AIM-ZO: ACTIVATION-INFORMED SUBSPACE MAINTENANCE FOR ZEROTH-ORDER LLM FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Zeroth-order (ZO) optimization offers a memory-efficient alternative for LLM fine-tuning by estimating updates only from forward evaluations of perturbed parameters, without backpropagation or activation storage. However, in billion-parameter LLMs, isotropic perturbations often waste many forward evaluations on weakly informative directions. To make these evaluations more informative, existing ZO methods restrict perturbations to low-dimensional subspaces. Yet the quality of these subspaces is critical: overly compressed or poorly maintained spaces can miss useful update directions. **To obtain a high-quality subspace for ZO updates**, this paper proposes AIM-ZO, a ZO fine-tuning method based on Activation-Informed Subspace Maintenance. AIM-ZO uses forward activations as local directional information and continuously integrates them into a broad, evolving subspace over training. To access broader gradient-relevant structure while keeping individual perturbations low-dimensional, AIM-ZO activates only a smaller set of shared and sampled directions, decoupling the maintained width from the active width. We evaluate AIM-ZO across 5 LLMs and 11 downstream tasks under matched forward-evaluation budgets; its six-task average exceeds the strongest fully evaluated ZO baseline by 1.26 percentage points on OPT-2.7B and MeZO by 2.85 percentage points on OPT-30B. Our code is available at https://anonymous.4open.science/r/AIM-ZO-0155/

1 INTRODUCTION

Fine-tuning large language models (LLMs) requires substantial memory because backpropagation stores intermediate activations for gradient computation. Zeroth-order (ZO) optimization avoids this cost by estimating gradients from forward loss evaluations at perturbed parameter points (Ghadimi & Lan, 2013; Nesterov & Spokoiny, 2017). **MeZO** shows that this approach can fine-tune LLMs with a memory footprint close to inference (Malladi et al., 2023). However, MeZO samples perturbation directions randomly from the full parameter space. As illustrated on the left of Figure 1, in billion-parameter models, many such random directions **are poorly aligned with useful update directions**, so a large fraction of forward evaluations provide only **weak optimization signals**.

Prior studies have shown that gradients can concentrate in low-dimensional subspaces that remain relatively stable over parts of training (Gur-Ari et al., 2018; Jaiswal et al., 2025). Building on this observation, recent ZO methods exploit low-dimensional structure through low-rank perturbations, random projections, or forward activations (Chen et al., 2025; Yu et al., 2025; Lin et al., 2026; Dong et al., 2026). Their effectiveness, however, depends critically on the quality of the perturbation subspace. Specifically, **1)** random or overly compressed perturbation subspaces may **miss useful gradient information**; and **2)** simply enlarging the perturbation subspace can capture more gradient information but also **make finite-sample estimation more difficult** as its width grows. These challenges motivate our central research question:

How can we obtain a high-quality subspace for low-dimensional ZO fine-tuning?

AGZO constructs perturbation subspaces from current activations, rebuilding them at each step without accumulating activation structure across training (Lin et al., 2026). This repeated construction adds computational cost and can leave the subspace sensitive to stochastic variation; using the same subspace for perturbations also couples gradient coverage to estimation variance.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

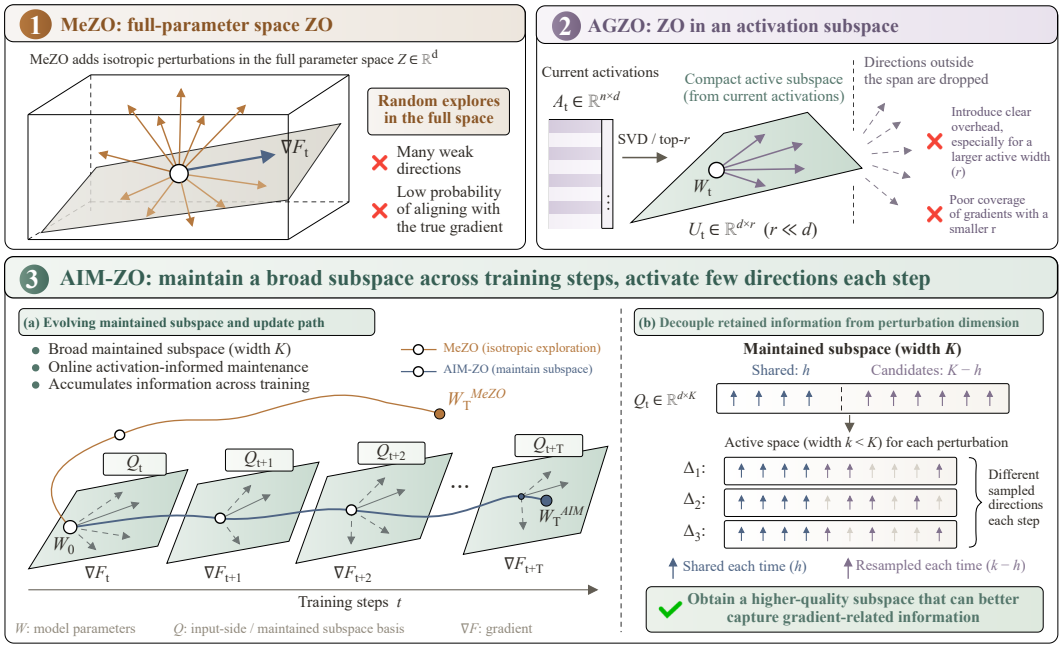

Figure 1: Perturbation spaces. MeZO explores the full parameter space; AGZO constructs a subspace from current activations. AIM-ZO maintains a width-K subspace across training and activates a width-k subspace with h shared and $k - h$ sampled directions per perturbation.

To address these limitations, we propose AIM-ZO, a ZO fine-tuning method based on **Activation-Informed Subspace Maintenance**. AIM-ZO continuously accumulates activation information across training into a broader width-K maintained subspace. Each perturbation then activates only $k < K$ directions, combining h shared directions with $k - h$ directions sampled from the remaining maintained space. This decouples the amount of gradient-relevant information retained by the maintained subspace from the dimensionality used in each perturbation, allowing low-dimensional perturbations to access different parts of a broader subspace over time. A population-centered one-sided estimator aggregates these perturbations into the parameter update. We evaluate AIM-ZO across 5 LLMs, ranging from 0.6B to 30B parameters, and 11 downstream tasks. AIM-ZO achieves stronger overall performance than existing ZO baselines, including a 1.26 percentage-point improvement over the strongest fully evaluated baseline on OPT-2.7B and a 2.85 percentage-point improvement over MeZO on OPT-30B. Our contributions are as follows:

- We propose AIM-ZO, which accumulates forward activation information across training to maintain a broad, evolving perturbation subspace. Our analysis gives a lower bound on its gradient capture, which we also measure empirically.
- We decouple maintained and active subspace widths, allowing low-dimensional perturbations to access a broader space. Our analysis identifies when estimates from the active subspace align better with the gradient than estimates from the full maintained subspace.
- We comprehensively evaluate AIM-ZO across 5 LLMs and 11 downstream tasks. The results show stronger overall performance than existing ZO baselines, while peak-memory measurements on Qwen3-0.6B show only modest overhead relative to MeZO.

2 Related Work

Zeroth-Order Fine-Tuning. ZO fine-tuning estimates parameter updates from forward loss evaluations without backpropagation. We consider $\min_w F(w)$, where $F(w) = \mathbb{E}_\xi[\mathcal{L}(w;\xi)]$ and w denotes the trainable parameters. For matrix parameters, we use $w = \mathrm{vec}(W)$ and $z = \mathrm{vec}(Z)$; full-model vectors concatenate the corresponding parameter blocks. Given a perturbation z and scale $\epsilon > 0$, a standard two-sided estimator for ZO is as follows:

$$\widehat{g} = \frac{\mathcal{L}(w + \epsilon z;\xi) - \mathcal{L}(w - \epsilon z;\xi)}{2\epsilon} z.$$

MeZO samples $z \sim \mathcal{N}(0, I)$ and regenerates perturbations from random seeds to avoid storing full perturbation vectors (Malladi et al., 2023). Subsequent methods improve how perturbations are generated or how gradient estimates are converted into updates.

Curvature-Guided Scaling and Coordinate Selection. These methods adjust perturbations according to coordinate-level information. HiZOO uses $z = \Sigma^{1/2}u$, where $u \sim \mathcal{N}(0, I)$ and Σ is a diagonal inverse-Hessian approximation estimated from forward evaluations; its factored variant HiZOO-L reduces the memory required for this state (Zhao et al., 2025). CurvZO constructs sparse perturbations $z = m \odot u$, sampling the mask m using accumulated curvature-proxy scores and correcting nonuniform sampling through probability reweighting (Wang et al., 2026). Sparse MeZO instead uses parameter magnitudes to select a subset of coordinates for fine-tuning (Liu et al., 2025).

Structured Perturbations and Random Subspaces. Another approach reduces the estimation or computational burden through low-rank structure and subspace restriction. LoZO constructs $Z = UV^\top$, reusing the random right factor V across multiple steps while resampling U (Chen et al., 2025). SubZero uses $Z = URV^\top$, with periodically refreshed random orthonormal bases U, V and fresh coefficients R (Yu et al., 2025). ZO-Muon uses $Z = QA$ with a random orthonormal basis Q, then applies Muon-style orthogonalization to the reduced-space estimate before mapping it back to the parameter space (Lang et al., 2026). These methods construct and reuse structured perturbation spaces, but their random bases do not explicitly target gradient-relevant directions.

Activation-Informed Subspaces. AGZO extracts a compact perturbation subspace from each minibatch's activations using power iteration (Lin et al., 2026), while ZO-Act reuses a subspace derived from initial activations (Dong et al., 2026). AGZO adapts to current activations, but repeated extraction incurs computational cost and does not retain activation structure observed at earlier training steps. ZO-Act avoids repeated extraction, but its fixed subspace cannot track changes in model representations. Both methods restrict perturbations to a truncated activation span, which excludes gradient components outside that span. Widening the span can improve coverage, but also increases basis storage and, when fully activated, estimation dimension.

AIM-ZO accumulates activation information across steps in a width-K maintained subspace, while each perturbation activates only $k < K$ directions: h shared directions and $k - h$ sampled directions. Directions omitted from one perturbation remain available to others, so the maintained subspace can cover more candidate directions without increasing the dimension of every estimate. Additional related work and a design comparison appear in Appendix I.

3 Methodology: AIM-ZO

AIM-ZO maintains a broad subspace using forward activations accumulated across training. At each step, it selects a smaller active subspace for each perturbation and aggregates the one-sided population evaluations into a parameter update.

3.1 Notation and Setup

Let $\mathcal{W} = \{W_\ell\}_{\ell=1}^{L}$ denote the trainable matrices, with $W_\ell \in \mathbb{R}^{p_\ell \times d_\ell}$ for layer ℓ. At iteration t, we draw a minibatch ξ_t and write $G_{\ell,t} = \nabla_{W_\ell}\mathcal{L}(\mathcal{W}_t; \xi_t)$ for the layer gradient used in the analysis. A perturbation population comprises N perturbations whose associated perturbed parameter points are evaluated on the same minibatch to estimate an update.

Throughout, ℓ, t, and i index layers, training iterations, and population members, respectively. For matrices of the same shape, write $\langle A, B\rangle_F = \mathrm{tr}(A^\top B)$. We use $\|A\|_F = \sqrt{\langle A, A\rangle_F}$ for the Frobenius norm and $\|A\|_2$ for the spectral norm. For any column-orthonormal basis Q, $P_Q = QQ^\top$ denotes the orthogonal projector onto its column space.

3.2 Activation-Informed Subspace Maintenance

At iteration t, let $H_{\ell,t} \in \mathbb{R}^{n_{\ell,t} \times d_\ell}$ be the input activations of a linear layer, where $n_{\ell,t}$ counts the activation rows and $Y_{\ell,t} = H_{\ell,t}W_{\ell,t}^\top$. The weight gradient satisfies

$$G_{\ell,t} = E_{\ell,t}^\top H_{\ell,t}, \tag{1}$$

Algorithm 1 One training run of AIM-ZO

Require: Iterations T; weights $\mathcal{W}_0$; bases $\{\widetilde{Q}_{\ell,0}\}$; widths K, h, k with $0 \le h \le k \le K$; population size $N \ge 2$; schedules $\{\epsilon_t\}, \{\eta_t\}$; Oja step size η_q

for $t = 0, \dots, T-1$ **do**
  Sample ξ_t; run a centre forward pass to collect $\{H_{\ell,t}\}$
  For each layer ℓ, update $Q^{\text{wide}}_{\ell,t}$ by Equation 2
  for $i = 1, \dots, N$ **do**
    For each layer ℓ, sample $S_{\ell,t,i}$, $a_{\ell,t,i}$, and $b_{\ell,t,i}$; form $Q^{\text{act}}_{\ell,t,i}$ and $Z_{\ell,t,i}$ by Equations 3–4
    Evaluate $y^+_{t,i} = \mathcal{L}(\mathcal{W}_t + \epsilon_t \mathcal{Z}_{t,i}; \xi_t)$ with $\mathcal{Z}_{t,i} = \{Z_{\ell,t,i}\}_{\ell=1}^{L}$
  end for
  For each ℓ, compute $\widehat{G}_{\ell,t}$ by Equation 5 and set $W_{\ell,t+1} = W_{\ell,t} - \eta_t \widehat{G}_{\ell,t}$
  Store $\widetilde{Q}_{\ell,t+1} \leftarrow Q^{\text{wide}}_{\ell,t}$ for all ℓ
end for
return $\mathcal{W}_T$

where $E_{\ell,t}$ is the loss derivative with respect to $Y_{\ell,t}$. Thus, the row space of $G_{\ell,t}$ lies in the row space of $H_{\ell,t}$ (Lin et al., 2026). We maintain an orthonormal basis $Q^{\text{wide}}_{\ell,t} \in \mathbb{R}^{d_\ell \times K}$ to track prominent activation directions across iterations.

An unperturbed centre pass updates the stored basis by Oja's rule (Huang et al., 2021):

$$Q^{\text{wide}}_{\ell,t} = \text{qf}\Big(\widetilde{Q}_{\ell,t} + \eta_q n^{-1}_{\ell,t} H^\top_{\ell,t}(H_{\ell,t}\widetilde{Q}_{\ell,t})\Big), \tag{2}$$

Here η_q is the Oja step size, and qf returns a thin, unpivoted QR basis with fixed signs. A Gaussian matrix initializes the basis. One update serves all N perturbations, after which we store $\widetilde{Q}_{\ell,t+1} = Q^{\text{wide}}_{\ell,t}$. Appendix D.1 compares gradient capture and construction time with AGZO's current-batch reconstruction (Lin et al., 2026); Appendix E.2 reports an online maintenance-frequency ablation.

