
A Limited-Capacity Minimax Theorem for Non-Convex Games or: How I Learned to Stop Worrying about Mixed-Nash and Love Neural Nets

Gauthier Gidel^{†,♦}
Mila, Université de Montréal

David Balduzzi[♦]
XTX Markets

Wojciech Marian Czarnecki
DeepMind

Marta Garnelo
DeepMind

Yoram Bachrach
DeepMind

Abstract

Adversarial training, a special case of multi-objective optimization, is an increasingly prevalent machine learning technique: some of its most notable applications include GAN-based generative modeling and self-play techniques in reinforcement learning which have been applied to complex games such as Go or Poker. In practice, a *single* pair of networks is typically trained to find an approximate equilibrium of a highly nonconcave-nonconvex adversarial problem. However, while a classic result in game theory states such an equilibrium exists in concave-convex games, there is no analogous guarantee if the payoff is nonconcave-nonconvex. Our main contribution is to provide an approximate minimax theorem for a large class of games where the players pick neural networks including WGAN, StarCraft II and Blotto Game. Our findings rely on the fact that despite being nonconcave-nonconvex with respect to the neural networks parameters, these games are concave-convex with respect to the actual models (e.g., functions or distributions) represented by these neural networks.

1 INTRODUCTION

Real-world games have been used as benchmarks in artificial intelligence for decades [Samuel, 1959, Tesauro, 1995], with recent progress on increasingly complex

domains such as poker [Brown and Sandholm, 2017, 2019], chess, Go [Silver et al., 2017], and StarCraft II [Vinyals et al., 2019]. Simultaneously, remarkable advances in generative modeling of images [Wu et al., 2019] and speech synthesis [Bińkowski et al., 2020] have resulted from two-player games *explicitly* designed to facilitate via carefully constructed arms races [Goodfellow et al., 2014]. Two-player zero-sum games also play a fundamental role in building classifiers that are robust to adversarial attacks [Madry et al., 2017].

The goal of the paper is to put learning—by neural nets—in two-player zero-sum games on a firm theoretical foundation to answer the question: *What does it mean to solve a game with neural nets?*

In single-objective optimization, performance is well-defined via a fixed objective. However, it is not obvious what counts as “optimal” in a two-player zero-sum nonconcave-nonconvex setting. Since each player’s goal is to maximize its payoff, it is natural to ask whether a player can maximize its utility independently of how the other player behaves. von Neumann et al. [1944] laid the foundation of game theory with the Minimax theorem, which provides a meaningful notion of optimal behavior against an unknown adversary. For a two-player zero-sum game, such a solution concept incorporates two notions: (i) a *value* V , (ii) a *strategy for each player* such that their average gain is at least V (resp. $-V$) no matter what the other does. The existence of such a value and optimal strategies in concave-convex games is guaranteed in Sion et al. [1958], an extension of von Neumann’s result.

From a game-theoretic perspective, minimax may not exist in nonconcave-nonconvex (NC-NC) games. Nevertheless, machine learning (ML) practitioners typically train a *single* pair of neural networks to solve

$$\min_{\theta \in \Theta} \max_{w \in \Omega} \varphi(w, \theta) \quad \text{where} \quad \varphi(\cdot, \cdot) \text{ is NC-NC.} \quad (1)$$

Previous work [Arora et al., 2017, Hsieh et al., 2019, Domingo-Enrich et al., 2020] coped with this

[†]Canada CIFAR AI Chair. [♦]Work done at DeepMind.
Correspondence: gauthier.gidel@mila.quebec

nonconcave-nonconvexity issue by transforming Eq. 1 into a bilinear minimax problem over the Borel distributions on Θ and Ω (a.k.a. lifting trick),

$$\min_{\mu \in \mathcal{P}(\Theta)} \max_{\nu \in \mathcal{P}(\Omega)} \langle \mu, A\nu \rangle := \mathbb{E}_{\theta \sim \mu} [\varphi(w, \theta)] \quad (2)$$

However, working on the space of distributions (a.k.a. mixed strategies) over the weights of a neural network is not practical and does not exactly correspond to the initial problem (1). That is why we do not consider (2) and focus instead on minimax theorems for (1).

Our main contribution is Theorem 1, an approximate limited-capacity minimax theorem for NC-NC games. This result contrasts with the negative result of Jin et al. [2019] who construct a NC-NC game where pure global minimax does not exist. The insights provided by our main theorem are three-fold; first, it provides a principled explanation of why practitioners have successfully trained a single pair of neural nets in games like GANs. Secondly, the equilibrium achieved in the theorem has a meaningful interpretation as the solution of a game where the players have *limited-capacity*. Finally, it shed light on some geometric properties of the space of neural networks (roughly, it is ‘almost convex’), that could be leveraged in diverse contexts of deep learning theory. All the proofs can be found in the appendix.

2 RELATED WORK

Minimax theorems in GANs. Many papers have adopted a game-theoretic perspective on GANs, motivating the computation of distributions of generators (in practice, finite collections) [Arora et al., 2017, Oliehoek et al., 2018, Hsieh et al., 2019, Grnarova et al., 2018, Domingo-Enrich et al., 2020] or an average of discriminators [Farnia and Tse [2018]]. However, these papers fail to explain why, in practice, it suffices to train only a single generator and discriminator (instead of collections) to achieve state-of-the-art performance [Brock et al., 2019]. Overall, even if we share motivations with the related work mentioned above (providing principled results), our results and conclusion are fundamentally different: we explain why using a single generator and discriminator—not a distribution over them—makes sense. We do so by proving that one can achieve a notion of nonconcave-nonconvex minimax equilibrium in GANs parametrized with neural nets.

Stackelberg games and local optimality. The literature has considered other notions of equilibrium. Recently, Fiez et al. [2020] proved results on games where the goal is to find a (local) Stackelberg equilibrium. Using that perspective, Zhang et al. [2020] and Jin et al. [2019] studied local-optimality in the

context of nonconcave-nonconvex games. Alternative notions of local equilibria have been proposed and studied by Berard et al. [2020], Farnia and Ozdaglar [2020], Schaefer et al. [2020] who argued that local Nash equilibria may not be the right notion of local optimality. Our work provides a complementary perspective by providing a *global* minimax optimality theorem in a restricted—though realistic—nonconcave-nonconvex setting. Stackelberg equilibria may be meaningful in some contexts, such as adversarial training, but we argue in §B that the minimax theorem is fundamental for defining a valid notion of solution for many machine learning applications.

