An Overview of Cooperative and Competitive Multiagent Learning

P.J. 't Hoen[1], K. Tuyls[2], L. Panait[3], S. Luke[3], and J.A. La Poutré[1,4]

[1] Center for Mathematics and Computer Science (CWI) P.O. Box 94079, 1090 GB Amsterdam, The Netherlands,
[2] Computer Science Department (IKAT), Tongersestraat 6, University of Maastricht, The Netherlands
[3] George Mason University, Fairfax, VA 22030
[4] TU Eindhoven, De Lismortel 2, 5600 MB Eindhoven, The Netherlands
hoen@cwi.nl, k.tuyls@cs.unimaas.nl, {lpanait,sean}@cs.gmu.edu, and hlp@cwi.nl

Abstract *Multi-agent systems* (MASs) is an area of distributed artificial intelligence that emphasizes the joint behaviors of agents with some degree of autonomy and the complexities arising from their interactions. The research on MASs is intensifying, as supported by a growing number of conferences, workshops, and journal papers. In this survey we give an overview of multi-agent learning research in a spectrum of areas, including reinforcement learning, evolutionary computation, game theory, complex systems, agent modeling, and robotics.
MASs range in their description from cooperative to being competitive in nature. To muddle the waters, competitive systems can show apparent cooperative behavior, and vice versa. In practice, agents can show a wide range of behaviors in a system, that may either fit the label of cooperative or competitive, depending on the circumstances. In this survey, we discuss current work on cooperative and competitive MASs and aim to make the distinctions and overlap between the two approaches more explicit.
Lastly, this paper summarizes the papers of the first International workshop on Learning and Adaptation in MAS (LAMAS) hosted at the fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'05) and places the work in the above survey.

1 Introduction

Multi-agent systems (MASs) is an area of distributed artificial intelligence that emphasizes the joint behaviors of agents with some degree of autonomy and the complexities arising from their interactions. The research on MASs is intensifying, as supported by a growing number of conferences, workshops, and journal papers. This book of the first International workshop on Learning and Adaptation in MAS (LAMAS), hosted at the fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'05), is a continuation of this trend.

The goal of the LAMAS workshop was to increase awareness and interest in adaptive agent research, encourage collaboration between Machine Learning (ML) experts and agent system experts, and give a representative overview of current research in the area of adaptive agents. The workshop served as an inclusive forum for the discussion of ongoing or completed work concerning both theoretical and practical issues. More precisely, researchers from the multi-agent learning community presented recent work and discussed their newest ideas for a first time with their peers. An important part of the workshop was dedicated to model MASs for different applications and to develop robust ML techniques. Contributions cover on how an agent can learn using ML techniques to act individually or to coordinate with one another towards individual or common goals. This is an open issue in real-time, noisy, collaborative and possibly adversarial environments.

This introductory article has a twofold goal. The first is to give a broad overview of current MASs research. We present our overview of MASs research from the two main perspectives to be found in the literature; the cooperative and competitive perspective. Secondly, we briefly present an overview of the included papers and invited contributions and place them in the global context of ongoing research.

In cooperative systems, as suggested by the label, the agents pursue a common goal. Such systems are characterized by the fact that the designers of the MAS are free in their design of the agents. The agents can be built and learn with extensive knowledge of the system and the agents can expect benevolent intentions from other agents. Note that we do not claim that it is easy to design a cooperative MAS to have good emergent behavior, on the contrary!

In contrast to cooperative MASs, agents in a competitive MAS setting have non-aligned goals, and individual agents seek only to maximize their own gains. Recent work in competitive MASs has aimed at moving Reinforcement Learning (RL) techniques from the domain of single-agent to multi-agent settings. There is a growing body of work, algorithms and evaluation criteria, which we cover in the second part of our survey. Furthermore, this section also covers a growing body of work on non-cooperative agents [189] for economical and societal settings that have received increasing interest only in recent years. Such agents have their own, possibly conflicting goals and aim for local optimization. Their owners can e.g. be competing companies or autonomous departments within a bigger organization, where the multi-agent systems should facilitate trading, allocation, or planning between these owners, e.g. by means of negotiation or auctioning.

The rest of this document is structured as follows. Section 2 first informally introduces agents playing simple matrix games. We use this section to initially introduce the concepts of play, and whether the agents can be labeled as cooperative, competitive, or as something in between. Section 3 presents our overview of cooperative MASs. Section 4 continues with our overview of competitive MASs. Sections 3 and 4 are intended to be largely self contained, although there are cross-links between the sections. Section 5 presents the papers of this LAMAS proceedings and places this work in the context of the survey of MASs work,

Sections 3 and 4. Lastly, Section 6 concludes with an agenda of future research opportunities for MASs. Appendix A includes some basic Game Theory (GT) concepts universal to the domain of cooperative and competitive MASs as a general background for readers not familiar with the subject.

The next section continues with a discussion on the labels of cooperative and competitive as applied to MASs.

2 Agents classified as Cooperative or Competitive

Multi-Agent Systems range in their description from cooperative to being competitive in nature. To muddle the waters, competitive systems can show apparent cooperative behavior, and vice versa. In practice, agents in a system, depending on the circumstances, can show a wide range of behaviors that may either fit the label of cooperative or competitive.

The fundamental distinction between systems labeled as cooperative or competitive is that for the former the agents are designed with as goal the maximization of a group utility. Competitive agents are solely focused on maximizing their own utility. We, in this section, label the agents as either utilitarian or selfish to stress more their intention, i.e. their design goal, than their actual behavior. For example, a competitive/selfish agent may cooperate with other agents in a temporary coalition. The selfish intentions of the agent are met due to a larger expected reward from cooperation. On the other hand, a cooperative/utilitarian agent may seem competitive if it accidentally hogs a resource to the detriment of other agents in its group. In complex cooperative systems, agents can easily hinder the other agents as the complexity of the interactions increase. The label utilitarian or selfish stresses more the intentional stance of the agent (and of its designer), as opposed to its apparent behavior.

The utilitarian stance for cooperative systems, as already mentioned in the introduction, is also reflected in the design of the agents. Commonly, a cooperative system is designed by one party (be that one designer or a team) to achieve a set of agreed upon goals. The behavior, or the algorithm that learns the behavior of the agents, is largely under the control of the designers of the system. This allows for possible intricate coordination to be a priori implemented in the system and many interactions in the system can be anticipated. An agent can essentially expect good intentions from other agents in the system. This is not the case for the competitive setting. Each agent is created by separate designers that all aim to achieve their own goals. This makes cooperation between selfish agents, even if this is rational, a more difficult and risky task. The designer of a competitive agent must also expend effort in considering the types of exploitive behavior that will be encountered. This distinction in design of agents for a cooperative or competitive setting must be kept in mind when choosing the range of strategies the agents can choose from.

2.1 Setting

In the following, we give a sample of the type of interactions that can be observed between agents. We discuss how these are a consequence of the utilitarian or selfish intentional stance.

We restrict our discussion to the well known two-agent, two-action matrix games. For a complete taxonomy we refer the reader to [132]. Of importance is that the listed games give an exhaustive overview of the types of settings that the agents can encounter. This gives a sound basis to inspect how agents can handle these types of games, both from the utilitarian and the selfish stance. We can then classify the agent behavior as either (apparent) cooperative, (apparent) competitive, or indistinguishable.

[132] classifies games from the perspective of selfish agents; the agents focus on maximizing their own gain, i.e. their private utility. Game theoretical notions prevail in the discussion of the choice of strategies of the agents. We take a slightly broader view and also focus on utilitarian agents and how they would play in the selected games. Utilitarian agents focus on achieving the highest possible group utility, i.e. the sum of their individual rewards.

Note that we only consider play between two selfish agents or between two utilitarian agents. We consider either a system of agents where all agents are intended to achieve a common goal, or a system of agents where all agents expect the worst. We do not cover the intricacies of a cooperative system that has to deal with selfish agents. For a more complete discussion of this topic, we refer the reader to [106] and Section A for a discussion on Evolutionary Stable Strategies.

The agents in the games know the complete payoff matrices. They know their own reward and that of their opponents for all joint actions[1]. They simultaneously must choose an action and receive their part of the reward based on the picked joint action. What they may not know is how the other agent, be that a malicious opponent or a benevolent agent, will play.

Note that we here as yet restrict ourselves to the single play of the presented matrix games. Agents may also have to learn these payoffs during repeated play of the game. We will give examples of this, along with a more formal treatment, in Section 4. After this initial exposition, we discuss how the choice of strategies can change due to repeated play.

2.2 Types of Games

From the viewpoint of selfish agents, [132] broadly classifies the matrix games as either trivial, games of no conflict, games of complete opposition, or as games of partial conflict. The latter is also called a mixed motive game. We discuss each of the categories below. For each category, we sketch the game, give an example, and discuss how selfish and utilitarian agents would cope with the game.

[1] Agents in most Game Theory literature know the payoff matrix before play

Trivial games: In trivial games (TG), the expected reward of an agent does not depend on the choice of action of the other agent. In Table 1, we show such a trivial game. The Row player can choose either action $A1$ or $B1$ while the Column player can choose from actions $A2$ or $B2$. The items in the table show the rewards for the Row player and Column player respectively for choice of action Ai or Bi respectively. For this game, the rewards of one player are not influenced by the choice of actions of the opponent. Such a game is therefore not of great interest in terms of formulating a best strategy. This strategy is based on what they think they should play given the logical action chosen by the opponent, a non-issue in this case.

TG	A2	B2
A1	2,2	2,2
B1	2,2	2,2

Table 1. A trivial game

Due to the simple nature of this game, there is no intrinsic difference in play between utilitarian and selfish agents.

No conflict games: In no-conflict games (NCG), both players benefit from choosing one, unambiguous joint action. Neither player benefits, in terms of individual rewards, by deviating from this logical choice. Consider the game in Table 2:

NCG	A2	B2
A1	4,4	2,3
B1	3,2	2,2

Table 2. A no-conflict game

Both the Row and Column player prefer the joint action $A1A2$ (we give first the Row, and the Column player action) as this gives the most individual reward. Neither player has an incentive to choose another action when the sole goal is maximizing the private utility for selfish agents. $A1A2$ is also the logical choice of action for the utilitarian players. We stress that in both cases, the Row and Column player individually choose $A1$ and $A2$ respectively without prior negotiations; the players base their individual choice solely on their own strategic reasoning.

Note that the Row player may prefer to play $B1A2$ when the Row player aims to maximize the relative utility of play; the Row player wants to have more utility than the Column player. This aspect is not an issue for utilitarian players.

As for trivial games, there is little difference in play between utilitarian and selfish agents. One distinction that can be made is that a utilitarian row player will not pick action $B1$ as such a player is not interested in achieving a higher reward than the other player. More importantly, this choice of action will lower the utility of the group and should be avoided.

Games of Complete opposition (also known as zerosum games): In games of complete opposition (CO), the gain of one agent is a loss for the other agent. The Table 3 shows a typical zerosum game (rewards for one joint action sum to 0). These games are characterized by fierce competition. On average, an agent can expect to have zero reward.

CO	A2	B2
A1	0,0	2,-2
B1	1,-1	-3,3

Table 3. A game of complete opposition

For selfish players, games of complete opposition are a difficult scenario. The best strategy for an unknown opponent, from a game theoretical viewpoint, is to play a random strategy; all actions are equally probable. More technically, this is a mixed strategy. See Section A for a more formal definition. For two utilitarian agents, the game is also problematical as coordination of joint actions, by definition of a zerosum game, will not lead to a higher aggregated reward.

Games of Partial Conflict Mixed Motive Games: Games of partial conflict (PC) allow for both agents to choose profitable actions, but the agents prefer different joint actions. The latter point is the distinction between the no-conflict games and the mixed-motive games. We give an example in Table 4.

PC1	A2	B2
A1	2,7	-1,-10
B1	1,-5	10,1

Table 4. A partial conflict game

The Row agent prefers joint action $B1B2$. The Column agent prefers joint action $A1A2$. Blindly choosing $B1$ by the Row player and $A2$ by the Column player results in joint action $B1A2$ that is preferred by neither player.

Games of partial conflict are difficult for selfish agents. Optimal play is achieved through a mixed strategy that maximizes expected utility. This aspect is handled in more detail in Appendix A.

Utilitarian agents that have as goal to maximize the group utility have a more clearcut strategy; choose the joint action that maximizes the total utility. For Table 4, joint action $B1B2$ is the clear choice. For Table 5, the utilitarian agents are however faced with the choice of playing joint action $A1A2$ or $B1B2$. The agents must however make their choices individually, with no a priori information of the action that will be played by the other agent.

PC2	A2	B2
A1	3,3	1,1
B1	1,1	3,3

Table 5. A second no-conflict game

2.3 Repeated Play

The above section has presented play for agents for single shot play of a selection of typical matrix games. We now focus on how the game can change if two agents repeatedly play the same game. Repeated play opens opportunities, especially to selfish agents, not available in single shot play of the game.

Force-vulnerable or Threat-vulnerable: [132] lists two opportunities in repeated play for selfish agents. Games can be threat-vulnerable or force-vulnerable. A player is called a disgruntled if he fails to achieve his most preferred outcome in initial play of the game. For example, the outcome for the disgruntled Row player is $A1A2$. Two cases can be distinguished: (i) Row's largest payoff is in $A1B2$, and (ii) Row's largest payoff is in $B1B2$.

Consider the first case. Row can only achieve this desired outcome if the Column player shifts away from the original outcome while Row sticks to $A1$. Now by threatening to shift unilaterally to $B1$, Row can effect an outcome where Column gets a smaller payoff than if Column where to shift unilaterally. This game is hence threat-vulnerable. The Row player can induce the Column player to switch by threatening to play an action that is even less preferred by the Column player.

For the second case, suppose now that Row's payoff in $B1B2$ is larger than in $A1A2$. The Column player can be induced to switch to this joint action if first the Row player actually switches to play $B1$. The Column player may then switch to $B2$ if the payoff for the Column player in $B1B2$ is higher than the payoff in $B1A2$. Such a game is called force-vulnerable.

Exploitation: In repeated play a player may learn about the strategy of the opponent. For example, in the games of complete opposition, the Row player may learn that the Column player is not purely random and has a slight bias for playing $B2$. This gives the Row player the opportunity to play action $A1$ more often

for payoff Table 3. For the single shot game, Game Theoretical considerations lead the Row player to play a perfect random strategy for the game of complete opposition. This behavior can change in repeated play as one player learns more of the opposing player and exploitation opportunities are observed.

