AIE6003: Optimization for Machine Learning

Xiaopeng Li
CUHK(SZ)
SAI

July 5, 2026

Outline

Black-box Optimization

Gradient may be unavailable in ML problems:

- nondifferentiable or simulator-based objectives;
- infeasible backpropagation;
- black-box oracle models.

Black-box optimization problem:

$$\min_{x \in \mathbb{R}^d} f(x), \qquad \text{oracle returns only } f(x), \text{ possibly noisy.}$$

Cost metric: number of function queries.

Synonyms:

black-box $\approx$ derivative-free $=$ gradient-free $=$ zeroth-order (ZO).

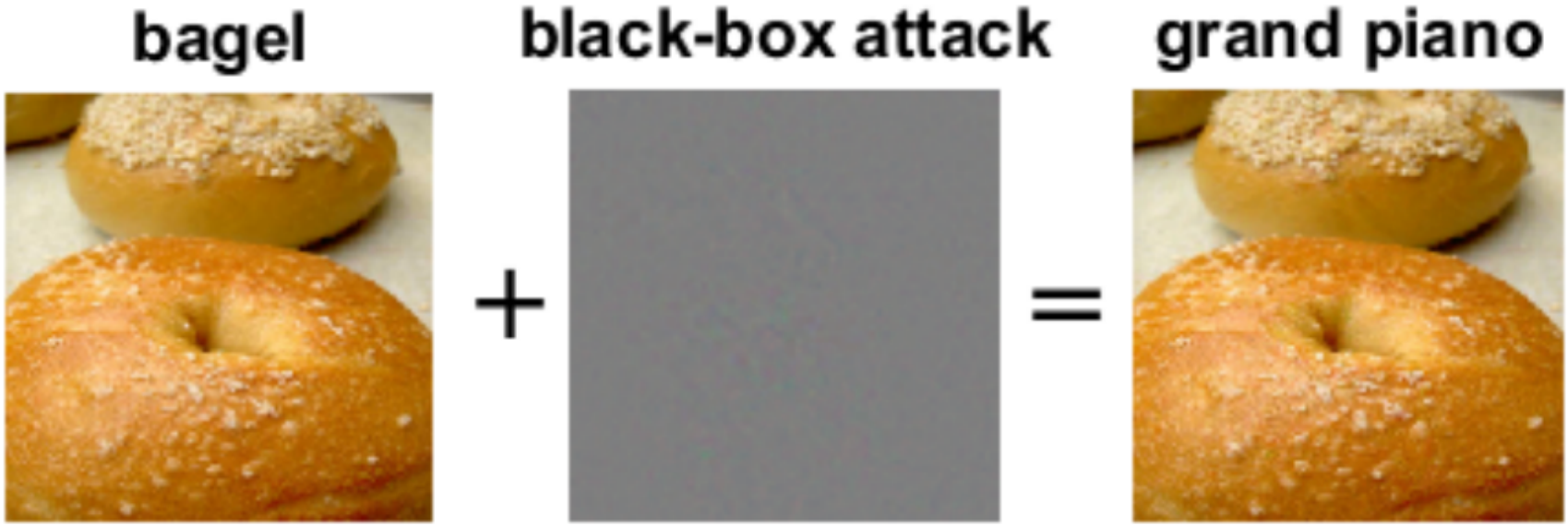
Application: black-box adversarial attacks

Setting: attacker queries deployed image classifier f, sees only softmax output. Cannot backprop through proprietary or cloud-hosted model.

Objective [Chen et al., 2017]:

$$\min_{\delta} \ \|\delta\|^2 + c \cdot \text{loss}\left(f(x+\delta),\, y_{\text{target}}\right).$$

Constraint: only $f(\cdot)$ evaluations available, no gradient access.



