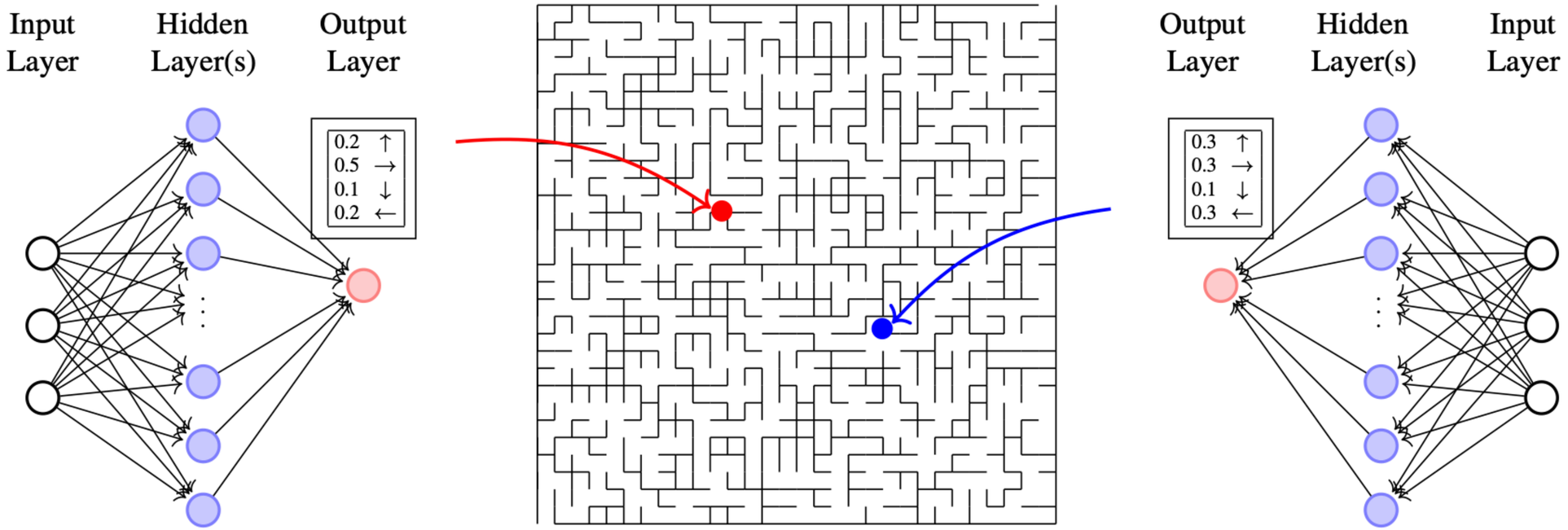
Solving Neural Min-Max Games: The Role of Architecture, Initialization & Dynamics*

Deep Patel & Manolis Vlatakis (UW-Madison, USA)

16th July 2025

*NeurIPS 2025 (Spotlight)

Question at the heart of this talk



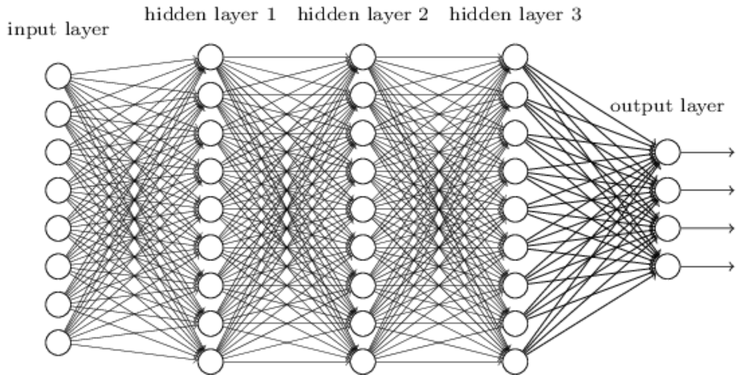
How can two neural networks be designed and trained to compute a solution to a zero-sum game?

Outline

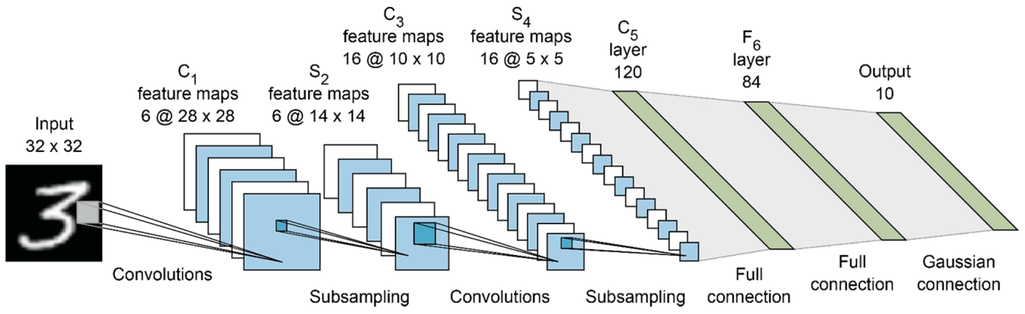
- Chapter 1: Preliminaries
- Chapter 2: From MIN to MIN-MAX
- Chapter 3: Hidden(-Convex-Concave) Games
- Chapter 4: Our Results

