State of the Art on: Efficient Solutions for Adversarial Team Games

Federico Cacciamani, federico.cacciamani@mail.polimi.it

1. Introduction to the Research Topic

The research is focused on Algorithmic Game Theory, that is the area of research that lies on the boundary between Economics and Computer Science. More specifically, Algorithmic Game Theory is the area that aims at modelling algorithms in strategic environments, with the objective of finding *strategies*[1] that allow the involved agents to reach an *equilibrium*[1].

The strategic environments are modelled using mathematical tools, specifically Game Theory, while Computer Science techniques are used both to design algorithms and to evaluate problems' complexity and their *hardness*[2]. In particular, to find solutions some algorithms proper of the fields of Reinforcement Learning[3], Artificial Intelligence and Online Convex Optimization are used, while the evaluation of difficulty falls under the scope of Theoretical Computer Science.

Conferences and Journals

Being the research topic characterized by its belonging to various research areas, the conferences and journals that can be relevant span across all of these areas. The different conferences and journals are evaluated using several factors to identify the most relevant with respect to our research. The most commonly adopted criteria in the scientific community are:

- GGS[4] and Microsoft Academic rankings[5] to evaluate the quality of conferenes;
- IP[6] and Microsoft Academic rankings[7] to evaluate the quality of journals;
- Acceptance rate[8];
- Number of influential articles and authors in the field[9];
- Opinion of researchers working in the field.

The most relevant conferences with respect to adversarial team games and their relative research areas are:

- *AAAI: Association for the Advancement of Artificial Intelligence* - Artificial Intelligence;
- *NIPS: Neural Information Processing Systems* - Artificial Intelligence;
- *IJCAI: International Joint Conference on Artificial Intelligence* - Artificial Intelligence;
- *AAMAS: Adaptive Agents and Multi-Agents Systems* - Game Theory;
- *CDC: Conference on Decision and Control* - Game Theory;

[1] See Section 1.1.
[2] In terms of computational complexity (e.g. NP-hardness).
[3] One of the three Machine Learning paradigms.
[4] The GII-GRIN-SCIE Conference Rating, 2018, http://gii-grin-scie-rating.scie.es/conferenceRating.jsf.
[5] https://academic.microsoft.com/conferences/.
[6] Impact Factor: the number of citations received in that year of articles published in a specific journal during the two preceding years, divided by the total number of publications in that journal during the two preceding years – higher is better.
[7] https://academic.microsoft.com/journals/.
[8] Lower is better.
[9] Higher is better.

- *ACM EC: Conference on Economics and Computation* - Game Theory, Theoretical Computer Science.

The most relevant journals with respect to adversarial team games and their relative research areas are:

- *Artificial Intelligence* - Artificial Intelligence;
- *arXiv: Artificial Intelligence* - Artificial Intelligence, Planning;
- *Journal of Artificial Intelligence Research* - Artificial Intelligence;
- *Games and Economic Behavior* - Game Theory;
- *International Journal of Game Theory* - Game Theory;
- *Algorithmica* - Theoretical Computer Science.

1.1. Preliminaries

The main concepts needed to understand what Game Theory and Reinforcement Learning are about, and then to be able to classify the main works in these fields are presented below.

1.1.1 Games Representations, Strategies and Equilibria

Game theory is the name given to the methodology of using mathematical tools to model and analyze situations of interactive decision making. These are situations involving several decision makers (called *players*) with different goals, in which the decision of each player affects the outcome for all the decision makers [13]. Many of those situations can be seen as *sequential games* and are usually modelled as *extensive-form games* or *normal-form games*:

Definition 1 (Game)**.** *A game is a process consisting in:*

- *A set of players;*
- *An initial situation;*
- *Rules that players must follow;*
- *All possible final situations;*
- *The preferences of players over the set of the final situations.*

Definition 2 (Sequential Game)**.** *A sequential game is a game in which players play in succession, taking turns.*