3.3 Active-subspace selection and parameter updates

From the width-K subspace maintained across iterations, AIM-ZO selects a width-k active subspace for each perturbation. Different perturbations can thus access different parts of the maintained subspace while keeping each estimate low-dimensional. An online comparison with a shared active subspace is in Appendix E.1.

For the layer-wise expressions below, we fix an iteration and suppress the layer and iteration indices. For each perturbation, we form an active subspace from the same first h shared basis directions $Q_h = Q^{\text{wide}}[:, 1\!:\!h]$ and $s = k - h$ columns sampled from the remaining pool, with $0 \le h \le k \le K$ and $k \ge 1$. Specifically, perturbation i uses

$$Q^{\text{act}}_i = \big[Q_h, Q^{\text{wide}}[:, S_i]\big], \qquad |S_i| = s, \tag{3}$$

where Q^{act}_i is a basis of the active subspace and S_i is sampled uniformly without replacement from $\{h+1, \dots, K\}$, independently across population members and layers. The shared columns follow the stored basis order.

Following the low-rank perturbation construction of LoZO (Chen et al., 2025), we independently draw $a_i \sim \mathcal{N}(0, I_p)$ and $b_i \sim \mathcal{N}(0, I_k)$ and form

$$U_i = a_i b_i^\top (Q^{\text{act}}_i)^\top, \qquad \alpha_i = \frac{\sqrt{pd}}{\|U_i\|_F + 10^{-12}}, \qquad Z_i = \alpha_i U_i. \tag{4}$$

This normalization matches the layer-wise perturbation magnitude to the root-mean-square Frobenius norm of MeZO's isotropic Gaussian perturbations.

For $N \ge 2$, each population member perturbs all layers on the same minibatch: $\mathcal{Z}_i = \{Z_{\ell,i}\}_{\ell=1}^{L}$, $y^+_i = \mathcal{L}(\mathcal{W} + \epsilon \mathcal{Z}_i; \xi)$, and $\bar{y}^+ = N^{-1}\sum_{i=1}^{N} y^+_i$. For each layer, the REINFORCE leave-one-out (RLOO) estimator is

$$\widehat{G} = \frac{1}{(N-1)\epsilon} \sum_{i=1}^{N} (y^+_i - \bar{y}^+) Z_i. \tag{5}$$

Each iteration uses one unperturbed and N perturbed forward passes, followed by a parameter update with learning rate η (Algorithm 1). Further ablations of population size and perturbation coefficient rank are in Appendix E.

4 Analysis of AIM-ZO Components

This section characterizes gradient capture by the activation-maintained subspace and derives a condition under which decoupling maintained and active widths improves directional alignment. It then examines one-sided and two-sided estimation at a matched evaluation budget. Proofs are provided in Appendices A–B.

The analysis uses the notation of Section 3.1 and suppresses the layer index: $W \in \mathbb{R}^{p\times d}$ denotes one weight matrix, f its minibatch loss with other weights fixed, and $G = \nabla f(W)$.

4.1 Maintaining gradient-relevant information

Gradient structure and capture. Let $Q_t := Q_t^{\text{wide}}$ denote the maintained basis. Define $S_t = H_t^\top H_t/n_t$ and $M_t = \mathbb{E}[S_t \mid \mathcal{F}_{t-1}]$, where $\mathcal{F}_{t-1}$ is the history before sampling the current batch. Let $V_t^\star$ span the top-K eigenspace of M_t. The tracking error between the maintained and target subspaces is

$$\varepsilon_{\text{trk},t} := \|P_{Q_t} - P_{V_t^\star}\|_2.$$

Let $\mathcal{I}_t$ index a sliding window of T_{w} training steps with nonzero minibatch gradients G_τ. All gradients in this window are compared against the same $V_t^\star$ and Q_t. Prior observations of concentrated and locally stable gradient directions (Gur-Ari et al., 2018; Jaiswal et al., 2025) motivate the following shared-subspace assumption.

Assumption 1 (Local shared gradient structure)**.** *For each window $\mathcal{I}_t$, there exists a shared column-orthonormal basis $U_t^\star \in \mathbb{R}^{d\times r}$, with $1 \le r \le K$, such that*

$$\left[\frac{1}{T_{\text{w}}}\sum_{\tau\in\mathcal{I}_t}\frac{\|G_\tau(I - P_{U_t^\star})\|_F^2}{\|G_\tau\|_F^2}\right]^{1/2} \le \varepsilon_{\text{sub},t}. \tag{6}$$

The activation target must also cover the shared gradient subspace.

Assumption 2 (Activation–gradient alignment)**.** *The width-K population activation subspace covers the gradient energy within the shared rank-r space spanned by the same $U_t^\star$ from Assumption 1, up to residual $\varepsilon_{\text{act},t}$:*

$$\left[\frac{1}{T_{\text{w}}}\sum_{\tau\in\mathcal{I}_t}\frac{\|G_\tau P_{U_t^\star}(I - P_{V_t^\star})\|_F^2}{\|G_\tau\|_F^2}\right]^{1/2} \le \varepsilon_{\text{act},t}. \tag{7}$$

Figure 2(a,b) reports diagnostics for these two residuals: the shared-gradient residual is smaller in middle and late layers, and the activation-alignment residual decreases with K.

Tracking the activation subspace. For the Oja update in Equation 2, define target drift as $\omega_t = \|P_{V_t^\star} - P_{V_{t-1}^\star}\|_F/\sqrt{2}$ and set $t = 0$ after burn-in.

Assumption 3 (Burn-in initialization)**.** *For $0 \le \delta_{\text{init}} < 1$, the initial error $\Phi_0 := \frac{1}{2}\|P_{Q_0} - P_{V_0^\star}\|_F^2$ is at most $\varepsilon_{\text{burn}}^2$ with probability at least $1 - \delta_{\text{init}}$.*

Stationary Oja results support this initialization condition under their sampling and step-size requirements (Huang et al., 2021). Post-burn-in tracking further requires the following conditions.

Assumption 4 (Post-burn-in tracking conditions)**.** *For $1 \le t \le T$, the following conditions hold almost surely: (i)* spectral separation, $\lambda_K(M_t) - \lambda_{K+1}(M_t) \ge \gamma > 0$*; (ii)* bounded observations, $\|S_t\|_2 \le \Lambda$ *and* $\mathbb{E}[\|S_t\|_2^2 \mid \mathcal{F}_{t-1}] \le \Sigma^2$*; and (iii)* bounded target drift, $\omega_t \le \bar{\omega}$*. Here $\lambda_j(M_t)$ is the jth largest eigenvalue of M_t.*

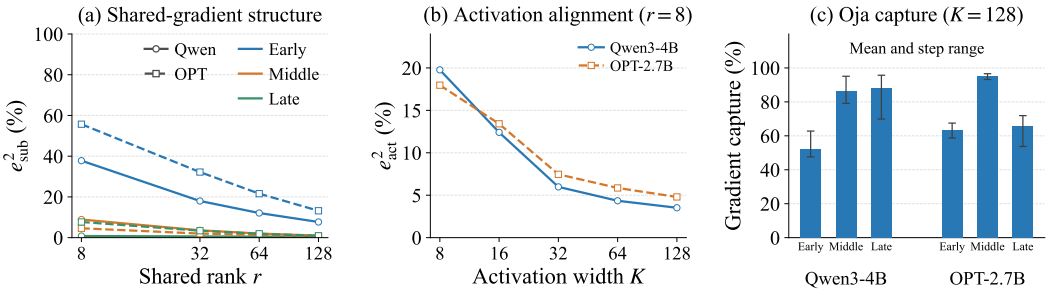

Figure 2: RTE diagnostics over 16 consecutive training steps using exact training-set mean gradients. (a) Held-out shared-gradient residual. (b) Activation-alignment residual ($r = 8$, averaged over three layers). (c) Maintained-subspace capture ($K = 128$); bars show means and whiskers min–max ranges. Protocols are in Appendix D.

Theorem 1 (Post-burn-in tracking). *Suppose Assumptions 3 and 4 and the step-size, drift, noise, and burn-in conditions in Appendix A hold. Then, for $\eta_q > 0$, tracking-bound failure probability $0 < \delta < 1$, and $t \le T$,*

$$\mathbb{E}\varepsilon^2_{\text{trk},t} \le \underbrace{O\left(e^{-\eta_q \gamma t/2}\varepsilon^2_{\text{burn}}\right)}_{\textit{initialization}} + \underbrace{O\left(\frac{\bar{\omega}^2}{\eta_q^2\gamma^2}\right)}_{\textit{tracking lag}} + \underbrace{O\left(\frac{K\eta_q\Sigma^2}{\gamma}\right)}_{\textit{stochastic error}} + \underbrace{O(K(\delta+\delta_{\text{init}}))}_{\textit{failure events}}. \tag{8}$$

A larger eigengap γ reduces tracking lag and stochastic error, while larger drift $\bar{\omega}$ or observation second moment Σ^2 increases them. Within the admissible range, increasing η_q speeds initialization decay and reduces lag but increases stochastic error; the last two terms also scale with K. Together with the structural and alignment residuals, this bound gives the following capture guarantee.

Proposition 1 (Maintained-subspace capture). *Under Assumptions 1–2,*

$$\frac{1}{T_{\text{w}}}\sum_{\tau\in\mathcal{I}_t}\frac{\|G_\tau P_{Q_t}\|_F^2}{\|G_\tau\|_F^2} \ge 1 - (\varepsilon_{\text{sub},t} + \varepsilon_{\text{act},t} + \varepsilon_{\text{trk},t})^2. \tag{9}$$

Figure 2(c) shows 52.2%–95.1% gradient capture by the maintained subspace.

4.2 Decoupling Maintained and Active Widths

A wider maintained subspace can improve gradient capture, but using all K basis directions in each estimate increases variance. Fix $G \neq 0$ and an ordered orthonormal maintained basis $Q \in \mathbb{R}^{d\times K}$. As in Section 3.3, Q_S uses $h = \theta k \in \mathbb{Z}$ shared columns and $k - h$ sampled columns, where $0 < \theta < 1$ and $1 \le k \le K$. Define

$$C_Q = \frac{\|GP_Q\|_F^2}{\|G\|_F^2} > 0, \qquad \tau_h = \frac{\|GP_{Q_h}\|_F^2}{\|GP_Q\|_F^2}, \qquad \rho = \frac{k-h}{K-h}.$$

For standard Gaussian $R \in \mathbb{R}^{p\times k}$, let $Z = RQ_S^\top$ and $\widehat{G}_{k,0} = \langle G, Z\rangle_F Z$. Applying the Gaussian subspace moments of Yu et al. (2025, Theorem 2) conditional on Q_S gives

$$\mathbb{E}_R\|\widehat{G}_{k,0} - G\|_F^2 = \underbrace{(pk+1)\|GP_{Q_S}\|_F^2}_{\text{estimation variance}} + \underbrace{\|G(I-P_{Q_S})\|_F^2}_{\text{squared projection bias}}. \tag{10}$$

Full activation retains all maintained signal at variance factor $pK + 1$, while increasing K at fixed h, k lowers tail inclusion ρ. The sampling rule and fixed-subspace cosine identity of Lin et al. (2026, Theorem 5.4) yield the following result.

Proposition 2 (Active capture and directional alignment). *Let $A_{K,k}$ denote the expected fraction of maintained gradient energy retained by the active subspace, $J_{K,k} := \mathbb{E}\cos(\widehat{G}_{k,0}, G)$, and $\beta_D := \Gamma(D/2)/[\sqrt{\pi}\Gamma((D+1)/2)]$, with $\cos(0, G) = 0$. Then*

$$\frac{\mathbb{E}_S\|GP_{Q_S}\|_F^2}{\|G\|_F^2} = C_Q A_{K,k}, \qquad A_{K,k} = \tau_h + \rho(1-\tau_h). \tag{11}$$

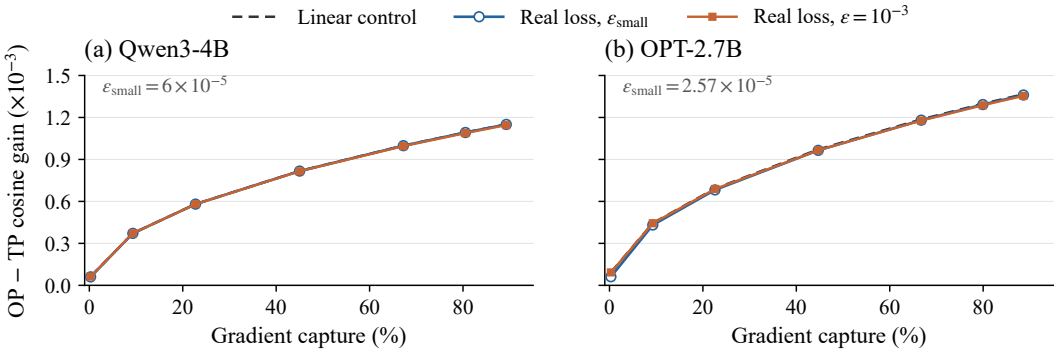

Figure 3: OP–TP cosine gain versus gradient capture on RTE. OP16 and TP8 each use 16 perturbed loss evaluations at $k = 64$. Points average matched populations across checkpoints and batches; dashed curves show the corresponding linear control.

For $k < K$, if $A_{K,k} > \beta_{pK}/\beta_{pk}$, then $J_{K,k} > J_{K,K}$: partial activation has higher expected cosine with G than full activation.

Since $\beta_D \simeq \sqrt{2/(\pi D)}$ for large D, the sufficient threshold is approximately $\sqrt{k/K}$. Offline capture diagnostics meet this condition in the evaluated settings (Appendix D.2). Online scans of maintained and shared widths are in Appendices E.3, E.4, and E.5; proofs are in Appendix B.

4.3 One-sided estimation in a high-capture subspace

Online training and offline ablations favor the RLOO estimator used by AIM-ZO over two-sided estimation (Appendices F.1 and D.3). To examine how subspace quality affects this advantage, we vary gradient capture at fixed active width and compare the directional alignment of the two estimators through $\Delta_{\cos} = \cos(\widehat{G}_{\text{OP}}, G) - \cos(\widehat{G}_{\text{TP}}, G)$.

Figure 3 shows that the OP–TP alignment gain increases with gradient capture. Actual-loss results closely follow a linear control constructed from the same-batch gradient, indicating that the trend is largely explained by the first-order signal. Further results are in Appendix F.2.