Chapter 1: Preliminaries

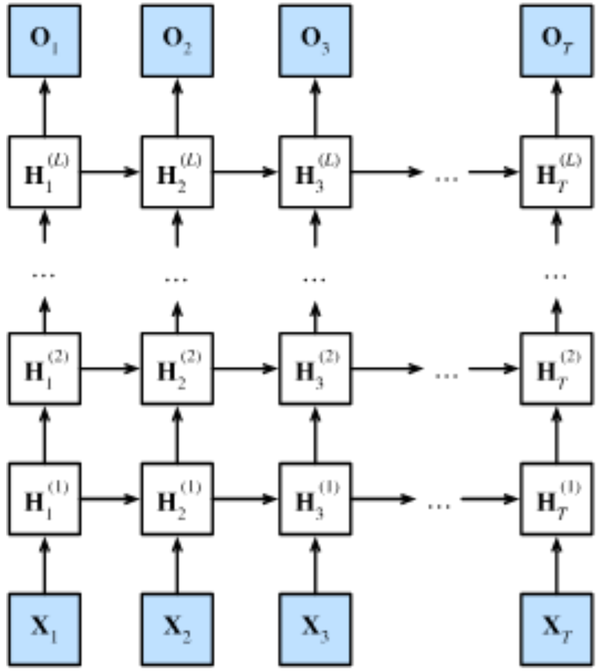
Success of Deep Learning



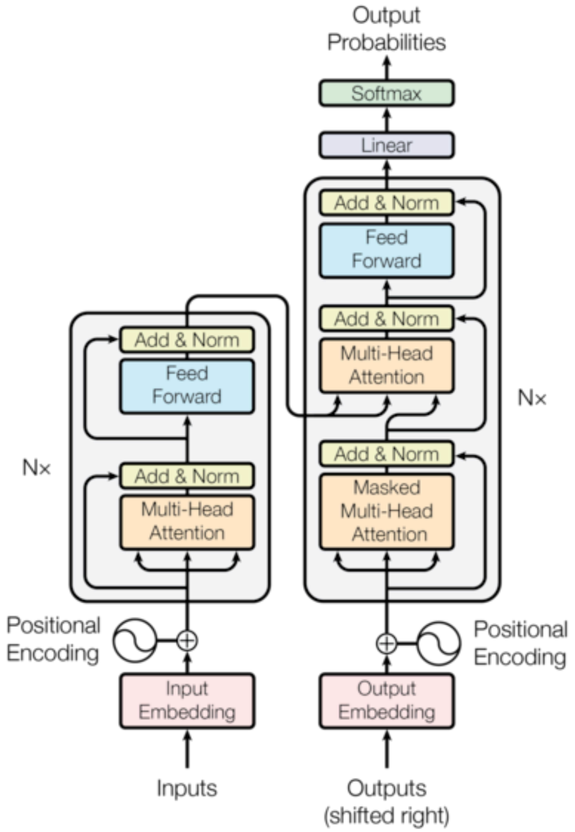
https://shorturl.at/mlTfn



https://shorturl.at/4VGKL



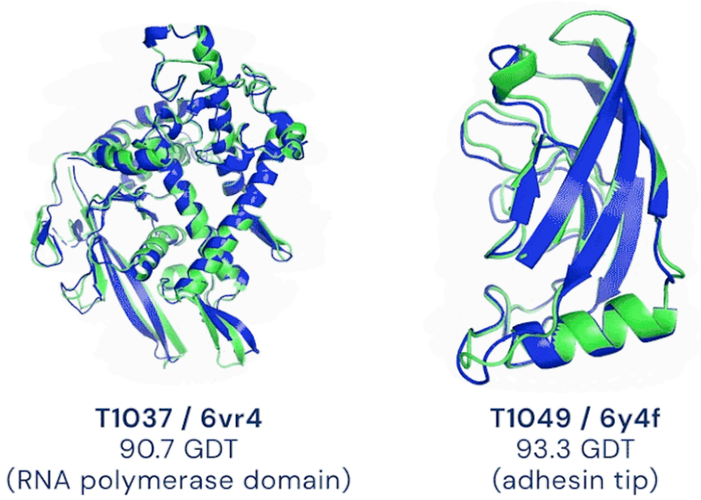
https://shorturl.at/GuvKT



https://shorturl.at/38aUe

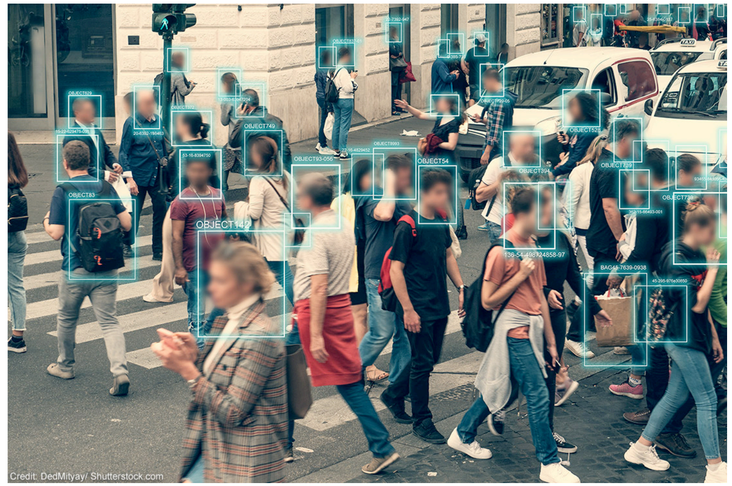
Success of Deep Learning

https://shorturl.at/2COlv

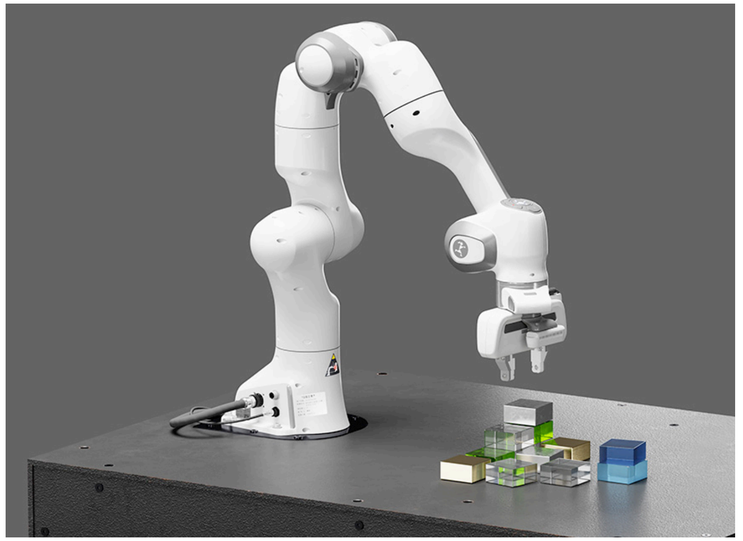


https://shorturl.at/6Oxs4

ChatGPT

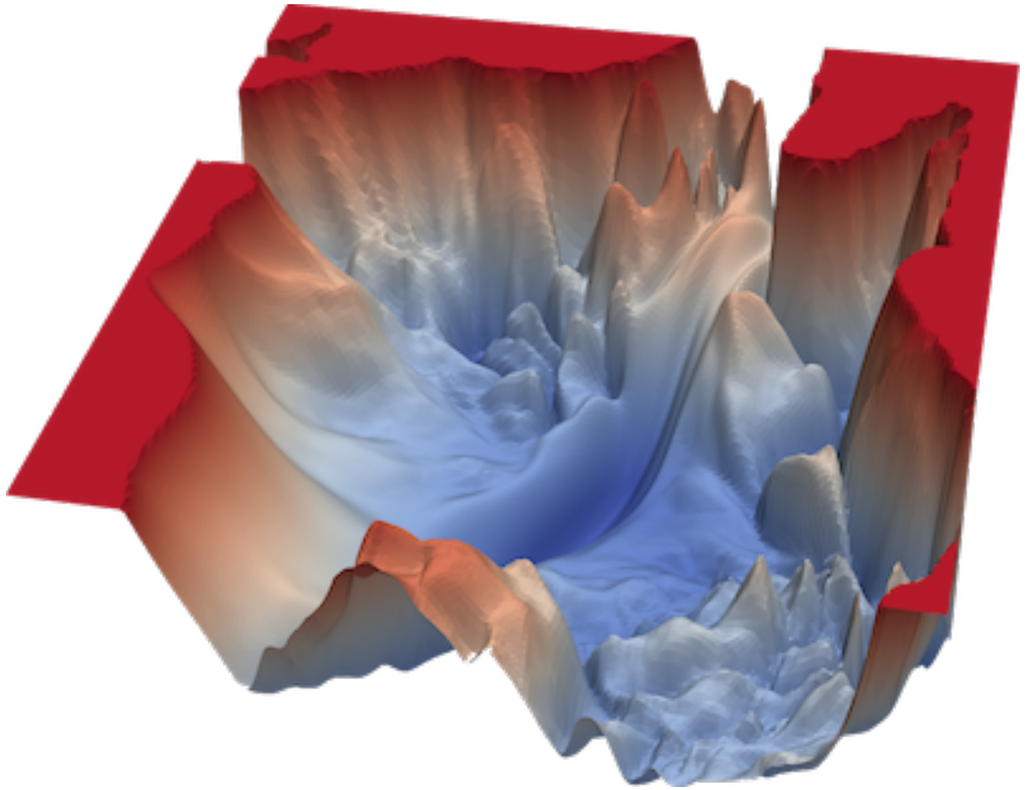


https://shorturl.at/eXGBP



https://shorturl.at/ya6zQ

How Theory Tries to Understand Success of Deep Learning



Gradient Descent Finds Global Minima of Deep Neural Networks

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, Xiyu Zhai *Proceedings of the 36th International Conference on Machine Learning*, PMLR 97:1675-1685, 2019.

A Convergence Theory for Deep Learning via Over-Parameterization

Zeyuan Allen-Zhu, Yuanzhi Li, Zhao Song *Proceedings of the 36th International Conference on Machine Learning*, PMLR 97:242-252, 2019.

The Loss Surface of Deep and Wide Neural Networks

Quynh Nguyen, Matthias Hein *Proceedings of the 34th International Conference on Machine Learning*, PMLR 70:2603-2612, 2017.

How SGD Selects the Global Minima in Over-parameterized Learning: A Dynamical Stability Perspective

Gradient descent optimizes over-parameterized deep ReLU networks

Difan Zou, Yuan Cao, Dongruo Zhou, Quanquan Gu

Image Source: https://www.cs.umd.edu/~tomg/project/landscapes/

Naturally we may ask…

How big a neural network should be so that vanilla methods like (S)GD can converge to global optima?

Formally, we want to know…

$$\theta^\star \in \arg\min_{\theta \in \mathbb{R}^m} \; \mathbb{E}_{(x,y)\sim P_{xy}} \left[L(f(x;\theta), y) \right]$$

- Loss function $L : \mathbb{R}^{d_{out}} \times \mathbb{R} \to \mathbb{R}_{\geq 0}$ (e.g. MSE, CCE, etc.)
- underlying data distribution P_{xy}
- Neural network $f(\,\cdot\,;\theta) : \mathbb{R}^{d_{in}} \to \mathbb{R}^{d_{out}}$ (e.g. MLPs, ResNets, CNNs, etc.)

How many parameters should the network f have so that vanilla methods like SGD can converge to a global optima $\theta^\star$?

