Stable Optimization in Deep Learning: Geometry and Games

Thomas Pethick

January 2026

EPFL

Introduction

Instabilities arises everywhere in deep learning:

- **Scaling up**: model size (depth, width), long runs
- **Multiple epochs**: overfitting
- Long context
- Recurrences: diffusion models, looped transformers, SSMs, TTT
- **Multiplayer games**: GANs, agents interacting

Aspiration: How do we design methods that work in the limit $N \to \infty$?

where N is width, depth, #epochs, context length etc...

Introduction

Optimization perspective: Unstable when our modeling is not faithful

Three parts:

- **Non-Euclidean methods**: Co-designing optimizer and architecture.
- **Long-run stability**: Avoiding overfitting.
- **Games**: Stabilizing multiplayer interactions.

Non-Euclidean methods

What is non-Euclidean methods?

Algorithmic Template

$$x^{k+1} \in \arg\min_x \gamma\langle d^k, x\rangle + h_k(x)$$

with gradient/dual feedback d^k.

- **Special case**: Gradient descent

$$h_k(x) = \tfrac{1}{2}\|x - x^k\|_2^2 \;\Rightarrow\; x^{k+1} = x^k - \gamma d^k$$

- **Processing of gradients**: can change the direction and magnitude

Based on:

Pethick et al., "Training Deep Learning Models with Norm-Constrained LMOs", 2025
Pethick et al., "Training Neural Networks at Any Scale", 2025

Why do we care about non-Euclidean methods?

They unify a wide range of widely used methods!

$$x^{k+1} \in \arg\min_{x \in \mathcal{X}} \gamma_k \langle d^k, x \rangle + h_k(x) \quad \text{with} \quad h_k(x) = \tfrac{1}{2}\|x - x^k\|^2_{2,H_k}.$$

Algorithm	**Dual feedback** d^k	**Preconditioner** H_k
AdaGrad / RMSProp	g^k	$[H_k]_{ii} = [H_{k-1}]_{ii} + (g_i^k)^2; \forall i \in [p]$
Adam / AdamW	$\beta_1 d^{k-1} + (1-\beta_1) g^k$	$[H_k]_{ii} = \beta_2 [H_{k-1}]_{ii} + (1-\beta_2)(g_i^k)^2; \forall i \in [p]$
ℓ_∞-descent	g^k	$[H_k]_{ii} = (g_i^k)^2; \forall i \in [p]$
SignSGD / Signum	$g^k / \beta_1 d^{k-1} + (1-\beta_1) g^k$	$[H_k]_{ii} = (g_i^k)^2; \forall i \in [p]$
Spectral descent	$G_k = \mathrm{mat}(g^k)$	$R_k^{1/2} \otimes L_k^{1/2} : \begin{cases} R_k = G_k G_k^\top \\ L_k = G_k^\top G_k \end{cases}$
Shampoo	$G_k = \mathrm{mat}(g^k)$	$R_k^{1/2} \otimes L_k^{1/2} : \begin{cases} R_k = \beta_2 R_{k-1} + (1-\beta_2) G_k G_k^\top \\ L_k = \beta_2 L_{k-1} + (1-\beta_2) G_k^\top G_k \end{cases}$
Scion / Muon (spectral)	$\beta_1 d^{k-1} + (1-\beta_1) G_k$	$R_k^{1/2} \otimes L_k^{1/2} : \begin{cases} R_k = G_k G_k^\top \\ L_k = G_k^\top G_k \end{cases}$

$\beta_2 \to 0$:

- RMSProp $\Rightarrow$ ℓ_∞-descent/SignSGD
- Adam $\Rightarrow$ Signum
- Shampoo $\Rightarrow$ Spectral descent

Table from "Training Neural Networks at Any Scale" [Pet+25b]

Why do we care about non-Euclidean methods?

Assumption Lipschitz: $\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\| \quad \forall x, y \in \mathbb{R}^d$.

Theorem Steepest descent, $h_k(x) = \|x - x^k\|^2$, with $\gamma = 1/L$ satisfies

$$\min_{1 \leq k \leq n} \|\nabla f(x^k)\|_* \leq \sqrt{\frac{2L(f(x^1) - f^*)}{n}}$$

Favorable geometry when $L < L_2$ and $\|\nabla f\|_* / \|\nabla f\|_2$ large:

- Max-norm ℓ_∞: $\|\nabla f\|_1 / \|\nabla f\|_2$ large (effective sparsity)
- Schatten-∞/Spectral: $\|\nabla f\|_{\mathcal{S}_1} / \|\nabla f\|_F$ large (effective rank)

Examples of favorable geometry:

- Max-norm ℓ_∞: under log-sum-exp loss [Car+15]
- Schatten-∞/Spectral: grads in DL have high effective rank [DD25] [1]

$\Rightarrow$ Improved dimensionality dependency aka scaling law coefficient

[1] Over single sample: rank-1, so Spectral descent $\equiv$ Frobenius descent.
Explains recent successful SGD pre-training for LLMs [SGO25; Mar+25]

Today

Modular norm [Lar+24] NNs are smooth *if the params are constrained*.