5 Experiments

In this section, we empirically evaluate the proposed AIM-ZO method over a wide collection of datasets and LLMs. Section 5.1 compares the performance of AIM-ZO on two mainly chosen LLMs for ZO. Section 5.3 evaluates the peak GPU memory consumption. Additional experiments are in the Appendix: Appendix D complements the analysis in Section 4 and Appendix H provides broader fine-tuning results, including all 11 OPT-2.7B tasks and additional evaluations on Qwen3 LLMs.

Experiment Settings. We consider a wide collection of LLMs, including OPT-2.7B, OPT-13B, OPT-30B, and Qwen3-8B-Base; supplementary experiments consider two more LLMs, Qwen3-0.6B-Base and Qwen3-4B. We fine-tune LLMs over the 11 downstream tasks included in Malladi et al. (2023); OPT-2.7B covers all 11 tasks, while the main tables report six-task comparisons.

Baselines. We include various baselines, including **1)** zero-shot scores provide a non-training reference. **2)** MeZO Malladi et al. (2023) with full perturbation. **3)** HiZOO (Zhao et al., 2025) and CurvZO (Wang et al., 2026) with curvature-guided scaling, and **4)** LoZO (Chen et al., 2025), AGZO (Lin et al., 2026), and ZO-Muon (Lang et al., 2026) as subspace-based perturbation baselines.

Implementation details. All fine-tuning methods use BF16 and approximately 40,000 training forward evaluations, with update counts adjusted for each method's forward evaluations per update. AIM-ZO uses 16 forward evaluations per update and thus runs for 2,500 updates, with $K = 128$, $h = 48$, and $k = 64$. Most results use three seeds; tables report mean percentages and available sample standard deviations. Data splits and evaluation protocols are in Appendix C; component ablations are in Appendix E.

Table 1: Multi-method comparison on OPT-2.7B and OPT-13B. Scores are accuracy (%) except SQuAD (F1). Avg. is the unweighted mean over six tasks. Best mean scores are bold; subscripts report available sample standard deviations. Evaluation details are in Appendix H.

Method	RTE	BoolQ	SST-2	WiC	WSC	SQuAD	Avg.
			OPT-2.7B				
Zero-shot	55.23	52.87	56.65	54.86	36.54	26.92	47.18
MeZO	$65.13_{\pm 1.67}$	$66.10_{\pm 2.26}$	$92.50_{\pm 0.54}$	$58.53_{\pm 0.41}$	$54.81_{\pm 1.52}$	$80.99_{\pm 1.42}$	69.68
CurvZO	$64.53_{\pm 4.83}$	$68.01_{\pm 0.62}$	$\mathbf{92.96}_{\pm 0.31}$	$57.96_{\pm 2.07}$	$46.47_{\pm 7.28}$	$58.71_{\pm 2.47}$	64.77
HiZOO	$60.29_{\pm 2.94}$	$67.31_{\pm 1.34}$	$91.77_{\pm 0.67}$	$58.88_{\pm 0.80}$	$53.53_{\pm 2.42}$	$73.81_{\pm 0.90}$	67.60
AGZO	$64.40_{\pm 3.10}$	$66.26_{\pm 1.46}$	$92.59_{\pm 0.51}$	$57.02_{\pm 1.73}$	$47.76_{\pm 2.22}$	$29.82_{\pm 5.54}$	59.64
ZO-Muon	$63.32_{\pm 2.53}$	$\mathbf{68.90}_{\pm 1.12}$	$92.75_{\pm 0.52}$	$\mathbf{61.13}_{\pm 1.12}$	$51.60_{\pm 3.38}$	$78.93_{\pm 1.13}$	69.44
LoZO	$61.88_{\pm 4.48}$	$65.33_{\pm 0.73}$	$91.51_{\pm 0.83}$	$53.67_{\pm 2.70}$	$50.96_{\pm 7.26}$	$43.52_{\pm 1.24}$	61.15
AIM-ZO (OURS)	$\mathbf{67.51}_{\pm 3.18}$	$67.22_{\pm 1.67}$	$92.87_{\pm 0.47}$	$60.13_{\pm 1.70}$	$\mathbf{56.73}_{\pm 4.35}$	$\mathbf{81.19}_{\pm 0.49}$	**70.94**
			OPT-13B				
Zero-shot	61.73	60.46	60.21	55.17	40.38	40.80	53.13
MeZO	$67.36_{\pm 1.54}$	$71.08_{\pm 1.37}$	$92.00_{\pm 0.69}$	$58.58_{\pm 1.93}$	$54.49_{\pm 8.72}$	$58.64_{\pm 1.42}$	67.02
CurvZO	$52.23_{\pm 2.73}$	$61.73_{\pm 0.42}$	$49.39_{\pm 0.78}$	$49.01_{\pm 1.07}$	$\mathbf{61.22}_{\pm 3.89}$	$65.56_{\pm 1.42}$	56.52
HiZOO	$\mathbf{73.65}_{\pm 1.65}$	$71.58_{\pm 0.12}$	$91.63_{\pm 1.71}$	$\mathbf{59.82}_{\pm 0.59}$	$58.01_{\pm 1.11}$	$75.45_{\pm 0.74}$	71.69
AGZO	$52.59_{\pm 5.76}$	$61.43_{\pm 1.10}$	$88.23_{\pm 0.98}$	$49.43_{\pm 2.20}$	$57.37_{\pm 2.00}$	$75.24_{\pm 1.24}$	64.05
ZO-Muon	$63.06_{\pm 1.50}$	$66.53_{\pm 1.39}$	$91.78_{\pm 0.65}$	$56.79_{\pm 0.94}$	$50.00_{\pm 5.85}$	$80.35_{\pm 0.93}$	68.08
LoZO	$49.70_{\pm 2.80}$	$62.15_{\pm 0.07}$	$50.27_{\pm 0.52}$	$47.39_{\pm 1.01}$	$57.37_{\pm 5.88}$	$47.40_{\pm 6.68}$	52.38
AIM-ZO (OURS)	$70.69_{\pm 2.97}$	$\mathbf{71.93}_{\pm 2.65}$	$\mathbf{93.78}_{\pm 0.39}$	$59.52_{\pm 0.39}$	$57.05_{\pm 2.42}$	$\mathbf{81.58}_{\pm 0.25}$	**72.42**

5.1 COMPARISON WITH ZEROTH-ORDER METHODS

Table 1 compares AIM-ZO with six representative ZO baselines on OPT-2.7B and OPT-13B under matched forward-evaluation budgets. All methods are evaluated using our unified protocol, which may differ from the settings reported in their original papers.

Observation 1: AIM-ZO achieves the strongest overall performance across tasks. AIM-ZO obtains the highest six-task average among methods with complete results on both OPT-2.7B (70.94%) and OPT-13B (72.42%). Relative to MeZO, it improves all six reported tasks at both model scales, and further achieves higher mean performance on nine of the full 11 OPT-2.7B tasks (Appendix H). Although several baselines remain stronger on individual tasks, AIM-ZO avoids the large task-specific degradations observed for some alternatives and therefore provides more balanced performance across the benchmark.

Observation 2: AIM-ZO consistently improves over existing subspace-based ZO methods. Compared with representative subspace methods, including LoZO, AGZO, and ZO-Muon, AIM-ZO achieves the highest six-task average at both scales. On OPT-2.7B, it improves the average from 69.44% for the strongest subspace baseline, ZO-Muon, to 70.94%; on OPT-13B, the corresponding gap increases from 68.08% to 72.42%. The advantage is particularly pronounced over AGZO and LoZO, whose compact perturbation spaces show substantial degradation on several tasks. These results suggest that the quality and maintenance of the perturbation subspace matter beyond simply restricting ZO updates to a low-dimensional space.

Besides OPT-2.7B and OPT-13B, Table 2 extends the comparison to OPT-30B and Qwen3-8B. On OPT-30B, AIM-ZO improves over MeZO on all six tasks, raising the average by 2.85 percentage points. On Qwen3-8B, it improves on five of six tasks and raises the average by 0.93 points over MeZO; SQuAD has the largest gain, while SST-2 declines. These results show stronger gains as OPT scales and positive gains on Qwen3.

5.2 ABLATION STUDIES

We ablate the three main design choices of AIM-ZO. First, **online subspace maintenance** is beneficial: on OPT-2.7B RTE, updating the maintained subspace every step achieves higher mean performance than either a fixed activation subspace or less frequent updates (Appendix E). Second, **per-perturbation active-subspace resampling** consistently improves over using one common active subspace for the whole population across all four tested tasks, supporting the benefit of exposing different perturbations to different parts of the maintained space (Appendix E.1). Third, the **population-centered one-sided estimator** outperforms the two-sided alternative at both the best

Table 2: Scaling and model-family comparisons on OPT-30B and Qwen3-8B (%). SQuAD uses F1; other tasks use accuracy. Avg. is the unweighted mean over six tasks. Best mean scores are bold; subscripts report standard deviations. Qwen3-8B entries use three seeds and development-selected checkpoints.

OPT-30B							
Method	RTE	BoolQ	SST-2	WiC	WSC	SQuAD	Avg.
Zero-shot	53.79	40.90	58.83	52.98	37.50	46.48	48.41
MeZO	$63.25_{\pm 1.79}$	$70.74_{\pm 1.17}$	$91.08_{\pm 0.36}$	$55.96_{\pm 1.81}$	$58.08_{\pm 2.85}$	$80.22_{\pm 0.99}$	69.89
AIM-ZO (OURS)	$\mathbf{68.16}_{\pm 2.66}$	$\mathbf{74.90}_{\pm 1.53}$	$\mathbf{93.47}_{\pm 0.56}$	$\mathbf{57.71}_{\pm 0.91}$	$\mathbf{59.23}_{\pm 2.77}$	$\mathbf{82.96}_{\pm 0.69}$	**72.74**
Qwen3-8B-Base							
Method	RTE	BoolQ	SST-2	WiC	MultiRC	SQuAD	Avg.
Zero-shot	86.28	74.60	58.49	65.67	72.77	83.67	73.58
MeZO	$88.33_{\pm 1.37}$	$85.75_{\pm 0.36}$	$\mathbf{90.94}_{\pm 0.40}$	$67.92_{\pm 1.31}$	$86.01_{\pm 0.48}$	$86.90_{\pm 1.95}$	84.31
AGZO	$86.88_{\pm 1.50}$	$79.40_{\pm 0.85}$	$81.42_{\pm 2.88}$	$65.78_{\pm 1.75}$	$83.76_{\pm 1.88}$	$68.20_{\pm 4.77}$	77.57
AIM-ZO (OURS)	$\mathbf{89.53}_{\pm 1.08}$	$\mathbf{85.76}_{\pm 0.13}$	$89.49_{\pm 0.81}$	$\mathbf{68.18}_{\pm 1.51}$	$\mathbf{86.68}_{\pm 0.16}$	$\mathbf{91.80}_{\pm 0.42}$	**85.24**

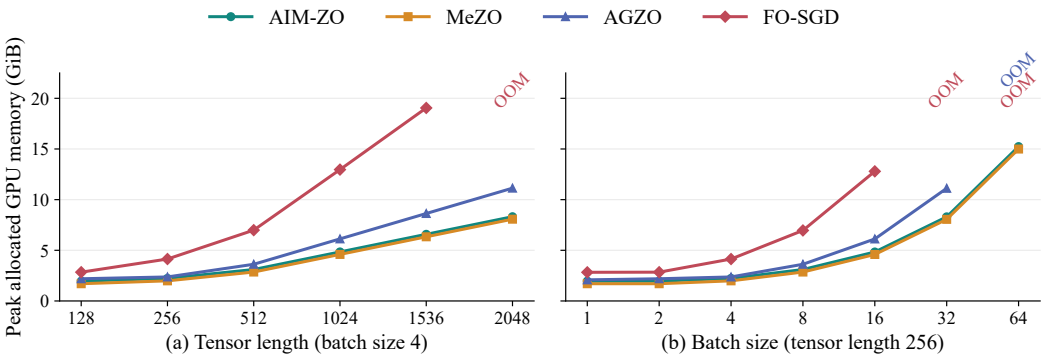

Figure 4: Peak allocated GPU memory on Qwen3-0.6B-Base/DROP with BF16 weights. Left: varying tensor length at batch size four. Right: varying batch size at tensor length 256. OOM marks observed failures, not measured memory values.

and final checkpoints on OPT-2.7B RTE (Appendix F.1). Additional width, population-size, and perturbation-rank ablations are reported in Appendix E.

5.3 PEAK GPU MEMORY

Figure 4 compares peak memory during complete training updates. On Qwen3-0.6B-Base/DROP, AIM-ZO uses 0.242–0.254 GiB more than MeZO and less than AGZO and full-parameter SGD across the measured configurations. SGD runs out of memory at the largest tested tensor lengths or batch sizes, and AGZO at batch size 64. Measurement details, runtime comparisons, and caching ablations are in Appendix G.

6 CONCLUSION

We presented AIM-ZO, which maintains an activation-informed subspace across training and selects smaller shared-and-sampled active subspaces for forward-only ZO updates. Our analysis bounds gradient capture by the maintained subspace and characterizes when the active subspace improves directional alignment over full activation. Across five models and 11 tasks, AIM-ZO attains the best six-task average among fully evaluated ZO baselines on OPT-2.7B and improves over MeZO on all six OPT-30B tasks. Gains on Qwen3 are smaller, with modest peak-memory overhead.

The analysis assumes post-burn-in tracking conditions and Gaussian perturbations, leaving the implemented rank-one updates and changing active subspaces only partially characterized. Adapting the preset widths and tail sampling using loss feedback remains future work.

AI USE STATEMENT

We used generative AI tools to assist with the theoretical analysis and proofs, the implementation of experimental code, the interpretation and presentation of results, and the drafting and editing of the manuscript. The central method and main experiments were designed by the authors. AI tools helped identify gaps in the ablation studies and suggest additional controls. The assumptions were developed by the authors or adapted from prior work, with AI assistance in organizing and articulating them. AI tools also assisted with literature search and the conceptual design of Figure 1; the figure itself was drawn manually, and the other figures were generated from experimental data. No AI-generated datasets were used. The authors checked the AI-assisted theoretical arguments, code, references, and reported results and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

An anonymous repository for the implementation is provided at https://anonymous.4open.science/r/AIM-ZO-0155/. Section 3 specifies the method and update procedure. Section 4 states the analytical assumptions and results, with proofs and auxiliary derivations in Appendices A, A.6, and B. Appendix C documents the datasets, data splits, training budgets, hyperparameters, and evaluation protocol. Additional results and ablations are reported in Appendices H and E.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. First efficient convergence for streaming k-PCA: A global, gap-free, and near-optimal rate. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, pp. 487–492, 2017. doi: 10.1109/FOCS.2017.51.

