Combining Teaching and Following in Repeated Games

Max Knobbout

Supervised by:
dr. Gerard A.W. Vreeswijk
dr. ir. Jan M. Broersen

January 4, 2011

CONTENTS

1. INTRODUCTION

1.1 Topic description and relevant context

In recent years, there has been a clear shift from single agent research to multiagent systems. Multiagent systems, in the broad sense, are systems that include multiple autonomous entities with either diverging information or diverging interests, or both. The capacity to learn is a key facet of intelligent behaviour, and it is no surprise that much attention has been devoted to the subject in the various disciplines that study intelligence and rationality. This is the area of multiagent learning, which is primarily composed of two major disciplines – artificial intelligence and game theory. Here game theory can be seen as a tool to model the interactions that can arise between different agents: a game describes a scenario in which agents can perform actions and achieve payoff described by a quantitative payoff function. In order to reach a solution of a game, the agents need to coordinate their respective actions. The problem of coordination between agents is very substantial and can be seen as one of the major topics within the subject of multiagent learning. In this thesis, we consider the setting in which agents will repeatedly play the stage game, or in other words a repeated game. Moreover, we consider that the agents are not pre-coordinated and have no explicit way of communication (except by implicit observations). From this perspective, the act of proposing (or forcing) an outcome to our adversaries makes sense, which we will informally describe as 'teaching' behaviour. On the other hand we have 'following' behaviour, which can be understood as the act of going along with such a proposal. Teaching behaviour does not only make sense in order to reach coordination, but often times adopting the role of a teacher allows us to 'steer' followers to outcomes that are more beneficial to us. However, it can lead to miscoordination if multiple agents try to teach different outcomes of the game. Without an external designation of these roles, it can be hard to decide whether to take on the role of a teacher or a follower. The topic of this thesis will be to combine these seemingly opposing

behaviours into a unified whole.

1.2 Motivation

As we already stated, the act of teaching and following intuitively makes perfect sense when trying to achieve coordination in a game. However, during our research we noticed that in many pieces of research the notion of teaching and following is never formally defined. The motivation for this thesis was not only to contribute a novel piece of research to the area of *Multiagent Learning*, *Game theory (cooperative and non-cooperative)* and *Agent Cooperation (coordination)*, but also to generate awareness that key notions like 'teaching' and 'following' have relied heavily on basic human intuition until now.

1.3 Objectives and Approach

In this thesis, we will try to formally define when a strategy can be called a strategy that is both able to teach and follow. To achieve this, we will first take a closer look of the individual behaviours of teaching and following in order to try to formulate a criterion that tries to capture and combine these opposing behaviours into a unified whole. A criterion is a formal requirement an algorithm should adhere to in order to achieve certain (beneficial) properties and many such criteria have already been formulated over recent years. The beneficialness of proposing such a criterion lies in the fact that it remains general enough to analyse and discuss, and specific enough to allow algorithmic implementation. We believe that the latter is also very important, since concepts discussed in this thesis should not only exist on the descriptional level.

1.4 Outline of the document

In the next chapter we will first give a set of basic definitions, which will act as a tool to understand the notions discussed in this thesis. In *Chapter 3* we will take a closer look at the individual behaviours of 'teaching' and 'following', and explain the difficulty of trying to distinguish them. In *Chapter 4* we will take a look at past research concerning the subject of combining these behaviours. In *Chapter 5* we will set up a general criterion and in *Chapter 6*

we will give an implementation following this criterion. In the final chapter we will summarize and discuss the basic concepts described in this thesis.

2. SETTING AND DEFINITIONS

Game theory attempts to capture behaviour in strategic situations, or games, in which an individual's success in making choices depends on the choices of others. In this chapter it is not our goal to completely describe the field of game theory, since there are plenty of books available for this purpose. This chapter should be viewed merely as a piece of reference to which the reader can direct, in which the basic concepts are explained related to this thesis.

2.1 Games in normal form

Under reasonable assumptions about the preference of agents, agents will always have utility functions whose expected payoff they want to maximize. However, in this thesis we will use the term *payoff* to denote utility, since it gives a more intuitive and concrete notion in the context of preference. We consider the setting in which two or more self-interested agents (here self-interested means that agents care only about maximizing their own respective utility) can perform actions that can affect the utilities of other agents. To model these kind of settings, we turn to game theory.

Let us first give a basic intuition to these kind of games, without giving a formal definition. In figure 2.1, we see that both agents can perform the actions C (Comply or Coordinate) or D (Deny, Defect). At first glance, it would be 'best' for both agents to play (C, C), which would give both agents a payoff of 2. However, reasoning from the perspective of an agent, if the opponent plays C it would be best for us to play D, allowing us to receive a much greater payoff. Thus applying this reasoning, it would be better for us to play D. But, if both agents would reason this way, we would end up playing the inferior action profile of (D, D).

The above example described a game in which the player's action completely determines the state of the world. In other words, there are no multiple states and there are is no randomness involved. If played by 2 players, these kind

	C	D
C	**3**,3	**0**,5
D	**5**,0	**1**,1

Fig. 2.1: Prisoner's Dilemma game.

of games are best characterized by tables (one of which we already showed in figure 2.1), and are more generally called *normal-form* games.

Definition 1. *A (finite, n-person) normal-form game is a tuple* (N, A, V)*, where:*

- N *is a finite set of* n *players, indexed by* i*;*
- $A = A_1 \times ... \times A_n$*, where* A_i *is a finite set of actions available to player* i*. Each vector* $a = (a_1, ..., a_n)$ *is called an action profile;*
- $V = (V_1, ..., V_n)$ *where* $V_i : A \mapsto \mathbb{R}$ *is a real valued payoff function for player* i*.*

From this definition however, it is unclear whether or not we can derive that the players in the game know about a) the existence of other players, b) the payoff of other players and c) the strategies available to other players. This is where the notion of *complete information* comes into play.

Definition 2. *A game is said to have* complete information *if every player knows the payoffs and strategies available to other players.*

In this thesis, we make the assumption of complete information, which enables the agents to have a much greater set of beliefs about the other agents.

2.1.1 Repeated games

In repeated games we can distinguish between *finite* and *infinite* repeated games. In repeated games, a given game (in our case a normal-form game) is played multiple times by the same set of players. The game being repeated is called the *stage game*. In finite repeated games, both players can perform a finite amount of actions before the game ends. In infinite repeated games, the game will continue indefinitely. This representation of the repeated game

obscures some key factors. Do agents see what the other agents played earlier? Do they remember what they knew? To this extent, the notion of *perfect information* is introduced in literature.

Definition 3. *A repeated game is said to have* perfect information *if all players know all moves that have taken place.*

In this thesis we consider the setting of perfect information infinite repeated games. Moreover, the problem is *undiscounted*, which means that the agents are equally interested in current payoffs as in future payoffs. If the discount factor is not too low, situations can arise in which cooperation can be sustained, which is arguably the more interesting situation which can arise in repeated games (otherwise the game will just reduce to repeatedly playing the single-shot variant of a game). In undiscounted games, the goal for each player is to maximize the average (expected) payoff he or she receives.

2.1.2 Strategies

A player's strategy in a game is a complete plan of action for whatever situation might arise which fully determines the player's behaviour. We can distinguish between strategies for a normal-form game (stage game) and strategies for a repeated game, the latter obviously being much richer.

When considering strategies for non-repeated games (alternatively – one-shot games), we can distinguish between *pure strategies* and *mixed strategies.*

Definition 4. *A strategy σ is said to be a* pure strategy *if it consists of a single action $a \in A$.*

Using the definition of pure strategies, we can define the set of *mixed strategies.*

Definition 5. *A strategy σ is said to be a* mixed strategy *if it assigns a probability distribution over the set of pure strategies.*

For example, a mixed strategy in the prisoner's dilemma game showed in figure 2.1, a mixed strategy would be a strategy that assign a probability of 0.3 to C and 0.7 to D, which is often times denoted by $(0.3, 0.7)$. Using the definition of pure and mixed strategies, we can identify two important strategies in 2 player games, namely the *Minimax* and *Maximin* strategies.

Definition 6. *In a two-player game, the minimax strategy for player i against player $-i$ is* $\text{argmin}_{\sigma_i} \max_{\sigma_{-i}} V_{-i}(\sigma_i, \sigma_{-i})$*, and player $-i$'s minimax value is* $\min_{\sigma_i} \max_{\sigma_{-i}} V_{-i}(\sigma_i, \sigma_{-i})$*.*

In words, our minimax strategy is the strategy that minimizes the opponent maximum (expected) payoff.

Definition 7. *In a two-player game, the maximin strategy for player i is* $\text{argmax}_{\sigma_i} \min_{\sigma_{-i}} V_i(\sigma_i, \sigma_{-i})$*, and the maximin value (or security value) for player i is* $\max_{\sigma_i} \min_{\sigma_{-i}} V_i(\sigma_i, \sigma_{-i})$*.*

In words, our maximin strategy is the strategy that maximizes our (expected) payoff, given that the opponent is trying to minimize our (expected) payoff. These two strategies play an important role in trying to explain certain general solution concepts, as we will see later.

In repeated games, the set of strategies available is much greater. Here the notion of a strategy can be mapped to our intuition of an *algorithm*. The strategies are commonly defined in terms of *histories*.

Definition 8. *A history $h = ((a_0, p_0), (a_1, p_1), ..., (a_n, p_n))$ is a sequence of action-payoff entries, where (a_t, p_t) denotes the action performed and payoff received at time point t. The length of history h in this case is n.*

One of the most basic strategies which we can define in terms of histories are ones that map a history to a probability distribution over actions. In repeated games, these strategies are contained by the set of *behavioural strategies*. For illustrational purposes, the definition found in literature is supplied below.

Definition 9. *A behavioural strategy $\sigma(h, a)$ returns the probability of playing action a for history h.*

Certainly more classes of strategies can be identified, but this lies beyond the scope of this introductory chapter.

2.2 Analysing games: Solution Concepts

In game theory, a solution concept is a formal rule for predicting how the game will be played. These predictions are called "solutions", and describe

which strategies will be adopted by players, therefore predicting the result of the game. The most commonly used solution concepts for non-repeated games are equilibrium concepts, most famously *Nash equilibrium*.

In game theory, Nash equilibrium is a solution concept of a game with two or more players, in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only his or her own strategy unilaterally. If each player commit to a strategy, and no player can benefit by changing his strategy while the other player's strategy remains the same, the current set of strategies constitutes a Nash equilibrium. It can be proven that every game has at least one Nash equilibrium. For example, the non-repeated prisoner's dilemma has exactly one Nash equilibrium, namely the case in which both players play D.

Another important concept in non-repeated games is that of *Pareto optimality* (alternatively *Pareto efficient*). A pair of strategies is pareto efficient if no agent can improve his expected payoff without another agent losing expected payoff. If we draw the convex hull of expected payoffs that can be reached by playing a mixed strategy in the payoff space, we see that the set of strategies that are Pareto efficient exist on the boundary of this convex hull. Thus, often we will use the notion of Pareto boundary or Pareto frontier to denote this boundary in the convex hull of the payoff space.

In repeated games, a much broader result holds which is showed by the class of theorems called the *Folk theorems*. These theorems state that it is possible for agents to sustain a 'social optimum' if a certain condition is met. This condition states that it must hold that all the agents at least get an outcome that exceeds their minimax value. This condition is formalized in the following definition.

Definition 10. *A payoff profile $V = (V_1, V_2, ...)$ is a payoff profile of some Nash equilibrium if it is feasible (this means that every payoff reachable by getting any combination of outcomes) and enforceable (this means that it must hold that V_i is greater than the minimax value for that respective player for all i).*

An easy way to see this by means of an intuitive 'proof' is by means of a grim trigger strategy. A grim trigger strategy is a strategy that plays the minimax strategy for the rest of the game if our adversary deviates from a certain solution , which causes our adversary to receive no more than his or her respective minimax value for the rest of the game. Now from the

perspective of our adversaries, it is never a good idea to deviate from the course of play if at least his Minimax value is received. If both agents apply the same reasoning, there is no advantage to any player for deviating from the course which will bring out the intended, and arbitrary, outcome, and the game will proceed in exactly the manner to bring about that outcome as long as all the agents receive their minimax value.

3. WHAT CONSTITUTES TEACHER AND FOLLOWER BEHAVIOUR

When attempting to take a closer look at the individual behaviours of teaching and following, the difference between the two is often not so clear-cut as one might presume. In this chapter we will take a closer look at strategies that intuïtively can be understood as teacher and follower strategies. Afterwards we will argue that there is no clear property that sets the teacher strategies apart from the follower strategies. Moreover, there exists large classes of strategies that have no clear intuition to whether or not they can be considered teacher or follower strategies.

3.1 Teacher Strategies

First and foremost, the basic intuition behind teacher strategies is that they try to force cooperation in one way of another. A prime example of this are strategies that are unwilling to deviate from their current way of play. Typical examples of such strategies are the *mixed strategies*.

3.1.1 Mixed Strategies

When talking about mixed strategies, a key facet to remember is that the resulting behaviour is *stationary*: The complete way of play can be given by a single (fixed) probability distribution. Mixed strategies can be seen as teacher strategies because they exhibit an unwillingness to change. In other words, they force an opponent to adapt to our strategy, since the opponent can do no better than this. Another way to view this is that adopting a mixed strategy is equivalent to fixing the setting for the opponent. Thus, for the opponent, the initial multiagent setting reduces to a single-agent setting where our agent is 'part of' the MDP (a very broad introduction to the are of single-agent reinforcement learning can be viewed in [4]).

Following this reasoning, any strategy that pre-calculates a mixed strategy (from the game matrix for example) can be intuitively seen as teacher strategies. One such example is Bully, which determines a pure strategy that maximizes its own payoff given that its opponent is a follower.

3.1.2 Bully

Bully [15] is a deterministic policy that at each round plays the action that maximizes his own payoff given that the opponent does the same. First Bully considers for each of his own actions what the opponent would play in order for the opponent to receive the maximum amount of reward. Then, Bully selects the action which gives him the highest payoff. More formally, Bully plays the action i^* defined by:

$$i^* = \arg\max_i V_i(i, j_i^*),$$

$$\text{where } j_i^* = \arg\max_j V_{-i}(i, j)$$

Here V_i is the payoff function of our agent and V_{-i} the payoff function of our opponent. In words, i^* is defined as the action that maximizes our own payoff, given that our opponent maximizes his payoff on our action (plays j_i^*). Bully can be seen as an algorithm that acts like it has first-mover advantage, or more formally presumes it is a Stackelberg leader in a Stackelberg competition (both of which we will formally define at a later stage in the thesis).

As mentioned before, Bully can be seen as a teacher strategy because it plays a stationary strategy (and in this case also a pure one). Contrary to other stationary strategies, Bully does take into account the payoffs of the other player. Bully behaves in a way that optimizes its payoffs assuming the other agent is a follower, and the follower optimizes its payoffs assuming Bully stays fixed. In this sense, the behaviour is Nash-like: both agents can do no better than to select their designated action, given the assumptions they make about each other.

In a similar line of work, we have strategies that try to punish adversaries as a way to force cooperation. This property makes that these strategies can also be can also be intuitively understood as teacher strategies.

3.1.3 Godfather

Godfather [15] can also be considered a teacher strategy that makes its opponent "an offer it can not refuse". Call a pair of deterministic policies a targetable pair if playing them results in each player receiving more than its security level. This targetable pair is a good solution concept as justified by the Folk Theorem [21]. Godfather chooses a targetable pair and plays its half (its own action belonging to the pair) in the first stage. From then on, if the opponent plays its half of the pair, Godfather continues to play the other half. If however its opponent deviates, Godfather forces its opponent to achieve its security level.

Godfather is a generalization of the Tit for Tat [2] strategy. It is also part of a more general class of strategies that use the threat of security level to maintain a mutually beneficial outcome. In this line of work, Littman and Stone (2004) [16] show that these types of strategies can be used to establish a Nash equilibrium in the repeated game. Since this work demonstrates how teaching strategies can enforce cooperation, we give a short summary of this work in the following paragraph.