Threat or Retributive strategies: We have discussed how games may be threat vulnerable. More generally, solutions to games that are not reachable in single play can be achieved in iterated play due to the possibility of retributive actions. A famous example is the Tit-for-Tat strategy [7] in iterated play of the Prisoner's Dilemma, of which an example is shown in Table 6:

PD	A2/C	B2/D
A1/C	3,3	0,5
B1/D	5,0	1,1

Table 6. A Prisoner's Dilemma game

Players receive a reward for jointly cooperating ($A1A2$), but receive a higher reward by unilaterally defecting ($A1B2$ or $B1A2$). The players however achieve a lower reward for a joint defection ($B1B2$). The dominant strategy in the single shot version of the game is to defect, due to the reasoning that the opponent will defect.

In repeated play of the game, a higher reward can be achieved by both players, be they selfish or utilitarian, by repeatedly jointly cooperating. Defections may be less common as a player has the possibility to threaten to punish a defection with defections of its own. This is encoded in the Tit-for-Tat strategy that initially starts the game by cooperating. A defection by the opponent is punished by a defection in the next round of play. The Tit-for-Tat player then reverts to playing cooperation until the next defection by the opponent. Players can achieve a high individual reward over multiple trials of the game, while the threat of retribution guards the player against exploitation by a malicious opponent.

In general, in repeated play, joint solutions of the game by selfish players are possible that are not apparent in single shot play of the game. Future interactions between agents allow for strategies that incorporate threats against exploitation, and at the same time allow for risky joint play.

Give and take: For the games of partial opposition, i.e. the mixed motive games, two agents are each able to gain in each round of play. The agents however have opposed preferences for the choice of joint actions in terms of received individual reward. Repeated play of the same game allows for give and take by both players to achieve a higher aggregated reward than if both players aggressively continuously strive for their own preferred action.

In Table 7, two utilitarian players are indifferent between play of $A1A2$, $A1B2$, or $B1A2$. As long as $B1B2$ is not played, the two agents together reap the highest possible reward. The situation is more complex for two selfish agents.

GT	A2	B2
A1	3,3	2,4
B1	4,2	2,2

Table 7. A game of *G*ive and *T*ake

Two selfish agents are indifferent between $A1A2 - A1A2$, $A1B2 - B1A2$, and $B1A2 - A1B2$ played over two iterations of the game. Both would prefer to receive a reward of 8 over two iterations; the row player would prefer $B1A2 - B1A2$ to be played. There is however the risk of playing $B1B2$ if the column player reasons in a similar manner. The utilitarian players can unilaterally choose to play the safe action $A1$ and $A2$ respectively for the role of Row and Column player as they are only concerned about the group utility.

The selfish players have basically also the above choice; repeatedly play $A1A2$ as a safe, guaranteed joint action. They can also settle for the option of the more complex interleaving of $B1A2$ with $A1B2$. The selfish agents are indifferent between the two strategies in terms of expected reward, although the latter interleaving is more difficult to achieve. For the strategy of repeated play of $A1A2$, both selfish agents have an incentive to deviate. The row player can unilaterally switch to $B1$ and the Column player can decide to unilaterally switch to $B2$ to try to reap the higher reward. This can lead the players to wind up playing $B1B2$, where neither player has a strong incentive to unilaterally switch back.

Observe once again Table 4. Two selfish agents should play $A1A2$ $\frac{9}{14}$ of the time and $B1B2$ $\frac{5}{14}$ of the time for both agents to reap the same average reward. This is a difficult coordination pattern to achieve. This pattern however achieves a higher reward than any mixed strategies the agents can choose due to the risk of penalties for actions $A1B2$ and $B1A2$. Unilaterally striving for their own preferred action by the Row or Column player will lead to lower reward than for the fine grained coordination. The game in Table 5 is hence a challenge for selfish algorithms.

In this section, we have sketched the differences between cooperative and competitive agents using simple matrix games. We have discussed the intricacies that arise when classifying the behavior of an agent from the perspective of single play of a game, and the possible changes in behavior for repeated play of the same game. Sections 3 and 4 then delve into the existing literature covering the state-of-the-art research on cooperative and competitive MASs.

3 Cooperative MASs

In this section, we will focus on the application of *machine learning* to problems in the MAS area. Machine learning explores ways to get a machine agent to discover on its own, often through repeated trials, how to solve a given task. Machine learning has proven a popular approach to solving multi-agent systems problems because the inherent complexity of many such problems can make solutions by hand prohibitively difficult. Automation is attractive. We will specifically focus on problem domains in which the multiple agents are *cooperating* to solve a joint task or to maximize utility; as opposed to *competing* with one another. This is covered in Section 4. We call this specific subdomain of interest *cooperative multi-agent learning*. Despite the relative youth of the field, the number of cooperative multi-agent learning papers is large, and we hope that this survey will prove helpful in navigating the current body of work.

We argue there are two major categories of cooperative multi-agent learning approaches. The first one, *team learning*, applies a single learner to search for behaviors for the entire team of agents. Such approaches are more along the lines of traditional machine learning techniques, but they may have scalability problems as the team size increases. To keep the search space manageable, team learning techniques might assign identical behaviors to multiple team members.

A second category of techniques, *concurent learning*, uses multiple concurrent learning processes. Rather than learning behaviors for the entire team, concurent learning approaches typically employ a learner for each team member, in the hope that this reduces the joint space by projecting it into N separate spaces. However, the presence of multiple concurrent learners makes the environment non-stationary, which is a violation of the assumptions behind most traditional machine learning techniques. For this reason, concurent learning requires new (or significantly modified versions of) machine learning methods.

The last section covers inter-agent communication.

3.1 Team Learning

In team learning, there is a single learner involved: but this learner is discovering a set of behaviors for a team of agents, rather than a single agent. Team learning is an easy approach to multi-agent learning because it can use standard single-agent machine learning techniques: there is a single entity that performs the learning process. Unfortunately, team learning may have problems when scaling to complex domains involving large numbers of agents: given an environment with S states, a team with N agents might be in as many S^N states (assuming multiple agents might be in the same state). This explosion in the state space size can be overwhelming for learning methods that explore the space of state utilities (such as reinforcement learning), but it may not as drastically affect techniques that explore the space of behaviors (such as evolutionary computation) [80, 140, 145]. For such reasons, evolutionary computation seems easier to scale up, and it is by far the most widely used team learning technique.

Team learning may be divided into two broad categories: *homogeneous* and *purely-heterogeneous* team learning. Homogeneous learners develop a single agent behavior which is used by every agent on the team. Purely-heterogeneous team learners develop a unique behavior for each agent - such approaches hold the promise of better solutions through agent specialization, but they must cope with larger search spaces. There exist approaches in the middle-ground between these two categories: for example, divide the team into groups, where group mates share the same behavior. We refer to these as *hybrid* team learning methods.

Choosing among these approaches depends on whether specialists are needed in the team or not. Balch[2] [9] suggests that domains where single agents can perform well (for example, foraging) are particularly suited for homogeneous learning, while domains that require task specialization (such as robotic soccer) are more suitable for heterogeneous approaches. Potter et al [127] suggest that the number of different skills required to solve the domain, and not domain difficulty, is a determinant factor requiring a heterogeneous approach.

Homogeneous Team Learning The assumption that all agents have the same behavior drastically reduces the learning search space. Research in this area includes analyses of the performance of the homogeneous team discovered by the learning process [68], comparisons of different learning paradigms [140], or the increased power added by indirect [131] and direct [83] communication abilities. Learning rules for cellular automata is an oft-overlooked paradigm for homogeneous team learning (a survey of this area is presented in [109]).

Purely-Heterogeneous Team Learning In heterogeneous team learning, the team is composed of agents with different behaviors, with a single learner trying to improve the team as a whole. This approach allows for more diversity in the team at the cost of increasing the search space. The bulk of research in heterogeneous team learning has concerned itself with the requirement for or the emergence of specialists. For example, Luke and Spector [98] compares different strategies for evolving heterogeneous team behaviors. Their results show that restricted breeding (preventing cross-breeding of behaviors for different specialists) works better than unrestricted breeding, which suggests that the specialization allowed by the heterogeneous team representation conflicts with the inter-agent genotype mixture allowed by the free interbreeding. However, the question is not fully answered, as the contradictory result in [69] shows.

Hybrid Team Learning In hybrid team learning, the set of agents is split into several groups, with each agent belonging to exactly one group. All agents in a group have the same behavior. One extreme (a single group), is equivalent to homogenous team learning, while the other extreme (one agent per group) is equivalent to heterogeneous team learning. Hybrid team learning thus permits the experimenter to achieve some of the advantages of each method. Luke et al

[2] Although both the work of Balch and that of Potter et al employ concurrent learning processes, their findings are particularly apropos to our discussion here.

compare the fully homogeneous results with a hybrid combination that divides the team into six groups of one or two agents each, and then evolves six behaviors, one per group [97]. Although homogeneous teams performed better, the authors suggest that hybrid teams might have outperformed the homogeneous ones given more time. Hara and Nagao [67] introduce a method that automatically discovers the optimum number of groups and their compositions.

3.2 Concurrent Learning

The most common alternative to team learning in cooperative multi-agent systems is concurrent learning, where multiple learning processes attempt to concurrently improve parts of the team. Most often, each agent has it own unique learning process to modify its behavior.

Concurrent learning and team learning each have their champions and detractors. While concurrent learning outperforms both homogeneous and heterogeneous team learning in [30, 79], team learning might be preferable in other situations [108]. When then would each method be preferred over the other? Jansen and Wiegand [81] argue that concurrent learning may be preferable in domains for which some decomposition is possible and helpful, and when it is useful to focus on each subproblem to some degree independently of the others.

The central challenge for concurrent learning is that each learner is adapting its behaviors in the context of other co-adapting learners over which it has no control. In single-agent scenarios (where traditional machine learning techniques are applicable), a learner explores its environment, and while doing so, improves its behavior. Things change with multiple learners: the agents' adaptation to the environment can change the environment itself in a way that makes that very adaptation invalid. This is a significant violation of the basic assumptions behind most traditional machine learning techniques.

There are three directions in concurrent learning research. First, research on the *credit assignment* problem deals with how to apportion the team reward to the individual learners. Second, there are challenges in the *dynamics of learning*. Such research aims to understand the impact of co-adaptation on the learning processes. Third, some work has been done on *modeling other agents* in order to improve the interactions (and collaboration) with them.

3.3 Credit Assignment

When dealing with multiple learners, one is faced with the task of divvying up among them the reward received through their joint actions. The simplest solution is to split the team reward equally among each of the learners, or in a larger sense, divide the reward such that whenever a learner's reward increases (or decreases), *all* learners' rewards increase (decrease). This credit assignment approach is usually termed *global reward*.

There are many situations where it might be desirable to assign credit in a different fashion, however. Clearly if certain learners' agents did the lion's share of the task, it might be helpful to specially reward those learners for their actions,

or to punish others for laziness. Similarly, Wolpert and Tumer [197] argue that global reward does not scale well to increasingly difficult problems because the learners do not have sufficient feedback tailored to their own specific actions. In other situations credit assignment *must* be done differently because global reward cannot be efficiently computed, particularly in distributed computation environments. For example, in a robotics foraging domain, it may not be easy to globally gather the information about all items discovered and foraged.

If team reward is not equally divided among the agents, what options are there, and how do they impact on learning? One extreme is to assess each agent's performance based solely on its individual behavior. This approach discourages laziness because it rewards agents only for those tasks they have actually accomplished. However, agents do not have any rational incentive to help other agents, and greedy behaviors may develop. We call this approach *local reward*.

Balch [8, 10] argues that local reward leads to faster learning rates, but not necessarily to better results than global reward. Using local reward leads to better performance in a foraging domain and to worse performance in a simulated soccer domain, as compared to global reward. A few other credit assignment schemes have been proposed as well. Chang et al [33] take a different approach to perform credit assignment: each agent employs a Kalman filter to compute its true contribution to the global reward. Rather than apportion rewards to an agent based on its contribution to the team, one might instead apportion reward based on how the team would have fared differently were the agent not present. Wolpert and Tumer [197] call this the *Wonderful Life Utility*, and argue that it is better than both local and global reward, particularly when scaling to large numbers of agents.

The wide variety of credit assignment methods have a significant impact on our coverage of research in the dynamics of learning, which follows in the next section. Our initial focus will be on the study of concurrent learning processes in fully cooperative scenarios, global reward is used. But other credit assignment schemes may run counter the researchers' intention for the agents to cooperate, resulting in dynamics resembling general-sum or even competitive games, which we also discuss in the next section.

3.4 The Dynamics of Learning

When applying single-agent learning to stationary environments, the agent experiments with different behaviors until hopefully discovering a globally optimal behavior. In dynamic environments, the agent may at best try to keep up with the changes in the environment and constantly track the shifting optimal behavior. Things are even more complicated in multi-agent systems, where the agents may adaptively change each others' learning environments. We believe two tools have the potential to help model and analyze the dynamics of concurrent learners across multiple learning techniques. The first one, Evolutionary Game Theory, EGT was successfully used to study the properties of cooperative coevolution [48, 195], to visualize basins of attraction to Nash equilibria for cooperative coevolution [121], and to study trajectories of concurrent Q-learning

processes [176, 166]. The other tool combines information on the rate of behavior change per agent, learning and retention rates, and the rate at which other agents are learning as well, to model and predict the behavior of existing concurrent learners.

Many studies in concurrent learning have investigated the problem from a game-theoretic perspective. A important concept for such investigations is that of a Nash equilibrium, which is a joint strategy (one strategy for each agent) such that no single agent has any rational incentive (in terms of better reward) to change its strategy away from the equilibrium. As the learners do not usually have control over each others' behaviors, creating alliances to escape this equilibrium is not trivial. For this reason, many concurrent learning methods will converge to Nash equilibria, even if such equilibria correspond to suboptimal team behaviors.

Fully Cooperative Scenarios Research in simple stateless environments shows that multiple cooperating concurrent learners can greatly benefit from being optimistic about their teammates: the goal is not to match well your current teammates, but to expect them to improve as well due to their learning [85, 122]. Scaling up to environments with states is computationally demanding. Wang and Sandholm [185] present the *Optimal Adaptive Learning* algorithm, which is guaranteed to converge to optimal Nash equilibria if there are a finite number of actions and states; unfortunately, the time required for the algorithm to achieve such optimality guarantees may be exponential in the number of agents. Environments where the state can only be partially observed (usually due to the agents' limited sensor capabilities) represent even more difficult (also more realistic) settings. The task of finding the optimal policies in partially observable Markov decision process (POMDP) is PSPACE-complete [124], and it becomes NEXP-complete for decentralized POMDPs [17]. Preliminary research for such domains is presented in [125, 114].