Daniel Bienstock, Minchan Jeong, Apurv Shukla, and Se-Young Yun. Robust streaming PCA. In *Advances in Neural Information Processing Systems*, volume 35, pp. 4231–4243, 2022.

Yiming Chen, Yuan Zhang, Liyuan Cao, Kun Yuan, and Zaiwen Wen. Enhancing zeroth-order fine-tuning for language models with low-rank structures. In *International Conference on Learning Representations*, 2025.

Krzysztof M. Choromanski, Aldo Pacchiano, Jack Parker-Holder, Yunhao Tang, and Vikas Sindhwani. From complexity to simplicity: Adaptive ES-active subspaces for blackbox optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Xun Dong, Yibo Xu, Naigang Wang, Xin Li, Penghang Yin, and Zi Yang. ZO-Act: Efficient zeroth-order fine-tuning via one-shot activation-informed low-rank subspaces. *arXiv preprint arXiv:2607.01125*, 2026.

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

Wentao Guo, Jikai Long, Yimeng Zeng, Zirui Liu, Xinyu Yang, Yide Ran, Jacob R. Gardner, Osbert Bastani, Christopher De Sa, Xiaodong Yu, Beidi Chen, and Zhaozhuo Xu. Zeroth-order fine-tuning of LLMs with transferable static sparsity. In *International Conference on Learning Representations*, 2025.

Guy Gur-Ari, Daniel A. Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

De Huang, Jonathan Niles-Weed, and Rachel Ward. Streaming k-PCA: Efficient guarantees for Oja's algorithm, beyond rank-one updates. In *Proceedings of the 34th Conference on Learning Theory*, volume 134, pp. 2463–2498. PMLR, 2021.

Ajay Kumar Jaiswal, Yifan Wang, Lu Yin, Shiwei Liu, Runjin Chen, Jiawei Zhao, Ananth Grama, Yuandong Tian, and Zhangyang Wang. From low rank gradient subspace stabilization to low-rank weights: Observations, theories, and applications. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 26740–26756. PMLR, 2025.

Yicheng Lang, Changsheng Wang, Yihua Zhang, Mingyi Hong, Zheng Zhang, Wotao Yin, and Sijia Liu. Powering up zeroth-order training via subspace gradient orthogonalization. *arXiv preprint arXiv:2602.17155*, 2026.

Kaizhao Liang, Bo Liu, Lizhang Chen, and Qiang Liu. Memory-efficient LLM training with online subspace descent. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2054.

Wei Lin, Yining Jiang, Qingyu Song, Qiao Xiang, and Hong Xu. AGZO: Activation-guided zeroth-order optimization for LLM fine-tuning. In *Proceedings of the International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2601.17261.

Yong Liu, Zirui Zhu, Chaoyu Gong, Minhao Cheng, Cho-Jui Hsieh, and Yang You. Sparse meZO: Less parameters for better performance in zeroth-order LLM fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D. Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53038–53075, 2023.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017. doi: 10.1007/s10208-015-9296-2.

Sahar Rajabi, Nayeema Nonta, and Sirisha Rambhatla. SubTrack++: Gradient subspace tracking for scalable LLM training. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1060.

Joel A. Tropp. Freedman's inequality for matrix martingales. *Electronic Communications in Probability*, 16:262–270, 2011. URL https://tropp.caltech.edu/papers/Tro11-Freedmans-Inequality.pdf.

Shuo Wang, Ziyu Chen, and Ming Tang. CurvZO: Adaptive curvature-guided sparse zeroth-order optimization for efficient LLM fine-tuning. *arXiv preprint arXiv:2603.21725*, 2026.

Ziming Yu, Pan Zhou, Sike Wang, Jia Li, Mi Tian, and Hua Huang. Zeroth-order fine-tuning of LLMs in random subspaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4475–4485, 2025.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. GaLore: Memory-efficient LLM training by gradient low-rank projection. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 61121–61143. PMLR, 2024.

Yanjun Zhao, Sizhe Dang, Haishan Ye, Guang Dai, Yi Qian, and Ivor W. Tsang. Second-order fine-tuning without pain for LLMs: A hessian informed zeroth-order optimizer. In *International Conference on Learning Representations*, 2025.

Appendix Contents

Key Notation 13
A. Maintained-Subspace Tracking and Gradient Capture 14
B. Energy Capture and Directional Alignment in Active Subspaces 18
C. Experimental Setup and Evaluation Protocol 20
D. Supplementary Experiments for Section 4 21
E. Active-Subspace Ablations 26
F. Space Quality and One-Sided Estimation 28
G. Runtime and Peak Memory 30
H. Additional Fine-Tuning Results 31
I. Additional Related Work 33

Key Notation

We summarize the main symbols used in the analysis; other quantities are defined where they first appear. Each basis matrix represents the subspace spanned by its columns, with $P_Q = QQ^\top$ denoting the corresponding projector. The unindexed Q denotes the maintained basis in Section 4.2 and Appendix B.

Symbol	Description
K, k, h, s	Maintained, active, prefix, and sampled-tail widths; $k = h + s \le K$
Q_t, Q_S	Maintained basis at step t and active basis selected from a fixed maintained basis
$U_t^\star, V_t^\star$	Shared gradient basis and population activation target basis
$r, \mathcal{I}_t, T_{\mathrm{w}}$	Shared-gradient rank, sliding window, and window length
S_t, M_t	Uncentered activation Gram observation $H_t^\top H_t/n_t$ and its conditional mean
$\varepsilon_{\mathrm{sub},t}$	Bound on the shared-gradient residual
$\varepsilon_{\mathrm{act},t}$	Bound on the activation-alignment residual
$\varepsilon_{\mathrm{trk},t}$	Spectral distance between maintained and target projectors
Φ_t	Squared chordal tracking error
$\omega_t, \bar{\omega}$	Target-subspace drift and its upper bound
γ, η_q	Eigengap lower bound and Oja step size
C_Q	Gradient capture $\|GP_Q\|_F^2/\|G\|_F^2$
τ_h, ρ	Prefix share of maintained energy and tail inclusion probability
θ	Fixed shared fraction in the active-width comparison; $h = \theta k$
$\beta_D, \mathcal{L}_{K,k}$	Gaussian direction factor and shared-and-sampled alignment lower bound
R, D	Gaussian coefficient matrix in the active-width analysis and its dimension $D = pk$
$\widehat{G}_{\mathrm{OP}}, \widehat{G}_{\mathrm{TP}}$	One-sided RLOO and two-sided gradient estimators

A Maintained-Subspace Tracking and Gradient Capture

A.1 Conditions and statement

Let $t = 0$ mark the end of burn-in and $T \geq 1$ the number of subsequent Oja updates. Define $P_t = Q_t Q_t^\top$, $P_{\star,t} = V_t^\star V_t^{\star\top}$, and

$$\Phi_t = K - \mathrm{tr}(P_{\star,t} P_t) = \tfrac{1}{2}\|P_t - P_{\star,t}\|_F^2, \qquad \varepsilon_{\mathrm{trk},t}^2 \leq \Phi_t \leq K.$$

The history $\mathcal{F}_{t-1}$ precedes the current observation S_t; $Q_{t-1}, M_t, P_{\star,t}$ are measurable with respect to this history. We impose the following conditions for a fixed horizon T.

1. There is an $\mathcal{F}_0$-measurable event $\mathcal{E}_0 = \{\Phi_0 \leq \varepsilon_{\mathrm{burn}}^2\}$ with probability at least $1 - \delta_{\mathrm{init}}$.
2. $S_t \succeq 0$, $M_t = \mathbb{E}[S_t \mid \mathcal{F}_{t-1}]$, and $\lambda_K(M_t) - \lambda_{K+1}(M_t) \geq \gamma > 0$.
3. Almost surely $\|S_t\|_2 \leq \Lambda$ and $\mathbb{E}[\|S_t\|_2^2 \mid \mathcal{F}_{t-1}] \leq \Sigma^2$.
4. $\omega_t = \|P_{\star,t} - P_{\star,t-1}\|_F/\sqrt{2} \leq \bar{\omega}$.

Initialization. Under the stationary sampling and two-phase step-size conditions of Huang et al. (2021), Gaussian-initialized Oja attains a prescribed projector accuracy with high probability. In particular, $\|P_{Q_0} - P_{V_0^\star}\|_2 \leq \varepsilon_{\mathrm{burn}}/\sqrt{K}$ implies $\Phi_0 \leq \varepsilon_{\mathrm{burn}}^2$, since $\Phi_0 \leq K\|P_{Q_0} - P_{V_0^\star}\|_2^2$. For the evolving activation sequence, we assume this entrance condition.

Take $\eta_q > 0$, $0 \leq \delta_{\mathrm{init}} < 1$, and $0 < \Sigma \leq \Lambda$. Set $a = \gamma/2$, $\kappa = 1 - \eta_q a$ and, for $0 < \delta < 1$,

$$b_T(\delta) = 2\sqrt{2}K\sqrt{\frac{\eta_q \Sigma^2}{a}\log\frac{T}{\delta}} + \frac{8}{3}K\eta_q\Lambda\log\frac{T}{\delta}.$$

The explicit sufficient step-size and entrance budgets are

$$\eta_q\Lambda \leq \tfrac{1}{8}, \qquad \varepsilon_{\mathrm{burn}}^2 \leq \tfrac{1}{16}, \qquad \bar{\omega} \leq \tfrac{\eta_q a}{4}, \qquad \frac{16K\eta_q\Lambda^2}{a} + b_T(\delta) \leq \tfrac{1}{8}. \tag{12}$$

Since $0 \preceq M_t \preceq \Lambda I$, we have $\gamma \leq \Lambda$, so they imply $0 < \eta_q a \leq 1/16$. Define

$$B_{\max} = \frac{\bar{\omega}^2}{\eta_q^2 a^2} + \frac{16K\eta_q\Lambda^2}{a}, \qquad r_{\mathrm{safe}}^2 = \max\{\varepsilon_{\mathrm{burn}}^2, B_{\max}\} + b_T(\delta).$$

We prove that, with probability at least $1 - \delta_{\mathrm{init}} - \delta$, $\Phi_t \leq r_{\mathrm{safe}}^2$ simultaneously for $t \leq T$. The explicit expected bound underlying Theorem 1 is

$$\mathbb{E}\varepsilon_{\mathrm{trk},t}^2 \leq \kappa^t\varepsilon_{\mathrm{burn}}^2 + (1-\kappa^t)\left[\frac{4\bar{\omega}^2}{\eta_q^2\gamma^2} + \frac{32K\eta_q\Sigma^2}{\gamma}\right] + K(\delta + \delta_{\mathrm{init}}). \tag{13}$$

Using $\kappa^t \leq e^{-\eta_q\gamma t/2}$ and $1 - \kappa^t \leq 1$ gives the simplified bound in the main text.

A.2 One-step expansion

Lemma 1 (Second-order expansion of one QR–Oja step). *Let $P = QQ^\top$, $Q^\top Q = I_K$, let $S = S^\top$, and let P^+ project onto $\mathrm{col}((I + \eta_q S)Q)$. If $\eta_q\|S\|_2 \leq 1/8$, then*

$$P^+ = P + \eta_q[(I-P)SP + PS(I-P)] + R,$$
$$\|R\|_2 \leq 16\eta_q^2\|S\|_2^2. \tag{14}$$

Proof. Set

$$E = \eta_q S, \qquad e = \|E\|_2 \leq \tfrac{1}{8}, \qquad \widetilde{Q} = (I+E)Q. \tag{15}$$

The update changes the Gram matrix of the columns of Q. Define its first- and second-order parts by

$$A_E = Q^\top EQ, \qquad B_E = Q^\top E^2 Q, \qquad \Delta_E = 2A_E + B_E. \tag{16}$$

Because $Q^\top Q = I_K$ and $E = E^\top$,

$$\widetilde{Q}^\top\widetilde{Q} = I + \Delta_E. \tag{17}$$

Moreover, $\|A_E\|_2 \le e$, $\|B_E\|_2 \le e^2$, and hence

$$\|\Delta_E\|_2 \le 2e + e^2 \le \frac{17}{64} < 1. \tag{18}$$

The exact resolvent identity $(I+\Delta_E)^{-1} = I - \Delta_E + \Delta_E^2(I+\Delta_E)^{-1}$ therefore gives

$$(I+\Delta_E)^{-1} = I - 2A_E + F_E, \qquad F_E = -B_E + \Delta_E^2(I+\Delta_E)^{-1}. \tag{19}$$

Since $\|(I+\Delta_E)^{-1}\|_2 \le (1-\|\Delta_E\|_2)^{-1}$,

$$\|F_E\|_2 \le e^2 + \frac{(2e+e^2)^2}{1-2e-e^2} \le 8e^2. \tag{20}$$

QR does not change the column space of $\widetilde{Q}$, so

$$P^+ = \widetilde{Q}(\widetilde{Q}^\top\widetilde{Q})^{-1}\widetilde{Q}^\top. \tag{21}$$

Substitution of the inverse expansion, together with $QA_EQ^\top = PEP$, yields

$$P^+ = P + EP + PE - 2PEP + R_E = P + (I-P)EP + PE(I-P) + R_E, \tag{22}$$

where the terms of order at least two are collected exactly as

$$\begin{aligned} R_E = {} & EPE - 2EPEP - 2PEPE - 2EPEPE \\ & + (I+E)QF_EQ^\top(I+E). \end{aligned} \tag{23}$$

Using $\|P\|_2 = \|Q\|_2 = 1$ and Equation 20,

$$\begin{aligned} \|R_E\|_2 &\le [1+2+2+2e+8(1+e)^2]e^2 \\ &\le 16e^2. \end{aligned} \tag{24}$$

Finally, substituting $E = \eta_q S$ and renaming R_E as R proves Equation 14. □

A.3 Global inequality and local contraction

Fix t and abbreviate $P = P_{t-1}$. Put $u_t = K - \operatorname{tr}(P_{\star,t}P)$. In the orthonormal eigenbasis of M_t, write

$$P = \begin{bmatrix} X & Y \\ Y^\top & Z \end{bmatrix}, \qquad M_t = \begin{bmatrix} M_t^{\|} & 0 \\ 0 & M_t^{\perp} \end{bmatrix}.$$

Since $P^2 = P$, $YY^\top = X - X^2$. Thus

$$\begin{aligned} A_t &:= \operatorname{tr}(P_{\star,t}(I-P)M_tP) \\ &= \operatorname{tr}(M_t^{\|}YY^\top) - \operatorname{tr}(M_t^{\perp}Y^\top Y) \ge \gamma\|Y\|_F^2. \end{aligned}$$

If θ_i are the principal angles, then $u_t = \sum_i \sin^2\theta_i$ and $\|Y\|_F^2 = \sum_i \sin^2\theta_i\cos^2\theta_i \ge u_t(1-u_t)_+$. Define the mean-zero increment

$$\zeta_t = -2\eta_q \operatorname{tr}(P_{\star,t}(I-P)(S_t - M_t)P).$$

The expansion lemma and $|\operatorname{tr}(P_{\star,t}R)| \le K\|R\|_2$ give globally

$$\Phi_t \le u_t - 2\eta_q\gamma u_t(1-u_t)_+ + 16K\eta_q^2\|S_t\|_2^2 + \zeta_t.$$

The projector distance satisfies the triangle inequality, so $\sqrt{u_t} \le \sqrt{\Phi_{t-1}} + \omega_t$. When $u_t \le 1/2$, the preceding inequality and

$$(1-2\eta_q a)(x+y)^2 \le (1-\eta_q a)x^2 + \frac{y^2}{\eta_q a} \quad (x,y \ge 0)$$

therefore imply

$$\Phi_t \le \kappa\Phi_{t-1} + \frac{\bar{\omega}^2}{\eta_q a} + 16K\eta_q^2\|S_t\|_2^2 + \zeta_t. \tag{25}$$

The following induction verifies $u_t \le 1/2$ throughout the finite trajectory on a high-probability event.