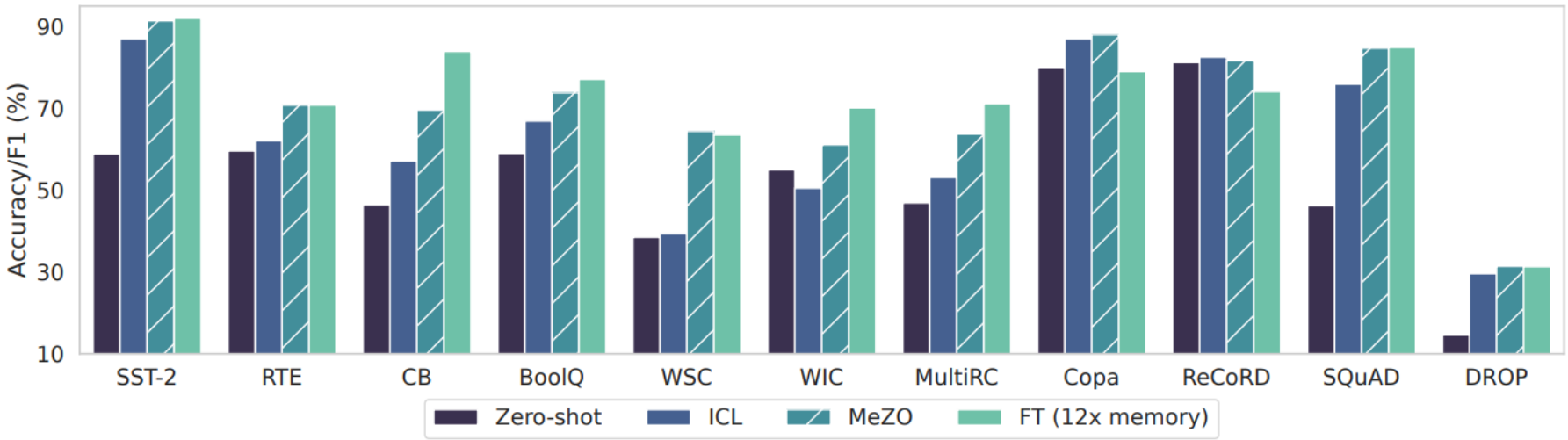
Application: Memory-efficient LLM fine-tuning

Setting: fine-tune pre-trained LLM. On A100 80GB, cannot fine-tune OPT-30B with backprop, but can run forward pass.

Objective [Malladi et al., 2023]:

$$\min_{\theta} \ \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\mathcal{L}(f_\theta(x),\, y)\big].$$

Constraint: only forward-pass evaluations of $\mathcal{L}$ affordable, no backprop.



Application: Reinforcement Learning

Setting: train a policy π_θ in a simulator. Discrete actions + non-differentiable dynamics $\implies$ cannot backprop through trajectories.

Objective [Salimans et al., 2017]:

$$\max_\theta \; J(\theta) := \mathbb{E}_{\tau \sim p_\theta}[R(\tau)].$$

Constraint: only rollout-based reward available, no gradient access.

Approach: Evolution Strategies, sample $u_i \sim \mathcal{N}(0, I)$, evaluate $J(\theta + \mu u_i)$, update along reward-weighted average $\sum_i J(\theta + \mu u_i)\, u_i$.

Competitive performance with policy gradient on MuJoCo or Atari.

Outline

Golden-section search

Setting. f unimodal on $[a, b]$. Golden ratio $\varphi = \frac{\sqrt{5}-1}{2} \approx 0.618$.

Probe placement. Place two interior points symmetrically:

$$c = b - \varphi(b - a), \qquad d = a + \varphi(b - a).$$

Idea. Evaluate $f(c), f(d)$; keep the subinterval with the minimum inside:

- if $f(c) < f(d)$: keep $[a, d]$ (new endpoints a, d);
- else keep $[c, b]$ (new endpoints c, b).

Properties.

- Interval shrinks by factor $\varphi \approx 0.618$ per iteration.
- One function evaluation per iteration (after the first).
- Linear rate, no derivatives required.

Why the golden ratio?

Goal. Reuse one of the old probes after shrinking the interval.

Setup. Let $L = b - a$. Probes at $c = b - \varphi L$ and $d = a + \varphi L$. Suppose $f(c) < f(d)$, so we keep $[a, d]$ with new length $L' = \varphi L$.

New probes on $[a, d]$. The new left probe sits at

$$d - \varphi L' \;=\; (a + \varphi L) - \varphi \cdot \varphi L \;=\; a + \varphi(1 - \varphi)\, L.$$

The old probe c sits at

$$c = b - \varphi L = a + (1 - \varphi)\, L.$$

They coincide iff $\varphi(1 - \varphi) = 1 - \varphi$, i.e. $\varphi^2 = 1 - \varphi$, giving $\varphi = \frac{\sqrt{5}-1}{2}$.

Consequence: the old probe c is the new left probe; only the new right probe needs to be evaluated. One function call per iteration.

Compass search

Algorithm:

- initialize x_0, step size $\mu > 0$.
- for $t = 0, 1, 2, \ldots$:
  - for $i = 1, \ldots, d$: try $x_t \pm \mu e_i$; accept first decrease.
  - if no decrease this round: $\mu \leftarrow \mu/2$.