In the paper [16], Littman and Stone propose a polynomial time algorithm that uses a punishment phase and a cooperation phase to establish a Nash equilibrium in the repeated game. The algorithm starts out by determining for each player the advantage matrix. The advantage matrix is the same as the reward matrix minus the Minimax value. In words, the advantage matrix describes the extra reward the opponent can get if we force him to receive his security value. The algorithm then proceeds by computing a *feasible* target solution. Here, 'feasible' is formally defined as a target solution that has positive advantages for both players, as justified by the Folk Theorem. As justified by Nash (1950) [17], one particular good solution is a point in the two-dimensional advantage space that maximizes the product of the players advantage. In the convex hull of the advantage space (all the points that can be reached by a linear combination of pure action advantages) this point is in the first quadrant (positive advantages) on the Pareto boundary (else we can find a better solution either up or right in the advantage space). Given this solution, the algorithm decides the pairs of joint actions (i_1, i_2) and (j_1, j_2) that must be played, together with the amount of times they must be played. This is how the cooperation phase should be played, but we have not yet talked about the punishment phase. If during the cooperation phase the

	Left	Right
Top	**1**,1	**0**,0
Bottom	**0**,0	**1**,1

Fig. 3.1: Coordination game.

opponent did not play his half of the target solution, he must be punished. The amount of punishment must be determined in such a way to ensure that cooperation is always a best response. To do this, the average advantage of cooperating must be larger than the average advantage of getting as much advantage as possible by deviating in the cooperation phase and afterwards receiving n number of times 0 advantage in the punishment phase (note that we can always ensure that the opponent receives 0 advantage by forcing him to his security level). From here on out it is possible to determine n, which completes our description of the algorithm. Because of the nature of the algorithm, a Nash equilibrium is reached where both agents can do no better than to cooperate synchronically according to the target solution.

To conclude our section on Godfather, we must still explain why Godfather can be considered a teaching strategy. They reason for this is that Godfather forces opponents to follow. This claim is backed up by the fact that cooperation/punishment-phase learners can indeed establish a Nash equilibrium. Quite literally, Godfather proposes "an offer you can not refuse", so it's best to start following.

3.1.4 Properties and shortcomings of teaching

Reaching this point in the chapter, one might wonder why strategies that can generally be understood a teacher strategies are not sufficient to solve the problem of multiagent learning. The answer is actually quite simple. Teacher strategies work well against opponents who follow, but fail against opponents who cannot be taught, such as other teacher strategies. Moreover, in some particular games, independent of the opponent, certain ways of teaching makes little to no sense. A classic example of the first is that of the coordination game, shown in figure 3.1. If both agents try to teach the other agent their cooperation point by stubbornly repeating that action, and they happen to have a different cooperation point, it results in 0 payoff for both agents forever. In the same manner, we can provide examples how Bully and

	C	D
C	**3**,3	**0**,2
D	**2**,0	**1**,1

Fig. 3.2: Assurance game.

	L	R
U	**1**,-1	**-1**,1
D	**-1**,1	**1**,-1

Fig. 3.3: Matching pennies game.

Godfather can fail against other teacher strategies. To demonstrate this particular shortcoming in the case of Bully, consider the assurance game shown in figure 3.2. In this example, Bully will always play the top row action (C). Of course, given that Bully will always play this, a best-response for the column player is to also choose C. But if Bully is playing against a stationary opponent, e.g. All-D (always play D), Bully will receive a payoff of 0 forever. From the perspective of our agent, it would be better to also play All-D against All-D, assuring a better payoff of 1. The second point we wanted to demonstrate is that in some games, teaching makes little to no sense. In the case of Bully, we refer to the cases in which games require mixed strategies to reach the only equilibria possible, as discussed by Singh, Kearns, and Mansour [24] and Bowling and Veloso [6] (a paper which we will discuss later). In some of these games, teaching a pure strategy makes little to no sense. One such game is the matching pennies game shown in figure 3.3. If we either teach U or D, our opponent will switch to the action that will cause us to receive the worst possible outcome of -1. In these types of games, teaching a pure strategy leads to worse payoff than playing our Maximin strategy. Notice that in literature, such as [19], mixed variants of Bully have been defined.

In the case of Godfather, we can again make the point clear that teaching strategies tend to work bad against other teaching strategies. To demonstrate this point, let us again consider the Prisoner's Dilemma shown in figure 3.4. Regardless of the Godfather variant we use, it will always use (C, C) as a mutual cooperation point, and use D as a punishment action. To make our example even more concrete, let's use the general cooperation/punishment-phase learner by Littman and Stone. This algorithm selects in the coopera-

	C	D
C	**3**,3	**0**,5
D	**5**,0	**1**,1

Fig. 3.4: Prisoner's Dilemma game.

tion phase action C once, and if the opponent deviated from this, it selects D twice. After either the successful cooperation phase or the punishment phase is over, it repeats this process. This strategy does not work well against Bully, which always selects D. The algorithm will repeatedly try to lure its opponent into playing C, but Bully will not listen. The average payoff that the cooperation/punishment-phase learner receives is $\frac{2}{3}$, while Bully receives a much higher average payoff of $2\frac{1}{3}$.

The other point is we were trying to make is that in some games teaching makes little to no sense with the use of particular strategies which we labelled teaching strategies. In the case of Godfather, one such particular case happens when there is no threat (the importance of such threats is described in e.g. [27]). Consider again the matching pennies game shown in figure 3.3. In this game, none of the cooperation points that are beneficial for our agent, namely (D, L) and (U, R), is able to be maintained with the use of threat.

In this section we saw that teacher strategies work well against followers, but not so well against other teacher strategies. From this we can carefully conclude that it is sometimes better to follow against a teacher than to remain stubborn. The next section describes several follower strategies we can identify from the literature.

3.2 Follower Strategies

Follower strategies can intuitively be understood as strategies that condition on the opponent. Here, players maintain beliefs about the opponent, and play according to a Best-Response. This can be summarized by the following scheme, which in literature is referred to as *model-based learners*.

1. Initialize beliefs about the opponents strategy.
2. Repeat:
   (a) Play a best response to the assessed strategy of the opponent.
   (b) Observe the opponents actual play and update beliefs accordingly.

A canonical model-based learner is Fictitious Play, which assumes that the past history of play completely determines the mixed strategy of the opponent, and plays a best response accordingly.

3.2.1 Fictitious Play

In fictitious play (first introduced in 1951 as an iterative way of computing Nash equilibria in zero-sum games [7]), an agent believes that his opponent plays the mixed strategy given by the empirical distribution of the opponent's previous actions. More formally let A be the set of the opponent's actions, and let for each $a \in A$ the number of times that particular action is played denote by $n(a)$. Then Fictitious Play determines that the probability that the opponent is playing a certain action a by:

$$P(a) = \frac{n(a)}{\Sigma_{a' \in A} n(a')}$$

From here on out is is possible to determine a Best-Response, which Fictitious Play then proceeds to play.

It is note-worthy that there are multiple versions and revisions of Fictitious Play, such as 'Fictitious Play with lookahead' and 'smooth Fictitious Play' (used in the regret minimization setting). However, all these variants use the same model-based approach and thus can be considered follower strategies.

3.2.2 Rational Learning

Rational Learning [13] uses the same model-based scheme as Fictitious Play, but the set of possible beliefs about the opponent is much richer. Here the agent considers a set S of all the possible strategies that the opponent can play. The algorithm starts out by determining an a-priori probability

	C	D
C	**0**,0	**1**,1
D	**1**,1	**0**,0

Fig. 3.5: Anti-coordination game.

distribution over every possible strategy in s, denoting the probability that the opponent is playing that strategy. Given any history h, it is possible to determine for each strategy the next action that will be played. Given these actions belonging to the strategies and the probability distribution over these strategies, it is possible for the agent to determine a best-response based on this.

Updating the probability that the opponent is playing a certain strategy s given a history h is done using basic Bayesian updating:

$$P(s|h) = \frac{P(h|s)P(s)}{\Sigma_{s' \in S} P(h|s')P(s')}$$

The main difference with Fictitious Play is that Rational Learning has a much bigger set of beliefs about the opponent. Fictitious Play only considers that its opponent is playing a mixed strategy, and that the frequency of his actions completely determine the distribution. But since Rational Learning uses the same model-based scheme as Fictitious Play, it can be considered a follower strategy.

3.2.3 Properties and shortcomings of following

Again we answer the question why strategies that intuitively can be understood as follower strategies do not solve the problem of Multiagent Learning. These strategies suffer from two major flaws:

1. They can lead to uncoordinated behaviour when facing other followers.

2. They can be easily exploited by teacher strategies.

To demonstrate the first flaw, let us consider the anti-coordination game shown in figure 3.5. Now let us consider that both players use Fictitious Play as strategy. If both players start out with C, they both receive 0 payoff and update accordingly by concluding that it is likely that the opponent will

	Left	Right
Top	**3**,1	**0**,0
Bottom	**0**,0	**1**,3

Fig. 3.6: Battle of the sexes game.

move C again. Thus in the next round, both players will play D, again getting 0 payoff and resulting in a situation where they are indifferent about the next round of play. From here they both start out with the same action as round 1, again receiving payoff 0. It should be no surprise that this process will go on forever. Note however that even though the agents will receive a payoff of 0 forever, they still converge to the repeated mixed Nash equilibrium of $(0.5, 0.5)$. More specifically, a well-known theorem in game theory states that if the empirical distribution of each players strategies converges in fictitious play, then it converges to a Nash equilibrium.

To demonstrate the second flaw, the fact that they are easily forced/exploited by teacher strategies, let us consider the battle of the sexes game shown in figure 3.6. Suppose Fictitious Play is playing against a strategy that repeatedly plays Bully, and switches to Fictitious Play if the average payoff of the last H rounds is less than its security value (where H is a parameter of the algorithm). Since the teacher strategy decides to start out with D (Bully), these strategies will end up playing the equilibrium strategy (D, D) for any H greater than 1. This implies that Fictitious Play will receive an average payoff of 1. If however instead of using Fictitious Play against this strategy, we use a teacher strategy such as Bully, these strategies will end up playing for any finite number H the equilibrium strategy (C, C). The average payoff we receive in the limit is substantially better now: 3 instead of 1.

To conclude this section about follower strategies, we saw that they generally work well against any type of opponent. However, they have drawbacks, such as the problem of uncoordinated behaviour or the fact that they are easily exploitable. A pure teacher strategy is not the answer, but neither is a pure follower. The subject of combining teacher and follower strategies will be the subject of the rest of this thesis.

	Left	Right
Top	**1**,0	**0**,0
Bottom	**0**,0	**1**,0

Fig. 3.7: No-teaching game.

3.3 A separation criterion for teaching and following

We saw that many strategies can intuitively be understood as teaching and following strategies. However, it is very unclear what the distinct property is between these two types of strategy. This might be because the distinction becomes apparent at the behavioural level (the execution of a strategy in a game) but remains unapparent at the algorithmic level (a strategy, which is a mapping from a history to a probability distribution over actions). An important question to ask is whether or not the notion of teaching can exist outside the scope of a game. The following example shown in figure 3.7 shows that this might very well be the case. In this game, the opponent is indifferent about all possible outcomes of the game. Forcing outcomes by leading (bully) or retaliating (godfather) is impossible in this game, since the opponent does not prefer any outcome over another. The Minimax value for this game is 0.5 by adopting the mixed strategy (0.5,0.5). Arguably, the best strategy to adopt in this game is to start out with our defensive Maximin strategy and afterwards play a best response based on the frequency of play of the opponent (e.g. in the case he plays Left more than Right), or in other words, following behaviour. These types of games are evidence that teaching might not be feasible in all games.

However, this does not entirely negate the fact that there may very well exist a formal criterion to set apart teacher and follower strategies. In [10], for example it is argued that what sets a teacher strategy apart from a follower strategy is the fact that teacher strategies also take into account the payoff of the opponent. Indeed, evidence as in [8] suggests that humans apply this process all the time, which might support this argument even further. However, we believe that this does not cover the load entirely, since some strategies can at least be identified intuitively as teacher strategies but do not have this particular property, such as Bully. Another piece of evidence supporting the fact that there is a substantial problem with separating teacher from follower behaviour is the fact that there exists large classes of strategies that cannot be simply promoted as ‘follower’ or ‘teacher’ strategies.

3.3.1 Grey Area

There exists large classes of strategies which cannot be simply promoted as 'follower' or 'teacher' strategies. Such examples include reinforcement learners and no-regret learners.

No-regret learners (a class of learning rules which can be tracked back to the 50s, see e.g. [5]) are also learners that use a bit of both. No-regret learners typically have a strategy pool S, and from this set they estimate (given a certain history) the extra payoff they could have gotten had they followed an optimal strategy from this pool S. This is defined as regret, and no-regret learners strive to minimize this. Since this set S can contain any arbitrary strategy, it can contain both teacher and follower strategies such as Bully and Fictitious Play. In this sense, no-regret learners are perfectly able to demonstrate teaching and following behaviour. On the other hand, no-regret learners are not concerned with teaching and following, they are merely concerned with regret minimization.

Reinforcement learners [25] typically learn by estimating a Q value belonging to an action in a certain state, that estimates how well that action generates payoff. It is not easy to simply state whether or not this approach has a follower mechanic in it (or a teacher for that matter). Although [15] describes these Q learning algorithms as 'follower' algorithms, we believe that we can identify a bit of both: if a reinforcement learner is very certain about a choice of play (i.e., one action has a much larger Q value than the other actions), it plays stubbornly according to that action (teacher). If however it is indifferent about the choice of play (i.e., there exists multiple actions with high Q values), it deploys more of a follower strategy since the opponent can easily steer our reinforcement learner to a certain solution by manipulating our beliefs. In other words, reinforcement strategies can be viewed as strategies that uses both teacher and follower types of behaviour, but on the other hand this is also a kind of a meta-speculation. Reinforcement learners are not concerned with teaching and following, they are merely concerned with average payoff maximization.

There are certainly more classes of algorithms that belong to the 'grey' area: they are neither pure follower nor pure teacher. It is however not our intention to see a teacher and a follower in any arbitrary strategy, thus we leave this subject 'as is'.

4. PAST RESEARCH ON COMBINING TEACHING AND FOLLOWING

This chapter is concerned with the subject of *combining* teaching and following, which is in clear contrast with the previous chapter where we tried to distinguish/categorize them. As it turns out, even though we have no clear formal definition of 'teaching' and 'following', many pieces of research still embody (in some way or another) a combination of the two. Within this research, a clear separation between economic research and AI research can be identified, both with a different intake. Economic research on the subject is mainly concerned with (market) behaviour. Here (market) leaders and followers make up the complex dynamics of the system that arises. The point of departure is the model. Given a realistic market model which can provoke both teaching and following behaviour, what are the interactions that arise? An example of this is the Stackelberg leadership model, which we will address in the next section.

AI research on the other hand is centralized around (artificial) agents. One of the key questions here is: How can we arrange the internal workings (program) of an agent in such a way that the agent will behave in a certain desired way, namely to follow and to teach? This approach is much less focused on the model and much more on the actual agents.

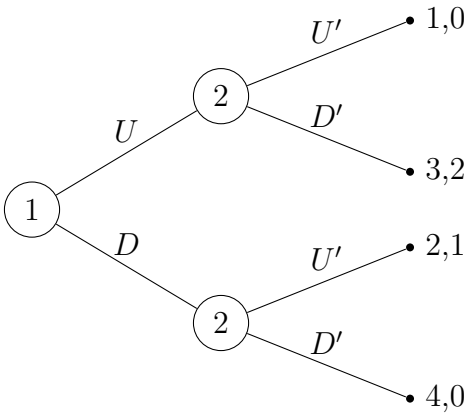
Both intakes can provide a relevant basis when discussing the subject of teaching and following. Thus, in the rest of this chapter, we will provide a short survey of some of the relevant research from both the economic and the AI lines of work. Note that this thesis is mostly concerned with the artificial intelligence approach; it is not our intention to give a comprehensive overview of all the different lines of work within the economic approach. Thus, the next section should be viewed merely as a very basic guide to explain some key concept within the economic line of work.

