THE DYNAMICS OF ARTIFICIAL INTELLIGENCE

A STOCHASTIC APPROXIMATION VIEWPOINT

Παναγιώτης Μερτικόπουλος

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Τμήμα Μαθηματικών

( Οικονομικό Πανεπιστήμιο Αθηνών | Σεμινάριο Στατιστικής | 28 Απριλίου, 2023 )
Outline

1 Background & motivation

2 Preliminaries

3 Applications to minimization problems

4 Applications to min-max problems
About

- Hsieh, M & Cevher, *The limits of min-max optimization algorithms: convergence to spurious non-critical sets*, ICML 2021
- M, Papadimitriou & Piliouras, *Cycles in adversarial regularized learning*, SODA 2018
Stochastic approximation: from the 1950's...

Stochastic approximation

Find a root of a nonlinear system involving unknown functions, accessible only via noisy evaluations

Herbert Robbins & Sutton Monro

Jack Kiefer & Jacob Wolfowitz
...to the 2020's

Which person is fake?

https://thispersondoesnotexist.com
...to the 2020's

Which person is fake?

https://thispersondoesnotexist.com
## Generative adversarial networks

A generative adversarial network (GAN) is a type of neural network that comprises two models: a generator and a discriminator.

1. **Generator** $G$:
   - Takes a random input $Z_i \in \mathbb{R}^p$.
   - Generates a sample $X_i$.

2. **Discriminator** $D$:
   - Takes a sample $X_i$.
   - Determines whether the sample is real or fake.

The model likelihood is defined as:

$$L(G, D) = \prod_{i=1}^{N} D(X_i) \times \prod_{i=1}^{N} \left(1 - D(G(Z_i))\right)$$
Generative adversarial networks

\[ G \]
Generative adversarial networks

$Z_i \in \mathbb{R}^P$

Generator

$G$

$G(Z_i)$
**Generative adversarial networks**

- **Gaussian seed**: \( Z_i \in \mathbb{R}^p \)
- **Generator**: \( G \)
- **Discriminator**: \( D \)

Model likelihood:

\[
L(G, D) = \prod_{i=1}^{N} D(X_i) \times \prod_{i=1}^{N} (1 - D(G(Z_i)))
\]
Generative adversarial networks

Gaussian seed

$Z_i \in \mathbb{R}^p$

Generator

$G$

Discriminator

$D$

$X_i \in \mathbb{R}^d$

$G(Z_i)$

(compare)

$\prod_{i=1}^{N} D(X_i) \times \prod_{i=1}^{N} D(G(Z_i))$
Generative adversarial networks

\[ Z_i \in \mathbb{R}^p \]

\[ G \]

\[ G(Z_i) \]

\[ D \]

\[ X_i \in \mathbb{R}^d \]

\[ \text{loss} \]

\[ \text{true/fake} \]

\[ \text{compare} \]
**Generative adversarial networks**

![Diagram of generative adversarial networks](image)

- **Gaussian seed**: $Z_i \in \mathbb{R}^p$
- **Generator**: $G(Z_i)$
- **Discriminator**: $D(X_i) \in \mathbb{R}^d$

**Model likelihood**: 

$$L(G, D) = \prod_{i=1}^{N} D(X_i) \times \prod_{i=1}^{N} (1 - D(G(Z_i)))$$
**GAN training**

How to find good generators (\(G \in \mathcal{G}\)) and discriminators (\(D \in \mathcal{D}\))? 

**Discriminator:** maximize (log-)likelihood estimation

\[
\max_{D \in \mathcal{D}} \log L(G, D)
\]

**Generator:** minimize the resulting divergence

\[
\min_{G \in \mathcal{G}} \max_{D \in \mathcal{D}} \log L(G, D)
\]

Training a GAN \(\iff\) solving a min-max problem
Loss surfaces

Figure: The loss landscape of a deep neural network [Li et al., 2018]
Main question: what is the long-run behavior of first-order training methods?

In minimization problems:
- Do gradient methods converge to critical points?
- Are non-minimizers avoided?

