ON ADAPTIVITY IN ZEROTH-ORDER OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate the effectiveness of adaptive zeroth-order (ZO) optimization for memory-constrained fine-tuning of large language models (LLMs). Contrary to prior claims, we show that adaptive ZO methods such as ZO-Adam provide no performance or convergence advantage over well-tuned ZO-SGD, while incurring significant memory overhead due to optimizer state. We provide theoretical justification for why Adam-like adaptivity is less effective in large-scale ZO settings, and confirm this through both synthetic and LLM fine-tuning experiments. Importantly, we show empirically that adaptivity can still provide a practical benefit in ZO optimization by improving robustness to step size selection. To retain this robustness without incurring the memory cost of optimizer state, we propose MEAZO, a **M**emory-**E**fficient **A**daptive **ZO** optimizer that tracks only a single scalar to adapt the global step size. We support our method with theoretical convergence guarantees under standard smoothness assumptions. Experiments on synthetic quadratic problems, multiple LLM families, and diverse fine-tuning tasks demonstrate that MEAZO performs comparably to competing methods while matching the robustness gains of ZO-Adam over ZO-SGD, without additional memory or latency overhead.

1 INTRODUCTION

Fine-tuning large language models (LLMs) can specialize their behavior to various use-cases: aligning to a specific domain, application, task (Gururangan et al., 2020) or integrating user preferences (Gao et al., 2024) are some examples thereof.

Backpropagation is a standard but memory and computationally heavy fine-tuning algorithm, due to its need of activation caching and floating-point precision. Zeroth-order (ZO) methods on the other hand have recently gained attention in the LLM literature as they allow model fine-tuning using only forward passes (Malladi et al., 2023), and are thus more memory and computationally efficient. In a nutshell, ZO optimization approximates the gradient through the loss difference at randomly-perturbed trainable parameters.

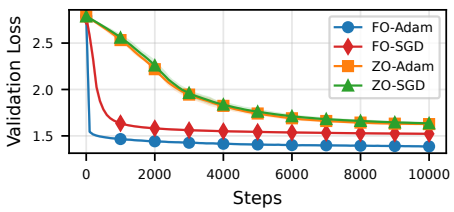


Figure 1: Unlike their FO counterparts, ZO-SGD and ZO-Adam exhibit similar performance, under step size tuning when fine-tuning Llama-3.2 1B on XSum.

Due to the inherent stochasticity of parameter perturbations, ZO gradients exhibit high-variance leading to slower convergence in terms of number of optimization steps compared to first-order (FO) methods. However, each ZO step is typically less computationally intensive than the backpropagation step required by FO methods. Therefore, ZO vs. FO poses a trade-off in speed/energy to convergence given specific hardware constraints and task-specific characteristics.

A substantial line of recent work aims to address this variance issue by introducing adaptive ZO algorithms, such as ZO-Adam and its variants (Chen et al., 2019; Nazari et al., 2020; Jiang et al., 2024; Shu et al., 2025). These methods mirror FO adaptive optimizers by tracking first and second moments of ZO gradients with the goal of improving convergence.

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.

In this work, we revisit this assumption and show that, contrary to prior claims, FO-style adaptivity provides limited benefits for ZO optimization in high-dimensional transformer models, as we observe in Fig. 1. Recent studies on FO training dynamics (Zhang et al., 2024b; Tomihari & Sato, 2025) demonstrate that Adam's advantage over SGD arises from gradient heterogeneity: a strong coordinate-wise variation in gradient magnitudes. We provide evidence that ZO gradient estimates, due to the isotropic nature of random perturbations in high dimensions, are essentially homogeneous. Consequently, adaptive per-coordinate normalization has little meaningful structure to exploit.

Contributions. Specifically, we make the following contributions: **(C1)** Through extensive experiments, we demonstrate that, contrary to prior claims, ZO adaptive methods provide no convergence or performance benefits over well-tuned ZO-SGD. **(C2)** We show, both theoretically and empirically, that ZO gradient estimates are nearly homogeneous due to isotropic random perturbations, leaving little exploitable structure for per-coordinate adaptivity. **(C3)** We demonstrate that adaptive ZO methods improve robustness to step size selection, redeeming them despite not showing performance benefits. And **(C4)** we propose MEAZO, a memory-efficient adaptive zeroth-order optimizer that achieves adaptive scaling using a single scalar while matching the memory footprint of ZO-SGD and maintaining the step-size robustness of ZO-Adam. We further establish a convergence guarantee for MEAZO under standard smooth non-convex assumptions and verify our claims across various synthetic and LLM-based benchmarks.

2 Problem Setup and Notation

We now begin by formalizing the context of our study.

Notation. We adopt the following conventions. Vectors are denoted by bold lowercase letters (*e.g.*, $\mathbf{x}$, $\mathbf{y}$). Matrices are denoted by bold uppercase letters (*e.g.*, $\mathbf{W}$, $\mathbf{A}$, $\mathbf{B}$). We use the following indexing notation: $[\mathbf{x}]_k := x_k$ to represent the k^{th} entry of $\mathbf{x}$.

Setup. We study the standard stochastic optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^d} F(\mathbf{x}) := \mathbb{E}_{\xi\sim\mathcal{D}}\left[f(\mathbf{x};\xi)\right], \tag{1}$$

where $F : \mathbb{R}^d \to \mathbb{R}$ is the objective function, $\xi \sim \mathcal{D}$ denotes samples from a data distribution $\mathcal{D}$, and f is the sample loss function. The variable $\mathbf{x} \in \mathbb{R}^d$ is the d-dimensional parameter vector we aim to optimize. When clear from context, we drop the dependence on ξ and write $f(\mathbf{x};\xi)$ as $f(\mathbf{x})$.