4.1 Economic Research

4.1.1 Stackelberg competition

In economics, the primary model to illustrate the dynamics of teaching and following behaviour is the so-called *Stackelberg leadership model*. It is named after the German economist Heinrich Freiherr von Stackelberg who published Market Structure and Equilibrium (Marktform und Gleichgewicht [26]) in 1934 which described the model. The players of this game are a *leader* and a *follower* and they compete on quantity. Here, the leader firm moves first and the follower firms move sequentially. The most important aspect of this model is that there exists a firm that has the ability to move first, and that the followers can observe the action taken by the leader firm (and the leader firm knows this, a case of perfect information). Firms may engage in Stackelberg competition if one has some sort of advantage enabling it to move first.

To give an specific example, consider the following Stackelberg game. The game is given in its *extensive-form*: each node (called a decision node) represents a choice that the agent can make as the game is played.



First player 1 moves with either D or U, afterwards the second player moves with either D' or U'. The leaf nodes denote the (joint) payoff of both the players sequentially. As argued by Simaan and Cruz [23], the first player that moves indeed has an advantage over the second player. The player that has the first-move advantage is called the *Stackelberg leader*. We will now introduce the notion of *subgame perfect Nash equilibrium* (SPNE). A strategy profile (strategy for each player) is a subgame perfect Nash equilibrium if it

	L	R
T	**1**,0	**3**,2
B	**2**,1	**4**,0

Fig. 4.1: Non-equivalent Stackelberg game in Cournot form.

represents a Nash equilibrium of every subgame of the original game. This means that if any of the players played any smaller game that consisted of only one part of the larger game that their behaviour represents a Nash equilibrium of that smaller game. The subgame-perfect Nash equilibrium can be deduced by means of backward induction, eliminating various outcomes of the game. To illustrate this, let us consider again our previous example. If player 1 would choose action U, player 2 can do no better than to play D' with joint payoff (3,2). If player 1 would choose action D, player 2 can do no better than to play U' with joint payoff (2,1). Given that player 2 plays his best action in any situation, it is best for player 1 to play action U. This means that the joint action profile (U, D') is a subgame perfect Nash equilibrium, since at every stage in the game neither player can do any better than to play their respective action belonging to this solution.

4.1.2 Cournot competition

The first move gives the leader in Stackelberg a crucial advantage. If this advantage disappears (for example if players have to make their actions simultaneously or if the players have no observability over actions made by the other players), the game reduces to a *Cournot competition*. A Cournot competition (named after Antoine Augustin Cournot) is a competition in which all the agents take their action independently, which is the 'default' game theoretic assumption. Given this model, there is no apparent distinction between teaching/leading and following. Consider the previous extensive form game, but now the advantage of player 1 disappears. The (non-equivalent(!)) Cournot counterpart of this game can be illustrated by a normal-form game representation in figure 4.1. In this game, it is clear that action T for player 1 is strictly dominated by B. If player 1 commits to B, the second player can do no better than to select L. The reverse is also true. Thus in this game, the joint action (B, L) in this game can be identified as the unique Nash equilibrium. The joint-action (T, R) however Pareto-dominates our found Nash equilibrium. A question that follows is: is it still possible to reach this

Pareto dominated solution by means of teaching and following and if this is the case, how do the teaching and following dynamics work in this game?

Even though it initially seems that the notion of teaching and following disappears when the first-mover advantage disappears, this is not entirely true. We can at least identify two cases in which this is not the case:

1. The notion of teaching and following in a Cournot game makes sense when we consider a large or infinitely repeated game. To demonstrate this, consider the previously mentioned game. Initially player 1 will start out with B, since it strictly dominates T. Thus, initially, both players will most likely end up playing (B, L). If however player 1 tries to teach T by repeatedly playing it, player 2 will eventually follow by playing R, since in this case it is better for player 2 to play R than L. Thus in this case, by repeatedly teaching an outcome, it is possible to reach that designated solution.

2. The notion of teaching and following in a Stackelberg game can also make sense if one of the agents acts like it is a Stackelberg competition. For example, if he (the leader) thinks that whatever action he takes, the follower will play L, the leader can do no better than to play D. However, this would not have been the best response of the leader were it that the follower would play R if he (the leader) played T.

4.1.3 Limitations of the economic approach

The economic approach uses the Stackelberg game to evoke the desired behaviour, but makes no claims about how the behaviour can arise in different games. An important question here is whether or not the notion of teaching and following can be reduced to an agent-level (when is an action taken by an agent a teaching action?), or that the notion only exists on a game-level, like the Stackelberg game. If the latter is a given fact, other questions arise, for example whether or not we can identify a teaching/following dynamic for every game (this was already briefly discussed in the previous chapter). This of course is very questionable. Shoham in his book [22] describes on page 200 that in a Cournot competition, the difference between learning and teaching becomes a grey area when we consider games like the coordination game. For these kinds of games, is it possible to extend the notion of Stackelberg leader to a Cournot competition?

4.2 AI Research

Many articles in AI research deal with algorithms that implicitly demonstrate teaching and following behaviour, but only a few of them explicitly mention where the teacher and follower component comes into play. For example, algorithms that converge to a stationary policy can be seen as algorithms that combine both following and teaching behaviour. Here the initial (unconverged) behaviour can be seen as following, and the converged behaviour as teaching. However, we have chosen to summarize and review articles that, regardless of convergence of the algorithm, show following and teaching behaviour more explicitly.

4.2.1 SPaM (Social and Payoff Maximizing)

In the paper "Learning to teach and follow" (2005) by Crandall and Goodrich [10], the authors propose an algorithm called SPaM (Social and Payoff Maximizing), which combines both teaching and following algorithms to create a novel algorithm for two player iterated matrix games. For the teacher component of the algorithm, the authors use an extended version of 'Godfather++', which in fact is the Godfather extension we discussed earlier by Littman and Stone [16]. The contribution of the article lies in the fact that SPaM addresses three issues with Godfather++: it addresses (1) the problem of coordinating actions when the target solution is a sequence of joint actions, (2) the problem of determining the length of the punishment phase and (3) the problem that punishment can sometimes be overly costly.

We will now provide an in-depth description of the algorithm. The initial step of the algorithm is to calculate a target solution c. A target solution is a sequence of joint actions that maximize the positive advantages of the agents. As mentioned, c can be a sequence of actions. Let c^t denote the corresponding joint action c assigns at time-step t, thus $c^t = (c_j^t, c_{-j}^t)$. An assumption that is implicitly being made is that the SPaM agent has complete knowledge of the payoff matrix. After determining the target solution, the algorithm selects an action. This is done by keeping track of both teacher utility and follower utility, by which the algorithm is able to combine both teaching and following behaviour. Let $T(s, a_i)$ be the teacher utility, given state s and action a_i. The teacher utility measures how well this action in the given state induces profitable behaviour from his adversary. The follower utility $F(s, a_i)$, given

a state s and an action a_i, estimates the material payoff the agent expects to receive when choosing this action. The way the algorithm combines both teaching and following is that it chooses (with high probability) the action which maximizes his follower utility (maximizes expected material payoff) given that it also has positive teacher utility (induces profitable behaviour). More formally, let S be the set of actions that have positive teacher utility and, if there is no action with positive teacher utility, contains the action with the highest teacher utility. The set S can be defined as follows:

$$S = \{a_i : T(s, a_i) \geq 0\} \cup \{\arg\max_a T(s, a)\}$$

The action selection proceeds as follows: with high probability $(1-\eta)$, select from this set S the action that maximizes the follower utility. With "low" probability (η), do one of the following: either (a) select an action that has high follower utility regardless whether or not it's in the set S (follower move), or (b) do a random move. More precisely, an action a_i^t at time t is selected using the following scheme:

$$a_i^t = \begin{cases} \arg\max_{a \in S} F(s, a) & \text{with probability } 1 - \eta \\ \arg\max_a F(s, a) & \text{with probability } \rho\eta \\ \text{random} & \text{with probability } (1 - \rho)\eta \end{cases}$$

Initially, for all s and a_i, $T(s, a_i) = 0$ and $F(s, a_i) = 0$. Thus, the agent will start out with a random action regardless, since the set S contains every possible action. In other words, the target solution c is not used to select an action. However, c will be used to construct $T(s, a_i)$ and $F(s, a_i)$ as will be seen later, which indirectly causes the agent to behave according to the target solution. The SPaM algorithm can be summarized with the following diagram:

1. Determine target solution c.
2. Repeat:
   (a) Select an action (based on $T(s, a_i)$ and $F(s, a_i)$).
   (b) Update $T(s, a_i)$ and $F(s, a_i)$ (based on the actions taken and c).

After the action selection procedure, the teacher and follower utilities are updated. To determine the new teacher utility, the notion of 'guilt' is introduced. In short, an agents guilt is the (positive) profit it has obtained from deviating to the target solution in the past. Guilt is tracked for both the player himself and his adversary, as will be seen is necessary for the update rules. When guilt is positive, the agent has benefited from deviating and should be punished. More formally, let r_j^t be the payoff to agent j at time t and let $r_j(c)$ be the average payoff agent j receives when following the target solution sequence c. Using both r_j^t and $r_j(c)$, and the actions that both j and $-j$ took at time step t (a_j^t and a_{-j}^t), the guilt for agent j at time step t can be determined, denoted by G_j^t. Initially, the agent is guiltless ($G_j^0 = 0$). After each action, guilt is updated using the following rules:

1. Player j deviated from the target solution ($a_j^t \neq c_j^t$), and it was guilty ($G_j^t > 0$). The extra reward player j received because he deviated is $r_j^t - r_j(c)$, thus this value should be added to the previous guilt value for player j. However, it is possible that $r_j^t < r_j(c)$, which means that player j deviated, but this deviation was not profitable. This can cause the guilt value for player j to drop below 0, even though he deviated. If this case occurs, set the guilt to some small value $\epsilon > 0$. The update rule becomes:

$$G_j^{t+1} = \max(\epsilon, G_j^t + r_j^t - r_j(c))$$

2. Player j deviated from the target solution ($a_j^t \neq c_j^t$), he is not guilty, but the adversary is. In this case (when the adversary is guilty and you are not), you are justified to deviate, thus you stay guiltless ($G_j^{t+1} = 0$).

3. Player j deviated from the target solution ($a_j^t \neq c_j^t$) and both the agent and the adversary are guiltless. Again, the extra reward player j received because he deviated is $r_j^t - r_j(c)$. However, in the case that $r_j^t = r_j(c)$, player j should still have guilt (because he deviated). This case happens when player j deviates from the target solution, but acquires a similar reward. To ensure in this case that the player still is guilty, a small value $\epsilon > 0$ is added to his guilt value. The update rule becomes:

$$G_j^{t+1} = r_j^t - r_j(c) + \epsilon$$

4. Player j did not deviate from the target solution ($a_j^t = c_j^t$), and he was guiltless. In this case, the player remains guiltless ($G_j^{t+1} = 0$).

5. Player j dit not deviate from the target solution ($a_j^t = c_j^t$), he was guilty and is being punished ($a_{-j}^t \neq c_j^t$), in this case, reduce the guilt by the amount of extra reward he could have gotten had the target solution been played. Thus, the update rule becomes:

$$G_j^{t+1} = G_j^t + r_j^t - r_j(c)$$

6. Player j did not deviate from the target solution ($a_j^t = c_j^t$), he was guilty but the adversary is also playing cooperatively. In this special case, to avoid confusion about the end of the punishment phase, absolve the guilt ($G_j^t = 0$).

After the guilt is determined (for both players), the exact value for the teacher utility can finally be computed. As mentioned before, the teacher utility is a measure of how well an action induces profitable behaviour. More exact, the teacher utility for an action is positive if this action either (a) belongs to the target solution in the case the adversary should not be punished (is guiltless) (b) is a suitable action to punish the adversary if he should be punished (has guilt). More formally, if the adversary has guilt, the teaching utility becomes:

$$T(s, a_i) = \begin{cases} 1 & \text{if } (a_i = c_i^t) \\ -1 & \text{otherwise} \end{cases}$$

Together with the action selection paradigm, this will make sure that with "high" probability $(1-\eta)$ only teaching actions are considered. On the other hand, if the adversary has guilt, he needs to be punished. Now $T(s, a_i)$ is a function of how well action a punishes player $-i$. The teaching utility then becomes:

$$T(s, a_i) = (r_{-i}(c^t) - f_{-i}(s, a_i)) - E_p$$

Here $f_{-i}(s, a_i)$ is a function that determines the average expected payoff for agent $-i$ when selecting action a_i in s. A way to determine this value is by simply looking at the past distribution of actions. The value $r_{-i}(c^t) - f_{-i}(s, a_i)$ is a measure of how well action a_i punishes agent $-i$ in state s. The smaller $f_{-i}(s, a_i)$ gets, and thus the larger $r_{-i}(c^t) - f_{-i}(s, a_i)$ gets, the more agent $-i$ gets punished. The authors note however, that this value alone is not sufficient to determine how well action a_i punishes agent $-i$, since the punishment phase should also be brief. Here, E_p comes into play: E_p denotes the expected punishment level. It is argued that the formula for E_p should be:

$$E_p = \min(G^t_{-i}, r_i(c^t) - m_{-i})$$

Where m_{-i} is the Minimax value for agent $-i$. Since agent $-i$ can guarantee himself a payoff of m_{-i}, it is argued that the agent should not punish his adversary more than $r_i c^t - m_i$. This fully describes the procedure of determining the teacher utility. However, the follower utility also needs to be determined. It is noted that any follower algorithm can be used to determine this utility. The authors use fictitious play with 1-step lookahead in which the follower utility for each action becomes the expected payoff of playing that action. The complete algorithm becomes:

1. Determine target solution c.
2. Instantiate all values to 0: $G^0_i = 0$, $G^0_{-i} = 0$, $T(s, a_i) = 0$ and $F(s, a_i) = 0$.
3. Repeat:
   (a) Select an action (based on $T(s, a_i)$ and $F(s, a_i)$).
   (b) Determine guilt for both players (G^t_i and G^t_{-i}).
   (c) Update $T(s, a_i)$:
   - If player $-i$ is guiltless ($G^t_{-i} \leq 0$), update using:
     $T(s, a_i) = \begin{cases} 1 & \text{if } (a_i = c^t_i) \\ -1 & \text{otherwise} \end{cases}$
   - If player $-i$ has guilt ($G^t_{-i} > 0$), update using:
     $T(s, a_i) = (r_{-i}(c^t) - f_{-i}(s, a_i)) - E_p$

   (d) Update $F(s, a_i)$ using fictitious play with 1-step lookahead.

Discussion

The way the algorithm determines the length of the punishment phase is quite obscure, and with regard to this we believe the authors probably made an error. The length of the punishment phase is determined using guilt: only if the adversary has guilt, he should be punished. However, the authors argue that the following update rule for teacher utility:

$$T(s, a_i) = (r_{-i}(c^t) - f_{-i}(s, a_i)) - E_p$$

is able to determine the length of the punishment phase (by constructing E_p). This claim seems false to us, since guilt is responsible for this. In other words, $(r_{-i}(c^t) - f_{-i}(s, a_i))$ should be sufficient measure to determine how well an action punishes the adversary. Moreover, the action selection paradigm is convoluted. For example, it is very unclear why the algorithm considers random actions, since it is not necessary for the agent to explore (an assumption the algorithm makes is that the full game matrix is known a-priori). Selecting random actions only makes it harder for the other player to predict the behaviour of the agent. Another false claim that is being made is that the algorithm supposedly solves the problem on how to coordinate actions when the target solution is a sequence of actions. The authors mention in the article that if c^t is not played at time t, then $c^{t+1} = c^t$. In other words, if either the agent or the adversary strayed from the target solution joint action at time t, then it is prolonged. This does not solve the coordination problem, since the adversary has no knowledge about the prolonging. A last remark can be made about guilt: this is by no means a "pure" measure for the amount of extra reward an agent has gotten by straying from the target solution. For example, the update rule for guilt when player j deviates from the target solution and was guilty is:

$$G_j^{t+1} = \max(\epsilon, G_j^t + r_j^t - r_j(c))$$

However, it is unclear why the agent should remain guilty. If the adversary strayed from the target solution, and this was not profitable for him, why insist on still punishing him with ϵ? These choices are not well enough motivated by the authors.

