Multi-Agent Reinforcement Learning - From Game Theory to Organic Computing

Maurice Gerczuk
University of Passau
Passau, Germany
gerczuk@fim.uni-passau.de

***Abstract*—Complex systems consisting of multiple agents that interact both with each other as well as their environment can often be found in both nature and technical applications. This paper gives an overview of important Multi-Agent Reinforcement Learning (MARL) concepts, challenges and current research directions. It shortly introduces traditional reinforcement learning and then shows how MARL problems can be modelled as stochastic games. Here, the type of problem and the system configuration can lead to different algorithms and training goals. Key challenges such as the curse of dimensionality, choosing the right learning goal and the coordination problem are outlined. Especially, aspects of MARL that have previously been considered from a critical point of view are discussed with regards to if and how the current research has addressed these criticism or shifted their focus. The wide range of possible MARL applications is hinted at by examples from recent research. Further, MARL is assessed from an Organic Computing point of view where it takes a central role in the context of self-learning and self-adapting systems.**

***Index Terms*—reinforcement learning, multi-agent, organic computing, game theory**

I. Introduction

Complex and intelligent systems consisting of multiple interacting agents sharing a common environment are finding application in a variety of areas including traffic control [1]–[3] and power management [4]. These systems can often be managed more easily in a distributed fashion, benefiting from reduced complexity and parallel computation [5], [6]. To ensure that such a system delivers the desired performance in a wide range of often unpredictable situations it needs to be able to adapt its behaviour. In traditional reinforcement learning, a single agent interacts with its environment and changes its policy based on rewards it receives for its actions [7]. Compared to this scenario, in a multi-agent setting, the individual agents not only adapt and learn from their shared environment but also from the actions and learning processes of all the other agents, making multi-agent reinforcement learning (*MARL*) a more complex problem overall.

This paper aims to give an overview over the complexity of MARL and how the field has traditionally been strongly linked to game theory. Furthermore, it addresses a critical perspective on some of the game-theoretic notions at the basis of MARL. It then tries to answer the research question of how the presented historical criticisms and agendas have been addressed by reviewing recent literature and developments in the field.

The paper is laid out in the following way: An overview of single-agent reinforcement learning is given in Section II after which the multi-agent case is described in Section III. The research question is addressed in Section IV and a short conclusion is given in Section V.

II. Reinforcement Learning

Reinforcement learning can be described as a subset of machine learning that distinguishes itself from other areas, like supervised machine learning, by not trying to learn from data but rather how an agent can learn to optimise its interaction with an environment in order to control it in a beneficial way [7]. A traditional reinforcement learning scenario is characterised by a model of the environment, reward and value functions that are assigned to specific actions and or environmental states, and the agent's action policy. The agent observes its environment and changes in it (e. g. by analysing sensor data). Based on these observations, it perceives the environment to be in a certain state and then proceeds to act according to its action policy. These actions in turn again transform the environment - a state transition. The agent can further assess its interaction by scalar reward values it receives for a specific state-action transition. It is essential to point out that, in contrast to supervised learning, the agent does not learn about the ”best” action it could have chosen in the state [8]. Depending on the scenario, immediate rewards may not always reflect upon the long-term reward an agent will receive. Instead, the agent tries to maximise the discounted return over the course of its entire interaction. An informal model of a reinforcement learning scenario is visualised in Figure 1.

A. Markov Decision Processes

Formally, single-agent reinforcement learning can be modelled as a Markov Decision Process (*MDP*) [9].

Definition 1: Let X be a finite set of environmental states and U contain all actions an agent can take in this environment. Further, a state transition function $f : X \times U \times X \to [0, 1]$ defines the probability of the environment transitioning from a specific state $x_k \in X$ to another state $x_{k+1} \in X$ if the agent takes a certain action $u_k \in U$. Lastly, a reward function $\rho : X \times U \times X \to \mathbb{R}$ determines a scalar reward the agent receives immediately after a certain state transition. The tuple $\langle X, U, f, \rho \rangle$ is called a Markov Decision Process.