We assume the sample loss is L-smooth, as standard in non-convex optimization.

Assumption 2.1 (L-smoothness)**.** The function $f : \mathbb{R}^d \to \mathbb{R}$ is differentiable and satisfies $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \le L\|\mathbf{x}-\mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

ZO Optimization. In the ZO setting, we aim to solve (1) without access to exact gradients of f. Instead, optimization relies solely on function evaluations. A central building block is the finite-difference approximation of the directional derivative:

Definition 2.2 (Projected Gradient)**.** For a point $\mathbf{x} \in \mathbb{R}^d$, a direction vector $\mathbf{u} \in \mathbb{R}^d$, and a perturbation magnitude $\varepsilon > 0$, the *ZO projected gradient* (finite-difference directional derivative) is defined as

$$\Delta f_\varepsilon(\mathbf{x};\mathbf{u}) := \frac{f(\mathbf{x}+\varepsilon\mathbf{u}) - f(\mathbf{x}-\varepsilon\mathbf{u})}{2\varepsilon} \approx \mathbf{u}^\top \nabla f(\mathbf{x}), \tag{2}$$

which provides a zeroth-order estimate of the directional derivative of f at $\mathbf{x}$ along the direction $\mathbf{u}$ (when f is differentiable).

Using this building block, we can construct an estimate of the full gradient through random sampling of direction vectors.

Definition 2.3 (ZO Gradient Estimator)**.** Given a function $f : \mathbb{R}^d \to \mathbb{R}$, the q-sample zeroth-order (ZO) gradient estimator (Malladi et al., 2023) is defined as

$$\hat{\nabla} f_\varepsilon^q(\mathbf{x}) := \frac{1}{q}\sum_{i=1}^{q} \Delta f_\varepsilon(\mathbf{x};\mathbf{u}_i)\,\mathbf{u}_i, \tag{3}$$

where $\{\mathbf{u}_i\}_{i=1}^q$ are i.i.d. random direction vectors satisfying $\mathbb{E}[\mathbf{u}] = \mathbf{0}$ and $\mathbb{E}[\mathbf{u}\mathbf{u}^\top] = \sigma\mathbf{I}_d$, and $\varepsilon > 0$ is the finite-difference perturbation scale.

Variants of (3) exist (*e.g.*, choice of sampling distribution[1], forward vs. central differences). Unless stated otherwise, we assume $\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. These choices typically have minor impact on performance, as ZO methods exhibit similar behavior under standard settings (see Appendix D.7).

The estimator in (3) provides an unbiased estimate of the gradient of a *smoothed* version of the objective, defined as $f_\varepsilon(\mathbf{x}) := \mathbb{E}_{\mathbf{v}\sim\mathcal{Q}}\big[f(\mathbf{x} + \varepsilon\mathbf{v})\big]$. The smoothing distribution $\mathcal{Q}$ depends on the perturbation scheme: for uniform perturbations, smoothing occurs over the unit ball $\mathbb{B}$ (Flaxman et al., 2004), whereas for Gaussian perturbations, smoothing is under a normal distribution (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013). This smoothing interpretation is key: ZO methods do not approximate the true gradient of f directly but rather the gradient of f_ε.

3 Is ZO-SGD All You Need?

Coordinate-wise adaptive optimizers such as Adam (Kingma & Ba, 2014) are the default choice for training and fine-tuning modern LLMs in first-order (FO) settings. Their effectiveness is commonly attributed to the use of per-coordinate gradient statistics, which enable updates to be scaled differently across dimensions according to local curvature and gradient magnitudes. Specifically, Adam maintains bias-corrected exponential moving averages (EMAs) of the first moment, $\hat{\mathbf{m}}_t \in \mathbb{R}^d$, and second moment, $\hat{\mathbf{v}}_t \in \mathbb{R}^d$, of the gradient and updates parameters $\mathbf{x} \in \mathbb{R}^d$ as:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} + \zeta}, \tag{4}$$

where $\eta > 0$ is the step size and $\zeta > 0$ is a small quantity to improve numerical stability.

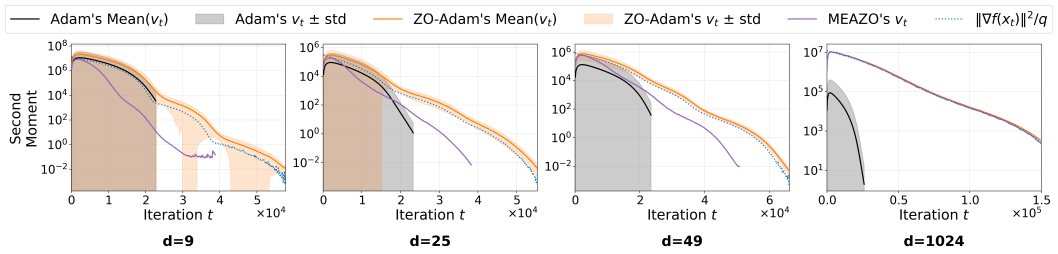
Given Adam's success in FO optimization, several works have adopted the same adaptive machinery to the ZO setting by simply replacing the true gradient with a ZO gradient estimator while retaining moment accumulation and coordinate-wise adaptivity. This yields optimizers commonly referred to as ZO-Adam or MeZO-Adam (Malladi et al., 2023; Zhang et al., 2024a). However, whether coordinate-wise adaptivity remains useful for ZO once gradients are replaced by noisy directional estimates remains largely unresolved.