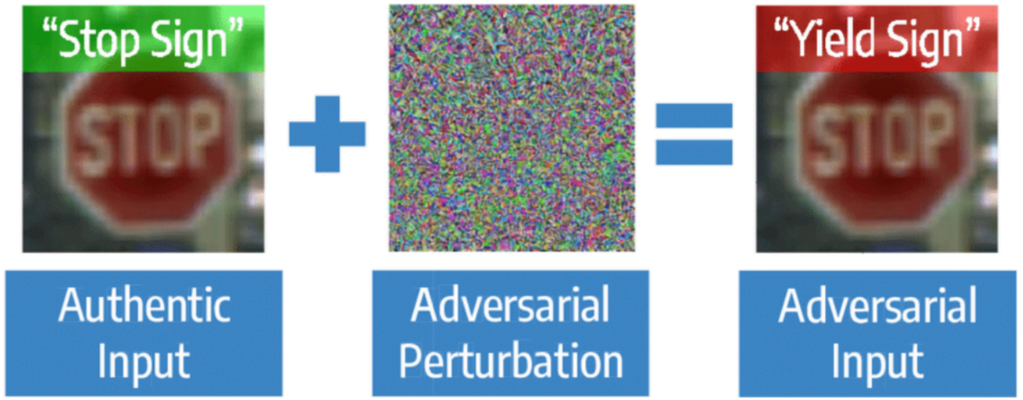
Table 3: Over-parameterization conditions for the convergence analysis of neural network under gradient descent training with squared loss. L is the depth of the network.

	Model	Depth	Initialization	Activation	Width
Allen-Zhu et al. [2019a]	FCNN/CNN	Deep	NTK	ReLU	$\Omega(N^{24}L^{12})$
Du et al. [2019a]	FCNN/CNN	Deep	NTK	Smooth	$\Omega(N^4 2^{\mathcal{O}(L)})$
Oymak and Soltanolkotabi [2020]	FCNN	Shallow	Standard Gaussian	ReLU	$\Omega(N^2)$
Zou and Gu [2019]	FCNN	Deep	He	ReLU	$\Omega(N^8L^{12})$
Du et al. [2019b]	FCNN	Shallow	NTK	ReLU	$\Omega(N^6)$
Nguyen [2021]	FCNN	Deep	LeCun	ReLU	$\Omega(N^3)$
Chen et al. [2021]	FCNN	Deep	NTK	ReLU	$\Omega(L^{22})$
Song et al. [2021]	FCNN	Shallow	He/Lecun	Smooth	$\Omega(N^{3/2})$
Bombari et al. [2022]	FCNN	Deep	He/LeCun	Smooth	$\Omega(\sqrt{N})$
Allen-Zhu et al. [2019b]	RNN	-	NTK	ReLU	$\Omega(N^c), c > 1$
Hron et al. [2020]	Transformer	Deep	NTK	ReLU	-
Yang [2020]	Transformer	Deep	NTK	Softmax+ReLU	-
Our	Transformer	Shallow	Table 1	Softmax+ReLU	$\Omega(N)$

*On the Convergence of Encoder-only Shallow Transformers (NeurIPS'23)

Ch. 2: From MIN to MIN-MAX

Rise of Multi-Agent Learning Applications



https://shorturl.at/e7Xbw

https://shorturl.at/krf6V

https://shorturl.at/Opeki

https://shorturl.at/DH09f

https://shorturl.at/l0g1p

~~MIN~~ MIN-MAX Optimization

- We are now modelling multiple agents learning and making decisions in a non-stationary environment that can react to these decisions. For example,
  - Agents having conflicting interests/objectives
  - Adversaries that can change/corrupt the data/distribution (label noise, distribution shifts)
  - Enforce constraints on learnt models such as those relating to causal inference, privacy, and fairness (and more).
- MIN can now be viewed as a *Single-Agent Learning* problem.

~~MIN~~ MIN-MAX Optimization

- We are now modelling multiple agents learning and making decisions in a non-stationary environment that can react to these decisions. For example,
  - Ag
  - A
    d
  - E
    inference, privacy, and fairness (and more).
- MIN can now be viewed as a *Single-Agent Learning* problem.

However, making Gradient Descent analogs "work" for MIN-MAX problems is hard due to "cycling" issues (more on this soon).

Natural Question: How big a neural network should be so that vanilla methods like ~~SGD~~ AltGDA can converge to ~~global optima~~ a saddle point?

More formally, we want to know…

$$(\theta^\star, \phi^\star) \in \arg\min_{\theta \in \mathbb{R}^m} \arg\max_{\phi \in \mathbb{R}^n} \; \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x;\theta), G(x';\phi)) \right]$$

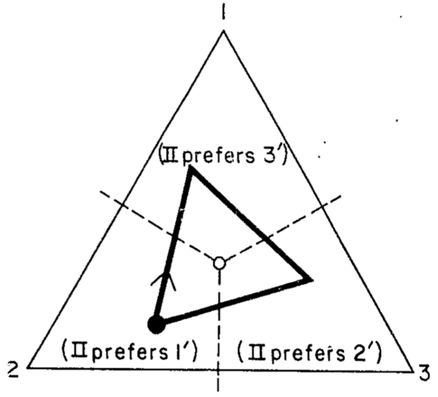
- Loss function $L : \mathbb{R}^{d_{out}^{(F)}} \times \mathbb{R}^{d_{out}^{(G)}} \to \mathbb{R}_{\geq 0}$
- data distribution $P_{xx'}$
- Neural networks $F(\,\cdot\,;\theta) : \mathbb{R}^{d_{in}^{(F)}} \to \mathbb{R}^{d_{out}^{(F)}}$, $G(\,\cdot\,;\phi) : \mathbb{R}^{d_{in}^{(G)}} \to \mathbb{R}^{d_{out}^{(G)}}$ (e.g. MLPs, ResNets, CNNs, etc.)

How many parameters should the networks F, G have so that vanilla methods like AltGDA can converge to a saddle point $(\theta^\star, \phi^\star)$?

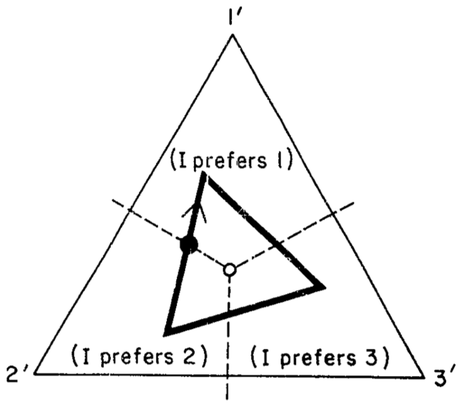
Can we just expect AltGDA to converge to saddle points (just like in the case of (S)GD for global optima)?

No… Hard to avoid “cycles”

- [BGP20] Finite Regret and Cycles with Fixed Step-Size via Alternating Gradient Descent-Ascent
  - Bilinear Matrix Games known to exhibit cycling behaviour (AltGDA with fixed step-sizes)
- Shapley (1964)* proved that in the game pictured here (a nonzero-sum version of Rock, Paper, Scissors), if the players start by choosing (a, B), the play will cycle indefinitely.



I's strategy space



II's strategy space

	A	*B*	*C*
a	0, 0	2, 1	1, 2
b	1, 2	0, 0	2, 1
c	2, 1	1, 2	0, 0

**Some topics in Two-Person Games (1964)*

63-4-1

MEMORANDUM
RM-3672-PR
JUNE 1963

SOME TOPICS IN TWO-PERSON GAMES

Lloyd S. Shapley

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

SUMMARY

This Memorandum consists of several loosely-related essays on the theory of finite, two-person games. The topics covered are, in brief, (1) the block decomposition of symmetric games, (2) saddlepoints in matrices having submatrices with saddlepoints, (3) generalized saddlepoints and "order matrices," (4) the existence of values in games with almost-perfect information, and (5) the nonconvergence of "ficitious play" in non-zero-sum games. Throughout, there is an emphasis on features of the theory that depend only on the ordering of the payoffs, as opposed to their numerical values.

No… Hard to avoid “cycles”

- Cycling in Adversarial learning (2018)
  - In a long run, every FTRL exhibits Poincaré Recurrence wandering around the equilibrium in a zero sum game.
- Training GANs with Optimism (2018)
  - Optimistic Mirror Descent exhibits the last-iterate convergence property in a zero sum game.
- The Limit Points of (Optimistic) Gradient Descent in Min-Max Optimization (2018)
  - The limits points of OMD are a superset of the local min-max solutions in a zero sum game.