So let us use a classical algorithm for constrained problems!

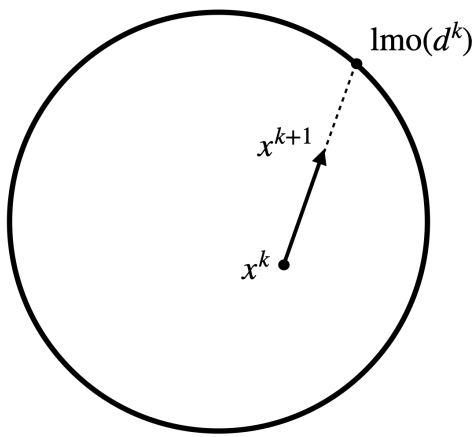
Stochastic Conditional Gradient (SCG) [MHK20]

$$d^k = (1-\alpha_k)d^{k-1} + \alpha_k \nabla f(x^k, \xi_k)$$
$$x^{k+1} = (1-\gamma_k)x^k + \gamma_k \mathrm{lmo}(d^k)$$

where lmo is the *linear minimization oracle*:

$$\mathrm{lmo}(d) \in \arg\min_x \langle d, x\rangle + h(x)$$

with $h(x) = 0$ if $\|x\| \leq \rho$ else ∞.



SOTA training with three simple components:

1. Non-Euclidean geometry
2. Momentum estimator
3. Weight decay

Properties of stochastic conditional gradient (SCG)

Theorem [Pet+25a] For L-smooth SCG with $\gamma = 1/n^{3/4}$ and $\alpha_k = 1/\sqrt{k}$:

$$\text{first order stationarity} \leq \mathcal{O}\left(\frac{1}{n^{1/4}} + \frac{L\rho}{n^{3/4}}\right)$$

❶ **Linear minimization oracle (LMO)**:

- *Adapt to geometry*: through (possibly non-Euclidean) norm $\|\cdot\|$
- *Scale invariant*: Lipschitz agnostic, converges for (L_0, L_1)-smooth

❷ **Stochastic**: Momentum is crucial!

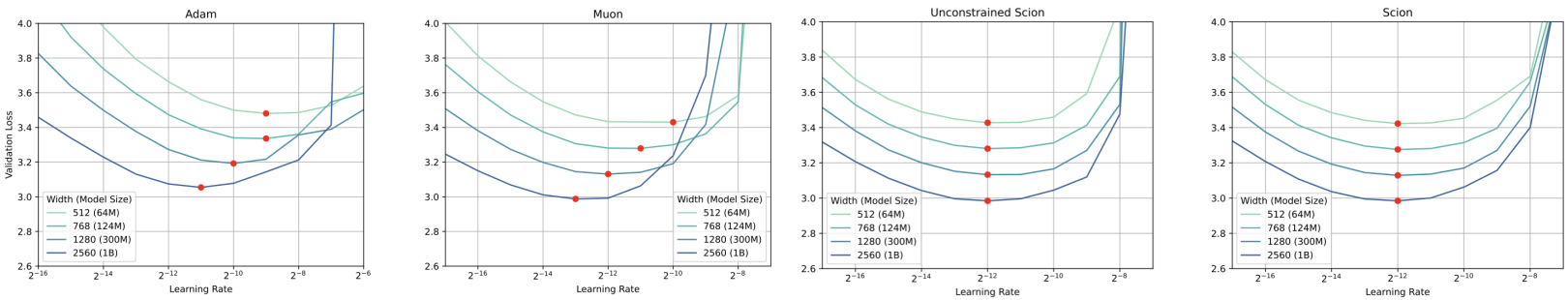
- *Naively*: $d^k = \nabla f(x^k, \xi_k)$, d^k unbiased $\not\Rightarrow$ $\mathrm{lmo}(d^k)$ unbiased
- *From the rate*: Large $\frac{\text{batchsize}}{\text{horizon}}$ needed

❸ **Regularization**: SCG solves a *constrained* problem

- *Naively*: $x^{k+1} = x^k + \gamma\,\mathrm{lmo}(d^k) \Rightarrow \|x^k\| \leq \rho \sum_{i=1}^k \gamma_i$
- *Norm control of SCG*: $\|x^k\| \leq \rho$
- Weight decay decouples optimization and regularization

Empirical observations

❶ **LMO** Stable across scales:

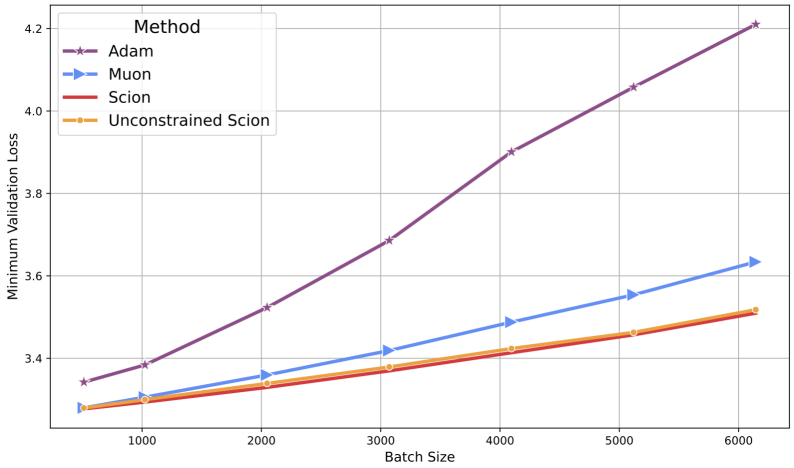


Weight norm (bias norm)		W_1 (1-hot encoded)	W_1 (image domain)	$(W_\ell)_{\ell\in[2,\dots,L-1]}$	W_L	b_ℓ
$\text{RMS} \to \text{RMS}$ (RMS)	lmo	$-\sqrt{d_{\text{out}}}UV^\top$	$-\sqrt{d_{\text{out}}/d_{\text{in}}}UV^\top$			$-\frac{b_\ell}{\|b_\ell\|_{\text{RMS}}}$
$1 \to \text{RMS}$ (RMS)	lmo	$\text{col}_j(W_1) \mapsto -\sqrt{d_{\text{out}}}\frac{\text{col}_j(W_1)}{\|\text{col}_j(W_1)\|_2}$	$\text{col}_j(W_\ell) \mapsto -\frac{\sqrt{d_{\text{out}}}}{d_{\text{in}}}\frac{\text{col}_j(W_\ell)}{\|\text{col}_j(W_\ell)\|_2}$			$-\frac{b_\ell}{\|b_\ell\|_{\text{RMS}}}$
$\text{RMS} \to \infty$ (RMS)	lmo	$\text{row}_i(W_1) \mapsto -\frac{\text{row}_i(W_1)}{\|\text{row}_i(W_1)\|_2}$	$\text{row}_i(W_\ell) \mapsto -\frac{\text{row}_i(W_\ell)}{\sqrt{d_{\text{in}}}\|\text{row}_i(W_\ell)\|_2}$			$-\frac{b_\ell}{\|b_\ell\|_{\text{RMS}}}$
$1 \to \infty$ (∞)	lmo	$-\text{sign}(W_1)$	$-\frac{1}{d_{\text{in}}}\text{sign}(W_\ell)$			$-\text{sign}(b_\ell)$