General Sum Games Unequal-share credit assignment techniques can inadvertently place learning in rather non-cooperative scenarios. For such reasons, general sum games are applicable to the cooperative learning paradigm, even though in some situations such games may not be in any way cooperative. Following the early work of Littman [92], there has been significant recent research in concurrent (and not necessarily cooperative) learning for general-sum games [26]. Concurrent learning algorithms for such settings[3] range from Nash-Q [76], Friend-or-Foe Q-learning [93], EXORL ([161]), Correlated-Q [62], to WolF [27].

3.5 Teammate Modeling

A final area of research in concurrent learning is teammate modeling: learning about other agents in the environment so as to make good guesses of their expected behavior, and to act accordingly (to cooperate with them more effectively,

[3] The algorithms are usually tested on general-sum and competitive domains, and only very rarely in cooperative problems.

for example). For example, agents may use Bayesian learning to create models of other agents, and use such models to anticipate their behavior [32]. Suryadi and Gmytrasiewicz [162] present a similar agent modeling approach consisting of learning the beliefs, capabilities and preferences of teammates. As the correct model cannot usually be computed, the system stores a set of such models together with their probability of being correct, given the observed behaviors of the other agents. On the other hand, modeling teammates is not a must for better coordination [146]. Finally, Wellman and Hu suggest that the resulting behaviors are highly sensitive to the agents' initial beliefs, and they recommend minimizing the assumptions about the other agents' policies [192].

3.6 Learning and Communication

For some problems communication is a necessity; for others, communication may nonetheless increase agent performance. We define *communication* very broadly: altering the state of the environment such that other agents can perceive the modification and decode information from it. Among other reasons, agents communicate in order to coordinate more effectively, to distribute more accurate models of the environment, and to learn subtask solutions from one another.

But are communicating agents really *multi-agent?* Stone and Veloso argue that unrestricted communication reduces a multi-agent system to something isomorphic to a single-agent system [160]. They do this by noting that without any restriction, the agents can send complete external state information to a "central agent", and to execute its commands in lock-step, in essence acting as effectors for the central agent.

Explicit communication can also significantly increase the learning method's search space, both by increasing the size of the external state available to the agent (it now knows state information communicated from other agents), and by increasing the agent's available choices (perhaps by adding a "communicate with agent i" action). As noted in [40], this increase in search space can hamper learning an optimal behavior by more than communication itself may help.

Direct Communication Many agent communication methods employ, or assume, an external communication method by which agents may share information with one another. The method may be constrained in terms of throughput, latency, locality, agent class, etc. Examples of direct communication include shared blackboards, signaling, and message-passing. The literature has examined both hard-coded communication methods and learned communication methods, and their effects on cooperative learning overall. Tan [169] and Berenji and Vengerov [15] suggest that cooperating learners can use communication to share different knowledge about the environment in order to improve team performance. Other research provides the agents with a communication channel but does not hard-code its purpose; the task is for the agents to discover a language for communication [183].

Indirect Communication Indirect communication methods are those which involve the *implicit* transfer of information from agent to agent through modification of the world environment. Examples of indirect communication include: leaving footsteps in snow, leaving a trail of bread crumbs in order to find one's way back home, and providing hints through the placement of objects in the environment (perhaps including the agent's body itself). Much of the indirect communication literature has drawn inspiration from social insects' use of pheromones to mark trails or to recruit other agents for tasks [75]. Pheromones are chemical compounds whose presence and concentration can be sensed by fellow insects [22], and like many other media for indirect communication, pheromones can last a long time in the environment, though they may diffuse or evaporate. Several pheromone-based learning algorithms have been proposed for foraging problem domains (such as [110]).

This section has presented cooperative MASs. The next section continues with MASs from a competitive perspective.

4 Competitive MASs

4.1 Preamble

The previous section has presented an overview of the literature concerning cooperative MASs. These systems are characterized by the fact that the agents implicitly or explicitly have as common goal to work together. The agents are benevolent and choose actions to promote the overall utility of the system. This is not an easy task, as discussed in Section 3, but the programmers of the agents in principle are free to design the agents that cooperate and truthfully exchange information to promote the desired cooperation. This is however not the case for more competitive settings where the individual agents have non-aligned goals.

Competition in inherent in human interaction. The field of economics is founded on this principle. Game Theory is an analytical offshoot where the goal is to mathematically analyze the strategies required for detailed scenarios, smaller in domain than usually encountered in economics. Electronic Agents in competitive settings have been introduced and studied for broadly two types of settings that we cover here:

- E-commerce;Market-Based Games; bargaining/negotiations, markets and market mechanisms, and auctions.
- Multi-Agent RL (MARL) usually for more restricted settings; matrix games.

The above two distinctions are not exhaustive for the field of competitive agents as a whole. They are however two dominant streams of research. We treat each in a separate section below, although there are overlaps.

4.2 Design of adaptive software agents for Market-Based multi-agent games

General Non-cooperative agents [189] for economical and societal settings, or competitive agents for short, received increasing interest only in recent years.

Such agents have their own, possibly conflicting goals and aim for local optimization. Their owners can e.g. be competing companies or autonomous departments within a bigger organization, where the multi-agent systems should facilitate trading, allocation, or planning between these owners, e.g. by means of negotiation or auctioning.

Due to the advances in the use of Internet technology, providing technology for autonomous or competitive parties has become crucial, both for computer science and for its applications [123, 82]. For competitive agents in a multi-agent system, the question is how such a system can work properly. Here, inspired by economics, competitive games appear to be important. Several important problems have very recently been addressed.

A game is given by a set of rules regarding some players that interact with each other, and it determines who gets which payoff at the end [18, 54, 111, 118, 119]. Examples are negotiation, auctioning, formation of interaction networks between parties, production decisions in an oligopoly economy, or planning and scheduling with self-interested parties [89, 138, 4]. In this section, we focus on prominent competitive games as above (i.e., games between competitive players [4].), viz. market games, and in particular, we mainly consider various types of negotiation and auctioning.

Various forms of negotiations and auctions exist. Examples of negotiation [18, 111, 19] are one-issue negotiations and multi-issue negotiation (dealing with just one or with multiple issues, respectively); bilateral negotiations between two parties; one party that negotiates simultaneously with multiple other parties about one or more goods; etcetera. Similarly, many types of auctions exist [89], such as classical auctions like the English ascending bid auction, the Dutch clock auction, the single sealed bid second price auction (Vickrey auction); multi-issue auctions; double auctions (buyers and sellers bid simultaneously, as in many financial markets); reverse auctions (procurement auctions); combinatorial auctions (for the allocation of a collection of multiple goods) [142]; etcetera.

In these market games, participating agents have to determine several aspects. Of course, the direct values of the bid or the bidding strategy is important to be determined. Similarly, other aspects can be important to get to good bidding behavior, like models of the (changing) preferences of the opponents in the market games, the (changing) actual strategies used by the opponents, or the actual value of the good at hand (e.g. being private, common, with externalities or with complementarities).

Some of the auctions have the properties that strategic behavior by the agents is filtered out and therefore not relevant: the "truth-revealing" auctions (e.g., Vickrey auctions, and VCG auctions: Vickrey-Clark-Grooves). Most market games, however, allow strategic behavior by agents to influence the outcomes of these games. Also, the bidding process in "truth-revealing auctions" becomes strategy-dependent as well at the moment that these auctions appear in a repeated or concurrent fashion (e.g. [16, 43]). An example is formed by simulta-

[4] We thus do not only address with "competitive games" the special constant-sum game, but the more general class of games played by competitive agents

neous auctions on the Internet, all dealing with similar goods, where an agent just needs to acquire one good. Therefore, strategy determination is important for agents playing in these multiple market games.

Thus, strategies, information and knowledge related to the market games are needed for individual agents. These are studied in fields like game theory and micro economics [18, 54, 111, 102]. Although recent game theory gives valuable insights, its settings and results are often highly stylized, and not applicable in or powerful enough for multi-agent systems [82, 44].[5]

Relations to other disciplines Related scientific disciplines are (evolutionary) game theory and economics. For competitive game settings with the above described characteristics, strategies and relevant knowledge for competitive software agents are not readily available from these disciplines, as already indicated above. In general, these disciplines address such settings at a higher abstraction level, while not taking into account the actual computational tractability of and learnability of strategies and related parameters. Issues address especially how and which equilibria can be reached or obtained [144, 53], for more idealized and abstract settings of learning in repeated games [53, 187] (with e.g. the usage of mathematical Bayesian rules [6] or coordinated learning in stylized games). This does not concern computational efficiency (or tractability) considerations, but rather whether strategies are computable (i.e., on Turing machines) [53, 112, 49]. Some of the insights, however, can be used for multi-agent systems, thus especially at a higher abstraction level or for more stylized settings, including the use of impossibility results.

Adaptive Solutions Participation of an agent in one competitive game cannot be seen in its isolation, is interdependent of e.g. the future and the past, and of e.g. slowly unraveling information about e.g. allocations, interdependencies, and (private) valuations. The strategy of an agent should thus be adaptive. This is also due to the limited capabilities of agents, as is also acknowledged by modern game theory and economics, stating that agents are not fully rational:

- the players in the market games are heterogeneous agents which are boundedly rational [139, 150, 6]: diverse agents that e.g. have only partial (incomplete) information (and knowledge) and limited computing power. [7]

Thus solutions to compute adaptive strategies are needed [82, 178]: adaptive solutions, which build on experience, and which determine, adapt and learn strategies and related models and knowledge. Adaptive solutions determine the strategies via appropriate models, that contain the strategy variables as well as other appropriate parameters, representations, and relationships, and for which

[5] We will briefly further discuss the relevance of the areas (evolutionary) game theory and (micro-) economics later.

[6] e.g. with infinite positive priors distributions.

[7] This does of course not only affect an agent because of its own abilities, but also via the abilities of its opponent agents.

parameter settings have to be determined by intelligent computational techniques.

Since market games are more context dependent than e.g. matrix games, the issues that must be learnt can be broader than for matrix games. Actually, market games are often embedded in some sort of (application) setting, which determines some of the opponent types and preferences, or e.g. some of the repeated game settings. Depending on the closeness of the market game to an application setting, the game settings can be considered to be more fundamental, applicable or even applied.

Learning Agents Feasible adaptive techniques for agents playing in market games are e.g. fuzzy techniques, evolutionary algorithms, various (learning) heuristics, neural networks, simulated annealing, and graphical models. Combining competitive agent systems and learning techniques for market-based games is currently appearing as one of the important ways to go in the research on multi-agent systems.

Until now, several papers on adaptive strategies on single or multiple competitive games in multiagent systems have appeared. Papers mainly presenting various kinds of heuristics, with possibly fuzzy or probabilistic models, are e.g. [5, 1, 23, 44, 45, 77, 72, 100, 116, 134, 21, 152, 154, 56, 180, 57]. Results with fully learning approaches as well as a focus on multiple competitive games have been rather limited until now. We will give some representative references in the sequel.

Typically, learning should be done in some kind of "multiple" settings. I.e., learning can be done in the "classical" way of repeated one-shot games, or one game with one opponent during the stepwise progress of the game. However, due to the tight connections with the economic and social application fields, learning can and should also be done in e.g. repeated interrelated games or concurrent games, learning while playing against e.g. multiple opponents, or about multiple goods. We will encounter instances in the sequel.

Opponent Modeling In several settings, opponent modeling can be of importance in order to derive good game outcomes [45, 154, 155, 88, 91, 137, 136]. In such models, (approximations of) preferences of opponents are represented, which can form the base for the actual agent strategy. This is especially important, when trade-offs between game outcomes between the different players can be made and some kind of Pareto-efficiency is involved, e.g. like in multi-issue negotiation. Opponent models can be determined for one opponent or for a class (type) of opponents. In the latter case, a distinction can be made between starting with a pre-existing opponent model (offline modeling) vs. starting from scratch and learning opponent models while repeatedly playing games (online modeling). Learning techniques that have been applied are e.g. simulated annealing [88], probabilistic approaches [154, 155], graphical models [137, 136], neural networks and evolutionary algorithms [21, 91]. Alternatively, opponent modeling papers exist for e.g. combinatorial auctions, in order to reduce the search space for the auctioneer (e.g. [78]).

In a related but different way, preference elicitation is an important issue. In this case, the modeling of some human (or agent) is done in a cooperative way, in order to get the preferences into an appropriate model: a user preference model for market games. This model can then be used in an agent when playing in market games, on behalf of that person. So, in this case, the agent is instructed which goals to reach, by means of the preference model. The learning process differs in that it is supposed to be carried out with a cooperating, willing "opponent": the human being. Several papers with different objectives and learning techniques exist in this area. Learning techniques include neural networks, evolutionary algorithms, and heuristics to obtain fuzzy constraints [99, 20, 66]. This area of research is close to the more general area of preference elicitation and knowledge acquisition from humans [74], but has a different objectives in that it concerns decision making during negotiations.

Market and Strategy Modeling In other settings, models of opponent preferences are less relevant, and parameters concerning the goods about which the market game is played, or the aggregate (anonymous) market behavior (determined by a substantial amount of fairly anonymous agents) is of more importance. In case of the underlying good, one may think of a good of which its valuation can be determined from participation in multiple games. E.g., the actual value for a seller of a customer click on a web advertisement can usually not be determined beforehand. This can be learning by approaches with e.g. neural networks or evolutionary algorithms [21]. Also, the valuation of a good can depend on the allocation of other goods to (other) agents [21, 165], leading to allocative interdependencies. The level of adaptivity of the involved agents can also influence the respective individual payoffs [165]. Similarly, aggregate market behavior is of importance. This means so much as e.g: what is the typical winning price for certain types of goods in certain types of markets [193, 21], which can be done by various learning techniques. We also refer to the trading agent competition (TAC) below. Some typical settings and results exist e.g. for multiple games with an aggregated stochastic approach [1, 23], for multiple goods in repeated auctions with bounded budgets [180], one-to-many negotiations [57, 116], concurrent games with price prediction [120] or valuation estimation [130], or using evolutionary or fuzzy neural techniques for one or multiple goods in overlapping auctions [5, 71]. Also, more specific and tailored models can be designed for market (price) prediction, e.g. for financial markets. This, however, quickly reaches an other discipline, viz., regression and prediction methods, especially if these markets are complex; this is outside the scope of this paper. Finally, market behavior determined by bidding agents can also be studied by simulations, in the form of evolutionary algorithms standing for populations of agents strategies, from which also proper bidding strategies can be obtained [58, 5, 31, 55].