Moreover, let's recall that…

Generative Adversarial Networks [Goodfellow et al. 16]

$$\arg\min_{\boldsymbol{\theta}^{(G)}} \max_{\boldsymbol{\theta}^{(D)}} V\left(\boldsymbol{\theta}^{(D)}, \boldsymbol{\theta}^{(G)}\right).$$

The minimax game is mostly of interest because it is easily amenable to theoretical analysis. Goodfellow *et al.* (2014b) used this variant of the GAN game to show that learning in this game resembles minimizing the Jensen-Shannon divergence between the data and the model distribution, and that the game converges to its equilibrium if both players' policies can be updated directly in function space. In practice, the players are represented with deep neural nets and updates are made in parameter space, so these results, which depend on convexity, do not apply.

So let’s focus on a class of min-max games that capture as many of the current deep learning applications as possible!

So let’s focus on a class of min-max games that capture as many of the current deep learning applications as possible and yet avoid cycles!

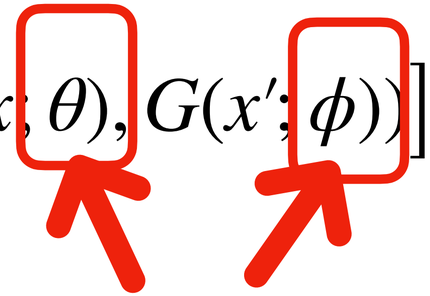
So let’s fo ames that
capture a learning
applicatio d cycles!

Hidden Games

Chapter 3: Hidden(-Convex-Concave) Games

Hidden(-Convex/Concave) Games

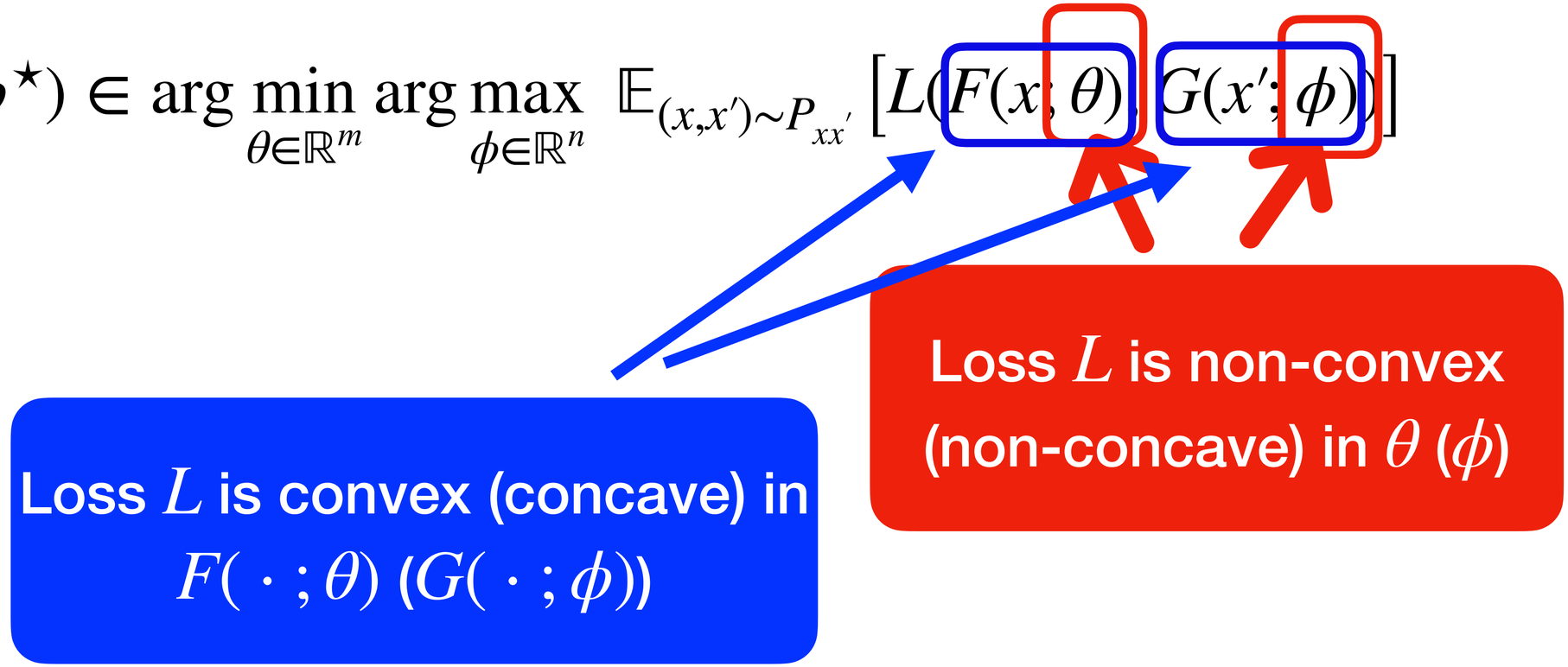
$$(\theta^\star, \phi^\star) \in \arg\min_{\theta \in \mathbb{R}^m} \arg\max_{\phi \in \mathbb{R}^n} \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x; \theta), G(x'; \phi)) \right]$$



Loss L is non-convex (non-concave) in θ (ϕ)

Hidden(-Convex/Concave) Games

$$(\theta^\star, \phi^\star) \in \arg\min_{\theta \in \mathbb{R}^m} \arg\max_{\phi \in \mathbb{R}^n} \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x; \theta), G(x'; \phi)) \right]$$



Convergence Results for Hidden Games

Poincaré Recurrence, Cycles and Spurious Equilibria in Gradient-Descent-Ascent for Non-Convex Non-Concave Zero-Sum Games

Authors: Lampros Flokas, Emmanouil-Vasileios [illegible]akis-Gka[illegible]ounis, Georgios Piliouras

Solving Min-Max Optimization with Hidden Structure via Gradient Descent Ascent

Authors: Lampros Flokas, Emmanouil-Vasil[illegible]s Vlata[illegible]-Gkaragkounis, Georgios Piliouras

Exploiting hidden structures in non-convex games for convergence to Nash equilibrium

Authors: Iosif Sakos, Emmanouil-Vasileios Vlat[illegible]s-Gkar[illegible]kounis, Panayotis Mertikopoulos, Georgios Piliouras

Global Convergence and Variance-Reduced Optimization for a Class of Nonconvex-Nonconcave Minimax Problems

Authors: Junchi Yang, Negar Kiyavash, Niao He

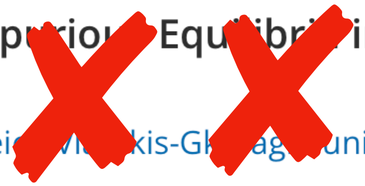
Solving Zero-Sum Convex Markov Games

Authors: Fivos Kalogiannis, Emmanouil-Vas[illegible]ios Vla[illegible]kis-Gkaragkounis, Ian Gemp, Georgios Piliouras

Convergence Results for Hidden Games

Poincaré Recurrence, Cycles and Spurious Equilibria in Gradient-Descent-Ascent for Non-Convex Non-Concave Zero-Sum Games

Authors: Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Georgios Piliouras



continuous dynamics

Solving Min-Max Optimization with Hidden Structure via Gradient Descent Ascent

Authors: Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Georgios Piliouras

GDA

Exploiting hidden structures in non-convex games for convergence to Nash equilibrium

Authors: Iosif Sakos, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Panayotis Mertikopoulos,

PHGD

Global Convergence and Variance-Reduced Optimization for a Class of Nonconvex-Nonconcave Minimax Problems

Authors: Junchi Yang, Negar Kiyavash, Niao He

AltGDA

Solving Zero-Sum Convex Markov Games

Authors: Fivos Kalogiannis, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Ian Gemp, Georgios Piliouras

General, constrained case

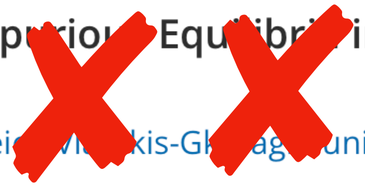
Also check out (12:00 — 12:30):

“Solving Hidden Monotone Variational Inequalities with Surrogate Losses”

Convergence Results for Hidden Games

Poincaré Recurrence, Cycles and Spurious Equilibria in Gradient-Descent-Ascent for Non-Convex Non-Concave Zero-Sum Games

Authors: Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Georgios Piliouras



continuous dynamics

Solving Min-Max Optimization with Hidden Structure via Gradient Descent Ascent

Authors: Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Georgios Piliouras

GDA

Exploiting hidden structures in non-convex games for convergence to Nash equilibrium