A.4 Concentration and invariant-region induction

The matrix $PP_{\star,t}(I-P)$ is predictable, has rank at most K and spectral norm at most one. Consequently

$$\mathbb{E}[\zeta_t \mid \mathcal{F}_{t-1}] = 0, \quad |\zeta_t| \le 4K\eta_q\Lambda, \quad \mathbb{E}[\zeta_t^2 \mid \mathcal{F}_{t-1}] \le 4K^2\eta_q^2\Sigma^2.$$

The variance bound follows by centering the scalar trace, whose second moment is at most $K^2\mathbb{E}[\|S_t\|_2^2 \mid \mathcal{F}_{t-1}]$. For each fixed endpoint t, apply the scalar Freedman inequality (Tropp, 2011, Theorem 1.1) to $\sum_{s=1}^{t}\kappa^{t-s}\zeta_s$. For each fixed t, the weights κ^{t-s} are deterministic; the partial sums in s form a martingale. Its predictable variance is at most $4K^2\eta_q\Sigma^2/a$ and its increments are bounded by $4K\eta_q\Lambda$. The bound $\sqrt{2vx}+2bx/3$ for variance v and increment bound b, followed by a union bound over $t \le T$, gives an event $\mathcal{E}$ such that

$$\Pr(\mathcal{E} \mid \mathcal{E}_0) \ge 1-\delta, \qquad \sum_{s=1}^{t}\kappa^{t-s}\zeta_s \le b_T(\delta) \quad \text{for every } t \le T.$$

Since $\mathcal{E}_0 \in \mathcal{F}_0$, the same bound holds conditionally on $\mathcal{E}_0$. Write $v = \bar{\omega}^2/(\eta_q^2 a^2) \le 1/16$ and $w = 16K\eta_q\Lambda^2/a$,

$$r_{\text{safe}}^2 \le \max\{\varepsilon_{\text{burn}}^2, v\} + w + b_T(\delta) \le \tfrac{1}{16} + \tfrac{1}{8} = \tfrac{3}{16} < \tfrac{1}{4}.$$

Thus the budgets imply $r_{\text{safe}} \le 1/2$, $\bar{\omega} \le 1/64$, and $r_{\text{safe}} + \bar{\omega} < 1/\sqrt{2}$. On $\mathcal{E}_0 \cap \mathcal{E}$, suppose inductively that $\Phi_j \le r_{\text{safe}}^2$ for all $j < t$. Then $u_s \le (r_{\text{safe}} + \bar{\omega})^2 < 1/2$ for every $s \le t$. Iterating Equation 25 and using the pathwise bound $\|S_s\|_2 \le \Lambda$ yields

$$\Phi_t \le \kappa^t\Phi_0 + (1-\kappa^t)B_{\max} + b_T(\delta) \le r_{\text{safe}}^2.$$

The base case is $\mathcal{E}_0$, completing the induction.

A.5 Expected tracking error

Condition on $\mathcal{E}_0 \in \mathcal{F}_0$, so the noise increments retain their conditional mean-zero property. Write $\mathbb{E}_0, \Pr_0$ for expectation and probability under this conditioning, and define $J_t = \mathbf{1}\{\Phi_j \le r_{\text{safe}}^2 \text{ for every } 0 \le j \le t\}$ and $U_t = \Phi_t J_t$. Because J_{t-1} is predictable, $J_t \le J_{t-1}$, and $\Phi_t \ge 0$, we have $U_t \le J_{t-1}\Phi_t$. On $\{J_{t-1}=1\}$ the local inequality applies; on its complement both U_t and U_{t-1} vanish. Predictability and mean-zero noise give

$$\mathbb{E}_0[U_t \mid \mathcal{F}_{t-1}] \le \kappa U_{t-1} + \frac{\bar{\omega}^2}{\eta_q a} + 16K\eta_q^2\Sigma^2.$$

Iteration and $\Pr(J_t = 0 \mid \mathcal{E}_0) \le \delta$ imply

$$\mathbb{E}[\Phi_t \mid \mathcal{E}_0] \le \kappa^t\varepsilon_{\text{burn}}^2 + (1-\kappa^t)\left[\frac{\bar{\omega}^2}{\eta_q^2 a^2} + \frac{16K\eta_q\Sigma^2}{a}\right] + K\delta.$$

Finally $\Phi_t \le K$ on burn-in failure. Substitute $a = \gamma/2$ and $\varepsilon_{\text{trk},t}^2 \le \Phi_t$ to obtain Equation 8.

A.6 Proof of maintained-subspace capture

All projectors in this section are orthogonal. Fix a time t. Let $Q_t \in \mathbb{R}^{d\times K}$ be the maintained orthonormal basis, $V_t^\star \in \mathbb{R}^{d\times K}$ the top-K eigenbasis of the population activation covariance M_t, and $U_t^\star \in \mathbb{R}^{d\times r}$ the shared rank-r right gradient basis for the window $\mathcal{I}_t$. These three bases are held fixed for every gradient in the window. The preceding tracking theorem controls $\varepsilon_{\text{trk},t} = \|P_{Q_t} - P_{V_t^\star}\|_2$; here we relate this error to gradient capture. The target $V_t^\star$ is defined by the conditional covariance M_t, not by a covariance averaged over future observations. For any nonzero matrix G, write

$$G(I-P_{Q_t}) = G(I-P_{U_t^\star})(I-P_{V_t^\star}) + GP_{U_t^\star}(I-P_{V_t^\star}) + G(P_{V_t^\star} - P_{Q_t}).$$

The first term has norm at most $\|G(I-P_{U_t^\star})\|_F$; the last has norm at most $\|G\|_F\varepsilon_{\text{trk},t}$. Divide by $\|G\|_F$ and apply the triangle inequality in the product Hilbert space over $\mathcal{I}_t$, with weight $1/T_{\text{w}}$. Assumptions 1–2 give

$$\left[\frac{1}{T_{\text{w}}}\sum_{\tau\in\mathcal{I}_t}\frac{\|G_\tau(I-P_{Q_t})\|_F^2}{\|G_\tau\|_F^2}\right]^{1/2} \le \varepsilon_{\text{sub},t} + \varepsilon_{\text{act},t} + \varepsilon_{\text{trk},t}.$$

Orthogonality gives $\|G\|_F^2 = \|GP_{Q_t}\|_F^2 + \|G(I - P_{Q_t})\|_F^2$, proving Proposition 1. For an individual gradient, the same argument holds with its individual structural residuals. A window-average assumption alone does not imply the corresponding pointwise bound.

B Energy Capture and Directional Alignment in Active Subspaces

B.1 Fixed-subspace identity

Fix $G \neq 0$ and an ordered orthonormal maintained basis $Q \in \mathbb{R}^{d\times K}$ with $C_Q > 0$. As in Section 4.2, let $0 < \theta < 1$, $1 \leq k \leq K$, and $h = \theta k \in \mathbb{Z}$. The active basis Q_S contains the first h columns of Q and $k-h$ columns sampled uniformly without replacement from the remaining $K-h$ columns. Condition on the active basis Q_S. Let $B_S = GQ_S$ and $D = pk$. The fixed-subspace cosine identity of Lin et al. (2026, Theorem 5.4), applied to Q_S, gives

$$\mathbb{E}_R \cos(\langle G, RQ_S^\top\rangle_F RQ_S^\top, G) = \beta_D \frac{\|B_S\|_F}{\|G\|_F}.$$

Indeed, the cosine is $|\langle B_S, R\rangle_F|/(\|G\|_F\|R\|_F)$. After aligning the first coefficient axis with B_S, the expected absolute first coordinate of a uniform unit vector is $\Gamma(D/2)/[\sqrt{\pi}\Gamma((D+1)/2)]$. We use the convention $\cos(0, G) = 0$, including when $B_S = 0$.

B.2 Proof of the estimation-error decomposition

Conditional on Q_S, write $B_S = GQ_S$ and $D = pk$. The Gaussian subspace moments of Yu et al. (2025, Theorem 2) give $\mathbb{E}_R[\langle B_S, R\rangle_F R] = B_S$ and $\mathbb{E}_R[\langle B_S, R\rangle_F^2\|R\|_F^2] = (D+2)\|B_S\|_F^2$. Consequently, the conditional mean of $\widehat{G}_{k,0}$ is GP_{Q_S}, and its conditional variance is $(D+1)\|B_S\|_F^2$. The decomposition

$$\widehat{G}_{k,0} - G = (\widehat{G}_{k,0} - GP_{Q_S}) - G(I - P_{Q_S})$$

has orthogonal components in the Frobenius inner product. Hence

$$\mathbb{E}_R\|\widehat{G}_{k,0} - G\|_F^2 = (pk+1)\|GP_{Q_S}\|_F^2 + \|G(I-P_{Q_S})\|_F^2.$$

This proves Equation 10.

B.3 Proof of Proposition 2

Write $Q = [q_1, \ldots, q_K]$ and $e_j = \|Gq_j\|_2^2$. Each non-shared column has inclusion probability $\rho = (k-h)/(K-h)$. Since the selected columns are orthonormal,

$$\begin{aligned}\mathbb{E}_S\|GQ_S\|_F^2 &= \sum_{j=1}^{h} e_j + \rho \sum_{j=h+1}^{K} e_j \\ &= \|GP_Q\|_F^2[\tau_h + \rho(1-\tau_h)].\end{aligned}$$

Dividing by $\|G\|_F^2 > 0$ proves the capture identity in Equation 11. Moreover, $A_{K,k} = 1 - (K-k)(1-\tau_h)/(K-h)$, so $\tau_h \geq 1-\varepsilon$ implies $A_{K,k} \geq 1 - \varepsilon(K-k)/(K-h)$.

Under the Gaussian model, define

$$\mathcal{L}_{K,k} := \beta_{pk}\sqrt{C_Q}[\tau_h + \rho(1-\tau_h)].$$

Let $X = \|GQ_S\|_F$ and $b = \|GP_Q\|_F > 0$. Since $0 \leq X \leq b$, $X \geq X^2/b$. The fixed-subspace identity therefore gives

$$\mathbb{E}_{S,R}\cos(\widehat{G}_{k,0}, G) = \frac{\beta_{pk}}{\|G\|_F}\mathbb{E}_S X \geq \beta_{pk}\sqrt{C_Q}[\tau_h + \rho(1-\tau_h)],$$

which establishes $J_{K,k} \geq \mathcal{L}_{K,k}$. At $k = K$, Q_S spans the maintained subspace, and the fixed-subspace identity gives $J_{K,K} = \beta_{pK}\sqrt{C_Q}$. Since $C_Q > 0$, the sufficient condition in Proposition 2 makes the lower bound on $J_{K,k}$ strictly exceed this exact full-activation value.

B.4 Supplementary directional-alignment bounds

For comparison, concavity of the square root yields the upper bound

$$\mathbb{E}_{S,R}\cos(\widehat{G}_{k,0}, G) \leq \beta_{pk}\sqrt{C_Q[\tau_h + \rho(1-\tau_h)]}.$$

B.5 Separating maintained and active widths

Substituting $h = \theta k$ into the exact bound gives

$$\mathcal{L}_{K,k} = \beta_{pk}\sqrt{C_Q}\left[\tau_{\theta k} + \frac{(1-\theta)k}{K-\theta k}(1-\tau_{\theta k})\right].$$

The gamma-ratio expansion $\beta_{pk} = \sqrt{2/(\pi pk)}[1 + O((pk)^{-1})]$ gives the decomposition

$$\mathcal{L}_{K,k} = \sqrt{\frac{2C_Q}{\pi p}}\left[\frac{\tau_{\theta k}}{\sqrt{k}} + \frac{(1-\theta)(1-\tau_{\theta k})\sqrt{k}}{K-\theta k}\right]\left[1 + O((pk)^{-1})\right]. \tag{26}$$

For fixed θ, the continuous extension of the inclusion probability has derivatives

$$\left.\frac{\partial \rho}{\partial K}\right|_k = -\frac{(1-\theta)k}{(K-\theta k)^2} < 0, \qquad \left.\frac{\partial \rho}{\partial k}\right|_K = \frac{(1-\theta)K}{(K-\theta k)^2} > 0.$$

The same monotonicities hold on admissible integer widths.

At $k = K$, every column participates, $\rho = 1$, and the fixed-subspace identity gives $\mathbb{E}\cos(\widehat{G}_{K,0}, G) = \beta_{pK}\sqrt{C_Q}$, recovering Lin et al. (2026, Theorem 5.4). For a fixed-prefix selection $k = h$, the identity instead gives $\beta_{pk}\sqrt{C_Q\tau_h}$.