Parametrized strategies in games. The notion of latent matrix games mentioned in this paper is similar to the pushforward technique proposed by Dou et al. [2019]. It can also be related to the latent policies used in some multi-agent reinforcement learning (RL) applications. For instance the agent trained by Vinyals et al. [2019] to play the game of StarCraft II had policies of form $\pi(a|s, z)$ where z belongs to structured space that corresponds to a particular way to start the game or to actions it should complete during the game (e.g., first constructed buildings). Moreover, using parametrized function to estimate strategies in games has been at the heart of multi-agent RL [Baxter et al., 2000, François-Lavet et al., 2018, Mnih et al., 2015]. The idea of using a latent space to parametrize distributions has also been widely used in the ML community [Goodfellow et al., 2014]. Our contribution regarding latent matrix games (and more broadly latent RL policies) is theoretical: we provide a framework to study equilibria in parametrized NC-NC games as well as an existence result for an equilibrium (Thm. 1).

Bounded rationality. In his seminal work, Simon [1969] introduced the principle of bounded rationality. Generally speaking, it aims to capture the idea that rational agents are actually incapable of dealing with the full complexity of a realistic environment, and thus by these limitations, achieve a sub-optimal solution. Neyman [1985], Papadimitriou and Yannakakis [1994], Rubinstein and Dalggaard [1998] modeled these limitations by constraining the computational resources of the players. Similarly, Quantal response equilibrium (QRE) [McKelvey and Palfrey, 1995] is a way to model bounded rationality: the players do not choose the best action, but assign higher probabilities to actions with higher reward. Overall, QRE, bounded rationality/computation have a similar goal as our notion of limited capacity equilibrium: to model players that are limited by computation/memory/reasoning. However, the way the limits are modeled differs since, in this work, the limitations of the players are embedded in the

representative power of the class of function considered.

Finding a Nash equilibrium of Colonel Blotto.

After its introduction by Borel [1921], finding a Nash equilibrium of the Colonel Blotto game has been an open question for 85 years. Roberson [2006] found an equilibrium solution for the continuous version of the game, later extended to the discrete symmetric case by Hart [2008]. The equilibrium computation in the general case remains open. Recently, Blotto has been used as a challenging use-case for equilibrium computation [Ahmadinejad et al., 2019]. Similarly, we consider a variant of Blotto to validate that we can practically find approximate equilibrium in games with a single pair of neural networks.

3 MOTIVATION: MINIMAX IN MACHINE LEARNING

A *two-player zero-sum game* is a *payoff function* $\varphi : \Omega \times \Theta \rightarrow \mathbb{R}$, that evaluates pairs of strategies (w, θ) . The goal of the game is to find an *equilibrium*, i.e., a pair of strategies (w^*, θ^*) such that,

$$\varphi(w, \theta^*) \leq \varphi(w^*, \theta^*) \leq \varphi(w^*, \theta), \quad \forall w \in \Omega, \theta \in \Theta.$$

Having such an equilibrium ensures that the order in which the players choose their respective strategy does *not* matter and that there is a *non-exploitable* strategy,

$$\min_{\theta \in \Theta} \max_{w \in \Omega} \varphi(w, \theta) = \max_{w \in \Omega} \min_{\theta \in \Theta} \varphi(w, \theta) = \varphi(w^*, \theta^*).$$

If the function φ is concave-convex and if the sets Θ and Ω are convex and compact then Sion et al. [1958]’s Minimax theorem guarantees a Nash equilibrium exists.

Previous theoretical work in the context of machine learning [Arora et al., 2017, Oliehoek et al., 2018, Grnarova et al., 2018, Hsieh et al., 2019] considered the model *parameters* w and θ as the strategies of the game. Arguably, the most well-known example of such a game is GANs.

Example 1. [Goodfellow et al., 2014] A GAN is a game where the first player picks a binary classifier D_w parametrized by $w \in \mathbb{R}^{p_1}$ called discriminator, and the second player picks a generator G_θ that is a parametrized mapping from a latent space to an output space. The payoff φ is then the ability of the first player to discriminate a real data distribution p_d from the generated distribution,

$$\varphi(w, \theta) := \mathbb{E}_{x \sim p_d} [\log D_w(x)] + \mathbb{E}_{z \sim \mathcal{N}(0, I_d)} [\log (1 - D_w(G_\theta(z)))]. \quad (3)$$

Unfortunately, Example 1 does *not* satisfy Sion Minimax theorem’s assumptions for the following reasons:

- (i) The parameter sets are not compact. (ii) The function φ is not concave-convex because of the non-convexity induced by the neural networks parametrization. While one can easily cope with the first issue—by for instance restricting ourselves to bounded weights or by leveraging Fan’s Theorem [Fan, 1953]—the second issue (ii) is an intrinsic part of learning by neural networks.

On the one hand, one cannot expect (3) to be valid for general nonconcave-nonconvex (NC-NC) games [Jin et al., 2019]. On the other hand, many games in the context of machine learning have a particular structure since, as we will see in the next section, their NC-NC aspect comes from the neural network parametrization.

Two complementary perspectives on a game.

Example 1 can be interpreted as a game between two players, one player, the *generator*, proposes a sample that the other player, the *discriminator* tries to distinguish from a real data distribution p_{data} . In that game, the parameters w and θ of the payoff function (3), do not explicitly correspond to any meaningful strategy—i.e., generating a sample or distinguishing data from generated samples—but they respectively parametrize models (a discriminator and a distribution) that have an intuitive interpretation in the GAN game.

Considering q_θ the generated distribution in (3), we then have a duality between *parameters* and *models*

$$\begin{aligned} \tilde{\varphi}(\underbrace{D_w, q_\theta}_{\text{models}}) &:= \mathbb{E}_{p_d} [\log D_w(x)] + \mathbb{E}_{q_\theta} [\log (1 - D_w(x'))] \\ &= \varphi(\underbrace{w, \theta}_{\text{params}}). \end{aligned} \quad (4)$$

A compelling aspect of this dual perspective is that even though, one *cannot* expect φ , the payoff function of the *parameters* w and θ , to be concave-convex, the payoff of the *models* $\tilde{\varphi}$ is *concave-convex*. Formally, for $w_i \in \Omega$, $\theta_i \in \Theta$, and $\lambda_i \in [0, 1]$, $i = 1 \dots K$, $\lambda_1 + \dots + \lambda_K = 1$ we have (by concavity of log and linearity of $q \mapsto \mathbb{E}_q$),

$$\begin{aligned} \tilde{\varphi}(\sum_{i=1}^K \lambda_i D_{w_i}, q_\theta) &\geq \sum_{i=1}^K \lambda_i \tilde{\varphi}(D_{w_i}, q_\theta), \quad \forall \theta \in \Theta, \\ \tilde{\varphi}(D_w, \sum_{i=1}^K \lambda_i q_{\theta_i}) &= \sum_{i=1}^K \lambda_i \tilde{\varphi}(D_w, q_{\theta_i}), \quad \forall w \in \Omega. \end{aligned}$$

Note that the notion of convex combination for the *models* is quite subtle here: $\lambda D_w + (1 - \lambda) D_{w'}$ corresponds to a convex combination of functions while $\lambda q_\theta + (1 - \lambda) q_{\theta'}$ corresponds to a convex combination (a.k.a, mixture) of distributions.