Shortcomings of the algorithm can be summarized by the following points. The algorithm:

1. Seems to contain errors and some choices seem arbitrary.

2. Has little scientific basis, merely empirical.

3. Leads to complex behaviour; it is hard for an adversary to predict this.

Especially the last point, the point that the algorithm leads to complex behaviour, implies that the algorithm might not be very suitable for a teacher-role. If it is very hard for the adversary to predict the target solution, one might wonder why SPaM should be favoured above e.g. Godfather++.

4.2.2 AWESOME

In the paper "AWESOME: A General Multiagent Learning Algorithm that Converges in Self-Play and Learns a Best Response against Stationary Opponents" (2006) by Fudenberg and Levine [9], the authors propose an algorithm called AWESOME (Adapt When Everybody is Stationary, Otherwise Move to Equilibrium). To get a good understanding of the algorithm and why it can be understood as an algorithm that possesses both teaching and following aspects, we will discuss it in detail below. We use the line numbers on the algorithmic chart to refer to specific parts of the algorithm. AWESOME starts out in line 1-3 by computing for each player its equilibrium strategy. As explained in the paper, this equilibrium can be any repeated Nash equilibrium, however it is noted that it is not necessarily the case that the algorithm converges to that specific Nash equilibrium. Also note that in order to guarantee convergence to a Nash equilibrium in self play, all the players must either (1) compute the same equilibrium strategy or (2) stochastically switch between strategies. However, the first assumption is acceptable since all the players are using the same algorithm.

Next, the algorithm proceeds in an infinite loop, where each loop represents a restart of the algorithm. The algorithm uses 2 variables APS (All Players Stationary) and APPE (All Players Playing Equilibrium) to keep track of 2 hypothesis: whether or not all players are still obeying the equilibrium and whether or not the players are indeed stationary. The idea is that initially the algorithm will assume that everybody is playing an equilibrium strategy. If however at some point this is assumed to be false, the algorithm will assume that everybody is stationary. If both of these hypothesis' are shown to be false, then the algorithm will restart and repeat everything all over again. We can see in the algorithm that at line 8 both these variables are instantiated at true, and at line 9 our initial play (denoted by ϕ) is indeed our equilibrium strategy. Notice that the variable Me denotes the AWESOME player.

The next while loop, running from line 11 to line 43, depicts whether both the hypothesis' still hold. Each iteration in the while loop depicts an epoch, and as can be seen, t denotes the current epoch we are in. At each epoch, the first thing we do is play for N^t iterations our currently computed way of play. Here N^t is a function that takes an epoch, and returns a number depicting the amount of iterations that need to play. As explained in the paper, N^t increases as t increases, which is needed to prove certain propositions. The

Algorithm 1 AWESOME

for all p **do**
  $\pi_p^* \leftarrow ComputeEquilibriumStrategy(p)$
end for
loop
  for all p **do**
    $h_p^{prev} \leftarrow \{\}$, $h_p^{curr} \leftarrow \{\}$
  end for
  $APPE \leftarrow true$, $APS \leftarrow true$, $\beta \leftarrow false$
  $\phi \leftarrow \pi_{Me}^*$
  $t \leftarrow 0$
  while APS **do**
    for N^t iterations **do**
      $playdistribution(\phi)$
      for all p **do**
        $update(h_p^{curr})$
      end for
    end for
    if $APPE = false$ **then**
      if $\beta = false$ **then**
        for all p **do**
          if $dist(h_p^{curr}, h_p^{prev}) > \epsilon_s^t$ **then**
            $APS \leftarrow false$
          end if
        end for
      end if
      $\beta \leftarrow false$
      $a \leftarrow \arg\max_a V(a, h_{-Me}^{curr})$
      if $V(a, h_{-Me}^{curr}) > V(\phi, h_{-Me}^{curr}) + n|A|\epsilon_s^{t+1}\mu$ **then**
        $\phi \leftarrow a$
      end if
    else
      for all p **do**
        if $dist(h_p^{curr}, \pi_p^*) > \epsilon_e^t$ **then**
          $APPE \leftarrow false$, $\beta \leftarrow true$
          $\phi \leftarrow RandomAction()$
        end if
      end for
    end if
    for all p **do**
      $h_p^{prev} \leftarrow h_p^{curr}$, $h_p^{curr} \leftarrow \{\}$
    end for
    $t \leftarrow t + 1$
  end while
end loop

basic idea behind it is that if the amount of iterations increase, then the probability that the observed way of play is indeed the true/intended way of play also increases. Also note that we also update h_p^{curr} and h_p^{prev}, which we have not explained until now, but they depict for each player p the sequence of actions played at the current and previous epoch respectively.

After playing for N^t iterations our computed strategy, we then proceed to update our hypothesis'. Since initially $APPE$ is set to *true*, we proceed to check in lines 31-38 if all players are obeying their equilibrium strategy. Here $dist$ is defined as the Manhattan distance between distributions, formally $dist(h, h') = \max_{a_i \in A_i} |p_h^{a_i} - p_{h'}^{a_i}|$, where p_ϕ^a is the percentage of time that action a is played in ϕ. The function ϵ_e^t is a monotonically decreasing function defined by the user that denotes the maximum allowed distance between distributions at epoch t when considering equilibrium players. The idea behind the check "$dist(h_p^{curr}, \pi_p^*) > \epsilon_e^t$" at line 21 is that since the length of h_p^{curr} is increasing each epoch, and ϵ_e^t converging to 0, we have at each round a much tighter estimation of whether or not all the players (including the AWESOME player) are actually playing an equilibrium strategy. If this is the case, $APPE$ is set to $false$ (as well as the variable β, which is nothing more than a temporary variable to denote that we just recently refuted the $APPE$ hypothesis) and in the next epoch we proceed to repeatedly play a random selected action. The reason for this becomes apparent in the proof where the author shows convergence to stationary opponents, since this decision implies that there is always a non-zero probability that this random selected action is already the optimal action. The reason to include our own player in this check (which also applies to the stationary check later in the algorithm) is that it allows for synchronization: in self-play, if one player sets $APPE$ to false, then the other players will also necessarily set $APPE$ to false.

When $APPE$ has been set to $false$, we will try to maintain or refute the APS hypothesis in the following epochs (lines 18-25). Observe that in the epoch immediately following when this occurs, β is still set to *true*, which causes us to skip the part where we can potentially set APS to false. The reason for this is that we need to guarantee that in h_{Me}^{prev} we did not play our equilibrium strategy in order to successfully compare h_{Me}^{curr} with h_{Me}^{prev}. At lines 27-30 we check if there is an action that is a better response to the observed history of play than the current action we are playing. The check "$V(a, h_{-Me}^{curr}) > V(\phi, h_{-Me}^{curr}) + n|A|\epsilon_s^{t+1}\mu$" at line 28 ensures that we do not switch to soon (because of possible fluctuations in the observed history). Here h_{-Me}^{curr} is the histories from all the players except the AWESOME player,

ϵ_s^t a monotonically decreasing function that determines when players are considered stationary, n is the amount of players, $|A|$ the maximum number of actions for a single player and μ (also a constant) the payoff difference between our best and worst outcome in the game. Since ϵ_s^t is decreasing, if we are playing a sub-optimal action, at some point in time with non-zero probability we will change our action. Lines 19-25 is again to check whether or not we refute the APS hypothesis. Since h_p^{curr} and h_p^{prev} for each player p are both increasing in length and ϵ_s^t converging to 0, we have again at each epoch a tighter estimation of whether or not all the players (including the AWESOME player) are playing a stationary strategy. Observe that when APS has been set to false, the algorithm will restart all over again, which concludes our description of the algorithm

The author proofs in his paper that the algorithm both converges in self-play to a Nash equilibrium and plays a best response to stationary opponents. However, as shown in the paper, one crucial aspect is that N^t must increase fast enough to ensure that there is a non-zero probability that APS will be set to $false$ on account of the stationary opponents. In Theorem 1 in the paper, the author proves that we can always find such a function for N^t. The completion for the functions N^t, e_e^t and e_s^t are then defined as a schedule. A valid schedule is then a schedule in which N^t grows sufficiently fast to ensure the former mentioned property.

Discussion

Even though the point is not explicitly made in the article, the AWESOME algorithm can be seen as an algorithm that tries to teach and follow. The teaching part consists of trying to coordinate towards a pre-calculated solution, while the following part consists of trying to play a best response if the adversary plays a stationary strategy. However, there is a fundamental problem with the teaching approach, which is the fact that coordination can only be guaranteed if the opponent also plays AWESOME and thus plays the corresponding strategy belonging to the pre-computed target solution. The reason for is because a repeated solution can have arbitrary complexity and thus entails that coordination can only arise if the solution is already pre-computed. Moreover, if agents have conflicting interests in deciding a point of coordination (which is possible in games like the battle of the sexes), this

	Left	Right
Top	$\mathbf{r_{11}}$,c_{11}	$\mathbf{r_{12}}$,c_{12}
Bottom	$\mathbf{r_{21}}$,c_{21}	$\mathbf{r_{22}}$,c_{12}

Fig. 4.2: General payoff game.

approach could remain unsatisfactory for at least one agent.

4.2.3 WoLF-IGA

In the paper "Multiagent learning using a variable learning rate" (2001), Bowling and Veloso propose a new principle named WoLF [6]. This technique states that in situations where the agent is 'winning', a low learning rate is desired, where in situations where the agent is 'losing', the learning rate should be scaled up. This principle is an extension of Infinitesimal Gradient Ascent learning (IGA) by Singh, Kearns and Mansour (2000), in which a variable learning rate is proposed. Contrary to the IGA approach, the WoLF modification allow the players to converge to a Nash equilibrium.

Gradient Ascent learning is based on the principle that players should adjust their (mixed) strategies in such a way to increase their expected payoff. Consider a two-player, two-action, iterated matrix game shown in figure 4.2. Let $V_r(\alpha, \beta)$ be the expected payoff for the row player, given a mixed strategy pair (α, β), and let $V_c(\alpha, \beta)$ denote the same for the column player. The values for $V_r(\alpha, \beta)$ and $V_c(\alpha, \beta)$ can be determined by considering α and β as probabilities, and using probability theory as follows:

$$V_r(\alpha, \beta) = \alpha\beta r_{11} + \alpha(1-\beta)r_{12} + (1-\alpha)\beta r_{21} + (1-\alpha)(1-\beta)r_{22}$$

$$V_c(\alpha, \beta) = \alpha\beta c_{11} + \alpha(1-\beta)c_{12} + (1-\alpha)\beta c_{21} + (1-\alpha)(1-\beta)c_{22}$$

A player can now consider the effect of changing his strategy by computing the partial derivative over these values with respect to his own strategy. This partial derivative for the row and column player are given by $\frac{\partial V_r(\alpha,\beta)}{\partial \alpha}$ and $\frac{\partial V_c(\alpha,\beta)}{\partial \beta}$ respectively. Given these values, the agents can change their mixed strategies in terms of the step size η as follows:

$$\alpha_{t+1} = \alpha t + \eta \frac{\partial V_r(\alpha,\beta)}{\partial \alpha}$$

$$\beta_{t+1} = \beta t + \eta \frac{\partial V_c(\alpha,\beta)}{\partial \beta}$$

If the step size is convergent to 0 ($lim_{\eta \mapsto 0}$), this algorithm is also called Infinitesimal Gradient Ascent (IGA). This learning algorithm can be extended by using a variable learning rate, while also maintaining the infinitesimal step size ($lim_{\eta \mapsto 0}$). The new update rules become:

$$\alpha_{t+1} = \alpha t + l_t^r (\eta \frac{\partial V_r(\alpha,\beta)}{\partial \alpha})$$

$$\beta_{t+1} = \beta t + l_t^c (\eta \frac{\partial V_c(\alpha,\beta)}{\partial \beta})$$

where,

$$0 < l_{min} \leq l_t^{r,c} \leq l_{max}$$

Here, l_{min} and l_{max} are parameters of the algorithm. To conclude the algorithm, the adjustment of l_t^r and l_t^c need to specified. Now the principle of WoLF comes in to play: if the agent is "winning", adopt a low learning rate (l_{min}), and if the agent is "losing", adopt a high learning rate (l_{max}). Still, it needs to be specified what it means to be "winning". Let α^e be an equilibrium strategy for the first player, and let β^e be an equilibrium strategy for the second player (not necessarily the same equilibrium). Then "winning" is defined for a player if it rather plays a different strategy than his equilibrium strategy. More formally, a player is winning if the expected payoff of playing his current strategy is bigger than the expected payoff of playing his equilibrium strategy. The update rules for l_t^r and l_t^c now become:

$$l_t^r = \begin{cases} l_{min} & \text{if } V_r(\alpha_t, \beta_t) > V_r(\alpha^e, \beta_t) \quad \text{(winning)} \\ l_{max} & \text{otherwise} \quad \text{(losing)} \end{cases}$$

$$l_t^c = \begin{cases} l_{min} & \text{if } V_c(\alpha_t, \beta_t) > V_r(\alpha_t, \beta^e) \quad \text{(winning)} \\ l_{max} & \text{otherwise} \quad \text{(losing)} \end{cases}$$

This algorithm, extended from the original IGA algorithm, is called the WoLF-IGA algorithm. The most important aspect of this algorithm is that if both players in a two-person, two-action iterated general-sum game use the WoLF-IGA algorithm, and given that $l_{max} > l_{min}$, the strategies will converge to a Nash equilibrium. The proof for this theorem is quite involved, and can be found in the original paper by Bowling and Veloso.

Discussion

Even though the point is not explicitly made in the article, the WoLF principle can be seen as a form of learning versus teaching. Adopting a low learning rate can be seen as unwillingness to change your strategy, hence teaching behaviour, while adopting a high learning rate can be seen as follower behaviour. On the other hand, the principle can be viewed merely as a technique for machine learning algorithms to ensure faster adaptation. In this context, the 'learning versus teaching' aspect is not so apparent. The article also does not specify any explicit condition when to (gradually) switch between learning and teaching behaviour, it merely provides a heuristic way to adapt to payoff. Also note that in a more recent line of work, this way of policy hill climbing can be exploited by opponents who are initially signalling cooperation by playing as another policy hill climbing player, but afterwards exploit with the use of an estimation of the opponents policy [12].

4.2.4 MetaStrategy

In a multitude of papers found in [19], [20] and [18], the authors propose a criterion that states that an algorithm should achieve a close to optimal payoff against certain classes of opponents with high probability. In this section we will discuss the MetaStrategy found in [19], since we believe it sufficiently shows the basic approach found in the other papers. In this paper, Shoham and Powers propose a new set of criteria for learning algorithms in multi-agent systems is, together with an algorithm that adheres to these properties. These criteria consist of three requirements, namely that a learning algorithm (1) achieves a payoff that approaches the best response payoff against a certain class of opponents (which is a parameter of the algorithm), (2) approaches his security value against other opponents, and (3) achieves a close to optimal payoff in self-play. Furthermore, a novel algorithm which meets this criteria for the class of stationary opponents is proposed. This algorithm, which also performs empirically well, incorporates both learning- and teaching behaviour in a case-by-case algorithm. Thus, this algorithm is showcases how to combine learning and teaching, while also providing a theoretical basis and a high empirical effectiveness.

The basis for the article lies in the three properties that are proposed by the authors. These properties are a continuation of the properties proposed

by Bowling and Veloso [6], namely rationality and convergence. The problem with those properties is that they can not be justified for *every* class of opponents. For example, we cannot demand a learner to converge against every opponent, namely an opponent who periodically switches strategies during play. Thus, three new properties are proposed which are specifically targeted to classes of opponents.