In case of market games on complex goods, strategies could be decomposed in some substrategies, that deal with different aspects of or paradigms in the game. E.g., a negotiation strategy can be decomposed into the concession strategy (how

much to concede in the overall value of a bid) and the Pareto-search strategy (search for Pareto-optimal deals) [152, 153, 45]. The concession strategy could be seen as market and strategy modeling, the Pareto-search strategy could be seen as opponent modeling.

Models of Application Settings Application settings and models that go further than the conventional game theoretic stylizations are important for this field. Market games are often studied related to more specific application models. We briefly mention some settings and models, e.g.

- The trading agent competitions: TAC [167, 194, 191] and TAC SCM (supply chain management) [167]. Both competitions deal with a modeled application settings, viz., a) travel agencies that have to buy and sell holiday trips consisting of complementary or substitutable constituents, and b) a 2-phase supply chain for computer manufacturing and sales, respectively. Both deal with several types of market mechanism for the distribution of goods and services with complementarities and substitutables, and the agents have to design strategies for both bidding in multiple games and determination of what to buy. Approaches that have been presented until now, are e.g. price prediction, equilibrium analysis, decision theory, and some forms of machine learning (like Reinforcement Learning) (e.g. [35, 159, 70, 61] or [193] for a survey).
- Market-based scheduling,resource allocation, and logistics [36, 190, 101, 134, 164].
- Information goods with negotiation and (dynamic) pricing [28, 86, 87, 156, 152, 57, 153, 63, 149].

Co-learning and Evaluation: State and Open Issues In addition to the development of adaptive systems for agents in market games, other aspects become important as well.

When applying adaptive techniques for competitive agents in multiagent systems, the quality of an adaptive strategy for an agent depends on the (adaptive) strategies of other agents. In the case that all agents use truly adaptive strategies as well, various forms of colearning occurs. Up to now, such environments of multiple agents are still rather restricted and mainly address learning in cooperative systems and e.g. stochastic (general-sum) games [188, 147]. Approaches and requirements address e.g. various settings with stationary opponent agents and best response, evolutionary simulations [171, 58, 5, 31, 55, 3, 46, 64], self-play (many results [37], also for co-evolution, e.g. [177]), or, for market settings, leveled learning and opponent modeling [181, 77], adaptivity and individual profits [165], and some mixed approaches (e.g. [184]). Thus, learning in a dynamic environment containing colearning competitive agents has still received limited attention, and still substantial questions exist about what feasible and relevant environments are [148, 25, 188, 37, 158, 184, 143, 47]. Environments of more or less arbitrary opponent agents are not possible in general (i.e., several impossibility results exist [113, 112]). Therefore, appropriate classes of competitive

opponent agents have to be given for which best learning strategies must be determined [148] (the "AI agenda"), while other robust evaluation criteria for resulting strategies must be determined and satisfied (e.g. [25, 37]). Still, appropriate further insight needs to be acquired for the effects of co-learning and for the way in which adaptive strategies can be evaluated.

4.3 MARL

In this section we give an overview of the state of art in multi-agent RL (MARL). This section is strongly inspired by the recent work of [148], [128], and [129]. These papers discuss current state of the art MARL algorithms and introduce new evaluation criteria (i.e. the AI agenda) for judging MARL algorithms. We also refer the interested reader to [84], [95], and [13] for alternative overview papers. We discuss a number of notable MARL algorithms along with novel evaluation criteria for competitive multi-agent RL learning. We have a bit of a chicken and the egg scenario as novel criteria are under development and are supported by novel learning algorithms that, of course, perform extremely well for the newly introduced norms. We first discuss the novel criteria, and then separately discuss the remaining algorithms. The next section begins with the basic concepts of Multi-Agent RL and the often chosen problem domain of matrix games.

Competitive Agents and Reinforcement Learning for Matrix games In this section we introduce some concepts from Reinforcement Learning. We repeat concepts from Game Theory in Section 3 and cast these to the the MARL perspective for the sake of reference.

In general, let S denote the set of states in the game and let A_i denote the set of actions that agent/player i may select in each state $s \in S$. Let $a = (a_1, a_2, \ldots, a_n)$, where $a_i \in A_i$ be a join action for n agents, and let $A = A_1 \times \cdots \times A_n$ be the set of possible joint actions. **Zero-sum games** are games where the rewards of the agents for each joint action sum to zero. **General sum games** allow for any sum of values for the reward of a joint action.

A **strategy (or policy)** for agent i is a probability distribution $\pi(\cdot)$ over its actions set A_i. Let $\pi(S)$ denote a strategy over all states $s \in S$ and let $\pi(s)$ (or π_i) denote a strategy in a single state s. A strategy may be a **pure strategy** (an agent selects an action deterministically) or according to a **mixed strategy** (a strategy that plays a random action, according a probability distribution). A **joint strategy** played by n agents is denoted by $\pi = (\pi_i, \ldots, \pi_n)$. Also, let a_{-i} and π_{-i} refer to the joint action and strategy of all agents except agent i.

We focus on the more restricted **matrix game**, defined by a set of matrices $R = \{R_1, \ldots, R_n\}$. Matrix games are the chosen domain for most recent MARL applications. We further restrict our presentation to two-player, two-action games as these are well classified [132] and often used. The algorithms presented in the rest of the paper are of course applicable to more general settings.

Let $R(\pi) = (R_1(\pi), \ldots, R(\pi_n))$ be a vector of expected payoffs when the joint strategy π is played. Also, let $R_i(\pi_i, \pi_{-i})$ be the expected payoff to agent i when it plays strategy π_i and the other agents play π_{-i}. A strategy then is **dominant** if, regardless of what any other players do, the strategy earns a player a larger payoff than any other strategy. Let $R_i(\begin{bmatrix} a_i \\ a_{-i} \end{bmatrix})$ be the payoff for agent i playing action a_i while the other agents play action a_{-i}. A strategy π_i is dominant, if and only if

$$\forall \pi_i' \forall \pi_{-i} \sum_{a_i, a_{-i}} \pi_i(a_i)\pi_{-i}(a_{-i}) R_i(\begin{bmatrix} a_i \\ a_{-i} \end{bmatrix}) >= \sum_{a_i, a_{-i}} \pi_i'(a_i)\pi_{-i}(a_{-i}) R_i(\begin{bmatrix} a_i \\ a_{-i} \end{bmatrix}) \quad (1)$$

Each individual matrix game has certain classic game theoretic values. The **minimax** value for player i is $m_i = \max_{\pi_i} \min_{a_{-i}} R_i(\pi_i, a_i)$, i.e. the least reward that can be achieved if the game is known and the game is only played once. A **Best-Response** (BR) to the opponents strategy π_{-i} is defined by

$$BR = \pi^* = max_\pi R_i(\pi, \pi_{-i}). \quad (2)$$

This is the most expected reward that can be gained playing assuming the game is known, the game is only played once, and the opponent strategy is known.

A **Nash Equilibrium** (Nash-Equilibrium) is then a joint strategy such that no agent may unilaterally change its strategy without lowering its expected payoff in the one shot play of the game. Nash [115] showed that every n player matrix game has at least one such Nash-Equilibrium. A **Pareto optimal** solution of the game is a joint strategy such that no agent may unilaterally increase its expected payoff without making another agent worse off. A joint strategy π_1 is said to **Pareto dominate** a strategy π_2 if the expected payoff for π_1 is at least as high as for π_2 and higher for at least one of the agents. A joint strategy is **Pareto deficient** if it is not Pareto optimal.

We assume that an agent can observe its own payoffs as well as the actions taken by all agents in each stage game, but only after the fact. All agents concurrently choose their actions. A possible adaption of the policy of the agents, i.e. learning as a result of observed opponent behavior, only takes effect in the next stage game. Each agent aims to maximize its reward for iterated play of the same matrix game, playing the same opponent.

Evaluation Criteria

General Background Classic Reinforcement Learning [186, 163] aims to converge to stationary policy π for an individual agent that maximizes the expected discounted future payoffs. This amounts to

$$\max_\pi E(\sum_{\tau=t}^{T} \gamma^{\tau-t} R^\tau(\pi)) \quad (3)$$

where T may be finite or infinite and $0 < \gamma < 1$ is the discount factor. An alternative measure is the average reward over the last t epochs. Both approaches however implicitly assume the agent is optimizing relative to a stationary environment, an assumption that in general does not hold for MARL. All current MARL algorithms therefore incorporate some modeling of the opponent in some form or other to include the opponent as part of the (changing) environment against which an agent is optimizing.

It should be noted that [113] prove that in general it is impossible to perfectly learn to play optimally against an adaptive opponent and at the same time perfectly estimate the policy of this opponent. Whether this theoretical result is relevant for specific games must be kept in mind. To complicate matters, [179] analyzes from an information theoretical perspective how much an agent can hinder an opponent in modeling by displaying limited random behavior, purely to hide its real preferences. Such strategic behavior is an example of how complex interactions between agents can be and how difficult it can be to learn a good policy when using an opponent model.

[34] introduces a first general classification of competence of MARL algorithms. The ranking of algorithms is based on the crossproduct of their possible strategies and their possible beliefs about the opponent's strategy. An agent's possible strategy can be classified based upon the amount of history it has in its memory. An agents beliefs mirrors the strategy classification. The different categories are supposed to be leagues of players. A fair opponent is any opponent from the same league or less. The idea is that a new learning algorithm should ideally be able to beat any fair opponent. [11] add to this classification scheme with new criterion of reactivity (see later in this section).

The focus to date in MARL algorithms has been mainly on game theoretical equilibriums from single shot games, i.e Nash-Equilibrium, Pareto-Optimal, minimax, etc .... Best-Response and Nash-Equilibrium are intertwined through the circular argument that if both players play BR players will arrive at a mirrored minimax outcome, a Nash-Equilibrium. This is the heart of **Fictitious play**; see [29], and [53].

A recent critique against the focus on such equilibriums has been launched by [128]. This work lists some well-known problems. Nash-Equilibrium are for example known not to be appropriate in repeated games, see also the Folk Theorem. The work of [96] shows how to construct equilibriums for players that are interested in average payoffs for repeated games in polynomial time. It is however unknown how the players should learn these during play as they discover the structure of the game, and the play of their opponents. Also problematical is the existence of multiple Nash-Equilibrium; how do the players choose to which they should converge if the criteria is convergence to a, not the, Nash-Equilibrium. Lastly, one-sided converge to a Nash-Equilibrium by one of the players may make it miss out on exploitation opportunities if the opponents do not follow suit. Algorithms aiming at a Nash-Equilibrium typically achieve this by updating their policy towards the BR with respect to the current policies of their opponents. Players will then, if all follow similar strategies, arrive at individual minimax

values of the game, which is a Nash-Equilibrium. Such properties have been proved for converge to Nash-Equilibrium in self-play in zero sum games [92], but have proved less tractable for general sum games. [128] provide suggestions for different criteria for evaluating MARL algorithms. Their main focus is on the **AI Agenda**.

The AI Agenda poses as evaluation criteria as to how well a given algorithm can perform against a restricted class of opponents. The general properties of an algorithm against any opponent, including game theoretical convergence properties, are deemed less important than the performance results when competing with the opponents that the agent will actually encounter. Maximizing personal reward is the criteria that we also feel should not be forgotten in the storm of newly presented evaluation criteria that MARL algorithms are ranked by. In the end, the only criteria of interest to a purely competitive agent for evaluating its learning algorithm in a specific game is how closely it approaches the highest aggregated reward possible during play for given opponents. Game theoretical notions should however not be ignored as they give a sense of how general the power of a MARL is. The AI agenda however allows for a lively competition possibility by introducing open competition on the extensive list of games generated, for example, in the GAMUT framework [117].

Other Criteria In the rest of this section we list several miscellaneous evaluation criteria that can play a role in ranking MARL algorithms.

The criterion of **asymptotic stability** was developed in [51]. This provides local dynamic robustness. Two conditions must me met: i) Any solution that is sufficiently close to the equilibrium remains arbitrarily close to it. This condition is called **Liapunov stability**. ii) Any solution that starts close enough to the equilibrium, converges to the equilibrium. These type of criteria recur in several papers as full convergence is a strong concept, but algorithms can be shown to come "close enough" to an equilibrium outcome and stay there.

An example of the above is that Hyper-Q [170]. This algorithm learns the value of joint mixed strategies,instead of joint base actions. In the rock-paper-scissors, the well-known children's game, in self play does not converge to the one third all equilibrium but cycles amongst a small number of grid points, with roughly zero average reward for both players. Quoted: "Conceivably, Hyper-Q could have converged to a cyclic Nash-Equilibrium, which would certainly be a nice outcome of self-play learning in a repeated game." This is an example where the learning algorithm achieves the same average reward as the Nash-Equilibrium, but in a dynamic setting. Note that both outcomes are as desirable from the AI agenda perspective.

Another example is the Extended Replicator Dynamics algorithm of [174]. Here the authors take a dynamical systems approach in which they first design the stable differential equations, reaching an asymptotic stable Nash equilibrium in all types of stateless matrix games. After this they constructed the approximating learning algorithm showing the same behavior as the pre-defined dynamical system, i.e. reaching a stable Nash equilibrium.

[37] introduces the AWESOME algorithm, short for "Adapt When Everybody is Stationary. Otherwise Move to Equilibrium". This algorithm converges to BR against stationary opponents, and otherwise converges to a precomputed Nash-Equilibrium in self play. These two properties are listed as minimal conditions for MARL algorithms.

[24] use the **No-regret-measure** with their GIGA-WOLF algorithm. Regret measures how much worse an algorithm performs compared to the best static strategy, with the goal to guarantee at least zero average regret, i.e. no-regret, in the limit. This is general compares the performance of a learner to the best possible hand-coded opponent that performs the best possible strategy, assuming this is computable, for a given game.

[11] define **Reactivity** that measures how fast a learner can adapt to an unexpected hypothetical change in an opponents policy; how fast can an agent learn a best response to an unexpected worst case switch in the opponent's policy. They show that it approximately predicts the performance of a learner as a function of the parameters of its learning algorithm in the matching pennies game. The criterion of reactivity is special to the MARL domain as it is a measure of how quickly an agent can react to being exploited. This is a relative non-issue in single agent RL and in Game Theory concerned with single stage games, but becomes an important factor in repeated play.