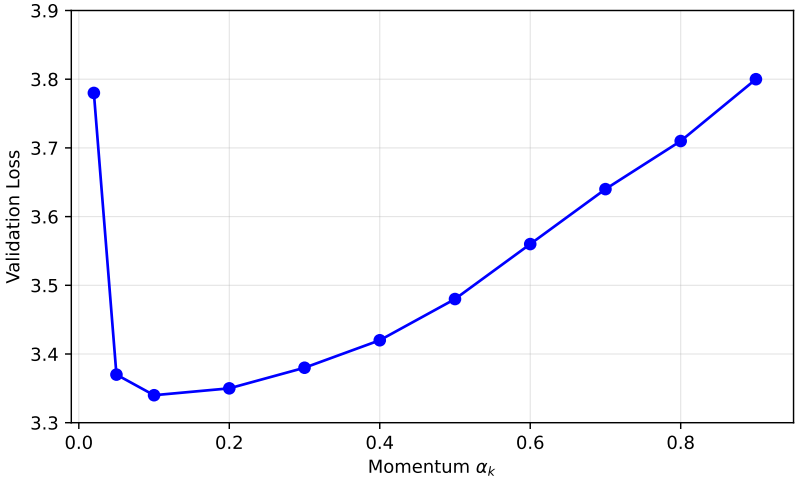
Empirical observations

❷ Stochastic

Improves when $\frac{\text{batchsize}}{\text{horizon}}$ is large



Momentum is crucial



Empirical observations

❸ Regularization

- Constraints especially important for multi-epoch

  Remove ad-hoc Frobenious normalization in Muon CIFAR-10 speedrun

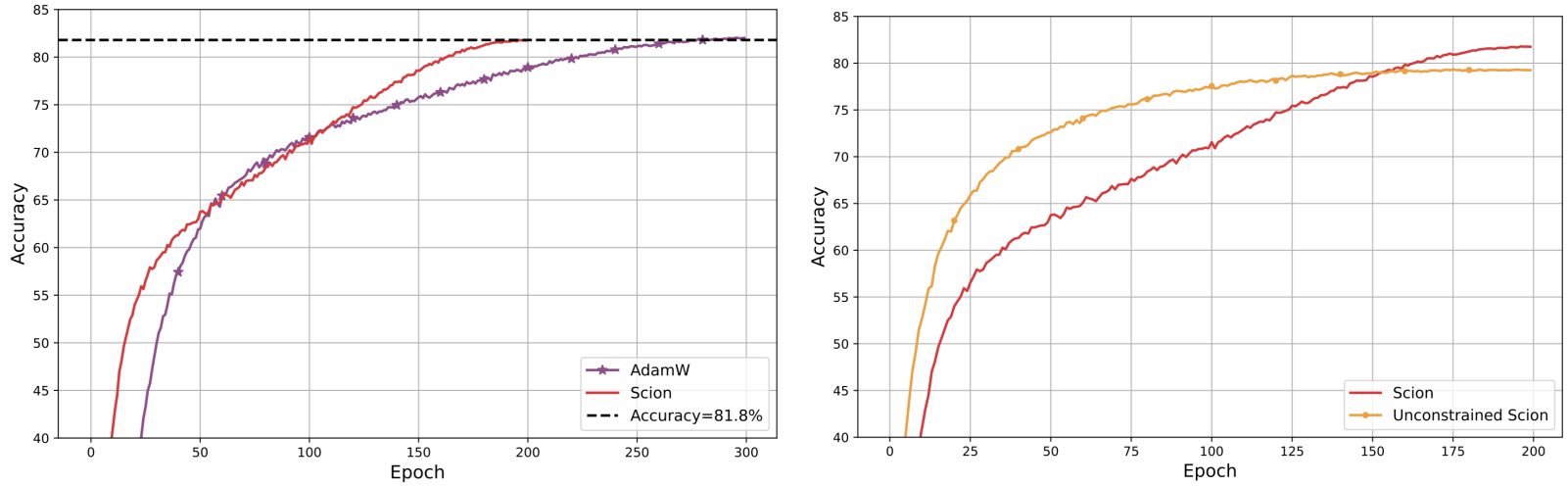


Figure 1: DeiT ImageNet

- Concurrently used to train a frontier model by Kimi AI [Liu+25]

Part II: Long-Run Stability

Multi-epochs: Avoiding overfitting

- **Early stopping** inherently does not allow us to train indefinitely
- **Alternative regularization**: Sharpness aware minimization [For+20]

$$\min_x \max_{\|\varepsilon\|_2 \le \rho} f(x+\varepsilon) \quad \Rightarrow \quad \begin{aligned} \widetilde{x}^k &= x^k + \rho \frac{\nabla f(x^k)}{\|\nabla f(x^k)\|_2} \\ x^{k+1} &= x^k - \gamma_k \nabla f(\widetilde{x}^k) \end{aligned} \qquad \text{(SAM)}$$

- We can break the sequentiality

$$\begin{aligned} \widetilde{x}^k &= x^k + \rho \frac{\nabla f(y^k)}{\|\nabla f(y^k)\|_2} \\ y^{k+1} &= x^k - \gamma_k \nabla f(y^k) \\ x^{k+1} &= x^k - \gamma_k \nabla f(\widetilde{x}^k) \end{aligned} \qquad \text{(SAMPa)}$$

- We can combine with optimistic gradient descent for free:

$$x^{k+1} = (1-\lambda)\,\mathrm{SAMPa}(x^k) + \lambda\,\mathrm{OptGD}(x^k) \qquad \text{(SAMPa-}\lambda\text{)}$$

Based on:
Xie, Pethick, and Cevher, "SAMPa: Sharpness-aware Minimization Parallelized", 2024

Multi-epochs: Results

Potential function: $\mathcal{V}^k := f(x^k) + \frac{1}{2}(1 - \gamma_k L)\|\nabla f(x^k) - \nabla f(y^k)\|_2^2$.

Theorem For convex and L-smooth f, SAMPa converges with $\rho > 0$.

Extremely predictive of practice:

Model Temporal cost/Epochs	**SGD** $\times 1/400$	**SAM** $\times 2/200$	**SAMPa-0** $\times 1/200$	**SAMPa-0.2** $\times 1/200$	**SAMPa-0.2** $\times 1/400$
DenseNet-121	$96.14_{\pm 0.09}$	$96.49_{\pm 0.14}$	$96.53_{\pm 0.11}$	$\mathbf{96.77}_{\pm 0.11}$	$96.92_{\pm 0.09}$
Resnet-56	$94.20_{\pm 0.39}$	$94.26_{\pm 0.70}$	$94.31_{\pm 0.43}$	$\mathbf{94.62}_{\pm 0.35}$	$95.43_{\pm 0.25}$
VGG19-BN	$94.76_{\pm 0.10}$	$95.05_{\pm 0.17}$	$95.06_{\pm 0.22}$	$\mathbf{95.11}_{\pm 0.10}$	$95.34_{\pm 0.07}$
WRN-28-2	$95.71_{\pm 0.19}$	$95.98_{\pm 0.10}$	$96.06_{\pm 0.10}$	$\mathbf{96.13}_{\pm 0.14}$	$96.31_{\pm 0.09}$
WRN-28-10	$96.77_{\pm 0.21}$	$97.25_{\pm 0.09}$	$97.24_{\pm 0.11}$	$\mathbf{97.34}_{\pm 0.09}$	$97.46_{\pm 0.07}$
Average	$95.52_{\pm 0.10}$	$95.81_{\pm 0.15}$	$95.86_{\pm 0.10}$	$\mathbf{95.99}_{\pm 0.08}$	$96.29_{\pm 0.06}$

- SAMPa-0 closely matches SAM
- SAMPa-0.2 leads to consistent improvement
- SAMPa-0.2 *effectively has a larger critical batch size*

