Contents

Preface

The vitality of the computing and communications industry is remarkable. Accomplished experts occasionally fall into the trap of thinking that all major inventions in their field have been made, and that what remains to be discovered are mere refinements to known methods. Advances in the field of computing and communications have proven experts wrong time and time again. Current notable examples include 5G communication networks and AI computing technologies. Such major technological advances are deeply rooted in mathematical science.

How can we ensure the industry preserves the vitality necessary to generate such advances? With this sizeable challenge in mind, we assembled a committee comprised of both mathematical scientists and engineering experts from the computing and communications industry to compile The Review of Mathematical Science in Computing and Communications. The committee is devoted to providing answers to the question presented above. The review intends to provide an array of insights from a variety of different perspectives.

Mathematics is a beautifully self-coherent body of interconnected concepts, but it was not developed in isolation – it has been accompanied all the way by natural science, especially physics. Bertrand Russell once said, "Physics is mathematical not because we know so much about the physical world, but because we know so little; it is only its mathematical properties that we can discover". Mathematics has been regarded as the universal language for natural science, and likewise, physics has been described as a rich source of inspiration and insight in mathematics. Cross-disciplinary work has led to some of the greatest discoveries of all time. For example, Newton's pursuit of classical mechanics resulted in the invention of calculus. David Hilbert, best known as the man who set the agenda for twentieth-century mathematics with his famous 23 problems, defined the differential equations of gravity that gave mathematical formulation to Einstein's theory of general relativity. Eugene Wigner went so far as to describe the intimacy between mathematics and physics as "a miracle", and his experiences consistently proved him right. Leading institutes around the world, such as IAS and IHES, have achieved great success in both mathematics and physics by promoting intimate exchanges across the two fields of study.

In contrast, the relationship between modern computing, communications and mathematics has been slightly less direct. A history can be traced back to Turing, Von Neumann and Shannon, the three mathematicians who have come to be seen as the founding fathers of computing and communications. Their work followed a familiar pattern: first, being drawn to a practical challenge (such as sending information across a noisy channel, or performing complex computational calculations) with a particular set of tools available (such as a transistor capable of processing bits); second, formulating the challenge into a mathematical problem; and last, developing mathematical solutions to prove that the theory addresses practical difficulties. In more recent times, Hinton's work on backpropagation generated so much excitement in the study of AI that neural network processing quickly became the new focus of the computing industry. Can the drive to obtain knowledge from massive amounts of data, to simulate complex phenomena accurately, dealing with intrinsic uncertainty, and communicating at the semantic level – rather than at the bit level – become the inspiration for a new generation of mathematicians? We believe this review will persuade many others to build fruitful relationships between computing, communications and mathematics.

Many industry visionaries have long realized that the key to business success is to convert scientific methods into technologies, subsequently applying them to product design through the process of research and development (R&D). The bulk of research is often conducted well before the R&D process. Such a realization has led to confusion among leaders about how to support research in mathematical science. We have designed this review to assuage this confusion by addressing the major challenges we face in the industry, and open up mathematical problems to more fruitful cross-disciplinary discussion. This will help strengthen the confidence and commitment from industry leaders to provide sustained support for mathematical studies, and to direct resources in the most effective directions.

Researchers and developers in the communications and computing industries are mostly trained in highly specialized domains. They often lack an up-to-date knowledge and awareness of mathematical science beyond their own niche. As a result, many opportunities for applying new mathematical methods to solve problems in engineering are lost, simply because they are developed in other fields. Many engineers also lack the training to be able to convert engineering problems into mathematical models, and so must seek help from mathematicians. This review aims to serve as a map for engineers, to help them navigate the boundary between engineering and mathematics.

Most areas of mathematical science are highly practical. Although some mathematicians primarily focus on proving theorems, others create and apply models to solve real-life problems. It is not uncommon for mathematicians to underestimate the impact of their work. Equally, many mathematicians are not fully aware of the key mathematical problems in any given applied domain. One major purpose of this review, therefore, is to highlight the main mathematical problems currently being dealt with in computing and communications industries, and to encourage more mathematicians to direct their efforts towards solving them. It is hoped that interdisciplinary research can be promoted to maximize both its academic impact and the benefits it brings for society.

It must be stressed that the role of advancing fundamental research in computing and communications industries cannot be undertaken by one organization alone. The Review is intended as a beacon to generate a new wave of excitement among the international research community. Ideally, this review will motivate policymakers and university executives to fund research and build education programs with a clearer purpose. We hope it will inspire a new generation of young mathematicians to join this grand effort. Finally, we intend to persuade researchers to check their direction against ours and set their new course accordingly.

The committee is extraordinary in its makeup, with scholars from the core of mathematics and experts who have made outstanding contributions to the foundation of modern communication networks and advanced computing devices. We greatly appreciate and sincerely thank the contributors for their capacity to envision a new era of mathematical science that will pave the way for the creation of new machines that can perceive, learn, communicate, think and create.

19
Multi-Agent Reinforcement Learning

Machine learning can be considered as the process of converting data into knowledge [365]. The input for a learning algorithm is training data (for example, images containing cats), and the output is some knowledge (for example, rules about how to detect cats in an image). This knowledge usually takes the form of a computer program that can perform some task (for example, an automatic cat detector). In the last decade, significant progress has been made by a special kind of machine learning technique: deep learning [LBH15]. Deep learning also involves the conversion of training data into output knowledge, but it incorporates DNNs in the learning process. This allows the software to train itself to perform new tasks rather than simply relying on the programmer. In this way, the different kinds of DNNs [357] are able to find and disentangle feature representations [29] from high-dimensional and more complex sets of data. An uncountable number of breakthroughs in real-world AI applications have been achieved through the usage of DNNs, with the domains of computer vision [KSH12] and natural language processing [99] being the biggest beneficiaries.

On top of feature recognition from existing data, modern AI applications often require computer programs to make decisions based on the acquired knowledge (see Figure 19.0.1). To illustrate the key components of decision making, let us consider the real-world example of controlling a car to safely drive through an intersection. At each time step, a robot car can move around by steering, accelerating and braking. Its goal is to exit the intersection safely and reach the destination (with decisions: go straight, or turn left/right into another lane). Therefore, in addition to being able to detect objects such as traffic lights, lane markings, or other cars (by converting data to knowledge), we aim to find a steering policy that can control the car to make a sequence of manoeuvres so as to achieve the goal (making decisions based on the knowledge gained). In a decision-making setting such as this one, two additional challenges arise:

1 Firstly, during the decision-making process, at each time step the robot car should not only consider the immediate value of its current action, but also the consequence of its current action in the future. For example, In the case of driving

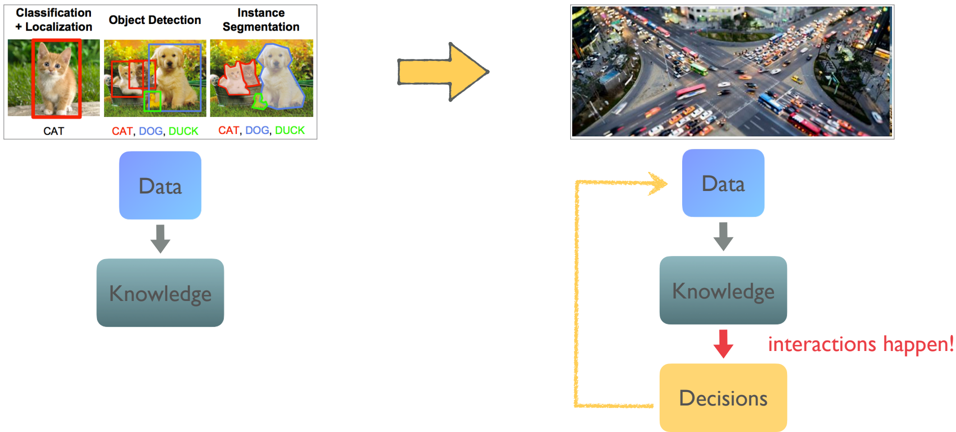

Figure 19.0.1 Modern AI applications are now being transformed from pure feature recognition (for example, detecting a cat in an image) to decision making (driving through a traffic intersection safely), where interaction among multiple agents inevitably occurs. As a result, each agent has to behave strategically. Furthermore, the problem becomes more challenging because current decisions influence the future outcomes

through an intersection, it would be detrimental to have a policy that chooses to steer in a safe direction at the beginning of the process if it would eventually lead to a car crash later on.

2 Secondly, for each decision to be made correctly and safely, the car must also consider the behavior of other cars and act correspondingly. As human drivers, for example, we often predict in advance other cars' movements and then take strategic moves in response (like giving way to an oncoming car, or speeding up to merge into another lane).

The need for an adaptive decision-making framework, together with the complexity of dealing with multiple interacting learners, has led to the development of multi-agent reinforcement learning (MARL).

MARL addresses the sequential decision-making problem of having multiple autonomous agents that operate in a common stochastic environment, each of which aims to maximize its own long-term profit by interacting with the environment and other agents. It is built on the knowledge of multi-agent systems (MAS) and reinforcement learning (RL).

19.1 Background of Reinforcement Learning

Reinforcement learning (RL) is a sub-section of machine learning, where agents learn how to behave optimally based on a trial-and-error procedure during their interaction with the environment. Unlike supervised learning that takes labeled data as its input (for example, an image labeled with cats), RL is goal-oriented: it constructs a learning model that learns to reach the optimal long-term goal by improvement through trial and error, with the learner having no labeled data to obtain knowledge from. The word "reinforcement" refers to the learning mechanism, since the actions that lead to satisfactory outcomes are reinforced in the learner's set of behaviors.

Historically, the reinforcement learning mechanism was originally observed from studying the behaviour of cats in a puzzle box [416]. [279] first proposed the computational model of reinforcement learning in his Ph.D. thesis, and named his resulting analog machine the *stochastic neural-analog reinforcement calculator*. Several years later, he first suggested the connection between the dynamic programming principle [28] and reinforcement learning [278]. In 1972, [216] integrated the trial-and-error learning process with the finding of *temporal difference (TD)*, learning from psychology. TD learning quickly turned out to be indispensable in scaling reinforcement learning for larger systems. With all these prior modes of dynamic programming and TD learning established, [441] then laid the foundation for present day RL by using the Markov decision process (MDP) and proposing the famous Q-learning method as the solver. As a dynamic programming method, the original Q-learning process inherits Bellman's "curse of dimensionality" [28], which strongly limits its applications when the number of state variables are large. To overcome such bottlenecks, [35] proposed approximate dynamic programming methods using neural networks. More recently, [283] from DeepMind made a significant breakthrough by introducing deep Q-learning (DQN) architecture that leverages the representation power of DNNs for approximate dynamic programming methods. DQN demonstrated human-level performance on 49 Atari games. Since then, deep RL techniques have become a normative approach in machine learning/AI, attracting tremendous attention from the research community.

RL originates from an understanding of animal behavior, since animals use trial-and-error to reinforce beneficial behaviors, which they then perform more frequently. During its development from this basis, computational RL incorporates ideas such as optimal control theory, and findings from psychology that help mimic the way humans make decisions, in order to maximize the long-term profit of decision making tasks. As a result, RL methods can be used naturally to train a computer program (an agent) to a level comparable to that of a human on certain tasks. The earliest success of RL methods against human players can be traced back to the game of backgammon [414]. In addition, DQN [283] shows a human level of performance playing Atari games. More recently, the advancement in using RL to solve sequential decision-making problems was marked by the remarkable success of AlphaGo series [372, 376, 374], a self-taught RL agent that beats top professional players of the game GO, a game whose search space (10^{761} possible games) is even greater than the number of atoms in the universe.

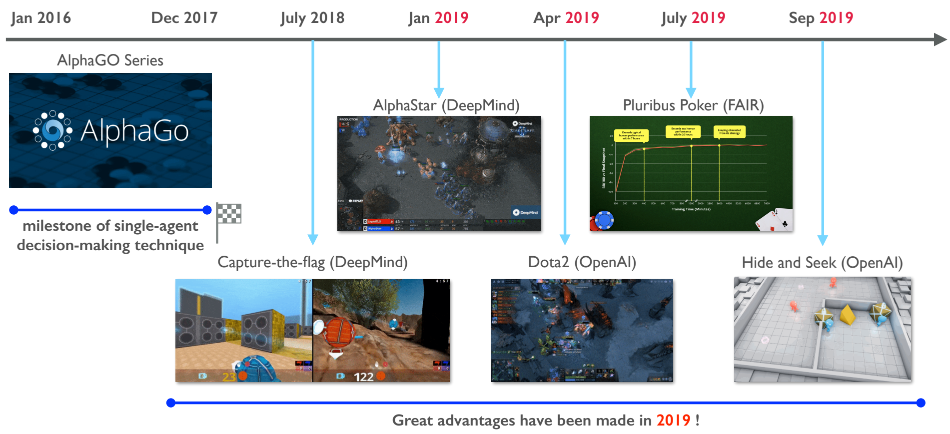

Figure 19.1.1 The success of the AlphaGo series marks the maturity of the single-agent decision-making process. The year of 2019 was a booming year for MARL techniques; remarkable progress was achieved in solving immensely challenging multi-player real-strategy video games and multi-player incomplete-information poker games

In fact, the majority of successful RL applications, such as in the game GO [1], robotic control [218], and autonomous driving [366], naturally involve the participation of multiple AI agents, which probe into the realm of MARL. As we would expect, the significant progress of single-agent RL methods - marked by the 2016 success in GO – foreshadowed the breakthroughs of multi-agent RL techniques in the following years. –

19.1.1 2019: Booming Year for MARL

The year 2019 was a booming year for MARL development as a series of breakthroughs were made in tackling immensely challenging multi-agent tasks, which people used to think were impossible to solve by AI. This being said, the progress made in the field of MARL, though remarkable, has been overshadowed to some extent by the prior success of AlphaGo [75]. It is possible that the AlphaGo series [372, 376, 374] has largely fulfilled people's expectations for the effectiveness of RL methods, such that there is lack of interest in the succeeding advancements of the field. The ripples caused by MARL progress were rather mild among the research community. In this section, we highlight several pieces of work that we believe are important and could have a profound impact on the future development of MARL techniques.

One popular test-bed of MARL is StarCraft II [431], a multi-player real-strategy

[1] Arguably, AlphaGo can also be treated as a multi-agent technique if we consider the opponent in self-play as another agent.

computer game that has its own professional league. In this game, each player has only limited information of the game state, and the dimension of the search space is orders of magnitude larger than the GO game (10^{26} possible choices for every move). Designing effective RL methods for StarCraft II was once believed a long-term challenge for AI [431]. A breakthrough was achieved by AlphaStar [430], which has demonstrated Grandmaster-level skills by ranking above 99.8% of human players. Another prominent video game-based testbed for MARL is Dota2. Dota2 is a zero-sum game play by two teams, each team having five players. From each agent's perspective, besides the difficulty of incomplete information (similar to StarCraft II), Dota 2 is more challenging in that both cooperation among teammates and competition against the opposing team need to be considered. The OpenAI Five AI system [321] demonstrated superhuman performances in Dota2 by defeating world champions in public e-sports competition.

Apart from StarCraft II and Dota2, [192] and [20] showed human-level performance in Capture-the-Flag and Hide-and-Seek game modes respectively. Although the games themselves are less sophisticated than either StarCraft II or Dota2, it is still non-trivial for AI agents to master their tactics, so the impressive performance of the agents once again proves the efficacy of MARL. Interestingly, both authors reported emergent behaviors in the AI, induced by their proposed MARL methods, that are able to be understood by humans, and are physically grounded.

One last remarkable achievement of MARL worth mentioning its application in the poker game, Texas hold' em, which is a multi-player extensive-form game with incomplete information accessible to the player. Heads-up (two player) no-limit hold'em has more than $6 * 10^{161}$ information states. Only recently have ground-breaking achievements in the game been made, thanks to MARL. Two independent programs, DeepStack [286] and Libratus [57] are both are able to beat professional human players. Even more recently, Libratus was upgraded to Pluribus [58] and showed remarkable performance by winning over one million dollars from five elite human professionals in a no-limit setting.

For a deeper understanding on RL and MARL, mathematical notation and deconstruction of the concepts is needed. In the next section, we will provide mathematical formulations for these concepts, starting from single-agent RL and progressing onto multi-agent RL methods.

19.2 Single-Agent Reinforcement Learning

Through trial and error, a RL agent tries to find the optimal policy that can maximize its long-term reward. Such a process is commonly formulated as a MDP.