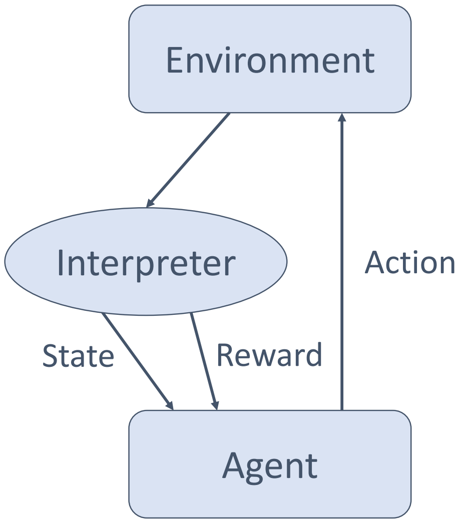

Fig. 1. In reinforcement learning, an agent perceives his environment through some sort of interpretation mechanism. It then performs actions accordingly. In response to its action, the environment may change its state and the agent receives a scalar reward.

How an agent acts in response to a certain environmental state is defined by its policy $\pi : X \times U \rightarrow [0, 1]$. It is important to note that the Markov Decision Process assumes the environment to be stationary and not influenced by any other adaptive agent [10], restrictions that are inherently not fulfilled in the multi-agent case.

B. Q-learning

One of the most popular algorithms for finding a solution to MDPs is called *Q-learning* [11]. This method relies on finding a strategy that maximises the state-value function Q which estimates the expected discounted return of a specific state action pair under a chosen policy π over the whole course of the interaction (expressed by the infinite sum over all steps t):

$$Q_\pi(x, u) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_{t+1} | \pi\right] \tag{1}$$

Here, γ is the discount factor which exponentially decreases rewards the further the corresponding state-action pairs are distanced from the initial state and action (x and u). The optimal Q function for a specific state action pair can therefore be written as $Q^*(x, u) = max_\pi Q_\pi(x, u)$. In *Q-learning* an iterative algorithm based on the Bellman equation is used to approximate this function [7]:

$$\begin{aligned} Q_{i+1}(x_i, u_i) = Q_i(x_i, u_i)+ \\ \alpha_i \left[r_{i+1} + \gamma \max_{u'} Q_i(x_{i+1}, u') - Q_i(x_i, u_i)\right] \end{aligned} \tag{2}$$

The *Q-values* for performing a specific action u_i in state x_i are updated with the immediate reward received for this action, r_{i+1}, and the discounted (by multiplication with γ) highest *Q-value* achievable in the following state x_{i+1}. $\alpha_i \in (0, 1]$ is the learning rate and is often decreased over the course of the algorithm [9]. Q-learning also forms the basis of a range of multi-agent reinforcement learning algorithms [12]–[14].

C. Deep Q-Networks

A modern variation of the algorithm comes in the form of Deep Q-Networks (*DQN*s) [15]. This algorithm combines the advances in training deep neural networks to learn useful high level representations of raw input data, e. g. in image classification [16], [17], with reinforcement learning. The *Q-function* is now no longer approximated using a linear function but by a deep neural network that introduces non-linearity. This neural network is fed raw sensory data as input from which it learns to derive a higher level representation that is useful for estimating the correct *Q-values* of actions. Using a non-linear function approximator for *Q-learning* has previously led to instability which is addressed in the DQN model by the usage of experience replay [18] and updating target values of the *Q-function* only periodically.

III. Multi-Agent Reinforcement Learning

By extending the framework of reinforcement learning to systems with multiple agents acting in shared environment, new challenges, benefits and perspectives arise. This section gives an overview over the game-theoretic and algorithmic basics of multi-agent reinforcement learning (MARL).

A. Markov Games as a model of Multi-Agent Reinforcement Learning

Compared to single-agent reinforcement learning, it becomes apparent that MARL is intrinsically linked to the field of game theory, the study of multiperson decision problems [19]. Each agent acting in a shared environment not only has to consider the effects of his own actions but is also influenced by the actions of the other agents [9].

From this point of view, the Markov Decision Process as a formal model of the single agent reinforcement learning problem can be generalised to the so-called *stochastic* or *Markov* game with multiple agents [9], [20].