Guarantee: converges to a stationary point under smoothness.

Curse of dimensionality: up to $2d$ queries per successful step. At ML scale, small CNN $d \approx 10^6$, modern LLM $d \approx 10^{10}$, cannot afford.

Goal: per-iteration query cost independent of d.

$$\Longrightarrow \text{ randomized gradient estimation.}$$

Outline

Finite differences

Forward difference: along coordinate i:

$$\widehat{\partial_i f}(x) = \frac{f(x + \mu e_i) - f(x)}{\mu}, \qquad \text{error } O(\mu),\ d{+}1 \text{ queries.}$$

Central difference: along coordinate i:

$$\widehat{\partial_i f}(x) = \frac{f(x + \mu e_i) - f(x - \mu e_i)}{2\mu}, \qquad \text{error } O(\mu^2),\ 2d \text{ queries.}$$

Stack across $i = 1, \ldots, d$ to build $\widehat{\nabla f}(x) \in \mathbb{R}^d$.

Cost grows linearly with d, same scaling problem as before.

The bias–noise trade-off

Noisy oracle: $\tilde{f}(x) = f(x) + \varepsilon$ with $\mathbb{E}[\varepsilon^2] = \sigma^2$.

- Bias of FD estimator: $O(\mu)$.
- Variance of FD estimator: $O(\sigma^2/\mu^2)$.

Optimal trade-off: $\mu \sim \sigma^{1/2}$ gives MSE $O(\sigma)$.

- Small μ: low bias, but variance explodes.
- Large μ: stable, but biased.

In ML we essentially never use coordinate-wise FD, the d factor is fatal.

The Gaussian-smoothed surrogate

Definition [Nesterov and Spokoiny, 2017]:

$$f_\mu(x) = \mathbb{E}_{u \sim \mathcal{N}(0, I_d)}[f(x + \mu u)].$$

Properties: assume f is L-smooth, then

- f_μ is differentiable, even if f is not;
- $|f_\mu(x) - f(x)| \leq \frac{\mu^2}{2} L d$;
- f_μ is itself L-smooth;
- as $\mu \to 0$, $f_\mu \to f$.

The key identity

Stein-type identity:

$$\boxed{\nabla f_\mu(x) = \mathbb{E}_{u\sim\mathcal{N}(0,I)}\left[\frac{f(x+\mu u)-f(x)}{\mu}u\right]}$$

Intuition: apply Stein's lemma to the Gaussian density:

$$\mathbb{E}[u\,g(u)] = \mathbb{E}[\nabla g(u)] \quad \text{for} \quad g(u) = f(x+\mu u).$$

Unbiased estimator of ∇f_μ from a single $(x, x+\mu u)$ pair.

Cost is independent of d.

The two-point estimator

Formula:

$$\hat{g}(x) = \frac{f(x + \mu u) - f(x - \mu u)}{2\mu}\, u, \qquad u \sim \mathcal{N}(0,\, I_d).$$

Properties:

- Unbiased for $\nabla f_\mu(x)$.
- Two queries per step, independent of d.
- Lower variance than the one-point version — the constant cancels.

This is the estimator used by **MeZO**, **ES**, and **ZO-SGD**.

Variance bound and the dimension penalty

Variance [Nesterov and Spokoiny, 2017]:

$$\mathbb{E}\left\|\hat{g}(x)\right\|^2 \leq 2(d+4)\left\|\nabla f(x)\right\|^2 + \frac{\mu^2 L^2 (d+6)^3}{2}.$$

- Leading term: $(d+4)\left\|\nabla f\right\|^2$, dimension penalty of ZO.
- Second term vanishes as $\mu \to 0$.

	One-point	**Two-point**
Estimator	$\frac{f(x+\mu u)}{\mu} u$	$\frac{f(x+\mu u) - f(x-\mu u)}{2\mu} u$
Queries	1	2
Variance at optimum	does not vanish	vanishes as $\|\nabla f\| \to 0$
Used in	bandit convex opt.	ES, MeZO, ZO-SGD

Outline

ZO Gradient Descent

Algorithm.

- Initialize x_0, step size η, smoothing μ.
- For $t = 0, 1, \ldots, T-1$:
  - Sample $u_t \sim \mathcal{N}(0, I_d)$.
  - Query $f(x_t + \mu u_t)$ and $f(x_t - \mu u_t)$.
  - Form $\hat{g}_t = \dfrac{f(x_t + \mu u_t) - f(x_t - \mu u_t)}{2\mu} u_t$.
  - Update $x_{t+1} = x_t - \eta\, \hat{g}_t$.