[62] introduce the notion of **Correlated Equilibrium**. Players maintain beliefs about their opponents. They are converged in a Correlated Equilibrium if both believe, based on their beliefs about their opponents, no longer see it as advantageous to adjust their policies.

Lastly, [126] presents and analysis a mathematical model of cuckoo parasitism. This work is of relevance to the MARL as it presents and in depth analysis of the cost of defense mechanisms. The main conclusion of the work is that every defense mechanism has a non-zero cost, and expending time and energy in defending against difficult and unlikely scenarios is not biologically smart. Likewise, an agent in a complex situation with limited computational resources may have to choose to focus on likely opponent strategic behavior, and not cover all bases.

The next section discusses state of the art MARL algorithms not listed above.

Other seminal work **Universal Consistency** is a strong concept from game theory. An algorithm with this property approximates the best-response stationary policy against any opponent. [52] and [50] independently show that a multiplicative-weight algorithm exhibits universal consistency. These algorithms however require the strong assumption that an agent know the opponent's policy at each time period, which is intractable in practice.

Nash-Q [76] for general-sum games, has as goal to converge to Nash-Equilibrium. This is accomplished for a limited class of games. Friend or Foe [93] treats other agents as either friend or foe and converges to Nash-Equilibrium with less restrictions than Nash-Q,

The following two papers are well known gradient-ascent type algorithms. The Policy Hill CLimber (PHC) is illustrated in [27]. PHC is a simple adaptive strategy based on its own actions and rewards. It maintains a Q-table of of values for each of its base actions, and at every time step it adjusts its mixed strategy by a small step towards the greedy policy of its current Q-function. In Infinitesimal Gradient Ascent (IGA) [151], an agent uses knowledge of the current strategy pair to make to make a small change in the direction of the gradient of its immediate payoff.

WOLF- Win or Learn Fast by [27, 24] deserves a special mention as it is one of the few, if not the first, MARL algorithm to update its learning parameters with as goal to exploit the opponent. The learning rate is made large if WOLF is losing. Otherwise, the learning rate is kept small as a good strategy has been found. Note that in [34] WOLF is exploited by a bluff and dash hand-tailored algorithm to exploit the small step increment of the latter algorithm.

The leader strategies: Bully and Godfather are introduced in [94]. These two strategies aim to threaten the opponent to play good equilibrium strategies, at least from the viewpoint of the threatening agent. This work shows that many known algorithms, like the gradient descent type, are vulnerable to exploitation by these type of hand-tailored strategies.

Predictive state representations [196] is a recent and growing new line of research. The optimization problem of an individual agent is handled by predicting future states from past observations. This is a step beyond the optimization of a policy by incorporating a link between past and future observations in the decisions on how to update the current policy.

Lastly, we list [188] with the NSCP-learners (Non-Stationary Converging Policies) for n-player general sum stochastic games. This work, as claimed, has a first proof of Convergence in self-play on general sum games. This is achieved by slowly decreasing the area of the state space in which the adaptive policies can "move". This locks in the agents to stationary, possibly mixed, strategies that are, by definition, converged.

More and more complex nested opponent models [77] will probably be the future norm in the MARL agents arms race. Although learning about an opponent while at the same time learning is problematic [113], there is still a need to be "smarter" than your opponents.

5 Contributions of this book

The previous two sections gave a comprehensive overview of the state-of-the-art research on MASs. This section discusses new contributions of the LAMAS workshop. This event included two prestigious invited talks, which have resulted in two extensive high quality papers included in this book.

The invited talk of Peter Stone, University of Texas at Austin, USA, has been shaped into the paper: Multi-Robot Learning for Continuous Area Sweeping, by Mazda Ahmadi and Peter Stone. In their paper they study the problem of multi-agent continuous area sweeping. In this problem agents are situated in a

particular environment in which they have to repeatedly visit every part of it such that they can detect events of interest for their global task and coordinate to minimize the total cost. Events are not uniformly distributed, such that agents need to visit locations non-uniformly. The authors formalize this problem and present an initial algorithm to solve it. Moreover they nicely illustrate their approach with a set of experiments in a routine surveillance task.

The second invited talk of the workshop was by Ann Nowé, professor in computer sciences at the university of Brussels, Belgium, resulting in the paper: Learning Automata as a Basis for Multiagent Reinforcement Learning, by Ann Nowé, Katja Verbeeck and Maarten Peeters. In their work they start with an overview on important theoretical results from the theory of Learning Automata in terms of game theoretic concepts and consider them as a policy iterator in the domain of Reinforcement Learning problems. Doing so they gradually move from the variable structure automaton, mapping to the single stage-single agent case, over learning automata games, mapping to the single stage multi-agent case, to interconnected Learning Automata, considering multi stage-multi agent problems. The authors also show the most interesting connection with the field of Ant Colony Optimization.

The entire program of LAMAS covered a quite wide area in learning and adaption in multi-agent systems, varying from typical application areas as traffic management, rover systems, ant systems and economical systems to more theoretical papers on state space representation, no-regret learning, evolution, exploration-exploitation and noise in cooperative systems.

Starting with the application papers, we have [182, 38, 14, 172]. In [182], the authors introduce a new kind of ant colony optimization algorithm, extending the classical algorithms with multiple types of ants. They use this kind of multi-agent approach for solving the problem of routing and backup trees in optical networks. More precisely, they assign an ant type to each working path and and backup tree.

In [38], the authors identify and explore several interesting opportunities, created by their reservation based mechanism for traffic management, for multi-agent learning. More precisely, their system consists of two kinds of agents, i.e. intersection managers and driver agents, for which they describe the learning opportunities and offer a first-cut solution to each of them. These opportunities, amongst others, include delayed response for the intersection manager, organizing an intersection as a market, agents bidding in this market and autonomous lane changing.

The topic of the paper [14] is coordination in large multi-agent systems, studying effects of guiding the decision process of individual agents. In their work they study this problem in the context of route guidance in traffic management. The guiding information can have different sources and agents are potential players. Simulations of this problem show that it can be beneficial to have a recommendation system for drivers. The authors discuss the different conditions for an optimal performing recommendation system.

Adaptive Multi-Rover Systems are the topic of paper [172]. More precisely, the authors describe how efficient reward methods can be applied to the coordination of multiple agents in a dynamic environment with limited communication possibilities. Difficulties lie in the design of the individual reward functions which need to be aligned with the global reward function and must stay aligned with changes in the reward of each individual agent. Their results show how factored reward functions, in combination with evolutionary computation, can be successful for real world applications.

One of the fundamental problems in RL is the exploration-exploitation dilemma, which is extensively studied in [135]. The authors propose a new algorithm based on meta-heuristics to tune the tradeoff between both and validate it on economic systems. Moreover it is shown to be a promising approach in comparison with other adaptive techniques.

Having a glance at the less application oriented and more theoretical papers, we find five contributions in this book [12, 2, 173, 42, 107].

In [12] the authors present a new multi-agent learning algorithm, which is a modification of the ReDVaLeR algorithm. The new algorithm achieves convergence to near-best response against eventually stationary opponents, no-regret payoff against arbitrary opponents and convergence to the Nash equilibrium in unique mixed equilibria games.

In [2] the authors extend their previous algorithm, which finds Pareto optimal solutions in general sum games, to so-called preferred Pareto Optimal solutions (PPO). A clear definition can be found in their paper. Moreover, they experiment with the opportunity of revelation in two-player two-action conflict games. Their experiments show that their new algorithm is an improvement over previous results.

In [173] the authors give a new direction to research in multi-agent learning by cross-fertilizing the multi-agent learning problem with relational reinforcement learning (RRL). More precisely, they propose to use a relational representation of the state space in multi-agent reinforcement learning as this has many proved benefits over the propositional one, as for instance handling large state spaces, a rich relational language, modeling of other agents without a computational explosion, and generalization over new derived knowledge. Their initial experiments show that the learning rates are quite good and promising when using a relational representation in coordination problems and that they can be increased by using the observations over other agents to learn a relational structure between the agents.

The authors of [42] present their methods for dealing with a noisy environment in cooperative multi-agent learning. More precisely, they introduce an algorithm to cope with perception, communication and position errors for cooperative multi-agent learning tasks. Although this offers interesting possibilities, the improvements are quite expensive seen from a computational perspective.

Tag-mediated interaction has shown to stimulate cooperation in populations of agent playing the Prisoner's Dilemma (PD) game. In [107], the authors try to answer why tags facilitate such cooperation. More precisely, they analyzed

the effects of the size of the tag space, mutation rate in the population, on cooperation in a population of agents playing the PD game. Additionally, they empirically analyzed why tags have this influence on this type of systems. The conclusion suggests that tags rather promote mimicry than cooperation.

6 Open Research Issues

Multi-agent learning is a relatively young field and as such its open research issues are still very much in flux. This section singles-out three important open questions that need to be addressed in order to make multi-agent learning more broadly successful as a technique in real world applications. These issues arise from the *multi* in multi-agent learning, and may eventually require new learning methods specifically tailored for multiple agents.

Scalability Scalability is a problem for many learning techniques, but especially so for multi-agent learning. The dimensionality of the search space grows rapidly with the number of agents, the complexity of their behaviors, and the size of the network of interactions among them. This search space grows so rapidly that one *cannot* learn the entire joint behavior of a large, heterogeneous, strongly intercommunicating multi-agent system. Effective learning in an area this complex requires some degree of sacrifice: either by isolating the learned behaviors among individual agents, by reducing the heterogeneity of the agents, or by reducing the complexity of the agent's capabilities. Techniques such as learning hybrid teams, decomposition, or partially restricting the locality of reinforcement provide promising solutions in this direction.

As problem complexity increases, it gives rise to the spectre of *emergent behavior*, where the global effects of simple agent behaviors cannot be readily predicted. This is an area of considerable study and excitement in artificial life: but it may also be a major problem for machine learning. How does emergence affect the smoothness of the search space? If small perturbations in agent behavior result in radical swings in emergent behavior, can learning methods be expected to scale well at all in this environment?

Adaptive Dynamics and Nash Equilibria Multi-agent systems are typically dynamic environments, with multiple learning agents vying for resources and tasks. This dynamism presents a unique challenge not normally found in single-agent learning: as the agents learn, their adaptation to one another changes the world scenario. How do agents learn in an environment where the goalposts are constantly and adaptively being moved? In many cases, existing learning methods may converge to suboptimal Nash equilibria. We echo opinions from [90] and express our concern with the use of Nash equilibria in cooperative multi-agent learning: such "rational" convergence to equilibria may well be movement away from globally *team-optimal* solutions [90]. We argue that, in the context of cooperative agents, the requirement of rationality should be secondary to that of optimal team behavior. Mutual trust may be a more useful concept in this context.

Large State Spaces The state space of a large, joint multi-agent task can be overwhelming. An obvious way to tackle this is to use domain knowledge to simplify the state space, often by providing a smaller set of more "powerful" actions customized for the problem domain. For example, agents may use higher-level descriptions of states and actions [104]. Another alternative has been to reduce complexity by heuristically decomposing the problem, and hence the *joint behavior*, into separate, simpler behaviors for the agents to learn. One approach to such decomposition is to learn basic behaviors first, then set them in stone and learn more complex behaviors based on them. This method is commonly known as *layered learning*, and was successfully applied to robotic soccer [157]. Another approach, *shaping*, gradually changes the reward function from favoring easier behaviors to favoring more complex ones based on those easy behaviors [103, 10].

Less work has been done on formal methods of decomposing tasks (and behaviors) into subtasks (sub-behaviors) appropriate for multi-agent solutions, how agents' sub-behaviors interact, and how and when learning of these sub-behaviors may be parallelized. Guestrin et al note that in many domains the actions of some agents may be independent [65]. Taking advantage of this, they suggest partially decomposing the joint team behavior based on a *coordination graph* that heuristically spells out which agents must interact in order to solve the problem. Ghavamzadeh and Mahadevan suggest a different hierarchical approach to simplifying the inter-agent coordination task, where agents coordinate their high-level behaviors, rather than each primitive action they may perform [59].

An alternative to problem decomposition, is the quest for other representations or formalisms for the state space. One such succesfull method in single-agent learning has been the cross fertilization between reinforcement learning and inductive logic programming [41, 39, 168]. More precisely, in this formalism states are represented in a relational form, that more directly represents the underlying world. Complex tasks as planning or information retrieval on the web can be represented more naturally in relational form than in propositional form, what is usually done in Reinforcement Learning. In [173], the authors are extending this single agent work to multi-agent planning and coordination tasks.

Competitive Agents Non-cooperative agents [189] for economical and societal settings, or competitive agents for short, are receiving increasing interest in recent years. Such agents have their own, possibly conflicting goals and aim for local optimization. Their owners can e.g. be competing companies or autonomous departments within a bigger organization, where the multi-agent systems should facilitate trading, allocation, or planning between these owners, e.g. by means of negotiation or auctioning.

Due to the advances in the use of Internet technology, providing technology for autonomous or competitive parties has become crucial, both for computer science and for its applications [123, 82]. For competitive agents in a multi-agent system, the continuing question is how such a system can work properly. Here, inspired by economics, competitive games appear to be important.

More and more complex nested opponent models [77] will likely be the future norm in the for agents in the competitive arms race. Although learning about an opponent while at the same time learning is problematic [113], there is still a need to be "smarter" than your opponents. The AI Agenda will play an important role.

A Introductory Notions from (Evolutionary) Game Theory

In this section, as an Appendix, we introduce elementary concepts from Game Theory (GT) and Evolutionary Game Theory (EGT) necessary to understand Sections 3 and 4 of this paper. Game Theory is an economical theory that models interactions between agents as games of two or more players. More precisely, the agents participating in such a game can choose from a set of strategies to play, according to their own preferences. Game Theory is the mathematical study of interactive decision making in the sense that the agents involved in the decisions take into account their own choices and those of others. Choices are determined by stable preferences concerning the outcomes of their possible decisions, and by the relation between their own choices and those of others.

After the stagnation of GT for many years, John Maynard Smith applied Game Theory to Biology, which made him relax the strong premises behind GT. Under these biological circumstances, it becomes impossible to judge what choices are the most rational ones. The question now becomes how a player can learn to optimize its behavior and maximize its return. This learning process is analogous to the concept of evolution in Biology. These new ideas have led to the development of the concept of Evolutionary Stable Strategies (ESS), a special case of the Nash condition. In contrast to GT, EGT is descriptive and starts from more realistic views of the game and its players. Here the game is no longer played exactly once by rational players who know all the details of the game. Details of the game include each others preferences over outcomes. Instead EGT assumes that the game is played repeatedly by players randomly drawn from large populations, uninformed of the preferences of the opponent players.