Definition 2: Let X be the set of states of a shared environment and U_1,...,U_n the action sets of n agents acting in this environment. State transitions are controlled by a transition function $f : X \times \mathbf{U} \times X \rightarrow [0, 1]$, i. e. they depend on the joint actions $\mathbf{U}$ of all agents. Furthermore, each agents reward function is defined by $\rho_i : X \times \mathbf{U} \times X \rightarrow \mathbb{R}$. The tuple $\langle X, U_1, ..., U_n, f, \rho_1, ..., \rho_n\rangle$ is then called a *Markov Game*. Like in the Markov Decision Process, each agent has a policy $\pi_i : X \times U_i \rightarrow [0, 1]$ but the expected returns of each agent now depend on the joint policy of all agents, as the reward functions of the individual agents in turn also depend on the joint actions of all agents.

Here, different kinds of Markov games can be distinguished, depending on how the state of the environment is incorporated into the model [9]. In the simplest case, agents play a *static* game, i. e. there is no dynamic environmental state [21]. Such a game can be formally described as a tuple $\langle U_1, ..., U_n, \rho_1, ..., \rho_n\rangle$. Each agent i again now has a corresponding action set U_i and a reward function $\rho_i : \mathbf{U} \rightarrow \mathbb{R}$ which solely depends on the joint action space $\mathbf{U} : U_1 \times U_2 \times ... \times U_n$ of all agents, i. e. the state of the environment is disregarded. When there are only two agents, these type of stochastic games are also often referred to as *matrix* games, as the reward functions of both agents can be expressed in a matrix, the columns responding to the action of the first agent and the rows to those of the second agent [21]. In a stateful environment, a *stage* game can be seen as a *static* game that is played in this particular fixed state. Lastly, a *repeated* game is simply a *stage* game that is played more than one time by a specific set of agents. In this context, an important criterion for finding a solution is the *Nash Equilibrium*: This type of equilibrium describes a sort of status-quo from which no agent has an incentive to deviate. A game state can be seen as a such an equilibrium if each agent's strategy is a best response to the other agents' strategies [22].

B. Cooperation

The models, techniques and algorithms applied to multi-agent reinforcement learning also greatly depend on the degree of cooperation between the individual agents. From this perspective, MARL settings can be organised into three categories. In a *fully cooperative* multi-agent system all agents aim to achieve the same common goal, maximising a common discounted reward. A fully cooperative MARL scenario with a central controller can further be modelled as a traditional reinforcement learning problem in the form of a Markov Decision Process [9]. In contrast, agents can also act in a competitive environment where their individual rewards are negatively impacting the rewards of other agents. A fully competitive setting is most often restricted to a case in which two agents have opposing goals, i. e. the reward function of one agent is the negative of the other agent's reward function. From a game theoretic point of view, these are therefore commonly referred to as *zero-sum* games. Finally, MARL scenarios that can neither be described as fully cooperative or fully competitive are referred to as *mixed*. In these cases, the returns received by individual agents are not the same but correlated to the returns of the other agents in some fashion.

C. Mutual Knowledge

Another distinction in MARL can be made about the knowledge each agent has about the other agents. Claus and Boutilier differentiate two types of learners: Independent learners and joint action learners [19]. The former learn independently from one another, i. e. each agent only has information about the shared environment and not the actions or policies of other agents. This also means, that they learn Q-values for their own actions exclusively. Joint Action learners on the other hand, are able to observe all actions taken by any agent and therefore also learn Q-values for every combination of actions of the individual agents.

D. Learning Goals

Specifying a good learning goal for MARL is challenging [9]. In general, there are two aspects of learning goals that are deemed desirable in the context of multi-agent systems [21]. For one, *stability* describes the policy of an agent to converge to a stationary policy after a certain amount of time. Adaptation on the other hand expresses how an agent deals with the changing behaviour of other agents [9]. One of the most common stability requirements is convergence e. g. towards a stationary strategy [23], [24] or to a kind of equilibrium [4], [25], often the Nash Equilibrium [12], [22] mentioned in Section III-A. Convergence to equilibria has been seen as problematic by Shoham et al. [26]. These criticisms are reviewed in Section IV-A. The notion of adaptation has been represented in different ways e. g. in the concept of rationality [23], [24]. Here, if the other agents converge towards a stationary strategy, a rational learning algorithm will converge towards a best-response strategy. It is important to note, that both *stability* and *adaptation* are needed for an efficient MARL algorithm. Furthermore, these two aspects are conflicting with each other, i. e. an algorithm cannot be both perfectly stable and perfectly adaptable [9].