Definition 3 (Game Tree [13])**.** *A game tree is a triple $G = (V, E, x^0)$ where (V, E) is a directed graph, $x^0 \in V$ is a vertex called the root of the tree, and for every vertex $x \in V$ there is a unique path in the graph from x^0 to x.*

Definition 4 (Normal-Form Games)**.** *A normal-form game is a tuple $< N, A, U >$, where:*

- *N is the set of players;*
- *$A = \times_{i \in N} A_i$ is the set of action profiles where A_i is the set of actions for player i;*
- *$U = (U_1, ..., U_n)$ is the set of the utility functions $U_i : A \to R$ each mapping an action profile to its respective payoff for player i.*

Definition 5 (Perfect-information extensive-form game [11])**.** *A perfect-information game in extensive form is a tuple $\Gamma = (N, A, V, L, \iota, \rho, \chi, U)$ where:*

- *N is a finite set of player;*

- *A is a set of actions;*
- *V is a set of nonterminal choice nodes;*
- *L is a set of terminal nodes, disjoint from V;*
- *$\rho : V \to 2^A$ is the action function, which assigns to each choice node a set of possible actions;*
- *$\iota : V \to N$ is the player function, which assigns to each nonterminal node a player $i \in N$ who chooses an action at that node;*
- *$\chi : V \times A \to V \cup L$ is the successor function, which maps a choice node and an action to a new choice node or terminal node;*
- *$u = (U_1, ..., U_n)$ where $U_i : L \to \mathbb{R}$ is a real-valued utility function for player i on the terminal nodes L.*

Definition 6 (Imperfect-information extensive-form game [11])**.** *An imperfect-information game in extensive form is a tuple $\Gamma = (N, A, V, L, \iota, \rho, \chi, U, H)$ where:*

- *$\Gamma' = (N, A, V, L, \iota, \rho, \chi, U)$ is a perfect-information extensive-form game;*
- *$H = (H_1, ..., H_n)$, is the set of information sets, in which H_i is a partition of V_i such that for any $x_1, x_2 \in V_i$, $\rho(x_1) = \rho(x_2)$ whenever there exists a $h \in H_i$ where $x_1 \in h$ and $x_2 \in h$.*

Moreover, an extensive-form game where, at each stage, the players recall the whole information they acquired in the previous stages is said to have *perfect recall*, otherwise is said to have *imperfect recall*.

Our study is focused in particular on the notion of *team* and on a particular class of games, characterized by the presence of a team of players facing a single adversary, denoted as *Extensive-Form Adversarial Team Games*.

Definition 7 (Team)**.** *Given an extensive form game with imperfect information Γ, a team $\mathcal{T}$ is an inclusion-wise maximal subset of players $\mathcal{T} \subseteq N$ such that for any $i, j \in \mathcal{T}$, for all $l \in L$, $U_i(l) = U_j(l)$ (all players in the same team have the same utility function).*

Players in a team can communicate in several ways. The formalization of the communication process is done through a mediator that interacts with the players of the team sending signals that influence their strategies. The type of mediators are two: one called *communication device* that supports preplay and intraplay communication and one called *correlation device* that supports only preplay communication. [22].

Definition 8 (Communication device)**.** *A communication device is a triple $(H_\mathcal{T}, A_\mathcal{T}, R^{Com})$ where $H_\mathcal{T}$ is the set of possible inputs (i.e. information sets), that teammates can communicate to the mediator, $A_\mathcal{T}$ is the set of outputs (i.e. actions) that the mediator can communicate to teammates and $R^{Com} : 2^{H_\mathcal{T}} \times 2^{A_\mathcal{T}} \to \Delta(A_\mathcal{T})$ is the recommendation function that associates each information set $h \in H_\mathcal{T}$ with a probability distribution over $\rho(h)$, as a function of information sets previously reported by teammates and of the actions recommended by the mediator in the past.*