Authors: Iosif Sakos, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Panayotis Mertikopoulos,

PHGD

Global Convergence and Variance-Reduced Optimization for a Class of Nonconvex-Nonconcave Minimax Problems

Authors: Junchi Yang, Negar Kiyavash, Niao He

AltGDA

Solving Zero-Sum Convex Markov Games

Authors: Fivos Kalogiannis, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Ian Gemp, Georgios Piliouras

General, constrained case

Let’s take a closer look at hidden (convex) optimization

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \qquad (\star)$$

- Consider the following objective $F(\theta)$ in $(\star)$ above:

$$F(\theta) = \|A\theta - b\|^2$$

$$H(\cdot) := \|\cdot\|^2$$

$$c(\theta) := A\theta - b$$

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \qquad (\star)$$

- Consider the following objective $F(\theta)$ in $(\star)$ above:

~~$H(\cdot) := \|\cdot\|^2$~~ $H(\cdot) :=$ μ-strongly-convex

~~$c(\theta) := A\theta - b$~~ $c(\theta) :=$ invertible operator

- This is now a hidden-convex optimization problem

Hidden Convex Optimization (Formally)

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \qquad (\star)$$

This problem is (μ_c, μ_H)-*hidden convex* if the following hold true:

- Domain $\mathcal{U} = c(\Theta)$ is convex. Function $H : \mathcal{U} \to \mathbb{R}$ ($\mu_H \geq 0$) satisfies the following*

$$H((1-\lambda)u + \lambda v) \leq (1-\lambda)H(u) + \lambda H(v) - \frac{(1-\lambda)\lambda\mu_H}{2}\|u - v\|^2 \quad \forall u, v \; \forall \lambda \in [0,1]$$

- The map $c : \Theta \to \mathcal{U}$ is invertible. There exists $\mu_c > 0$ s.t.

$$\|c(\theta) - c(\theta')\| \geq \mu_c \|\theta - \theta'\| \quad \forall \theta, \theta' \in \Theta$$

*and convex reformulation of ($\star$) admits a solution $u^\star \in \mathcal{U}$)

Hidden Convex Optimization (Formally)

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \qquad (\star)$$

This problem is (μ_c, μ_H)-*hidden convex* if the following hold true:

- Domain $\mathcal{U} = c(\Theta)$ is convex. Function $H : \mathcal{U} \to \mathbb{R}$ ($\mu_H \geq 0$) satisfies the following*

$$H((1-\lambda)u + \lambda v) \leq (1-\lambda)H(u) + \lambda H(v) - \frac{(1-\lambda)\lambda\mu_H}{2}\|u - v\|^2 \quad \forall u, v \; \forall \lambda \in [0,1]$$

Strong convexity of the "latent" strongly convex landscape

- The map $c : \Theta \to$ [illegible] > 0 s.t.

$$\|c(\theta) - c(\theta')\| \geq \mu_c \|\theta - \theta'\| \quad \forall \theta, \theta' \in \Theta$$

Lipschitzness of the inversion $\approx$ condition-number of $c^{-1}(\cdot)$

*and convex reformulation of $(\star)$ admits a solution $u^\star \in \mathcal{U}$)

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

E.1. Globally optimal solution

The following proposition suggests that every stationary point of a hidden convex function is a global minima.

Proposition 7 *Let $F(\cdot)$ be hidden convex and $\bar{x} \in \mathcal{X}$ be its stationary point. If the map $c(\cdot)$ is differentiable at $\bar{x}$, then $\bar{x}$ is a global solution for (3), i.e., $F(\bar{x}) \leq F(x)$ for any $x \in \mathcal{X}$.*

Proposition 8 *Let $F(\cdot)$ be differentiable, hidden strongly convex ($\mu_H > 0$), and the map $c(\cdot)$ be differentiable on $\mathcal{X}$, then the optimization problem satisfies the global KŁ condition.*

$$\min_{h_x \in \partial \delta_{\mathcal{X}}(x)} \|\nabla F(x) + h_x\|^2 \geq 2\mu_H \mu_c^2 \left(F(x) - F^*\right) \qquad \text{for all } x \in \mathcal{X}. \tag{13}$$

Stochastic Optimization under Hidden Convexity (2023)

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

E.1. Globally optima

The following propos … en convex function is a global minima.

The smaller the gradient
Closer to the minimum

Proposition 7 *Let $F(\cdot)$ be hidden conv x and $\bar{x} \in \mathcal{X}$ be its stationary point. If the map $c(\cdot)$ is differentiable at $\bar{x}$, then $\bar{x}$ is a global so ution for (3), i.e., $F(\bar{x}) \leq F(x)$ for any $x \in \mathcal{X}$.*

Proposition 8 *Let $F(\cdot)$ be differe iable, hidden strongly convex ($\mu_H > 0$), and the map $c(\cdot)$ be differentiable on $\mathcal{X}$, then the optim zation problem satisfies the global KŁ condition.*

$$\min_{h_x \in \partial \delta_{\mathcal{X}}(x)} \|\nabla F(x) + h_x\|^2 \geq 2\mu_H \mu_c^2 (F(x) - F^*) \quad \text{for all } x \in \mathcal{X}. \tag{13}$$

Stochastic Optimization under Hidden Convexity (2023)

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

E.1. Globally optimal so

The following propositio hidden convex function is a global minima.

Greater latent convexity,
smoother inversion,
nearer the minimum.

Proposition 7 *Let $F(\cdot)$ be hidden convex and $\bar{x} \in \mathcal{X}$ be its stationary point. If the map $c(\cdot)$ is differentiable at $\bar{x}$, then $\bar{x}$ is a global solution f (3), i.e., $F(\bar{x}) \leq F(x)$ for any $x \in \mathcal{X}$.*

Proposition 8 *Let $F(\cdot)$ be differentiable, hi strongly convex ($\mu_H > 0$), and the map $c(\cdot)$ be differentiable on $\mathcal{X}$, then the optimization problem satisfies the global KŁ condition.*

$$\min_{h_x \in \partial \delta_{\mathcal{X}}(x)} \|\nabla F(x) + h_x\|^2 \geq 2\mu_H \mu_c^2 (F(x) - F^*) \qquad \text{for all } x \in \mathcal{X}. \tag{13}$$

Stochastic Optimization under Hidden Convexity (2023)

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

F.1. Glob

Hidden Strongly Convex $\Longrightarrow$ PŁ-condition

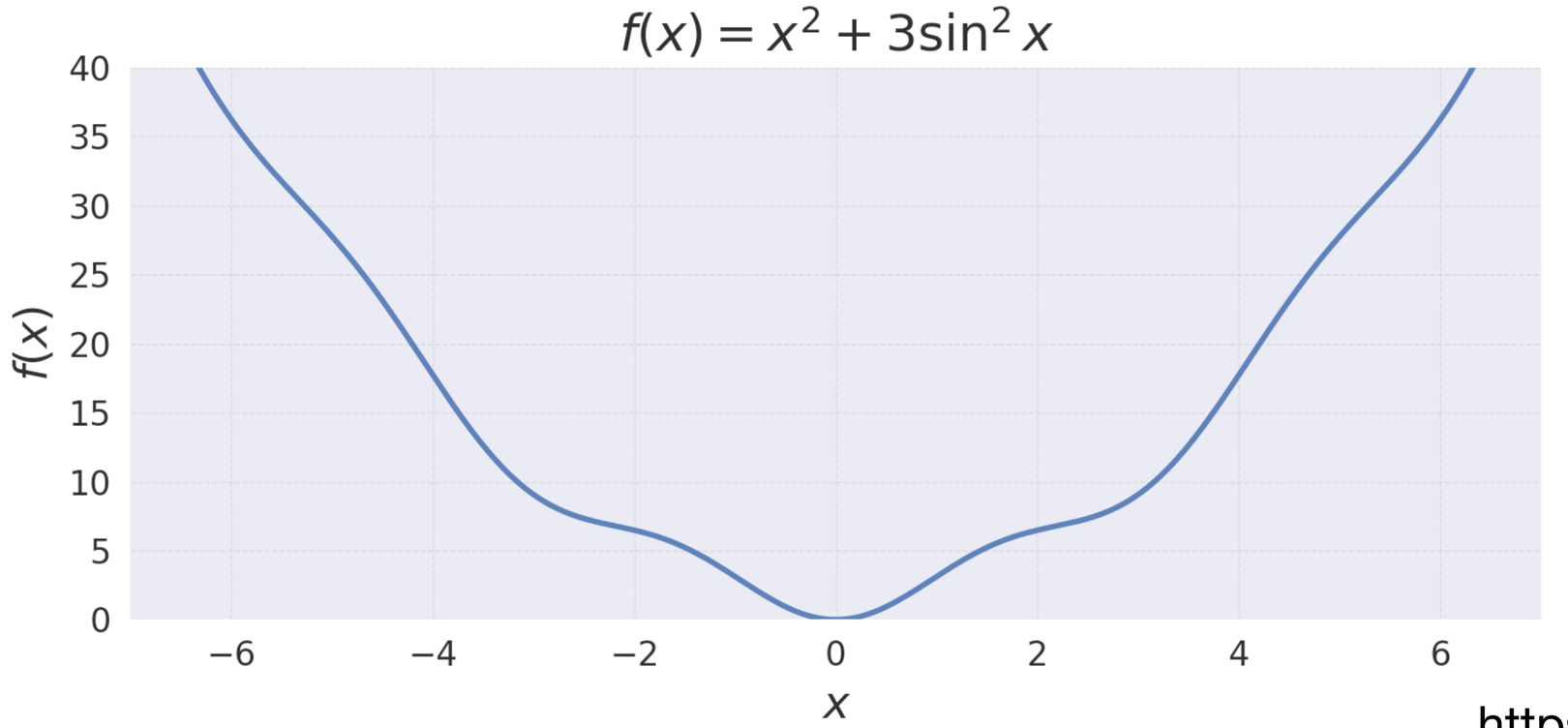
Let $F(\cdot)$ be differentiable, hidden strongly convex ($\mu_H > 0$), and the map $c(\cdot)$ be differentiable on $\mathcal{X}$, then the optimization problem satisfies the global KŁ condition.