Prior evidence on adaptivity in ZO. Empirical evidence on the benefits of adaptivity in ZO optimization remains mixed. Malladi et al. (2023) reported occasional gains from MeZO-Adam over MeZO, but observed substantial variability across tasks and concluded that understanding when adaptive methods help remains an open problem[2]. Similarly, Zhang et al. (2024a) found broadly comparable performance between ZO-SGD and ZO-Adam across a diverse benchmark suite, with no clear winner emerging. Nevertheless, several works argue that adaptivity should be particularly beneficial in ZO settings due to the high variance of gradient estimates. For example, Shu et al. (2025) motivate $\mathcal{R}$-AdaZO through variance reduction and adaptive scaling, while Chen et al. (2019) introduce ZO-AdaMM, a coordinate-wise adaptive optimizer inspired by Adam. However, neither study provides conclusive evidence that improvements arise from adaptivity itself rather than hyperparameter choices, as *comparisons against strong ZO-SGD baselines are limited*. More recently, alternative adaptive strategies have been explored. Ye & Luo (2026) and FZOO (Dang et al., 2025) adapt a global step size based on function-value statistics, whereas HiZOO (Zhao et al., 2025) incorporates approximate second-order information. These approaches differ substantially from the first-order, *coordinate-wise cross-steps adaptivity* considered in this work.

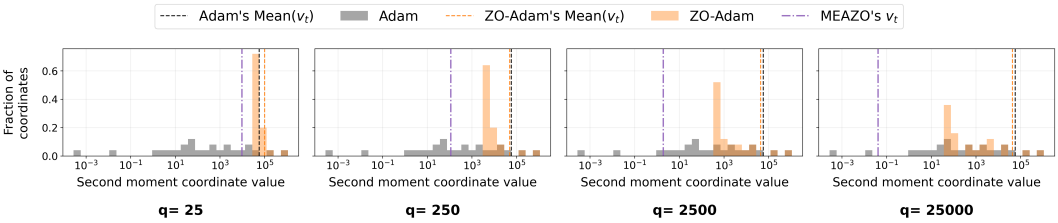
Taken together, the literature has contributed to the widespread belief that the noisy nature of ZO optimization necessitates adaptive methods. However, because existing empirical studies rarely perform extensive step size tuning, it remains unclear whether reported gains reflect the adaptive mechanisms themselves or merely favorable hyperparameter selection.

[1] When $\mathbf{u} \sim \mathrm{Uniform}(\mathbb{S})$, the estimator is scaled by d.

[2] Authors of Malladi et al. (2023) did not release an efficient implementation of MeZO-Adam, citing its *"comparable or worse"* performance than ZO-SGD (see https://github.com/princeton-nlp/MeZO/issues/37).



(a) Effect of d, the problem dimension ($q = 10$). In large dimensions, MEAZO and ZO-Adam overlap.



(b) Effect of q, the number of perturbations per step ($d = 25$, step $t = 5000$).

Figure 2: Distribution of second moment estimates of FO-Adam, ZO-Adam and MEAZO on synthetic heterogeneous quadratic objectives.

Well-tuned ZO-SGD is empirically highly competitive. Figure 1 illustrates a representative example on Llama-3.2 1B fine-tuning for XSum. Consistent with conventional wisdom, FO-Adam converges faster and achieves lower loss than FO-SGD. In contrast, the gap nearly disappears in the ZO setting, where ZO-SGD and ZO-Adam exhibit remarkably similar optimization behavior when both methods are carefully tuned (details on the step size sweep in Appendix C.2). Extensive experiments across models and tasks in Section 6 confirm that this phenomenon is not isolated.

Why coordinate-wise adaptivity becomes ineffective. Unlike FO gradients, the ZO estimator in (3) is dominated by randomness for practical values of q rather than per-coordinate gradient information. The estimator averages over only q random directions, where typically $q \ll d$. Consequently, even when averaging, the signal remains highly isotropic and does not provide rich per-coordinate patterns. We argue that applying coordinate-wise adaptive methods such as Adam in this setting introduces unnecessary complexity and computational overhead without providing meaningful benefits. Proposition 3.1 formalizes this intuition by analyzing the expected squared gradient:

Proposition 3.1 (Gaussian). *Let $f : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, and let $\hat{\nabla} f_\varepsilon^q(\mathbf{x})$ denote the q-sample ZO estimator in (3), where $\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ for all $i \in [q] := \{1, \dots, q\}$. Then in the $\varepsilon \to 0$ limit we have:*

$$\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right] \approx \frac{1}{q}\left(\|\nabla f(\mathbf{x})\|^2 \mathbf{1}_d + \nabla f(\mathbf{x})^2\right) + \nabla f(\mathbf{x})^2, \tag{5}$$

where $(\cdot)^2$ stands for the element-wise power of 2 and $\mathbb{E}_{\mathbf{u}}$ denotes the expectation over $\mathbf{u}_1, \dots, \mathbf{u}_q$.

The proof of Proposition 3.1 appears in Appendix F.1, and the corresponding result for the uniform perturbation estimator is given in Appendix F.2.