We provide definitions of strategic games, as well zero sum as general sum, and introduce concepts as Nash equilibrium, Pareto optimality, Pareto Dominance, Evolutionary Stable Strategies and Population Dynamics. For the connection between these concepts we refer the interested reader to [175, 133, 187].

A.1 Strategic games

In this section we define n-player normal form games as a conflict situation involving gains and losses between n players. In such a game n players repeatedly interact with each other by all choosing an action (or strategy) to play. All players choose their strategy at the same time. For reasons of simplicity, we limit the pure strategy set of the players to 2 strategies. A strategy is defined as a probability distribution over all possible actions. In the 2-pure strategies

case, we have: $s_1 = (1, 0)$ and $s_2 = (0, 1)$. A mixed strategy s_m is then defined by $s_m = (x_1, x_2)$ with $x_1, x_2 \neq 0$ and $x_1 + x_2 = 1$.

Defining a game more formally we restrict ourselves to the 2-player 2-action game. Nevertheless, an extension to n-players n-actions games is straightforward, but examples in the n-player case do not show the same illustrative strength as in the 2-player case. A game $G = (S_1, S_2, P_1, P_2)$ is defined by the payoff functions P_1, P_2 and their strategy sets S_1 for the first player and S_2 for the second player. In the 2-player 2-strategies case, the payoff functions $P_1 : S_1 \times S_2 \rightarrow \Re$ and $P_2 : S_1 \times S_2 \rightarrow \Re$ are defined by the payoff matrices, A for the first player and B for the second player, see Table 8. The payoff tables A, B define the instantaneous rewards. Element a_{ij} is the reward the row-player (player 1) receives for choosing pure strategy s_i from set S_1 when the column-player (player 2) chooses the pure strategy s_j from set S_2. Element b_{ij} is the reward for the column-player for choosing the pure strategy s_j from set S_2 when the row-player chooses pure strategy s_i from set S_1.

If now $a_{ij} + b_{ij} = 0$ for all i and j, we call the game a *zero sum game*. This means that the sum of what is won by one agent (positive) and lost by another (negative) equals zero. This corresponds to a situation of *pure competition*. In case that $a_{ij} + b_{ij} \neq 0$ for all i and j we call the game a *general sum game*. In this situation it might be very beneficial for the different agents to cooperate with one another.

The family of 2×2 games is usually classified in three subclasses, as follows [133],

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

Table 8. The left matrix (A) defines the payoff for the row player, the right matrix (B) defines the payoff for the column player

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

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

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

The first subclass includes those type of games where each player has a dominant strategy[8], as for instance the prisoner's dilemma. However it includes a

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

larger collection of games since only one of the players needs to have a dominant strategy. In the second subclass none of the players has a dominated strategy (e.g. battle of the sexes). But both players receive the highest payoff by both playing their first or second strategy. This is expressed in the condition $(a_{11} - a_{21})(b_{11} - b_{12}) > 0$. The third subclass only differs from the second in the fact that the players do not receive their highest payoff by both playing the first or the second strategy (e.g. matching pennies game). This is expressed by the condition $(a_{11} - a_{21})(b_{11} - b_{12}) < 0$.

A.2 Nash equilibrium

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

Formally, a Nash equilibrium is defined as follows. When 2 players play the strategy profile $s = (s_i, s_j)$ belonging to the product set $S_1 \times S_2$ then s is a Nash equilibrium if $P_1(s_i, s_j) \geq P_1(s_x, s_j)\ \forall x \in \{1, ..., n\}$ and $P_2(s_i, s_j) \geq P_2(s_i, s_x)$ $\forall x \in \{1, ..., m\}$ [9].

A.3 Minimax and Maximin

In the context of zero-sum games two specific value are of particular interest, i.e. *minimax* and *maximin*. More precisely, recall from Section A.1 that in case of zero-sum games we have, $a_{ij} + b_{ij} = 0$ or $a_{ij} = -b_{ij}$. Player one will try to maximize this value and player two will try to minimize it. Intuitively, *maximin* is the maximum payoff that player one will receive if player two responds optimally to every strategy of player one by minimizing one's payoff. Formally, we have

$$maximin = \max_{s_i \in S_1} \min_{s_j \in S_2} P(s_i, s_j) \tag{4}$$

$$\max_{s_i \in S_1} \min_{s_j \in S_2} s_i A s_j^T \tag{5}$$

Note that s_i and s_j need to be interpreted as probability distributions with $s_i = (x_1, x_2)$ where $x_1, x_2 \geq 0$ and $x_1 + x_2 = 1$.

Analogously, *minimax* is defined as follows for the second player,

$$minimax = \min_{s_j \in S_2} \max_{s_i \in S_1} s_i A s_j^T \tag{6}$$

[9] For a definition in terms of best reply or best response functions we refer the reader to [187]

Von Neumann proved that for any zero sum game there exists a $v \in R$ such that $minimax = maximin = v$. This means that for any 2-player finite zero sum game *maximin* and *minimax* always coincide. Moreover, for every Nash equilibrium $(s_i^*, s_j^*))$ holds: $s_i^* A s_j^* = v$. The interested reader can find the proofs in [133].

A.4 Pareto optimality

The concept of Pareto optimality is named after the Italian economist Vilfredo Pareto(1848-1923). Intuitively a Pareto optimal solution of a game can be defined as follows: a combination of actions of agents in a game is Pareto optimal if there is no other solution for which all players do at least as well and at least one agent is strictly better off.

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

Another related concept is that of Pareto Dominance: An outcome of a game is Pareto dominated if some other outcome would make at least one player better off without hurting any other player. That is, some other outcome is weakly preferred by all players and strictly preferred by at least one player. If an outcome is not Pareto dominated by any other, than it is Pareto optimal.

A.5 Evolutionary Stable Strategies

The core equilibrium concept of Evolutionary Game Theory is that of an Evolutionary Stable Strategy (ESS). The idea of an evolutionarily stable strategy was introduced by John Maynard Smith and Price in 1973 [106]. Imagine a population of agents playing the same strategy. Assume that this population is invaded by a different strategy, which is initially played by a small number of the total population. If the reproductive success of the new strategy is smaller than the original one, it will not overrule the original strategy and will eventually disappear. In this case we say that the strategy is evolutionary stable against this new appearing strategy. More generally, we say a strategy is an Evolutionary Stable strategy if it is robust against evolutionary pressure from any appearing mutant strategy.

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

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

and being a mutant is,

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

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

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

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

A.6 Population Dynamics

In this section we discuss the Replicator Dynamics in a single population setting. For a discussion on the multi-population setting we refer the reader to [60, 133, 187].

The basic concepts and techniques developed in EGT were initially formulated in the context of evolutionary biology [105, 187, 141]. In this context, the strategies of all the players are genetically encoded (called genotype). Each genotype refers to a particular behavior which is used to calculate the payoff of the player. The payoff of each player's genotype is determined by the frequency of other player types in the environment.

One way in which EGT proceeds is by constructing a dynamic process in which the proportions of various strategies in a population evolve. Examining the expected value of this process gives an approximation which is called the RD. An abstraction of an evolutionary process usually combines two basic elements: **selection** and **mutation**. Selection favors some varieties over others, while mutation provides variety in the population. The replicator dynamics highlight the role of selection, it describes how systems consisting of different strategies change over time. They are formalized as a system of differential equations. Each replicator (or genotype) represents one (pure) strategy s_i. This strategy is inherited by all the offspring of the replicator. The general form of a replicator dynamic is the following:

$$\frac{dx_i}{dt} = [(A\mathbf{x})_i - \mathbf{x} \cdot A\mathbf{x}]x_i \tag{7}$$

In equation (7), x_i represents the density of strategy s_i in the population, A is the payoff matrix which describes the different payoff values each individual replicator receives when interacting with other replicators in the population. The state of the population ($\mathbf{x}$) can be described as a probability vector $\mathbf{x} = (x_1, x_2, ..., x_J)$ which expresses the different densities of all the different types of replicators in the population. Hence $(A\mathbf{x})_i$ is the payoff which replicator s_i receives in a population with state x and $\mathbf{x} \cdot A\mathbf{x}$ describes the average payoff in the population. The growth rate $\frac{\frac{dx_i}{dt}}{x_i}$ of the population share using strategy s_i equals the difference between the strategy's current payoff and the average payoff in the population. For further details we refer the reader to [187, 73].

Bibliography

[1] C. P. A. Byde and N. Jennings. Decision procedures for multiple auctions. In *Proceedings of the 1st Int. Conf. Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, 2002.

[2] S. Airiau and S. Sen. Towards a pareto-optimal solution in general-sum games, study in 2x2 games. In *LAMAS*, 2005.

[3] A. Alkemade, J. La Poutré, and H. Amman. On social learning and robust evolutionary algorithm design in economic games. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 2445–2452. IEEE Press, 2005.

[4] F. Alkemade and J. La Poutré. Heterogeneous, boundedly rational agents in the cournot duopoly. In *In: R. Cowan and N. Jonard (eds.), Heterogenous Agents, Interactions and Economic Performance, Springer Lecture Notes in Economics and Mathematical Systems (LNEMS) 521*, pages 3–17. Springer Verlag, 2002.

[5] P. Anthony and N. Jennings. Developing a bidding agent for multiple heterogeneous auctions. In *ACM Transactions on Internet Technology (ACM TOIT) 3*, pages 185–217, 2003.

[6] W. Arthur. Inductive reasoning and bounded rationality. *American Economic Review 84*, pages 406–411, 1994.

[7] R. Axelrod. *The evolution of cooperation*. Basic Books, New York, NY, 1984.

[8] T. Balch. Learning roles: Behavioral diversity in robot teams. Technical Report GIT-CC-97-12, Georgia Institute of Technology, 1997.

[9] T. Balch. *Behavioral Diversity in Learning Robot Teams*. PhD thesis, College of Computing, Georgia Institute of Technology, 1998.

[10] T. Balch. Reward and diversity in multirobot foraging. In *IJCAI-99 Workshop on Agents Learning About, From and With other Agents*, pages 92–99, 1999.

[11] B. Banerjee and J. Peng. The role of reactivity in multiagent learning. In *Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 538–545, 2004.

[12] B. Banerjee and J. Peng. Convergence of no-regret learning in multiagent systems. In *LAMAS*, 2005.

[13] A. Barto and S. Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete-Event Systems journal*, 13:41–77, 2003.

[14] A. L. C. Bazzan, M. Fehler, and F. Klugl. Learning to coordinate in a network of social drivers: the role of information. In *LAMAS*, 2005.

[15] H. Berenji and D. Vengerov. Advantages of cooperation between reinforcement learning agents in difficult stochastic problems. In *Proceedings of 9th IEEE International Conference on Fuzzy Systems*, 2000.

[16] D. Bernhardt and D. Scoones. A note on sequential auctions. *The American Economic Review*, 84(3):653–657, 1994.

[17] D. Bernstein, S. Zilberstein, and N. Immerman. The complexity of decentralized control of MDPs. In *Proceedings of UAI-2000: The Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 819–840, 2000.

[18] K. Binmore. *Fun and Games*. D.C. Heath and Company, Lexington, MA, 1992.

[19] K. Binmore and N. Vulkan. Applying game theory to automated negotiation. *Netnomics*, 1:1–9, 1999.

[20] A. Biso, F. Rossi, and A. Sperdutti. Experimental results on learning soft constraints. In *A. G. Cohn, F. Giunchiglia, and B. Selman (eds.), Proceedings of KR2000: Principles of Knowledge Representation and Reasoning*, pages 435–444, 2000.

[21] S. Bohté, E. Gerding, and J. La Poutré. Market-based recommendation: Agents that compete for consumer attention. *ACM Transactions on Internet Technology (ACM TOIT), (Special Issue on Machine Learning on the Internet)*, 4(4):420–448, 2004.

[22] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. SFI Studies in the Sciences of Complexity. Oxford University Press, 1999.

[23] C. Boutilier, M. Goldszmidt, , and B. Sabata. Sequential auctions for the allocation of resources with complementaries. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI 99)*, pages 527–534, 1999.

[24] M. Bowling. Convergence and no-regret in multiagent learning. In *Advances in Neural Information Processing Systems*, volume 17, pages 209–216, 2004.

[25] M. Bowling and M. Velose. Rational and convergent learning in stochastic games. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1021–1026, 2001.

[26] M. Bowling and M. Veloso. An analysis of stochastic game theory for multiagent reinforcement learning. Technical Report CMU-CS-00-165, Computer Science Department, Carnegie Mellon University, 2000.

[27] M. Bowling and M. Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136(2):215–250, 2002.

[28] C. Brooks, S. Fay, R. Das, J. MacKie-Mason, J. Kephart, and E. Durfee. Automated strategy searches in an electronic goods market: Learning complex price schedules. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, pages 31–41. ACM Press, 1999.

[29] G. W. Brown. Iterative solution of games by Fictitious Play, 1951. In Activity Analysis of Production and Allocation (T.C. Koopmans, Ed.), pp. 374-376, Wiley: New York.

[30] L. Bull and T. C. Fogarty. Evolving cooperative communicating classifier systems. In A. V. Sebald and L. J. Fogel, editors, *Proceedings of the Fourth Annual Conference on Evolutionary Programming (EP94)*, pages 308–315, 1994.

[31] A. Byde. Applying evolutionary game theory to auction mechanism design. In *ACM Conference on E-Commerce (ACM-EC'03)*, 2003.

[32] G. Chalkiadakis and C. Boutilier. Coordination in multiagent reinforcement learning: A Bayesian approach. In *Proceedings of The Second International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS 2003)*, pages 709–716. ACM, 2003.

[33] Y.-H. Chang, T. Ho, and L. Kaelbling. All learning is local: Multi-agent learning in global reward games. In *Proceedings of Neural Information Processing Systems (NIPS-03)*, 2003.

[34] Y.-H. Chang and L. P. Kaelbling. Playing is believing: the role of beliefs in multi-agent learning. In *Advances in Neural Information Processing Systems-(NIPS)*, volume 14, 2002.

[35] S.-F. Cheng, E. Leung, K. Lochner, K. O'Malley, D. Reeves, L. Schvartzman, and M. Wellman. Walverine: A Walrasian trading agent. *Decision Support Systems*, 39:169–184, 2005.