Even if the payoff (4) is concave-convex with respect to (D, p) , one *cannot* apply (yet) any standard minimax theorem for the following reason: given $w_1, w_2 \in \Omega$ and $\lambda \in [0, 1]$ we may have

$$\nexists w \in \Omega, \quad \text{s.t.} \quad \lambda D_{w_1} + (1 - \lambda) D_{w_2} = D_w, \quad (5)$$

meaning that the set of functions $\mathcal{F}_\Omega := \{D_w \mid w \in \Omega\}$ may *not* be convex. However, for the particular case

of parametrized neural networks we will show that the set \mathcal{F} is “almost convex” (see Prop. 2 and 3). It is one of the core results used in Thm. 1’s proof.

4 AN ASSUMPTION FOR NC-NC GAMES

The games arising in machine learning are not classical normal- or extensive-form games. Rather, they often use neural nets models to approximate complex functions and high dimensional distributions [Brook et al., 2019, Razavi et al., 2019]. That is why they are often considered *general nonconcave-nonconvex (NC-NC) games* (1). However, as illustrated in (4), in the machine learning context, many games have a particular structure where the models’ payoff is concave-convex.

Assumption 1. *The NC-NC game (1) is assumed to have a concave-convex models’ payoff, i.e., w and θ respectively parametrize f_w and q_θ such that,*

$$\varphi(\underbrace{w, \theta}_{\text{params}}) = \tilde{\varphi}(\underbrace{f_w, q_\theta}_{\text{models}}) \quad (6)$$

where $(f, q) \mapsto \tilde{\varphi}(f, q)$ is concave-convex. We call f_w and q_θ *the models picked by the players*, they can either be a parametrized function or distribution.

The intuition behind this assumption is that the nonconvex-nonconcavity of the problem comes from the (neural network) parametrization of the models. One example of such a game has been developed in (4) where the first player picks a function D_w and the second one a distribution over images q_θ . Another closely related example is the Wasserstein GAN (WGAN).

Example 2. [Arjovsky et al., 2017] *The WGAN formulation is a minimax game with a payoff φ s.t.,*

$$\varphi(w, \theta) = \tilde{\varphi}(D_w, q_\theta) := \mathbb{E}_{x \sim p_{\text{data}}} D_w(x) - \mathbb{E}_{x' \sim q_\theta} D_w(x')$$

where the discriminator D_w has to be 1-Lipschitz, i.e., $\|D_w\|_L \leq 1$. By bilinearity of the function $(D, p) \mapsto \mathbb{E}_p[D(x)]$ we have that $\tilde{\varphi}$ is bilinear and thus satisfies Assumption 1.

Finally, we present how Assumption 1 holds when trying to solve a matrix game with a very large (or even infinite) number of strategies by parametrizing mixed strategies.¹

¹Note that here we do not claim the novelty of parametrizing policies/strategies, such idea has been used in many games and RL applications (see related work section). We instead focus on illustrating how a particular way of parametrizing leads to game that satisfies Assump. 1.

Using function approximation to solve matrix games In the case of matrix games, the payoff function $\varphi : A \times B \rightarrow \mathbb{R}$ has no concave-convex structure, and the sets A and B are often even discrete. von Neumann et al. [1944] introduced mixed strategies $p \in \Delta(A)$, where $\Delta(A)$ is the set of probability distributions over A , in order to guarantee the existence of an equilibrium. In game-theory, a well-known example of a challenging matrix game is the Colonel Blotto game.

Example 3 (Colonel Blotto Game). *Consider two players who control armies of S_1 and S_2 soldiers respectively. Each colonel allocates their soldiers on K battlefields. A strategy for player- i is an allocation $a_i \in A_i$ and the payoff of the first player is the number of battlefields won*

$$\varphi(a_1, a_2) := \frac{1}{K} \sum_{k=1}^K \mathbf{1}\{[a_1]_k > [a_2]_k\} \quad (7)$$

where $A_i := \{a \in \mathbb{N}^K : \sum_{k=1}^K [a]_k \leq S_i, 1 \leq i \leq 2\}$ and $[a]_k$ correspond to the k^{th} coordinate of a .

In Example 3, the number of strategies grows *exponentially fast* as K grows. Consequently, one cannot afford to work with an explicit distribution over the strategies. A tractable way to compute an equilibrium of the Colonel Blotto Game has been an open question for decades. The GANs examples (Example 1 & 2) suggest to consider distributions implicitly defined with a generator. Given a latent space \mathcal{Z} , a latent distribution π on \mathcal{Z} and a mapping $g_\theta : \mathcal{Z} \rightarrow A$, we can define the distribution $q_\theta \in \Delta(A)$ as

$$a \sim q_\theta : a = g_\theta(z), z \sim \pi. \quad (8)$$

Definition 1 (Latent Matrix Game). *A latent matrix game $(\varphi, \mathcal{F}, \mathcal{G})$ is a two-player zero-sum game where the players pick $f_w \in \mathcal{F}$ and $g_\theta \in \mathcal{G}$ and, given π and π' two fixed distributions, obtain payoffs*

$$\varphi(w, \theta) := \mathbb{E}_{z \sim \pi, z' \sim \pi'} [\varphi(f_w(z), g_\theta(z'))].$$

The reformulation of any matrix game as a latent game satisfies Assumption 1.

Example 3 (Latent Blotto). *Consider the functions $f_w : \mathbb{R}^p \rightarrow A_1$ and $g_\theta : \mathbb{R}^p \rightarrow A_2$. The payoff is*

$$\varphi(w, \theta) := \frac{1}{K} \sum_{k=1}^K \mathbb{P}([f_w(Z_1)]_k > [g_\theta(Z_2)]_k) \quad (9)$$

where $Z_1, Z_2 \sim \mathcal{N}(0, I_p)$ are independent Gaussians and A_i is defined in (7).