Proposition 3.1 reveals that the use of ZO gradient estimators introduces a bias in the expected squared gradient. This bias term scales as $(1/q)\|\nabla f(\mathbf{x})\|^2$ and vanishes as $q \to \infty$. However, in practical settings where $q \ll d$, and under the reasonable assumption that $\|\nabla f(\mathbf{x})\|^2$ grows with dimension d faster than $1/q$, this bias becomes dominant. Importantly, the dominant component depends on the global gradient norm rather than individual coordinates, making it effectively coordinate-agnostic. The same observation also applies to Proposition F.1 with uniform perturbations. Consequently, using coordinate-wise adaptive optimizers (*e.g.*, Adam) is misaligned with the underlying statistics, since the estimator tracks global rather than per-coordinate information in expectation.

Empirical observation of second moment collapse. The theoretical result above characterizes the *expected* behavior of the ZO estimator. We now investigate whether this phenomenon manifests during optimization. Specifically, we examine the second moment estimates maintained by Adam and ZO-Adam throughout training. Figure 2a visualizes the evolution of the second moment vector $\mathbf{v}_t$ when optimizing synthetic quadratic objectives of increasing dimension (experimental details are deferred to Appendix B). Consistent with Proposition 3.1, we observe that in low dimensions FO-Adam and ZO-Adam maintain broadly similar second moment statistics. As the dimensionality increases, however, the coordinate-wise structure present in FO-Adam largely disappears in ZO-Adam: the entries of $\mathbf{v}_t$ collapse toward a common value that is well approximated by $(1/q)\|\nabla f(\mathbf{x}_t)\|^2$. In other words, the second moment estimate becomes increasingly dominated by a global, coordinate-agnostic signal rather than meaningful per-coordinate information.

Figure 2b further shows that, even for low-dimensional problems, increasing q by orders of magnitudes, ZO-Adam fails to recover FO-Adam's second moment distribution. This effect is not limited to synthetic objectives. In Appendix D.1, we observe a similar concentration of ZO-Adam's second moment statistics during LLM fine-tuning. Together, these results suggest that the coordinate-wise adaptive mechanism that underlies Adam's success in FO optimization becomes progressively less informative in high-dimensional ZO settings, and one might conclude that adaptivity has no place in practical ZO LLM fine-tuning scenarios.

4 Adaptivity Improves Robustness, Not Performance

Although per-coordinate adaptive ZO methods offer little improvement over well-tuned ZO-SGD in terms of final performance, they provide a clear practical advantage: **improved robustness to step size selection**. To quantify this effect, we perform **dense** step size sweeps in two stages (roughly 18 values and 3 seeds per setting) and measure a robustness score (introduced next). Since such exhaustive tuning is prohibitively expensive in practice (see Table 5), robustness is a desirable property, reducing sensitivity to hyperparameter selection when extensive sweeps are infeasible.

Definition 4.1 (Robustness Score)**.** For a given training run with step size $\eta \in \mathbb{R}$ for $T \in \mathbb{N}^*$ steps, let $\ell(\eta) \coloneqq \min_{t \le T} \ell_{\text{val}}(\eta, t)$ denote the best validation loss attained when training with step size η, swept over a grid $\eta \in [\eta_{\min}, \eta_{\max}] \subset \mathbb{R}$. For a *ceiling* $\tau > 0$, the step size robustness score is

$$A_\tau(\ell) = \frac{1}{L} \int_{\eta_{\min}}^{\eta_{\max}} \max\big(\tau - \ell(\eta),\, 0\big)\, \mathrm{d} \log_{10} \eta \ , \tag{6}$$

where $L > 0$ is a *loss normalization constant* that puts the score in dimensionless, comparable units.

Interpretation. The score $A_\tau(\ell)$ measures the area between a method's loss curve $\ell(\eta)$ and the shared ceiling τ, restricted to the region where the curve lies under the ceiling (the "cup"). Unlike a simple bandwidth metric that only records the interval of acceptable step sizes, $A_\tau(\ell)$ jointly *captures both the extent* of the near-optimal region *and the performance* within that region. Because A_τ is defined over $\log_{10} \eta$, it is invariant to multiplicative rescalings of the step size, *i.e.*, $A_\tau(\ell(a\eta)) = A_\tau(\ell(\eta))$ for all $a > 0$ (see Figure 3).

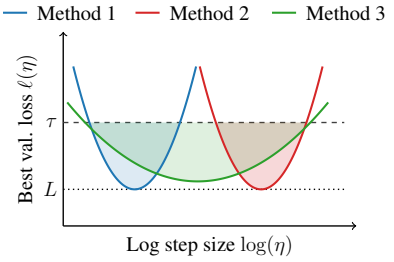


Figure 3: Visualization of the step size robustness score A_τ. Method 1 and 2 attain the same robustness score despite operating in different step size regions. Method 3 achieves a slightly worse optimal loss, yet attains a larger robustness score due to its substantially wider near-optimal step size region.

Choosing τ and L. Our goal is to compare optimizers under a fixed model/dataset setting, and therefore all methods share the same values of τ and L. We set $L = \min_{m,\eta} \ell_m(\eta)$, the best loss attained by any method m over the swept step sizes. To avoid underestimating robustness for methods whose loss curves remain below the ceiling at the sweep boundaries, we choose $\tau = \min_m \min(\ell_m(\eta_{\min}), \ell_m(\eta_{\max}))$, capped by the common initial validation loss. Our dense step size sweeps (Appendix C.2) ensure that the resulting robustness areas are reliably estimated. We also study alternative choices of τ in Appendix D.2, finding that while they induce different trade-offs, the qualitative conclusions remain unchanged.