1. **PS-Property 1**: With a probability $1 - \delta$ the algorithm achieves a payoff ϵ-close to V_{BR} against a member *selected from our opponent-class* (which is a parameter of the algorithm), where V_{BR} is the value of the best response to this opponent. The performance bound on this requirement is that it also achieves this goal in polynomial time in terms of $\frac{1}{\delta}$ and $\frac{1}{\epsilon}$ and constants of the game.

2. **PS-Property 2**: Same as the previous, only now in *self-play* with the corresponding self play value $V_{selfPlay}$. Here $V_{selfPlay}$ is defined as the minimum value achieved by a (non-dominated) Nash-equilibrium.

3. **PS-Property 3**: Same as the previous two, only now against *any opponent* with the corresponding security value $V_{security}$, which is defined as normal as his Maxmin value.

The case-by-case algorithm (which we will refer to as 'MetaStrategy') discussed in the paper meets these three properties, but only if we use the stationary opponents as parameter of the first condition. MetaStrategy uses two internal strategies, $BR_\epsilon(\pi)$, which is a ϵ-close best-response strategy to π (note that if we use the distribution of the opponent actions in the full history as π, we get Fictitious Play). More formally, let $EV(\pi_1, \pi_2)$ be the expected payoff of the agent given that the agent plays mixed strategy π_1 and given that the opponent plays π_2, and let $EOV(\pi_1, \pi_2)$ be the expected payoff of the opponent respectively. Then $BR_\epsilon(\pi)$ is defined as follows:

$$BR_\epsilon(\pi) \leftarrow \arg\max_{x \in X}(EOV(x, \pi)),$$

$$\text{where } X = \{y \in \textstyle\prod_1 : EV(y, \pi) \geq \max_{z \in \prod_1}(EV(z, \pi)) - \epsilon\}.$$

In words, X is the set of mixed strategies that are ϵ close to the strategy that maximizes the expected payoff against π, and $BR_\epsilon(\pi)$ selects from this set the action that maximizes the expected opponent payoff against this action.

Moreover, the algorithm also uses $BullyMixed$, which is Bully extended to consider the full set of mixed strategies. More formally, $BullyMixed$ is defined as:

$$BullyMixed \leftarrow \arg\max_{x \in X}(EOV(x, BR(x))),$$
$$\text{where } X = \{y \in \prod\nolimits_1 : EV(y, BR(y)) = \max_{z \in \prod_1}(EV(z, BR(z)))\}$$

In words, X is the set of mixed strategies that maximize the players payoff given that the opponent plays a best response to that strategy. Given this set, select the strategy that maximizes the opponents payoff. The algorithm starts out with an exploration phase in which it determines what the class of his opponent is. If it has reason to believe the opponent is stationary, it settles on $BR_\epsilon(d_0^t)$ where d_0^t is the distribution of opponent actions between 0 and t and ϵ is a closeness parameter that ensures that the algorithm adheres to the mentioned properties. In words, it settles on an ϵ-close best response to the history up until that point. Otherwise if $BullyMixed$ has been performing well, it maintains it. If neither of the conditions hold, it plays a variant of Fictitious Play (based on the last H rounds, where H is a parameter of the algorithm). If, however, it turns out that playing one of these three strategies will result in a lower reward than his security value, the algorithm adopts a Maxmin strategy (which ensures that we reach our safety value).

To summarize MetaStrategy, we can identify a clear step by step approach in the considerations the algorithm makes. The algorithm (1) first considers that the opponent is stationary (thus it plays 'Best Response (based on the first few rounds)'), (2) afterwards considers that the opponent is a follower (thus it plays 'BullyMixed'), (3) if no considerations can be made, concludes that the best we can do is follow (thus it plays 'Fictitious Play').

Discussion

MetaStrategy tries to predict what type of class the opponent is, and based on that either plays a teacher strategy or a follower strategy. Behind this approach lies a beautiful idea: "If our opponent can be influenced, try to influence him, otherwise we will just follow him". This, more or less, can be seen as the very essence of learning and teaching: only teach when our opponent can be taught. But again, the point about teaching and following is not explicitly made. However, as we will motivate in the next chapter, we believe that this approach, using beliefs about our opponent to decide whether to teach or to follow, is the way to go.

5. GENERAL CRITERION AND AN ALGORITHMIC FRAMEWORK FOR TEACHING AND FOLLOWING

5.1 Introduction

In the currently existing AI literature concerning teaching and following, we discovered that there is no formal notion of teaching and following yet defined. To formalize a notion, we need to somehow state what it means to teach and follow. A suitable way to do this is by means of criteria: a criterion is a formal requirement an algorithm should adhere to in order to achieve certain (beneficial) properties. These beneficial properties are usually payoff and convergence insurances (such as convergence to a Nash equilibrium). The first ones to formalize such criteria were Bowling and Veloso in the earlier discussed paper [6], where they stated that an algorithm should be both 'rational' and 'convergent'. The latter requirement states that the learner will necessarily converge to a stationary policy. In their paper however, Bowling and Veloso (as well as Conitzer and Sandholm who made a similar conclusion [9], one which we will look further into later) focused their attention to convergence in self play, since convergence can never be ensured against an arbitrary class of opponents. The idea, to put forth formal criteria for learning, but also effectively limiting the target class, is the basis of the idea found in this chapter.

Before moving on, it is worthwhile to take a step back and summarize what needs to be specified in order to create a successful algorithm that teaches and follows. The following questions summarize the difficulties of following and teaching:

1. What do we teach? What can be taught? This can be seen as the *declarative* aspect of following/teaching.

2. When do we teach and when do we follow? This can be seen as the *conditional* aspect of following/teaching.

3. How do we teach and how do we follow? This can be seen as the *procedural* aspect of following/teaching.

As we will see later, setting up a general criterion for teaching and following that (partly) answers these questions helps us to identify, construct and analyze algorithms that are specifically designed to teach and follow. In the next three sub-sections, we will briefly discuss all the relevant information we have come across in this thesis concerning these three questions. Answering these questions will give us the basic requirements to formulate a criterion for teaching and following.

5.1.1 What to teach (declarative)

It is impossible to simply state what to teach if we have no prior knowledge about the adversaries. Like a tutor knows when mentoring his pupils, if the message you are trying to convey is overly complex, it is wasted effort since it is impossible to actually teach the message such that the pupils can understand it. To relate this to previous MAL literature, consider he earlier discussed paper "Implicit Negotiation in Repeated Games". In this paper Littman and Stone propose an algorithm to establish a Nash equilibrium in the repeated game. The usefulness of adopting such a strategy as an actual way of teaching might not always be a good idea since the message we are trying to teach (in this case a point on the Pareto boundary) can not be taught to every opponent. This is basically one of the most important rules of teaching: only teach something if it can be taught. Thus, the question on what to teach is somehow dependent on our adversaries. We can answer our question with the following: Given that we know/have reason to believe that our adversaries have certain capabilities, the message we teach is the message that gives us the most amount of profit given that it is within the capabilities of our adversaries.

5.1.2 When to teach and when to follow (conditional)

We already stated one of the most important rules of teaching; namely to only teach when something can be taught. If we somehow have a measure of how well something is taught (or can be taught), then we at least know when not to teach. However, this rule is part of a more broader rule, that is only to teach when it is beneficial to do so. Trivial as this rule may seem, the consequence of this (as we will see later) is that teaching simply cannot

occur in every situation (independent of the opponent).
The second aspect of the "when" is timing: when do we play our teaching and following moves? We already concluded that teaching and following behaviour are incompatible in nature. With the tools that current game theory gives us, the solution to this is either to construct a mixed strategy consisting of a teacher and a following strategy, or to alternate them in some sort of way. We believe that the first approach a bit problematic. As argued by Aumann [1], mixed strategies are "intuitively problematic". This requires us to justify why teaching and following should be decided by 'a roll of the die'. The other approach is to combine it on a sequential basis. That is, at some iterations teach, and at other iterations follow: the behaviour remains disjoint. But how do we fill these iterations appropriately with teaching and following behaviour? This should be the choice of the teacher, and generally there is more than one possibility. We can reserve a time period in which we will put all our energy into teaching something to our opponent (which is our approach as we will see later), we can have repeated intervals of teaching and following, we can have teaching behaviour only if a certain condition is met or we can have any combination of these. Again we stress that there is no definite answer to this question, so we note that it is best to choose an option that can at least guarantee the above stated requirements for an algorithm that combines teaching and following behaviour.

5.1.3 How to teach and how to follow (procedural)

We already saw that there are multiple ways to teach, and multiple ways to follow. Teaching and following is always achieved by a form of indirect communication: we are merely able to select an action without stating our intention. To this sense, the declarative and the procedural aspects are closely related. The declarative aspect already told us that we teach the message that gives us the most amount of profit given that it is within the capabilities of our adversaries. The procedural aspect should then simply be the course of action to achieve this, given our knowledge about the temporal aspect of following/teaching.

5.2 A general criterion

In this section, we define a general criterion for teaching and following. We first discuss the notion of 'sequential targeted optimality', which is a criterion

that can be used for combining strategies. Afterwards we show that this criterion is still too general to identify a strategy as a strategy to combine both teaching and following, thus we lay a restriction on the class of sequential targeted optimal strategies to filter out the strategies that actually combine teaching and following.

5.2.1 Targeted Optimality

In the previous section we mentioned that our approach is based on the fact that we try to act within the capabilities of our opponent. This idea to get a best response value against strategies that belong to a certain class of strategies is discussed by Shoham and Brown in [22], where they discuss the concept of targeted learning and use a criterion named (efficient) *targeted optimality*. The following definition is similar, except that we replaced the somewhat vague notion of 'class of opponents' to a set of strategies, which can be any subset of the full strategy set available.

Definition 11. *Given a (finite or infinite) strategy set S, a strategy is said to be* targeted optimal *if it holds that for any choice of $\epsilon > 0$ and $\delta > 0$ there should exist a number of rounds τ, polynomial in $\frac{1}{\delta}$ and $\frac{1}{\epsilon}$, such that for every number of rounds $t > \tau$ the strategy against an arbitrary strategy $\sigma \in S$ achieves average payoff of at least $V_{BR}(\sigma) - \epsilon$ with probability $1 - \delta$, where $V_{BR}(\sigma)$ is the value of the best response given that the opponent plays σ. If for a choice of ϵ and δ the average payoff when playing our strategy remains ϵ-close to the best response value for every number of rounds $t > \tau$, where τ is defined as previous, we say that the property of targeted optimality is* maintained.

Notice that (ϵ, δ)-optimality is quite a weak notion of optimality. To explain this choice in the context of teaching and following, the choice of ϵ gives us room to explore, and the choice of δ gives us room to uncommit to the criteria. To explain this choice in the context of teaching and following, the first can be explained by the fact that sometimes we need room to identify whether or not the opponent can be taught. The latter choice can be explained by the fact that we can never be certain whether or not the opponent actually belongs to the target class. Here, δ is a measure to determine when to abandon our hopes to achieve an average payoff ϵ-close to the best response value. This measure is something which we need when it comes to teaching.

5.2.2 A new criterion for sequential teaching and following

The first step in our construction of our criterion is to use the notion of targeted optimality, and create a new notion in which it is applied sequentially. We propose this novel criterion as *sequentially targeted optimality* (we drop the 'efficient' adjective to keep the criterion name more compact).

Definition 12. *A strategy σ is said to be* sequentially targeted optimal *given strategy sets S^p and S^s if it holds that this strategy first deploys a strategy, referred to as σ^p, and σ^p should be targeted optimal given strategy set S^p. If for a choice of ϵ and δ the property of targeted optimality is not maintained (either because (1) the strategy of the opponent indeed belongs to S^p but with probability δ we have not achieved an average payoff ϵ-close to the best response value or (2) the strategy of the opponent does not belong to S^p), then our strategy should deploy another strategy, referred to as σ^s, and σ^s should be targeted optimal given S^s. If a strategy is sequentially targeted optimal with respect to S^p and S^s, we refer to the first deployed strategy σ^p as the primary strategy, and the second deployed strategy σ^s as the secondary strategy.*

The reason that we also applied the weaker notion of (ϵ, δ)-optimality to the secondary strategy is simply because we want to have room for an algorithm to also adhere to other criteria (and not just one criterion which overrules any other possible criterion). Notice that this criterion already states some of the aspects of teaching and following. It states the "what": we try to achieve the best possible payoff (or at least arbitrary close to) given that we condition on the opponent. It also (partly) states the "when": first we could have a period in which we try to 'teach' the opponent, and if that fails, we could have a period in which we try to 'follow'. However, if we just use an arbitrary primary strategy set and secondary strategy set to create a sequential targeted optimal strategy, this resulting strategy can definitely not be labelled a teaching- and following strategy in all cases. This is because we have not laid any restrictions on these strategy sets and because we have to show that teaching can indeed be beneficial. However, formalizing a notion of teaching and following strategies is problematic, since it is often a grey area as explained earlier in the thesis. To overcome this problem, we will try to define when a sequential targeted optimal strategy is a sequentially teaching-following strategy as a whole, without defining its specific parts S^p and S^s. To do this, we will first introduce the notion of *self-teachability*.

Definition 13. *A strategy σ is self-teachable if it is sequentially targeted optimal given S^p and S^s, using primary strategy σ^p and secondary strategy σ^s, if it holds that $\sigma^p \in S^s$ and $\sigma^s \in S^p$.*

Loosely speaking, a strategy is partially self teachable if we are able to 'follow' (and get our desired best response value) on the strategy which we use to 'teach'. A strategy is fully self-teachable if the previous is the case and if we are also able to 'teach' (and get our desired best response value) on the strategy which we use to 'follow'. Thus, if a strategy is self-teachable it contains some sort of symmetry within the different strategies that are deployed. Using this notion of symmetry, we propose a novel criterion that tries to capture both teaching and following behaviour, which is given by the *sequential teaching-following* criterion in the next definition.

Definition 14. *A strategy is said to be a* sequential teaching-following strategy *if it is self-teachable in a set of games G (that is, it achieves the property of self-teachability in all these games) using strategy sets S^p and S^s and if it holds that in all games belonging to G, the guaranteed best response value of playing against a strategy from S^p is at least as high as the guaranteed best response value of playing against a strategy from S^s:*

$$\min_{\sigma \in S^p} V_{BR}(\sigma) \geq \min_{\sigma' \in S^s} V_{BR}(\sigma')$$

If a strategy is a sequential teaching-following strategy, we refer to the primary strategy as the teacher strategy, and the secondary strategy as the follower strategy.

This definition states when a sequential targeted optimal strategy is a sequential teaching-following strategy without explicitly specifying its parts S^p and S^s and it states a certain beneficialness which is restricted to a set of games. The beneficialness is stated in terms of payoff guarantees (and not in terms of e.g. maximum payoff or expected payoff), because minimum payoff is an important concept in repeated games to identify enforceable outcomes. The restriction to a set of games allows us to form sequential teaching-following strategies that use pure strategies like Bully and Godfather as teaching strategies (recall that some games require mixed equilibrium strategies), without necessarily consorting to mixed variants, since we can just restrict the set of games. Moreover, there is nothing restricting anyone to drop the requirement by creating a sequential teaching-following strategy

that conditions over every game. We believe that this notion captures the essence of teaching and following: here teaching is defined in the very broad sense as 'forcing a more beneficial outcome against certain opponent' and following as 'getting forced in a less beneficial outcome when playing against the former'. As it will turn out, the symmetry (self-teachability) that we demand not only serves as a way to distinguish teaching from following strategy, but also ensures that the algorithm has certain beneficial properties in self-play which we will discuss in the next section.

5.2.3 Formal Properties

The notion of full self-teachability allows us to identify when, in self play, a sequential targeted optimal strategy (such as a sequential teaching-following strategy) is able to engage in an equilibrium. We first introduce the notion of ϵ-Nash equilibrium (for further reading, the concept is well explained in [14]).