Latent matrix games encompass multi-agent RL games played with RL policies such as the setting used by Vinyals et al. [2019] to play StarCraft II. The agent, called AlphaStar, has a latent-conditioned policy

$\pi(a|s, z)$ where z belongs to a structured space that represents information about how to start constructing units and buildings, and that is sampled from an expert human player distribution: $z \sim p_{human}(z)$. Given two agents $\pi_1(a|s, z)$ and $\pi_2(a|s, z)$, the payoff in the latent game is $\varphi(\pi_1, \pi_2) = \mathbb{P}(\pi_1 \text{ beats } \pi_2)$. The classes \mathcal{F} and \mathcal{G} correspond to the neural architectures used to parametrize the policies; the priors π and π' are the human expert distribution p_{human} .

In that example, and more generally in multi-agent RL zero-sum games played with policies parametrized by neural networks, the payoff $\varphi(w, \theta) = \mathbb{P}(\pi_w \text{ beats } \pi_\theta)$ is a potentially nonconcave-nonconvex function of the parameters but satisfies Assumption 1.

5 A MINIMAX THEOREM

We want to prove a minimax theorem for some NC-NC games (1) that satisfy Assumption 1. We start with an informal statement of our result.

Theorem 1. *[Informal] Let φ be a nonconcave-nonconvex payoff that satisfies Assump. 1 with $\tilde{\varphi}$ bilinear and where the players pick (w, θ) , the parameters of two neural networks. For any $\epsilon > 0$ there exists $(w_\epsilon, \theta_\epsilon)$ achieving an approximate limited-capacity equilibrium.*

The notion of approximate limited-capacity equilibrium mentioned in that informal statement is detailed in the complete statement of Theorem 1. When played with neural networks Example 2 and 3 satisfy the hypothesis of this theorem. The proof of this Theorem is split into 3 main steps: (i) in §5.1 by using the fact that $\varphi(w, \theta) = \tilde{\varphi}(f_w, q_\theta)$ we provide the existence of a limited-capacity equilibrium in the convex hull of the models' space of f_w and q_θ . Note that, since we are working in the convex hull, one can *only* expect (in general) to achieve this equilibrium with a collection of parameters $(w_i, \theta_i)_{i \in I}$. (ii) in §5.2 we show that approximate equilibrium can be achieved with a relatively small convex combination. (iii) in §5.3 we show that when using neural networks, such small convex combination of models can be achieved by a single larger neural network. A formal definition of convex combination of models is provided in §5.1.

5.1 Limited Capacity Equilibrium in the Models' Space

Recall that by Assump. 1, the NC-NC payoff φ can be written as,

$$\varphi(w, \theta) = \tilde{\varphi}(f_w, q_\theta) \quad (10)$$

where $(f, q) \mapsto \tilde{\varphi}(f, q)$ is concave-convex. The models f_w and q_θ are either **functions** or **distributions** respectively **parametrized by w and θ** . For instance, in the

context of WGAN (Example 2), f_w would be the discriminator and q_θ would be the generated probability distribution. In that example, notice that $\tilde{\varphi}(f_w, \cdot)$ is *not* convex with respect to the generator function but only with respect to the generated distribution. Similarly, if we computed convex combinations generator's parameters, the payoff φ would not be convex in general. Moreover, the Lipchitz constraint in Example 2 is a natural constraint in the **function space**, but it is challenging to translate it into a constraint in the **parameter space**.² Overall, using (10) one can rewrite (1) as follows,

$$\min_{f \in \mathcal{F}_\Omega} \max_{g \in \mathcal{G}_\Theta} \tilde{\varphi}(f, g) \quad (11)$$

where \mathcal{F}_Ω and \mathcal{G}_Θ are function or distribution spaces (depending on the application) incorporating the limited capacity constraints of the problem, e.g., Lipschitz constraint. **In the following**, for simplicity of the discussion, we discuss what the formal definitions of a convex combination are when \mathcal{F}_Ω is a **function space** and \mathcal{G}_Θ is a **distribution space** when we have no additional constraint aside from the parametrization, i.e., $\mathcal{F}_\Omega := \{f_w \mid w \in \Omega\}$ and $\mathcal{G}_\Theta := \{q_\theta \mid \theta \in \Theta\}$. However, these notions and our results extend if we consider that both models are distributions (e.g., in Example 3), or if we add any convex constraint on the functions or the distributions, see Example 2.

Convex combination of functions. Let us consider w_1 and $w_2 \in \Omega$, the convex combination of the models f_{w_1} and f_{w_2} is their point-wise averaging. The convex hull of \mathcal{F}_Ω can be defined as,

$$\begin{aligned} \text{hull}(\mathcal{F}_\Omega) &:= \{\text{Averages from } \mathcal{F}_\Omega\} \\ &= \left\{ \sum_{i=1}^K \lambda_i f_{w_i} \mid w_i \in \Omega, \lambda \in \Delta_K, K \geq 0 \right\}. \end{aligned} \quad (12)$$

where Δ_K is the K -dimensional simplex, i.e., $\{\lambda \in \mathbb{R}^K, \lambda_i \geq 0, \sum_{i=1}^K \lambda_i = 1\}$.

Convex combination of distributions. Consider latent mappings θ_1 and $\theta_2 \in \Theta$ that parametrize probability distribution q_{θ_1} and q_{θ_2} over a set \mathcal{X} . The *convex combination* q_λ of q_{θ_1} and q_{θ_2} with $\lambda \in [0, 1]$ is the mixture of these two probability distributions, $q_\lambda := \lambda q_{\theta_1} + (1 - \lambda) q_{\theta_2}$.