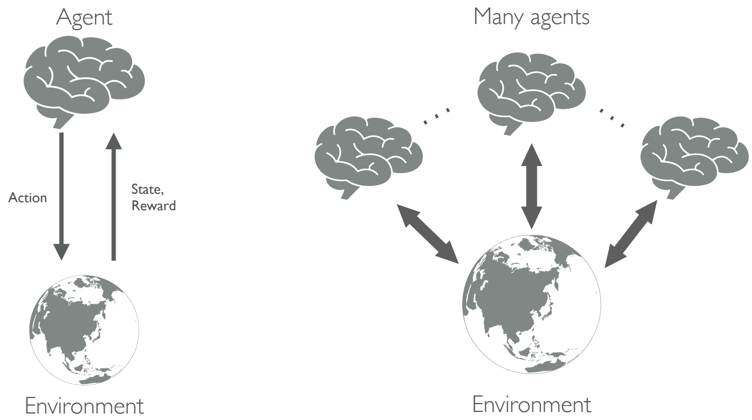

Figure 19.1.2 Diagram of a single-agent MDP (left) and multi-agent MDP/stochastic game (right)

19.2.1 Problem Formulation: Markov Decision Process

Definition 19.1 (Markov Decision Process) An MDP can be described by a number of key elements $\langle \mathbb{S}, \mathbb{A}, P, R, \gamma \rangle$.

- $\mathbb{S}$: the set of environmental states.
- $\mathbb{A}$: the set of agent's possible actions.
- $P : \mathbb{S} \times \mathbb{A} \to \Delta(\mathbb{S})$: for each timestep $t \in \mathbb{N}$, given agent's action $a \in \mathbb{A}$, the transition probability from a state $s \in \mathbb{S}$ to the state in the next timestep $s' \in \mathbb{S}$.
- $R : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$: the reward function that returns a scalar value to the agent for a transition from (s, a) to s'. The rewards have absolute values uniformly bounded by $R_{\max}$.
- $\gamma \in [0, 1]$ is the discount factor that represents the value of time.

At each time t, the environment has a state s_t. The learning agent observes this [2] and executes an action a_t. The action makes the environment transition into the next state $s_{t+1} \sim P(\cdot|s_t, a_t)$, and the new environment returns an immediate reward $R(s_t, a_t, s_{t+1})$ to the agent. The goal of the agent is to solve the MDP: to find the optimal policy that maximises reward over time. Mathematically, one common objective is for the agent to find a Markovian and stationary policy[3] function $\pi : \mathbb{S} \to \Delta(\mathbb{A})$ that can guide it to take sequential actions such that

[2] The agent can only observe part of the full environment state. The partially observable setting is introduced in Definition (19.7) as a special case of Dec-PODMP.

[3] Such an optimal policy exists as long as the transition function and the reward function are both Markovian and stationary [112].

the discounted cumulative reward is maximized:

$$\mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t)}\left[\sum_{t\geq 0}\gamma^t R\left(s_t,a_t,s_{t+1}\right)\Big| a_t\sim\pi\left(\cdot\mid s_t\right),s_0\right]. \tag{19.1}$$

Another common mathematical objective of MDP is to maximize the time-average reward:

$$\lim_{T\to\infty}\mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t)}\left[\frac{1}{T}\sum_{t=0}^{T-1}R(s_t,a_t,s_{t+1})\Big| a_t\sim\pi\left(\cdot\mid s_t\right),s_0\right], \tag{19.2}$$

which we do not consider in this work. Refer to [263] for a full analysis of this objective.

Based on the objective function of Equation 19.1, under a given policy π, we can define: the state-action function (namely the Q-function, which determines the expected return from undertaking action a in state s) and the value function (which determines the return associated with the policy) by:

$$Q^\pi(s,a)=\mathbb{E}^\pi\left[\sum_{t\geq 0}\gamma^t R\left(s_t,a_t,s_{t+1}\right)\Big| a_0=a,s_0=s\right],\forall s\in\mathbb{S},a\in\mathbb{A}, \tag{19.3}$$

$$V^\pi(s)=\mathbb{E}^\pi\left[\sum_{t\geq 0}\gamma^t R\left(s_t,a_t,s_{t+1}\right)\Big| s_0=s\right],\forall s\in\mathbb{S}, \tag{19.4}$$

where $\mathbb{E}^\pi$ is the expectation under the probability measure $\mathbb{P}^\pi$ over the set of infinitely long state-action trajectories $\tau=(s_0,a_0,s_1,a_1,...)$, and where $\mathbb{P}^\pi$ is induced by a state transition probability P, the policy π, the initial state s and an initial action a (in the case of Q-function). The connection between Q-function and value function is $V^\pi(s)=\mathbb{E}_{a\sim\pi(\cdot|s)}[Q^\pi(s,a)]$ and $Q^\pi=\mathbb{E}_{s'\sim P(\cdot|s,a)}[R(s,a,s')+V^\pi(s')]$.

19.2.2 Justification of Reward Maximisation

The current model for RL, as given by Equation 19.1, suggests that a single reward function is sufficient for whatever problem we want our “intelligent agents” to solve. The justification for this idea is deeply rooted in the *von Neumann-Morgenstern utility theory* [433]. This theory essentially proves that an agent is rational if and only if there exists a real-valued utility/reward function such that every preference of the agent is characterized by maximizing the single expected reward. In the case of the multi-objective MDP, we are still able to convert multiple objectives into a single-objective MDP by the help of a *scalarization function* through a two-timescale process, which is described in more detail in [351].

19.2.3 Solving Markov Decision Processes

One commonly used notion in MDP is the (discounted-normalized) occupancy measure $\mu^{\pi}(s,a)$ that uniquely corresponds to a given policy π and vice versa [402, Theorem 2]. This is defined by:

$$\begin{aligned}\mu^{\pi}(s,a) &= \mathbb{E}_{s_t\sim P, a_t\sim\pi}\left[(1-\gamma)\sum_{t\geq 0}\gamma^t \mathbb{1}_{(s_t=s\wedge a_t=a)}\right]\\ &= (1-\gamma)\sum_{t\geq 0}\gamma^t\mathbb{P}^{\pi}(s_t=s, a_t=a),\end{aligned} \tag{19.5}$$

where $\mathbb{1}$ is the indicator function. Note that in Equation 19.5, P is the state transitional probability, and $\mathbb{P}^{\pi}$ is the probability of state-action pairs when following the stationary policy π.

The real meaning of $\mu^{\pi}(s,a)$ is as a measure of probability, that counts the expected discounted number of visits of the individual's admissible state-action pairs. Correspondingly, $\mu^{\pi}(s) = \sum_a \mu^{\pi}(s,a)$ is the discounted state visitation frequency; the stationary distribution of the Markov process induced by π. With the occupancy measure, we can write the Equation 19.4 as an inner product of $V^{\pi}(s) = \frac{1}{1-\gamma}\langle\mu^{\pi}(s,a), R(s,a)\rangle$. This implies that solving a MDP can be regarded as a solving linear program (LP) of $\max_{\mu}\langle\mu(s,a), R(s,a)\rangle$, and so the optimal policy is then $\pi^*(a|s) = \mu^*(s,a)/\mu^*(s)$. However, this method for solving the MDP remains at a text-book level, aiming to offer theoretical insights but lacking practically in the case of a large-scale LP with millions of variables [326].

In the context of optimal control [33], dynamic-programming approaches, such as policy iteration and value iteration, can also be applied to solve the optimal policy that maximizes Equation 19.3 & Equation 19.4, but they require knowledge of the exact form of the model, the state transition function $P(\cdot|s,a)$, and the reward function $R(s,a,s')$.

In the setting of RL, on the other hand, the agent learns the optimal policy by a trial-and-error process during its interaction with the environment, rather than prior knowledge of the model. The word "learning" essentially means that the agent turns the experiences that are collected during the interaction into knowledge about the model of the environment. Based on the solution target, either the optimal policy or the optimal value function, RL algorithms can be categorized into two types: value-based methods and policy-based methods.

Value-Based RL Method

It is guaranteed that for all MDPs with finite states and actions, there exists at least one deterministic stationary optimal policy [403, 399]. Value-based methods are introduced to find the optimal Q-function Q^*, that maximizes Equation 19.3. Correspondingly, the optimal policy can be derived by taking the greedy action of

$\pi^* = \arg\max_a Q^*(s,a)$. The classical Q-learning algorithm [441] approximates Q^* by $\hat{Q}$ and updates its value via temporal-difference learning.

$$\underbrace{\hat{Q}(s_t,a_t)}_{\text{new value}} \leftarrow \underbrace{\hat{Q}(s_t,a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\Bigg(\underbrace{\underbrace{R}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_{a\in\mathbb{A}} \hat{Q}(s_{t+1},a)}_{\text{estimate of optimal value}}}_{\text{new value (temporal difference target)}} - \underbrace{\hat{Q}(s_t,a_t)}_{\text{old value}} \Bigg)}^{\text{temporal difference error (TD error)}}, \tag{19.6}$$

Theoretically, given the Bellman optimality operator $\mathbf{H}^*$, defined by:

$$(\mathbf{H}^*Q)(s,a) = \sum_{s'} P(s'|s,a) \left[R(s,a,s') + \gamma \max_{b\in\mathbb{A}} Q(s,b) \right], \tag{19.7}$$

we know it is a contraction mapping and that the optimal Q-function is the unique[4] fixed point, $\mathbf{H}^*(Q^*) = Q^*$. The Q-learning algorithm draws random samples of (s,a,R,s') in Equation 19.6 to approximate Equation 19.7, but it is still guaranteed to converge to the optimal Q-function [404] under the assumptions that the state-action sets are discrete and finite, and are visited an infinite amount of times. [291] extended the convergence result to a more realistic setting, by deriving the high probability error bound for an infinite state space with a finite number of samples.

More recently, [283] applied neural networks as a function approximator for the Q-function in updating Equation 19.6. Specifically, DQN performs the following optimization:

$$\min_\theta \mathbb{E}_{(s_t,a_t,R_t,s_{t+1})\sim\mathcal{D}} \left[\left(R_t + \gamma \max_{a\in\mathbb{A}} Q_{\theta^-}(s_{t+1},a) - Q_\theta(s_t,a_t) \right)^2 \right], \tag{19.8}$$

The neural network parameter θ is fitted by drawing i.i.d samples from the replay buffer $\mathcal{D}$, and then being updated in a supervised learning fashion. Q_{θ^-} is a slowly-updated target network that helps stabilize training. The convergence property and finite sample analysis of DQN has been studied by [462].

Policy-Based RL Method

Policy-based methods are designed to directly search over the policy space to find the optimal policy π^*. One can parameterize the policy expression: $\pi^* \approx \pi_\theta(\cdot|s)$ and update the parameter θ in the direction of maximizing the cumulative reward: $\theta \leftarrow \theta + \alpha\nabla_\theta V^{\pi_\theta}(s)$ in order to find the optimal policy. However, the gradient depends on the unknown effect of policy changes on the state distribution. The famous policy gradient (PG) theorem [400] derives an analytical solution that does not involve the state distribution, that is:

$$\nabla_\theta V^{\pi_\theta}(s) = \mathbb{E}_{s\sim\mu^{\pi_\theta}(\cdot),a\sim\pi_\theta(\cdot|s)} \Big[\nabla_\theta \log \pi_\theta(a|s) \cdot Q^{\pi_\theta}(s,a) \Big], \tag{19.9}$$

[4] Note that although the optimal Q-function is unique, its corresponding optimal policies may not be.

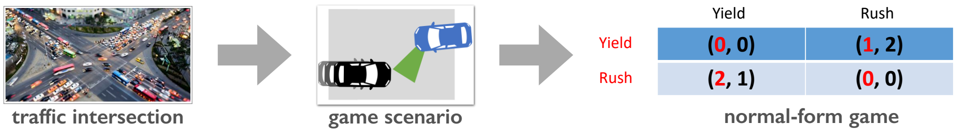

Figure 19.3.1 A snapshot of stochastic time in the intersection example. The scenario is abstracted such that there are two cars, with each car taking one of two possible actions: to yield or to rush. The outcome of each joint action pair is represented by a normal-form game, with the reward value for the row player denoted in red, and column player denoted in black. The Nash equilibria (NE) of this game are (rush, yield) and (yield, rush). If both cars maximize their own reward selfishly without considering the others, then they will end up with an accident

where μ^{π_θ} is the state occupancy measure under policy π_θ, and $\nabla \log \pi_\theta(a|s)$ is the updating score of the policy. When the policy is deterministic and the action set is continuous, we get the deterministic policy gradient (DPG) theorem [375] written as:

$$\nabla_\theta V^{\pi_\theta}(s) = \mathbb{E}_{s\sim\mu^{\pi_\theta}(\cdot)}\Big[\nabla_\theta \pi_\theta(a|s) \cdot \nabla_a Q^{\pi_\theta}(s,a)\big|_{a=\pi_\theta(s)}\Big]. \tag{19.10}$$

A classical implementation of PG theorem is REINFORCE [449] that uses a sample return $R_t = \sum_{i=t}^{T} \gamma^{i-t} r_i$ to estimate Q^{π_θ}. Alternatively, one can use a model of Q_ω (also called *critic*) to approximate the true Q^{π_θ}, and update the parameter ω via TD learning. This gives rise to the famous actor-critic methods [220, 337]. Important variants of actor-critic methods include trust-region methods [359, 360], PG with optimal baselines [443, 476], soft actor-critic methods [156], and deep deterministic policy gradient (DDPG) methods [250].

19.3 Multi-Agent Reinforcement Learning

When it comes to a multi-agent world, much like in the single-agent scenario, each agent is still trying solve the sequential decision-making problem through a trial-and-error procedure. The difference is that the evolution of the environmental state, and the reward function that each agent receives, will be now influenced by the joint actions of all agents (see Figure 19.1.2). As a result, agents need to interact not only with the environment, but also other learning agents. A decision-making process involving multiple agents is usually modeled by a stochastic game [367], also known as a Markov game [254].

19.3.1 Problem Formulation: Stochastic Game

Definition 19.2 (Stochastic Game) A stochastic game can be regarded as a multi-player[5] extension to the MDP in Definition 19.1. Therefore, it is also defined by a set of key elements $\langle N, \mathbb{S}, \{\mathbb{A}^i\}_{i\in\{1,\ldots,N\}}, P, \{R^i\}_{i\in\{1,\ldots,N\}}, \gamma\rangle$.

- N: the number of agents, $N = 1$ degenerates to single-agent MDP.
- $\mathbb{S}$: the set of environmental states shared by all agents.
- $\mathbb{A}^i$: the set of actions of agent i. We denote $\boldsymbol{\mathbb{A}} := \mathbb{A}^1 \times \cdots \times \mathbb{A}^N$.
- $P : \mathbb{S} \times \boldsymbol{\mathbb{A}} \to \Delta(\mathbb{S})$: for each timestep $t \in \mathbb{N}$, given agents' joint actions $\boldsymbol{a} \in \boldsymbol{\mathbb{A}}$, the transition probability from a state $s \in \mathbb{S}$ to the state $s' \in \mathbb{S}$ in the next timestep.
- $R^i : \mathbb{S} \times \boldsymbol{\mathbb{A}} \times \mathbb{S} \to \mathbb{R}$: the reward function that returns a scalar value to the $i - th$ agent for a transition from $(s, \boldsymbol{a})$ to s'. The rewards have absolute values uniformly bounded by $R_{\max}$.
- $\gamma \in [0, 1]$ is the discount factor that represents the value of time.

We use the superfix of $(\cdot^i, \cdot^{-i})$ (for example, $\boldsymbol{a} = (a^i, a^{-i})$), when it is necessary to distinguish between agent i and all the other $N - 1$ opponents.

Ultimately, the stochastic game (SG) acts as a framework that allows simultaneous moves from agents in a decision-making scenario[6]. The game can be described sequentially, as follows: At each time t, the environment has a state s_t, based on which each agent then executes its action a_t^i simultaneously with all others. The joint action from all agents makes the environment transition into the next state $s_{t+1} \sim P(\cdot|s_t, \boldsymbol{a}_t)$, and then the environment determines an immediate reward $R^i(s_t, \boldsymbol{a}_t, s_{t+1})$ for each agent. As seen in the single-agent MDP scenario, the goal of each agent i is to solve the SG. In other words, each agent aims to find a behavioral policy (or a mixed strategy[7] in game theory terminology) $\pi^i \in \Pi^i : \mathbb{S} \to \Delta(\mathbb{A}^i)$ that can guide the agent to take sequential actions, such that the discounted cumulative reward[8] in Equation 19.11 is maximized. Here $\Delta(\cdot)$ is the probability simplex on a set. In game theory, π^i is also called a pure strategy (vs a mixed strategy) if

[5] Player is a common word used in game theory domain; agent is more commonly used in machine learning domain. We do not discriminate their usage in this work, as well as strategy vs policy, utility/payoff vs reward. Each pair refers to the same idea of game theory usage vs machine learning usage

[6] Extensive-form games allow agents to take sequential moves, we refer the full description to [Chapter 5] of [370].

[7] Behavioural policy refers to a function map from the history $(s_0, a_0^i, s_1, a_1^i, \ldots, s_{t-1})$ to an action. Usually the policy is assumed to be Markovian such that it only depends on the current state s_t rather than the entire history. A mixed strategy refers to a randomization over pure strategies(for example, the actions). In SGs, behavioral policy and mixed policy are exactly the same. In extensive-form games, they are different, but if the agent retains history of previous actions and states (has perfect recall), each behavioral strategy has a realization-equivalent mixed strategy, and vice versa [226].