Empirical robustness benefits of adaptivity. Figure 4 illustrates $A_\tau(\ell)$ being computed for Qwen-3 8B on SQuAD, where both adaptive methods (ZO-Adam and MEAZO) attain a larger cup area than ZO-SGD. It previews a consistent trend that we revisit in Section 6, where adaptive ZO methods are consistently at least as robust as ZO-SGD to step size selection, and often more so. Improving robustness to step size selection eventually reduces the need for expensive grid search for practitioners. As the figure illustrates, computing this score requires a dense step size sweep for every method, itself an expensive procedure (see Table 5 for compute costs).

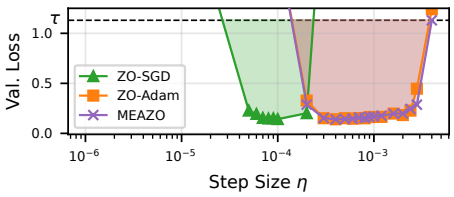


Figure 4: Best validation loss robustness and cup area robustness score for Qwen-3 8B on SQuAD: adaptive methods (ZO-Adam, MEAZO) sustain low loss over a wider step size range than ZO-SGD, yielding a larger shaded area.

Across all fine-tuning tasks considered in this work, ZO-Adam attains a step size robustness score at least as high as ZO-SGD's. While this robustness is valuable, it comes at the *memory cost* of maintaining additional optimizer states for every parameter. In the next section, we show how to recover much of this *robustness without the overhead* of full coordinate-wise adaptivity.

5 MEAZO: Adaptivity Robustness with ZO-SGD Cost

MEAZO (Algorithm 1) is motivated by the observations of Section 4. While adaptive methods such as Adam improve robustness to step size selection, their coordinate-wise statistics become largely uninformative in the ZO regime. At the same time, maintaining first and second moment estimates for every parameter incurs significant memory overhead. MEAZO addresses this mismatch by replacing coordinate-wise adaptation with a single global adaptive statistic.

Rather than tracking per-parameter second moments, MEAZO maintains a *single global statistic* of the squared averaged projected gradient norm $v_t \approx \mathrm{EMA}(g_t^2)$, where $g_t = \frac{1}{q}\sum_{i=1}^{q} \Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_i) \in \mathbb{R}$. This statistic captures the overall scale of the projected-gradients along the training and is used to adapt the global step size, while avoiding the cost of storing coordinate-wise moments.

Update Rule. The MEAZO update is given by

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \frac{\eta}{\sqrt{\hat{v}_t + \zeta}} \left(\frac{1}{q} \sum_{i=1}^{q} \Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_i) \mathbf{u}_i \right), \tag{7}$$

where $\hat{v}_t$ denotes the bias-corrected EMA of g_t^2 and $\zeta > 0$ is a numerical-stability constant. In contrast to Adam, which requires storing a first- and second moment estimate for every parameter, MEAZO maintains only a single scalar state. Consequently, it retains the memory footprint of ZO-SGD while preserving the robustness benefits of adaptive step size scaling. The role of the first moment in adaptive ZO methods, and its omission in MEAZO, are discussed in Appendix D.6.

Figure 2a supports MEAZO's design: as the dimension increases, the coordinates of ZO-Adam's second moment vector collapse toward a single value, which is accurately approximated by MEAZO's scalar second moment estimate.

5.1 Theoretical Guarantees

We consider the setup in Section 2 under the following assumptions, which are compared to those used in prior work in Appendix G.1.

Assumption 5.1 (Uniform perturbations independent from the mini-batch)**.** For each iteration $t = 1, \ldots, T$, the mini-batch ξ_t and the perturbation $\mathbf{u}_t$ are sampled independently. Moreover, the perturbations are uniformly sampled over the unit $\mathbb{R}^d$ sphere, that is $\mathbf{u}_t \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ for all $t \in \mathbb{N}^*$ and $i = 1, \ldots, q$, where $\mathbb{S}^{d-1} := \{\mathbf{u} \in \mathbb{R}^d : \|\mathbf{u}\| = 1\}$. The perturbations satisfy $\mathbb{E}[\mathbf{u}_t] = 0$, $\mathbb{E}[\mathbf{u}_t \mathbf{u}_t^\top] = \frac{1}{d} I_d$ and $\|\mathbf{u}_t\|^2 = 1$.

Algorithm 1 MEAZO Algorithm

Input: initial parameters $\mathbf{x}_0$, objective function f, step size η, decay rate β, number of noise samples q, distribution P, number of iterations T
Output: solution $\mathbf{x}_T$
Initialize: v_0
for $t \in \{1, ..., T\}$ **do**
  sample mini-batch ξ, sample q noise vectors $\mathbf{u}_i \sim P$
  compute q projected gradient terms $\Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_i, \xi)$ via (2)
  $g \leftarrow \frac{1}{q}\sum_{i=1}^{q} \Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_i, \xi)$
  $v_t \leftarrow \beta v_{t-1} + (1-\beta) g^2$
  $\hat{v} \leftarrow \frac{v_t}{1-\beta^t}$ ▷ bias correction (if $v_0 = 0$)
  $\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} - \frac{\eta}{\sqrt{\hat{v}+\zeta}} \left(\frac{1}{q}\sum_{i=1}^{q} \Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_i, \xi)\, \mathbf{u}_i\right)$ ▷ update parameters
end for
return $\mathbf{x}_T$

Assumption 5.2 (Lower boundedness of the objective)**.** There exist $F_\star > -\infty$ such that for all $\mathbf{x} \in \mathbb{R}^d$, $F(\mathbf{x}) \geq F_\star$. Consequently, the smoothed objective is also lower bounded, that is $F_\varepsilon(\mathbf{x}) := \mathbb{E}_{\mathbf{v}\sim\mathrm{Unif}(\mathbb{B}^d)}[F(\mathbf{x}+\varepsilon\mathbf{v})] \geq F_\star$ for all $\mathbf{x} \in \mathbb{R}^d$, where $\mathbb{B}^d := \{\mathbf{v} \in \mathbb{R}^d : \|\mathbf{v}\| \leq 1\}$.