E. Challenges Compared to Single-Agent Case

Extending the reinforcement learning framework to the multi-agent case inherently comes with some challenges that can either be seen as intensifications of problems found in the single agent case or as entirely unique to MARL.

1) Curse of Dimensionality: A problem that can also be found with many single-agent reinforcement learning algorithms that rely on discrete state and action spaces, the so-called *Curse of Dimensionality*, becomes even more pronounced in the multi-agent case. Algorithms that estimate *Q-values* for every state-action pair are exponential in complexity with regards to the size of the state and action space. In multi-agent settings, action spaces for each agent exist, further exponentially increasing the complexity per agent.

2) Coordination: The problem of coordination between agents can arise in different ways in multi-agent reinforcement learning. For one, individual agents are always influenced not only by the shared environment but also the actions of other agents. This can lead to situations in which agents must decide in a consistent way on which one of multiple equally good joint actions to take in order to reach an optimal outcome [9]. In relation to the learning goals of MARL algorithms, coordination is also needed in cases where multiple equilibria exist. Here, agents not only have to converge towards the same but for the best result also the optimal equilibrium.

F. Sample MARL Algorithms

In the following, two MARL algorithms that aim to extend the single agent *Q-learning* method to the multi-agent case are outlined and put into the context of aspects described above.

1) Independent Q-learning: The most popular approach for MARL is known as independent Q-learning [13]. Here, each agent solves its own single-agent reinforcement problem with Q-learning in a shared environment. The agents further have no mutual knowledge about each other and only receive state and reward signals from the environment, making the

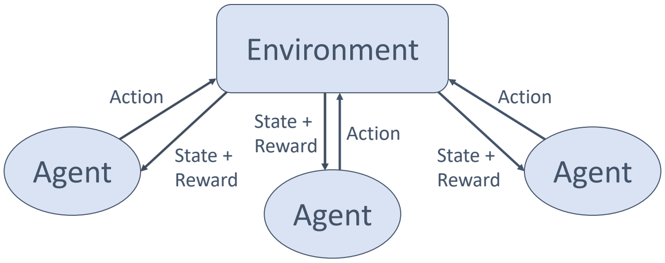

Fig. 2. In independent reinforcement learning, such as independent Q-learning, agents disregard the presence and actions of other agents in the shared environment and only act on state signals and rewards they receive individually.

approach distributed and scalable. An independent Q-learning scenario is visualised in Figure 2. This approach has also been combined with modern Deep Q-Networks [27].

2) Nash Q-learning: Nash Q-learning on the other hand displays a vastly different perspective to transfer Q-learning to general sum (as in mixed cooperation and competition) multi-agent scenarios: A centralised learning agent updates Q-values of the joint actions of all agents based on assuming Nash Equilibrium behaviour [12]. In contrast to independent Q-learning, this algorithm inherently requires full observability of the environmental state and the actions of individual agents, thus making it less scalable to systems with many agents.

IV. MARL Between Game Theory and Machine Learning

As made clear in the overview given in the previous sections, multi-agent reinforcement learning sits at the intersection of game theory and machine learning. In this section, problematic aspects of the game-theoretical approach, e. g. the focus on equilibria, are outlined. Specifically, criticisms made by Shoham et al. [26] are reviewed and put into the context of current research. It is also discussed, how their criticisms and suggestions have been addressed by current trends in MARL.

A. Problems with Game Theoretical Approach

In their critical survey published in 2003 [26], Shoham et al. reviewed a sample of multi-agent reinforcement learning literature, pointing out problems with what they call the *Bellman Heritage*, i. e. the strong focus on the Bellman equations that are at the core of many popular reinforcement learning algorithms, like *Q-learning* [11].