Part III: Games

Motivation

Multiple agents with different objectives:

$$z_i^\star \in \arg\min_{z_i} \phi_i(z_i; z_{-i}^\star) \quad \forall i = 1, ..., N$$

General vector field

$$\dot{z} = F(z) := \begin{pmatrix} \nabla_{z_1} \phi_1(z_1; z_{-1}) \\ \vdots \\ \nabla_{z_N} \phi_N(z_N; z_{-N}) \end{pmatrix}$$

- F is not necessarily a potential

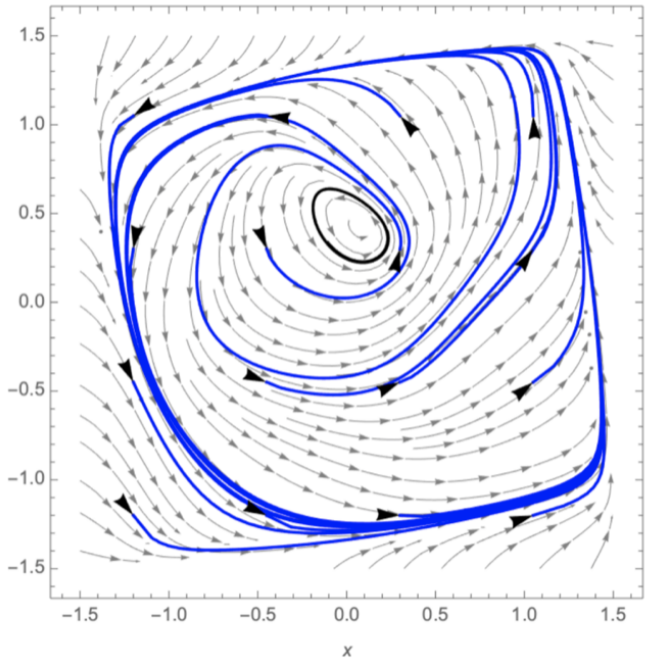


Figure 2: Even the celebrated extragradient method designed for multiplayer settings can exhibit divergence or cycling dynamics.

Motivation

Under how weak conditions can convergence be established for games?

Several works:

- Relaxed extragradient methods [Pet+22]
- Stochastic case via bias-correction [Pet+23]
- Linear interpolation and applications to GANs [PXC23]
- **Hyperplane projection** [PMC25]
  ↳ *weakest conditions while remaining efficient*

Note: In this part we focus on the Euclidean case $\|\cdot\| = \|\cdot\|_2$.

By the end: How to escape limit cycles and stabilize GAN training

Why is (convex) minimization easy?

Definition 1 (Minimization)

Given a function $f : \mathbb{R}^d \to \mathbb{R}$, find

$$x^\star \in \arg\min_{x \in \mathbb{R}^d} f(x).$$

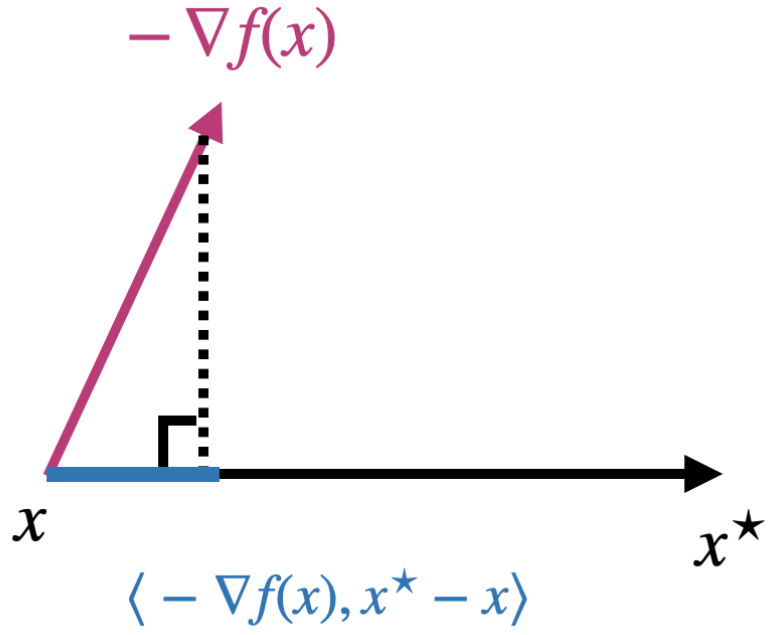
Operator view $F = \nabla f$

(Star)-convexity for all $x \in \mathbb{R}^d$

$$\langle \nabla f(x), x - x^\star \rangle \geq f(x) - f(x^\star)$$

(Star)-convexity + L-Lipschitz

$$\langle \nabla f(x), x - x^\star \rangle \geq \tfrac{1}{L} \|\nabla f(x)\|^2$$



The gradient direction always points towards the solution

So the gradient method suffice:

$$x^{k+1} = x^k - \gamma_k \nabla f(x^k)$$

for some stepsize $\gamma_k > 0$.

Why are games hard?