[36] S. Clearwater. *Market based Control of Distributed Systems*. World Scientific Press, Singapore, 1995.

[37] V. Conitzer and T. Sandholm. AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. In *20th International Conference on Machine Learning (ICML)*, pages 83–90, 2003.

[38] K. Dresner and P. Stone. Multiagent traffic management: Opportunities for multiagent learning. In *LAMAS*, 2005.

[39] K. Driessens and S. Dzeroski. Integrating guidance into relational reinforcement learning. *Machine Learning*, 57(3):271–304, Dec. 2004.

[40] E. Durfee, V. Lesser, and D. Corkill. Coherent cooperation among communicating problem solvers. *IEEE Transactions on Computers*, C-36(11):1275–1291, 1987.

[41] S. Dzeroski, L. D. Raedt, and K. Driessens. Relational reinforcement learning. *Machine Learning*, 43:7–52, 2001.

[42] C. O. e Sousa and L. Custodio. Dealing with errors in a cooperative multiagent learning system. In *LAMAS*, 2005.

[43] W. Elmaghraby. The importance of ordering in sequential auctions. *Management Science*, 49(5):673–682, 2003.

[44] P. Faratin, C. Sierra, and N. Jennings. Negotiation decision functions for autonomous agents. *International Journal of Robotics and Autonomous Systems*, 34(24):159–182, 1998.

[45] P. Faratin, C. Sierra, and N. Jennings. Using similarity criteria to make issue trade-offs. *Artificial Intelligence*, 142:205–237, 2002.

[46] S. Fatima, M. Wooldridge, and N. Jennings. Comparing equilibria for game theoretic and evolutionary bargaining models. In *Proceedings of the 5th International Workshop on Agent-Mediated Electronic Commerce (AMEC V)*, pages 70–77, 2003.

[47] S. Ficici, O. Melnik, and J. Pollack. *Selection in Coevolutionary Algorithms and the Inverse Problem*, pages 277–294. Springer, 2004.

[48] S. Ficici and J. Pollack. A game-theoretic approach to the simple coevolutionary algorithm. In *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature (PPSN VI)*. Springer Verlag, 2000.

[49] D. P. Foster and H. P. Young. On the impossibility of predicting behavior of rational agents. In *PNAS (Proceedings of the National Academy of Sciences of the USA) 98 (22)*, 2001.

[50] Y. Freund and R. E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.

[51] M. Frisch and S. Smale. *Differential Equations, Dynamical Systems and Linear Algebra*. Academic Press, Inc, 1974.

[52] D. Fudenberg and D. Levine. Consistency and cautious fictitious play. *Journal of Economic Dynamics and Control*, 19:1065–1089, 1995.

[53] D. Fudenberg and D. K. Levine. *The Theory of Learning in Games*. Cambridge, Massachusetts: MIT Press, 1999.

[54] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[55] E. Gerding and J. La Poutré. Bargaining with posterior opportunities: An evolutionary social simulation. In *M. Gallegati and A.P. Kirman and M. Marsili (eds.), The Complex Dynamics of Economic Interactions, Springer Lecture Notes in Economics and Mathematical Systems (LNEMS) 531*, pages 241–256, 2003.

[56] E. Gerding, K. Somefun, and H. La Poutré. Automated bilateral bargaining about multiple attributes in a one-to-many setting. In *Proceedings of the Sixth International Conference on Electronic Commerce (ICEC04)*, pages 105–112. ACM Press, 2004.

[57] E. Gerding, K. Somefun, and H. La Poutré. Bilateral bargaining in a one-to-many bargaining setting. In *Agent Mediated Electronic Commerce VI (AMEC-VI), Springer Lecture Notes in Artificial Intelligence (LNAI), Springer Verlag, (invited for publication)*, 2004. to appear.

[58] E. Gerding, D. van Bragt, and J. La Poutré. Multi-issue negotiation processes by evolutionary simulation: Validation and social extensions. *Computational Economics*, 22:39–63, 2003.

[59] M. Ghavamzadeh and S. Mahadevan. Learning to communicate and act using hierarchical reinforcement learning. In *AAMAS-2004 — Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems*, pages 1114–1121, 2004.

[60] C. Gintis. *Game Theory Evolving*. University Press, Princeton, 2000.

[61] A. Greenwald and J. Boyan. Bidding under uncertainty: Theory and experiments. In *Twentieth Conference on Uncertainty in Artificial Intelligence*, pages 209–216, 2004.

[62] A. Greenwald and K. Hall. Correlated Q-learning. In *Proceedings of the Twentieth International Conference on Machine Learning, ICML*, pages 242–249, 2003.

[63] A. Greenwald and J. Kephart. Shopbots and pricebots. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI 99)*, pages 506–511, 1999.

[64] J. Grefenstette and R. Daley. Methods for competitive and cooperative coevolution. In *Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium*, pages 45–50. AAAI Press., 1996. Technical Report SS-96-01.

[65] C. Guestrin, M. Lagoudakis, and R. Parr. Coordinated reinforcement learning. In *Proceedings of the 2002 AAAI Symposium Series: Collaborative Learning Agents*, pages 227 – 234, 2002.

[66] Y. Guo, J. Muller, and C. Weinhardt. Learning user preferences for multiattribute negotiation: An evolutionary approach. In *In. J. Muller, V. Marik, and M. Pechoucek (eds.), Multi-Agent Systems and Applications III, Springer Lecture Notes in Artificial Intelligence, Vol. 2691*, pages 303–313. Springer-Verlag, 2003.

[67] A. Hara and T. Nagao. Emergence of cooperative behavior using ADG; Automatically Defined Groups. In *Proceedings of the 1999 Genetic and Evolutionary Computation Conference (GECCO-99)*, pages 1038–1046, 1999.

[68] T. Haynes and S. Sen. Evolving behavioral strategies in predators and prey. In G. Weiß and S. Sen, editors, *Adaptation and Learning in Multiagent Systems*, Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin, Germany, 1995.

[69] T. D. Haynes and S. Sen. Co-adaptation in a team. *International Journal of Computational Intelligence and Organizations (IJCIO)*, 1(4), 1997.

[70] M. He and N. R. Jennings. Southampton TAC: An adaptive autonomous trading agent. *ACM Transactions on Internet Technology*, 3:218–235, 2003.

[71] M. He, N. R. Jennings, and A. Prgel-Bennett. A heuristic bidding strategy for buying multiple goods in multiple english auctions. *ACM Transactions on Internet Technology*, 2006. to appear.

[72] M. He, H. Leung, and N. R. Jennings. A fuzzy logic based bidding strategy for autonomous agents in continuous double auctions. *IEEE Trans. on Knowledge and Data Engineering*, 15:1345–1363, 2003.

[73] J. Hofbauer and K. Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998.

[74] R. Hoffman and N. Shadbolt. Eliciting knowledge from experts: A methodological analysis. *Organizational and Human Decision Process*, 62(2):129–158, 1995.

[75] B. Hölldobler and E. O. Wilson. *The Ants*. Harvard University Press, 1990.

[76] J. Hu and M. Wellman. Multiagent reinforcement learning: theoretical framework and an algorithm. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 242–250. Morgan Kaufmann, San Francisco, CA, 1998.

[77] J. Hu and M. Wellman. Online learning about other agents in a dynamic multiagent system. In K. P. Sycara and M. Wooldridge, editors, *Proceedings of the Second International Conference on Autonomous Agents (Agents'98)*, pages 239–246, New York, 1998. ACM Press.

[78] B. Hudson and T. Sandholm. Effectiveness of preference elicitation in combinatorial auctions. In *J. Padget, O. Shehory, D. Parkes, N. Sadeh, and W.E. Walsh (eds.), Agent-Mediated Electronic Commerce IV (AMEC IV): Designing Mechanisms and Systems, Springer Lecture Notes in Computer Science, Vol. 2531*, pages 69–86. Springer-Verlag, 2002.

[79] H. Iba. Evolutionary learning of communicating agents. *Information Sciences*, 108:181–206, 1998.

[80] H. Iba. Evolving multiple agents by genetic programming. In L. Spector, W. Langdon, U.-M. O'Reilly, and P. Angeline, editors, *Advances in Genetic Programming 3*, pages 447–466. The MIT Press, Cambridge, MA, 1999.

[81] T. Jansen and R. P. Wiegand. Exploring the explorative advantage of the cooperative coevolutionary (1+1) EA. In E. Cantu-Paz *et al*, editor, *Prooceedings of the Genetic and Evolutionary Computation Conference (GECCO)*. Springer-Verlag, 2003.

[82] N. Jennings, P. Faratin, A. Lomuscio, S. Parsons, C. Sierra, and M. Wooldrigde. Automated negotiation: prospects, methods, and challenges. *International Journal of Group Decision and Negotiation*, 10:199–215, 2001.

[83] K.-C. Jim and C. L. Giles. Talking helps: Evolving communicating agents for the predator-prey pursuit problem. *Artificial Life*, 6(3):237–254, 2000.

[84] L. P. Kaelbling, M. L. Littman, and A. P. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[85] S. Kapetanakis and D. Kudenko. Reinforcement learning of coordination in cooperative multi-agent systems. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI02)*, 2002.

[86] J. Kephart, C. Brooks, and R. Das. Pricing information bundles in a dynamic environment. In *Proceedings of the 3rd ACM Conference on Electronic Commerce (ACMEC)*, pages 180–190. ACM Press, 2001.

[87] J. Kephart, J. Hanson, , and A. Greenwald. Dynamic pricing by software agents. *Computer Networks*, 36(6):731–752, 2000.

[88] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12:111–125, 2003.

[89] V. Krishna. *Auction Theory*. Academic Press, 2002.

[90] M. I. Lichbach. *The cooperator's dilemma*. University of Michigan Press, 1996.

[91] R. Lin. Bilateral multi-issue contract negotiation for task redistribution using a mediation service. In *Proceedings Agent Mediated Electronic Commerce VI*, 2004. to appear.

[92] M. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, pages 157–163, New Brunswick, NJ, 1994. Morgan Kaufmann.

[93] M. Littman. Friend-or-foe Q-learning in general-sum games. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 322–328. Morgan Kaufmann Publishers Inc., 2001.

[94] M. Littman and P. Stone. Leading best-response strategies in repeated games. In *Seventeenth International Joint Conference on Artificial Intelligence (IJCAI) workshop on Economic Agents, Models, and Mechanisms*, 2001.

[95] M. L. Littman and S. M. Majercik. Large-scale planning under uncertainty: A survey. In *Workshop on Planning and Scheduling for Space*, 1997.

[96] M. L. Littman and P. Stone. A polynomial-time nash equilibrium algorithm for repeated games. In *Proceedings of the 4th ACM conference on Electronic commerce*, 2003. also appeared in Decision Support Systems,39:55–66,2005.

[97] S. Luke. Genetic programming produced competitive soccer softbot teams for RoboCup97. In J. R. Koza *et al*, editor, *Genetic Programming 1998: Proceedings of the Third Annual Conference*, pages 214–222. Morgan Kaufmann, 1998.

[98] S. Luke and L. Spector. Evolving teamwork and coordination with genetic programming. In J. R. Koza, D. E. Goldberg, D. B. Fogel, and R. L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 150–156, Stanford University, CA, USA, 28–31 1996. MIT Press.

[99] X. Luo, N. R. Jennings, and N. Shadbolt. Acquiring tradeoff preferences for automated negotiations: A case study. In *proceedings of the 5th International Workshop on Agent-Mediated Electronic Commerce (AMEC V)*, pages 37–55, 2003.

[100] X. Luo, N. R. Jennings, N. Shadbolt, H. Leung, and J. H. Lee. A fuzzy constraint based model for bilateral multi-issue negotiations in semi-competitive environments. *Artificial Intelligence Journal*, 148(1-2):53–102, 2003.

[101] J. K. MacKie-Mason, A. Osepayshvili, D. M. Reeves, and M. P. Wellman. Price prediction strategies for market-based scheduling. In *Fourteenth International Conference on Automated Planning and Scheduling*, pages 244–252, 2004.

[102] A. Mas-Collel, M. Whinston, and J. Green. *Microeconomic Theory*. Oxford University Press, 1995.

[103] M. Mataric. Reinforcement learning in the multi-robot domain. *Autonomous Robots*, 4(1):73–83, 1997.

[104] M. Mataric. Using communication to reduce locality in distributed multi-agent learning. *Joint Special Issue on Learning in Autonomous Robots, Machine Learning, 31(1-3), 141-167, and Autonomous Robots, 5(3-4), Jul/Aug 1998, 335-354*, 1998.

[105] J. Maynard-Smith. *Evolution and the Theory of Games*. Cambridge University Press, 1982.

[106] J. Maynard Smith and J. Price. The logic of animal conflict. *Nature*, 146:15–18, 1973.

[107] A. McDonald and S. Sen. The success and failure of tag-mediated evolution of cooperation. In *LAMAS*, 2005.

[108] T. Miconi. When evolving populations is better than coevolving individuals: The blind mice problem. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 647–652, 2003.

[109] M. Mitchell, J. Crutchfield, and R. Das. Evolving cellular automata with genetic algorithms: A review of recent work. In *Proceedings of the First International Conference on Evolutionary Computation and its Applications (EvCA'96)*, 1996.

[110] N. D. Monekosso and P. Remagnino. Phe-Q: A pheromone based Q-learning. In *Australian Joint Conference on Artificial Intelligence*, pages 345–355, 2001.

[111] R. B. Myerson. *Game Theory. Analysis of Conflict*. Harvard University Press, 1991.

[112] J. Nachbar. Prediction, optimization, and learning in repeated games. *Econometrica*, 65(2):275–309, 1997.

[113] J. H. Nachbar and W. R. Zame. Non-computable strategies and discounted repeated games. *Economic Theory*, 8:103–122, 1996.

[114] R. Nair, D. Pynadath, M. Yokoo, M. Tambe, and S. Marsella. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, 2003.

[115] J. Nash. Non-cooperative games. *Annals of Mathematics*, 54:286–295, 1951.

[116] T. Nguyen and N. Jennings. Coordinating multiple concurrent negotiations. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS 2004)*. ACM Press, 2004.

[117] E. Nudelman, J. Wortman, Y. Shoham, and K. Leyton-Brown. Run the GAMUT: A comprehensive approach to evaluating game-theoretic algorithms. In *Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 2004.

[118] M. Osborne and A. Rubinstein. *Bargaining and Markets*. Academic Press, 1990.

