted q-learning. In *Proceedings of 22nd Benelux Conference on Artificial Intelligence (BNAIC 2010)*, pages 19–26. University of Luxembourg.

Borgers, T. and Sarin, R. (1997). Learning through reinforcement and replicator dynamics. *Journal of Economic Theory*, 77(1):1–14.

Borgers, T. and Sarin, R. (2000). Naive reinforcement learning with endogenous aspirations. *International Economic Review*, 41(4):921–50.

Boyan, J. and Littman, M. (1994). Packet routing in dynamically changing networks: A reinforcement learning approach. *Advances in neural information processing systems*, pages 671–671.

Brown, G. W. (1951). Iterative solution of games by fictitious play. *Activity analysis of production and allocation*, 13(1):374–376.

Bush, R. R. and Mosteller, F. (1951). A mathematical model for simple learning. *Psychological Review*, 58(5):313–323.

Bush, R. R. and Mosteller, F. (1955). *Stochastic models for learning*. Wiley, New York.

Cheng, S.-f., Reeves, D. M., Vorobeychik, Y., and Wellman, M. P. (2004). Notes on equilibria in symmetric games. In *In Proceedings of the 6th International Workshop On Game Theoretic And Decision Theoretic Agents (GTDT)*, pages 71–78.

Cho, I. and Kreps, D. (1987). Signaling games and stable equilibria. *The Quarterly Journal of Economics*, 102(2):179–221.

Claus, C. and Boutilier, C. (1998). The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the National Conference on Artificial Intelligence*, pages 746–752. JOHN WILEY & SONS LTD.

Cournot, A. A. (1838). *Recherches sur les principes mathématiques de la théorie des richesses*. L. Hachette.

Daskalakis, C., Goldberg, P. W., and Papadimitriou, C. H. (2006). The complexity of computing a nash equilibrium. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, STOC 2006, pages 71–78, New York, NY, USA. ACM.

Debbah, M., Lasaulce, S., and Tembine, H. (2011). *Game Theory and Learning for Wireless Networks: Fundamentals and Applications*. Academic Press. Academic Pr.

Deutsch, K. (1968). *The analysis of international relations*, volume 12. Prentice-Hall Englewood Cliffs, New Jersey.

Dobson, J. (1999). *The art of management and the aesthetic manager: The coming way of business*. Praeger Pub Text.

Ebbinghaus, H. (1913). *Memory: A Contribution to Experimental Psychology*. Columbia University. Teachers College. Educational reprints. no. 3. Teachers College, Columbia University.

Foster, D. and Vohra, R. (1999). Regret in the on-line decision problem. *Games and Economic Behavior*, 29(1-2):7–35.

Foster, D. P. and Vohra, R. V. (1993). A randomization rule for selecting forecasts. *Operations Research*, 41:704–709.

Foster, D. P. and Vohra, R. V. (1997). Calibrated learning and correlated equilibrium. *Games and Economic Behavior*, 21(1-2):40–55.

Foster, D. P. and Vohra, R. V. (1998). Asymptotic calibration. *Biometrika*, 85(2):379–390.

Foster, D. P. and Young, H. P. (2003). Regret testing: a simple pay-off based procedure for learning nash equilibrium. *Theoretical Economics*, 1(3):341–367.

Foster, D. P. and Young, H. P. (2006). Regret testing: learning to play nash equilibrium without knowing you have an opponent. *Theoretical Economics*.

Fudenberg, D. and Levine, D. (1993). The theory of learning in games. *Econometrica*, 61(5):1019–1045.

Fudenberg, D. and Tirole, J. (1991). *Game Theory*. Mit Press.

Gibbons, R. (1992). *A primer in game theory*. FT Prentice Hall.

Gintis, H. (2000). *Game Theory Evolving: A Problem-Centered Introduction to Modeling Strategic Behavior*. Economics / Princeton University Press. Princeton University Press.

Gintis, H. (2009). *Game Theory Evolving: A Problem-Centered Introduction to Modeling Strategic Interaction*. Princeton University Press.

Green, J. (1859). *Gambling in its Infancy and Progress, Dissuasive to the Young against Games of Chance*. Sheldon.