In min-max problems / games:
- Do gradient methods converge to critical points?
- Are non-equilibrium sets avoided?
Outline

1 Background & motivation

2 Preliminaries

3 Applications to minimization problems

4 Applications to min-max problems
Mathematical formulation

Minimization problems

\[
\min_{x \in \mathcal{X}} f(x) \quad \text{(Opt)}
\]

Saddle-point problems

\[
\min_{x_1 \in \mathcal{X}_1} \max_{x_2 \in \mathcal{X}_2} f(x_1, x_2) \quad \text{(SP)}
\]
**Mathematical formulation**

**Minimization problems (stochastic)**

\[
\min_{x \in \mathcal{X}} f(x) = \mathbb{E}_\theta [F(x; \theta)] \quad \text{(Opt)}
\]

**Saddle-point problems (stochastic)**

\[
\min_{x_1 \in \mathcal{X}_1} \max_{x_2 \in \mathcal{X}_2} f(x_1, x_2) = \mathbb{E}_\theta [F(x_1, x_2; \theta)] \quad \text{(SP)}
\]
Problem formulation

Main difficulties:

- No convex structure
- Difficult to manipulate \( f \) in closed form

# technical assumptions later

# black-box oracle methods
**Problem formulation**

**Main difficulties:**

- No convex structure
- Difficult to manipulate \( f \) in closed form

Focus on **critical points:**

Find \( x^* \) such that \( g(x^*) = 0 \)

where \( g(x) \) is the problem’s **defining vector field:**

- **Gradient field** for (Opt):

\[
g(x) = \nabla f(x)
\]

- **Hamiltonian field** for (SP):

\[
g(x) = (\nabla_{x_1} f(x_1, x_2), -\nabla_{x_2} f(x_1, x_2))
\]

# Notation: \( x \leftarrow (x_1, x_2), \mathcal{X} \leftarrow \mathcal{X}_1 \times \mathcal{X}_2 \)
Assumptions

Blanket assumptions

- **Unconstrained problems:**
  \[\mathcal{X} = \text{finite-dimensional Euclidean space}\]

- **Existence of solutions:**
  \[\text{crit}(f) := \{x^* \in \mathcal{X} : g(x^*) = 0\} \text{ is nonempty}\]

- **Lipschitz continuity:**
  \[|f(x') - f(x)| \leq G\|x' - x\| \quad \text{for all } x, x' \in \mathcal{X}\]  \hspace{1cm} \text{(LC)}

- **Lipschitz smoothness:**
  \[\|g(x') - g(x)\| \leq L\|x' - x\| \quad \text{for all } x, x' \in \mathcal{X}\]  \hspace{1cm} \text{(LS)}
Stochastic approximation algorithms

Stochastic approximation template

\[ X_{n+1} = X_n - \gamma_n \hat{g}_n \]  \hspace{1cm} (SA)

where:
- \( X_n \in \mathbb{R}^d \) is the state of the method at epoch \( n = 1, 2, \ldots \)
- \( \gamma_n > 0 \) is a variable step-size parameter
- \( \hat{g}_n \in \mathbb{R}^d \) is a stochastic approximation of \( g(X_n) \)

Blanket assumptions

1. **Step-size sequence:**
   \[ \gamma_n \propto \gamma / n^p \]
   \# \( \gamma > 0, p \in [0, 1] \)

2. **Stochastic approximation:**
   \[ \hat{g}_n = g(X_n) + U_n + b_n \]
   where:
   - \( U_n = \hat{g}_n - \mathbb{E}[\hat{g}_n \mid \mathcal{F}_n] \) is the noise in the method
   - \( b_n = \mathbb{E}[\hat{g}_n \mid \mathcal{F}_n] - g(X_n) \) is the offset of the method
   \# \( \mathbb{E}[\|U_n\|^q \mid \mathcal{F}_n] \leq \sigma_n^q \)
   \# \( \mathbb{E}[\|b_n\| \mid \mathcal{F}_n] \leq B_n \)
Methods, I: Gradient descent/ascent