Definition 9 (Correlation device)**.** *A correlation device is a pair $(\{P_i\}_{i \in \mathcal{T}}, R^{Cor})$ where $R^{Cor} : \times_{i \in \mathcal{T}} P_i \to \Delta(\times_{i \in \mathcal{T}} P_i)$ is the recommendation function which returns a probability distribution over the reduced joint plans of the teammates*

The study of any type of game can be conducted in several ways. One of the most straightforward is based on the *normal form* representation of the game, in which for each player $i \in N$ actions are plans $p \in P_i$, called *pure strategies*, specifying a move at each information set. A *mixed strategy* σ_i is defined as a probability distribution over the set of pure strategies P_i.

Another possible game representation is the agent form [4], in which players play *behavioral strategies* $\pi(h, a)$, each specifying a probability distribution over the actions $\rho(h)$ available at information set $h \in H_i$.

Objective of the study is to find *equilibria* of the game. The most common notion of equilibrium is the concept of Nash Equilibrium (NE) [2]. A NE is a strategy profile in which no player can improve his or her utility by deviating from his/her strategy once fixed the strategy of all other players.

A strategy solution defined in the context of adversarial team games is the *Team-Maxmin Equilibrium* (TME) [6], that is the NE maximizing the team expected utility. Starting from the TME we can define two different Team-Maxmin Equilibria obtained when different mediators for communication are used [22]. Those are TMEcor, that is the TME reachable when a correlation device is used, and TMEcom, that is the TME reachable when a communication device is used.

1.1.2 Reinforcement Learning

Reinforcement Learning is one of the three Machine Learning paradigms, together with Supervised Learning and Unsupervised Learning. RL can be defined as learning what to do (how to map situations to actions) so as to maximize a numerical reward signal. In this section basic notions of RL are given, for details see Sutton and Barto in [28].

In general, RL operates in the context of Markov Decision Processes (MDPs).

Definition 10 (Markov Decision Process)**.** *An MDP is a tuple* $< \mathcal{S}, \mathcal{A}, P, R, \gamma, \mu >$ *where:*

- $\mathcal{S}$ *is a set of states;*
- $\mathcal{A}$ *is a set of actions;*
- P *is a state transition matrix,* $P(s'|s, a)$*;*
- R *is a reward function,* $R(s, a) = E[r|s, a]$*;*
- γ *is a discount factor* $\gamma \in [0, 1]$*;*
- μ *is a set of initial probabilities* $\mu_i = P(X_0 = i)\ \forall i$*.*

At each state $s \in \mathcal{S}$ an agent selects an action $a \in \mathcal{A}$ according to a *policy* $\pi : \mathcal{S} \to \Delta(\mathcal{A})$. The learning objective is to find the policy π^* that maximizes the expected discounted return:

$$\pi^* \in argmax\ \mathbb{E}_\pi \left[R_0\right],$$

where $R_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i}$.

There are two main alternatives to compute the optimal policy; the first one consists in using *value-based algorithms*. Such algorithms compute π^* by estimating $v_\pi(s) := \mathbb{E}_\pi \left[R_t|S_t = s\right]$ (state-value function) or $Q_\pi(s, a) := \mathbb{E}_\pi \left[R_t|S_t = s, A_t = a\right]$ (action-value function) via temporal difference learning (TD-learning).

The alternative to value based algorithms is represented by *policy-based algorithms* that are a class of algorithms that directly employ *policy gradient methods*. In this case the idea is to compute the parameters of a differentiable and parametrized policy π_θ by performing gradient ascent on a score function $J(\pi_\theta)$. Within this framework it is possible to learn an approximation of the true action-value function and then alternate between the two steps of *policy evaluation* and *policy improvement* exploiting the estimated value function to improve the policy. We refer to the policy as the *actor* and to the value function as the *critic* denoting the whole framework as the *actor-critic framework*.