Definition 15. *An* ϵ-Nash equilibrium *in a 2-player game is a strategy profile* $\sigma = (\sigma_0, \sigma_1)$ *such that*

$$\forall \sigma_i' \in S_i, i \in \{0,1\}\ V_i(\sigma) \geq V_i(\sigma_i', \sigma_{-1}) - \epsilon$$

Where S_i *is the total strategy pool for player* i *and* V_i *is the expected payoff for player* i *given strategy profile* σ*. In other words, each player can not gain more than* ϵ *in expected payoff by unilaterally deviating from* σ*.*

With this definition, we can show that there exists an alternate way to show when a strategy profile is in fact an ϵ-Nash equilibrium. With this definition, we can show that when an arbitrary self-teachable strategy engages in self play and one player maintains his primary strategy while the other his secondary, a repeated Nash equilibrium is eventually reached.

Proposition 1. *When using a sequential teaching-following strategy in self-play, if it is the case that one player maintains its teacher strategy* $\sigma^p \in S^s$ *while the other maintains his follower strategy* $\sigma^s \in S^p$*, then both players converge to a Nash equilibrium.*

Proof. Since the strategy σ^p is targeted optimal given strategy set S^p for any arbitrary choice of $\epsilon > 0$, and strategy σ^s is targeted optimal given strategy set S^s for every choice of $\epsilon' > 0$, we know that the first player will achieve for

any ϵ an average payoff ϵ-close to $V_{BR}(\sigma^s)$ while the second player will achieve for any ϵ' a payoff ϵ'-close to $V_{BR}(\sigma^p)$. This means that, given an arbitrary ϵ and ϵ', it holds that for the first player there are no strategies available such that more than ϵ expected payoff can be gained and for the second player there are no strategies available such that more than ϵ' expected payoff can be gained. Thus both players can not gain more than $\max(\epsilon, \epsilon')$ by deviating unilaterally, which implies a $\max(\epsilon, \epsilon')$-Nash equilibrium. Since the players maintain their strategies, as time progresses we know that eventually $\epsilon \to 0$ and $\epsilon' \to 0$, and thus $\max(\epsilon, \epsilon') \to 0$, which means that in the limit the players converge to a Nash equilibrium. □

This proposition is important when we want to show when a specific teaching-following strategy converges to a Nash equilibrium in self-play. As we will see later, in order to guarantee convergence to a Nash equilibrium in self-play we also need to consider the case in which both the players maintain their teaching strategy (if possible) and the case in which both players maintain their following strategy. The teaching-following criterion we supplied tried to incorporate the aspects of teaching and following, such as the "what" and the "when". Based on the criterion, it can be argued that in infinitely repeated games, it can be beneficial to first try to teach an outcome that allows us to receive a greater guaranteed outcome. This is especially the case for conservative agents that care more about payoff guarantees than payoff maximization. Many known strategies can be extended to have a teaching phase, so there is not really anything to lose given that the game is not finite. If the rate of convergence plays a role, the criterion also states that the properties should be achieved in efficient time. Moreover, as we will see later on with our algorithm, combining two strategies with the use of the criterion will cause the resulting strategy to maintain many of the properties of the original strategies. In other words, the criterion not only tries to capture the essence of teaching and following, but it is also a beneficial criterion for algorithms to adhere to. Moreover, it allows authors to create strategies in terms of 'weaknesses': what works good against what in which situations? In the next section we will create an algorithm that adheres to our proposed criterion.

5.3 From a general criterion to a specific algorithm

In this chapter, we gave a general criterion for learning and teaching in the form of the sequential targeted optimality criterion. We showed that within all the strategies that adhere to this criterion, some of them can be identified as teaching and following algorithm. We then gave a criterion to identify which strategies that are, called the sequential teaching-following strategies. But, as all the multiagent literature showed us, beyond a basic criterion lies room for implementation. In the next chapter we will answer the "how": how can a strategy adhere to the properties in order to be called a sequential teaching-following strategy?

6. A SEQUENTIAL TEACHING-FOLLOWING STRATEGY

In this chapter we will construct a sequential teaching-following strategy. In the first part, we will devise the teaching part of our strategy and in the second part our following strategy. In the final part we will prove that the combination of these strategies in a sequential way will indeed result in a sequential teaching-following strategy for a specific set of games. The main purpose of this chapter is not only to provide the reader with a sequential teaching-following strategy, but also to make the key concepts (such as proving properties like targeted optimality) more clear.

6.1 The Teaching Strategy

For the teaching part of our strategy, we were inspired to use (a variant of) Bully. We already saw that intuitively this strategy is indeed a teaching strategy, since stationary strategy exhibit "an unwillingness to change". From an economic perspective, Bully is also a great example of a leader strategy, since it assumes it has Stackelberg leader advantage. On the other hand, we also saw that Bully does not work well in all games, in particular some games that require mixed equilibria. However, as we already saw earlier, the sequential teaching-following criteria can be specified for a particular set of games, thus in the long run this will pose no problem for our algorithm.

The idea is that Bully, in some games, works specifically well against opponents that are willing 'to go along with the proposal', such as learning rules that play a best response to the distribution of play. As it turns out, the class of strategies that are susceptible to Bully is very broad and covers many examples found in literature. In the next section we will discuss some of these strategies, called *pure consistent* strategies.

6.1.1 Pure consistent strategies

Before we proceed, we will first re-state the formal definition of *consistent strategies* found in [11].

Definition 16. *A strategy is said to be* ϵ-consistent *if there exists a* T *such that against any stationary strategy* σ_{-i} *and for any* $t > T$ *the strategy achieves a payoff* ϵ*-close to* $V_{BR}(\sigma_{-i})$ *with probability* $1 - \epsilon$*. A strategy is* consistent *if it is* ϵ*-consistent for every positive* ϵ*.*

With this definition, we will introduce a superclass of these consistent strategies, called *pure consistent strategies*.

Definition 17. *A strategy is said to be* ϵ-pure consistent *if there exists a* T *such that against any pure strategy* σ_{-i} *and for any* $t > T$ *the strategy achieves a payoff* ϵ*-close to* $V_{BR}(\sigma_{-i})$ *with probability* $1 - \epsilon$*. A strategy is* pure consistent *if it is* ϵ*-pure consistent for every positive* ϵ *and is said to have a polynomial rate of convergence if* T *is polynomial in* $\frac{1}{\epsilon}$*.*

Since the pure strategies are a subset of the stationary strategies, it is easy to see that every consistent strategy is also pure consistent. In other words, the class of pure consistent strategies is indeed a superclass of the consistent strategies as we mentioned. As it turns out, many strategies can be proven to be pure consistent. This is again one of the beautiful aspects of teaching and following: if the message we are trying to teach is simple, the class we can target is much larger than in the case in which we are trying to teach a more complex message.

In the remainder of this section, we will provide the reader with some classes of algorithms that are known/can be shown to be pure consistent. The first of these is also one of the most well-known, namely Fictitious Play.

Proposition 2. *If* σ *is a Fictitious Play behaviour rule (that is, a fictitious play rule with an arbitrary prior distribution and prior precision, for details see [11] page 2-1) , then it is consistent.*

For the proof we refer to [11] page 2-3. The idea behind a proof for this proposition is that for a long enough history, fictitious play plays a best response to the observed history, and with the use of a probability bound such as Chebyshev's inequality we can show that the probability that the

observed distribution is close to the true distribution increases as the length of the history increases.

Another popular class of strategies are strategies that are universally consistent. As it is noted in [22], universally consistency, Hannan consistency, and exhibiting no regret are all synonymous terms. Loosely speaking, a strategy is universal consistent if it plays a best response against the empirical distribution of actions whether or not the opponent is indeed stationary. Notice that an opponent that is universal consistent is both safe *and* consistent, which implies that any universal consistent also is pure consistent.

The last class of strategies we consider are *rational* strategies. Originally from the Bowling and Veloso [6] paper discussed earlier, where they defined it as follows.

Definition 18. *A strategy is rational, if it holds that if the other players strategies converge to stationary strategies then the learning algorithm will converge to a strategy that is a best-response to the other players strategies.*

Although this definition greatly resembles that of consistent opponents, there is still one key difference. In the case of consistency, it was assumed that the opponent is already stationary. In the above case however, an extra condition is layered upon the definition to show convergence when/if the other players converge. But since a pure strategy is already converged, using the mathematical definition of convergence as ϵ-close to the actual best response value for every $\epsilon > 0$, it immediately becomes apparent that any rational strategy is pure consistent. As it will turn out later, the strategy which we use for the following part of our sequential teaching-following strategy falls in this category (although many authors refuse to use the term "rational" since it coincides with the economic definition of rationality).

In the following section, we will first show which pure strategy is a best response to the set of pure consistent opponents. We will then proceed to show targeted optimality for this pure strategy against the set of pure consistent opponents.

6.1.2 Targeted optimality versus pure consistent opponents

The best response against a pure consistent strategy is to play the pure strategy that, given that the opponent plays his best response, gives us the

	Left	Right
Top	**3**,1	**1**,1
Bottom	**2**,1	**0**,0

Fig. 6.1: Game where two interpretations of Bully are possible.

greatest amount of payoff. One might expect Bully gives us this pure strategy, however, Bully is not well defined for all cases. Consider for example the game shown in figure 6.1. Now given the interpretation we give to "arg max", the action Bully selects can be either U (up) or D (down), since it is not clear what the opponent will choose had we choose the upper row action. The reason for this is that the opponent, given that we play U, is indifferent between the two outcomes L (left) and R (right). To cope with this possible indifference between outcomes, we define our teacher value, $V_{teacher}$, and action, $a_{teacher}$, in the following way:

Definition 19. *The teacher value, $V_{teacher}$, is defined as:*

$$V_{teacher} = \max_i V_i(i, j_i^*)$$

where

$$j_i^* = \operatorname*{argmin}_{j \in J_i} V_i(i, j)$$

and

$$J_i = \{\, a \mid V_{-i}(i, a) = \max_j V_{-i}(i, j) \,\}$$

In short, $V_{teacher}$ is defined as the best possible payoff the agent can guarantee by assuming it has first-mover advantage and by assuming that the opponent plays a best response to this pure strategy which is least beneficial to us. The action belonging to $V_{teacher}$ is defined as $a_{teacher}$.

Observe that $V_{teacher}$ is indeed a best response value against an arbitrary pure consistent opponent (notice that it can still be considered a best response value in repeated games if the opponent is (universally) consistent, as long as we are teaching an enforceable outcome). However, if it is the case that $V_{teacher} < V_{Maximin}$ (recall for example the matching pennies game), it is arguably better to play our (possibly mixed) Maximin strategy. As we will see later, this will not pose a problem since the notion of teaching-following can be restricted to a set of games. The proof that we will use is unique

in the sense that it does not rely on probability bounds to show a probability dependent payoff guarantee. This is because our opponent is using a learning/adaptive strategy (which cannot be simply captured by a Random variable). However, there is a workaround, that is, if we play a pure strategy against a pure consistent strategy, the strategy we play also seems to show 'consistent behaviour'. This idea will be the basis of the upcoming proposal, in which we will show targeted optimality against the set of pure consistent strategies by adopting the strategy in which we repeatedly play $a_{teacher}$.

Proposition 3. *For any choice of $\epsilon > 0$ and $\delta > 0$ against an opponent that uses a pure consistent strategy σ_{-i} with a polynomial convergence rate, there exists a finite T, polynomial in $\frac{1}{\epsilon}$, $\frac{1}{\delta}$ and constants of the game, such that playing $a_{teacher}$ repeatedly will for any $t > T$ result in an average payoff of at least $V_{teacher} - \epsilon$ with probability $1 - \delta$ against this opponent.*

Proof. We will show that for any given value of ϵ, there exists an $\epsilon' > 0$, such that if it is the case that our opponent with probability equal or greater than $1 - \epsilon'$ receives an average payoff ϵ'-close to his optimal payoff, we receive an average payoff ϵ-close to $V_{teacher}$. Since our opponent has a polynomial rate of convergence, we use a polynomial function $T_{-i}(\frac{1}{\epsilon'})$ to denote the actual time steps needed to achieve the property of pure consistency. Without loss of generality, we consider that there is another action profile in the vector, (a_i, a_{-i}) with payoff p'_i and p'_{-i} respectively such that p'_i is the worst payoff in the vector for our agent and p'_{-i} the (second) best for the other agent. Let's also consider that $p_i > p'_i + \epsilon$, since otherwise any combination of actions by the opponent would guarantee that the average payoff we receive is larger or equal than $V_{teacher} - \epsilon$. Similarly we have that $p_{-i} > p'_{-i}$ since by definition of $a_{teacher}$ we have that any action with payoff equal to p_{-i} will net our agent a payoff of at least $V_{teacher}$. For every possible ϵ, the worst-case candidate h to violate the property is playing k proportion $(a_{teacher}, BR(a_{teacher}))$ and $(1-k)$ proportion (a_i, a_{-i}) such that it holds that our opponent still receives an average payoff ϵ'-close to his optimal payoff. Since in this case $V_{teacher} = p_i$ and $V_{BR}(a_{teacher}) = p_{-i}$, we have to find an ϵ' such that the proportion k is high enough such that:

$$k * p_{-i} + (1 - k) * p'_{-i} + \epsilon' \geq p_{-i}$$

implies that the following also holds:

$$k * p_i + (1 - k) * p'_i + \epsilon \geq p_i$$

Solving for ϵ', we see that

$$\epsilon' \leq \epsilon * \left(\frac{p_{-i} - p'_{-i}}{p_i - p'_i} \right)$$

Since we know that $p_i > p'_i$, $p_{-i} > p'_{-i}$ and $\epsilon > 0$, this outcome is strictly positive. Thus for ϵ^* any value in the interval $(0...b]$, where $b = \epsilon * \left(\frac{p_{-i} - p'_{-i}}{p_i - p'_i} \right)$ guarantees that if our opponent (with probability $1 - \epsilon^*$) receives a payoff ϵ^*-close to his optimal payoff then our agent receives a payoff ϵ-close to $V_{teacher}$. Notice that this happens after $T_{-i}(\frac{1}{\epsilon'})$ iterations.

The second step in our proof is to generalize this result to an arbitrary game. The former proof was a worst-case scenario where there exists a joint payoff in the payoff vector such that we achieve the lowest possible payoff and our opponent achieves his second best. We replace the former equation with the following formula:

$$\epsilon' \leq \epsilon * \kappa$$

where

$$\kappa = \left(\frac{\max\limits_{a \in A_2} V_{-i}(a_{teacher}, a) - \max\limits_{a \in S} V_{-i}(a_{teacher}, a)}{V_{teacher} - \min\limits_{a \in A_2} V_i(a_{teacher}, a)} \right)$$

and

$$S = A_2 \backslash \{\ a \mid V_{-i}(a_{teacher}, a) = \max_{a' \in A_2} V_{-i}(a_{teacher}, a')\ \}$$

Notice that this formula is nothing more than a generalisation of our former found value for ϵ' (e.g. p_{-i} corresponds to $\max\limits_{a \in A_2} V_{-i}(a_{teacher}, a)$ and p_i to $V_{teacher}$): the main difference is that we did not demand any of the payoffs to belong to the same action profile. However, by definition $V_{teacher}$ and $\max\limits_{a \in A_2} V_{-i}(a_{teacher}, a)$ are payoffs that belong to the same action profile. It is not hard to see that fixating both the amount of times this action profile is played and the amount of times another arbitrary action profile is played, we can always find a larger value for ϵ' than in the case of repeatedly getting the worst possible payoff for our agent and the second best for the other agent (which was the case for our previous found value for ϵ'). In other words, this is the largest possible range we can find for ϵ' that is small enough to ensure the property.

The third step in our proof is to show that for every later iteration than $T_{-i}(\frac{1}{\epsilon'})$, the average payoff will not decrease. Note that for a small enough

value of ϵ' for the opponent (namely small enough such that there exists no other payoff in the payoff vector that is smaller than $\max_{a \in A_2} V_{-i}(a_{teacher}, a)$ and larger or equal than $\max_{a \in A_2} V_{-i}(a_{teacher}, a) -' \epsilon'$) the opponent can do no better to maintain or increase the proportion k in which $BR(a_{teacher})$ is played. Thus, for a small enough value of ϵ for our agent, the proportion in which we receive $V_{teacher}$ is also maintained or increased. Since in the second part of our proof the calculation for ϵ' was based on achieving the worst possible payoff in the remaining proportion of rounds, it is impossible that our average payoff also drops lower; it is enough that the proportion in which $V_{teacher}$ is achieved remains constant or increases.