Gradient descent/ascent

\[ X_{n+1} = X_n - \gamma_n g(X_n) \]  

\[ \sigma_n = 0 \]

\[ B_n = 0 \]
Methods, II: Proximal point method

Proximal point method

\[ X_{n+1} = X_n - \gamma_n g(X_{n+1}) \]  

(PPM)

\[ x^+ \rightarrow x^\ast \]

✓ Deterministic:
\[ \sigma_n = 0 \]

⚠ Offset:
\[ B_n = \mathcal{O}(\gamma_n) \]
**Methods, III: Extra-gradient**

Extra-gradient \[ [\text{Korpelevich, 1976; Nemirovski, 2004}] \]

\[
X_{n+1} = X_n - \gamma_n g(X_{n+1/2}) \quad X_{n+1/2} = X_n - \gamma_n g(X_n)
\]

\[(\text{EG})\]

- **Deterministic:** \[ \sigma_n = 0 \]
- **Offset:** \[ B_n = O(\gamma_n) \]
### Methods, III: Extra-gradient

**Extra-gradient**

\[\begin{align*}
X_{n+1} &= X_n - \gamma_n g(X_{n+1/2}) \\
X_{n+1/2} &= X_n - \gamma_n g(X_n)
\end{align*}\]  

\textbf{Deterministic:} \quad \sigma_n = 0

\textbf{Offset:} \quad B_n = \mathcal{O}(\gamma_n)
Methods, III: Extra-gradient

Extra-gradient

\[ X_{n+1} = X_n - \gamma_n g(X_{n+1/2}) \quad X_{n+1/2} = X_n - \gamma_n g(X_n) \]  

(EG)

\[ x = x^* - \gamma g(x_{\text{lead}}) \]

\[ x + \gamma g(x) \]

\[ x_{\text{lead}} \]

\[ x^* \]

✓ Deterministic:
\[ \sigma_n = 0 \]

⚠ Offset:
\[ B_n = O(\gamma_n) \]

[Korpelevich, 1976; Nemirovski, 2004]
Methods, IV: Optimistic gradient

Optimistic gradient

\[ X_{n+1} = X_n - \gamma_n g(X_{n+1/2}) \quad X_{n+1/2} = X_n - \gamma_n g(X_{n-1/2}) \]  

(OG)

\[ x \]

\[ x \rightarrow x_{\text{lag}} \]

\[ x \rightarrow x_{\text{lead}} \]

Deterministic:
\[ \sigma_n = 0 \]

Offset:
\[ B_n = \mathcal{O}(\gamma_n) \]
**Methods, IV: Optimistic gradient**

**Optimistic gradient**

\[
\begin{align*}
X_{n+1} &= X_n - \gamma_n g(X_{n+1/2}) \\
X_{n+1/2} &= X_n - \gamma_n g(X_{n-1/2})
\end{align*}
\] (OG)

\[x^* - \gamma g(x_{\text{lag}})\]

\[x_{\text{lead}} - \gamma g(x_{\text{lead}})\]

- **Deterministic:**
  \[\sigma_n = 0\]

- **Offset:**
  \[B_n = O(\gamma_n)\]

[Popov, 1980; Rakhlin & Sridharan, 2013]
Methods, IV: Optimistic gradient

Optimistic gradient [Popov, 1980; Rakhlin & Sridharan, 2013]

\[ X_{n+1} = X_n - \gamma_n g(X_{n+1/2}) \quad X_{n+1/2} = X_n - \gamma_n g(X_{n-1/2}) \]  

\[ \text{OG} \]

- **Deterministic:**  
  \[ \sigma_n = 0 \]

- **Offset:**  
  \[ B_n = \mathcal{O}(\gamma_n) \]
Oracle feedback

In many applications, perfect gradient information is unavailable / too costly:

- **Machine Learning:**
  \[ f(x) = \sum_{i=1}^{N} f_i(x) \] and only a batch of \( \nabla f_i(x) \) is computable per iteration