1.2. Research Topic

The results within the field of algorithmic game theory are mainly focused on two-players zero-sum games. The most important solutions for this class of games are based on regret-based algorithms. Examples are *Libratus* [25] and *Pluribus* [29] that were developed by Brown and Sandholm with applications in the game of heads-up no-limit poker.

In the last years the class of games in which at least a team is present gained more relevance mainly due to the fact that research in this field can be mapped to various real-world scenarios, for instance security (coordination of defenses against a malicious attacker), development of team strategies for car races or strategic bidding in the game of bridge. Standard techniques for two-players games can't be directly applied to the cases of teams and specific algorithms have to be developed. The goal of our research is to develop efficient and scalable algorithms to approximate solution for Adversarial Team Games.

There are different solution concepts that have been investigated in literature. The most important is, without any doubt, the concept of Nash Equilibrium, introduced for the first time by John Nash in [2]. In games where players are grouped in teams, however, NE is not a good solution for players in the team. Like Basilico et al. in [19] show, the NE can be arbitrarily inefficient, from the point of view of team players, with respect to another solution concept, introduced for the first time by von Stengel and Koller in [6] and called Team Maxmin Equilibrium. For this reason the community started to focus on developing algorithms to approach TME.

Basilico et al. in [19] show also that playing uncorrelated strategies can result in an arbitrary inefficient solution for the team. However, a correlation of the strategies can be very difficult to compute, due to the size of the team strategy space that grows exponentially with the number of players in the team.

In this scenario the effort is put in trying to approximate the TME with iterative algorithms. The algorithm that is proposed by Farina et al. in [26], called Fictitious Team Play, goes in this direction. In the last years, due to the growth in popularity of deep learning techniques, also in the field of adversarial team games some solution that exploit deep reinforcement learning have been proposed. One of them is presented by Celli et al. in [30], where off-policy deep reinforcement learning techniques are used.

Our intuition is to pursue the research in the direction traced in the last years and in particular to exploit techniques proper of the deep reinforcement learning field adapting them to work in combination with typical algorithms of the field of algorithmic game theory.

2. Main Related Works

2.1. Classification of the Main Related Works

The relevant results in literature are divided in three categories:

- Results on team games;
- Results on iterative solutions for two-players zero-sum games;
- Results on Reinforcement Learning in large action spaces.

Relevant Results		
Team Games	Iterative Solutions for Two-Players Z-S Games	RL in large action spaces
[26] [22] [8] [19] [30]	[9] [12] [24] [20] [7] [5] [1] [3] [17]	[16] [31] [32] [23]

2.2. Brief Description of the Main Related Works

The most relevant related works have been classified in Section 2.1. In this section they are briefly described highlighting their contributions and limitations.

2.2.1 Team Games

The presence of a team in a game brings several difficulties in computing equilibria. One of them is related to the difficulty of capturing end enforcing correlation between strategies of the team members. Basilico et al. in [19] show that for a team, playing at an NE different than the TME can be arbitrarily inefficient with respect to playing at the TME. Moreover, they introduce four different algorithms to compute or approximate a Team Maxmin Equilibrium. The best performing one is based on a global optimization approach that requires exponential time and thus is characterized by a limited scalability. In addition, Hansen et al. in [8] prove that finding a TME is FNP-hard and that its value is inapproximable in additive sense even in cases with binary payoffs.

Celli and Gatti in [22] introduce different forms of communication among team members in the form of different communication devices, describing different notions of TME, each one linked with one of the communication forms. In particular, they introduce the notion of *pre-play* and *intra-play communication*, supported by a communication device and characterized by the TMEcom equilibrium, and the notion of *ex ante communication*, supported by a correlation device and characterized by a TMEcor equilibrium. They prove that TMEcom can be found in polynomial time and that finding a TMEcor is still FNP-hard. They also introduce an algorithm to compute optimal correlated strategies for team members, that is built upon a column generation mechanism that exploits an hybrid representation of the game. Altough the use a communication device makes the problem of finding an equilibrium relatively easy (polynomial time) many practical applications don't allow intra-play communication. For this reason, together with the fact that not correlating the strategies of team members can be arbitrarily inefficient for the team, the study of situations in which a correlation device is used is pursued. Substantial effort is put in this sense by Farina et al. in [26] that develop an algorithm to find TMEcor by adapting the well known *fictitious play* [1, 3] algorithm to the case of a team and finding the best response (BR) with a MILP[10] formulation. Setting a time limit for the mixed integer linear program allows to approximate its solution and contextually find an approximate TMEcor. This solution, however, while on the one hand avoids the NP-hardness of computing the BR, on the other hand limits the application to games of small sizes, due to the presence of the MILP.