The fourth and final step is to prove the actual proposition. Using the earlier defined function T_{-i} and our found value for κ, we see that after $T_{-i}(\max(\frac{1}{\delta}, \frac{1}{\kappa * \epsilon}))$ time steps, we receive for any later time step a payoff ϵ-close to $V_{teacher}$ with probability $1 - \delta$. Since T_{-i} is a polynomial function, we also achieve this polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$. □

With this proof, we have everything we need to proceed to the next part of this chapters, which is the discussion of the following part of our strategy.

6.2 Following Strategy

In the previous section we provided a pure strategy that is targeted optimal with respect to the class of polynomial pure consistent opponents. In this section, if we can come up with strategy that is both polynomial pure consistent and targeted optimal with respect to the class of pure strategies, we can combine these strategies to form a sequential targeted optimal strategy which is self-teachable. Now ideally, we would also like this strategy to perform well in self-play, since this allows our sequential targeted optimal strategy to perform well in self play too. An observant reader might quickly conclude that the algorithm shown in table 'Algorithm 2' is both pure-consistent and converges to a Nash equilibrium in self-play. This is because the hypothesis that our opponent is playing a pure strategy is easily refuted when he is playing at some iteration a different action from the previous iteration. However, we have chosen to use an existing algorithm from the literature, without using any modifications. This particular algorithm not only eventually achieves a best-response value against pure strategies, but also against stationary strategies. This algorithm, in the context of this

Algorithm 2 Pure consistent strategy that converges to NE in self-play

Require: Strategy σ that converges to NE in self-play

$t \leftarrow 0$
$isPure \leftarrow true$
while $isPure$ **do**
  if $t < 2$ **then**
    $playaction(a_{Maxmin})$
  else
    $playaction(BR(a_{-i}^{t-1}))$
    if $a_{-i}^{t-2} \neq a_{-i}^{t-1}$ **then**
      $isPure \leftarrow false$
    end if
  end if
  $t \leftarrow t + 1$
end while
$playstrategy(\sigma)$

thesis, has the following advantageous properties:

1. It converges to a Nash equilibrium in self play and converges to a best response not only against any pure strategy, but also against any stationary strategy.

2. It is highly likely that this particular algorithm is targeted optimal against the set of stationary opponents, meaning it opens the door for possible future research in constructing sequential teaching-following strategies. In the end of this section, we will give a short outline for this particular proof.

3. We are able to show to the reader what is needed to prove targeted optimality for existing algorithms. This again opens the door for possible future research.

The downside is that, in contrast to the algorithm shown in table 'Algorithm 2 , this particular property is a bit trickier to prove.

Looking at existing algorithms that both converge to a Nash equilibrium in self-play and learn to play a best-response against stationary opponents, immediately we see that the earlier discussed WoLF-IGA algorithm is a suitable candidate. Unfortunately, this algorithm is only well-defined for 2-player

2-action games. But since we consider n-action games in this thesis, this algorithm is not suited for our needs. In a similar line of work the ReDVaLeR algorithm [3] does achieve these properties for n-player n-action games. This algorithm, just like the WoLF-IGA algorithm, makes two major assumptions:

1. The opponent's mixed strategy (distribution over actions) is observable.
2. Gradient ascent of infinitesimally small step sizes can be used.

Another algorithm that tries to achieve these 2 properties without making these 2 assumptions is AWESOME [9], which we already discussed earlier in this thesis. This algorithm, on top of being able to converge to a best-response if the opponent converges to a stationary policy, also converges to a Nash equilibrium in self play. In the next section we will prove that AWESOME is targeted optimal with respect to the class of pure strategies and that it is polynomial pure consistent.

6.2.1 Targeted optimality versus pure strategies

Since we are already equipped with a sufficient understanding of AWESOME, we can show a few formal properties of AWESOME. Let us first express the connection between epochs in iterations, using the next proposition.

Proposition 4. *If AWESOME has run a polynomial amount of epochs and given that*

$$N^t = \left\lceil \frac{|A|_\Sigma}{\left(1 - 2^{-\frac{1}{t^2}}\right)\left(\frac{1}{t^2}\right)} \right\rceil$$

where $|A|_\Sigma$ is the total number of actions summed over all players, then the algorithm also has run a polynomial amount of iterations.

Proof. Observe that the function we use for N_t implies that $e_e^t = \frac{1}{t}$ which by Theorem 1 from [9] both belong to a valid schedule. We let $f(V)$ be a polynomial function with respect to variables V that depicts how many epochs the algorithm has run. Observe that since N_t is an increasing function with respect to t, we have that the total amount of iterations the algorithm has run is always equal or less than $\sum_{t=1}^{f(V)} N_t$. But since N_t is increasing, it also

holds that if $t' < t$ then $N'_t < N_t$. From this we can conclude that

$$\sum_{t=1}^{f(V)} N_t \leq f(V).N_{f(V)}$$

Also observe that for all t, $N_t \leq 2(|A|_\Sigma)t^4$. Using this, we see that $f(V).N_{f(V)} \leq 2(|A|_\Sigma)f(V)^5$, and thus we have that

$$\sum_{t=1}^{f(V)} N_t \leq 2(|A|_\Sigma)f(V)^5$$

which is indeed a polynomial function in $f(V)$. □

Proposition 5. *When using the AWESOME algorithm, for any $\delta > 0$ and $\epsilon > 0$, there exists a τ, polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$, such that for any number of rounds $t \geq \tau$ the strategy against an arbitrary pure strategy σ achieves average payoff of at least $V_{BR}(\sigma) - \epsilon$ with probability $1 - \delta$, where $V_{BR}(\sigma)$ is the value of the best response against σ.*

Proof. Using the fact that the observed distribution of play always equals the true distribution ($\phi^t = \phi$ for every epoch t), the problem can be greatly reduced. Notice that since the opponent is playing a pure strategy, we will never restart on account of the opponent, since $dist(h^{curr}, h^{prev}) = 0$ for every possible epoch.
Now assume that the pure strategy our opponent is playing is within d of the pre-computed equilibrium strategy π^*. Also assume that for ϵ_e^t and ϵ_s^t we are using the function $\frac{1}{t}$. Notice that these functions are both monotonically decreasing, thus by theorem 1 in the paper indeed belong to a valid schedule. Observe that AWESOME will set APPE to false when $dist(h^{curr}, \pi^*) > \epsilon_e^t$. Using the fact that $dist(h^{curr}, \pi^*) = d$, it follows that this happens after $\lceil \frac{1}{d} \rceil$ epochs.
After APPE has been set to false, we play a random action. Either this random action is the best response to the pure strategy or there exists another action that allows us to receive greater payoff. If the first situation happens with probability equal or greater than $\frac{1}{|A|}$, we will play this action for the rest of the game. This is because the observed distribution is always the true distribution, which implies that AWESOME will also not restart on account of our own strategy. Given this fact and the fact that AWESOME will also

not restart on account of the opponents, we will play this action for the rest of the game. If the other situation happens with probability equal or less than $\frac{|A|-1}{|A|}$ we will eventually restart the algorithm on behalf of ourselves. Notice that in the first epoch after $APPE$ is set to $false$, β is set to $true$ which disables us to switch. After the next epoch, we will switch when

$$V_i(a^*, \sigma) > V_i(a, \sigma) + n|A|\epsilon_s^{t+1}\mu$$

where a^* is the optimal action, a is the sub-optimal action we are currently playing and σ is the pure strategy of our opponent. Recall that n (the number of players which in our case is 2), $|A|$ (the maximum number of actions for a single player) and μ (the payoff difference between our best and worst outcome in the game) are all constants. We let k be the lowest possible payoff difference between the best and the second best action over each vector in the payoff matrix. Then in the worst possible case (the case which costs us the largest amount of epochs before we switch to $a*$) it holds that:

$$V_i(a^*, \sigma) - V_i(a, \sigma) > n|A|\epsilon_s^{t+1}\mu \tag{6.1}$$

$$k > n|A|\frac{1}{t+1}\mu \tag{6.2}$$

$$t > \frac{n|A|\mu}{k} - 1 \tag{6.3}$$

Thus after $\left\lceil \frac{n|A|\mu}{k} \right\rceil$ epochs, we are guaranteed to have switched actions. Observe that whenever we switch actions, we will restart the algorithm in the next epoch. This is because $dist(h^{curr}, h^{prev}) = 1$ for our agent, and because for all $t > 1$ it holds that $\epsilon_s^t < 1$. Thus after $APPE$ has been set to false, we will either restart after exactly 2 epochs or after $\left\lceil \frac{n|A|\mu}{k} \right\rceil + 1$ epochs, thus after $\max(2, \left\lceil \frac{n|A|\mu}{k} \right\rceil + 1)$ epochs. Now if we let

$$\tau = \left\lceil \frac{1}{d} \right\rceil + \max(2, \left\lceil \frac{n|A|\mu}{k} \right\rceil + 1)$$

we can conclude that we either with probability equal or greater than $\frac{1}{|A|}$ play our optimal action for the rest of the game, or we switch after at most τ epochs.

The final step of our proof is to determine how many restarts we need in

order to increase the probability that we will play a best response to $1-\delta$. The probability that we, after r restarts, play our best response is given by $1-\left(1-\frac{1}{|A|}\right)^r$. Solving the following inequality for r:

$$1-\delta \leq 1-\left(1-\frac{1}{|A|}\right)^r \tag{6.4}$$

$$\delta \geq \left(1-\frac{1}{|A|}\right)^r \tag{6.5}$$

$$r \geq \frac{\ln \delta}{\ln(|A|-1)-\ln(|A|)} \tag{6.6}$$

Thus setting r to $\left\lceil \frac{\ln \delta}{\ln(|A|-1)-\ln(|A|)} \right\rceil$ and using the fact that each restart lasts at most τ epochs, we can conclude that after $r\tau$ epochs we are guaranteed to achieve a payoff that is the actual best response value with probability $1-\delta$. Now we have to calculate how many epochs we have to play in total before the average payoff is ϵ-close to the best response value. Letting T be the total amount of epochs needed, and using the fact that the worst case average payoff after $r\tau$ epochs is $r\tau(V_{BR}(\sigma)-\mu)$, not taking into account that every epoch after $r\tau$ is increasing and assuming that $\epsilon < \mu$, this happens after

$$T(V_{BR}(\sigma)-\epsilon) = r\tau(V_{BR}(\sigma)-\mu) + (T-r\tau)V_{BR}(\sigma) \tag{6.7}$$

$$T = r\tau\mu\frac{1}{\epsilon} \tag{6.8}$$

After $r\tau\mu\frac{1}{\epsilon}$ epochs. Notice that this term is polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$. Using Proposition 4, we can conclude that the amount of actual iterations needed is also polynomial in $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$. □

Proposition 6. *AWESOME is pure ϵ-consistent for every ϵ with a polynomial rate of convergence.*

Proof. This follows immediately from Proposition 5. Filling in for $\delta = \epsilon$,we have that there exists a T such that for any $t > T$ there is a subset of histories of length t, H_t, with $P(\sigma_i, \sigma_{-i})[H_t] \geq 1-\epsilon$, where σ_i is the AWESOME strategy and σ_{-i} is the pure strategy by our opponent, and for all $h \in H_t$ we have that:

$$V_i(\sigma_i, \sigma_{-i})[h] + \epsilon \geq \hat{V}_i(h)$$

□

We have proven that AWESOME is targeted optimal with respect to the set of pure strategies. We have also shown that it is pure ϵ-consistent, which implies that we can potentially use it to follow against pure strategies. In the next section, we will use all our cumulated knowledge to devise an actual sequential teaching-following strategy. However, before we move there, we will devote an (optional) paragraph to give a proof outline for (ϵ, δ)-optimality with respect to the full set of stationary strategies. This will both demonstrate the extra level of complexity this introduces as well as open up the way for possible future research.

The extra complexity that arises is due to the fact that for any given epoch the observed distribution of play does not always equal the true distribution of the opponent. However, we can use the Chebyshev inequality to estimate with what probability the observed distribution at epoch t, ϕ^t, is within the true distribution ϕ. Using the inequality, we know that the probability that ϕ^t is within ϵ of ϕ is at least $1 - \frac{1}{4N^t\epsilon^2}$. We see that as t increases, both N^t and thus the probability that the observed distribution is close to the true distribution increases as well. We can use the fact that if all the payoffs in the payoff matrix are in the range [0...1], then repeatedly playing a best response against a distribution ϕ^t which is within ϵ of the true distribution ϕ, will receive us expected payoff ϵ-close to the optimal payoff (see for example [20] Lemma 1 for this observation). With this inequality, we can give bounds on the probability in which we set $APPE$ to false and the probability that we are getting an ϵ-close best response value. Things get more complex, as there is a non-zero probability that APS will be set to false on account of the opponent (when playing against a pure strategy, this probability was 0). Again using Chebyshev's inequality and using Theorem 1, we can observe that for the valid schedule given in the theorem the probability that at epoch τ we will never restart again is equal to or greater than $2^{-\sum_{t=\tau}^{\infty}(\frac{1}{t})^2}$. In order to solve this infinite series, we can use the Riemann zeta function which provides a functional solution to these types of infinite sums. Solving our infinite sum, we see that this equals to $2^{-\zeta(2)+\sum_{t=1}^{\tau-1}(\frac{1}{t})^2}$ where ζ is the Riemann zeta function (where $\zeta(2) = \frac{\pi^2}{6}$). Now solving the series $\sum_{t=1}^{\tau-1}(\frac{1}{t})^2 \geq C$ for τ, where $C \in [0...\zeta(2))$ is an arbitrary constant, we see that $\tau \geq \frac{1}{\zeta(2)-C} + 1$. From this we see that at every subsequently epoch, the probability that we start again on behalf of our opponent decreases inversely. Observe that we also

have to combine this with the fact that there is also a non-zero probability at every epoch that the algorithm will restart on behalf of ourselves. The challenge is to use all these probability factors (which increase/decrease over time) to eventually prove (ϵ, δ)-targeted optimality with respect to stationary strategies. We again stress that our intent was merely to give a proof-outline, the actual proof is not within the scope of this thesis.

6.3 The algorithm

In this section we will construct the actual sequential teaching-following algorithm. The algorithm will use for the teaching component the pure strategy that is targeted optimal given the class of polynomial pure consistent opponents and for the following component the polynomial pure consistent strategy that is targeted optimal given the class of pure strategies.

6.3.1 A sequential teaching-following strategy

The table 'Algorithm 3' below contains the full algorithmic description. The

Algorithm 3 Sequential teaching-following strategy for games in which $V_{teacher} \geq$

Require: $\langle(\epsilon^p, \delta^p), (\epsilon^s, \delta^s)\rangle$
Ensure: $\epsilon^p > 0$, $\delta^p > 0$, $\epsilon^s > 0$, $\delta^s > 0$
1: $t \leftarrow 0$
2: **while** $(t < T_{-i}(\max(\frac{1}{\delta}, \frac{1}{\kappa * \epsilon})) \vee (\text{AvgPayoff} \geq V_{teacher} - \epsilon^p)$ **do**
3: $\quad playaction(a_{teacher})$
4: $\quad t \leftarrow t + 1$
5: **end while**
6: $playstrategy$(AWESOME)

input parameter $\langle(\epsilon^p, \delta^p), (\epsilon^s, \delta^s)\rangle$ should always be the same for any sequential targeted optimal algorithm: it contains a pair of ϵ and δ values for both the primary and secondary strategy. These parameters, as previously discussed, depict the closeness of the average payoff required and the probability that this will be reached. As we have seen, the lower the values, the longer the teaching/following process will take. The meaning of the κ variable can be found in proposition 3 and the function T_{-i} is a polynomial function that estimates the rate of convergence of the opponent, and can effectively limit

the target class to slow or fast learners (notice that we cannot make the teaching phase too short, since we also have to retain the self-teachability criterion). We again see a beautiful aspect of teaching arise: if the opponent is a slow learner, we might stop on teaching our opponent prematurely. Notice that ϵ^s and δ^s are not used. However, we have already proven that AWESOME achieves average payoff ϵ^s-close to the best response payoff with probability $1-\delta^s$ in polynomial time. A way to see this is that even after the following phase fails (the average payoff is below the best response value), the algorithm will still maintain AWESOME. The two main reasons for this are (both of which will be proven later later):

1. This decision ensures a convergence to a Nash equilibrium in self play for the set of games in which it holds that $V_{teacher} \geq V'_{Minimax}$, where $V'_{Minimax}$ is the pure strategy Minimax value.