$$\min_{h_x \in \partial \delta_{\mathcal{X}}(x)} \|\nabla F(x) + h_x\|^2 \geq 2\mu_H \mu_c^2 \left(F(x) - F^*\right) \qquad \text{for all } x \in \mathcal{X}. \tag{13}$$

Stochastic Optimization under Hidden Convexity (2023)

Quick Primer on PŁ-condition

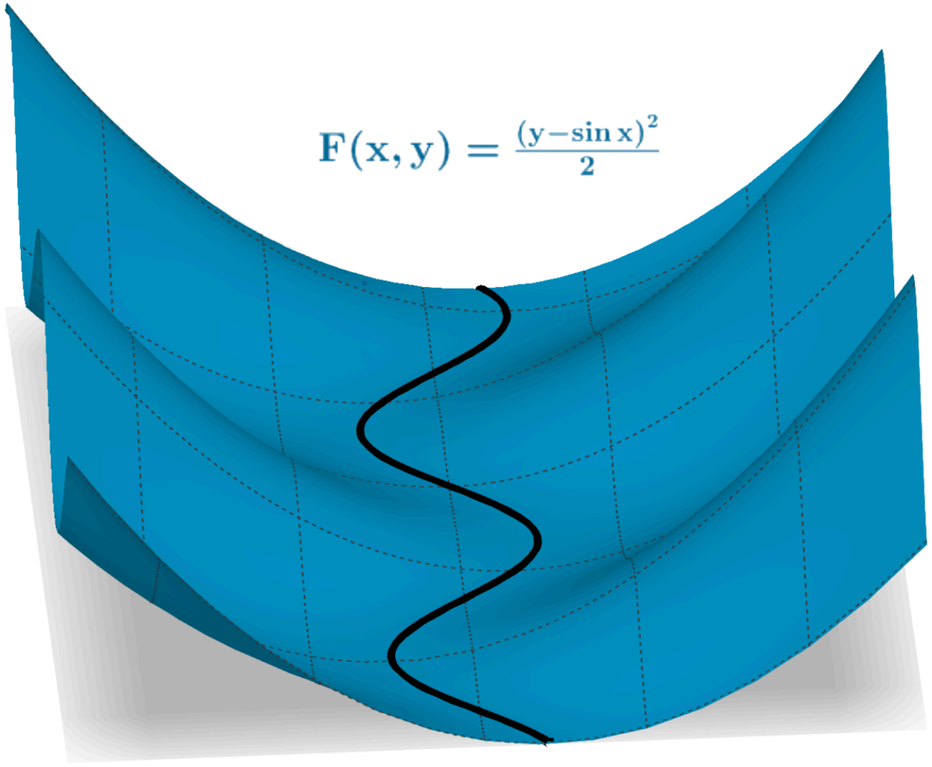
$$f(x) - f(x^\star) \leq \frac{1}{2\mu}\|\nabla f(x)\|_2^2 \qquad \text{(PŁ-condition)}$$



https://shorturl.at/gfQd5

Quick Primer on PŁ-condition

$$f(x) - f(x^\star) \leq \frac{1}{2\mu}\|\nabla f(x)\|_2^2 \qquad \text{(PŁ-condition)}$$



https://arxiv.org/pdf/2107.10123

Quick Primer on PŁ-condition

$$f(x) - f(x^\star) \leq \frac{1}{2\mu}\|\nabla f(x)\|_2^2 \qquad \text{(PŁ-condition)}$$

- "Generalisation" of *strong convexity* condition whilst still guaranteeing linear convergence rates for *Gradient Descent* method
- μ-strongly-convex $\implies$ μ-PŁ condition
- μ-PŁ condition $\not\implies$ μ-strongly-convex (*or* convex!)

Back to Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

$$f(x) - f(x^\star) \leq \frac{1}{2\mu} \|\nabla f(x)\|_2^2 \qquad \text{(PŁ-condition)}$$

What we have is:

- (μ_c, μ_H)-hidden-strongly-convex $\implies$ $\mu_c^2 \mu_H$-PŁ condition

Hidden Convex Optimization

$$\min_{\theta \in \Theta} F(\theta) := H(c(\theta)) \quad ((\mu_c, \mu_H) - \text{hidden convex})$$

Assumption 2. The singular values of the Jacobian $\mathbf{J}(\theta)$ of the representation $c(\theta)$ are bounded as

$$\mu_c \equiv \sigma_{\min}^2 \leq \text{eig}(\mathbf{J}(\theta)\mathbf{J}(\theta)^\intercal) \leq \sigma_{\max}^2 \tag{16}$$

for some $\sigma_{\min}, \sigma_{\max} \in (0, \infty)$ and for all $\theta \in \Theta$.

With all this in hand, we begin by studying the behavior of (PHGD) in games with a hidden monotone structure.

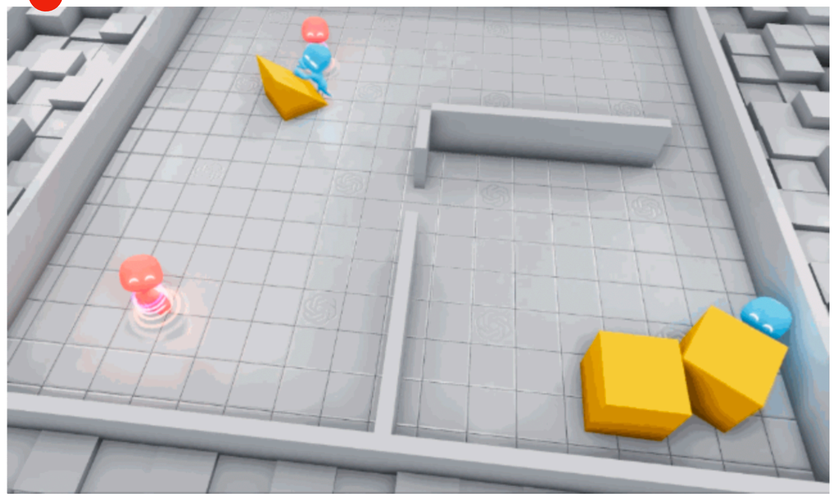
Intuition

Assumption 2. The singular values of the Jacobian $\mathbf{J}(\theta)$ of the representation $c(\theta)$ are bounded as

$$\mu_c \equiv \sigma_{\min}^2 \leq \text{eig}(\mathbf{J}(\theta)\mathbf{J}(\theta)^\intercal) \leq \sigma_{\max}^2 \tag{16}$$

Multi-Agent RL & Neural Nets as Strategy Producers

If $\sigma_{\min} = 0$ then NN-map loses part(s) of the Strategy Space!!!



- **Key Observation:**

  Many interesting games (i.e., stochastic & external form) can be expressed as *Classical games with **very large action space**!*

- **Modern Approach:**

  Substitute every agent with a neural network map!!!