Finally, to mimic the behavior of a coordination device while using a model-free approach based on a Deep Neural Network in order to improve scalability of the algorithm, Celli et al. in [30] propose a framework based on RL to address such problem. The proposed solution is based on an architecture called *STAC* (Soft Team Actor-Critic) that is developed starting from *SAC* [27] and adapted to the case in which a team is present. For the policy evaluation step, the algorithm exploits a *value-conditioner network* whose parameters are produced by an *hypernetwork*, conditioned on the signal observed from the coordination device. The parameters of the hypernetwork, however, are fixed and can't be learned since that has proved to worsen the performance of the whole algorithm and, in some cases, even prevent convergence, thus resulting in a decrease of the learning capabilities.

2.2.2 Iterative Solutions for Two-Players Zero-Sum Games

Two-players zero-sum games have been extensively studied by the scientific community. Among the most successful achievements we can cite *Libratus* [25], an AI that was able to defeat in one versus one games four top-class human poker players in heads-up no-limit Texas hold'em. At its core, Libratus uses an improved version of an algorithm called MCCFR (Monte Carlo CFR) [10] that, in turn, is based on another algorithm called CFR [9]. CFR is an example of an iterative, regret-based algorithm. The objective of regret-based algorithms is to minimize the overall *regret*; CFR does this by minimizing an index called *counterfactual regret* that allows to work with the compact extensive-form representation of the game, thus avoiding the exponential increase in the game size resulting from a cast of the game in the normal-form representation. In games in which players have perfect recall, CFR has been proven to converge to a NE. Many variants of CFR have been proposed. The aforementioned MCCFR traverses only a portion of the game tree at each iteration, resulting in a more efficient computation. In games like poker in which the space of actions can be very large CFR can become impractical to use. To address such a problem, CFR-BR is introduced by Johanson et al. in [12], in which, at each iteration, one agent plays

[10]Mixed Integer Linear Programming.

according to CFR in an abstracted[11] game to reduce the complexity of its action space, while the other agent plays according to a best response strategy in an unabstracted game. CFR-BR has been proved to converge to an $\epsilon - NE$, with ϵ depending on the game size and on the iterations performed, and brings advantages for what concerns spatial complexity.

An adaptation of CFR has also been formulated in terms of deep learning in [24], denoted as Deep CFR. The goal of Deep CFR is to approximate the behavior of CFR without calculating and accumulating regrets at each infoset, by generalizing across similar infosets using function approximation via deep neural networks. On each iteration the algorithm conducts a constant number of game tree traversals according to a sampling similar to the one performed by MCCFR. At each encountered infoset it plays a strategy given by the output of a neural network. The objective of the training algorithm is for the neural network function to be approximately proportional to the regret that tabular CFR would have produced. Asymptotically, the average regret is proved to be bounded.

CFR-based is not the only class of iterative algorithms that have been studied and developed for two-players zero-sum games. Another notable class of games are those based on an algorithm called Fictitious Play (FP) described for the first time by George W. Brown in [1]. The claim is that in the context of a two-players zero-sum game if at each iteration each player plays the optimal pure strategy (best response) with respect to the average mixed strategy played by the opponent, then the average strategies of the players converge to a NE. The result is very important, however, the difficulty in computing best responses in large games (they require solving a linear program, that can be costly) limit its scalability. To overcome this issue, van der Genugten in [5] introduces the notion of Weakened Fictitious Play, that is a form of Fictitious Play in which best responses are approximated (ϵ-BR), with $\epsilon \to 0$ as time progresses. Following this path, Leslie and Collins formulate a generalization of a FP process, generalizing the best response differential inclusion principle proper of FP (for further details see [7]).