Just GD with the smoothed surrogate's stochastic gradient.

Convergence: convex, L-smooth

Convergence of ZO-GD [Nesterov and Spokoiny, 2017]

Let f be convex and L-smooth. With $\eta = O(\frac{1}{dL})$ and μ sufficiently small,

$$\mathbb{E}[f(x_T) - f^\star] \leq O\left(\frac{dL \left\|x_0 - x^\star\right\|^2}{T}\right).$$

Compare with GD:

$$O\left(\frac{L \left\|x_0 - x^\star\right\|^2}{T}\right).$$

ZO is a factor of d slower than GD.

Intuition: variance bound inflates each step's progress by factor d.

More Complexity

Setting	**First-order (GD)**	**Zeroth-order**
Convex, L-smooth	$O(L/\varepsilon)$	$\boxed{O(dL/\varepsilon)}$
μ-strongly convex	$O(\kappa \log(1/\varepsilon))$	$\boxed{O(d\,\kappa \log(1/\varepsilon))}$
Nonconvex (stationary)	$O(L/\varepsilon^2)$	$\boxed{O(dL/\varepsilon^2)}$

Conclusion:

- ZO $=$ GD slowed by a factor of d.
- This is the price of not seeing gradients.

For details, see [Ghadimi and Lan, 2013].

ZO-SGD for stochastic ML objectives

Stochastic objective:

$$f(x) \;=\; \mathbb{E}_{\xi}\big[F(x;\xi)\big].$$

Algorithm. Replace $f(x \pm \mu u)$ in the two-point estimator with $F(x \pm \mu u;\, \xi)$ on a minibatch.

For nonconvex L-smooth F, bounded variance σ^2, same complexity as ZO-GD.

Noise from random directions dominates gradient noise.

Application revisit: ZOO adversarial attack

Objective:

$$\min_{\delta} \quad \|\delta\|_2^2 \; + \; c \cdot \mathrm{loss}\big(f(x+\delta),\, y_{\mathsf{target}}\big).$$

Threat model: attacker has only query access to f.

- White-box attack: backprop through f.
- Black-box ZOO: only forward queries.

Why coordinate-wise ZO, not Gaussian? Image perturbations are spatially structured and sparse. A few pixels carry the signal, so one should pick coordinates strategically.

ZOO Algorithm [Chen et al., 2017]

Inner loop.

- **Pick** a random pixel coordinate i.
- **Estimate** $\partial L / \partial \delta_i$ via central difference:

$$\hat{g}_i \;=\; \frac{L(\delta + \mu e_i) - L(\delta - \mu e_i)}{2\mu}.$$

- **Update** δ_i with Adam-style adaptive rule.

Two practical accelerators:

- Importance sampling, focus queries on high-impact pixels.
- Hierarchical attack, optimize at low resolution, then upsample.

ZOO results

Empirical performance.

- MNIST/CIFAR-10: $\sim 99\%$ attack success.
- Distortion comparable to white-box C&W attack.
- ImageNet (Inception-v3): $\sim 90\%$ success, visually imperceptible perturbations.

Implications.

- No model internals used, only forward queries.
- Cloud-hosted classifiers are not protected by hiding the weights.
- Motivates query-budget-based defenses (rate limiting, query distribution checks).

Wrap-up

Week 14: Key takeaways

- Black-box optimization, adversarial attacks, fine-tuning, and RL.
- Gaussian smoothing, properties of surrogate.
- gradient estimator, one-point or two-point estimators.
- ZO complexity across convex, strongly convex, and nonconvex regimes.

Thank you for your attendance!

Questions?

References I

Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. (2017).

Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models.

In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26.

Ghadimi, S. and Lan, G. (2013).

Stochastic first-and zeroth-order methods for nonconvex stochastic programming.

SIAM journal on optimization, 23(4):2341–2368.

Malladi, S., Gao, T., Nichani, E., Damian, A., Lee, J. D., Chen, D., and Arora, S. (2023).

Fine-tuning language models with just forward passes.

Advances in Neural Information Processing Systems, 36:53038–53075.

Nesterov, Y. and Spokoiny, V. (2017).

Random gradient-free minimization of convex functions.

Foundations of Computational Mathematics, 17(2):527–566.

References II

Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I. (2017). Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*.