To sample from q_λ , flip a biased coin with $\mathbb{P}(\text{heads}) = \lambda$. If the result is heads then sample a strategy from q_{θ_1} and if the result is tails then sample from q_{θ_2} . The

²In practice, parameters are clipped [Arjovsky et al., 2017] or the Lipchitz constant of the network is approximated [Miyato et al., 2018]. These approximations can be arbitrarily far from the original constraint.

convex hull of \mathcal{G}_Θ is,

$$\begin{aligned} \text{hull}(\mathcal{G}_\Theta) &:= \{\text{Mixtures from } \mathcal{G}_\Theta\} \\ &= \left\{ \sum_{i=1}^K \lambda_i q_{\theta_i} \mid \theta_i \in \Theta, \sum_{i=1}^K \lambda_i \in \Delta_K, K \geq 0 \right\}. \end{aligned} \quad (13)$$

The set $\text{hull}(\mathcal{G}_\Theta)$ is a subset of $\mathcal{P}(\mathcal{X})$, the set of probability distributions on \mathcal{X} . This set is different from the set of distributions supported on \mathcal{G}_Θ considered by Arora et al. [2017], Hsieh et al. [2019]. It contains ‘smaller’ mixtures because there may be many distributions supported on \mathcal{G}_Θ that correspond to the same $p \in \text{hull}(\mathcal{G}_\Theta)$. Moreover these works did not take advantage of the convexity with respect to the discriminator function (see Example 1 and 2) by considering (12).

Existence of an equilibrium by playing in the convex hulls. Our first result is that there exists an equilibrium by allowing functions or distributions to be picked from their convex hulls.

Proposition 1. *Let φ be a game that follows Assumption 1. If \mathcal{G}_Θ and \mathcal{F}_Ω are compact, then there exist a value for the game such that,*

$$\begin{aligned} V(\Omega, \Theta) &:= \sup_{f \in \text{hull}(\mathcal{F}_\Omega)} \inf_{q \in \text{hull}(\mathcal{G}_\Theta)} \tilde{\varphi}(f, q) \\ &= \inf_{q \in \text{hull}(\mathcal{G}_\Theta)} \sup_{f \in \text{hull}(\mathcal{F}_\Omega)} \tilde{\varphi}(f, q), \end{aligned} \quad (14)$$

where $\text{hull}(\mathcal{G}_\Theta)$ and $\text{hull}(\mathcal{F}_\Omega)$ are either defined in (12) or in (13), depending on the type model.

After showing that the closure of $\text{hull}(\mathcal{G}_\Theta)$ and $\text{hull}(\mathcal{F}_\Omega)$ are compact, this proposition is a corollary of Sion et al. [1958]’s minimax theorem (see §C). Note that Ω and Θ are arbitrary and that this equilibrium differs from the infinite-capacity equilibrium of the game (11) where we would allow f and g to be any function or distribution (i.e. with no parametrization restriction). Because we consider the convex hull of \mathcal{F}_Ω and \mathcal{G}_Θ , this equilibrium is achieved with *convex combinations* (Eq. 12 & 13) of q_{θ_i} , $i \geq 0$ (resp. f_{w_i}) and thus there is no reason to expect to achieve this equilibrium with a single pair of weights (w, θ) in general. However in §5.2, we show that one can approximate such an equilibrium with relatively small convex combinations.

5.2 Approximate minimax equilibrium

Approximate equilibria for (14) are the pairs of models ϵ -close to achieving the value of the game.

Definition 2 (ϵ -equilibrium). *A pair $(f_\epsilon^*, q_\epsilon^*) \in \text{hull}(\mathcal{F}) \times \text{hull}(\mathcal{G})$ is an ϵ -equilibrium if,*

$$\begin{aligned} \min_{q \in \text{hull}(\mathcal{G}_\Theta)} \tilde{\varphi}(f_\epsilon^*, q) &\geq V(\Omega, \Theta) - \epsilon \\ \text{and } \max_{f \in \text{hull}(\mathcal{F}_\Omega)} \tilde{\varphi}(f, q_\epsilon^*) &\leq V(\Omega, \Theta) + \epsilon. \end{aligned} \quad (15)$$

Note that f_ϵ^* does not depend on p_ϵ^* and vice-versa. We will show that such approximate equilibria are achieved with finite convex combinations. Considering $f_k \in \mathcal{F}_\Omega$ and $q'_k \in \mathcal{G}_\Theta$ (that can either be functions or distributions) we aim at finding the smallest convex combination that is an ϵ -equilibrium.

$$\begin{aligned} (K_\epsilon^\Omega, K_\epsilon^\Theta) &:= \text{Smallest } K \text{ and } K' \in \mathbb{N} \\ \text{s.t. } &\left(\sum_{k=1}^K \lambda_k f_k, \sum_{k=1}^{K'} \lambda'_k q'_k \right) \text{ is an } \epsilon\text{-equilibrium.} \end{aligned} \quad (16)$$

Our goal is to provide a bound that depends on ϵ and on some properties of the classes \mathcal{F}_Ω and \mathcal{G}_Θ .

Theorem 2. *Let φ a game that satisfies Assumption 1. If $\tilde{\varphi}$ is bilinear, $\|\theta\| \leq R$, $\|w\| \leq R$, $\forall w, \theta \in \Omega \times \Theta \subset \mathbb{R}^d \times \mathbb{R}^p$ and φ is L -Lipschitz then,*

$$K_\epsilon^\Omega \leq \frac{4D_w^2 p}{\epsilon^2} \ln\left(\frac{6RL}{\epsilon^2}\right) \text{ and } K_\epsilon^\Theta \leq \frac{4D_\theta^2 d}{\epsilon^2} \ln\left(\frac{6RL}{\epsilon^2}\right) \quad (17)$$

where $D_w := \max_{w, w', \theta} \varphi(w, \theta) - \varphi(w', \theta)$ and $D_\theta := \max_{w, \theta, \theta'} \varphi(w, \theta) - \varphi(w, \theta')$.

Roughly, the number K_ϵ^Θ expresses to what extent the set of distributions induced by the mappings in \mathcal{G}_Θ has to be ‘convexified’ to achieve an approximate equilibrium. Note that in practice we expect this quantity to be small. For instance, in the context of GANs, if the class of discriminators \mathcal{F}_Ω contains the constant function $D(\cdot) = .5$ then $K_\epsilon^\Omega = 1$ since $\varphi(D, G) = 0$, $\forall G \in \mathcal{G}$.

Two close related results are [Farnia and Tse, 2018, Theorem 4] and [Arora et al., 2017, Theorem 4.2]. In the former, the authors point out that the functional space of neural networks is not convex and prove that any function in the convex hull of neural networks can be approximated by a finite average of function. In the latter, the authors show that by considering uniform mixtures of generators and discriminators one can approximate the equilibrium of a GAN. Our results generalize these two ideas by showing that for any payoff function φ that satisfy Assumption 1 one can find an approximate equilibrium with a finite convex combination of function or distributions (depending on the context). We thus simultaneously handle the case of convex combinations of functions or distributions and consider a more general class of payoff functions.