B.6 Additional comparison under shared-energy concentration

Since the shared subspace is contained in every active subspace, which is contained in the maintained subspace, orthogonal projection gives

$$\|GP_{Q_h}\|_F^2 \le \|GP_{Q_S}\|_F^2 \le \|GP_Q\|_F^2.$$

Since $\|GQ_S\|_F = \|GP_{Q_S}\|_F$, taking square roots, dividing by $\|G\|_F$, and applying the fixed-subspace identity yields

$$\beta_{pk}\sqrt{\tau_{\theta k}C_Q} \le \mathbb{E}_{S,R}\cos(\widehat{G}_{k,0}, G) \le \beta_{pk}\sqrt{C_Q}.$$

Now suppose $\tau_{\theta k_0} \ge 1 - \varepsilon$, with $0 \le \varepsilon < 1$. Applying the lower bound at k_0 and the upper bound at k_1 gives

$$\mathbb{E}\cos(\widehat{G}_{k_0,0}, G) \ge \beta_{pk_0}\sqrt{(1-\varepsilon)C_Q},$$
$$\mathbb{E}\cos(\widehat{G}_{k_1,0}, G) \le \beta_{pk_1}\sqrt{C_Q}.$$

Since $C_Q > 0$, if $\beta_{pk_1}/\beta_{pk_0} < \sqrt{1-\varepsilon}$, then $\mathbb{E}\cos(\widehat{G}_{k_0,0}, G) > \mathbb{E}\cos(\widehat{G}_{k_1,0}, G)$.

Table 3: AIM-ZO learning rates and initial perturbation scales.

Model	Learning rate η	Initial scale ϵ_0
OPT-2.7B	1.5×10^{-3}	8×10^{-5}
OPT-13B	1×10^{-3}	8×10^{-5}
OPT-30B	5×10^{-4}	6×10^{-5}
Qwen3-0.6B-Base	1×10^{-3}	8×10^{-5}
Qwen3-8B-Base	1×10^{-3}	8×10^{-5}

C Experimental Setup and Evaluation Protocol

Models, tasks, and metrics. The main comparisons use OPT-2.7B, OPT-13B, OPT-30B, Qwen3-0.6B-Base, and Qwen3-8B-Base. The 11 tasks are RTE, BoolQ, SST-2, WiC, WSC, COPA, CB, MultiRC, ReCoRD, SQuAD, and DROP; OPT-2.7B is evaluated on all 11. The main OPT tables report RTE, BoolQ, SST-2, WiC, WSC, and SQuAD; the Qwen tables replace WSC with MultiRC. Qwen3-4B is used in supplementary experiments. Baselines are MeZO, LoZO, AGZO, HiZOO, CurvZO, and ZO-Muon, with zero-shot scores as a non-training reference. We report accuracy for classification tasks and F1 for extractive question answering. Additional metrics and task results appear in Appendix H.

Data split. Following the MeZO sampling protocol (Malladi et al., 2023), training and development subsets are sampled without overlap from each task's official training split. For the six-task main comparisons, the subsets contain 1,000 training and 500 development examples, except WSC, which uses 454/100. The training minibatch size is 16. The additional OPT-2.7B CB and COPA comparisons use 150/100 and 300/100, respectively. Methods compared on a task use the same sampled splits. The OPT-2.7B and OPT-13B comparisons use five evenly spaced development checkpoints; Qwen3-0.6B-Base runs evaluate every 500 training steps. The OPT-13B MeZO, CurvZO, and LoZO SQuAD runs select checkpoints by generated-answer development F1. Other runs follow their recorded evaluation schedules. The selected checkpoint is evaluated on a separate official validation set. In the OPT-2.7B eleven-task comparison, BoolQ, ReCoRD, SQuAD, and DROP use fixed 1,000-example official-validation subsets; the other tasks use their full official validation sets.

Training budget. Training budgets are matched by perturbed and unperturbed forward evaluations, with approximately 40,000 training forwards per run; validation and checkpoint I/O are excluded from this budget. AIM-ZO uses 15 perturbed evaluations and one unperturbed centre evaluation per update, giving 2,500 updates. MeZO uses two perturbed evaluations per update, giving 20,000 updates. Update counts for the other baselines account for their respective evaluations per update.

Training hardware. OPT-30B and Qwen3-8B-Base experiments use 140 GiB NVIDIA H200 GPUs. The other main experiments run on NVIDIA RTX 4090 GPUs with 24 or 48 GiB of memory. Training ablations run on RTX 4090 GPUs.

Method settings. AIM-ZO uses BF16 and updates all model parameters. Each trainable layer maintains $K = 128$ directions and activates $k = 64$ per perturbation: $h = 48$ shared directions and 16 directions sampled independently from the maintained tail. The maintained subspace is updated at every training step with Oja step size $\eta_q = 0.3$. Each perturbation uses the rank-one construction and layer-wise Frobenius normalization in Section 3.3. The learning rate is constant and weight decay is zero. Over $T = 2{,}500$ updates, the perturbation scale follows $\epsilon_t = \epsilon_0[\frac{1}{4} + \frac{3}{8}(1 + \cos(\pi t/T))]$, from the initial values in Table 3 to $\epsilon_0/4$. Table 4 summarises the baseline hyperparameter search ranges.

Reporting. Tables report mean scores in percent and sample standard deviations when available. Six-task averages are unweighted and are reported only when all six task scores are available. Additional evaluation records are given in Appendix H.

Ablation protocol. Unless stated otherwise, training ablations follow the corresponding main comparison's data split, batch size, forward-evaluation budget, checkpoint-selection rule, and final evaluation set. Changed components and seed counts are specified with each ablation.

Table 4: Baseline hyperparameter search ranges and calls per update for OPT-13B and Qwen3-0.6B, with OPT-2.7B MeZO also included.

Method	Calls/update	Learning rate	Perturbation scale
MeZO	2	$(1\text{–}5) \times 10^{-7}$	10^{-3}
CurvZO	2	$(1\text{–}8) \times 10^{-7}$	5×10^{-4}–10^{-3}
AGZO	3	5×10^{-7}	10^{-4}–10^{-3}
HiZOO	3	5×10^{-7}	5×10^{-4}–10^{-3}
LoZO	2	10^{-7}–10^{-6}	10^{-3}
ZO-Muon	5	10^{-2}	10^{-3}

Peak-memory measurement. Figure 4 reports peak allocated GPU memory from complete parameter-update steps, varying batch size and padded tensor length on Qwen3-0.6B-Base/DROP. The measurement uses BF16 weights, SDPA attention, and a 24 GiB RTX 4090, with no gradient checkpointing or key-value cache. Each configuration runs ten updates in a separate process; OOM denotes an observed failure. Further implementation details are in Appendix G.

D Supplementary Experiments for Section 4

We provide additional measurements for the three analyses in Section 4: gradient capture by the maintained subspace, the relation between maintained and active widths, and one-sided estimation.

D.1 Gradient structure and maintained-subspace capture

We evaluate whether gradients concentrate in a shared subspace, whether activations identify that subspace, and how well Oja maintains it during training. The first two measurements use both fixed checkpoints and consecutive training checkpoints; the third compares saved Oja subspaces with current-batch references.

Fixed-checkpoint protocol. We evaluate RTE checkpoints of Qwen3-4B at steps 500 and 2500, OPT-2.7B at step 2000, and OPT-30B at step 1000. Three disjoint sets contain 128 activation-calibration batches, 16 gradient-fit batches, and 16 gradient-test batches. They construct the top-K activation subspace, fit the shared rank-r right gradient subspace, and evaluate residuals and capture, respectively. Early, middle, and late refer to `q_proj` layers 0, 18, and 35 in Qwen3-4B; 0, 16, and 31 in OPT-2.7B; and 0, 24, and 47 in OPT-30B.

The frozen-checkpoint rank scan measures the squared structural residual by layer at $r \in \{1, 8, 32, 64, 128\}$. The width scan fixes the fitted shared space at $r = 8$ and varies $K \in \{8, 16, 32, 64, 128\}$, reporting the arithmetic mean of the three layer-wise squared alignment residuals. Residuals are normalized by each test gradient's squared Frobenius norm before averaging over batches; they are held-out counterparts of the quantities in Assumptions 1–2. Specifically, let $\widehat{U}_r$ be the rank-r basis fitted on the gradient-fit batches and $\widehat{V}_K$ the width-K activation basis fitted on the calibration batches. For the held-out set $\mathcal{B}_{\text{test}}$ of size $B_{\text{test}} = 16$, we report

$$e^2_{\text{sub}}(r) = \frac{1}{B_{\text{test}}} \sum_{b \in \mathcal{B}_{\text{test}}} \frac{\|G_b(I - P_{\widehat{U}_r})\|_F^2}{\|G_b\|_F^2}, \qquad e^2_{\text{act}}(r, K) = \frac{1}{B_{\text{test}}} \sum_{b \in \mathcal{B}_{\text{test}}} \frac{\|G_b P_{\widehat{U}_r}(I - P_{\widehat{V}_K})\|_F^2}{\|G_b\|_F^2}.$$

The hats distinguish fitted subspaces from the theoretical window optimizer $U_t^\star$ and population target $V_t^\star$. The frozen-checkpoint capture comparison evaluates the calibration-based activation target and the checkpoint Oja subspace at $K = 128$.

Moving training trajectory. Figure 2 uses 16 consecutive pre-update checkpoints: steps 500–515 for Qwen3-4B and 2000–2015 for OPT-2.7B, on RTE (seed 42). At each distinct parameter value W_t, we compute

$$\bar{G}_t = \nabla_W \frac{1}{1000} \sum_{x \in \mathcal{D}_{\text{train}}} \ell(x; W_t).$$

BF16 checkpoint weights are promoted to FP32. Autograd gradients are accumulated over four-example microbatches with sample-count weighting, retaining the three selected `q_proj` matrices. The first eight gradients fit a shared right space; the last eight evaluate its normalized capture. This temporal split tests transfer across training steps without fitting on the evaluation gradients. Unlike the minibatch gradients in Assumption 1, $\bar{G}_t$ is the exact mean over the finite training set.

Table 5: Temporal held-out shared-space capture. The rank-r space is fitted on eight steps and evaluated on the next eight.

Model	Layer	$r = 8$	$r = 32$	$r = 64$	$r = 128$
Qwen3-4B	Early	.622	.820	.879	.923
	Middle	.911	.964	.980	.990
	Late	.992	.994	.995	.995
OPT-2.7B	Early	.443	.678	.784	.868
	Middle	.954	.980	.987	.992
	Late	.923	.966	.981	.991

Middle and late layers exhibit concentrated shared mean-gradient structure, whereas early layers require a larger rank.

For activation alignment, a disjoint 500-example train-dev split provides activation observations at the first eight checkpoints. The fitted activation target and the rank-8 shared gradient space are evaluated using the last eight full-training-set gradients, with the same normalized residual formula as above. The fitted target provides an empirical window reference for $V_t^\star$.

Table 6: Temporal held-out activation alignment residual e^2_{act} at fixed shared rank $r = 8$.

Model	Layer	$K = 8$	$K = 16$	$K = 32$	$K = 64$	$K = 128$
Qwen3-4B	Early	.2710	.1242	.0523	.0491	.0462
	Middle	.1120	.0909	.0636	.0469	.0370
	Late	.2101	.1572	.0636	.0344	.0225
OPT-2.7B	Early	.2439	.1550	.0526	.0295	.0209
	Middle	.0394	.0223	.0142	.0064	.0046
	Late	.2557	.2252	.1571	.1398	.1184

The alignment residual decreases with K, although its magnitude differs substantially by layer; the late OPT layer retains a residual of .1184 at $K = 128$.

Saved Oja subspaces and current-batch references. The saved Oja subspace is evaluated against $\bar{G}_t$ at every checkpoint. An exact-SVD top-128 space from the current training batch provides a separate reference (Table 7).

Table 7: Capture of the finite-training-set mean gradient over 16 steps at $K = 128$; the current-batch reference uses exact SVD.

Model	Layer	Oja mean	Oja range	Current SVD mean
Qwen3-4B	Early	.522	.476–.628	.695
	Middle	.862	.791–.951	.890
	Late	.878	.699–.957	.926
OPT-2.7B	Early	.635	.587–.675	.717
	Middle	.951	.932–.966	.964
	Late	.657	.537–.719	.770

The exact current-batch SVD has higher capture in all six layers.

A second control reconstructs current-batch spaces using the actual training batch and five-step subspace iteration with Rayleigh–Ritz extraction (PI-5). At each checkpoint, both banks are evaluated

against FP32 autograd gradients of 16 independent 16-example batches from the disjoint 500-example train-dev split. Four-example microbatch accumulation preserves each batch mean. Each layer has 256 paired batch/checkpoint comparisons. Active width 64 uses the average over 15 selections of the first 48 columns plus 16 random tail columns. The saved Qwen trajectory uses $\eta_q = .3$.

Table 8: Oja minus PI-5 current-batch capture in percentage points (pp), and Oja win rates. Pairs share checkpoints and are not independent training runs.

		Wide subspace ($K = 128$)		Active space ($k = 64$)	
Model	Layer	Difference (pp)	Win rate	Difference (pp)	Win rate
Qwen3-4B	Early	-11.06	3.9%	-10.31	11.7%
	Middle	+4.71	98.8%	+4.06	94.5%
	Late	+1.02	49.6%	+2.38	64.5%
OPT-2.7B	Early	+8.95	96.1%	+6.29	83.2%
	Middle	+1.54	80.9%	+2.19	83.6%
	Late	+4.82	61.3%	+4.68	63.3%

Oja improves mean independent-batch capture in five of the six measured layers, with the early Qwen layer as the exception.

Oja maintenance takes 0.124 s versus 0.229 s for current-batch PI basis construction, a 45.9% reduction; total step times are 0.774 s and 0.760 s, respectively.

Stationary Oja replay. At frozen checkpoints, Oja is replayed over the calibration activation stream from three random initializations. Table 9 reports final normalized projector error and held-out gradient capture. The preferred step size varies by layer: a larger step can reduce transient error, whereas smaller steps can reduce the observation floor.

Table 9: Stationary replay: (normalized Φ, gradient capture). Qwen and OPT use steps 500 and 2000, respectively.

Model	Layer	$\eta_q = .03$	.1	.3	.6
Qwen3-4B	Early	(.943,.140)	(.926,.315)	(.892,.414)	(.837,.549)
	Middle	(.527,.898)	(.330,.901)	(.338,.894)	(.410,.886)
	Late	(.533,.924)	(.618,.923)	(.657,.922)	(.670,.921)
OPT-2.7B	Early	(.161,.713)	(.263,.677)	(.349,.646)	(.384,.635)
	Middle	(.349,.943)	(.417,.940)	(.559,.935)	(.617,.933)
	Late	(.440,.704)	(.572,.662)	(.663,.648)	(.697,.644)

D.2 Active-width diagnostics

The following scan compares gradient capture and single-direction alignment across active widths.