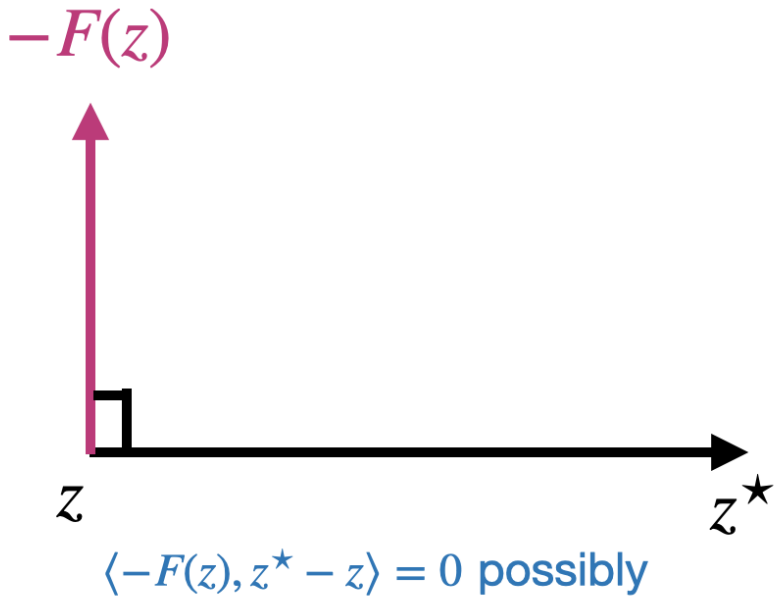
Definition 2 (Minimax)

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ \underset{y\in\mathbb{R}^m}{\text{maximize}}\ f(x,y)$$

Operator view with $z=(x,y)$:
$F(z)=(\nabla_x f(x,y), -\nabla_y f(x,y))$

First order stationarity
Find $z\in\mathbb{R}^d$ such that $F(z)=0$

Monotonicity
$\langle F(z), z-z^\star\rangle \geq 0, \quad$ for all $z\in\mathbb{R}^d$
Equivalent to convex-concavity.



The gradient direction might not point towards the solution

In fact, *never* the case for **bilinear**:

$$f(x,y)=\langle x, Ay\rangle$$

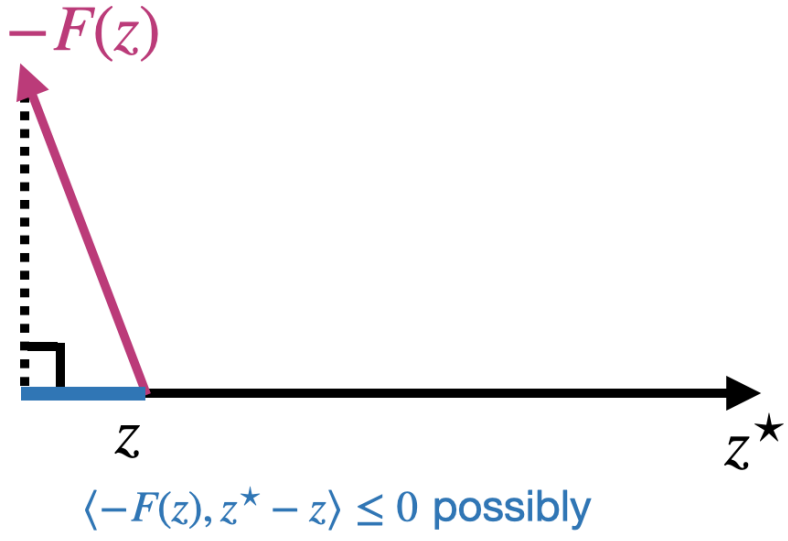
Very common, e.g., in Lagrangian (re)formulations and game theory

Beyond monotonicity: The weak MVI

Weak Minty variational inequality
For all $z \in \mathbb{R}^d$

$$\langle F(z), z - z^\star \rangle \geq \rho \|F(z)\|^2$$

where $\rho \in \mathbb{R}$ can be negative.



The gradient direction can point away from the solution

Root finding

Task 1 (Root finding)

Find $z \in \mathbb{R}^d$ *such that*

$$0 \in T(z) := F(z) + A(z).$$

Operator splitting

- the operator $F : \mathbb{R}^d \to \mathbb{R}^d$ is single-valued
- the operator $A : \mathbb{R}^d \rightrightarrows \mathbb{R}^d$ is a *set-valued operator*