[8] Similar to single-agent MDP, we can also adopt the objective of time-average rewards.

$\Delta(\cdot)$ is replaced by a Dirac measure.

$$V^{\pi^i,\pi^{-i}}(s) = \mathbb{E}_{s_{t+1}\sim P(\cdot|s_t,a_t),a^{-i}\sim\pi^{-i}(\cdot|s_t)}\left[\sum_{t\geq 0}\gamma^t R_t^i\left(s_t,\boldsymbol{a}_t,s_{t+1}\right)\middle| a_t^i\sim\pi^i\left(\cdot\mid s_t\right),s_0\right]. \tag{19.11}$$

Comparing Equation 19.11 with Equation 19.4, it is clear that the optimal policy of each agent is not only determined by its own policy, but also the policies of the other agents in the game. This leads to fundamental differences in the *solution concept* between single-agent RL and multi-agent RL.

19.3.2 Solving Stochastic Games

A SG can be considered as a sequence of normal-form games, which are games that can be represented in a matrix. Take the original intersection scenario as an example (see Figure 19.3.1). A snapshot of the stochastic game at time t (stage game) can be represented by a normal-form game in the matrix format. The rows correspond to the action set $\mathbb{A}^1$ for agent 1, and the columns correspond to the action set $\mathbb{A}^2$ for agent 2. The values of the matrix are the rewards given for each of the joint action pairs. In this scenario, if both agents only care about maximizing their own possible reward with no consideration of other agents (the solution concept in a single-agent RL) and choose the action to rush, they will reach the outcome of crashing into each other. Of course, this is unsafe and so sub-optimal for each agent in the end, despite the fact that the possible reward was highest for each agent when rushing. Therefore, to solve a stochastic game and truly maximise cumulative reward, each agent has to take strategic actions with consideration of others when determining their optimal policy.

Unfortunately, unlike MDPs that have polynomial time-solvable linear-programming formulations, solving SGs usually involves applying Newton's method for solving nonlinear programs. However, there are two special cases of two-player general-sum discounted-reward SGs that can still be written as LPs [370] [Chapter 6.2]. They are as follows:

- *single-controller SG*: the transition dynamics are determined by a single player; $P(\cdot|\boldsymbol{a},s) = P(\cdot|a^i,s)$ if $\boldsymbol{a}[i] = a^i, \forall s\in\mathbb{S}, \forall\boldsymbol{a}\in\mathbb{A}$.
- *separable reward state independent transitions (SR-SIT) SG*: the states and the actions have independent effects on the reward function, and the transition function only depends on the joint actions:

$$\exists\alpha:\mathbb{S}\rightarrow\mathbb{R}, \beta:\mathbb{A}\rightarrow\mathbb{R}$$

such that these two conditions satisfy:

$$R^i(s,\boldsymbol{a}) = \alpha(s) + \gamma(\boldsymbol{a}), \forall i\in\{1,...,N\}, \forall s\in\mathbb{S}, \forall\boldsymbol{a}\in\mathbb{A},$$

and:

$$2)P(\cdot|s', \boldsymbol{a}) = P(\cdot|s, \boldsymbol{a}), \forall \boldsymbol{a} \in \mathbb{A}, \forall s, s' \in \mathbb{S}.$$

Value-Based MARL Method

The single-agent Q-learning process in Equation 19.6 still holds in solving the multi-agent case, but with mild adjustments [63] as follows:

$$\hat{Q}^i(s_t, \boldsymbol{a}_t) \leftarrow \hat{Q}^i(s_t, \boldsymbol{a}_t) + \alpha \cdot \bigg(R^i + \gamma \cdot \mathbf{eval}^i \Big(\big\{ Q^i(s_{t+1}, \cdot) \big\}_{i \in \{1,\dots,N\}} \Big) - Q^i(s_t, \boldsymbol{a}_t) \bigg) \tag{19.12}$$

Compared to Equation 19.6, the max operator is changed to $\mathbf{eval}^i\big(\{Q^i(s_{t+1}, \cdot)\}_{i\in\{1,\dots,N\}}\big)$ to reflect the fact each agent can no longer only consider itself, but has to also evaluate the situation of the stage game at time-step $t+1$ by considering all agents' interests, represented by the set of their Q-functions. Then, it has to be solved for the optimal policy: $\mathbf{solve}^i\big(\{Q^i(s_{t+1}, \cdot)\}_{i\in\{1,\dots,N\}}\big) = \pi^{i,*}$. Therefore, we can further write the evaluation operator as:

$$\mathbf{eval}^i \Big(\big\{ Q^i(s_{t+1}, \cdot) \big\}_{i \in \{1,\dots,N\}} \Big) = V^i \Big(s_{t+1}, \Big\{ \mathbf{solve}^i \big(\{ Q^i(s_{t+1}, \cdot) \}_{i \in \{1,\dots,N\}} \big) \Big\}_{i \in \{1,\dots,N\}} \Big). \tag{19.13}$$

In a nutshell, $\mathbf{solve}^i$ returns agent i's optimal policy at some equilibrium point (not necessarily corresponding to its largest possible reward), and $\mathbf{eval}^i$ gives agent i's expected long-term reward under this equilibrium, assuming all other agents agree to play the same equilibrium.

Policy-Based MARL Method

Value-based approaches suffer from the curse of dimensionality, due to the combinatorial nature of multi-agent systems (for further discussion see: Section (19.4.1)). This necessitates the development of policy-based algorithms with function approximations. In particular, each agent learns its own optimal policy $\pi^i_{\theta^i} : \mathbb{S} \to \Delta(\mathbb{A}^i)$ by updating the parameter θ^i of, for example, neural networks. Let $\theta = (\theta^i)_{i\in\{1,\dots,N\}}$ represent the collection of policy parameters for all agents, and $\boldsymbol{\pi}_\theta := \prod_{i\in\{1,\dots,N\}} \pi^i_{\theta^i}(a^i|s)$ be the joint policy. To optimise the parameter θ^i, the policy gradient theorem in Section (19.2.3) can be extended for the multi-agent context. Given agent i's objective function being $J^i(\theta) = \mathbb{E}_{s\sim P, \boldsymbol{a}\sim\boldsymbol{\pi}_\theta}\big[\sum_{t\geq 0} \gamma_t R^i_t\big]$, we have:

$$\nabla_{\theta^i} J^i(\theta) = \mathbb{E}_{s\sim\mu^{\boldsymbol{\pi}_\theta}(\cdot), \boldsymbol{a}\sim\boldsymbol{\pi}_\theta(\cdot|s)} \Big[\nabla_{\theta^i} \log \pi_{\theta^i}(a^i|s) \cdot Q^{i,\boldsymbol{\pi}_\theta}(s, \boldsymbol{a}) \Big]. \tag{19.14}$$

Considering the continuous action set with deterministic policy, we have the multi-agent deterministic policy gradient (MADDPG) [260], written as:

$$\nabla_{\theta^i} J^i(\theta) = \mathbb{E}_{s\sim\mu^{\boldsymbol{\pi}_\theta}(\cdot)} \Big[\nabla_{\theta^i} \log \pi_{\theta^i}(a^i|s) \cdot \nabla_{a_i} Q^{i,\boldsymbol{\pi}_\theta}(s, \boldsymbol{a}) \big|_{\boldsymbol{a}=\boldsymbol{\pi}_\theta(s)} \Big]. \tag{19.15}$$

Note that in both Equations (19.14) & (19.15), the expectation over the joint policy $\boldsymbol{\pi}_\theta$ implies that it requires to observe other agents' policies.

19.3.3 Solution Concept of Nash Equilibrium

Game theory plays a significant role in multi-agent learning by offering *solution concepts* that describe the outcomes of a game by showing which strategies will finally be adopted by players. There are many types of solution concepts for MARL (see Section 19.4.2), among which the most famous in non-cooperative[9] game theory [295] is probably the NE.

In a normal-form game, the NE characterizes an equilibrium point of the joint strategy profile $(\pi^{1,*}, ..., \pi^{N,*})$, where each agent acts in their **best response** to the others. The best response produces the optimal outcome for the player once all other players' strategies have been considered. Player i's best response[10] to π^{-i} is a set of policies such that:

$$\pi^{i,*} \in \mathbf{Br}(\pi^{-i}) = \Big\{ \arg \max_{\hat{\pi} \in \Delta(\mathbb{A}^i)} \mathbb{E}_{\hat{\pi}^i, \pi^{-i}}[R^i] \Big\}. \tag{19.16}$$

NE states that if all players are perfectly rational, none of the them will have motivation to deviate from best their response $\pi^{i,*}$ given others are playing $\pi^{-i,*}$. Note that NE is defined in terms of best response, which relies on relative reward values, suggesting that the exact values of rewards are not required for identifying NE. In fact, NE is invariant under positive affine transformations of a players' reward functions. By applying Brouwer's fixed point theorem, [295] proved that for any game with a finite set of actions, a mixed-strategy NE always exists. In the example of driving through intersections in Figure 19.3.1, the NE are $(yield, rush)$ and $(rush, yield)$.

For a SG, one commonly used equilibrium is a stronger version of the NE, called the Markov Perfect NE. [267]. It s defined by:

Definition 19.3 (Nash Equilibrium for Stochastic Game) A Markovian strategy profile $\boldsymbol{\pi}^* = (\pi^{i,*}, \pi^{-i,*})$ is a Markov perfect NE of a SG – as defined in Definition (19.2) - if the following condition holds:

$$V^{\pi^{i,*}, \pi^{-i,*}}(s) \geq V^{\pi^i, \pi^{-i,*}}(s), \quad \forall s \in \mathbb{S}, \forall \pi^i \in \Pi^i, \forall i \in \{1, ..., N\}. \tag{19.17}$$

"Markovian" means the Nash policies are measurable with respect to a particular partition of possible histories (usually referring to the last state). The word "perfect" means that the equilibrium is also subgame-perfect [362] regardless of the starting state. Considering the sequential nature of SGs, these assumptions are necessary, while still maintaining generality. Hereafter, The Markov perfect NE will be referred to as NE. It has been proven that a mixed-strategy NE [11] always

[9] "Non-cooperative" does not mean agents cannot collaborate or have to fight against each other all the time, rather it means each agent maximizes its own reward independently, and cannot group into coalitions to take joint actions.

[10] Best responses may not be unique, if a mixed-strategy best response exists, there must be at least one best response that is also a pure strategy.

[11] Note that this is different from a single-agent MDP where a single, "pure" strategy optimal policy always exists. A simple example is the Rock-Paper-Scissors game where none of the pure strategies is the NE, and the only NE is to equally mix between the three.

exists for both discounted and average-reward [12] SGs [116], though they may not be unique. In fact, checking its uniqueness is NP-hard [81]. With the NE as the solution concept of optimality, we can re-write Equation 19.13 as:

$$\mathbf{eval}^i_{\text{Nash}}\Big(\big\{Q^i(s_{t+1},\cdot)\big\}_{i\in\{1,\ldots,N\}}\Big) = V^i\Big(s_{t+1}, \Big\{\mathbf{Nash}^i\big(\{Q^i(s_{t+1},\cdot)\}_{i\in\{1,\ldots,N\}}\big)\Big\}_{i\in\{1,\ldots,N\}}\Big). \tag{19.18}$$

In Equation 19.18, $\mathbf{Nash}^i(\cdot) = \pi^{i,*}$ computes the NE of agent i's strategy, and $V^i\big(s, \{\mathbf{Nash^i}\}_{i\in\{1,\ldots,N\}}\big)$ is the expected payoff for agent i from state s onwards under this equilibrium. Equation 19.18 together with Equation 19.12 form the learning steps of Nash Q-learning [181]. This essentially leads to the outcome of a learnt set of optimal policies that reach NE for every single stage game encountered. Furthermore, similar to normal Q-learning, the Nash-Q operator defined in Equation 19.19 is also proved to be a contraction mapping, and the stochastic updating rule probably converges to NE for all states when the NE is unique:

$$(\mathbf{H}^{\text{Nash}}Q)(s,a) = \sum_{s'} P(s'|s,a)\Big[R(s,a,s') + \gamma\cdot\mathbf{eval}^i_{\text{Nash}}\Big(\big\{Q^i(s_{t+1},\cdot)\big\}_{i\in\{1,\ldots,N\}}\Big)\Big]. \tag{19.19}$$

Finding a NE in a two-player general-sum game can be formulated as a linear complementarity problem (LCP), which can then be solved using the Lemke-Howson algorithm [368]. However, the exact solution for games with more than three players is unknown. In fact, finding the NE is computationally demanding. Even in the case of two-player games, the complexity of solving the NE is $PPAD$-hard[13] (polynomial parity arguments on directed graphs) [93, 76], meaning that in the worst case scenario, it will take time that is exponential in relation to the size of the game. This prohibits any brute force or exhaustive search solutions unless $P = NP$ (see Figure 19.3.2). As we would expect, it is much more difficult to solve the NE for general SGs. In SGs, determining whether a pure-strategy NE exists is $PSPACE$-hard. Even if the SG has a finite time horizon, it still remains NP-hard [82].

19.3.4 Special Types of Stochastic Game

To summarize the solutions to SGs, one can think of the "master" equation to be:

Normal-form game solver + MDP solver = Stochastic game solver.

The first term refers to solving an equilibrium (NE) for the stage game encountered at every time-step. The second term refers to applying a RL technique (like Q-learning) to model temporal structure in the sequential decision-making process.

[12] The average-reward SGs require more subtleties because the limit of Equation 19.2 in the multi-agent setting may be a cycle and thus not exist. Instead, NE are proved to exist on a special class of irreducible SGs, where every stage game can be reached regardless of the adopted policy.

[13] The class of NP-complete is not suitable to describe the complexity of solving the NE, because the NE is proven to always exist [295], while a typical NP-complete problem - the traveling salesman problem (TSP) for example - asks the solution for the question: "Given a distance matrix and a budget B, find a tour that is cheaper than B, or report that none exists".

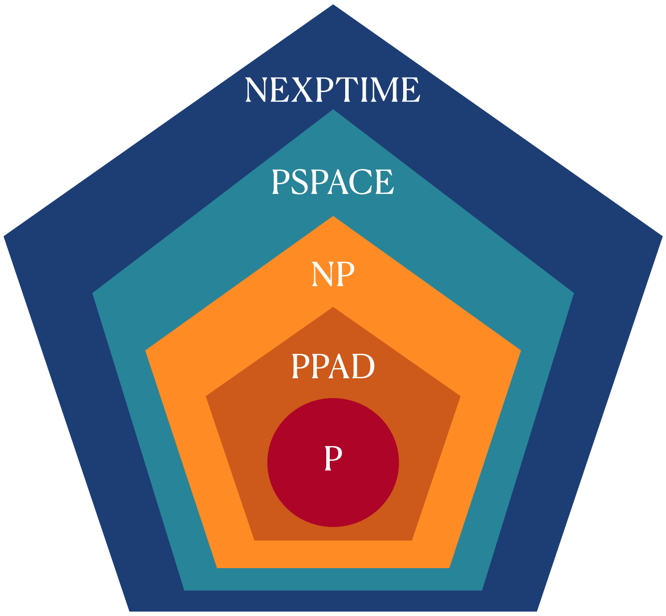

Figure 19.3.2 The landscape of different complexity classes. Relevant examples are: 1) solving the NE in a two-player zero-sum game, P [303]. 2) solving the NE in general-sum game, $PPAD$-hard [93]. 3) checking the uniqueness of the NE, NP-hard [81]. 4) checking whether a pure-strategy NE exists in a stochastic game, $PSPACE$-hard [82]. 5) solving Dec-POMDP, $NEXPTIME$-hard [32]

The combination of the two gives a solution to SGs where agents reach a particular equilibrium at each and every time-step during the game.

Since solving general SGs with NE as the solution concept is computationally challenging, researchers instead aim to study special types of SGs that have tractable solution concepts. In this section, we give a brief summary of these special types of games.

Definition 19.4 (Special Types of Stochastic Games) Given the general form of a SG defined in Definition (19.2), we have the following special cases:

- **normal-form game / repeated game**: $|S| = 1$, see the example in Figure 19.3.1. These games have only a single state. Though not theoretically grounded, it is practically easier to solve a small-scale SG.
- **identical-interest setting**[14]: agents share the same learning objective, which we denote as R. Since all agents are treated independently, each agent can safely

[14] In some of the literature on this topic, identical-interest games are equivalent to team games. Here we refer to it as a more general class of games where there exists a shared objective function that all agents collectively optimize, though their individual reward functions can still be different.

choose the action that only maximizes its own reward. As a result, single-agent RL algorithms can be applied safely, and a decentralized method developed. Several types of SGs fall in this category.