Greenwald, A. and Jafari, A. (2003). A general class of no-regret learning algorithms and game-theoretic equilibria. *Learning Theory and Kernel Machines*, pages 2–12.

Hannan, J. (1957). Approximation to bayes risk in repeated plays. *Contributions to the Theory of Games*, 3:97–139.

Hardin, R. (1787). Blackmailing for mutual good. *University of Pennsylvania Law Review141*, 1815.

Hart, S. (2005). Adaptive heuristics. *Econometrica*, 73(5):1401–1430.

Hart, S. and Mas-Colell, A. (2000). A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68:1127–1150.

Heidemann, J., Ye, W., Wills, J., Syed, A., and Li, Y. (2006). Research challenges and applications for underwater sensor networking. In *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, volume 1, pages 228–235. IEEE.

Hillas, J., Kohlberg, E., Pratt, J., and of Research, H. B. S. D. (2007). *Correlated Equilibrium and Nash Equilibrium as an Observer's Assessment of the Game*. Division of Research, Harvard Business School.

Jordan, J. (1991). Bayesian learning in normal form games. *Games and Economic Behavior*, 3(1):60–81.

Jordan, J. (1993). Three problems in learning mixed-strategy nash equilibria. *Games and Economic Behavior*, 5(3):368–386.

Jordan, J. (1995). Bayesian learning in repeated games. *Games and Economic Behavior*, 9(1):8–20.

Kaisers, M. and Tuyls, K. (2010). *Frequency adjusted multi-agent Q-learning*, pages 309–316. International Foundation for Autonomous Agents and Multiagent Systems.

Karandikar, R., Mookherjee, D., Ray, D., and Vega-Redondo, F. (1998). Evolving aspirations and cooperation. *Journal of Economic Theory*, 80(2):292–331.

Kopalle, P. and Shumsky, R. (2010). Game theory models of pricing. In *The Oxford Handbook of Pricing Management*.

Koulovatianos, C. and Wieland, V. (2011). Asset pricing under rational learning about rare disasters. *CEPR Discussion Papers*.

Kreps, D. M. and Wilson, R. (1982). Sequential equilibria. *Econometrica*, 50(4):863–94.

Lipnowski, I. and Maital, S. (1983). Voluntary provision of a pure public good as the game of chicken. *Journal of Public Economics*, 20(3):381–386.

Luce, R. and Raïffa, H. (1957). *Games and decisions: introduction and critical survey*. Wiley.

Matsumoto, M. and Nishimura, T. (1998). Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30.

Miyazawa, K. (1961). *On the Convergence of the Learning Process in a 2 X 2 Non-Zero-sum Two-person Game*. Princeton University. Econometric Research Program. Research memorandum. Princeton University.

Moulin, H. and Vial, J. (1978). Strategically zero-sum games: The class of games whose completely mixed equilibria cannot be improved upon. *International Journal of Game Theory*, 7:201–221.

Myerson, R. (1999). Nash equilibrium and the history of economic theory. *Journal of Economic Literature*, 37(3):1067–1082.

Myerson, R. B. (1978). Refinements of the nash equilibrium concept. *International Journal of Game Theory*, 7:73–80.

Nachbar, J. (2005). Beliefs in repeated games. *Econometrica*, 73(2):459–480.

Nash, J. (1951). Non-Cooperative Games. *The Annals of Mathematics*, 54(2):286–295.

Nash, J. (1953). Two-person cooperative games. *Econometrica: Journal of the Econometric Society*, pages 128–140.

Newell, A., Rosenbloom, P. S., and Anderson, J. R. (1981). *Mechanisms of skill acquisition and the law of practice*, pages 1–55. Erlsbaum, Hillsdale, NJ.

Newton, S. and Chittenden, N. (1848). *Newton's Principia: The mathematical principles of natural philosophy*. D. Adee.

Nisan, N. (2007). *Algorithmic game theory*. Cambridge Univ Pr.

Nowak, M. A. (2006). *Evolutionary Dynamics: Exploring the Equations of Life*. Harvard University Press.

Osborne, M. and Rubinstein, A. (1994). *A course in game theory*. The MIT press.

Panait, L., Sullivan, K., and Luke, S. (2006). Lenience towards teammates helps in cooperative multiagent learning. In *Proceedings of Fifth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS 2006)*.