5.3 Achieving a Mixture or an Average with a Single Neural Net

We showed above that under the assumption of Theorem 2, approximate equilibria can be achieved with finite convex combinations. In this section, we investigate how it is possible to achieve such approximate

equilibria with a single neural network. Formally, a neural network $g : \mathbb{R}^{d_{in}} \rightarrow \mathbb{R}^d$ can be written as,

$$g_\theta(x) = b_l + W_l \sigma(b_{l-1} + W_{l-1} \dots \sigma(b_1 + W_1 x)), \quad (18)$$

where W_1, \dots, W_l are the weight matrices, b_1, \dots, b_l the biases and σ is a given non-linearity. We note θ the concatenation of all the parameters of this neural network. We present two results on the geometry of the space of neural networks. The first one concerns *mixtures of distributions* represented by latent neural nets with ReLU non-linearity, and the second one concerns *convex combinations* of neural nets as functions.

Neural Nets Represent Mixtures of Sub-Nets.

First, we get interested in the probability distributions q_θ induced by g_θ , $\theta \in \Theta$, defined as

$$a \sim q_\theta : a = g_\theta(z) \text{ where } z \sim U([0, 1]). \quad (19)$$

One of the motivations of this work is to represent distribution over images usually represented by a high dimensional vector in $[0, 1]^d$. That is why we will assume that our generator function take its value in $[0, 1]^d$. Moreover, for this proposition, we only consider the ReLU non-linearity $\sigma(x) = \text{ReLU}(x) := \max(0, x)$.

Proposition 2. *Let $\theta_k \in [-R, R]^p$, $k = 1 \dots K$, be the parameters of k ReLU nets with p parameters. If the input latent variable is of dimension $d_{in} = 1$ and if for all $k = 1 \dots K$, $z \in [0, 1]$, $g_{\theta_k}(z) \in [0, 1]^d$ and g_{θ_k} is constant outside of $[0, 1]$, then there exists a ReLU net with $K(p + 6)$ non-linearities $\theta \in [-KR, KR]^{K(p+6)}$ such that $d_{TV}(\frac{1}{n} \sum_{k=1}^K q_{\theta_k}, q_\theta) \leq 1/R$ where d_{TV} is the total variation distance.*

Fig. 2b (in §C) illustrates how g_θ is constructed. Unlike the universal approximation theorem, Prop. 2 shows that a single neural network can represent mixtures. On the one hand, when one wants to approximate an arbitrary continuous function, the number of required hidden units may be prohibitively large [Lu et al., 2017] as the error ϵ vanishes. On the other hand, the dimension of θ in Prop. 2 does not depend on any vanishing quantity. The high-level insight is that a large enough ReLU net can represent mixtures of distributions induced by smaller ReLU nets, with a width that grows linearly with the size of the mixture.

Neural Nets can represent an average of Sub-Nets. If we consider averages of functions as described in (12), we can show that point-wise averages of neural networks can be represented by a wider neural network. This result is valid for any non-linearity.

Proposition 3. *Let $w_k \in [-R, R]^p$, $k = 1 \dots K$ be the parameters of k neural nets with p parameters, there exists a neural net parametrized by $w \in [-R, R]^{Kp}$, such that $\frac{1}{n} \sum_{k=1}^K f_{w_k} = f_w$.*

Figure 2a shows how f_w is constructed. Similarly as the Prop. 2, Prop. 3 is a representation theorem that shows that the space of neural networks with a fixed number of parameters is ‘almost’ convex.

5.4 A Minimax Theorem for NC-NC Games Played with Neural Nets

Prop. 2 and 3 give insights about the representative power of neural nets: as their number of parameters grows, neural nets can express larger mixtures/averages of sub-nets. Combining these properties with Thm. 2, we show that approximate equilibria can be achieved for such nonconcave-nonconvex (NC-NC) payoff.

Theorem 1. *Let φ be a NC-NC game that follows Assumption 1. We assume that the models’ payoff $\tilde{\varphi}$ is bilinear and consider two cases where the models have p parameters bounded by R , i.e., $(w, \theta) \in [-R, R]^{2p}$:*

1. [Function vs. distribution]: $\varphi(w, \theta) = \tilde{\varphi}(f_w, q_\theta)$ where q_θ is a distribution parametrized by a ReLU net with $d_{in} = 1$ (see Eq. 19), and f_w is a neural network with any non-linearity. Applies to Ex. 2.
2. [Distribution vs. distribution]: $\varphi(w, \theta) = \tilde{\varphi}(q_w, q_\theta)$ where q_θ and q_w are distributions parametrized by a ReLU net with $d_{in} = 1$. Applies to Example 3.

If $\tilde{\varphi}$ is \tilde{L} -Lipschitz, and if φ is L -Lipschitz, then for any $\epsilon > 0$, there exists $(w_\epsilon^*, \theta_\epsilon^*) \in [-R, R]^{2p}$, such that,

$$\min_{\theta \in [-R, R]^{p_\epsilon}} \varphi(w_\epsilon^*, \theta) + \epsilon + \frac{2\tilde{L}}{R} \geq \max_{w \in [-R, R]^{p_\epsilon}} \varphi(w, \theta_\epsilon^*),$$

where $p_\epsilon \geq C\epsilon \sqrt{\frac{p}{\log(R/\sqrt{p}/\epsilon)}}$, $R_\epsilon \geq R \frac{p_\epsilon}{p}$, and the subnet networks generating the distributions (see Eq. 19) takes their values in $[0, 1]^d$ and are constant outside of $[0, 1]$.

An explicit formula for C is provided in §C.5 as well as variants of this theorem when both players pick a model that is a function. Theorem 1 shows the existence of a notion of weaker-capacity-equilibrium for a nonconcave-nonconvex game where the models use a *standard fully connected architecture*. This result differs from Arora et al. [2017, Theorem 4.3] who, only in the context of GANs, design a *specific architecture* to achieve a different notion of approximate equilibrium.

The notion of weaker-capacity is encompassed within the fact that w_ϵ and θ_ϵ are of dimension $p_\epsilon \leq p$ and are bounded by $R_\epsilon \leq R$. Roughly, this theorem can be interpreted as follows: if one considers a minimax game where the players pick neural networks and where the payoff is bilinear with respect to the models (functions on distributions) represented by these neural networks,

then there exist a combination of parameters that can achieve an ϵ -approximate minimax against subnetworks with p_ϵ parameters, i.e., the first (resp. second) player cannot be beaten by more than ϵ by any sub-network of the second (resp. first) player.