- **team games / fully-cooperative games / multi-agent MDP (MMDP)**: agents are assumed to be homogeneous and interchangeable, so importantly, they share the same reward function[15], $\mathsf{R} = R^1 = R^2 = \cdots = R^N$.
- **team-average reward games / networked multi-agent MDP (M-MDP)** : agents can have different reward functions, but they share the same objective, $\mathsf{R} = \frac{1}{N}\sum_{i=1}^{N} R^i$.
- **stochastic potential games**: agents can have different reward functions, but their mutual interests are described by a shared potential function $\mathsf{R} = \phi$, defined as, $\phi : \mathbb{S} \times \mathbb{A} \rightarrow \mathbb{R}$ such that $\forall (a^i, a^{-i}), (b^i, a^{-i}) \in \mathbb{A}, \forall i \in \{1, ..., N\}, \forall s \in \mathbb{S}$ and the following equation holds:

$$R^i\left(s, \left(a^i, a^{-i}\right)\right) - R^i\left(s, \left(b^i, a_t^{-i}\right)\right) = \phi\left(s, \left(a^i, a^{-i}\right)\right) - \phi\left(s, \left(b^i, a^{-i}\right)\right). \quad (19.20)$$

  Games of this type are guaranteed to have a pure-strategy NE. It can also be seen that potential games degenerate to team games if one chooses the reward function to be a potential function.

- **zero-sum setting**: agents share the opposite interest and act competitively, and each agent optimizes against the worst-case scenario. Elegantly, computing the NE in a zero-sum setting can be solved using a linear program (LP) in polynomial time thanks to a minimax theorem developed by [303]. The idea of min-max values is also deeply rooted in robust learning. We can subdivide the zero-sum setting:
  - **two-player constant-sum games**: $R^1(s, a, s') + R^2(s, a, s') = c, \forall (s, a, s')$, where c is a constant and usually $c = 0$.
  - **two-team competitive games**: two teams compete against each other, with team size N_1 and N_2 respectively. Their reward functions are:

$$\{R^{1,1}, ..., R^{1,N_1}, R^{2,1}, ..., R^{2,N_2}\}.$$

    Team members within a team share the same objective of either:

$$\mathsf{R}^1 = \sum_{i \in \{1,...,N_1\}} R^{1,i}/N_1$$

    , or:

$$\mathsf{R}^2 = \sum_{j \in \{1,...,N_2\}} R^{2,j}/N_2$$

    , and $\mathsf{R}^1 + \mathsf{R}^2 = 0$.

[15] In some of the literature on this topic (for example, [439]), agents are assumed to receive the same expected reward in a team game, which means in the presence of noise, different agents may receive different reward values at a particular moment.

– **harmonic games**: Any normal-form games can be decomposed into a potential game plus a harmonic game [69]. A harmonic game (for example, the Rock-Paper-Scissor game) can be regarded a general class of zero-sum games with a harmonic property. Let $\forall \boldsymbol{p} \in \mathbb{A}$ be a joint pure-strategy profile and $\mathbb{A}^{[-i]} = \{\boldsymbol{q} \in \mathbb{A} : \boldsymbol{q}^i \neq \boldsymbol{p}^i, \boldsymbol{q}^{-i} = \boldsymbol{p}^{-i}\}$ be the set of strategies that differ from $\boldsymbol{p}$ on agent i, then the harmonic property is:

$$\sum_{i \in \{1,\dots,N\}} \sum_{\boldsymbol{q} \in \mathbb{A}^{[-i]}} \big(R^i(\boldsymbol{p}) - R^i(\boldsymbol{q})\big) = 0, \qquad \forall \boldsymbol{p} \in \mathbb{A}.$$

- **linear-quadratic (LQ) setting**: the reward function is quadratic with respect to the states and actions, and the transition model follows linear dynamics. Compared to a black-box reward function, LQ games offer a much simpler setting. For example, actor-critic methods are known to facilitate convergence to the NE of zero-sum LQ games [6]. Again, the LQ setting can be subdivided:

– **two-player zero-sum LQ games**: $Q \in \mathbb{R}^{|\mathbb{S}|}, U^1 \in \mathbb{R}^{|\mathbb{A}^1|}$ and $W^2 \in \mathbb{R}^{|\mathbb{A}^2|}$ are the known cost matrices for the state and action spaces respectively, while the matrices $A \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{S}|}, B \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}^1|}, C \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}^2|}$ are usually unknown to the agent:

$$\begin{aligned} s_{t+1} &= As_t + Ba_t^1 + Ca_t^2, \qquad s_0 \sim P_0, \\ R^1(a_t^1, a_t^2) = -R^2(a_t^1, a_t^2) &= -\mathbb{E}_{s_0 \sim P_0}\left[\sum_{t\geq 0} s_t^T Q s_t + {a_t^1}^T U^1 a_t^1 - {a_t^2}^T W^2 a_t^2\right]. \end{aligned} \tag{19.21}$$

– **multi-player general-sum LQ games**: the difference with respect to a two-player game is that here the summation of agent's reward does not necessarily equal to zero:

$$\begin{aligned} s_{t+1} &= As_t + B\boldsymbol{a}_t, \qquad s_0 \sim P_0, \\ R^i(\boldsymbol{a}) &= -\mathbb{E}_{s_0 \sim P_0}\left[\sum_{t\geq 0} s_t^T Q^i s_t + {a_t^i}^T U^i a_t^i\right]. \end{aligned} \tag{19.22}$$

19.3.5 Partially Observable Setting

A partially-observable stochastic game (POSG) assumes that agents have no access to the exact environmental state, but only an observation of the state through an observation function. Formally, it is defined by:

Definition 19.5 (partially-observable stochastic games) A POSG is defined by the set $\langle N, \mathbb{S}, \{\mathbb{A}^i\}_{i\in\{1,...,N\}}, P, \{R^i\}_{i\in\{1,...,N\}}, \gamma, \underbrace{\{\mathbb{O}^i\}_{i\in\{1,...,N\}}, O}_{\text{newly added}}\rangle$, on top of the SG defined in Definition (19.2), POSGs add the following additional terms:

- $\mathbb{O}^i$: an observation set for each agent i. The joint observation set is defined $\mathbf{O} := \mathbb{O}^1 \times \cdots \times \mathbb{O}^N$.
- $O : S \times \mathbf{A} \to \Delta(\mathbf{O})$: an observation function $O(\boldsymbol{o}|\boldsymbol{a}, s')$ denotes the probability of observing $\boldsymbol{o} \in \mathbf{O}$ given the action $\boldsymbol{a} \in \mathbf{A}$, and is taken to the next state s'.

Each agent's policy now changes to $\pi^i \in \Pi^i : \mathbb{O} \to \Delta(\mathbb{A}^i)$.

Although the added partial-observability constraint is common in practice for many real-world applications, theoretically, it only exacerbates the difficulty of solving SGs. Even in the simplest setting of a two-player fully-cooperative finite-horizon game, solving a POSG is NEXP-hard (see Figure 19.3.2), which means it requires super-exponential time to solve in the worst case scenario [32]. However, the benefits of studying games in the partially-observable setting come from algorithmic advantages. Centralized-training-with-decentralized-execution methods [312, 260, 121, 345, 460] have shown many empirical successes in solving POSGs, and together with DNNs, they hold great promise.

A POSG is one of the most general class of games. An important subclass of POSGs are decentralised partially-observable MDPs (Dec-POMDP), where rewards are shared across all agents. Formally, it is defined as follows:

Definition 19.6 (Dec-POMDP) A Dec-POMDP is a special type of POSG, as defined in Definition (19.5), with $R^1 = R^2 = \cdots = R^N$.

Dec-POMDPs can be connected with a single-agent MDP through partially-observability, or connected with a stochastic team game through the assumption of identical rewards. Therefore, versions of both single-agent MDPs and team games are special types of Dec-POMDPs.

Definition 19.7 (Special types of Dec-POMDPs) The following games are special types of Dec-POMDPs.

- **partially-observable MDP (POMDP)**: there is only one agent of interest, $N = 1$. It is equivalent to a single-agent MDP in Definition (19.1) with a partial-observability constraint.
- **decentralised MDP (Dec-MDP)**: the agents in a Dec-MDP have joint full observability. That is, if all agents share their observations, they can recover the state of the Dec-MDP unambiguously. Mathematically, we have $\forall \boldsymbol{o} \in \mathbf{O}, \exists s \in \mathbb{S}$ such that $\mathbb{P}(S_t = s|\mathbf{O}_t = \boldsymbol{o}) = 1$.
- **fully-cooperative stochastic games**: assuming each agent has full observability, $\forall i = \{1, ..., N\}, \forall o^i \in O^i, \exists s \in \mathbb{S}$ such that $\mathbb{P}(S_t = s|\mathbb{O}_t = o^i) = 1$. The fully-cooperative SG from Definition (19.4) is a type of Dec-POMDP.

19.4 Grand Challenges

Compared to single-agent RL, multi-agent RL is a general framework which better matches the broad scope of real-world AI applications. However, due to the existence of multiple learning agents simultaneously, MARL methods suffer from more theoretical challenges, in addition to those already present in single-agent RL.

19.4.1 Combinatorial Complexity

In the context of multi-agent learning, each agent has to consider the other opponents' actions in order to take the best response; this is deeply rooted in each agent's reward function and shown as the joint action $\boldsymbol{a}$ in their Q-function $Q^i(s, \boldsymbol{a})$ in Equation 19.12. The size of such the joint action space is $|\mathbb{A}|^N$, which grows exponentially with the number of agents and thus largely constrains the scalability of MARL methods. Furthermore, the combinatorial complexity is worsened by the fact that solving a NE in game theory is $PPAD$-hard, even for two-player games. Therefore, for multi-player general-sum games (neither team games nor zero-sum games), it is non-trivial to find an applicable solution concept.

One common way to address this issue is by assuming certain factorized structures on action dependency, so that the reward function or the Q-function can be largely simplified. For example, a graphical game assumes an agent's reward is only affected by its neighboring agents, defined by the graph from [210]. This directly leads to a polynomial-time solution for the computation of a NE in certain tree graphs [211], though the scope of applications is rather limited beyond this.

Recent progress has also been made on leveraging special neural network architectures for Q-function decomposition [395, 345, 460]. Aside from the fact these methods can only work for the team-game setting, the majority of them lack theoretical backing. There are still open questions that need answering, such as understanding the representational power (the approximation error) of the factorized Q-functions in a multi-agent task, and how factorization itself can be learnt from scratch.

19.4.2 Multi-Dimensional Learning Objectives

Compared to single-agent RL, where the only goal is to maximize an agent's long-term reward, the learning goals in MARL are naturally multi-dimensional, as the objective of all agents are not necessarily aligned. [54, 55] proposed to classify the goals of the learning task into two types: **rationality** and **convergence**. Rationality ensures an agent takes the best possible response to the opponents when they are stationary, and convergence ensures the learning dynamics will eventually lead to a stable policy against a given class of opponents. Reaching both rationality and convergence gives rise to the achievement of the NE.

In terms of rationality, the NE characterizes a fixed point of a joint optimal strategy profile from which no agents would be motivated to deviate, as long as all of them are perfectly rational. However, in practice, an agent's rationality can be easily bound by either the cognitive limitation and/or the tractability of the decision problem. In these scenarios, the rationality assumption can be relaxed to include other types of solution concepts such as: the recursive reasoning equilibrium, which results from modeling the reasoning process recursively among agents with finite levels of hierarchical thinking (for example, an agent may reason in the following way: I believe that you believe that I believe ...) [448, 447]; best response against a target type of opponent [342]; the mean-field game equilibrium, that describes multi-agent interactions as a two-agent interaction between each agent itself and the population mean [153, 459, 458]; evolutionary stable strategies, that describes an equilibrium strategy based on its evolutionary advantage of resisting invasion by rare emerging mutant strategies [422**?**]; and the robust equilibrium (also called trembling-hand perfect equilibrium in game theory) which is stable against adversarial disturbance [247, 23, 456].

In terms of convergence, although most MARL algorithms are contrived to converge to the NE, the majority of them either lack rigorous convergence guarantee [471], or potentially converge only under strong assumptions such as the existence of a unique NE [256, 180], or are provably non-convergent in all cases [271]. [478] identified the non-convergent behavior of value-iteration methods in general-sum SGs, and instead, he proposed an alternative solution concept to the NE - *cyclic equilibria* - that value-based methods converge to. The concept of no regret (also called the Hannan consistent in game theory [158]), measures convergence by comparison against the best possible strategy in hindsight. This was also proposed as a new criteria to evaluate convergence in zero-sum self-plays [52, 160, 479]. In the two-player zero-sum games with a non-convex non-concave loss landscape (training GANs [145]), gradient-descent-ascent methods are found to reach a Stackelberg equilibrium [252, 115] or a local differential NE [272] rather than the general NE.

Finally, it is worth mentioning that despite the above solution concepts accounting for convergence, building a convergent objective for MARL methods with DNNs is still an uncharted area. This is partly because the global convergence of a single-agent deep RL algorithm, for example neural policy gradient methods [437, 258] and neural TD learning algorithms [65], have not been studied yet.

19.4.3 Non-Stationarity Issue

The most well-known challenge of multi-agent learning versus single-agent learning is probably the non-stationarity issue. Since there are multiple agents concurrently improving their policies according to their own interests, from each agent's perspective, the environmental dynamics become non-stationary and difficult to interpret when learning. This is because the agent itself cannot tell whether the state tran-

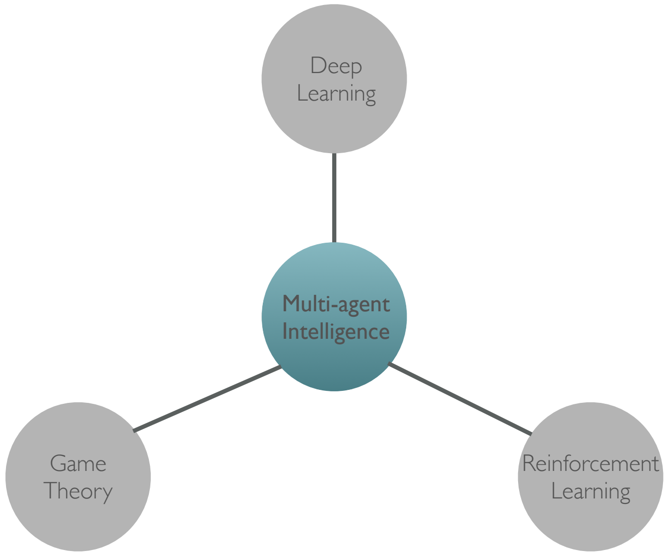

Figure 19.4.1 The scope of multi-agent intelligence, as described here, consists of three pillars. Deep learning serves as a powerful function approximation tool for the learning process. Game theory provides an effective approach to describe the outcome of learning. Reinforcement learning offers a valid approach to describe agents' incentives in multi-agent systems

sition - or the change in reward - is a genuine outcome due to its own action, or if it is due to its opponent's explorations. Although learning independently by ignoring the other agents completely can sometimes generate surprisingly powerful empirical performance [327, 270], this approach essentially harms the stationarity assumption that supports the theoretical convergence guarantee of single-agent learning methods [408]. As a result, the Markovian property of the environment is lost, and the state occupancy measure of a stationary policy in Equation 19.5 no longer exists. For example, the convergence result of single-agent policy gradient methods in MARL are provably negative even in the setting of linear-quadratic games [272].

The non-stationarity issue can be further aggravated by TD learning, which occurs with the replay buffer that most deep RL methods adopt currently [122]. In single-agent TD learning (see Equation 19.8), the agent bootstraps the current estimate of the TD error, saves it in the replay buffer, and samples the data in the replay buffer to update the value function. In the context of multi-agent learning, since the value function for one agent also depends on other agents' actions, the bootstrap process in TD learning also requires sampling the actions from other agents. This brings about two problems. First, the sampled actions barely represent the full behavior of other agents' underlying policies across different states.

Second, an agent's policy can change during training, so the samples in the replay buffer can be soon outdated. This essentially means that the dynamics that generated the data in the agent's replay buffer needs to be constantly updated to reflect the current dynamics in which it is learning. This exacerbates the non-stationarity issue.

In a nutshell, the non-stationarity issue forbids reusing the same mathematical tool for analyzing single-agent algorithms in the multi-agent context. However, there is one exception, which is the identical-interesting game in Definition (19.4). In such settings, each agent can safely perform selfishly without considering each other's policies, since it knows other agents will act in its own interest as well. The stationarity is thus maintained, so single-agent RL algorithms can still be applied.

19.4.4 Scalability Issue when $N \gg 2$

Combinatorial complexity, multi-dimensional learning objectives, and the issue of non-stationarity all result in the fact that the majority of MARL algorithms are capable of solving games with only two players, and in particular, two-player zero-sum games [471]. As a result, solutions to general-sum settings with more than two agents (for example, the many-agent problem) remains an open challenge. Such a challenge needs to be addressed from all three perspectives of multi-agent intelligence (see Figure 19.4.1): game theory, which provides realistic and tractable solution concepts to describe learning outcomes of a many-agent system; reinforcement learning algorithms, that offer provably convergent learning algorithms that can reach stable and rational equilibria in the sequential decision-making process; and finally deep learning techniques, that empower the learning algorithms with expressive function approximators.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] Baher Abdulhai, Rob Pringle, and Grigoris J Karakoulas. Reinforcement learning for true adaptive traffic signal control. *Journal of Transportation Engineering*, 129(3):278–285, 2003.