Papadimitriou, C. and Tsitsiklis, J. (1987). The complexity of markov decision processes. *Mathematics of operations research*, 12(3).

Papadimitriou, C. H. and Roughgarden, T. (2008). Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)*, 55(3):14:1–14:29.

Rasmusen, E. (2007). *Games and Information: An Introduction to Game Theory*. Blackwell Pub.

Robinson, D. and Goforth, D. (2005). *The Topology of the 2x2 Games: A New Periodic Table*. Routledge Advances in Game Theory. Routledge.

Robinson, J. (1951). An iterative method of solving a game. *Annals of Mathematics*, 54(2):296–301.

Roth, A. E. and Erev, I. (1995). Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. *Games and Economic Behavior*, 8(1):164–212.

Russell, B. (1959). *Common sense and nuclear warfare*. Simon and Schuster.

Scodel, A., Minas, J., Ratoosh, P., and Lipetz, M. (1959). Some descriptive aspects of two-person non-zero-sum games. *The Journal of Conflict Resolution*, 3(2):114–119.

Selten, R. (1975). Reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory*, 4:25–55.

Sermat, V. and Gregovich, R. (1966). The effect of experimental manipulation on cooperative behavior in a chicken game. *Psychonomic Science*.

Shapley, L. S. (1964). Some topics in two-person games. In *Advances in Game Theory*. Princeton University Press.

Shoham, Y. and Leyton-Brown, K. (2009). *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press.

Sigmund, K. and Nowak, M. (1999). Evolutionary game theory. *Current Biology*, 9(14):503–505.

Singh, S., Jaakkola, T., and Jordan, M. (1994). Learning without state-estimation in partially observable markovian decision processes. In *Proceedings of the Eleventh International Conference on Machine Learning*, volume 31, page 37. New Jersey, USA.

Singla, N., Hall, M., Shands, B., and Chamberlain, R. D. (2008). Financial monte carlo simulation on architecturally diverse systems. In *High Performance Computational Finance, 2008. WHPCF 2008. Workshop on*, pages 1–7. IEEE.

Snoddy, G. (1926). *Learning and Stability: A Psychophysiological Analysis of a Case of Motor Learning with Clinical Applications*. Journal of Applied Psychology. Journal of Applied Psychology.

Sutton, R. (1988). Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement learning: An introduction*, volume 1. Cambridge Univ Press.

Takahashi, K., Kaizu, K., Hu, B., and Tomita, M. (2004). A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics*, 20(4):538–546.

Thorndike, E. L. (1898). Animal intelligence: An experimental study of the associative processes in animals. *Psychological Review: Monograph Supplements*.

Tsitsiklis, J. N. (1994). Asynchronous stochastic approximation and q-learning. *Machine Learning*, 16:185–202.

Tuyls, K., Verbeeck, K., and Lenaerts, T. (2003). A selection-mutation model for q-learning in multi-agent systems. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 693–700. ACM.

Vega-Redondo, F. (2003). *Economics And The Theory Of Games*. Cambridge University Press.

von Neumann, J. and Morgenstern, O. (1944). Theory of games and economic behavior. *Library*.

Wang, H. X. and Yang, B. Z. (2003). Classification of 2x2 games and strategic business behavior. *The American Economist*.

Watkins, C. and Dayan, P. (1992). Q-learning. *Machine learning*, 8(3):279–292.

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK.

Wolf, M., Van Doorn, G., Leimar, O., and Weissing, F. (2007). Life-history trade-offs favour the evolution of animal personalities. *Nature*, 447(7144):581–584.

Wunder, M., Littman, M., and Babes, M. (2010). Classes of multiagent q-learning dynamics with e-greedy exploration. In *Proceedings of the 27th International Conference on Machine Learning (ICML 2010)*, pages 1167–1174.

Yao, J., Chen, J., and Sun, Z. (2002). An application in robocup combining q-learning with adversarial planning. In *Intelligent Control and Automation, 2002. Proceedings of the 4th World Congress on*, volume 1, pages 496–500. IEEE.

Young, H. P. (2004). *Strategic learning and its limits*. Number 2002 in Arne Ryde memorial lectures. Oxford University Press, USA.

Young, H. P. (2009). Learning by trial and error. *Games and economic behavior*, 65(2):626–643.