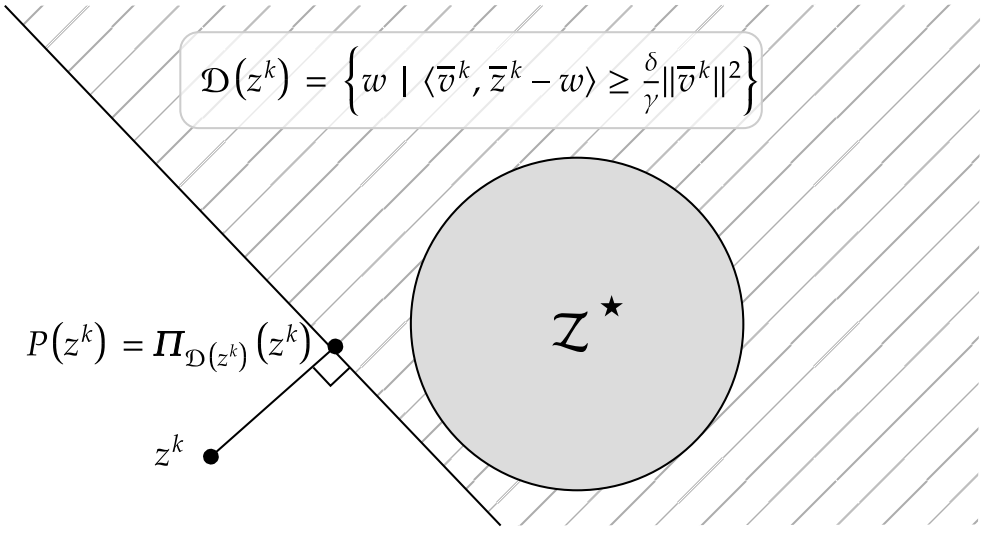
Example (Composite Minimization)

$$\min_{x \in \mathbb{R}^d} f(x) + g(x)$$

Root finding captures the first order stationary conditions:

- *Gradient* $F = \nabla f$ of a smooth function $f : \mathbb{R}^d \to \mathbb{R}$
- *Subdifferential* $A = \partial g$ of a function $g : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$

Halfspace projection



Definition 3 (Halfspace construction)

Given approximate PPA $\bar{z} = z - (\bar{v} + \varepsilon)$ where $\bar{v} \in \gamma T(\bar{z})$, construct

$$\mathcal{D}(z) = \left\{ w \mid \langle \bar{v}, \bar{z} - w \rangle \geq \frac{\rho}{\gamma} \|\bar{v}\|^2 \right\}$$

- Contains the solution set, i.e., $\mathcal{Z}^\star \subseteq \mathcal{D}(z)$
- The projection $P(z) := \Pi_{\mathcal{D}(z)}(z)$ moves z towards a fixed point
- We just need to choose ε such that $\operatorname{fix} P \subseteq \operatorname{zer} T$!

The method

Using the closed form solution to the hyperplane projection:

A Hybrid Proximal Extragradient method (HPE)

$$\begin{aligned} \text{find} \quad & \bar{z}^k \in \mathbb{R}^d \quad \text{and} \quad \bar{v}^k \in \gamma(A+F)\bar{z}^k \\ \text{s.t.} \quad & \bar{z}^k = z^k - (\bar{v}^k + \varepsilon^k) \quad \text{and} \quad -\langle \varepsilon^k, \bar{v}^k \rangle \leq \sigma \|\bar{v}^k\|^2 \\ \text{update} \quad & z^{k+1} = z^k - \lambda_k \alpha_k \bar{v}^k \quad \alpha_k = \frac{\langle \bar{v}^k, z^k - \bar{z}^k \rangle}{\|\bar{v}^k\|^2} + \frac{\rho}{\gamma} \end{aligned}$$

where $\lambda_k \in (0, 2)$ is an overrelaxation parameter.

Unification of many methods:

- (relaxed) proximal point algorithm: $\varepsilon^k = 0$
- (relaxed) extragradient method: $\varepsilon^k = \gamma(Fz^k - F\bar{z}^k)$, $A \equiv 0$
- Forward-backward-forward: $\varepsilon^k = \gamma(Fz^k - F\bar{z}^k)$, $\lambda_k = 1$, $\rho = 0$
- Optimistic gradient: $\varepsilon^k = \gamma(F\bar{z}^{k-1} - F\bar{z}^k)$

Conclusion

Thm $\mathcal{O}(1/k)$ rate as long as $\rho > -(1-\sigma)\gamma$

Key takeaway Large stepsizes γ are important for convergence in nonmonotone problems.