[3] Jeffrey L Adler and Victor J Blue. A cooperative multi-agent transportation management and route guidance system. *Transportation Research Part C: Emerging Technologies*, 10(5-6):433–454, 2002.

[4] A. G. Agarwal. Proceedings of the Fifth Low Temperature Conference, Madison, WI, 1999. *Semiconductors*, 66:1238, 2001.

[5] Adrian K Agogino and Kagan Tumer. Analyzing and visualizing multiagent rewards in dynamic and stochastic domains. *Autonomous Agents and Multi-Agent Systems*, 17(2):320–338, 2008.

[6] Asma Al-Tamimi, Frank L Lewis, and Murad Abu-Khalaf. Model-free q-learning designs for linear discrete-time zero-sum games with application to h-infinity control. *Automatica*, 43(3):473–481, 2007.

[7] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[8] J McKenzie Alexander. Evolutionary explanations of distributive justice. *Philosophy of Science*, 67(3):490–516, 2000.

[9] Jason Alexander and Brian Skyrms. Bargaining with neighbors: Is justice contagious? *The Journal of philosophy*, 96(11):588–598, 1999.

[10] Aamena Alshamsi and Sherief Abdallah. Multiagent self-organization for a taxi dispatch system. In *Proceedings of 8th International Conference of Autonomous Agents and Multiagent Systems, 2009*, pages 89–96, 2009.

[11] Christopher Amato, Daniel S Bernstein, and Shlomo Zilberstein. Optimizing fixed-size stochastic controllers for pomdps and decentralized pomdps. *Autonomous Agents and Multi-Agent Systems*, 21(3):293–320, 2010.

[12] Andrew Amey, John Attanucci, and Rabi Mishalani. Real-time ridesharing: opportunities and challenges in using mobile phone technology to improve rideshare services. *Transportation Research Record: Journal of the Transportation Research Board*, (2217):103–110, 2011.

[13] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[14] Gürdal Arslan and Serdar Yüksel. Decentralized q-learning for stochastic teams and games. *IEEE Transactions on Automatic Control*, 62(4):1545–1558, 2016.

[15] W. Brian Arthur. Inductive reasoning and bounded rationality. *The American Economic Review*, 84(2):406–411, 1994.

[16] Kavosh Asadi and Michael L. Littman. A new softmax operator for reinforcement learning. *CoRR*, abs/1612.05628, 2016.

[17] W Ross Ashby. Principles of the self-organizing system. In *Facets of Systems Science*, pages 521–536. Springer, 1991.

[18] Steve Baigent. Lotka-volterra dynamics - an introduction. 2006.

[19] Per Bak. *How nature works: the science of self-organized criticality.* Springer Science & Business Media, 2013.

[20] BOWEN BAKER, INGMAR KANITSCHEIDER, TODOR MARKOV, Y Wu, GLENN POWELL, B McGrew, and IGOR MORDATCH. Emergent tool use from multi-agent interaction, 2019.

[21] Maria-Florina Balcan and Kilian Q. Weinberger, editors. *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48. JMLR.org, 2016.

[22] David Balduzzi, Karl Tuyls, Julien Perolat, and Thore Graepel. Re-evaluating evaluation. In *Advances in Neural Information Processing Systems*, pages 3268–3279, 2018.

[23] R. Ballagh and C.M. Savage. Bose-einstein condensation: from atomic physics to quantum fluids. In C.M. Savage and M. Das, editors, *Proceedings of the 13th Physics Summer School*. World Scientific, Singapore, 2000.

[24] R. Ballagh and C.M. Savage. *Bose-Einstein condensation: from atomic physics to quantum fluids, Proceedings of the 13th Physics Summer School*. World Scientific, Singapore, 2000.

[25] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[26] Jeffrey A Barrett. Dynamic partitioning and the conventionality of kinds. *Philosophy of Science*, 74(4):527–546, 2007.

[27] Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors. *Advances in Neural Information Processing Systems 15 [Neural Information Processing Systems, NIPS 2002, December 9-14, 2002, Vancouver, British Columbia, Canada]*. MIT Press, 2003.

[28] Richard Bellman. On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, 38(8):716, 1952.

[29] Yoshua Bengio. *Learning deep architectures for AI*. Now Publishers Inc, 2009.

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

[31] Daniel S Bernstein, Christopher Amato, Eric A Hansen, and Shlomo Zilberstein. Policy iteration for decentralized control of markov decision processes. *Journal of Artificial Intelligence Research*, 34:89–132, 2009.

[32] Daniel S Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of operations research*, 27(4):819–840, 2002.

[33] Dimitri P Bertsekas. The dynamic programming algorithm. *Dynamic Programming and Optimal Control; Athena Scientific: Nashua, NH, USA*, pages 2–51, 2005.

[34] Dimitri P Bertsekas. Weighted sup-norm contractions in dynamic programming: A review and some new applications. *Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. LIDS-P-2884*, 2012.

[35] Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-dynamic programming.* Athena Scientific, 1996.

[36] Graeme Best, Oliver M Cliff, Timothy Patten, Ramgopal R Mettu, and Robert Fitch. Dec-mcts: Decentralized planning for multi-robot active perception. *The International Journal of Robotics Research*, 38(2-3):316–337, 2019.

[37] E. Beutler. volume 2, chapter 7, pages 654–662. McGraw-Hill, New York, 5 edition, 1994.

[38] E. Beutler. In E. Beutler, M. A. Lichtman, B. W. Coller, and T. S. Kipps, editors, *Williams Hematology*, volume 2, chapter 7, pages 654–662. McGraw-Hill, New York, 5 edition, 1994.

[39] Kurt Binder, Dieter Heermann, Lyle Roelofs, A John Mallinckrodt, and Susan McKay. Monte carlo simulation in statistical physics. *Computers in Physics*, 7(2):156–157, 1993.

[40] N. D. Birell and P. C. W. Davies. *Quantum Fields in Curved Space.* Cambridge University Press, 1982.

[41] Daan Bloembergen, Karl Tuyls, Daniel Hennes, and Michael Kaisers. Evolutionary dynamics of multi-agent learning: A survey. *Journal of Artificial Intelligence Research*, 53:659–697, 2015.

[42] Avrim Blum and Yishay Monsour. Learning, regret minimization, and equilibria. 2007.

[43] Lawrence E Blume. The statistical mechanics of strategic interaction. *Games and economic behavior*, 5(3):387–424, 1993.

[44] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[45] Christopher Boehm. The evolutionary development of morality as an effect of dominance behavior and conflict interference. *Journal of Social and Biological Structures*, 5(4):413–421, 1982.

[46] Ludwig Boltzmann. The second law of thermodynamics. In *Theoretical physics and philosophical problems*, pages 13–32. Springer, 1974.

[47] Eric Bonabeau, Guy Theraulaz, Jean-Louls Deneubourg, Serge Aron, and Scott Camazine. Self-organization in social insects. *Trends in Ecology & Evolution*, 12(5):188–193, 1997.

[48] Tilman Börgers and Rajiv Sarin. Learning through reinforcement and replicator dynamics. *Journal of Economic Theory*, 77(1):1–14, 1997.

[49] Craig Boutilier. Sequential optimality and coordination in multiagent systems. In *IJCAI*, volume 99, pages 478–485, 1999.

[50] Craig Boutilier, Thomas Dean, and Steve Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.

[51] Michael Bowling. Convergence problems of general-sum multiagent reinforcement learning. In *ICML*, pages 89–94, 2000.

[52] Michael Bowling. Convergence and no-regret in multiagent learning. In *Advances in neural information processing systems*, pages 209–216, 2005.

[53] Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold'em poker is solved. *Science*, 347(6218):145–149, 2015.

[54] Michael Bowling and Manuela Veloso. Rational and convergent learning in stochastic games. In *International joint conference on artificial intelligence*, volume 17, pages 1021–1026. Lawrence Erlbaum Associates Ltd, 2001.

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

[56] Ronen I. Brafman and Moshe Tennenholtz. Efficient learning equilibrium. In Becker et al. [27], pages 1603–1610.

[57] Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

[58] Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, 365(6456):885–890, 2019.

[59] Jingjing Bu, Lillian J Ratliff, and Mehran Mesbahi. Global convergence of policy gradient for sequential zero-sum linear quadratic dynamic games. *arXiv*, pages arXiv–1911, 2019.

[60] Wolfram Burgard, Mark Moors, Cyrill Stachniss, and Frank E Schneider. Coordinated multi-robot exploration. *IEEE Transactions on robotics*, 21(3):376–386, 2005.

[61] Y. Burstyn. Proceedings of the 5th International Molecular Beam Epitaxy Conference, Santa Fe, NM. (unpublished), 5–8 October 2004.

[62] Lucian Busoniu, Robert Babuska, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Trans. Systems, Man, and Cybernetics, Part C*, 38(2):156–172, 2008.

[63] Lucian Busoniu, Robert Babuka, and Bart De Schutter. Multi-agent reinforcement learning: An overview. In *Innovations in multi-agent systems and applications-1*, pages 183–221. Springer, 2010.

[64] H Cai, K Ren, W Zhag, K Malialis, and J Wang. Real-time bidding by reinforcement learning in display advertising. In *WSDM*. ACM, 2017.

[65] Qi Cai, Zhuoran Yang, Jason D Lee, and Zhaoran Wang. Neural temporal-difference learning converges to global optima. In *Advances in Neural Information Processing Systems*, pages 11315–11326, 2019.

[66] Scott Camazine. *Self-organization in biological systems*. Princeton University Press, 2003.

[67] Colin F Camerer, Teck-Hua Ho, and Juin-Kuan Chong. Sophisticated experience-weighted attraction learning and strategic teaching in repeated games. *Journal of Economic theory*, 104(1):137–188, 2002.

[68] Raul Campos-Rodriguez, Luis Gonzalez-Jimenez, Francisco Cervantes-Alvarez, Francisco Amezcua-Garcia, and Miguel Fernandez-Garcia. Multi-agent systems in automotive applications. *Multi-agent Systems*, page 43, 2017.

[69] Ozan Candogan, Ishai Menache, Asuman Ozdaglar, and Pablo A Parrilo. Flows and decompositions of games: Harmonic and potential games. *Mathematics of Operations Research*, 36(3):474–503, 2011.

[70] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. ACM, 2007.

[71] Alvaro Ovalle Castaneda. Deep reinforcement learning variants of multi-agent learning algorithms. 2016.

[72] Halim Ceylan and Michael GH Bell. Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. *Transportation Research Part B: Methodological*, 38(4):329–342, 2004.

[73] Georgios Chalkiadakis and Craig Boutilier. Coordination in multiagent reinforcement learning: A bayesian approach. In *AAMAS*, pages 709–716. ACM, 2003.

[74] Damien Challet and Neil F. Johnson. Optimal combinations of imperfect objects. *Phys. Rev. Lett.*, 89:028701, Jun 2002.

[75] Calum Chalmers. Is reinforcement learning worth the hype? 2020. *URL https://www.capgemini.com/gb-en/2020/05/is-reinforcement-learning-worth-the-hype/*, 2020.

[76] Xi Chen and Xiaotie Deng. Settling the complexity of two-player nash equilibrium. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 261–272. IEEE, 2006.

[77] Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. *AAAI/IAAI*, 1998:746–752, 1998.

[78] Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. *AAAI/IAAI*, 1998:746–752, 1998.

[79] Christiane Clemens and Thomas Riechmann. Evolutionary dynamics in public good games. *Computational Economics*, 28(4):399–420, 2006.

[80] Mitchell K Colby, Sepideh Kharaghani, Chris HolmesParker, and Kagan Tumer. Counterfactual exploration for improving multiagent learning. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 171–179. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[81] Vincent Conitzer and Tuomas Sandholm. Complexity results about nash equilibria. *arXiv preprint cs/0205074*, 2002.

[82] Vincent Conitzer and Tuomas Sandholm. New complexity results about nash equilibria. *Games and Economic Behavior*, 63(2):621–641, 2008.

[83] Iain Couzin. Collective minds. *Nature*, 445(7129):715–715, 2007.

[84] Iain D Couzin. Collective cognition in animal groups. *Trends in cognitive sciences*, 13(1):36–43, 2009.

[85] Iain D Couzin, Jens Krause, Richard James, Graeme D Ruxton, and Nigel R Franks. Collective memory and spatial sorting in animal groups. *Journal of theoretical biology*, 218(1):1–11, 2002.

[86] Oliver Curry and Robin IM Dunbar. Do birds of a feather flock together? *Human Nature*, 24(3):336–347, 2013.

[87] András Czirók, Albert-László Barabási, and Tamás Vicsek. Collective motion of self-propelled particles: Kinetic phase transition in one dimension. *Physical Review Letters*, 82(1):209, 1999.

[88] András Czirók and Tamás Vicsek. Collective behavior of interacting self-propelled particles. *Physica A: Statistical Mechanics and its Applications*, 281(1):17–29, 2000.

[89] Felipe Leno Da Silva and Anna Helena Reali Costa. A survey on transfer learning for multiagent reinforcement learning systems. *Journal of Artificial Intelligence Research*, 64:645–703, 2019.

[90] Emiliano Dall'Anese, Hao Zhu, and Georgios B Giannakis. Distributed optimal power flow for smart microgrids. *IEEE Transactions on Smart Grid*, 4(3):1464–1475, 2013.

[91] Justin D'Arms. Sex, fairness, and the theory of games. *The Journal of philosophy*, 93(12):615–627, 1996.

[92] Justin D'Arms, Robert Batterman, and Krzyzstof Gorny. Game theoretic explanations and the evolution of justice. *Philosophy of Science*, 65(1):76–102, 1998.

[93] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.

[94] T Dávid-Barrett and RIM Dunbar. Processing power limits social group size: computational evidence for the cognitive costs of sociality. In *Proc. R. Soc. B*, volume 280, page 20131151. The Royal Society, 2013.

[95] E. B. Davies and L. Parns. Trapped modes in acoustic waveguides. *Q. J. Mech. Appl. Math.*, 51:477–492, 1988.

[96] Enrique Munoz de Cote and Michael L. Littman. A polynomial-time nash equilibrium algorithm for repeated stochastic games. In McAllester and Myllymäki [273], pages 419–426.

[97] Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. GuessWhat?! Visual object discovery through multi-modal dialogue. nov 2016.

[98] Farnaz Derakhshan and Shamim Yousefi. A review on the applications of multiagent systems in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 15(5):1550147719850767, 2019.

[99] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[100] Sam Devlin and Daniel Kudenko. Theoretical considerations of potential-based reward shaping for multi-agent systems. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 225–232. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[101] Sam Devlin, Logan Yliniemi, Daniel Kudenko, and Kagan Tumer. Potential-based difference rewards for multiagent reinforcement learning. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 165–172. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[102] Jilles Dibangoye and Olivier Buffet. Learning to act in decentralized partially observable mdps. In *International Conference on Machine Learning*, pages 1233–1242, 2018.

[103] Jilles Steeve Dibangoye, Christopher Amato, Olivier Buffet, and François Charpillet. Optimally solving dec-pomdps as continuous-state mdps. *Journal of Artificial Intelligence Research*, 55:443–497, 2016.

[104] RIM Dunbar. Group size, vocal grooming and the origins of language. *Psychonomic Bulletin & Review*, pages 1–4, 2016.

[105] Robin IM Dunbar. Coevolution of neocortical size, group size and language in humans. *Behavioral and brain sciences*, 16(04):681–694, 1993.

[106] Paul R Ehrlich and Simon A Levin. The evolution of norms. *PLoS Biol*, 3(6):e194, 2005.

[107] A. Einstein, Yu Podolsky, and N. Rosen. *Phys. Rev.*, 47:777, 1935.

[108] Magnus Enquist and Stefano Ghirlanda. Evolution of social learning does not explain the origin of human cumulative culture. *Journal of theoretical biology*, 246(1):129–135, 2007.

[109] Magnus Enquist, Stefano Ghirlanda, Arne Jarrick, and C-A Wachtmeister. Why does human culture increase exponentially? *Theoretical population biology*, 74(1):46–55, 2008.

[110] Ido Erev and Alvin E Roth. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *American economic review*, pages 848–881, 1998.

[111] I Farkas, D Helbing, and T Vicsek. Human waves in stadiums. *Physica A: Statistical Mechanics and its Applications*, 330(1):18–24, 2003.

[112] Eugene A Feinberg. Total expected discounted reward mdps: existence of optimal policies. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

[113] R. P. Feynman. *Phys. Rev.*, 94:262, 1954.

[114] W. K. Fields. ECE Report No. AL944, 2005. Required institution missing.