Let $q = 1$, and define the unscaled directional finite difference $\delta_t := \frac{f(\mathbf{x}_t+\varepsilon\mathbf{u}_t;\xi_t) - f(\mathbf{x}_t-\varepsilon\mathbf{u}_t;\xi_t)}{2\varepsilon}$. The projected gradient for a uniform perturbation is then

$$g_t := d\delta_t = d\frac{f(\mathbf{x}_t+\varepsilon\mathbf{u}_t;\xi_t) - f(\mathbf{x}_t-\varepsilon\mathbf{u}_t;\xi_t)}{2\varepsilon} \quad . \tag{8}$$

Remark 5.3. In (8), it is important not to forget the rescaling factor d for uniform perturbations.

Assumption 5.4 (Bounded directional finite differences)**.** There exists a dimension-independent constant $B > 0$ such that, almost surely, $|\delta_t| \leq B$ for all $t \in \mathbb{N}^*$. Let us also define the projected gradient bound with explicit dimension dependency $R_d := \sqrt{\zeta} + dB$. Then, by (8),

$$0 \leq g_t^2 \leq d^2B^2 = \left(R_d - \sqrt{\zeta}\right)^2, \qquad \forall t \in \mathbb{N}^* \quad . \tag{9}$$

We also assume that the L-smoothness Assumption 2.1 holds for the stochastic loss $f(\cdot;\xi)$. Consequently, F and F_ε are also L-smooth.

We then develop formal convergence guarantees for MEAZO (Algorithm 1) for the ZO estimator (3).

Theorem 5.5 (Convergence of MEAZO)**.** *Suppose that Assumptions 5.1, 5.2, 2.1 and 5.4 hold. Let $q = 1$, $\eta > 0$, $\varepsilon > 0$, and $\beta \in (0,1)$. Let $\{\mathbf{x}_t\}_{t\in[T]}$ be the iterates generated by Algorithm 1, then*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\,\|\nabla F(\mathbf{x}_t)\|^2 \leq \frac{4R_d\,[F(\mathbf{x}_1) - F_\star]}{\eta T} + \frac{4R_d L\varepsilon^2}{\eta T} + \left(\frac{8R_d^2}{\sqrt{1-\beta}} + \frac{2R_d L\eta}{1-\beta}\right)\left[\frac{1}{T}\log\left(1 + \frac{R_d^2}{(1-\beta)\zeta}\right) - \log(\beta)\right] + 2L^2\varepsilon^2 \quad . \tag{10}$$

The proof, adapted from Défossez et al. (2020), is deferred to Appendix G.

Corollary 5.6 (Convergence rate)**.** *Under the assumptions of Theorem 5.5, choose $\eta = \eta_0/\sqrt{T}$, $\varepsilon^2 = \varepsilon_0^2/\sqrt{T}$, $\beta = 1 - b/T$, where $\eta_0, b > 0$ and $T \geq 2b$. Consequently, MEAZO achieves a convergence rate of $\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\,\|\nabla F(\mathbf{x}_t)\|^2 = \mathcal{O}\left(\frac{d^2\log\left(1+d^2T/\zeta\right)}{\sqrt{T}}\right) = \widetilde{\mathcal{O}}\left(\frac{d^2}{\sqrt{T}}\right)$, where $\widetilde{\mathcal{O}}$ hides logarithmic factors.*

Remark 5.7. Both MEAZO and $\mathcal{R}$-AdaZO (Shu et al., 2025) achieve an $\mathcal{O}\left(T^{-1/4}\right)$ averaged gradient norm convergence rate up to logarithmic factors. In the variance-dominated setting, $\mathcal{R}$-AdaZO obtains an $\mathcal{O}(d^{3/4})$ dimension factor, whereas our analysis yields $\widetilde{\mathcal{O}}(d)$. Tightening our theoretical bound is left for future work.

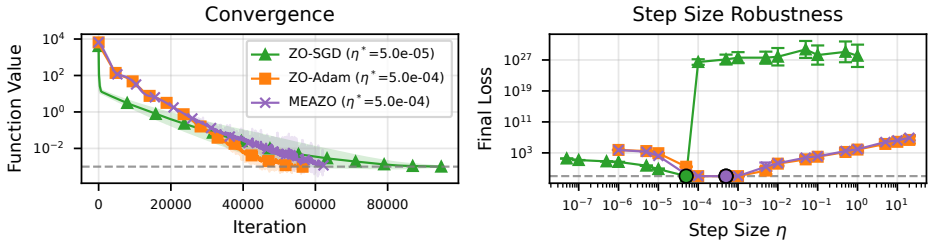


Figure 5: The performance of GD-ZO methods on homogeneous block quadratics under convergence analysis (left) and step size robustness evaluation (right). Robustness scores are ZO-SGD: 9.21×10^4, ZO-Adam: 1.03×10^5, MEAZO: 9.46×10^4.