Table 10: Full active-space width scan using independent Gaussian coefficient matrices. Each cell gives learned capture / random capture / mean learned single-direction first-order cosine.

Width	Qwen3-4B	OPT-2.7B
8	.7022 / .0020 / .002272	.8137 / .0026 / .002855
16	.7758 / .0068 / .001615	.8458 / .0043 / .002052
32	.8243 / .0121 / .001072	.8736 / .0082 / .001374
64	.8624 / .0277 / .000829	.8975 / .0183 / .000982
128	.8866 / .0510 / .000564	.9166 / .0416 / .000752

Capture increases while single-direction estimate alignment decreases as coefficient dimension grows. At active width 64, first-48 plus random-tail-16 capture is .8526 on Qwen and .8928 on OPT, compared with .8624 and .8975 for top-64.

D.3 ONE-SIDED AND TWO-SIDED ESTIMATION

We examine one-sided and two-sided estimates under a fixed objective, a finite-training-set gradient reference, and the BF16 rank-one implementation.

Fixed-objective Gaussian test. The checkpoint, minibatch, top-64 basis, and three representative `q_proj` matrices are fixed. OP uses N positive endpoints with population centering and denominator $N-1$; TP uses $N/2$ pairs, for the same number of perturbed loss evaluations. Estimates are compared with the exact autograd gradient of that minibatch, including energy outside the active subspace. Across five perturbation scales and $N \in \{4, 8, 16\}$, FP32 OP has lower mean MSE in all 15 settings per model, in both learned and random spaces. At $N = 16$ and $\epsilon = 8 \times 10^{-5}$, learned-space MSE reductions are 45.8% on Qwen and 48.1% on OPT. Unnormalized Gaussian BF16 tests favor OP in only 3/30 learned-space and 4/30 random-space settings; they are numerically sensitive and differ from the normalized implementation.

Matched perturbations in learned, random, and full spaces. A separate FP32 control uses full Gaussian $AQ^\top$ for learned and random spaces and full iid Gaussian matrices for full-space perturbations. Directions are layer-Frobenius normalized to $\sqrt{pd}$; structured estimates are divided by the induced covariance factor d/K. At batch size 1, $K = 64$, $\epsilon = 8 \times 10^{-5}$, and 16 repeated perturbation populations, we obtain Table 11. OP16 uses 16 one-sided directions and TP8 uses eight two-sided directions; both require 16 perturbed loss evaluations.

Table 11: Matched normalized Gaussian control against a fixed batch gradient. MSE is normalized by gradient energy; cosine gap is OP minus TP.

Model	Space	OP MSE	TP MSE	Reduction	Cosine gap
Qwen3-4B	Learned	50,357	94,002	46.4%	1.192×10^{-3}
	Random	1,042	1,680	38.0%	1.911×10^{-4}
	Dense	1,825,752	3,487,915	47.7%	1.647×10^{-4}
OPT-2.7B	Learned	30,755	65,777	53.2%	1.129×10^{-3}
	Random	764	1,369	44.2%	2.530×10^{-4}
	Dense	1,115,083	1,998,325	44.2%	2.280×10^{-4}

OP lowers MSE in all three spaces. The learned-minus-random interaction in the OP–TP cosine gap is 1.001×10^{-3} on Qwen (95% Monte Carlo interval $[7.64 \times 10^{-4}, 1.24 \times 10^{-3}]$) and 8.756×10^{-4} on OPT ($[3.58 \times 10^{-4}, 1.39 \times 10^{-3}]$). Against full-space perturbations the interactions are 1.027×10^{-3} ($[8.17 \times 10^{-4}, 1.24 \times 10^{-3}]$) and 9.006×10^{-4} ($[4.50 \times 10^{-4}, 1.35 \times 10^{-3}]$).

Finite-training-set gradient. OP16 and TP8 estimates from 16 mutually disjoint 16-example minibatches are compared with the exact 1000-example mean gradient at a fixed checkpoint. The selected three matrices are evaluated jointly using FP32 full-Gaussian structured or full-space perturbations. This protocol incorporates minibatch noise.

Table 12: Finite-training-set reference at $\epsilon = 8 \times 10^{-5}$, $K = 64$, and 16 loss evaluations.

Model	Space	OP MSE	TP MSE	Reduction	Cosine gap
Qwen3-4B	Learned	1,995,835	3,743,201	46.7%	$+7.47 \times 10^{-5}$
	Random	45,462	91,242	50.2%	$+3.06 \times 10^{-5}$
	Dense	96,707,409	192,548,225	49.8%	-3.72×10^{-5}
OPT-2.7B	Learned	1,374,688	2,154,535	36.2%	$+8.44 \times 10^{-5}$
	Random	31,731	53,723	40.9%	$+7.29 \times 10^{-5}$
	Dense	63,415,059	116,824,528	45.7%	-2.92×10^{-5}

At small tested scales, learned-space mean cosine favors OP whereas dense mean cosine favors TP in both models. At 10^{-3}, the learned-space gain disappears on Qwen and shrinks on OPT. The paired 95% intervals for learned-space gains include zero with 16 repetitions; OP has lower MSE in every tested space and scale.

Table 13: Perturbation-scale control against the exact training gradient. Entries are OP MSE reduction (%) / OP–TP cosine gap in units of 10^{-5}.

Model	ϵ	Learned	Random	Dense
Qwen3-4B	6×10^{-5}	46.7 / +7.64	50.2 / +2.99	49.8 / -3.69
	7.43×10^{-5}	46.7 / +7.48	50.2 / +3.04	49.9 / -3.72
	10^{-3}	39.1 / -1.17	50.3 / +3.07	50.6 / -3.70
OPT-2.7B	2.57×10^{-5}	36.5 / +7.66	41.1 / +7.49	45.7 / -2.68
	6×10^{-5}	36.3 / +8.51	41.2 / +7.28	45.7 / -2.98
	10^{-3}	33.9 / +2.55	40.9 / +7.21	45.7 / -2.82

Implementation-level direction alignment. This control compares rank-one $ABQ^\top$, layer-Frobenius-normalized BF16 perturbations at batch size 16 with dense Gaussian perturbations. OP uses 15 positive endpoints plus one center; TP uses eight pairs, each costing 16 loss evaluations. Cosine is measured against the exact gradient of the same fixed minibatch at $\epsilon = 8 \times 10^{-5}$, with 16 paired repetitions.

Table 14: BF16 implementation-level cosine. All numeric entries are in units of 10^{-5}; intervals are paired 95% intervals conditional on the checkpoint and batch.

Model	Space	OP	TP	Difference	95% interval
Qwen3-4B	Learned	38.72	32.93	5.791	[1.96,9.63]
	Dense	2.774	2.443	.3307	[-.514,1.18]
OPT-2.7B	Learned	40.48	32.60	7.882	[4.17,11.6]
	Dense	5.482	4.246	1.236	[.536,1.94]

Learned-space directions have higher cosine alignment, and their mean OP–TP gain is larger in both models.

E ACTIVE-SUBSPACE ABLATIONS

E.1 SHARED VERSUS INDEPENDENTLY RESAMPLED ACTIVE SUBSPACES

We compare using one common active basis for the perturbations within an update with independently resampling the candidate columns for each perturbation. Both variants retain the shared directions and use the same population estimator. Table 15 reports development-selected official validation scores on OPT-2.7B, using five matched seeds and 40,000 training forward evaluations. Independent resampling gives higher mean scores on all four tasks in this comparison.

Table 15: Common active basis versus per-perturbation resampling on OPT-2.7B. Classification scores are accuracy; SQuAD uses F1 (%).

Variant	RTE	BoolQ	SST-2	SQuAD
MeZO	65.13 ± 1.67	66.10 ± 2.26	$92.50 \pm .54$	80.99 ± 1.42
Common active basis	$66.14 \pm .78$	$66.84 \pm .63$	$92.48 \pm .78$	81.06 ± 1.02
AIM-ZO	67.51 ± 3.18	67.22 ± 1.67	$92.87 \pm .47$	$81.19 \pm .49$

E.2 MAINTENANCE REFRESH FREQUENCY

On OPT-2.7B RTE, we compare a fixed activation subspace with Oja updates every ten steps and every step. Table 16 reports final development accuracy after 2,500 updates over five seeds; per-step updates have the highest mean.

Table 16: OPT-2.7B RTE maintenance-frequency ablation (five seeds; final development accuracy, %).

Maintenance rule	Accuracy
Fixed subspace	68.40 ± 2.04
Oja every ten steps	68.92 ± 3.38
Oja every step	70.20 ± 1.95

E.3 MAINTAINED WIDTH AND FIXED-PREFIX SELECTION

Table 17 compares width-64 and width-128 maintenance with a fixed prefix of 64 active directions. The AIM-ZO column gives the main-result reference at $K = 128$, $k = 64$, and $h = 48$. The fixed-prefix variants use three seeds and 40,000 training forward evaluations; the main-result reference uses five seeds. All scores are official RTE accuracy on 277 examples. The two fixed-prefix widths use separately initialized maintained spaces, so this is a configuration ablation rather than a test of additional columns acting on an identical prefix.

Table 17: RTE accuracy (%) with fixed-prefix activation and the AIM-ZO main-result reference.

Model	$K = 64,\ k = h = 64$	$K = 128,\ k = h = 64$	AIM-ZO
OPT-2.7B	64.26 ± 3.82	66.55 ± 1.10	$\mathbf{67.51} \pm 3.18$
Qwen3-0.6B	77.38 ± 1.71	$77.38 \pm .42$	$\mathbf{77.62} \pm .92$

E.4 MAINTAINED WIDTH WITH SAMPLED TAILS

We vary K while fixing $k = 64$ and $h = 48$ on OPT-2.7B RTE. All three variants use BF16, three seeds, 40,000 training forward evaluations, and an Oja refresh every ten steps. The $K = 128$ row shares the three-seed rank-one setting of Table 21. At $K = 64$, all 16 tail columns are selected; at larger widths, they are sampled from the maintained tail. Table 18 reports the resulting accuracy.

Table 18: OPT-2.7B RTE maintained-width scan at $k = 64$ and $h = 48$ (three seeds; official-validation accuracy, %).

Maintained width K	Accuracy
64	65.70 ± 1.25
128	65.46 ± 3.62
256	67.15 ± 0.72

E.5 Shared width

At fixed $K = 128$ and $k = 64$, we vary the number h of shared directions in the same OPT-2.7B RTE protocol. The $h = 48$ row reuses the rank-one control above; $h = 64$ is the fixed-prefix OPT-2.7B result from Table 17. Table 19 reports the three-seed RTE comparison.

Table 19: OPT-2.7B RTE shared-width scan at $K = 128$ and $k = 64$ (three seeds; official-validation accuracy, %).

Shared width h	Sampled tail $k - h$	Accuracy
0	64	60.77 ± 1.99
48	16	65.46 ± 3.62
64	0	66.55 ± 1.10

E.6 Population size

We compare eight and 15 one-sided population members on OPT-2.7B RTE at $K = 128$, $k = 64$, and $h = 48$. The $N = 8$ runs use 4,444 updates with nine forward evaluations per update (39,996 in total); the $N = 15$ control uses 2,500 updates with 16 evaluations per update (40,000 in total). Both use three seeds and Oja refresh every ten steps. Table 20 reports development-selected official-validation accuracy. The larger population has the higher mean under this matched forward-evaluation budget.

Table 20: OPT-2.7B RTE population-size ablation (three seeds; official-validation accuracy, %).

Population size N	Training forwards	Accuracy
8	39,996	61.01 ± 3.77
15	40,000	65.46 ± 3.62

E.7 Perturbation coefficient rank

Table 21 compares rank-one, rank-four, and full Gaussian coefficients in the active subspace on OPT-2.7B RTE. Each variant uses three seeds, BF16, and 40,000 training forward evaluations. Mean accuracies are similar; step times summarize the median non-evaluation time of each training run.

Table 21: Coefficient-rank ablation on OPT-2.7B RTE over three seeds. Accuracy is official-validation accuracy (%); time is mean per-run median non-evaluation step time; memory is peak CUDA allocated memory.

Coefficient form	Accuracy (%)	Step time (s)	Memory (GiB)
Rank-one $ab^\top Q^\top$	65.46 ± 3.62	10.35	11.214 ± 0.158
Rank-four $ABQ^\top$	64.98 ± 2.01	9.77	11.216 ± 0.160
Full $RQ^\top$	65.58 ± 2.21	12.24	11.153 ± 0.105

F Space Quality and One-Sided Estimation

F.1 Online estimator ablation

We first compare OP15-RLOO with raw TP8 in complete OPT-2.7B RTE training runs over five matched seeds. The two variants share the training data, 2,500 updates, Oja maintenance, and perturbation settings. OP15-RLOO uses 15 positive evaluations and one centre evaluation per update; TP8 uses eight positive–negative pairs and one centre evaluation, giving 40,000 and 42,500 loss evaluations in total. Table 22 reports development accuracy. OP15-RLOO has a 1.36-point higher mean at the best-accuracy checkpoint and a 1.84-point higher mean after the final update.

Table 22: Online OPT-2.7B RTE estimator ablation over five matched seeds (development accuracy, %).

Estimator	Best checkpoint	Final update
OP15-RLOO	71.40 ± 3.02	71.12 ± 3.29
Raw TP8	70.04 ± 4.38	69.28 ± 4.62

F.2 Space quality at fixed active width

We isolate space quality at fixed active width $k = 64$ by rotating a learned orthonormal basis toward an orthogonal random complement:

$$Q(s) = \sqrt{s}\, Q_{\text{learned}} + \sqrt{1-s}\, Q_{\perp}, \qquad Q_{\text{learned}}^{\top} Q_{\perp} = 0.$$

Each basis has the same dimension. The seven rotation levels are $s \in \{0, .1, .25, .5, .75, .9, 1\}$, and capture is measured at each level. All levels use matched full-Gaussian coefficient matrices, FP32 arithmetic, and no layer-wise Frobenius normalization. OP uses 16 population-centered one-sided directions and TP uses eight two-sided pairs, each with 16 perturbed loss evaluations. For $Z_i = A_i Q(s)^{\top}$, let $y_i^{\pm} = f(W \pm \epsilon Z_i)$ and $\bar{y}^{+} = N^{-1} \sum_{i=1}^{N} y_i^{+}$. The compared estimators are

$$\widehat{G}_{\text{OP}} = \frac{1}{\epsilon(N-1)} \sum_{i=1}^{N} (y_i^{+} - \bar{y}^{+}) Z_i, \qquad \widehat{G}_{\text{TP}} = \frac{1}{M} \sum_{i=1}^{M} \frac{y_i^{+} - y_i^{-}}{2\epsilon} Z_i, \tag{27}$$

where $N = 16$ and $M = 8$. We measure $\Delta_{\cos} = \cos(\widehat{G}_{\text{OP}}, G) - \cos(\widehat{G}_{\text{TP}}, G)$.