Regarding the number of parameters p_ϵ of these sub-networks, on the one hand, if $\epsilon\sqrt{p} < 1$, then the lower-bound on p_ϵ is vacuous, on the other, the number of parameters of the higher-capacity networks p only needs to (roughly) grow *quadratically* with ϵ to achieve a non-vacuous bound. Hence, a consequence of Theorem 1 is that, for the NC-NC games that follow the assumptions of Theorem 1, *highly over parametrized networks* can provably achieve a non-vacuous notion of approximate equilibrium.

It is worth discussing what happens when some assumptions fail to hold. We think the Lipschitz assumptions are quite standard. We focus on the case of distributions parametrized by a ReLU net with $d_{in} = 1$. We think the assumptions that the non-linearities must be ReLU is an artefact from the proof technique and we believe that with a bit more work a similar version of Theorem 1 would hold with any non-linearities with which one can arbitrarily approximate indicator functions. Extending theorem to $d_{in} > 1$ seems possible though very challenging. The difficulty relies on extending Proposition 2 to $d_{in} \geq 2$. One idea, may be to consider deeper networks to represent a mixture of smaller nets. We leave this question open.

6 APPLICATION: SOLVING COLONEL BLOTTO GAME

Finally, we present some experiments with $d_{in} \geq 2$ on the Colonel Blotto Game to illustrate the soundness of Theorem 1 and explore whether equilibria can be achieved with larger input spaces. We apply the latent game approach developed in Definition 2 to solve a differentiable version of Example 3. We consider a continuous relaxation of the strategy space where $S_1 = S_2$. After renormalization we have that $A_1 = A_2 = \Delta_K$, where Δ_K is the K -dimensional simplex. It is important to notice that in that case an *allocation* corresponds to a *point on the simplex* and a *mixture of allocation* corresponds to a *distribution over the simplex*. We replace the payoff (9) of Latent Blotto by a differentiable one,

$$\varphi(w, \theta) := \mathbb{E}_{\substack{z \sim \pi \\ z' \sim \pi'}} \left[\frac{1}{K} \sum_{k=1}^K \sigma([f_w(z) - g_\theta(z')]_k) \right] \quad (20)$$

where σ is a sigmoid minus $1/2$ and $f_w, g_\theta : \mathbb{R}^p \rightarrow \Delta_K$. This game has been theoretically analyzed by Ferdowsi et al. [2018] when $S_1 > S_2$.

The generative functions f_w and g_θ are dense ReLU nets with 4 hidden layers, 16 hidden units per layer, and a K -dimensional softmax output with a Gaussian latent variable ($d_{in} = 16$). We trained the models using gradient descent ascent on the parameters of f and g with the Adam optimizer [Kingma and Ba, 2015] with $\beta_1 = .5$ and $\beta_2 = .99$.

In Fig. 1b, we present the performance of the agents against a best response. To compute it, we sampled 5000 strategies and computed the best response against this mixed strategy using gradient ascent on the simplex. We also computed the norm of the (stochastic) gradient used to update f . In Fig. 1a, we plotted samples from f at different training times. As we get closer to convergence to a non-exploitable strategy, we can see that this distribution avoids the center of the simplex (putting troops evenly on the battlefields) and the corners (focusing on a single battlefield) that are strategies easily exploitable by focusing on two battlefields, this correlates with the decrease of the gradient and of the suboptimality indicating that the agents learned how to play Blotto.

7 DISCUSSION

Nonconcave-nonconvex games radically differ from minimization problems since equilibria may not exist in general. How, then, can neural nets regularly find meaningful solutions to games like GANs?

In this work, we partially answer this question by leveraging the structure of GANs to show that a single pair of ReLU nets can achieve a notion of limited-capacity-equilibrium. The intuition underlying our theorem is as follows: neural nets have a particular structure that interleaves matrix multiplications and simple non-linearities. The matrix multiplications in one layer of a neural net compute linear combinations of functions encoded by the other layers. In other words, neural nets are (non-)linear mixtures of their sub-networks.

Our main result can be related to games with bounded rationality [Simon, 1969]: when the players pick parametrized models, they are limited by the representational power of the class of models accessible (e.g., a fixed architecture). It is instructive to discuss the relative merits of that limited-capacity aspect occurring. On the one hand, if one had access to any function/distribution an infinite-capacity equilibrium would exist (because the whole function/distribution space is convex). However, this quantity may not be realistic, e.g., in GANs, the optimal infinite-capacity generator must represent the distribution of ‘real-world’ images. If such a concept is not tractable, it seems unrealistic to expect limited capacity agents, such as humans or computers, to find it [Papadimitriou, 2007]. On the

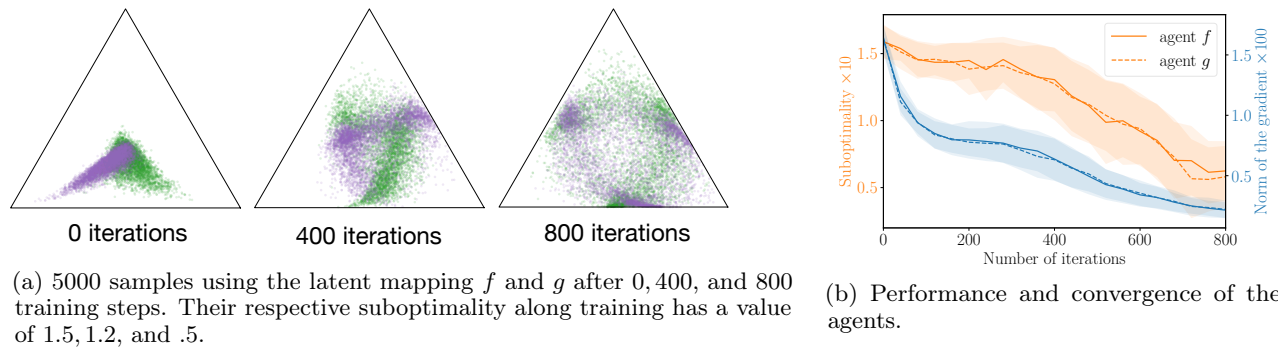


Figure 1: Training of latent agents to play differentiable Blotto with $K = 3$. **Right:** The suboptimality corresponds to the payoff of the agent against a best response. The curves correspond to averages over 40 random seeds with standard deviation.

other hand, our work shows that one can efficiently approximate some equilibria when working with neural networks. In a similar vein as the games with bounded rationality, these equilibria capture the notion that agents—and humans—that play complex games have a limited capacity. It seems to be a more reasonable concept to consider the optimal way to play complex games such as Poker or StarCraft II that are multi-step with imperfect information.