[115] Tanner Fiez, Benjamin Chasnov, and Lillian J Ratliff. Convergence of learning dynamics in stackelberg games. *arXiv*, pages arXiv–1906, 2019.

[116] Jerzy Filar and Koos Vrieze. *Competitive Markov decision processes*. Springer Science & Business Media, 2012.

[117] Arlington M Fink et al. Equilibrium in a stochastic n-person game. *Journal of science of the hiroshima university, series ai (mathematics)*, 28(1):89–93, 1964.

[118] Michael A Fishman. Involuntary defection and the evolutionary origins of empathy. *Journal of theoretical biology*, 242(4):873–879, 2006.

[119] Jeffrey A Fletcher and Martin Zwick. The evolution of altruism: Game theory in multilevel selection and inclusive fitness. *Journal of theoretical biology*, 245(1):26–36, 2007.

[120] Jakob Foerster, Yannis M Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2137–2145, 2016.

[121] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*, 2017.

[122] Jakob Foerster, Nantas Nardelli, Gregory Farquhar, Philip Torr, Pushmeet Kohli, Shimon Whiteson, et al. Stabilising experience replay for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1702.08887*, 2017.

[123] Jakob N. Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. *CoRR*, abs/1709.04326, 2017.

[124] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In McIlraith and Weinberger [274].

[125] Jakob N. Foerster, Nantas Nardelli, Gregory Farquhar, Triantafyllos Afouras, Philip H. S. Torr, Pushmeet Kohli, and Shimon Whiteson. Stabilising experience replay for deep multi-agent reinforcement learning. In Precup and Teh [343], pages 1146–1155.

[126] Maria Fox and David Poole, editors. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press, 2010.

[127] Michael C Frank and Noah D Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012.

[128] Erwin Frey. Evolutionary game theory: Theoretical concepts and applications to microbial communities. *Physica A: Statistical Mechanics and its Applications*, 389(20):4265–4298, 2010.

[129] John M Fryxell, Anna Mosser, Anthony RE Sinclair, and Craig Packer. Group formation stabilizes predator–prey dynamics. *Nature*, 449(7165):1041–1043, 2007.

[130] Masabumi Furuhata, Maged Dessouky, Fernando Ordóñez, Marc-Etienne Brunet, Xiaoqing Wang, and Sven Koenig. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57:28–46, 2013.

[131] Jr. G. P. Berman and Jr. F. M. Izrailev. Stability of nonlinear modes. *Physica D*, 88:445, 1983.

[132] Serge Galam and Serge Moscovici. Towards a theory of collective phenomena: consensus and attitude changes in groups. *European Journal of Social Psychology*, 21(1):49–74, 1991.

[133] Serge Galam and Bernard Walliser. Ising model versus normal form game. *Physica A: Statistical Mechanics and its Applications*, 389(3):481–489, 2010.

[134] Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

[135] Chris Gaskett. Reinforcement learning under circumstances beyond its control. 2003.

[136] Robin Gasser and Michael N Huhns. *Distributed Artificial Intelligence: Volume II*, volume 2. Morgan Kaufmann, 2014.

[137] Jean-François Gerard and Patrice Loisel. Spontaneous emergence of a relationship between habitat openness and mean group size and its possible evolutionary consequences in large herbivores. *Journal of theoretical biology*, 176(4):511–522, 1995.

[138] Michael E Gilpin. Do hares eat lynx? *The American Naturalist*, 107(957):727–730, 1973.

[139] Herbert Gintis. Classical versus evolutionary game theory. *Journal of Consciousness Studies*, 7(1-2):300–304, 2000.

[140] Herbert Gintis, Samuel Bowles, Robert Boyd, and Ernst Fehr. Explaining altruistic behavior in humans. *Evolution and Human Behavior*, 24(3):153–172, 2003.

[141] Piotr J Gmytrasiewicz and Prashant Doshi. A framework for sequential planning in multi-agent settings. *J. Artif. Intell. Res.(JAIR)*, 24:49–79, 2005.

[142] David González-Sánchez and Onésimo Hernández-Lerma. *Discrete–time stochastic control and dynamic potential games: the Euler–Equation approach*. Springer Science & Business Media, 2013.

[143] Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

[144] Ian Goodfellow, Nicolas Papernot, Sandy Huang, Yan Duan, Pieter Abbeel, and Jack Clark. Attacking machine learning with adversarial examples. *Open AI Blog*, 2017.

[145] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[146] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[147] Simon Goss, Serge Aron, Jean-Louis Deneubourg, and Jacques Marie Pasteels. Self-organized shortcuts in the argentine ant. *Naturwissenschaften*, 76(12):579–581, 1989.

[148] Carlos Gracia-Lazaro, Jesus Gomez-Gardenes, Luis Mario Floria, and Yamir Moreno. Intergroup information exchange drives cooperation in the public goods game. *Physical Review E*, 90(4):042808, 2014.

[149] Amy Greenwald, Keith Hall, and Roberto Serrano. Correlated q-learning. In *ICML*, volume 3, pages 242–249, 2003.

[150] Carlos Guestrin, Daphne Koller, and Ronald Parr. Multiagent planning with factored mdps. In *Advances in neural information processing systems*, pages 1523–1530, 2002.

[151] Carlos Guestrin, Michail Lagoudakis, and Ronald Parr. Coordinated reinforcement learning. In *ICML*, volume 2, pages 227–234. Citeseer, 2002.

[152] Mauro F Guillen. Business groups in emerging economies: A resource-based view. *academy of Management Journal*, 43(3):362–380, 2000.

[153] Xin Guo, Anran Hu, Renyuan Xu, and Junzi Zhang. Learning mean-field games. In *Advances in Neural Information Processing Systems*, pages 4966–4976, 2019.

[154] Jayesh K Gupta, Maxim Egorov, and Mykel Kochenderfer. Cooperative multi-agent control using deep reinforcement learning. In *AAMAS*, pages 66–83. Springer, 2017.

[155] Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors. *NIPS*, 2017.

[156] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018.

[157] James Hannan. Approximation to bayes risk in repeated play. *Contributions to the Theory of Games*, 3:97–139, 1957.

[158] Nikolaus Hansen, Sibylle D Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation*, 11(1):1–18, 2003.

[159] William Harms and Brian Skyrms. Evolution of moral norms. 2008.

[160] Sergiu Hart and Andreu Mas-Colell. A reinforcement procedure leading to correlated equilibrium. In *Economics Essays*, pages 181–200. Springer, 2001.

[161] Hado V Hasselt. Double q-learning. In *NIPS*, pages 2613–2621, 2010.

[162] Christoph Hauert. Spatial effects in social dilemmas. *Journal of Theoretical Biology*, 240(4):627–636, 2006.

[163] Christoph Hauert, Silvia De Monte, Josef Hofbauer, and Karl Sigmund. Replicator dynamics for optional public good games. *Journal of Theoretical Biology*, 218(2):187–194, 2002.

[164] Christoph Hauert, Franziska Michor, Martin A Nowak, and Michael Doebeli. Synergy and discounting of cooperation in social dilemmas. *Journal of theoretical biology*, 239(2):195–202, 2006.

[165] Kjell Hausken and Jack Hirshleifer. Truthful signalling, the heritability paradox, and the malthusian equi-marginal principle. *Theoretical population biology*, 73(1):11–23, 2008.

[166] Thomas Haynes and Sandip Sen. Evolving behavioral strategies in predators and prey. *Adaption and learning in multi-agent systems*, pages 113–126, 1996.

[167] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tieyan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828, 2016.

[168] He He and Jordan L. Boyd-Graber. Opponent modeling in deep reinforcement learning. In Balcan and Weinberger [21], pages 1804–1813.

[169] Matthias Heger. Consideration of risk in reinforcement learning. In *ICML*, pages 105–111, 1994.

[170] Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *ICML*, pages 805–813, 2015.

[171] Pablo Hernandez-Leal, Michael Kaisers, Tim Baarslag, and Enrique Munoz de Cote. A survey of learning in multiagent environments: Dealing with non-stationarity. *arXiv preprint arXiv:1707.09183*, 2017.

[172] Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 33(6):750–797, 2019.

[173] DJ Hoare, Iain D Couzin, J-GJ Godin, and J Krause. Context-dependent group size choice in fish. *Animal Behaviour*, 67(1):155–164, 2004.

[174] Josef Hofbauer and Karl Sigmund. *Evolutionary games and population dynamics*. Cambridge university press, 1998.

[175] Hans A Hofmann, Annaliese K Beery, Daniel T Blumstein, Iain D Couzin, Ryan L Earley, Loren D Hayes, Peter L Hurd, Eileen A Lacey, Steven M Phelps, Nancy G Solomon, et al. An evolutionary framework for studying mechanisms of social behavior. *Trends in ecology & evolution*, 29(10):581–589, 2014.

[176] C HolmesParker, M Taylor, Y Zhan, and K Tumer. Exploiting structure and agent-centric rewards to promote coordination in large multiagent systems. In *Adaptive and Learning Agents Workshop*, 2014.

[177] Vincent Hom and Joe Marks. Automatic design of balanced board games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, pages 25–30, 2007.

[178] Ronald A Howard. Dynamic programming and markov processes. 1960.

[179] Junling Hu and Michael P Wellman. Experimental results on q-learning for general-sum stochastic games. In *ICML*, pages 407–414. Morgan Kaufmann Publishers Inc., 2000.

[180] Junling Hu and Michael P Wellman. Nash q-learning for general-sum stochastic games. *Journal of Machine learning research*, 4(Nov):1039–1069, 2003.

[181] Junling Hu, Michael P Wellman, et al. Multiagent reinforcement learning: theoretical framework and an algorithm. In *ICML*, volume 98, pages 242–250. Citeseer, 1998.

[182] Minyi Huang, Roland P Malhamé, Peter E Caines, et al. Large population stochastic dynamic games: closed-loop mckean-vlasov systems and the nash certainty equivalence principle. *Communications in Information & Systems*, 6(3):221–252, 2006.

[183] Bernardo A Huberman and Natalie S Glance. Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences*, 90(16):7716–7718, 1993.

[184] Michael N Huhns. *Distributed Artificial Intelligence: Volume I*, volume 1. Elsevier, 2012.

[185] Robin Hunicke, Marc LeBlanc, and Robert Zubek. Mda: A formal approach to game design and game research. In *AAAI Workshop on Challenges in Game AI*, volume 4, page 1722, 2004.

[186] Peter L Hurd. Communication in discrete action-response games. *Journal of Theoretical Biology*, 174(2):217–222, 1995.

[187] Yoshinobu Inada and Keiji Kawachi. Order and flexibility in the motion of fish schools. *Journal of theoretical Biology*, 214(3):371–387, 2002.

[188] Christos C Ioannou, Vishwesha Guttal, and Iain D Couzin. Predatory fish select for coordinated collective motion in virtual prey. *Science*, 337(6099):1212–1215, 2012.

[189] Ernst Ising. Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik*, 31(1):253–258, 1925.

[190] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

[191] Tommi Jaakkola, Michael I Jordan, and Satinder P Singh. Convergence of stochastic iterative dynamic programming algorithms. In *NIPS*, pages 703–710, 1994.

[192] Max Jaderberg, Wojciech M Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castaneda, Charles Beattie, Neil C Rabinowitz, Ari S Morcos, Avraham Ruderman, et al. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019.

[193] Gerhard Jäger. Evolutionary stability conditions for signaling games with costly signals. *Journal of theoretical biology*, 253(1):131–141, 2008.

[194] Pieter Jan't Hoen, Karl Tuyls, Liviu Panait, Sean Luke, and Johannes A La Poutre. An overview of cooperative and competitive multiagent learning. In *International Workshop on Learning and Adaption in Multi-Agent Systems*, pages 1–46. Springer, 2005.

[195] Marco Alberto Javarone and Daniele Marinazzo. Evolutionary dynamics of group formation. *arXiv preprint arXiv:1612.03834*, 2016.

[196] Marco Alberto Javarone and Daniele Marinazzo. Evolutionary Dynamics of Group Formation. dec 2016.

[197] Robert L Jeanne. *Interindividual behavioral variability in social insects*. Westview Press, 1988.

[198] Seong Hoon Jeong, Ah Reum Kang, and Huy Kang Kim. Analysis of game bot's behavioral characteristics in social interaction networks of mmorpg. In

ACM SIGCOMM Computer Communication Review, volume 45, pages 99–100. ACM, 2015.

[199] Katerina V-A Johnson and Robin IM Dunbar. Pain tolerance predicts human social network size. *Scientific reports*, 6, 2016.

[200] M. P. Johnson, K. L. Miller, and K. Smith. personal communication, 1 May 2007.

[201] Samuel Johnson, Joaquín J Torres, J Marro, and Miguel A Munoz. Entropic origin of disassortativity in complex networks. *Physical review letters*, 104(10):108702, 2010.

[202] Leo P Kadanoff. More is the same; phase transitions and mean field theories. *Journal of Statistical Physics*, 137(5-6):777, 2009.

[203] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285, 1996.

[204] Tatsuya Kameda and Daisuke Nakanishi. Does social/cultural learning increase human adaptability?: Rogers's question revisited. *Evolution and Human Behavior*, 24(4):242–260, 2003.

[205] Albert B Kao and Iain D Couzin. Decision accuracy in complex environments is often maximized by small group sizes. In *Proc. R. Soc. B*, volume 281, page 20133305. The Royal Society, 2014.

[206] Albert B Kao, Noam Miller, Colin Torney, Andrew Hartnett, and Iain D Couzin. Collective learning and optimal consensus decisions in social animal groups. *PLoS Comput Biol*, 10(8):e1003762, 2014.

[207] Spiros Kapetanakis and Daniel Kudenko. Reinforcement learning of coordination in cooperative multi-agent systems. In *NCAI*, pages 326–331, Menlo Park, CA, USA, 2002.

[208] Stuart A Kauffman. *The origins of order: Self-organization and selection in evolution*. Oxford University Press, USA, 1993.

[209] S. R. Kawa and S.-J. Lin. *J. Geophys. Res.*, 108(D6):4201, 2003. DOI:10.1029/2002JD002268.

[210] Michael Kearns. Graphical games. *Algorithmic game theory*, 3:159–180, 2007.

[211] Michael Kearns, Michael L Littman, and Satinder Singh. Graphical models for game theory. *arXiv preprint arXiv:1301.2281*, 2013.

[212] Jeremy Kendal, Marcus W Feldman, and Kenichi Aoki. Cultural coevolution of norm adoption and enforcement when punishers are rewarded or non-punishers are punished. *Theoretical Population Biology*, 70(1):10–25, 2006.

[213] James Kennedy. Swarm intelligence. In *Handbook of nature-inspired and innovative computing*, pages 187–219. Springer, 2006.

[214] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[215] Philip Kitcher and Brian Skyrms. Games social animals play: Commentary on brian skyrms's evolution of the social contract, 1999.

[216] A Harry Klopf. *Brain function and adaptive systems: a heterostatic theory.* Number 133. Air Force Cambridge Research Laboratories, Air Force Systems Command, United . . . , 1972.

[217] Jill C. Knvth. The programming of computer art. Vernier Art Center, Stanford, California, February 1988. A full BOOKLET entry.

[218] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

[219] Jelle R Kok and Nikos Vlassis. Sparse cooperative q-learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 61, 2004.

[220] Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pages 1008–1014, 2000.

[221] Vijay R. Konda and John N. Tsitsiklis. Actor-critic algorithms. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 1008–1014. MIT Press, 2000.

[222] Amanda H Korstjens, Ingrid Lugo Verhoeckx, and Robin IM Dunbar. Time as a constraint on group size in spider monkeys. *Behavioral Ecology and Sociobiology*, 60(5):683, 2006.

[223] Dennis Krebs. Evolutionary games and morality. *Journal of Consciousness Studies*, 7(1-2):313–321, 2000.

[224] Erwin Kreyszig. *Introductory functional analysis with applications*, volume 1. wiley New York, 1978.

[225] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[226] Harold W Kuhn. Extensive games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(10):570, 1950.

[227] Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *arXiv preprint arXiv:1207.6083*, 2012.

[228] H Cai W Zhang J Wang Y Yu L Zheng, J Yang. Magent: A many-agent reinforcement learning research platform for artificial collective intelligence. *Advances in Neural Information Processing Systems Demonstration*, 2017.

[229] Quang Duy Lã, Yong Huat Chew, and Boon-Hee Soong. *Potential Game Theory*. Springer, 2016.

[230] Sascha Lange, Thomas Gabel, and Martin Riedmiller. Batch reinforcement learning. In *Reinforcement learning*, pages 45–73. Springer, 2012.

[231] Jean-Michel Lasry and Pierre-Louis Lions. Mean field games. *Japanese journal of mathematics*, 2(1):229–260, 2007.

[232] Martin Lauer and Martin Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *ICML*. Citeseer, 2000.

[233] Guillaume J Laurent, Laëtitia Matignon, Le Fort-Piat, et al. The world of independent learners is not markovian. *International Journal of Knowledge-based and Intelligent Engineering Systems*, 15(1):55–64, 2011.