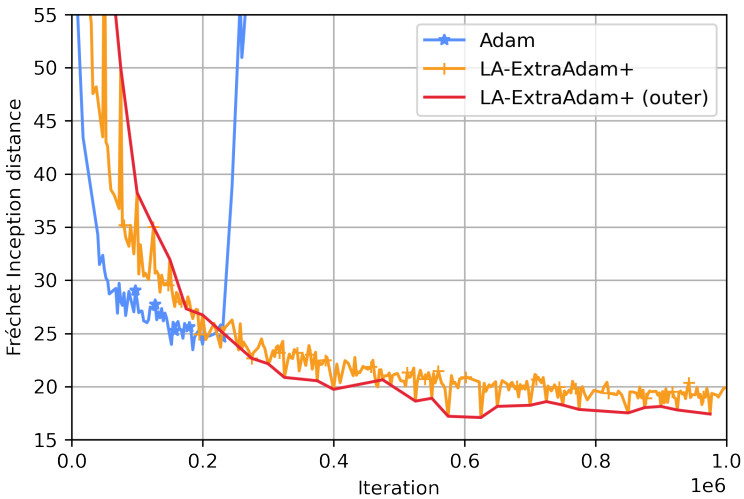


Figure 3: GAN training

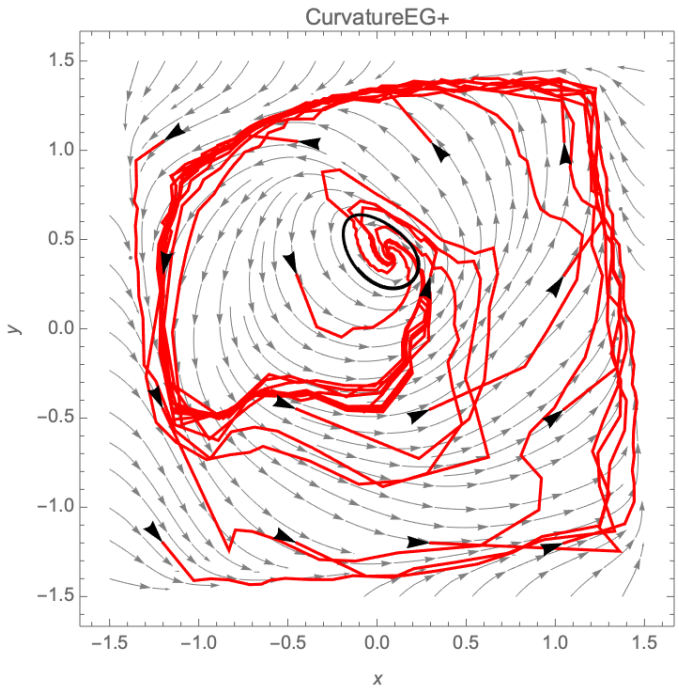


Figure 4: Adaptivity picking γ can solve hard instances

Conclusion

What is next?

Returning to the initial list:

- **Scaling up (Part I)**: batch size, model size (depth, width), long runs
- **Multiple epochs (Part II)**: overfitting
- Long context
- Recurrences: diffusion models, looped transformers, SSMs, TTT
- **Multiplayer games (Part III)**: GANs, agents interacting

How do we build stable systems that requires minimal oversight?

- How to co-design optimizers and architectures for other modules?
- Treat diffusion, looped models, depth etc. as fixed point iterations
- How to be robust to repeated data?
- How to be robust to other agents?

Thank you!

References i

[Car+15] David E Carlson et al. "Preconditioned spectral descent for deep learning". In: *Advances in neural information processing systems* 28 (2015).

[DD25] Damek Davis and Dmitriy Drusvyatskiy. "When do spectral gradient updates help in deep learning?" In: (2025).

[For+20] Pierre Foret et al. "Sharpness-aware minimization for efficiently improving generalization". In: *arXiv preprint arXiv:2010.01412* (2020).

[Lar+24] Tim Large et al. "Scalable Optimization in the Modular Norm". In: *arXiv preprint arXiv:2405.14813* (2024).

[Liu+25] Jingyuan Liu et al. "Muon is Scalable For LLM Training". In: (2025).

References ii

[Mar+25] Martin Marek et al. “Small batch size training for language models: When vanilla sgd works, and why gradient accumulation is wasteful”. In: *arXiv preprint arXiv:2507.07101* (2025).

[MHK20] Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. “Stochastic conditional gradient methods: From convex minimization to submodular maximization”. In: *Journal of machine learning research* 21.105 (2020), pp. 1–49.

[Pet+22] Thomas Pethick et al. “Escaping limit cycles: Global convergence for constrained nonconvex-nonconcave minimax problems”. In: *International Conference on Learning Representations (ICLR)*. 2022.

References iii

[Pet+23] Thomas Pethick et al. “Solving stochastic weak Minty variational inequalities without increasing batch size”. In: *International Conference on Learning Representations (ICLR)*. 2023. URL: https://openreview.net/forum?id=ejR4E1jaH9k.

[Pet+25a] Thomas Pethick et al. “Training Deep Learning Models with Norm-Constrained LMOs”. In: *International Conference on Machine Learning (ICML)*. 2025.

[Pet+25b] Thomas Pethick et al. “Training Neural Networks at Any Scale”. In: *arXiv preprint arXiv:2511.11163* (2025).

References iv

[PMC25] Thomas Pethick, Ioannis Mavrothalassitis, and Volkan Cevher. “Efficient interpolation between extragradient and proximal methods for weak MVIs”. In: *International Conference on Learning Representations (ICLR)*. 2025.

[PXC23] Thomas Pethick, Wanyun Xie, and Volkan Cevher. “Stable nonconvex-nonconcave training via linear interpolation”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2023.

[SGO25] Teodora Srećković, Jonas Geiping, and Antonio Orvieto. “Is your batch size the problem? Revisiting the Adam-SGD gap in language modeling”. In: *arXiv preprint arXiv:2506.12543* (2025).

References v

[XPC24] Wanyun Xie, Thomas Pethick, and Volkan Cevher. "SAMPa: Sharpness-aware Minimization Parallelized". In: *Neural Information Processing Systems (NeurIPS)*. 2024.