Checking Hidden (Strong) Convexity

Q-1. Is Assumption 2 correct?

Checking Hidden (Strong) Convexity

~~Q-1. Is Assumption 2 correct?~~

Q-1. Is it true that $\sigma^2_{\min}(\mathbf{J}(\theta)) > 0$?

Checking Hidden (Strong) Convexity

Q-2. Assumption 2 holds for all parameter iterates?

Checking Hidden (Strong) Convexity

~~Q-2. Assumption 2 holds for all parameter iterates?~~

Q-2. Is $\sigma^2_{\min}(\mathbf{J}(\theta_t)) > 0 \quad \forall t \in \{0, \ldots, T\}$

Checking Hidden (Strong) Convexity

Q-1. Is it true that $\sigma^2_{\min}(\mathbf{J}(\theta)) > 0$?

Q-2. Is $\sigma^2_{\min}(\mathbf{J}(\theta_t)) > 0 \quad \forall t \in \{0, \ldots, T\}$

How are Q-1 and Q-2 connected with “initialisation” and “architecture” of Neural Networks?

Convergence to saddle points

Q-3. How do we know that AltGDA converges to saddle points?

Convergence to saddle points

~~Q-3. How do we know that AltGDA converges to saddle points?~~

Q-3. For AltGDA, do $\lim_{t \to \infty} (\theta_t, \phi_t) = (\theta^\star, \phi^\star)$ where

$$L(\theta^\star, \phi) \leq L(\theta^\star, \phi^\star) \leq L(\theta, \phi^\star) \; \forall \theta, \phi$$

Fact: AltGDA converges to saddle points for Min-Max objectives satisfying two-sided PŁ-condition*

**Global Convergence to and Variance Reduction for a Class of Nonconvex-Nonconcave Minimax Problems (NeurIPS'20)*

Convergence to saddle points

~~Q-3. How do we know that AltGDA converges to saddle points?~~

Q-3. For AltG

Hidden Strongly Convex(/Concave) $\Longrightarrow$ PŁ-condition

Fact: A points for Min-Max objectives satisfying two-sided PŁ-condition*

**Global Convergence to and Variance Reduction for a Class of Nonconvex-Nonconcave Minimax Problems (NeurIPS'20)*

Chapter 4: Our Results

Informally, we show that…

Informal Theorem (Theorem 3.7)**.** *There exists a decentralized, gradient-based method (eq.* (Alt-GDA)*) that computes, with high probability under suitable Gaussian random initialization, an ϵ-approximate Nash equilibrium for any $\epsilon > 0$ in broad class of hidden convex-concave zero-sum games, where each player's strategy is parameterized by a sufficiently wide two-layer neural network.*

- *The number of iterations required scales as*

$$O\left(\text{poly}\left(\frac{1}{width_1}, \frac{1}{width_2}, \frac{1}{n}, d_{input}\right) \times \frac{L^3}{\mu^3} \times \log\left(\frac{1}{\epsilon}\right)\right),$$

  where $width_1, width_2$ are the hidden layer widths, n is the number of training samples, d_{input} is the input dimension, L is the smoothness constant, and μ is the strong convexity modulus of the latent objective.

- *This guarantee holds provided the network* $width_{1,2} = \tilde{\Omega}\left(\mu^2 \frac{n^3}{d_{input}}\right)$.

For Hidden-Strongly-Convex-Strongly-Concave Games

Informally, we show that…

Informal Theorem (Theorem 3.7)**.** *There exists a decentralized, gradient-based method (eq.* (Alt-
GD
an
ϵ
um
g
rk.

We see a “gap” between overparameterization needed for MIN and MIN-MAX:

- MIN [SKPEC21*]: $\tilde{\Omega}(n^{1.5})$ overparameterization needed
- MIN-MAX [Ours]: $\tilde{\Omega}(n^{3})$ overparameterization needed

put
us

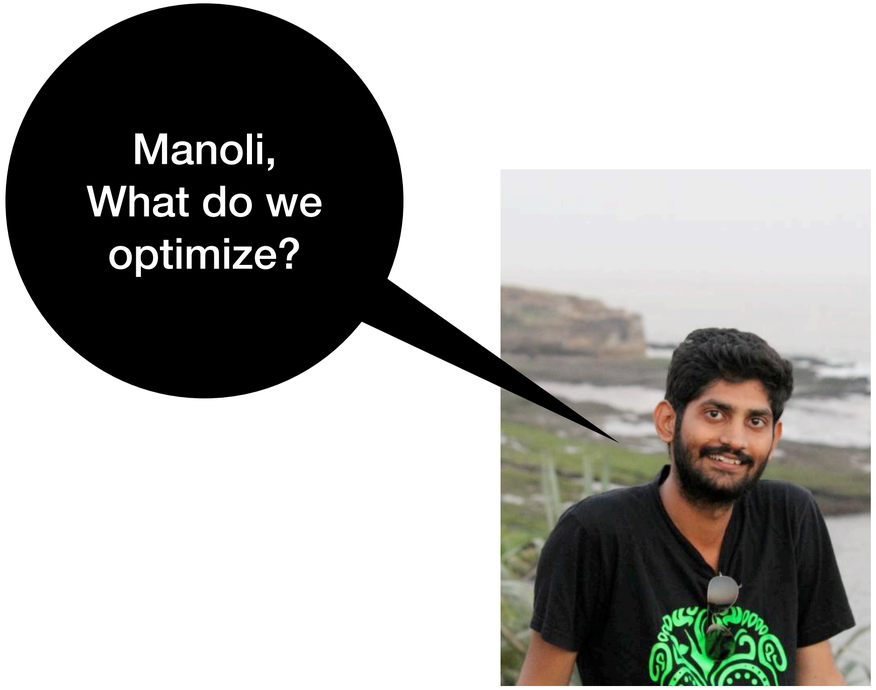
- *This guarantee holds provided the network width*$_{1,2} = \Omega\left(\mu^2 \frac{n}{d_{input}}\right)$.

*Subquadratic Overparameterization for Shallow Neural Networks (2021)

A closer look at the Hidden Games we consider

The Two Hidden Min-Max Settings

The Two Hidden Min-Max Settings



The Two Hidden Min-Max Settings

What do you mean?
Fix the dataset and find
Minmax NN parameters

Manoli,
What do we
optimize?

The Two Hidden Min-Max Settings

The Two Hidden Min-Max Settings

The Two Hidden Min-Max Settings

Setting 1: Input-Optimization Games

$$\min_{x_{Alice} \in \mathscr{D}_F} \max_{x_{Bob} \in \mathscr{D}_G} L(F(x_{Alice}; \theta), G(x_{Bob}; \phi))$$

- Network parameters θ and ϕ are fixed (e.g. random initializations)
- Optimizing over inputs — Adversarial example generation*
- Examples: Adversarial Attack over Data Transformations, Universal Perturbation over Multiple Examples, Ensemble Attack over Multiple Models

**Adversarial attack generation empowered by min-max optimization (NeurIPS'21)*

The Two Hidden Min-Max Settings

Setting 1: Input-Optimization Games

$$\min_{x_{Alice} \in \mathscr{D}_F} \max_{x_{Bob} \in \mathscr{D}_G} L(F(x_{Alice}; \theta), G(x_{Bob}; \phi))$$

Example B.6 (Ensemble Attack over Multiple Models)**.** Given K machine learning models $\{\mathcal{M}_i\}_{i=1}^K$, the goal is to find a universal perturbation δ that simultaneously fools all models. The corresponding input-optimization game reads:

$$\min_{\delta \in \mathcal{X}} \max_{w \in \mathcal{P}} \sum_{i=1}^{K} w_i f(\delta; x_0, y_0, \mathcal{M}_i) - \frac{\gamma}{2} \|w - 1/K\|_2^2,$$

where w encodes the relative difficulty of attacking each model, and γ is a regularization parameter.

**Adversarial attack generation empowered by min-max optimization (NeurIPS'21)*

The Two Hidden Min-Max Settings

Setting 1: Input-Optimization Games

$$\min_{x_{Alice} \in \mathscr{D}_F} \max_{x_{Bob} \in \mathscr{D}_G} L(F(x_{Alice}; \theta), G(x_{Bob}; \phi))$$

Example B.8 (Adversarial Attack over Data Transformations)**.** Consider robustness against transformations (e.g., rotations, translations) applied to the inputs. Given categories of transformations $\{p_i\}$, the optimization reads:

$$\min_{\delta \in \mathcal{X}} \max_{w \in \mathcal{P}} \sum_{i=1}^{K} w_i \mathbb{E}_{t \sim p_i} \left[f(t(x_0 + \delta); y_0, \mathcal{M}) \right] - \frac{\gamma}{2} \|w - 1/K\|_2^2,$$

where t denotes a random transformation sampled from p_i. When $w = 1/K$, this recovers the expectation-over-transformation (EOT) setup.