Even if ϵ-BR is used, however, the algorithm still suffers from the curse of dimensionality. For this reason Heinrich et al. propose Fictitious Self Play in [17], a version of FP in which best response and average strategy update routines are approximated via machine learning approaches. Following the developments in last years to the theory of neural networks and deep learning, Heinrich and Silver develop Neural Fictitious Self-Play [20], a deep reinforcement learning method for learning approximate NE of imperfect-information games. NFSP consists in a combination of FSP with neural network function approximation. The technique used is to combine off-policy reinforcement learning to predict action values and supervised learning to define the agent average strategy.

2.2.3 RL in Large Action Spaces

In the last years, in part due to the huge development of deep learning techniques, RL gained an increasing importance in the research community. For our problem we considered situations of large action spaces, that are those situations in which a full exploration of the action space is impractical and better ways to exploit the acquired experience must be developed.

A first approach is introduced by Arnold et al. in [16]. They introduce an architecture called *Wolpertinger Architecture* based on the classical actor-critic framework. Both actor and critic are implemented through deep neural networks. The actor, given the current state, returns a selected action in an embedding space called *proto-action* from which k different actions are obtained using a KNN[12] mapping and the best one of them, according to the critic, is selected. The whole architecture is trained using Deep Deterministic Policy Gradient [18].Sharama et al. in [23] propose a solution based on an hand-crafted factorization of actions. In particular, they analyze action spaces similar to the one of the Atari 2600, in which actions are a combination of an horizontal factor (go left, go right, no move), a vertical factor (go up, go down, no move) and an action factor (fire, no fire). Their insight is to learn about the single factors, instead of learning about the whole composite actions in order to improve generalization, and achieve this by using a deep neural network. While they show that this approach, applied to well-known standard techniques like Asynchronous Advantage Actor-Critic (A3C) [21] and

[11] Abstractions are smaller versions of the original game, with the purpose of capturing the most essential information.
[12] K-Nearest Neighbour.

Asynchronous N-step Q-learning (AQL) [21] is able to significantly improve performances, a similar situation in which action factors are explicitly identifiable is not comparable to our case.

Tennenholtz and Mannor in [32] present a work based on the famous Skip-Gram model used for NLP[13] [14]. The insight that justifies the adaptation of NLP techniques to RL case is that actions can take a different meaning depending on the context they are used inside. Following this intuition they introduce the model called Act2Vec and trained with SGNS[14] [15]. Finally, Chandak et al. in [31] propose to obtain a policy by combining two different components, one internal policy π_i that is able to map from state to a probability distribution over action embeddings and a function f that after sampling one action embedding from the probability distribution given by π_i maps it to a valid action. The training algorithm proposed is called *Policy Gradient with Representation for Action* (PG-RA) and it's a combination of a supervised learning algorithm for f and a reinforcement learning algorithm for π_i.

2.3. Discussion

To conclude the analysis of the available literature on Adversarial Team Games, we present a summary of what have been the main topics around which the scientific community focused the research and what are the open issues that need to be analyzed in the future.

Since the beginning of the studies in algorithmic game theory, researchers have been focusing mainly on fining Nash Equilibria two-players zero-sum games, and in the last years, such research process was able to reach outstanding results years (e.g. *Libratus* and *Pluribus*).

On the other hand, the interest for games in which more than one team of players is present has grown only in the last years and, due to the higher complications that such setting brings, the topic hasn't been completely explored. Moreover, with the increasing computational power made available by the technological progress, deep learning recently broke into the scene, offering new possibilities to complete tasks that were not approachable before because of their extreme complexity. However, being by nature DL techniques model free, they can suffer if being applied in huge spaces, preventing algorithms to offer satisfactory results.