Table 1: Evaluation metrics before and after 16-rank LoRA fine-tuning (L: Llama-3.2, Q: Qwen-3). Results are averaged over 3 seeds. Higher is better.

	XSum (ROUGE-L)			SST-2 (Accuracy)			SQuAD (F1)		
Method	L1B	Q0.6B	Q8B	L1B	Q0.6B	Q8B	L1B	Q0.6B	Q8B
Base	13.62	15.09	18.60	49.08	50.69	93.00	24.63	31.76	31.50
ZO-SGD	$21.99_{\pm 0.07}$	$20.08_{\pm 0.12}$	$26.96_{\pm 0.28}$	$87.96_{\pm 0.95}$	$87.19_{\pm 1.09}$	$93.73_{\pm 0.46}$	$76.16_{\pm 0.40}$	$69.30_{\pm 0.84}$	$82.69_{\pm 0.26}$
ZO-Adam	$22.36_{\pm 0.07}$	$20.08_{\pm 0.09}$	$27.45_{\pm 0.10}$	$88.88_{\pm 0.43}$	$87.27_{\pm 0.50}$	$93.46_{\pm 0.52}$	$76.24_{\pm 0.43}$	$68.91_{\pm 0.83}$	$83.16_{\pm 0.18}$
RAdaZO	$22.39_{\pm 0.17}$	$20.12_{\pm 0.05}$	$27.34_{\pm 0.19}$	$88.42_{\pm 0.80}$	$87.31_{\pm 0.38}$	$93.69_{\pm 0.61}$	$74.54_{\pm 0.98}$	$69.50_{\pm 0.56}$	$82.88_{\pm 0.21}$
MEAZO	$22.39_{\pm 0.19}$	$20.08_{\pm 0.04}$	$27.31_{\pm 0.27}$	$88.38_{\pm 1.13}$	$87.04_{\pm 1.50}$	$93.39_{\pm 0.52}$	$75.77_{\pm 0.41}$	$68.72_{\pm 0.33}$	$82.94_{\pm 0.87}$

6 Experimental Results

6.1 Synthetic Problems

Setup. We first consider a synthetic quadratic problem $F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x}$ used in section D of Orvieto & Gower (2025). The Hessian $\mathbf{H} \in \mathbb{R}^{9\times 9}$ has a homogeneous 3×3 diagonal block structure (see Appendix B.2). We perform a grid search for the step size and minimize this function with ZO-SGD, ZO-Adam and MEAZO[3].

Results. Recent works (Zhang et al., 2024b; Orvieto & Gower, 2025) show that adaptive optimizers do not outperform SGD in the presence of a homogeneous loss landscape. This phenomenon is verified in Fig. 5 (left), where we observe only a slightly faster convergence of ZO adaptive methods compared to ZO-SGD. However, we observe a significant improvement in step size robustness (Fig. 5, right). Although ZO-SGD is slightly better than adaptive methods in the underconverged regime of small step size, it diverges dramatically from a step size twice as big as the optimal one. In contrast, ZO-Adam or MEAZO remain stable across a much wider range of step sizes. We study the latter in more detail in Section 6.2.

6.2 LLM Fine-tuning

Setup. We fine tune two model families, Llama 3 (Grattafiori et al., 2024) (L: Llama-3.2) and Qwen 3 (Yang et al., 2025) (Q: Qwen-3), across diverse datasets including SST-2, XSum, and SQuAD. We compare MEAZO against ZO-SGD (MeZO), ZO-Adam (Chen et al., 2019), $\mathcal{R}$-AdaZO (Shu et al., 2025), and FZOO (Dang et al., 2025). Full experimental details are provided in Appendix C. Additional ViT-B/16 (Wu et al., 2020) and Ministral 14B (Liu et al., 2026) results appear in Appendix D.

LoRA finetuning. Tables 1 and 2 report final performance and step size robustness, respectively, for LoRA fine-tuning across models and optimizers. We find that a well-tuned ZO-SGD baseline remains competitive with adaptive methods in terms of final performance, while *MEAZO and ZO-Adam consistently achieve higher robustness*, supporting our central claim. Additional learning and robustness curves are provided in Appendix D.3.

[3]With an abuse of notation, as we use the true gradients on the full synthetic dataset.

Table 2: Step-size robustness score for 16-rank LoRA fine-tuning (L: Llama-3.2, Q: Qwen-3).

Method	XSum L1B	XSum Q0.6B	XSum Q8B	SST-2 L1B	SST-2 Q0.6B	SST-2 Q8B	SQuAD L1B	SQuAD Q0.6B	SQuAD Q8B
ZO-SGD	1	1.2	0.37	3.9	4.4	1e+02	5	11	5.2
ZO-Adam	1.1	1.6	0.55	4.9	5.4	1.5e+02	5.7	12	8.5
MEAZO	1.1	1.6	0.55	4.6	4.8	1.5e+02	5.6	12	8.7

Table 3: FZOO vs. MEAZO-Fwd on SST-2 (Llama-3.2 1B) LoRA fine-tuning.

Method	q	η^*	Accuracy (%)	Robustness
FZOO	2	5e−4	$49.81_{\pm 1.40}$	0.000
FZOO	8	1e−3	$89.56_{\pm 1.35}$	1.709
MEAZO-Fwd	1	3e−4	$88.61_{\pm 1.77}$	0.794
MEAZO-Fwd	2	3e−4	$89.11_{\pm 0.09}$	1.064
MEAZO-Fwd	8	3e−4	$90.71_{\pm 0.28}$	1.729

Table 4: Full-model fine-tuning of Llama-3.2 1B on SST-2. 3 seeds, latency and peak GPU memory measured over 10 batches (batch size 8) and average sequence length of 108.5 tokens.