1) Problems with Equilibria as Training Goals: The authors argue that MARL research at that time has focused too heavily on convergence to equilibria which take a central role in game theory [22], especially the Nash Equilibrium. In their opinion, this is especially problematic in the case when the multi-agent problem cannot be described as fully competitive or fully cooperative, but mixed. They point out that this convergence criterion is not only used for evaluation but also directly incorporated into popular algorithms at that time, e. g. Nash-Q [12] or Correlated Q-learning (*CE-Q*) [28]. Using equilibria for the execution of the training algorithm can be seen as questionable and problematic in several ways using the example of Nash-Q: First of all, Nash Equilibria only describe a sort of status-quo when learning should stop, rather than making any prescriptive assumptions prior to that. This is especially the case when multiple equilibria exist in a stochastic game, then the need for a kind of oracle driving the agents towards the same optimal equilibrium arises. The concept of the Nash Equilibrium is also limited to stage games but the importance of convergence to an equilibrium in every stage game of an extended stochastic game is also questionable in the authors' opinion. These concerns about the usefulness of equilibria have also been extended and reinforced by research. Panait et al. point out, that also in cooperative settings convergence to a Nash Equilibrium might be away from team optimal solutions [29]. In their work on incorporating emotional behaviour into multi-agent systems dealing with social dilemmas, Yu et al. specifically argue that standard MARL convergence towards an equilibrium leads to mutual defection among self-interested agents preventing cooperative behaviour [30]. Shoham et al. further make the argument that equilibria might not be reached in a reasonable amount of time for complex problem spaces [26].

2) Bounded Rationality and Real-World Applicability: Another important issue with the game theory centric modelling of MARL systems can be seen in how game theory approaches the concept of ”bounded rationality”. Many MARL algorithms require exact measurements of the state and also of the other agents' actions [9] and some go further in assuming infinite mutual modelling of the other agents [26]. This view is especially inappropriate for applying MARL algorithms to real world applications where the state and action spaces are complex and it is not computationally feasible for individual agents to make comprehensive observations about their surroundings [15].

Shoham et al. also gave their opinions and suggestions on how the field of MARL research should continue to progress and outlined directions that they deemed fruitful [26]. In their paper, they postulated research agendas ranging from the field of behavioural studies to machine learning. First of all, they argue that psychological research should be made into the learning behaviours of humans in order to find a well-reasoned model for multi-agent learning settings. They further mention distributed control settings in which a central designer gives agents in a distributed system adaptive policies as a direction that excludes equilibrium analysis. Finally, they describe the so-called *AI agenda* as the most important one for the field. This agenda expresses a wish to move away from game theory and instead focus on approaches that are more strongly rooted in machine learning. Here, the question should become how an effective agent can be designed given its environment and the other agents.

B. Current Directions and Resolutions

In the following, a sample of recent literature concerning MARL algorithms and applications is reviewed with regards to how it fits into the aforementioned critique and research agendas. It should be noted that the samples have been chosen because they represent some aspect of the agendas and problems discussed above.

1) Inspiration from Human Learning Behaviours for MARL: Two examples of research on how to incorporate human learning behaviours into MARL algorithms can be found in the works of Sukhbaatar et al. [31] and Yu et al. [30]. The former investigate how effective communication between agents can be learned using backpropagation. Here, the agents are controlled by deep feedforward networks with an additional shared communication channel, represented by a continuous vector. Furthermore, their approach also incorporates the concept of "bounded rationality" as the individual agents can only partially observe their environment. To improve their performance, they learn to transmit and evaluate continuous signals about their local environments and actions to each other. They test their approach on a set of tasks including a traffic junction simulation and achieve significant improvements over baseline models without communication.

Yu et al. investigate how the emotional dynamics of self-interested humans interacting in a shared environment can be modelled to improve the performance of MARL methods for social dilemmas [30]. In a social dilemma, selfish agents must decide between pursuing strategies to increase their individual short-term rewards and choosing actions that will benefit the whole group over a larger period of time. An example of a social dilemma occurring in a multi-agent system can be found in load balancing and package routing in wireless networks [32]. If no altruistic incentives are introduced, standard MARL algorithms will often converge to a Nash Equilibrium of mutual defection. The authors argue that this goes against what can be observed and has been extensively studied about human interaction in similar settings where altruistic behaviour and in turn cooperation naturally emerge. Two appraisal variables are used in different ways to derive an emotional state and intrinsic rewards for each agent: social fairness and personal well-being. Different prioritisations and combinations of these two variables are evaluated and the authors find that for experiments on the classic prisoner's dilemma task, choosing fairness as the core appraisal variable and after that considering individual well-being leads to the highest amount of cooperation and overall rewards for all agents.