We believe that deep learning, and in particular deep reinforcement learning, can be successfully used to reach approximate solutions in adversarial team games and our intuition is that if applied in combination with more model-oriented algorithms proper of the field of algorithmic game theory, interesting results can be achieved, leading to advancements in the knowledge of this kind of games.

[13] Natural Language Processing.
[14] Negative-Sampling Procedure.

References

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

[2] J.F. Nash. "Non-cooperative Games". In: *Annals of Mathematics* 54.2 (1951), pp. 286–295.

[3] J. Robinson. "An iterative method of solving a game". In: *Annals of mathematics* (1951), pp. 296–301.

[4] R. Selten. "Reexamination of the perfectness concept for equilibrium points in extensive games". In: *International Journal of Game Theory* 4 (1 1975). 10.1007/BF01766400, pp. 25–55. ISSN: 0020-7276. URL: http://dx.doi.org/10.1007/BF01766400.

[5] Ben van der Genugten. "A Weakened Form of Fictitious Play in Two-Person Zero-Sum Games". In: *IGTR* 2 (2000), pp. 307–328.

[6] Bernhard von Stengel and Daphne Koller. "Team-Maxmin Equilibria". In: *Games and Economic Behavior* 21 (July 2000). DOI: 10.1006/game.1997.0527.

[7] David Leslie and E.J. Collins. "Generalised weakened fictitious play". In: *Games and Economic Behavior* 56 (Aug. 2006), pp. 285–298. DOI: 10.1016/j.geb.2005.08.005.

[8] Kristoffer Arnsfelt Hansen, Thomas Dueholm Hansen, Peter Bro Miltersen, and Troels Bjerre Sørensen. "Approximability and parameterized complexity of minmax values". In: *CoRR* abs/0806.4344 (2008). arXiv: 0806.4344. URL: http://arxiv.org/abs/0806.4344.

[9] Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. "Regret Minimization in Games with Incomplete Information". In: *Advances in Neural Information Processing Systems 20*. Ed. by J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis. Curran Associates, Inc., 2008, pp. 1729–1736. URL: http://papers.nips.cc/paper/3306-regret-minimization-in-games-with-incomplete-information.pdf.

[10] Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. "Monte Carlo Sampling for Regret Minimization in Extensive Games". In: *Advances in Neural Information Processing Systems 22*. Ed. by Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta. Curran Associates, Inc., 2009, pp. 1078–1086. URL: http://papers.nips.cc/paper/3713-monte-carlo-sampling-for-regret-minimization-in-extensive-games.pdf.

[11] Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge, UK: Cambridge University Press, 2009. ISBN: 978-0-521-89943-7.

[12] Michael Johanson, Nolan Bard, Neil Burch, and Michael Bowling. "Finding Optimal Abstract Strategies in Extensive-Form Games". In: *AAAI*. 2012. URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/5156.

[13] Michael Maschler, Eilon Solan, and Shmuel Zamir. *Game Theory*. Cambridge University Press, 2013. DOI: 10.1017/CBO9780511794216.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. *Efficient Estimation of Word Representations in Vector Space*. 2013. arXiv: 1301.3781 [cs.CL].

[15] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. "Distributed Representations of Words and Phrases and their Compositionality". In: *CoRR* abs/1310.4546 (2013). arXiv: 1310.4546. URL: http://arxiv.org/abs/1310.4546.

[16] Gabriel Dulac-Arnold, Richard Evans, Peter Sunehag, and Ben Coppin. "Reinforcement Learning in Large Discrete Action Spaces". In: *CoRR* abs/1512.07679 (2015). arXiv: 1512.07679. URL: http://arxiv.org/abs/1512.07679.