[234] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. *CoRR*, abs/1612.07182, 2016.

[235] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. *CoRR*, abs/1612.07182, 2016.

[236] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[237] Der-Horng Lee, Hao Wang, Ruey Cheu, and Siew Teo. Taxi dispatch system based on current demands and real-time traffic conditions. *Transportation Research Record: Journal of the Transportation Research Board*, (1882):193–200, 2004.

[238] Jeho Lee, Kyungmook Lee, and Sangkyu Rho. An evolutionary perspective on strategic group emergence: a genetic algorithm-based model. *Strategic Management Journal*, 23(8):727–746, 2002.

[239] Junghoon Lee, Gyung-Leen Park, Hanil Kim, Young-Kyu Yang, Pankoo Kim, and Sang-Wook Kim. A telematics service system based on the linux cluster. In Yong Shi, Geert Dick van Albada, Jack Dongarra, and Peter M. A. Sloot, editors, *Computational Science – ICCS 2007*, pages 660–667, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[240] Julia Lehmann and RIM Dunbar. Network cohesion, group size and neocortex size in female-bonded old world primates. *Proceedings of the Royal Society of London B: Biological Sciences*, page rspb20091409, 2009.

[241] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. Multi-agent reinforcement learning in sequential social dilemmas. *arXiv preprint arXiv:1702.03037*, 2017.

[242] Carlton E Lemke and Joseph T Howson, Jr. Equilibrium points of bimatrix games. *Journal of the Society for Industrial and Applied Mathematics*, 12(2):413–423, 1964.

[243] David S Leslie and Edmund J Collins. Generalised weakened fictitious play. *Games and Economic Behavior*, 56(2):285–298, 2006.

[244] David S Leslie, EJ Collins, et al. Convergent multiple-timescales reinforcement learning algorithms in normal form games. *The Annals of Applied Probability*, 13(4):1231–1251, 2003.

[245] Richard C Lewontin. Evolution and the theory of games. *Journal of theoretical biology*, 1(3):382–403, 1961.

[246] Kevin Leyton-Brown and Moshe Tennenholtz. Local-effect games. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2005.

[247] Shihui Li, Yi Wu, Xinyue Cui, Honghua Dong, Fei Fang, and Stuart Russell. Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4213–4220, 2019.

[248] Yuxi Li. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, 2017.

[249] Ziru Li, Yili Hong, and Zhongju Zhang. An empirical analysis of on-demand ride sharing and traffic congestion. In *2016 International Conference on Information Systems, ICIS 2016*. Association for Information Systems, 2016.

[250] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[251] Kaixiang Lin, Renyu Zhao, Zhe Xu, and Jiayu Zhou. Efficient large-scale fleet management via multi-agent deep reinforcement learning. *arXiv preprint arXiv:1802.06444*, 2018.

[252] Tianyi Lin, Chi Jin, and Michael I Jordan. On gradient descent ascent for nonconvex-concave minimax problems. *arXiv preprint arXiv:1906.00331*, 2019.

[253] Daniel D. Lincoll. Semigroups of recurrences. In David J. Lipcoll, D. H. Lawrie, and A. H. Sameh, editors, *High Speed Computer and Algorithm Organization*, number 23 in Fast Computers, part 3, pages 179–183. Academic Press, New York, third edition, September 1977. A full INCOLLECTION entry.

[254] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the eleventh international conference on machine learning*, volume 157, pages 157–163, 1994.

[255] Michael L Littman. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pages 322–328, 2001.

[256] Michael L Littman. Value-function reinforcement learning in markov games. *Cognitive Systems Research*, 2(1):55–66, 2001.

[257] Michael L Littman and Peter Stone. A polynomial-time nash equilibrium algorithm for repeated games. *Decision Support Systems*, 39(1):55–66, 2005.

[258] Boyi Liu, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural proximal/trust region policy optimization attains globally optimal policy. *arXiv preprint arXiv:1906.10306*, 2019.

[259] Alfred J Lotka. Elements of physical biology. 1925.

[260] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017.

[261] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In Guyon et al. [155], pages 6382–6393.

[262] Sergio Valcarcel Macua, Javier Zazo, and Santiago Zazo. Learning parametric closed-loop policies for markov potential games. In *International Conference on Learning Representations*, 2018.

[263] Sridhar Mahadevan. Average reward reinforcement learning: Foundations, algorithms, and empirical results. *Machine learning*, 22(1-3):159–195, 1996.

[264] Kleanthis Malialis, Sam Devlin, and Daniel Kudenko. Resource abstraction for reinforcement learning in multiagent congestion problems. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, AAMAS '16, pages 503–511, Richland, SC, 2016. International Foundation for Autonomous Agents and Multiagent Systems.

[265] Kleanthis Malialis, Jun Wang, Gary Brooks, and George Frangou. Feature selection as a multiagent coordination problem. *CoRR*, abs/1603.05152, 2016.

[266] Larry Manmaker. *The Definitive Computer Manual*. Chips-R-Us, Silicon Valley, silver edition, 1986. A full MANUAL entry.

[267] Eric Maskin and Jean Tirole. Markov perfect equilibrium: I. observable actions. *Journal of Economic Theory*, 100(2):191–219, 2001.

[268] Édouard Masterly. Mastering thesis writing. Master's project, Stanford University, English Department, June-August 1988. A full MASTERSTHESIS entry.

[269] Laëtitia Matignon, Guillaume J Laurent, and Nadine Le Fort-Piat. Hysteretic q-learning: an algorithm for decentralized reinforcement learning in cooperative multi-agent teams. In *Intelligent Robots and Systems, 2007. IROS 2007.*, pages 64–69. IEEE, 2007.

[270] Laetitia Matignon, Guillaume J Laurent, and Nadine Le Fort-Piat. Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems. 2012.

[271] Eric Mazumdar, Lillian J Ratliff, Shankar Sastry, and Michael I Jordan. Policy gradient in linear quadratic dynamic games has no convergence guarantees. Smooth Games Optimization and Machine Learning Workshop: Bridging Game . . . , 2019.

[272] Eric V Mazumdar, Michael I Jordan, and S Shankar Sastry. On finding local nash equilibria (and only local nash equilibria) in zero-sum games. *arXiv preprint arXiv:1901.00838*, 2019.

[273] David A. McAllester and Petri Myllymäki, editors. *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008*. AUAI Press, 2008.

[274] Sheila A. McIlraith and Kilian Q. Weinberger, editors. *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018*. AAAI Press, 2018.

[275] Francisco S Melo, Sean P Meyn, and M Isabel Ribeiro. An analysis of reinforcement learning with function approximation. In *Proceedings of the 25th international conference on Machine learning*, pages 664–671. ACM, 2008.

[276] David Mguni. Stochastic potential games. *arXiv preprint arXiv:2005.13527*, 2020.

[277] Richard E Michod. The group covariance effect and fitness trade-offs during evolutionary transitions in individuality. *Proceedings of the National Academy of Sciences*, 103(24):9113–9117, 2006.

[278] Marvin Minsky. Steps toward artificial intelligence. *Proceedings of the IRE*, 49(1):8–30, 1961.

[279] Marvin Lee Minsky. *Theory of neural-analog reinforcement systems and its application to the brain model problem*. Princeton University., 1954.

[280] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[281] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, pages 1928–1937, 2016.

[282] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[283] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[284] Dov Monderer and Lloyd S Shapley. Potential games. *Games and economic behavior*, 14(1):124–143, 1996.

[285] Andrew W Moore and Christopher G Atkeson. Prioritized sweeping: Reinforcement learning with less data and less time. *Machine learning*, 13(1):103–130, 1993.

[286] Matej Moravcík, Martin Schmid, Neil Burch, Viliam Lisỳ, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael

Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.
[287] Igor Mordatch and Pieter Abbeel. Emergence of Grounded Compositional Language in Multi-Agent Populations. mar 2017.
[288] Koichiro Morihiro, Teijiro Isokawa, Haruhiko Nishimura, and Nobuyuki Matsui. Emergence of flocking behavior based on reinforcement learning. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 699–706. Springer, 2006.
[289] Koichiro Morihiro, Nobuyuki Matsui, Teijiro Isokawa, and Haruhiko Nishimura. Reinforcement learning scheme for grouping and characterization of multi-agent network. *Knowledge-Based and Intelligent Information and Engineering Systems*, pages 592–601, 2010.
[290] Jörg P Müller and Klaus Fischer. Application impact of multi-agent systems and technologies: A survey. In *Agent-oriented software engineering*, pages 27–53. Springer, 2014.
[291] Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(May):815–857, 2008.
[292] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.
[293] Wataru Nakahashi. The evolution of conformist transmission in social learning when the environment changes periodically. *Theoretical population biology*, 72(1):52–66, 2007.
[294] Taewoo Nam and Theresa A Pardo. Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of the 12th annual international digital government research conference: digital government innovation in challenging times*, pages 282–291. ACM, 2011.
[295] John Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.
[296] Ashutosh Nayyar, Aditya Mahajan, and Demosthenis Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58(7):1644–1658, 2013.
[297] Zoltán Néda, Erzsébet Ravasz, Tamás Vicsek, Yves Brechet, and Albert-Lázló Barabási. Physics of the rhythmic applause. *Physical Review E*, 61(6):6987, 2000.
[298] Angelia Nedic, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.
[299] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.
[300] J. Nelson. U.S. Patent No. 5,693,000, 12 December 2005.
[301] J. Nelson. TWI Report 666/1999, January 1999. Required institution missing.
[302] J. K. Nelson. M.S. thesis, New York University, 1999.
[303] J v Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.
[304] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[305] Thanh Thi Nguyen, Ngoc Duy Nguyen, and Saeid Nahavandi. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE transactions on cybernetics*, 2020.

[306] Gregoire Nicolis, Ilya Prigogine, et al. *Self-organization in nonequilibrium systems*, volume 191977. Wiley, New York, 1977.

[307] Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[308] Hiro-Sato Niwa. Self-organizing dynamic model of fish schooling. *Journal of theoretical Biology*, 171(2):123–136, 1994.

[309] Martin A Nowak, Joshua B Plotkin, and David C Krakauer. The evolutionary language game. *Journal of Theoretical Biology*, 200(2):147–162, 1999.

[310] Ann Nowé, Peter Vrancx, and Yann-Michaël De Hauwere. Game theory and multi-agent reinforcement learning. In *Reinforcement Learning*, pages 441–470. Springer, 2012.

[311] Alfred V. Oaho, Jeffrey D. Ullman, and Mihalis Yannakakis. On notions of information transfer in VLSI circuits. In Wizard V. Oz and Mihalis Yannakakis, editors, *Proc. Fifteenth Annual ACM*, number 17 in All ACM Conferences, pages 133–139, New York, March 1983. ACM, Academic Press. A full INPROCEDINGS entry.

[312] Frans A Oliehoek, Christopher Amato, et al. *A concise introduction to decentralized POMDPs*, volume 1. Springer, 2016.

[313] Megan M Olsen and Rachel Fraczkowski. Co-evolution in predator prey through reinforcement learning. *Journal of Computational Science*, 9:118–124, 2015.

[314] Shayegan Omidshafiei, Jason Pazis, Christopher Amato, Jonathan P. How, and John Vian. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In Precup and Teh [343], pages 2681–2690.

[315] Norihiko Ono and Kenji Fukumoto. Multi-agent reinforcement learning: A modular approach. In *Second International Conference on Multiagent Systems*, pages 252–258, 1996.

[316] W. Opechowski and R. Guccione. *Introduction to the Theory of Normal Metals*, volume IIa, page 105. Academic Press, New York, 1965.

[317] W. Opechowski and R. Guccione. Introduction to the theory of normal metals. In G. T. Rado and H. Suhl, editors, *Magnetism*, volume IIa, page 105. Academic Press, New York, 1965.

[318] W. Opechowski and R. Guccione. Introduction to the theory of normal metals. In G. T. Rado and H. Suhl, editors, *Magnetism*, volume IIa, page 105, New York, 1965. Academic Press.

[319] Afshin OroojlooyJadid and Davood Hajinezhad. A review of cooperative multi-agent deep reinforcement learning. *arXiv preprint arXiv:1908.03963*, 2019.

[320] Wizard V. Oz and Mihalis Yannakakis, editors. *Proc. Fifteenth Annual*, number 17 in All ACM Conferences, Boston, March 1983. ACM, Academic Press. A full PROCEEDINGS entry.

[321] Jakub Pachocki, Greg Brockman, Jonathan Raiman, Susan Zhang, Henrique

Pondé, Jie Tang, Filip Wolski, Christy Dennison, Rafal Jozefowicz, Przemyslaw Debiak, et al. Openai five, 2018. *URL https://blog. openai. com/openai-five*, 2018.

[322] Karen M Page and Martin A Nowak. Empathy leads to fairness. *Bulletin of mathematical biology*, 64(6):1101–1116, 2002.

[323] Ada Palmer. *Reading Lucretius in the Renaissance*, volume 16. Harvard University Press, 2014.

[324] Liviu Panait and Sean Luke. Cooperative multi-agent learning: The state of the art. *Autonomous agents and multi-agent systems*, 11(3):387–434, 2005.

[325] Tanya Pankiw and Robert E Page Jr. Response thresholds to sucrose predict foraging division of labor in honeybees. *Behavioral Ecology and Sociobiology*, 47(4):265–267, 2000.

[326] Christos H Papadimitriou and John N Tsitsiklis. The complexity of markov decision processes. *Mathematics of operations research*, 12(3):441–450, 1987.

[327] Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V Albrecht. Comparative evaluation of multi-agent deep reinforcement learning algorithms. *arXiv preprint arXiv:2006.07869*, 2020.

[328] Julia K Parrish and Leah Edelstein-Keshet. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science*, 284(5411):99–101, 1999.

[329] Santiago Pascual, Antonio Bonafonte, and Joan Serrà. Segan: Speech enhancement generative adversarial network. *arXiv preprint arXiv:1703.09452*, 2017.

[330] Christina Pawlowitsch. Finite populations choose an optimal language. *Journal of Theoretical Biology*, 249(3):606–616, 2007.

[331] Peng Peng, Quan Yuan, Ying Wen, Yaodong Yang, Zhenkun Tang, Haitao Long, and Jun Wang. Multiagent bidirectionally-coordinated nets for learning to play starcraft combat games. *arXiv preprint arXiv:1703.10069*, 2017.

[332] Peng Peng, Quan Yuan, Ying Wen, Yaodong Yang, Zhenkun Tang, Haitao Long, and Jun Wang. Multiagent bidirectionally-coordinated nets for learning to play starcraft combat games. *CoRR*, abs/1703.10069, 2017.

[333] Alan Penn. The complexity of the elementary interface: shopping space. In *Proceedings to the 5th International Space Syntax Symposium*, volume 1, pages 25–42. Akkelies van Nes, 2005.

[334] Matjav Perc, Jesús Gómez-Gardeñes, Attila Szolnoki, Luis M Floría, and Yamir Moreno. Evolutionary dynamics of group interactions on structured populations: a review. *Journal of the royal society interface*, 10(80):20120997, 2013.

[335] Julien Perolat, Bilal Piot, and Olivier Pietquin. Actor-critic fictitious play in simultaneous move multistage games. 2018.

[336] Julien Pérolat, Florian Strub, Bilal Piot, and Olivier Pietquin. Learning nash equilibrium for general-sum markov games from batch data. *arXiv preprint arXiv:1606.08718*, 2016.

[337] Jan Peters and Stefan Schaal. Natural actor-critic. *Neurocomputing*, 71(7-9):1180–1190, 2008.

[338] F. Phidias Phony-Baloney. *Fighting Fire with Fire: Festooning French Phrases*. PhD dissertation, Fanstord University, Department of French, June-August 1988. A full PHDTHESIS entry.

[339] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. *arXiv preprint arXiv:1703.02702*, 2017.

[340] Matthias Plappert. keras-rl, 2016.

[341] Rob Powers and Yoav Shoham. Learning against opponents with bounded memory. In *IJCAI*, volume 5, pages 817–822, 2005.

[342] Rob Powers and Yoav Shoham. New criteria and a new algorithm for learning in multi-agent systems. In *Advances in neural information processing systems*, pages 1089–1096, 2005.

[343] Doina Precup and Yee Whye Teh, editors. *ICML*, volume 70. PMLR, 2017.

[344] B. Quinn, editor. *Proceedings of the 2003 Particle Accelerator Conference, Portland, OR, 12-16 May 2005*, New York, 2001. Wiley. Albeit the conference was held in 2005, it was the 2003 conference, and the proceedings were published in 2001; go figure.

[345] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304, 2018.

[346] Stephan G Reebs. Can a minority of informed leaders determine the foraging movements of a fish shoal? *Animal behaviour*, 59(2):403–409, 2000.

[347] Steffen Rendle. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(3):57, 2012.

[348] Craig W Reynolds. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH computer graphics*, 21(4):25–34, 1987.

[349] Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pages 317–328. Springer, 2005.