- **Reinforcement Learning / Control:**
  \[ f(x) = \mathbb{E}[F(x; \theta)] \] and only \( \nabla F(x; \theta) \) can be observed for a random \( \theta \)

- **Game Theory / Bandits:**
  Only \( f(x) \) is observable

Stochastic first-order oracle

A *stochastic first-order oracle (SFO)* is a random field \( G(x; \theta) \) with the following properties

1. **Unbiasedness:**
   \[ \mathbb{E}_{\theta}[G(x; \theta)] = g(x) \]

2. **Finite variance:**
   \[ \mathbb{E}_{\theta}[\|G(x; \theta) - g(x)\|^2] \leq \sigma^2 \]
Methods, V: Robbins–Monro

Robbins–Monro (stochastic gradient descent) \[ X_{n+1} = X_n - \gamma_n G(X_n; \theta_n) \] (RM)

⚠️ Stochastic:
\[ \sigma_n = O(1) \]

✔️ No offset:
\[ B_n = 0 \]
**Methods, VI: Kiefer–Wolfowitz**

The Kiefer-Wolfowitz algorithm

\[
X_{n+1} = X_n \pm \gamma_n \frac{f(X_n + \delta_n \theta_n) - f(X_n - \delta_n \theta_n)}{2\delta_n} \theta_n \tag{KW}
\]

where \( \theta_n \sim \text{unif}\{e_1, \ldots, e_d\} \) is a *random direction* and \( \delta_n \) is the *width* of the finite difference quotient.

⚠️ **Stochastic:**
\[ \sigma_n = \mathcal{O}(1) \]

⚠️ **Offset:**
\[ B_n = \mathcal{O}(\delta_n) \]
From algorithms to flows

Characteristic property of SA schemes

\[
\frac{X_{n+1} - X_n}{\gamma_n} = -g(X_n) + Z_n \approx -g(X_n) \quad \text{“on average”}
\]

Mean dynamics

\[
\dot{x}(t) = -g(x(t)) \quad \text{(MD)}
\]
Characteristic property of SA schemes

\[
\frac{X_{n+1} - X_n}{\gamma_n} = -g(X_n) + Z_n \approx -g(X_n) \quad \text{“on average”}
\]

Mean dynamics

\[
\dot{x}(t) = -g(x(t)) \quad (\text{MD})
\]

**Basic idea:** If \( \gamma_n \) is “small”, the errors wash out and “\( \lim_{t\to\infty} (SA) = \lim_{t\to\infty} (MD) \)”
Outline

1. Background & motivation

2. Preliminaries

3. Applications to minimization problems

4. Applications to min-max problems
Convergence of gradient flows

**Gradient flow**

\[ \dot{x}(t) = -\nabla f(x(t)) \]  

(GF)

**Main property:** \( f \) is a (strict) *Lyapunov function* for (GF)

\[ \frac{df}{dt} = -\|\nabla f(x(t))\|^2 \leq 0 \]  

w/ equality iff \( \nabla f(x) = 0 \)
Convergence of trajectories

Controlling the algorithms' behavior

(A) \( g \) is subcoercive:
\[
\langle g(x), x \rangle \geq 0 \quad \text{for sufficiently large } x
\]

(B) The parameters of (SA) satisfy:
- \( \sum_n \gamma_n = \infty \)
- \( \sum_n \gamma_n B_n < \infty \)
- \( \sum_n \gamma_n^2 \sigma_n^2 < \infty \)
- \( \sum_n \gamma_n^2 \sigma_n^2 < \infty \)

Theorem (Bertsekas & Tsitsiklis, 2000; M, Hallak, Kavis & Cevher, 2020)

**Assume:** (A) + (B)

**Then:** \( X_n \) converges (a.s.) to a component of \( \text{crit}(f) \) where \( f \) is constant.
Are all critical points desirable?