[17] Johannes Heinrich, Marc Lanctot, and David Silver. "Fictitious Self-Play in Extensive-Form Games". In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 805–813. URL: http://proceedings.mlr.press/v37/heinrich15.html.

[18] Timothy P. Lillicrap et al. *Continuous control with deep reinforcement learning*. 2015. arXiv: 1509.02971 [cs.LG].

[19] Nicola Basilico, Andrea Celli, Giuseppe De Nittis, and Nicola Gatti. "Team-maxmin equilibrium: efficiency bounds and algorithms". In: *CoRR* abs/1611.06134 (2016). arXiv: 1611.06134. URL: http://arxiv.org/abs/1611.06134.

[20] Johannes Heinrich and David Silver. "Deep Reinforcement Learning from Self-Play in Imperfect-Information Games". In: *CoRR* abs/1603.01121 (2016). arXiv: 1603.01121. URL: http://arxiv.org/abs/1603.01121.

[21] Volodymyr Mnih et al. "Asynchronous Methods for Deep Reinforcement Learning". In: *CoRR* abs/1602.01783 (2016). arXiv: 1602.01783. URL: http://arxiv.org/abs/1602.01783.

[22] Andrea Celli and Nicola Gatti. "Computational Results for Extensive-Form Adversarial Team Games". In: *CoRR* abs/1711.06930 (2017). arXiv: 1711.06930. URL: http://arxiv.org/abs/1711.06930.

[23] Sahil Sharma, Aravind Suresh, Rahul Ramesh, and Balaraman Ravindran. "Learning to Factor Policies and Action-Value Functions: Factored Action Space Representations for Deep Reinforcement learning". In: *CoRR* abs/1705.07269 (2017). arXiv: 1705.07269. URL: http://arxiv.org/abs/1705.07269.

[24] Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. "Deep Counterfactual Regret Minimization". In: *CoRR* abs/1811.00164 (2018). arXiv: 1811.00164. URL: http://arxiv.org/abs/1811.00164.

[25] Noam Brown and Tuomas Sandholm. "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals". In: *Science* 359.6374 (2018), pp. 418–424. DOI: 10.1126/science.aao1733.

[26] Gabriele Farina, Andrea Celli, Nicola Gatti, and Tuomas Sandholm. "Ex ante coordination and collusion in zero-sum multi-player extensive-form games." In: *NeurIPS*. Ed. by Samy Bengio et al. 2018, pp. 9661–9671. URL: http://dblp.uni-trier.de/db/conf/nips/nips2018.html#FarinaCOS18.

[27] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor". In: *CoRR* abs/1801.01290 (2018). arXiv: 1801.01290. URL: http://arxiv.org/abs/1801.01290.

[28] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. Second. The MIT Press, 2018. URL: http://incompleteideas.net/book/the-book-2nd.html.

[29] Noam Brown and Tuomas Sandholm. "Superhuman AI for multiplayer poker". In: *Science* 365.6456 (2019), pp. 885–890. ISSN: 0036-8075. DOI: 10.1126/science.aay2400. eprint: https://science.sciencemag.org/content/365/6456/885.full.pdf. URL: https://science.sciencemag.org/content/365/6456/885.

[30] Andrea Celli, Marco Ciccone, Raffaele Bongo, and Nicola Gatti. *Coordination in Adversarial Sequential Team Games via Multi-Agent Deep Reinforcement Learning*. 2019. arXiv: 1912.07712 [cs.AI].

[31] Yash Chandak, Georgios Theocharous, James Kostas, Scott M. Jordan, and Philip S. Thomas. "Learning Action Representations for Reinforcement Learning". In: *CoRR* abs/1902.00183 (2019). arXiv: 1902.00183. URL: http://arxiv.org/abs/1902.00183.

[32] Guy Tennenholtz and Shie Mannor. "The Natural Language of Actions". In: *CoRR* abs/1902.01119 (2019). arXiv: 1902.01119. URL: http://arxiv.org/abs/1902.01119.