[119] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[120] A. Osepayshvili, M. P. Wellman, D. M. Reeves, and J. K. MacKie-Mason. Self-confirming price prediction for bidding in simultaneous ascending auctions. In *Twenty First Conference on Uncertainty in Artificial Intelligence*, pages 441–449, 2005.

[121] L. Panait, R. P. Wiegand, and S. Luke. A visual demonstration of convergence properties of cooperative coevolution. In *Parallel Problem Solving from Nature — PPSN-2004*, pages 892–901. Springer, 2004.

[122] L. A. Panait, R. P. Wiegand, and S. Luke. Improving coevolutionary search for optimal multiagent behaviors. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, 2003.

[123] C. Papadimitriou. Algorithms, games, and the internet. In *Proceedings of the ACM Symposium on Theory of Computing (STOC 2001)*, pages 749–753. ACM Press, 2001.

[124] C. Papadimitriou and J. Tsitsiklis. Complexity of markov decision processes. *Mathematics of Operations Research*, 12(3):441–450, 1987.

[125] L. Peshkin, K.-E. Kim, N. Meuleau, and L. Kaelbling. Learning to cooperate via policy search. In *Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 307–314. Morgan Kaufmann, 2000.

[126] R. Planqué, N. Britton, N. Franks, and M. A. Peletier. The adaptiveness of defense strategies against cuckoo parasitism. *Bull. Math. Biol.*, 64:1045–1068, 2001.

[127] M. Potter, L. Meeden, and A. Schultz. Heterogeneity in the coevolved behaviors of mobile robots: The emergence of specialists. In *Proceedings of The Seventeenth International Conference on Artificial Intelligence (IJCAI-2001)*, 2001.

[128] R. Powers and Y. Shoham. New criteria and a new algorithm for learning in multi-agent systems. In *Neural Information Processing Systems (NIPS)*, 2004.

[129] R. Powers and Y. Shoham. Learning against opponents with bounded memory. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[130] C. Preist, A. Byde, , and C. Bartolini. Economic dynamics of agents in multiple autions. In *Proceedings of the fifth International Conference on Autonomous Agents*, pages 545–551, 2001.

[131] M. Quinn. Evolving communication without dedicated communication channels. In *Advances in Artificial Life: Sixth European Conference on Artificial Life (ECAL01)*, 2001.

[132] A. Rapoport, M. Guyer, , and D. Gordon. *The 2x2 Game*. MI: University of Michigan Press, 1976.

[133] F. Redondo. *Game Theory and Economics*. Cambridge University Press, 2001.

[134] D. M. Reeves, M. P. Wellman, J. K. MacKie-Mason, and A. Osepayshvili. Exploring bidding strategies for market-based scheduling. *Decision Support Systems*, 39:67–85, 2005.

[135] L. Rejeb, Z. Guessoum, and R. MHallah. An adaptive approach for the exploration-exploitation dilemma and its application to economic systems. In *LAMAS*, 2005.

[136] V. Robu and J. La Poutré. Learning the structure of utility graphs used in multi-issue negotiation through collaborative filtering. In *Proceedings of the Pacific Rim International Workshop on Multi-Agents (PRIMA'05), Springer Lecture Notes in Artificial Intelligence (LNCS / LNAI)*, 2005. to appear.

[137] V. Robu, K. Somefun, and J. La Poutré. Modeling complex multi-issue negotiations using utility graphs. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS 2005)*. ACM Press, 2005.

[138] J. Rosenschein and G. Zlotkin. *Rules of Encounter*. MIT Press, 1994.

[139] A. Rubinstein. *Modeling Bounded Rationality*. MIT Press, Cambridge, MA, 1998.

[140] R. Salustowicz, M. Wiering, and J. Schmidhuber. Learning team strategies with multiple policy-sharing agents: A soccer case study. Technical report, ISDIA, Corso Elvezia 36, 6900 Lugano, Switzerland, 1997.

[141] L. Samuelson. *Evolutionary Games and Equilibrium Selection*. MIT Press, Cambridge, MA, 1997.

[142] T. Sandholm and S. Suri. BOB: Improved winner determination in combinatorial auctions and generalizations. *Artificial Intelligence*, 145:33–58, 2003.

[143] T. W. Sandholm and R. H. Crites. On multiagent Q-learning in a semi-competitive domain. In *G. Weiss and S. Sen, editors, Adaptation and Learning in Multiagent Systems*, pages 191–205. Springer Verlag, 1996.

[144] H. Scarf and T. Hansen. *The Computation of Economic Equilibria*. Yale University Press, 1973.

[145] S. Sen and M. Sekaran. Multiagent coordination with learning classifier systems. In G. Weiß and S. Sen, editors, *Proceedings of the IJCAI Workshop on Adaption and Learning in Multi-Agent Systems*, volume 1042, pages 218–233. Springer Verlag, 1996.

[146] S. Sen and M. Sekaran. Individual learning of coordination knowledge. *Journal of Experimental and Theoretical Artificial Intelligence*, 10(3):333–356, 1998.

[147] S. Sen and G. Weiss. *Learning in Multiagent Systems*, chapter 6. MIT Press, Cambridge MA, 1999.

[148] Y. Shoham, R. Powers, and T. Grenager. Multi-agent reinforcement learning: a critical survey. In *AAAI Fall Symposium on Artificial Multi-Agent Learning*, 2004.

[149] C. Sierra. Agent-mediated electronic commerce. *Autonomous Agents and MultiAgent Systems*, 9(3):285–301, 2004.

[150] H. Simon. *Models of Bounded Rationality, volume 2*. MIT Press, 1982.

[151] S. P. Singh, M. J. Kearns, and Y. Mansour. Nash convergence of gradient dynamics in general-sum games. In *UAI '00: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 541–548, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[152] K. Somefun, E. Gerding, S. Bohté, and J. La Poutré. Automated negotiation and bundling of information goods. In *In: Agent Mediated Electronic Commerce V (AMECV), Springer Lecture Notes in Artificial Intelligence (LNAI), 3048*, pages 1–17, 2004.

[153] K. Somefun, E. Gerding, S. Bohté, and J. La Poutré. Efficient methods for automated multi-issue negotiation: Negotiating over a two-part tariff. *International Journal of Intelligent Systems (special issue on Learning Approaches for Negotiation Agents and Automated Negotiation)*, 2006. to appear.

[154] K. Somefun, T. Klos, and H. La Poutré. Negotiating over bundles and prices using aggregate knowledge. In *Proceedings of the 5th International Conference on Electronic Commerce and Web Technologies (EC-Web), Springer Lecture Notes in Computer Science (LNCS), 3182*, pages 218–227, 2004.

[155] K. Somefun, T. Klos, and H. La Poutré. Online learning of aggregate knowledge about nonlinear preferences applied to negotiating prices and bundles. In *Proceedings of the Sixth International Conference on Electronic Commerce (ICEC04)*, pages 361–370. ACM Press, 2005.

[156] K. Somefun and J. La Poutré. Bundling and pricing for information brokerage: Customer satisfaction as a means to profit optimization. In *Proceedings of the IEEE/WIC International Conference on Web Intellingence (WI2003)*, pages 182–189. IEEE Computer Society press, 2003.

[157] P. Stone. *Layered Learning in Multi-Agent Systems*. PhD thesis, Carnegie Mellon University, 1998.

[158] P. Stone and M. Littman. Implicit negotiation in repeated games. In *In: J.-J. Meyer and M. Tambe (eds.), Proceedings of The Eighth International Workshop on Agent Theories, Architectures, and Languages (ATAL-2001)*, pages 393–404, 2001.

[159] P. Stone, R. S. P., M. L. Littman, J. A. Csirik, and D. McAllester. Decision-theoretic bidding based on learned density models in simultaneous, interacting auctions. *Journal of Artificial Intelligence Research*, 19:209–242, 2003.

[160] P. Stone and M. M. Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383, 2000.

[161] N. Suematsu and A. Hayashi. A multiagent reinforcement learning algorithm using extended optimal response. In *Proceedings of First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pages 370–377, 2002.

[162] D. Suryadi and P. J. Gmytrasiewicz. Learning models of other agents using influence diagrams. In *Preceedings of the 1999 International Conference on User Modeling*, pages 223–232, 1999.

[163] R. Sutton and A. Barto. *Reinforcement Learning: An introduction.* Cambridge, MA: MIT Press, 1998.

[164] P. 't Hoen and J. La Poutré. A decommitment strategy in a competitive multiagent transportation setting. In *Agent Mediated Electronic Commerce V (AMEC-V), Springer Lecture Notes in Artificial Intelligence (LNAI), 3048*, pages 56–72, 2004.

[165] P. 't Hoen and J. La Poutré. Repeated auctions with complementarities. In *Proceedings of the 7th International Workshop on Agent-Mediated Electronic Commerce (AMEC VII), Springer Lecture Note in Artificial Intelligence*, 2006. to appear.

[166] P. 't Hoen and K. Tuyls. Analyzing multi-agent reinforcement learning using evolutionary dynamics. In *Proceedings of the 15th European Conference on Machine Learning (ECML)*, 2004.

[167] tac dev@sics.se. Trading agent competitition (tac): Tac classic and TAC supply chain management (scm), http://www.sics.se/tac, 2006.

[168] P. Tadepalli, R. Givan, and K. Driessens. Relational reinforcement learning: An overview. In P. Tadepalli, R. Givan, and K. Driessens, editors, *Proceedings of the ICML'04 Workshop on Relational Reinforcement Learning*, pages 1–9, 2004.

[169] M. Tan. Multi-agent reinforcement learning: Independent vs. cooperative learning. In M. N. Huhns and M. P. Singh, editors, *Readings in Agents*, pages 487–494. Morgan Kaufmann, San Francisco, CA, USA, 1993.

[170] G. Tesauro. Extending Q-learning to general adaptive multi-agent systems. In *Neural Information Processing Systems (NIPS)*, 2003.
[171] L. Tesfatsion. Introduction to the special issue on agent-based computational economics. *Journal of Economic Dynamics and Control*, 25:281–293, 2001.
[172] K. Tumer and A. Agogino. Efficient reward functions for adaptive multi-rover systems. In *LAMAS*, 2005.
[173] K. Tuyls, T. Croonenborghs, J. Ramon, R. Goetschalckx, and M. Bruynooghe. Multi-agent relational reinforcement learning. In *LAMAS*, 2005.
[174] K. Tuyls, D. Heytens, A. Now, and B. Manderick. Extended replicator dynamics as a key to reinforcement learning in multi-agent systems. In *14th European Conference on Machine Learning, Cavtat-Dubrovnik, Croatia, September 22-26, 2003*, volume Lecture Notes in Computer Science 2837, pages 421–431, 2003.
[175] K. Tuyls and A. Nowé. Evolutionary game theory and multi-agent reinforcement learning. *The Knowledge Engineering Review*, 20(01):63–90, 2006.
[176] K. Tuyls, K. Verbeeck, and T. Lenaerts. A selection-mutation model for Q-learning in Multi-Agent Systems. In *The second International Joint Conference on Autonomous Agents and Multi-Agent Systems. ACM Press, Melbourne, Australia*, 2003.
[177] D. van Bragt and J. La Poutré. Co-evolving automata negotiate with a variety of opponents. In *Proceedings of the IEEE Congress on Evolutionary Computation 2002 (CEC 2002)*, volume 2, pages 1426–1431. IEEE Press, 2002.
[178] D. van Bragt and J. La Poutré. Why agents for automated negotiation should be adaptive. *Netnomics*, 5:101–118, 2003.
[179] S. van Otterloo. The value of privacy. In *AAMAS*, 2005.
[180] I. Vermeulen, K. Somefun, and H. La Poutré. An efficient turnkey agent for repeated trading with overall budget and preferences. In *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems (CIS 2004)*, pages 1072–1077. IEEE Press, 2004.
[181] J. Vidal and E. Durfee. The impact of nested agent models in an information economy. In *Proceedings Of the 2nd Intern. Conf. On Multiagent Systems*, pages 377–384. AAAI press, 1996.
[182] P. Vrancx, A. Nowé, and K. Steenhaut. Multi-type ACO for light path protection. In *LAMAS*, 2005.
[183] K. Wagner. Cooperative strategies and the evolution of communication. *Artificial Life*, 6(2):149–179, Spring 2000.
[184] W. Walsh, R. Das, G. Tesauro, , and J. Kephart. Analyzing complex strategic interactions in multi-agent games. In *Proceedings of the The Eighteenth National Conference on Artificial Intelligence (AAAI-02) Workshop on Game Theoretic and Decision Theoretic Agents*, pages 109–118, 2002.
[185] X. Wang and T. Sandholm. Reinforcement learning to play an optimal Nash equilibrium in team Markov games. In *Advances in Neural Information Processing Systems (NIPS-2002)*, 2002.

[186] C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, University of Cambridge, 1989.

[187] J. Weibull. *Evolutionary Game Theory*. MIT Press, 1996.

[188] M. Weinberg and J. S. Rosenschein. Best-response multiagent learning in non-stationary environments. In *The Third International Joint Conference on Autonomous Agents and Multiagent Systems*, New York, July 2004.

[189] G. Weiss. *Multi-agent Systems: A Modern Approach to Distributed Artificial Intelligence*. MIT Press, Cambridge, 1999.

[190] M. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research*, 1:1–23, 1993.

[191] M. Wellman, A. Greenwald, P. Stone, and P. Wurman. The 2001 Trading Agent Competition. *Electronic Markets*, 13:4–12, 2003.

[192] M. Wellman and J. Hu. Conjectural equilibrium in multiagent learning. *Machine Learning*, 33(2-3):179–200, 1998.

[193] M. Wellman, D. Reeves, and a. Y. V. K. Lochner. Price prediction in a trading agent competition. *Journal of Artificial Intelligence Research*, 21:19–36, 2004.

[194] M. Wellman, P. Wurman, K. O'Malley, R. Bangera, S. d. Lin, D. Reeves, and W. Walsh. Designing the market game for the trading agent competition. *IEEE Internet Computing*, 5:43–51, 2001.

[195] R. P. Wiegand. *Analysis of Cooperative Coevolutionary Algorithms*. PhD thesis, Department of Computer Science, George Mason University, 2003.

[196] B. Wolfe, M. R. James, and S. Singh. Learning predictive state representations in dynamical systems without reset. In *Proceedings of the 2005 International Conference on Machine Learning*, 2005.

[197] D. H. Wolpert and K. Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(2/3):265–279, 2001.