**Figure:** A hyperbolic ridge manifold, typical of ResNet loss landscapes [Li et al., 2018]
**Are traps avoided?**

**Hyperbolic saddle** (isolated non-minimizing critical point)

\[ \lambda_{\text{min}}(\text{Hess}(f(x^*)) < 0, \quad \text{det}(\text{Hess}(f(x^*))) \neq 0 \]

\[ \Rightarrow \text{the flow is linearly unstable near } x^* \]

\[ \Rightarrow \text{convergence to } x^* \text{ unlikely} \]
**Are traps avoided?**

**Hyperbolic saddle** (isolated non-minimizing critical point)

\[ \lambda_{\text{min}}(\text{Hess}(f(x^*))) < 0, \quad \det(\text{Hess}(f(x^*))) \neq 0 \]

\[ \Rightarrow \text{the flow is linearly unstable near } x^* \]

\[ \Rightarrow \text{convergence to } x^* \text{ unlikely} \]

---

**Theorem (Pemantle, 1990)**

**Assume:**

- \( x^* \) is a hyperbolic saddle point
- \( b_n = 0 \)
- \( U_n \) is uniformly bounded (a.s.) and uniformly exciting

\[ \mathbb{E}[\langle U, z \rangle_{+}] \geq c \quad \text{for all unit vectors } z \in \mathbb{S}^{d-1}, x \in \mathcal{X} \]

- \( \gamma_n \propto 1/n \)

**Then:** \( \mathbb{P}(\lim_{n \to \infty} X_n = x^*) = 0 \)
Escape from non-hyperbolic traps

Strict saddles

\[ \lambda_{\min}(\text{Hess}(f(x^*))) < 0 \]
**Escape from non-hyperbolic traps**

**Strict saddles**

\[ \lambda_{\min}(\text{Hess}(f(x^*))) < 0 \]

---

**Theorem (Ge et al., 2015)**

**Given:** tolerance level \( \zeta > 0 \)

**Assume:**

- \( f \) is bounded and satisfies (LS)
- \( \text{Hess}(f(x)) \) is Lipschitz continuous
- for all \( x \in \mathcal{X} \): (a) \( \|\nabla f(x)\| \geq \varepsilon \); or (b) \( \lambda_{\min}(\text{Hess}(f(x))) \leq -\beta \); or (c) \( x \) is \( \delta \)-close to a local minimum \( x^* \) of \( f \) around which \( f \) is \( \alpha \)-strongly convex
- \( b_n = 0 \)
- \( U_n \) is uniformly bounded (a.s.) and contains a component uniformly sampled from the unit sphere
- \( \gamma_n \equiv \gamma \) with \( \gamma = \mathcal{O}(1/\log(1/\zeta)) \)

**Then:** with probability at least \( 1 - \zeta \), SGD produces after \( \mathcal{O}(\gamma^{-2} \log(1/(\gamma \zeta))) \) iterations a point which is \( \mathcal{O}(\sqrt{\gamma} \log(1/(\gamma \zeta))) \)-close to \( x^* \)
Are non-hyperbolic traps avoided almost surely?

**Theorem (M, Hallak, Kavis & Cevher, 2020)**

**Assume:**
- The offset term is bounded as $b_n = \mathcal{O}(\gamma_n)$
- The noise term $U_n$ is bounded (a.s.) and **uniformly exciting**
  $$\mathbb{E}[\langle U, z \rangle^+] \geq c \quad \text{for all unit vectors } z \in \mathbb{S}^{d-1}, \ x \in X$$
- $\gamma_n \propto 1/n^p$ for some $p \in (0, 1]$

**Then:**
$$\mathbb{P}(X_n \text{ converges to a set of strict saddle points}) = 0$$
Outline

1. Background & motivation
2. Preliminaries
3. Applications to minimization problems
4. Applications to min-max problems
Minimization vs. min-max optimization

In minimization problems:

- RM methods converge to the problem’s critical set
- RM methods avoid spurious, non-minimizing critical manifolds
Minimization vs. min-max optimization

In minimization problems:

- RM methods converge to the problem’s critical set
- RM methods avoid spurious, non-minimizing critical manifolds

Do these properties carry over to min-max optimization problems?
Minimization vs. min-max optimization

In minimization problems:

✓ RM methods converge to the problem’s critical set
✓ RM methods avoid spurious, non-minimizing critical manifolds

Do these properties carry over to min-max optimization problems?