[350] Alan R Rogers. Does biology constrain culture? *American Anthropologist*, 90(4):819–831, 1988.

[351] Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48:67–113, 2013.

[352] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, pages 2226–2234, 2016.

[353] Paul A Samuelson. Generalized predator-prey oscillations in ecological and economic equilibrium. *Proceedings of the National Academy of Sciences*, 68(5):980–983, 1971.

[354] Angel Sánchez and José A Cuesta. Altruism may arise from individual selection. *Journal of Theoretical Biology*, 235(2):233–240, 2005.

[355] Tuomas W Sandholm and Robert H Crites. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems*, 37(1-2):147–166, 1996.

[356] Jurgen Schmidhuber. A general method for multi-agent reinforcement learning in unrestricted environments. In *Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium*, pages 84–87, 1996.

[357] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[358] Erwin Schrodinger. *What is life?* University Press: Cambridge, 1943.

[359] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897, 2015.

[360] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[361] Thomas D Seeley. *The wisdom of the hive: the social physiology of honey bee colonies*. Harvard University Press, 2009.

[362] Reinhard Selten. Spieltheoretische behandlung eines oligopolmodells mit nachfrageträgheit: Teil i: Bestimmung des dynamischen preisgleichgewichts. *Zeitschrift für die gesamte Staatswissenschaft/Journal of Institutional and Theoretical Economics*, (H. 2):301–324, 1965.

[363] Kiam Tian Seow, Nam Hai Dang, and Der-Horng Lee. A collaborative multi-agent taxi-dispatch system. *IEEE Transactions on Automation Science and Engineering*, 7(3):607–616, 2010.

[364] Elhadi M Shakshuki and Malcolm Reid. Multi-agent system applications in healthcare: current technology and future roadmap. In *ANT/SEIT*, pages 252–261, 2015.

[365] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[366] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

[367] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

[368] Lloyd S Shapley. A note on the lemke-howson algorithm. In *Pivoting and Extension*, pages 175–189. Springer, 1974.

[369] Paul Shepard. *Thinking animals: Animals and the development of human intelligence*. University of Georgia Press, 1978.

[370] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[371] Yoav Shoham, Rob Powers, and Trond Grenager. If multi-agent learning is the answer, what is the question? *Artificial intelligence*, 171(7):365–377, 2007.

[372] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[373] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

[374] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[375] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *ICML*, pages 387–395, 2014.

[376] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

[377] Satinder Singh, Tommi Jaakkola, Michael L Littman, and Csaba Szepesvári. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine learning*, 38(3):287–308, 2000.

[378] J. Smith. *Proc. SPIE*, 124:367, 2007. Required title is missing.

[379] J. Smith, editor. *AIP Conf. Proc.*, volume 841, 2007.

[380] J. M. Smith. In C. Brown, editor, *Molecular Dynamics*. Academic, New York, 1980.

[381] J. M. Smith. *Molecular Dynamics*. Academic, New York, 1980.

[382] J. S. Smith and G. W. Johnson. *Philos. Trans. R. Soc. London, Ser. B*, 777:1395, 2005.

[383] R. Smith. Hummingbirds are our friends. *J. Appl. Phys. (these proceedings)*, 2001. Abstract No. DA-01.

[384] S. M. Smith. Ph.D. thesis, Massachusetts Institute of Technology, 2003.

[385] V. K. Smith, K. Johnson, and M. O. Klein. Surface chemistry and preferential crystal orientation on a silicon surface. J. Appl. Phys. (submitted), 2010.

[386] W. J. Smith, T. J. Johnson, and B. G. Miller. Surface chemistry and preferential crystal orientation on a silicon surface. J. Appl. Phys. (unpublished), 2010.

[387] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 5887–5896, 2019.

[388] D Sonnette. Critical phenomena in natural science, 2003.

[389] H Eugene Stanley. Phase transitions and critical phenomena. *Clarendon, Oxford*, 9, 1971.

[390] Peter Stone. Multiagent learning is not the answer. it is the question. *Artificial Intelligence*, 171(7):402–405, 2007.

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

[392] Cédric Sueur, Jean-Louis Deneubourg, Odile Petit, and Iain D Couzin. Group size, grooming and fission in primates: A modeling approach based on group structure. *Journal of Theoretical Biology*, 273(1):156–166, 2011.

[393] Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, pages 2244–2252, 2016.

[394] David JT Sumpter. The principles of collective animal behaviour. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 361(1465):5–22, 2006.

[395] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *AAMAS*, pages 2085–2087, 2018.

[396] Wesley Suttle, Zhuoran Yang, Kaiqing Zhang, Zhaoran Wang, Tamer Basar, and Ji Liu. A multi-agent off-policy actor-critic algorithm for distributed reinforcement learning. *arXiv preprint arXiv:1903.06372*, 2019.

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

[398] Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *ICML*, pages 216–224, 1990.

[399] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[400] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000.

[401] Richard S Sutton, David A McAllester, Satinder P Singh, Yishay Mansour, et al. Policy gradient methods for reinforcement learning with function approximation. In *NIPS*, volume 99, pages 1057–1063, 1999.

[402] Umar Syed, Michael Bowling, and Robert E Schapire. Apprenticeship learning using linear programming. In *Proceedings of the 25th international conference on Machine learning*, pages 1032–1039, 2008.

[403] Csaba Szepesvári. Algorithms for reinforcement learning. *Synthesis lectures on artificial intelligence and machine learning*, 4(1):1–103, 2010.

[404] Csaba Szepesvári and Michael L Littman. A unified analysis of value-function-based reinforcement-learning algorithms. *Neural computation*, 11(8):2017–2060, 1999.

[405] Daniel Szer, François Charpillet, and Shlomo Zilberstein. Maa*: A heuristic search algorithm for solving decentralized pomdps. 2005.

[406] Aviv Tamar, Huan Xu, and Shie Mannor. Scaling up robust mdps by reinforcement learning. *arXiv preprint arXiv:1306.6189*, 2013.

[407] Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. Multiagent cooperation and competition with deep reinforcement learning. *PloS One*, 12(4):e0172395, 2017.

[408] Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pages 330–337, 1993.

[409] Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(7), 2009.

[410] Peter D Taylor and Leo B Jonker. Evolutionary stable strategies and game dynamics. *Mathematical biosciences*, 40(1-2):145–156, 1978.

[411] Hamidou Tembine. Mean field stochastic games: convergence, q/h-learning and optimality. In *American Control Conference (ACC), 2011*, pages 2423–2428. IEEE, 2011.

[412] Hamidou Tembine, Raul Tempone, and Pedro Vilanova. Mean-field learning: a survey. *arXiv preprint arXiv:1210.4657*, 2012.

[413] Tom Térrific. An $O(n \log n / \log \log n)$ sorting algorithm. Wishful Research Result 7, Fanstord University, Computer Science Department, Fanstord, California, October 1988. A full TECHREPORT entry.

[414] Gerald Tesauro. Temporal difference learning and td-gammon. *Communications of the ACM*, 38(3):58–68, 1995.
[415] Gerald Tesauro. Extending q-learning to general adaptive multi-agent systems. In *NIPS*, pages 871–878, 2004.
[416] Edward L Thorndike. Animal intelligence: an experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, 2(4):i, 1898.
[417] Randy Thornhill. Sexual selection and paternal investment in insects. *The American Naturalist*, 110(971):153–163, 1976.
[418] Julian Togelius and Jurgen Schmidhuber. An experiment in automatic game design. In *Computational Intelligence and Games, 2008. CIG'08. IEEE Symposium On*, pages 111–118. IEEE, 2008.
[419] Marc Toussaint, Laurent Charlin, and Pascal Poupart. Hierarchical pomdp controller optimization by likelihood maximization. In *UAI*, volume 24, pages 562–570, 2008.
[420] Robert L Trivers. The evolution of reciprocal altruism. *The Quarterly review of biology*, 46(1):35–57, 1971.
[421] Carol Alvarez Troy. Envisioning stock trading where the brokers are bots. *New York Times*, 16, 1997.
[422] Karl Tuyls and Ann Nowé. Evolutionary game theory and multi-agent reinforcement learning. 2005.
[423] Karl Tuyls and Simon Parsons. What evolutionary game theory tells us about multiagent learning. *Artificial Intelligence*, 171(7):406–416, 2007.
[424] Karl Tuyls and Gerhard Weiss. Multiagent learning: Basics, challenges, and prospects. *Ai Magazine*, 33(3):41–41, 2012.
[425] Ulrich Ünderwood, Ned Ñet, and Paul P̄ot. Lower bounds for wishful research results. Talk at Fanstord University (A full UNPUBLISHED entry), November, December 1988.
[426] Nicolas Usunier, Gabriel Synnaeve, Zeming Lin, and Soumith Chintala. Episodic exploration for deep deterministic policies: An application to starcraft micromanagement tasks. *arXiv preprint arXiv:1609.02993*, 2016.
[427] William Uther and Manuela Veloso. Adversarial reinforcement learning. Technical report, Technical report, Carnegie Mellon University, 1997. Unpublished, 1997.
[428] J v. Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.
[429] Johannes van der Wal, Johannes van der Wal, Johannes van der Wal, Pays-Bas Mathématicien, Johannes van der Wal, and Netherlands Mathematician. *Stochastic Dynamic Programming: successive approximations and nearly optimal strategies for Markov decision processes and Markov games*. Mathematisch centrum, 1981.
[430] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.
[431] Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, Alireza Makhzani, Heinrich Küttler, John Agapiou, Julian Schrittwieser, et al. Starcraft ii: A new challenge for reinforcement learning. *arXiv preprint arXiv:1708.04782*, 2017.

[432] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. Princeton University Press Princeton, NJ, 1945.

[433] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.

[434] Joe Yuichiro Wakano and Kenichi Aoki. A mixed strategy model for the emergence and intensification of social learning in a periodically changing natural environment. *Theoretical population biology*, 70(4):486–497, 2006.

[435] Joe Yuichiro Wakano, Kenichi Aoki, and Marcus W Feldman. Evolution of social learning: a mathematical analysis. *Theoretical population biology*, 66(3):249–258, 2004.

[436] Jun Wang, Weinan Zhang, Shuai Yuan, et al. Display advertising with real-time bidding (rtb) and behavioural targeting. *Foundations and Trends® in Information Retrieval*, 11(4-5):297–435, 2017.

[437] Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural policy gradient methods: Global optimality and rates of convergence. In *International Conference on Learning Representations*, 2019.

[438] Sida I Wang, Percy Liang, and Christopher D Manning. Learning language games through interaction. *arXiv preprint arXiv:1606.02447*, 2016.

[439] Xiaofeng Wang and Tuomas Sandholm. Reinforcement learning to play an optimal nash equilibrium in team markov games. In *Advances in neural information processing systems*, pages 1603–1610, 2003.

[440] Zhaodong Wang, Zhiwei Qin, Xiaocheng Tang, Jieping Ye, and Hongtu Zhu. Deep reinforcement learning with knowledge transfer for online rides order dispatching. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 617–626. IEEE, 2018.

[441] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[442] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[443] Lex Weaver and Nigel Tao. The optimal reward baseline for gradient-based reinforcement learning. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 538–545, 2001.

[444] Anja Weidenmüller. The control of nest climate in bumblebee (bombus terrestris) colonies: interindividual variability and self reinforcement in fanning response. *Behavioral Ecology*, 15(1):120–128, 2004.

[445] Gabriel Y Weintraub, Lanier Benkard, and Benjamin Van Roy. Oblivious equilibrium: A mean field approximation for large-scale dynamic games. In *Advances in neural information processing systems*, pages 1489–1496, 2006.

[446] Gerhard Weiss. *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT press, 1999.

[447] Ying Wen, Yaodong Yang, Rui Luo, and Jun Wang. Modelling bounded rationality in multi-agent interactions by generalized recursive reasoning. *IJCAI*, 2019.

[448] Ying Wen, Yaodong Yang, Rui Luo, Jun Wang, and Wei Pan. Probabilistic recursive reasoning for multi-agent reinforcement learning. In *International Conference on Learning Representations*, 2018.

[449] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[450] Edward O Wilson et al. The insect societies. *The insect societies.*, 1971.

[451] Edward Witten, 2001.
[452] Michael Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.
[453] Feng Wu, Shlomo Zilberstein, and Xiaoping Chen. Rollout sampling policy iteration for decentralized pomdps. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pages 666–673, 2010.
[454] Feng Wu, Shlomo Zilberstein, and Nicholas R Jennings. Monte-carlo expectation maximization for decentralized pomdps. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
[455] Zhe Xu, Zhixin Li, Qingwen Guan, Dingshui Zhang, Qiang Li, Junxiao Nan, Chunyang Liu, Wei Bian, and Jieping Ye. Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, pages 905–913, New York, NY, USA, 2018. ACM.
[456] Yuichi Yabu, Makoto Yokoo, and Atsushi Iwasaki. Multiagent planning with trembling-hand perfect equilibrium in multiagent pomdps. In *Pacific Rim International Conference on Multi-Agents*, pages 13–24. Springer, 2007.
[457] Jiachen Yang, Xiaojing Ye, Rakshit Trivedi, Huan Xu, and Hongyuan Zha. Deep mean field games for learning optimal behavior policy of large populations. *CoRR*, abs/1711.03156, 2017.
[458] Jiachen Yang, Xiaojing Ye, Rakshit Trivedi, Huan Xu, and Hongyuan Zha. Learning deep mean field games for modeling large population behavior. In *International Conference on Learning Representations*, 2018.
[459] Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 5571–5580, 2018.
[460] Yaodong Yang, Ying Wen, Lihuan Chen, Jun Wang, Kun Shao, David Mguni, and Weinan Zhang. Multi-agent determinantal q-learning. 2020.
[461] Yaodong Yang, Lantao Yu, Yiwei Bai, Jun Wang, Weinan Zhang, Ying Wen, and Yong Yu. An empirical study of AI population dynamics with million-agent reinforcement learning. *arXiv preprint arXiv:1709.04511*, 2017.
[462] Zhuora Yang, Yuchen Xie, and Zhaoran Wang. A theoretical analysis of deep q-learning. *arXiv preprint arXiv:1901.00137*, 2019.
[463] Bora Yongacoglu, Gürdal Arslan, and Serdar Yüksel. Learning team-optimality for decentralized stochastic control and dynamic games. *arXiv preprint arXiv:1903.05812*, 2019.
[464] H Peyton Young. The evolution of conventions. *Econometrica: Journal of the Econometric Society*, pages 57–84, 1993.
[465] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858, 2017.
[466] V. E. Zakharov and A. B. Shabat. Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media. *Zh. Eksp. Teor. Fiz.*, 61:118–134, 1971.
[467] Y. M. Zalkins. e-print arXiv:cond-mat/040426, 2008.
[468] Santiago Zazo, Sergio Valcarcel Macua, Matilde Sánchez-Fernández, and Javier Zazo. Dynamic potential games in communications: Fundamentals and applications. *arXiv*, pages arXiv–1509, 2015.

[469] Chongjie Zhang and Victor R. Lesser. Multi-agent learning with policy prediction. In Fox and Poole [126].

[470] Kaiqing Zhang, Zhuoran Yang, and Tamer Basar. Networked multi-agent reinforcement learning in continuous spaces. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2771–2776. IEEE, 2018.

[471] Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635*, 2019.

[472] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Başar. Finite-sample analysis for decentralized batch multi-agent reinforcement learning with networked agents. *arXiv preprint arXiv:1812.02783*, 2018.

[473] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pages 5872–5881, 2018.

[474] Lingyu Zhang, Tao Hu, Yue Min, Guobin Wu, Junying Zhang, Pengcheng Feng, Pinghua Gong, and Jieping Ye. A taxi order dispatch model based on combinatorial optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, pages 2151–2159, New York, NY, USA, 2017. ACM.

[475] Yan Zhang and Michael M Zavlanos. Distributed off-policy actor-critic reinforcement learning with policy consensus. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4674–4679. IEEE, 2019.

[476] Tingting Zhao, Hirotaka Hachiya, Gang Niu, and Masashi Sugiyama. Analysis and improvement of policy gradient estimation. In *Advances in Neural Information Processing Systems*, pages 262–270, 2011.

[477] Lianmin Zheng, Jiacheng Yang, Han Cai, Ming Zhou, Weinan Zhang, Jun Wang, and Yong Yu. Magent: A many-agent reinforcement learning platform for artificial collective intelligence. In McIlraith and Weinberger [274].

[478] Martin Zinkevich, Amy Greenwald, and Michael L Littman. Cyclic equilibria in markov games. In *Advances in Neural Information Processing Systems*, pages 1641–1648, 2006.

[479] Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Advances in neural information processing systems*, pages 1729–1736, 2008.

[480] Kevin JS Zollman. Talking to neighbors: The evolution of regional meaning. *Philosophy of Science*, 72(1):69–85, 2005.

[481] Alexander Zook and Mark O Riedl. Automatic game design via mechanic generation. In *AAAI*, pages 530–537, 2014.

[482] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.