2) Influence of Deep Learning: In recent years, the increase in computational power has enabled a shift in machine learning away from the traditional careful handcrafting of algorithms and feature representations towards a trend of using deep neural networks fed with raw signals, like image, speech and video data to learn representations in a datadriven way [33]. This trend has also impacted research in the domain of reinforcement learning and in consequence also influenced MARL. Mnih et al. introduced a deep neural network model for end-to-end reinforcement learning from raw sensory input data [15], as described in section II-C. Since then, research has been made into how this model can be transferred to the multi-agent case [34], [35]. Most approaches rely on the most popular MARL algorithm, *independent Q-learning* [13] in which each agent learns separately, disregarding the other agents' presence in the environment. In their work, Tampuu et al. [27] combine independent Q-learning with DQNs to train a multi-agent system for the game of Pong. They do not focus on convergence of the algorithm towards an equilibrium but are interested in how competitive and cooperative behaviour emerges when altering the reward functions. In the author's opinion, the hype in deep learning has put a larger emphasis on artificial intelligence in MARL, conforming to the AI agenda proposed by Shoham et al. [26].

3) Connection to Organic Computing: The field of Organic Computing (*OC*) is another recent research direction that relates to MARL and the dichotomy of game theoretic and machine learning approaches taken in the field. OC is concerned with systems of autonomous sub-systems which perceive and interact with their environment and each other using sensors and actuators. These systems should be able to organise, adapt and improve themselves over the course of their runtime. OC also draws heavily from nature as inspiration on how to design such systems [36], [37]. MARL can be seen as a central component of the self-adaptation and self-learning properties of such systems. In contrast to the game theoretic view on MARL, OC also shifts the focus of MARL towards stronger imitation of natural/human behaviours. It is also more interested in emergent behaviour in multi-agent systems. Bounded rationality is also inherent to OC systems, as individual agents often only perceive their immediate environments with sensors.

V. Conclusion

In this paper, an overview of the problem of multi-agent reinforcement learning has been given. It has been outlined how MARL differs from single agent reinforcement learning and also how it is more closely related to the field of game theory. A critical perspective on early research in the field has also been reviewed and analysed with regards to how it fits into the current sphere of MARL. Specifically, the reservations about the former state of MARL research with its focus on game theory and equilibrium based methods put forward by Shoham et al. [26] have been put into context of recent trends in the field. Examples of incorporating inspiration from human behaviour, the rise of deep learning based methods and MARL's strong connection to the field of Organic Computing show ways in which these reservations have been addressed. Further research into this subject could include taking a closer look at state-of-the-art MARL algorithms or reviewing learning strategies employed in organic computing systems with respects to identifying more current paradigms. Overall, multi-agent reinforcement learning is more important than ever before in a wide range of research domains and it will be interesting to see in which ways the field might evolve in the future.

REFERENCES

[1] H. Prothmann, S. Tomforde, J. Branke, J. Hähner, C. Müller-Schloer, and H. Schmeck, “Organic traffic control,” in *Organic Computing—A Paradigm Shift for Complex Systems*. Springer, 2011, pp. 431–446.

[2] H. Prothmann, F. Rochner, S. Tomforde, J. Branke, C. Müller-Schloer, and H. Schmeck, “Organic control of traffic lights,” in *International Conference on Autonomic and Trusted Computing*. Springer, 2008, pp. 219–233.

[3] S. Tomforde, H. Prothmann, F. Rochner, J. Branke, J. Hähner, C. Müller-Schloer, and H. Schmeck, “Decentralised progressive signal systems for organic traffic control,” in *Self-Adaptive and Self-Organizing Systems, 2008. SASO’08. Second IEEE International Conference on*. IEEE, 2008, pp. 413–422.

[4] T. Yu, H. Z. Wang, B. Zhou, K. W. Chan, and J. Tang, “Multi-agent correlated equilibrium q(λ) learning for coordinated smart generation control of interconnected power grids,” *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1669–1679, Jul. 2015.

[5] N. Vlassis, “A concise introduction to multiagent systems and distributed artificial intelligence,” *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 1, no. 1, pp. 1–71, 2007.

[6] J. Ferber and G. Weiss, *Multi-agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley Reading, 1999, vol. 1.

[7] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. MIT press Cambridge, 1998, vol. 135.

[8] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.

[9] L. Busoniu, R. Babuška, and B. De Schutter, “Multi-agent reinforcement learning: An overview,” *Innovations in multi-agent systems and applications-1*, vol. 310, pp. 183–221, 2010.