Do min-max algorithms

☐ Converge to unilaterally stable/stationary points?
☐ Avoid spurious, non-equilibrium sets?
Min-max dynamics

Mean dynamics

\[ \dot{x}(t) = -g(x(t)) \]  

(MD)

✓ Minimization problems:  (MD) is a gradient flow

✗ Min-max problems:  (MD) can be arbitrarily complicated

\[ \# g = \nabla f \]

\[ \# \text{non-potential } g \]
Min-max dynamics

Mean dynamics

\[ \dot{x}(t) = -g(x(t)) \quad (MD) \]

✓ Minimization problems: (MD) is a gradient flow

✗ Min-max problems: (MD) can be arbitrarily complicated

Theorem (Hsieh et al., 2021)

Assume:

- The offset term is bounded as \( b_n = O(\gamma_n) \)
- The noise term \( U_n \) is bounded (a.s.) and uniformly exciting
  \[ \mathbb{E}[\langle U, z \rangle^+] \geq c \quad \text{for all unit vectors } z \in S^{d-1}, x \in \mathcal{X} \]
- \( \gamma_n \propto 1/n^p \) for some \( p \in (0, 1] \)

Then: \( \mathbb{P}(X_n \text{ converges to an unstable point / periodic orbit}) = 0 \)
Minimization vs. min-max optimization

Qualitatively similar landscape (??)

- Components of critical points ↔ chain transitive sets
- Avoidance of strict saddles ↔ avoidance of unstable periodic orbits

Is there a fundamental difference between min and min-max problems?
Minimization vs. min-max optimization

Qualitatively similar landscape (??)

- Components of critical points ↔ chain transitive sets  
  
- Avoidance of strict saddles ↔ avoidance of unstable periodic orbits

Is there a fundamental difference between min and min-max problems?

Non-gradient problems may have spurious invariant sets!

# Spurious \(\implies\) contains no critical points
**Toy example: bilinear problems**

**Bilinear min-max problems**

\[
\min_{x_1 \in \mathcal{X}_1} \max_{x_2 \in \mathcal{X}_2} f(x_1, x_2) = (x_1 - b_1)^\top A(x_2 - b_2)
\]

**Mean dynamics:**

\[
\begin{align*}
\dot{x}_1 &= -A(x_2 - b_2) \\
\dot{x}_2 &= A^\top (x_1 - b_1)
\end{align*}
\]
**Toy example: bilinear problems**

**Bilinear min-max problems**

\[
\min_{x_1 \in \mathcal{X}_1} \max_{x_2 \in \mathcal{X}_2} f(x_1, x_2) = (x_1 - b_1)^\top A(x_2 - b_2)
\]

**Mean dynamics:**

\[
\dot{x}_1 = -A(x_2 - b_2) \\
\dot{x}_2 = A^\top (x_1 - b_1)
\]

**Energy function:**

\[
E(x) = \frac{1}{2}\|x_1 - b_1\|^2 + \frac{1}{2}\|x_2 - b_2\|^2
\]

**Lyapunov property:**

\[
\frac{dE}{dt} \leq 0 \quad \text{w/ equality if } A = A^\top
\]

\[\implies \text{distance to solutions (weakly) decreasing along (MD)}\]
Periodic orbits

Roadblock: the energy may be a constant of motion

Figure: Hamiltonian flow of $f(x_1, x_2) = x_1 x_2$
**Poincaré recurrence**

**Definition (Poincaré, 1890’s)**

A system is **Poincaré recurrent** if almost every orbit returns *infinitely close* to its starting point *infinitely often*.
**Poincaré recurrence**

**Definition (Poincaré, 1890’s)**

A system is **Poincaré recurrent** if almost every orbit returns *infinitely close* to its starting point *infinitely often*.