**Adversarial attack generation empowered by min-max optimization (NeurIPS'21)*

The Two Hidden Min-Max Settings

Setting 2: Neural Games

$$\min_{\theta \in \mathbb{R}^m} \max_{\phi \in \mathbb{R}^n} \; \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x;\theta), G(x';\phi)) \right]$$

Example B.1 (Generative Adversarial Networks (GANs)). A *Generative Adversarial Network* (GAN) formulates a two-player minimax game where the generator G_θ seeks to produce samples that resemble a reference distribution p_{data}, while the discriminator D_ϕ attempts to distinguish generated samples from real data. The corresponding min-max problem reads:

$$\min_{\theta} \max_{\phi} \quad \Psi(\theta, \phi) := \mathbb{E}_{x \sim p_{\text{data}}} [\log D_\phi(x)] + \mathbb{E}_{x \sim p_\theta} [\log(1 - D_\phi(x))] \, .$$

The Two Hidden Min-Max Settings

Setting 2: Neural Games

$$\min_{\theta \in \mathbb{R}^m} \max_{\phi \in \mathbb{R}^n} \; \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x;\theta), G(x';\phi)) \right]$$

Example B.2 (Domain-Invariant Representation Learning (DIRL)). Domain adaptation aims to train models that generalize across different domains, despite distribution shifts between training (source) and deployment (target) environments. A popular approach [42] involves learning representations that are: (i) predictive of labels in the source domain, and (ii) invariant to the domain classifier distinguishing source versus target samples. This leads to the following min-max problem:

$$\min_{\theta_f, \theta_g} \max_{\theta_{f'}} \mathbb{E}_{(x,y) \sim P_{\text{source}}} \left[\ell(f_{\theta_f}(g_{\theta_g}(x)), y) \right] - \lambda \mathbb{E}_{(x,y') \sim P_{\text{mix}}} \left[\ell(f'_{\theta_{f'}}(g_{\theta_g}(x)), y') \right],$$

Back to our results

Setting 1: Input-Optimization Games

$$\min_{x_{Alice} \in \mathcal{D}_F} \max_{x_{Bob} \in \mathcal{D}_G} L($$

Open Question from NeurIPS 2019

- We consider bilinear objective

$$F(x_{Alice}; \theta)^\top A G(x_{Bob}; \phi)$$

We show that w.h.p. AltGDA converges to $\epsilon-$saddle point in $O(\text{poly}(1/\epsilon))$ if the Gaussian-randomly-initialized mappings F and G (1-hidden-layer neural networks) satisfy

$$\sigma_{F/G}^2 = \tilde{\Theta}_{1/\sigma_{\max}(A)}\left(\text{poly}(\frac{1}{width_F})\right)$$

Back to our results

Setting 2: Neural Games

$$\min_{\theta \in \mathbb{R}^m} \max_{\phi \in \mathbb{R}^n} \mathbb{E}_{(x,x') \sim P_{xx'}} \left[L(F(x;\theta), G(x';\phi)) \right]$$

- We consider separable latent min-max objectives of the form

$$\mathbb{E}_{(x,x') \sim P_{xx'}} \left[\ell_F(F(x;\theta), x) + F(x;\theta)^\top A G(x';\phi) - \ell_G(G(x';\phi), x') \right]$$

where ℓ_F (ℓ_G) is hidden-strongly-convex (hidden-strongly-concave).

- Both F and G are 1-hidden-neural network with Gaussian random initializations.

Back to our results

Setting 2: Neural Games

$$\min_{\theta\in\mathbb{R}^m}\max_{\phi\in\mathbb{R}^n}\ \mathbb{E}_{(x,x')\sim P_{xx'}}\left[L(F(x;\theta),G(x';\phi))\right] \qquad \text{(HSCSC}^*\text{)}$$

We show that w.h.p. AltGDA converges to a saddle point if the Gaussian-random initializations and hidden-layer width of the networks F and G satisfy

$$\sigma_{1,F/G}\cdot\sigma_{2,F/G}\lesssim\frac{1}{\sqrt{d_{in,F/G}\cdot \text{width}_{F/G}}} \quad \text{and} \quad \text{width}_{F/G}=\tilde{\Omega}\left(\mu_{\theta/\phi}^2\frac{n^3}{d_{in,F/G}}\right)$$

**HSCSC = Hidden-Strongly-Convex-Strongly-Concave*

Proof Outline

1. Choose Gaussian random initializations (θ_0, ϕ_0) such that the Jacobian for networks F and G is “well-conditioned” w.h.p.

2. Define radius R of a Euclidean ball $\mathscr{B}((\theta_0, \phi_0), R)$ such that the Jacobian remains well-conditioned within it.

3. Compute path length bound of AltGDA iterates (θ_t, ϕ_t) in terms of P_0, a special Lyapunov potential at time $t = 0$.

4. Find sufficient conditions on hidden layer width of networks F, G to ensure this path length is smaller than the ball radius R

Proof Outline (Remarks)

- Regarding (2.) and (4.):
  - Similar analysis in case of MIN is easier and relies on careful selection of step-size instead.
  - Staying within ball $\Longrightarrow$ μ-hidden-strongly-convex $\Longrightarrow$ PŁ-condition
  - Why bother? μ-hidden-strongly-convex $\Longrightarrow$ $\mu\sigma_{\min}^2(\mathbf{J}(\theta))$-PŁ-condition

Proof Outline (Remarks)

- Regarding (3.)—(4.):
  - Lyapunov Potential P_t for min-max objective $L(\theta, \phi)$ with saddle point $(\theta^\star, \phi^\star)$:

$$P_t := \left(\max_{\phi} L(\theta_t, \phi) - L(\theta^\star, \phi^\star) \right) + \lambda \left(\max_{\phi} L(\theta_t, \phi) - L(\theta_t, \phi_t) \right)$$

  - Intuitively, we are looking for (θ_0, ϕ_0) s.t. we are somewhat close to the saddle point to begin with.
  - Finding the sufficient width to ensure staying within $\mathscr{B}((\theta_0, \phi_0), R)$ crucially relies on the geometry of the input data $(\sigma_{\max}(X), \sigma_{\min}(X^{*t}))$

Proof Outline (Remarks)

- Regarding (3.)—(4.):

$$P_t := \left(\max_{\phi} L(\theta_t, \phi) - L(\theta^\star, \phi^\star) \right) + \lambda \left(\max_{\phi} L(\theta_t, \phi) - L(\theta_t, \phi_t) \right)$$

  - Controlling P_0 so that it's small boils down to requiring the following:

$$\|\nabla_\theta L(\theta_0, \phi_0)\| + \|\nabla_\phi L(\theta_0, \phi_0)\| \lesssim R^2$$

    - Recall that the min-max objective is HSCSC ($L(\theta, \phi) = L(F_\theta, G_\phi)$). Another reason for why Jacobian singular values appear in analysis. (Chain rule!)

Proof Outline (Remarks)

- Regarding (3)—(4.):
  - Ensuring potential P_0 is “small” boils down to the following:

$$\sigma_{\max}(\mathbf{J}(\theta_0)) \cdot \left(C_1\sigma_{\max}(X) + C_2\right) \lesssim \sigma^2_{\min}(\mathbf{J}(\theta_0))$$

$$\iff \boxed{\text{width} \gtrsim \frac{n\mu^2\sigma^6_{\max}(X)}{\sigma^4_{\min}(X^{*t})}} \qquad (n \simeq d^t_{in}; t \geq 2)$$

$$\iff \boxed{\text{width} = \tilde{\Omega}\left(\mu^2\frac{n^3}{d_{in}}\right)} \quad \left(\sigma_{\max}(X) \simeq \sqrt{\frac{n}{d_{in}}}; \sigma_{\min}(X^{*t}) \simeq 1\right)$$

Future Work

- The width (and hence the overparameterization) condition on the neural networks F, G is a **sufficient condition**. Is it also necessary?
- Analysis assumes differentiable activation functions (excluded ReLU, for example).
- Connect results with those for extensive-form games.
- Extend to Hidden MVIs for polyhedral settings

Thank you!