2. This decision ensures that the strategy is rational-like: if the strategy of the opponent converges to a stationary policy, the algorithm will converge to a best-response given this stationary policy or we will achieve an average payoff ϵ^p-close to $V_{teacher}$.

Since the best response value against an arbitrary pure strategy is $V'_{Minimax}$, we know that this strategy is a teaching-following for all games in which $V_{teacher} \geq V'_{Minimax}$ (observe that $V'_{Minimax} \geq V_{Maximin}$, which settles our earlier concern that repeatedly playing $a_{teacher}$ is not a best response in games in which $V_{teacher} < V_{Maximin}$). From a game-theoretic viewpoint, this result also makes perfect sense, since in this case we are indeed teaching an enforceable outcome, as justified by the general feasibility theorem/Folk theorem. This observation can be used to prove convergence to a Nash equilibrium in self-play and afterwards we will prove that if the strategy of the opponent converges to a stationary policy, the algorithm will converge to a best-response given this stationary policy or we will achieve an average payoff ϵ^p to $V_{teacher}$

Proposition 7. *In infinitely repeated games, our teaching-following algorithm, restricted to its set of games, will necessarily converge to a Nash equilibrium in self-play if it holds that ϵ_i^p and ϵ_{-i}^p are sufficiently small.*

Proof. First let us define what 'sufficiently small' means: the values for ϵ_i^p and ϵ_{-i}^p are sufficiently small if for both players it holds that there exists no other payoff-profile in the payoff matrix that is greater than or equal to $V_{teacher} - \epsilon_i^p$ for player i and greater than or equal to $V_{teacher} - \epsilon_{-i}^p$ for player

$-i$. Notice that this is not a big restriction, since we can just compute this, and pick such a small value for ϵ^p accordingly.
We distinguish the following three cases in self-play:

1. Both players maintain their primary strategy σ^p. This happens when both agents coincidently achieve an ϵ^p-close best response value while making false assumptions about their opponent. However, our demand for the values of ϵ^p ensure we are indeed teaching $V_{teacher}$ and not settling on another payoff profile. Since we know that this outcome is both feasible and enforceable in our set of games, we know that this outcome belongs to an outcome of a a repeated Nash equilibrium.

2. Both players achieve their best response value when one player uses primary strategy σ^p while the other uses secondary strategy σ^s. If this condition holds, we know that σ^p is targeted optimal given σ^s and vice versa, which implies by Proposition 1 a Nash equilibrium.

3. Both players maintain their secondary strategy σ^s for the rest of the game. Convergence to a Nash equilibrium in this specific case is proven in [9].

□

Moreover, the next proposition shows that our algorithm shows rational-like behaviour.

Proposition 8. *If the strategy of the opponent converges to a stationary policy, the algorithm will converge to a best-response given this stationary policy or we will achieve an average payoff ϵ^p-close to $V_{teacher}$.*

Proof. We can distinguish two cases:

1. We maintain our primary strategy σ^p. This means that our average payoff is ϵ^p-close to $V_{teacher}$.

2. We maintain our secondary strategy σ^s. This strategy ensures that if the strategy of the opponent converges to a stationary policy, the algorithm will converge to a best-response given this stationary policy as proven in [9].

□

	Left	Right
Top	**3**,1	**0**,0
Bottom	**0**,0	**1**,3

Fig. 6.2: Battle of the sexes game.

	Left	Right
Top	**1**,0	**3**,2
Bottom	**2**,1	**4**,0

Fig. 6.3: Stackelberg game.

6.3.2 Discussion of the algorithm

Our teaching-following strategy enables us to teach a repeated Nash equilibrium which provably can be learned by a very broad class of opponents (in contrary to e.g. just playing AWESOME) in efficient time and allows us to switch if the former fails. On top of the beneficial theoretical properties of our algorithm, we believe we can make our discussion of our algorithm even more convincing by looking at some specific games. The following games are examples in which our algorithm is able to perform particularly well.

1. In the battle of the sexes game shown in figure 6.2, our algorithm is able to teach the beneficial outcome of 3. This is in contrast to AWESOME, where it is possible that our agent is assigned (for synchronization purposes in self-play) the equilibrium strategy with the outcome of 1.

2. Our algorithm is able to signal repeated Nash equilibrium outcomes that are easy to learn by the opponent and ensure greater payoff than e.g. the equilibrium of the stage game. This is the case in Stackelberg games (with 'Stackelberg games' we do not mean the formal definition, but rather we refer to [22] where they use this name to distinguish a particular type of game) shown in figure 6.3. In this particular game, our sequential teaching-following strategy is able to teach the outcome that will give our agent a payoff of 3, where as the equilibrium strategy of the stage game gives us a lower payoff of 2.

Furthermore, we have identified the cases in which signalling (teaching) such an 'easy-to-learn' outcome makes sense. Recall that in games like the matching pennies game (in which $V_{teacher} < V_{security}$) shown in figure 6.4, it is better

	Left	Right
Top	**1**,-1	**-1**,1
Bottom	**-1**,1	**1**,-1

Fig. 6.4: Matching pennies game.

to play AWESOME than to repeatedly play $a_{teacher}$.

In this chapter, we constructed a teaching-following algorithm. This strategy uses of a modified variant of Bully to try to teach a feasible and enforceable solution to the class of pure consistent opponents. If this process fails, the strategy will consort to AWESOME, which works well against stationary opponents, and in our particular case the class of pure strategies. We argued that it indeed makes sense to try to teach a feasible and enforceable outcome that is easy to learn instead of playing a complicated social payoff strategy. However, our main goal in this chapter was not necessarily to devise a new algorithm, more so to try and make the key concepts (such as proving properties like targeted optimality) more clear. This chapter (hopefully) opens the way for future research on this particular subject, as we believe this is an exciting new direction to go in. In the next chapter we will provide the reader with some discussion and conclusions to round up the subject of this thesis.

7. DISCUSSION AND FUTURE RESEARCH

In this thesis we discussed the subject of teaching and following in repeated games. First we showed several strategies that can be intuitively understood as 'teacher' strategies, such as Bully. In a similar manner, we showed strategies that can be intuitively labelled as 'follower' strategies, such as fictitious play. We then moved the subject to combining the two, in which we distinguished an economic approach and an A.I. approach. The economic approach was more focussed around games and the A.I. approach more around the strategies of individual agents. An important question that arised was whether or not the subject or teaching and following can be completely separated from games. We then proceeded to give a formal criteria for sequential teaching and following, and argued why this notion is intuitively correct. This notion was also restricted to a set of games, which is in line of the economic approach. A strategy that adheres to the criterion for a set of games, first tries to teach a beneficial outcome to its primary target set by repeatedly playing a certain strategy, and afterwards, if this process has failed, tries to follow (and target a different set of opponents) on the strategy on which it tried to teach. This symmetry in the deployed strategies was a way to distinguish teaching from following strategy without specifically laying restrictions on the deployed strategy as well as serving as a way to show certain beneficial behaviour in self-play. We then gave an example of a teaching-following strategy that as its teaching strategy uses a modified variant of Bully to target the set of pure consistent strategies, and if this process fails consorts to his following strategy (AWESOME) to target the set of pure strategies.

7.1 Discussion

In this thesis we presented the sequential teaching-following criterion as a way of teaching and following. This is not to say that we made a few crucial choices that could potentially be made different. First of all, we must make it clear that it was never our intention to wholly capture the essence of teaching

and following. One such important choice becomes apparent when we defined the notion of self-teachability. The use of a self-teachable strategy helped us to define when a sequential targeted optimal strategy is a teaching-following strategy without specifically defining its parts. Suppose for a moment that we have a sequential teaching-following strategy for a set of games G. Now it is perfectly possible that by reversing the teaching and following component of our strategy, we get a teaching-following strategy for games not in G. In other words, in some settings what we understand as a 'teaching' strategy can function as a 'following' strategy and vice versa. We believe that this is not really a problem since what we wanted was to define teaching in the broad sense as 'forcing cooperation'. But admittedly, there is something more to the fact of what it means to be a teaching strategy and what it means to be a following strategy. Another choice immediately becomes apparent when we define the beneficialness of the teaching part over the following part. We used payoff guarantees to define this beneficialness, which makes sense from the viewpoint of a conservative agent. On the other hand, expected payoff or maximal payoff guarantees also make sense when we consider e.g. greedy agents or risk-taking agents. We made this choice mainly because a minimal payoff guarantee allows us to identify cases in which playing a strategy will necessarily lead to a feasible and enforceable outcome. But again we stress that this was nothing more than a choice.

Another important point of discussion is the fact that the notion is restricted to a set of games. By showing that in some games teaching strategies (other than our Maximin strategy) are not really feasible, we tried to make the point more clear that we really need this restriction. However, this restriction also has its problems. For example: what does it mean that a strategy is restricted to a set of games? Does it mean that the strategy is useless in other games? We have not really give an interpretation to this restriction. It becomes even more troublesome when the payoff matrices are not known. When do we know which strategy to use? We stress that this was never our intention to define; we are merely interested in defining the set of games in which 'it makes sense' to use such a strategy. The exact interpretation of this restriction is up to the creator of the strategy.

We also made a choice with the switching criterion in our definition of sequential targeted optimality. As shown in [19], by smart use of the probability factors δ, we can devise an algorithm that is targeted optimal simultaneously given different classes of opponents, instead of sequentially in our algorithm. If we would allow simultaneous optimization, it could lead to a

different definition of teaching and following.

As a final point of discussion, we note that our definition of sequential teaching-following was not concerned with safety and convergence to Nash equilibria in self-play (although we have given conditions in which this can happen). We note that the latter is the least of our worries, since in a teacher and follower setting one might be less concerned about self-play. It is questionable why we even need to perform well given that we face ourselves, given that we are only concerned whether or not our opponent is a follower. The first point; a safety condition, is arguably more important. Any strategy should be safe to use, else we can just play our security strategy instead. However, we did not feel the need to include this in our criterion; this can simply be a separate criterion instead when devising a sequential teaching-following strategy.

7.2 Future Research

There are many possibilities for future research. First of all, we would really like to see a sequential teaching-following strategy that uses BullyMixed as its teaching strategy and e.g. AWESOME as its following strategy. This strategy targets in its teaching phase the set of consistent opponents (and not necessarily the *pure* consistent opponents) and its following phase the set of stationary opponents (and not necessarily pure strategies). However, we already mentioned that proving targeted optimality for both of these can be a tricky feet.

Another possible point of direction is to extend the notion of teaching-following to n-player games. In this particular case, we have to take into account the fact that our opponents might belong to different categories. The notion of targeted optimality has to be extended to cope with this fact. As shown in [20], checking if multiple opponents belong to a single class can become quite tricky, but is definitely an interesting direction to go in.

In the discussion part, we (more ore less) mentioned another possible point of future research. And that is to investigate the possibility of different forms of teaching and following. In this thesis, we focussed on the sequential case. But it might be perfectly possible to teach and follow in different ways (such as periodic), as discussed earlier in the thesis. For example, dropping sequential optimality in favour of simultaneous optimality might cause interesting behaviour. This might open up the way to new insights

concerning the subject.

Another point of departure is to investigate the exact nature of teaching and following. We used the self-teachability criterion, but we also noted that teacher and follower strategies also have certain properties that allow us to identify them as such. The challenge becomes to devise a formal notion of when a strategy is a teaching strategy and when a strategy is a following strategy.

A last point for possible future research is in settings where the payoff matrices (initially) are not known. If the payoff matrix of the adversary stays hidden throughout, it can be troublesome for teaching strategies, since they rely heavily on the payoff matrix on the opponent. In these belief-based settings, it might be interesting to investigate how teaching and following can still arise.

BIBLIOGRAPHY

[1] Robert J. Aumann. What is game theory trying to accomplish? *Frontiers of Economics*, pages 909–924, 1985.

[2] Robert Axelrod. *The Evolution of Cooperation*. Basic Books, 1984.

[3] B. Banerjee and J. Peng. Performance bounded reinforcement learning in strategic interactions. In *proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 2–7, 2004.

[4] R. Bellman. A markov decision process. *Journal of Mathematical Mechanics*, 6:679–684, 1957.

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

[6] M. Bowling and M. Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136:215–250, 2002.

[7] G.W. Brown. Iterative solutions of games by fictitious play. *In Activity Analysis of Production and Allocation*, 1951.

[8] Colin F. Camerer, Teck H. Ho, J-K. Chong, and Juin kuan Chong. Sophisticated ewa learning and strategic teaching in repeated games. *Journal of Economic Theory*, 104:137–188, 2000.

[9] Vincent Conitzer and Tuomas Sandholm. AWESOME: A General Multiagent Learning Algorithm that Converges in Self-Play and Learns a Best Response against Stationary Opponents. In *proceedings of the 20th International Conference on Machine Learning*, pages 83–90, 2006.

[10] Jacob W. Crandall and Michael A. Goodrich. Learning to teach and follow in repeated games. 2005. Appeared as appendix in *Learning Successful Strategies in Repeated General-sum Games: Appendix A*.

[11] Drew Fudenberg and David K. Levine. Consistency and cautious fictitious play. October 1996.

[12] Yu han Chang and Leslie Pack Kaelbling. Playing is believing: The role of beliefs in multi-agent learning. In *In Advances in Neural Information Processing Systems 14*, pages 1483–1490. MIT Press, 2001.

[13] Ehud Kalai and Ehud Lehrer. Rational learning leads to nash equilibrium. *Econometrica*, 61(5):1019–45, September 1993.

[14] Kevin Leyton-Brown and Yoav Shoham. *Essentials of Game Theory: A Concise, Multidisciplinary Introduction*. Morgan and Claypool Publishers, 2008.

[15] Michael L. Littman and Peter Stone. Implicit negotiation in repeated games. In *proceedings of The Eighth International Workshop on Agent Theories, Architectures, and Languages (ATAL-2001)*, pages 393–404, 2001.

[16] Michael L. Littman and Peter Stone. A polynomial-time nash equilibrium algorithm for repeated games. In *proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, pages 48–54, 2004.

[17] John F. Nash. The bargaining problem. *Econometrica*, 28:155–162, 1950.

[18] Rob Powers and Yoav Shoham. Learning against opponents with bounded memory. In *In IJCAI*, pages 817–822, 2005.

[19] Rob Powers and Yoav Shoham. New criteria and a new algorithm for learning in multi-agent systems. 2005. This work was supported in part by a *Benchmark Stanford Graduate Fellowship*.

[20] Rob Powers, Yoav Shoham, and Thuc Vu. A general criterion and an algorithmic framework for learning in multi-agent systems. In *Machine Learning*, 2007.

[21] Ariel Rubinstein. Equilibrium in supergames with the overtaking criterion. *Journal of Economic Theory*, 21(1):1–9, 1979.

[22] Yoav Shoham and Leyton Brown. *Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations*. Cambridge University Press, 2009. Chapter 7: Learning and Teaching.

[23] M. Simaan and J.B. Cruz Jr. On the stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Applications*, 11(5):533–555, May 1973.

[24] Satinder Singh, Michael Kearns, and Yishay Mansour. Nash convergence of gradient dynamics in general-sum games. In *In Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 541–548. Morgan, 2000.

[25] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[26] Heinrich von Stackelberg. *Marktform und Gleichgewicht*. Springer, Vienna, 1934.

[27] Robert J. Weber. Making more from less: Strategic demand reduction in the fcc spectrum auctions. *Journal of Economics and Management Strategy*, 6(3):529–548, 1997.