---

**Theorem (M, Papadimitriou, Piliouras, 2018; unconstrained version)**

*(MD) is Poincaré recurrent in all bilinear min-max problems that admit an equilibrium*
Figure: Behavior of gradient and extra-gradient methods with stochastic feedback

First-order training methods converge to a (random) periodic orbit

# But see also Chavdarova et al., 2019; Hsieh et al., 2020
The Kupka-Smale theorem

Systems with the structure of bilinear games are rare:

**Theorem (Kupka, 1963)**

Let $\mathcal{V} = C^2(\mathbb{R}^d; \mathbb{R}^d)$ be the space of $C^2$ vector fields on $\mathbb{R}^d$ endowed with the Whitney topology. Then the set of vector fields with a non-trivial recurrent set is *meager* (in the Baire category sense).

**Theorem (Smale, 1963)**

For any vector field $\mathbf{g} \in \mathcal{V}$, the following properties are generic (in the Baire category sense):

- All closed orbits are *hyperbolic*
- Heteroclinic orbits are *transversal* (i.e., stable and unstable manifolds intersect transversally)

**TLDR:** non-attracting periodic orbits are *non-generic* (they occur negligibly often)
Convergence to attractors

**Attractors** $\sim$ natural solution concepts for non-min problems

**Theorem (Hsieh et al., 2021)**

**Assume:** $S$ is an attractor of (MD) + step-size conditions (B)

**Then:** For every tolerance level $\alpha > 0$, there exists a neighborhood $U$ of $S$ such that

$$\mathbb{P}(X_n \text{ converges to } S \mid X_1 \in U) \geq 1 - \alpha$$
Almost bilinear games

Consider the “almost bilinear” game

\[
\min_{x_1 \in \mathcal{X}_1} \max_{x_2 \in \mathcal{X}_2} f(x_1, x_2) = x_1 x_2 + \varepsilon \phi(x_2)
\]

where \( \varepsilon > 0 \) and \( \phi(x) = (1/2)x^2 - (1/4)x^4 \)

Properties:

- Unique critical point at the origin
- Unstable under (MD)
- All RM algorithms attracted to spurious limit cycle from almost all initial conditions

\( \triangleright \) Hsieh et al., 2021
Spurious attractors in almost bilinear games

RM algorithms converge to a spurious limit cycle with no critical points

Figure: Convergence to a spurious attractor. Left: stochastic gradient descent; right: stochastic extra-gradient
Forsaken solutions

Another almost bilinear game

\[
\min_{x_1 \in X_1} \max_{x_2 \in X_2} f(x_1, x_2) = x_1 x_2 + \varepsilon [\phi(x_1) - \phi(x_2)]
\]

where \( \varepsilon > 0 \) and \( \phi(x) = (1/4)x^2 - (1/2)x^4 + (1/6)x^6 \)

Properties:

- Unique critical point near the origin
- Stable under (MD), but **not a local min-max**
- **Two isolated periodic orbits:**
  - One **unstable**, shielding critical point, but small
  - One **stable**, attracts all trajectories of (MD) outside small basin

\[\text{Hsieh et al., 2021}\]
ForSaken solutions in almost bilinear games

With high probability, all Robbins–Monro (RM) algorithms forsake the game’s unique (local) equilibrium

Figure: Convergence to a spurious attractor. Left: stochastic gradient descent; right: stochastic extra-gradient
Conclusions

Minimization and min-max optimization problems are fundamentally different:

- Min-max methods may have limit points that are neither stable nor stationary
- Bilinear games are not representative case studies for min-max optimization
- Cannot avoid spurious, non-equilibrium sets with positive probability
- Different approach needed (mixed-strategy learning, multiple-timescales, adaptive methods...)

Many open questions:

- What about second-order methods?
- Applications to finite games (where bilinear games are no longer fragile)?
- Which equilibria are stable under first-order methods for learning in games?
- ...
References


References II