[10] A. G. Barto, R. S. Sutton, and C. J. Watkins, “Learning and sequential decision making,” in *Learning and computational neuroscience*. Citeseer, 1989.

[11] C. J. Watkins and P. Dayan, “Q-learning,” *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[12] J. Hu and M. P. Wellman, “Nash q-learning for general-sum stochastic games,” *Journal of machine learning research*, vol. 4, no. Nov, pp. 1039–1069, 2003.

[13] M. Tan, “Multi-agent reinforcement learning: Independent vs. cooperative agents,” in *Proceedings of the tenth international conference on machine learning*, 1993, pp. 330–337.

[14] G. Tesauro, “Extending q-learning to general adaptive multi-agent systems,” in *Advances in neural information processing systems*, 2004, pp. 871–878.

[15] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, p. 529, 2015.

[16] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[18] L.-J. Lin, “Reinforcement learning for robots using neural networks,” Tech. Rep., 1993.

[19] C. Claus and C. Boutilier, “The dynamics of reinforcement learning in cooperative multiagent systems,” *AAAI/IAAI*, vol. 1998, pp. 746–752, 1998.

[20] M. L. Littman, “Markov games as a framework for multi-agent reinforcement learning,” in *Machine Learning Proceedings 1994*. Elsevier, 1994, pp. 157–163.

[21] M. Bowling and M. Veloso, “An analysis of stochastic game theory for multiagent reinforcement learning,” Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science, Tech. Rep., 2000.

[22] R. Gibbons, *A primer in game theory*. Harvester Wheatsheaf, 1992.

[23] M. Bowling and M. Veloso, “Rational and convergent learning in stochastic games,” in *International joint conference on artificial intelligence*, vol. 17, no. 1. Lawrence Erlbaum Associates Ltd, 2001, pp. 1021–1026.

[24] ——, “Multiagent learning using a variable learning rate,” *Artificial Intelligence*, vol. 136, no. 2, pp. 215–250, 2002.

[25] Y. Hu, Y. Gao, and B. An, “Multiagent reinforcement learning with unshared value functions,” *IEEE Transactions on Cybernetics*, vol. 45, no. 4, pp. 647–662, Apr. 2015.

[26] Y. Shoham, R. Powers, and T. Grenager, “Multi-agent reinforcement learning: a critical survey,” Technical report, Stanford University, Tech. Rep., 2003.

[27] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, “Multiagent cooperation and competition with deep reinforcement learning,” *PloS one*, vol. 12, no. 4, p. e0172395, 2017.

[28] A. Greenwald, K. Hall, and R. Serrano, “Correlated q-learning,” in *ICML*, vol. 3, 2003, pp. 242–249.

[29] L. Panait and S. Luke, “Cooperative multi-agent learning: The state of the art,” *Autonomous agents and multi-agent systems*, vol. 11, no. 3, pp. 387–434, 2005.

[30] C. Yu, M. Zhang, F. Ren, and G. Tan, “Emotional multiagent reinforcement learning in spatial social dilemmas,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 12, pp. 3083–3096, Dec. 2015.

[31] S. Sukhbaatar, R. Fergus *et al.*, “Learning multiagent communication with backpropagation,” in *Advances in Neural Information Processing Systems*, 2016, pp. 2244–2252.

[32] N. Salazar, J. A. Rodriguez-Aguilar, J. L. Arcos, A. Peleteiro, and J. C. Burguillo-Rial, “Emerging cooperation on complex networks,” in *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*. International Foundation for Autonomous Agents and Multiagent Systems, 2011, pp. 669–676.

[33] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, p. 436, 2015.

[34] J. Foerster, N. Nardelli, G. Farquhar, T. Afouras, P. H. Torr, P. Kohli, and S. Whiteson, “Stabilising experience replay for deep multi-agent reinforcement learning,” *arXiv preprint arXiv:1702.08887*, 2017.

[35] J. Foerster, I. A. Assael, N. de Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” in *Advances in Neural Information Processing Systems*, 2016, pp. 2137–2145.

[36] C. Müller-Schloer and S. Tomforde, *Organic Computing–Technical Systems for Survival in the Real World*. Springer, 2017.

[37] C. Müller-Schloer, H. Schmeck, and T. Ungerer, *Organic Computing—a Paradigm Shift for Complex Systems*. Springer Science & Business Media, 2011.