Acknowledgments

Gauthier Gidel would like to thank Ian Gemp for the helpful discussions.

References

- A. Ahmadinejad, S. Dehghani, M. Hajiaghayi, B. Lucier, H. Mahini, and S. Seddighin. From duels to battlefields: Computing equilibria of Blotto and other games. *Mathematics of Operations Research*, 2019.
- C. D. Aliprantis and K. C. Border. *Infinite Dimensional Analysis A Hitchhiker’s Guide*. Springer, 2006.
- M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017.
- S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In *ICML*, 2017.
- J. Baxter, A. Tridgell, and L. Weaver. Learning to play chess using temporal differences. *Machine Learning*, 2000.
- H. Berard, G. Gidel, A. Almahairi, P. Vincent, and S. Lacoste-Julien. A closer look at the optimization landscapes of generative adversarial networks. In *ICLR*, 2020.
- M. Bińkowski, J. Donahue, S. Dieleman, A. Clark, E. Elsen, N. Casagrande, L. C. Cobo, and K. Simonyan. High fidelity speech synthesis with adversarial networks. In *ICLR*, 2020.
- E. Borel. La théorie du jeu et les équations intégrales à noyau symétrique. *Comptes rendus de l’Académie des Sciences*, 1921.
- A. Brock, J. Donahue, and K. Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.
- N. Brown and T. Sandholm. Safe and nested subgame solving for imperfect-information games. In *Advances in neural information processing systems*, 2017.
- N. Brown and T. Sandholm. Superhuman AI for multiplayer poker. *Science*, 2019.
- V. Conitzer and T. Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, 2006.
- C. Domingo-Enrich, S. Jelassi, A. Mensch, G. Rotskoff, and J. Bruna. A mean-field analysis of two-player zero-sum games. *arXiv preprint arXiv:2002.06277*, 2020.
- Z. Dou, X. Yan, D. Wang, and X. Deng. Finding mixed strategy nash equilibrium for continuous games through deep learning. *arXiv preprint arXiv:1910.12075*, 2019.
- K. Fan. Minimax theorems. *Proceedings of the National Academy of Sciences of the United States of America*, 1953.
- F. Farnia and A. Ozdaglar. Do GANs always have Nash equilibria? In *ICML*, 2020.
- F. Farnia and D. Tse. A convex duality framework for GANs. In *NeurIPS*, 2018.
- A. Ferdowsi, A. Sanjab, W. Saad, and T. Basar. Generalized colonel Blotto game. In *2018 Annual American Control Conference (ACC)*, 2018.

- T. Fiez, B. Chasnov, and L. J. Ratliff. Implicit learning dynamics in stackelberg games: Equilibria characterization, convergence analysis, and empirical study. In *ICML*, 2020.
- V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau. An introduction to deep reinforcement learning. *arXiv preprint arXiv:1811.12560*, 2018.
- I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, 2014.
- P. Grnarova, K. Y. Levy, A. Lucchi, T. Hofmann, and A. Krause. An online learning approach to generative adversarial networks. In *ICLR*, 2018.
- J. Y. Halpern and R. Pass. Algorithmic rationality: Game theory with costly computation. *Journal of Economic Theory*, 2015.
- S. Hart. Discrete colonel Blotto and general lotto games. *International Journal of Game Theory*, 2008.
- Y.-P. Hsieh, C. Liu, and V. Cevher. Finding mixed nash equilibria of generative adversarial networks. In *ICML*, 2019.
- G. Huang, H. Berard, A. Touati, G. Gidel, P. Vincent, and S. Lacoste-Julien. Parametric adversarial divergences are good task losses for generative modeling. *arXiv preprint arXiv:1708.02511*, 2017.
- C. Jin, P. Netrapalli, and M. I. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? *arXiv preprint arXiv:1902.00618*, 2019.
- E. Kalai. Bounded rationality and strategic complexity in repeated games. In *Game theory and applications*. Elsevier, 1990.
- D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.
- R. J. Lipton and N. E. Young. Simple strategies for large zero-sum games with applications to complexity theory. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 734–740, 1994.
- Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang. The expressive power of neural networks: A view from the width. In *NeurIPS*, 2017.
- A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.
- R. D. McKelvey and T. R. Palfrey. Quantal response equilibria for normal form games. *Games and economic behavior*, 1995.
- T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.
- V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 2015.
- M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2012.
- A. Neyman. Bounded complexity justifies cooperation in the finitely repeated prisoners’ dilemma. *Economics letters*, 1985.
- F. A. Oliehhoek, R. Savani, J. Gallego, E. van der Pol, and R. Groß. Beyond local nash equilibria for adversarial networks. *arXiv preprint arXiv:1806.07268*, 2018.
- C. H. Papadimitriou. The complexity of finding nash equilibria. *Algorithmic game theory*, 2007.
- C. H. Papadimitriou and M. Yannakakis. On complexity as bounded rationality. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, 1994.
- A. Razavi, A. van den Oord, and O. Vinyals. Generating diverse high-fidelity images with VQ-VAE-2. In *Advances in Neural Information Processing Systems*, 2019.
- B. Roberson. The colonel Blotto game. *Economic Theory*, 2006.
- A. Rubinstein and C.-j. Dalggaard. *Modeling bounded rationality*. MIT press, 1998.
- A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 1959.
- F. Schaefer, H. Zheng, and A. Anandkumar. Implicit competitive regularization in gans. In *ICML*, 2020.
- S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of go without human knowledge. *nature*, 2017.
- H. A. Simon. *The sciences of the artificial*. MIT press, 1969.
- M. Sion et al. On general minimax theorems. *Pacific Journal of mathematics*, 1958.
- G. Tesauro. Temporal difference learning and td-gammon. *Communications of the ACM*, 1995.

- O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 2019.
- J. von Neumann, O. Morgenstern, and H. W. Kuhn. *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 1944.
- Y. Wu, J. Donahue, D. Balduzzi, K. Simonyan, and T. Lillicrap. LOGAN: Latent optimisation for generative adversarial networks. *arXiv preprint arXiv:1912.00953*, 2019.
- G. Zhang, P. Poupart, and Y. Yu. Optimality and stability in non-convex-non-concave min-max optimization. *arXiv preprint arXiv:2002.11875*, 2020.