Exact-linear control. At eight saved checkpoints per model, the reference is the exact mean gradient over the 1,000-example RTE training set. In this control, the endpoint values are $y_i^{\pm} = f(W) \pm \epsilon \langle G, Z_i \rangle_F$. Each rotation level uses 64 paired perturbation populations with an exact linear loss oracle. Table 23 averages populations within each checkpoint and then averages the checkpoint means. The alignment gain increases with capture at every checkpoint; Pearson correlations over checkpoint-level quality means are .970 for Qwen3-4B and .965 for OPT-2.7B.

Table 23: Fixed-width space-quality control with an exact linear oracle. Alignment gains are reported in units of 10^{-3}.

	Qwen3-4B		OPT-2.7B	
s	Capture	$\Delta_{\cos}$	Capture	$\Delta_{\cos}$
0	.0046	.0795	.0020	.0668
.10	.0857	.365	.0931	.448
.25	.2072	.567	.2304	.705
.50	.4094	.798	.4598	.996
.75	.6116	.975	.6894	1.220
.90	.7329	1.067	.8274	1.337
1	.8136	1.124	.9199	1.410

Real-loss control. The corresponding FP32 forward evaluations cover three checkpoints per model and four independent one-example batches per checkpoint, with two perturbation scales and 16 paired populations per quality level. The reference is the exact gradient of the fixed batch objective. Across all 48 checkpoint/batch/scale settings, the mean alignment gain increases with capture. All 336 checkpoint/batch/scale/quality mean-MSE comparisons favor OP (Table 24). These tests complement the rank-one BF16 implementation diagnostic in Table 14.

Table 24: Real-loss space-quality sweep. Correlation relates measured capture to the OP–TP alignment gain; monotone counts refer to checkpoint/batch settings.

Model	ϵ	Correlation	Monotone	OP MSE wins
Qwen3-4B	6×10^{-5}	.952	12/12	84/84
	10^{-3}	.952	12/12	84/84
OPT-2.7B	2.57×10^{-5}	.927	12/12	84/84
	10^{-3}	.926	12/12	84/84

G Runtime and Peak Memory

G.1 Peak-memory scaling on DROP

We measure Qwen3-0.6B-Base on DROP using BF16, SDPA attention, `use_cache=False`, and no gradient checkpointing on a 24 GiB RTX 4090. Each of 48 unique method/batch/length configurations runs ten actual parameter updates in a separate process. Peak memory is PyTorch CUDA allocated memory, including model weights and the training update; OOM denotes an observed out-of-memory failure. FO-SGD uses full-parameter SGD without momentum or FP32 master weights. AIM-ZO uses rank-one perturbations, $K = 128$, $k = 64$, $h = 48$, 15 population members, and factor restoration caching.

Figure 4 shows that AIM-ZO uses .242–.254 GiB more memory than MeZO across the measured configurations and less than AGZO and FO-SGD. Inputs are padded to the stated tensor length, so the length sweep measures tensor-shape scaling.

G.2 Complete runs at a matched evaluation budget

Table 25 reports accumulated training-step time from completed 40,000-evaluation runs, excluding validation and checkpoint I/O. Each row averages three matched seeds. Tasks and execution conditions differ across rows; the ratios are within-row comparisons.

Table 25: Training-step hours at 40,000 forward evaluations. Ratio is AIM-ZO time divided by MeZO time.

Model	Task	AIM-ZO	MeZO	Ratio
Qwen3-0.6B	RTE	4.339	1.871	2.319
OPT-2.7B	RTE	7.363	3.508	2.099
OPT-13B	BoolQ	26.603	21.919	1.214

G.3 Large-model short-run measurements

Each model and method runs ten BF16 training updates in an independent process, with the first batch matched across methods. Table 26 reports mean step time over steps 6–10 and peak allocated memory; time per evaluation divides mean step time by the number of forward evaluations.

Table 26: Large-model short-run time and memory.

Model	Method	Evaluations/step	s/step	s/eval.	GiB
Qwen3-8B	MeZO	2	.653	.327	19.929
	AIM-ZO	16	7.396	.462	19.937
	AGZO	3	1.196	.399	18.772
OPT-30B	MeZO	2	1.810	.905	58.552
	AIM-ZO	16	14.052	.878	60.034
	AGZO	3	2.727	.909	58.087

G.4 Restoration caching

On Qwen3-0.6B-Base/SQuAD, we compare three AIM-ZO caching variants over ten updates on an RTX 4090 with matched batches. All variants produce identical parameter updates. Factor caching reduces mean step time by 20.6% for an additional .036 GiB of allocated memory (Table 27); full-noise caching yields little further speedup.

Table 27: Cache ablation for AIM-ZO. Time averages all ten steps; allocated and reserved memory are reported separately.

Cache	s/step	Allocated GiB	Reserved GiB
None	9.164	10.251	19.111
Factor restoration	7.273	10.287	19.143
Factor restoration + full noise	7.248	10.576	20.516

H Additional Fine-Tuning Results

This section provides additional task results and evaluation metrics. Experimental settings are described in Appendix C.

Main-table evaluation details. OPT-2.7B CurvZO RTE and OPT-13B AIM-ZO WiC use four seeds, while OPT-2.7B HiZOO WiC uses two. OPT-13B LoZO RTE and BoolQ use seeds 42, 142, and 242. MeZO, CurvZO, and LoZO SQuAD results select checkpoints by generated-answer development F1 and evaluate on an independent 1,000-example official-validation subset. The MeZO SQuAD entry uses full-parameter generation CE. A separate reproduction of MeZO's five-token-prefix F1 setup reaches 75.29 ± 2.61 F1 over three seeds; the main table uses the full-parameter setting.

H.1 Complete OPT-2.7B task coverage

Table 28: Five-seed official-validation results for OPT-2.7B under 40,000 matched training forwards. Higher is better. W/T/L counts paired-seed outcomes for AIM-ZO relative to MeZO. Scores are percentages.

Dataset	Metric	MeZO	AIM-ZO	W/T/L
RTE	Accuracy	65.13 ± 1.67	$\mathbf{67.51 \pm 3.18}$	4/1/0
BoolQ	Accuracy	66.10 ± 2.26	$\mathbf{67.22 \pm 1.67}$	5/0/0
SST-2	Accuracy	92.50 ± 0.54	$\mathbf{92.87 \pm 0.47}$	5/0/0
WiC	Accuracy	58.53 ± 0.41	$\mathbf{60.13 \pm 1.70}$	5/0/0
MultiRC	Accuracy	61.96 ± 2.04	$\mathbf{63.94 \pm 1.59}$	4/0/1
MultiRC	F1a	34.71 ± 10.81	$\mathbf{48.91 \pm 4.09}$	5/0/0
ReCoRD	F1	88.07 ± 0.37	$\mathbf{88.55 \pm 0.20}$	4/0/1
SQuAD	F1	80.99 ± 1.42	$\mathbf{81.19 \pm 0.49}$	3/0/2
DROP	F1	24.50 ± 0.46	$\mathbf{25.78 \pm 0.45}$	5/0/0
WSC	Accuracy	54.81 ± 1.52	$\mathbf{56.73 \pm 4.35}$	4/0/1
COPA	Accuracy	$\mathbf{80.80 \pm 2.17}$	80.60 ± 3.21	2/0/3
CB	Accuracy	$\mathbf{68.93 \pm 2.40}$	67.86 ± 3.57	2/0/3
CB	Macro-F1	$\mathbf{53.66 \pm 8.83}$	49.36 ± 6.82	1/0/4

Using each task's primary metric, AIM-ZO has the higher five-seed mean on 9 of 11 tasks, with 43 wins, 1 tie, and 11 losses across the 55 paired task–seed comparisons. The largest gain is on MultiRC F1a. COPA is approximately tied, while CB favors MeZO under both accuracy and macro-F1.

Additional SQuAD metrics. Zero-shot EM is 11.00 on OPT-2.7B and 21.90 on OPT-13B. On OPT-13B, MeZO, CurvZO, LoZO, and AIM-ZO obtain EM scores of 44.13 ± 1.76, 51.73 ± 1.26, 34.23 ± 6.79, and 71.07 ± 0.29, respectively. On OPT-30B, MeZO and AIM-ZO obtain 70.10 ± 1.36 and 72.88 ± 0.88 EM over five seeds.

H.2 Qwen3-0.6B results

Table 29 reports the six-task comparison on Qwen3-0.6B-Base. The task set replaces WSC with MultiRC. Trained methods use three seeds and approximately 40,000 training forward evaluations; SQuAD is evaluated on a fixed 1,000-example official-validation subset.

Table 29: Multi-method comparison on Qwen3-0.6B-Base (%). SQuAD uses F1; other tasks use accuracy. Avg. is the unweighted mean over six tasks; subscripts report available sample standard deviations.

Method	RTE	BoolQ	SST-2	WiC	MultiRC	SQuAD	Avg.
Zero-shot	58.48	62.20	58.26	51.25	59.08	56.22	57.58
MeZO	$77.62_{\pm 1.11}$	$73.87_{\pm 0.57}$	$88.42_{\pm 1.35}$	$55.36_{\pm 2.44}$	$72.78_{\pm 1.49}$	$83.23_{\pm 0.76}$	75.21
CurvZO	$77.14_{\pm 1.04}$	$68.28_{\pm 0.80}$	$82.87_{\pm 1.15}$	$49.74_{\pm 1.37}$	$71.72_{\pm 2.86}$	$71.29_{\pm 3.46}$	70.17
HiZOO	$77.86_{\pm 0.55}$	$72.76_{\pm 0.84}$	$88.80_{\pm 0.88}$	$53.92_{\pm 1.96}$	$76.11_{\pm 0.93}$	$80.42_{\pm 0.89}$	74.98
AGZO	$69.68_{\pm 4.39}$	$68.40_{\pm 0.26}$	$84.75_{\pm 3.04}$	$52.77_{\pm 3.70}$	$73.89_{\pm 0.38}$	$73.17_{\pm 1.43}$	70.44
ZO-Muon	$75.69_{\pm 0.21}$	$72.73_{\pm 0.79}$	$88.76_{\pm 0.70}$	$57.79_{\pm 2.05}$	$75.10_{\pm 0.63}$	$81.10_{\pm 1.36}$	75.20
LoZO	$73.53_{\pm 0.21}$	$69.88_{\pm 1.49}$	$85.67_{\pm 0.75}$	$52.93_{\pm 0.90}$	$72.59_{\pm 0.70}$	$76.74_{\pm 1.17}$	71.89
AIM-ZO	$77.62_{\pm 0.92}$	$73.84_{\pm 0.56}$	$88.94_{\pm 0.70}$	$54.92_{\pm 2.11}$	$76.83_{\pm 0.54}$	$81.97_{\pm 0.57}$	75.69

For SQuAD, zero-shot EM is 39.00. MeZO, CurvZO, HiZOO, AGZO, ZO-Muon, LoZO, and AIM-ZO obtain EM scores of 74.57 ± 1.12, 60.13 ± 4.01, 71.77 ± 1.15, 61.63 ± 1.77, 71.67 ± 1.45, 66.30 ± 1.57, and 72.87 ± 0.59, respectively.

H.3 Additional Qwen3-8B metrics

Table 30 gives Qwen3-8B metrics omitted from the six-task comparison in Table 2. The MeZO and AIM-ZO results use the same three seeds and development-selected checkpoints as the main table. The AGZO main-table row uses three seeds, BF16, and 39,999 training forward evaluations, with checkpoints selected from five development milestones.

Table 30: Additional Qwen3-8B-Base evaluation metrics (%; three seeds for trained methods).

Metric	Zero-shot	MeZO	AIM-ZO
MultiRC F1a	54.70	83.95 ± 0.25	84.48 ± 0.37
MultiRC question EM	28.23	54.60 ± 1.37	56.31 ± 0.44
SQuAD EM	72.40	78.20 ± 2.23	84.63 ± 0.42

I Additional Related Work

Table 31 summarizes the design differences among representative ZO methods.

Table 31: Design properties of representative ZO methods. Sparse coordinate spaces count as subspace perturbations; guidance and history refer to information used in perturbation construction.

Method	Subspace perturbations	Information-guided	History-informed	Width decoupling
MeZO	×	×	×	×
HiZOO	×	✓	✓	×
CurvZO	✓	✓	✓	×
LoZO	✓	×	×	×
SubZero	✓	×	×	×
ZO-Muon	✓	×	×	×
AGZO	✓	✓	×	×
AIM-ZO	✓	✓	✓	✓

Fixed activation-informed subspaces. ZO-Act constructs a fixed low-rank basis from an initial activation snapshot and uses it to restrict perturbations throughout fine-tuning (Dong et al., 2026). This avoids repeated basis construction, but does not refresh the activation-derived basis as model parameters change.

Sparse parameter selection. Sparse MeZO selects parameters according to their magnitudes and sensitivity to ZO estimation noise (Liu et al., 2025). Transferable Static Sparsity instead reuses a predetermined parameter mask throughout training (Guo et al., 2025). These approaches restrict perturbations in parameter coordinates, whereas a matrix subspace can represent combinations of coordinates through its basis vectors.

Adaptive spaces and temporal gradient structure. ASEBO estimates an adaptive space from the covariance of full-dimensional evolution-strategy gradient estimates and allocates function evaluations between that space and its orthogonal complement (Choromanski et al., 2019). AIM-ZO maintains layer-wise bases from forward activations and samples smaller active subspaces within them. Prior studies have observed that gradient information can concentrate in low-dimensional subspaces that remain stable or evolve gradually over portions of training (Gur-Ari et al., 2018; Jaiswal et al., 2025). First-order optimisers exploit related temporal structure: GaLore periodically refreshes projection bases from gradient SVDs, whereas Online Subspace Descent and SubTrack++ update or track them from backpropagated gradients (Zhao et al., 2024; Liang et al., 2024; Rajabi et al., 2025).

Streaming subspace estimation. Streaming PCA methods such as block Oja provide tools for incrementally estimating a leading eigenspace (Allen-Zhu & Li, 2017; Huang et al., 2021). Classical guarantees typically assume a stationary covariance; robust streaming PCA also studies Oja updates under temporally drifting covariances (Bienstock et al., 2022). AIM-ZO uses this maintained activation space to construct ZO perturbations.