Method	Accuracy (%)	Avg. latency (s/step)	Peak memory (GB)	Robustness score
ZO-SGD	88.99 ± 0.41	1.69	10.21	56.41
ZO-Adam	88.99 ± 0.82	1.74	23.03	57.11
MEAZO	89.64 ± 0.94	1.75	10.21	57.47

Comparison with FZOO. We also benchmark against FZOO using a controlled variant (MEAZO-Fwd) that matches FZOO's one-sided shared-baseline estimator and forward-pass budget. We find that MEAZO's cross-step adaptive normalization yields greater stability at small q and improved robustness at $q = 8$ (Table 3). Notably, FZOO is not applicable at $q = 1$ and becomes unstable at $q = 2$, where performance collapses to near chance. Additional analysis, learning curves, and robustness sweeps are provided in Appendix D.8.

Full model finetuning. Full-model fine-tuning can outperform parameter-efficient fine-tuning methods (PEFT) (Malladi et al., 2023). Table 4 shows that the insights from LoRA fine-tuning generalize: comparable accuracy subject to step-size tuning and improved robustness for adaptive methods, while MEAZO maintains the latency and memory footprint of ZO-SGD. Larger scale models with longer sequences exacerbate this computational advantage (reported in Appendix D.4).

7 Discussion and Future Work

Related works are discussed in Appendix A. We showed that well-tuned ZO-SGD can match or outperform adaptive methods like ZO-Adam, due to the lack of coordinate-wise structure in ZO gradients in high dimensions. This is especially consequential for on-device learning: ZO methods offer low memory requirements and can leverage optimized inference kernels, but heterogeneous data across devices makes step-size tuning difficult. MEAZO addresses this tension through scalar adaptation, providing ZO-Adam-level robustness to step-size selection with the memory footprint of ZO-SGD. Future work includes sharpening the convergence guarantees of MEAZO.

Recent work by Song et al. (2026) studies the edge of stability (EoS) of ZO methods and shows that it is controlled by global curvature quantities related to the Hessian trace (as opposed to its top eigenvalue for FO methods). An interesting direction for future work would be to connect this perspective with our robustness analysis and investigate whether adaptive methods that track such global curvature statistics can further improve upon MEAZO.

AI USE STATEMENT

In this work, we used generative AI tools for polishing the writing, especially to shorten and improve the clarity of certain paragraphs, as well as to debug LaTeX commands. We did not use generative AI tools for literature review, research ideation, experiment design or execution, generating datasets, or proving mathematical claims. All AI-assisted content was reviewed by the authors, and the manuscript was independently reviewed by at least two of the authors. We take full responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of all experimental setups, datasets, hyperparameters, system prompts, etc in the appendix, together with additional figures, ablations, and optimal step size tables. For most experiments, we evaluate all methods across three random seeds and perform coarse and then fine step size sweeps to ensure robust comparisons. We also report latency and memory measurements to provide a complete characterization of the computational cost of each method.

REFERENCES

Albert S Berahas, Liyuan Cao, Krzysztof Choromanski, and Katya Scheinberg. A theoretical and empirical comparison of gradient approximations in derivative-free optimization. *Foundations of Computational Mathematics*, 22(2):507–560, 2022.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in neural information processing systems*, 32, 2019.

Sizhe Dang, Yangyang Guo, Yanjun Zhao, Haishan Ye, Xiaodong Zheng, Guang Dai, and Ivor Tsang. Fzoo: Fast zeroth-order optimizer for fine-tuning large language models towards adam-scale speed. *arXiv preprint arXiv:2506.09034*, 2025.

Alexandre Défossez, Léon Bottou, Francis Bach, and Nicolas Usunier. A simple convergence proof of adam and adagrad. *arXiv preprint arXiv:2003.02395*, 2020.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*, 2004.

Ge Gao, Alexey Taymanov, Eduardo Salinas, Paul Mineiro, and Dipendra Misra. Aligning llm agents by learning latent preference from user edits. *Advances in Neural Information Processing Systems*, 37:136873–136896, 2024.

Xiang Gao, Bo Jiang, and Shuzhong Zhang. On the information-adaptive variants of the admm: an iteration complexity perspective. *Journal of Scientific Computing*, 76(1):327–363, 2018.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

Yusu Hong and Junhong Lin. On convergence of adam for stochastic optimization under relaxed assumptions. *Advances in Neural Information Processing Systems*, 37:10827–10877, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. LoRA: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Shuoran Jiang, Qingcai Chen, Youcheng Pan, Yang Xiang, Yukang Lin, Xiangping Wu, Chuanyi Liu, and Xiaobao Song. Zo-adamu optimizer: Adapting perturbation by the momentum and uncertainty in zeroth-order optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38 (16), pp. 18363–18371, 2024.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

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

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *ArXiv*, abs/1808.08745, 2018.

Parvin Nazari, Davoud Ataee Tarzanagh, and George Michailidis. Adaptive first-and zeroth-order methods for weakly convex stochastic optimization problems. *arXiv preprint arXiv:2005.09261*, 2020.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

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

Antonio Orvieto and Robert Gower. In search of adam's secret sauce. *arXiv preprint arXiv:2505.21829*, 2025.

Yan Pan and Yuanzhi Li. Toward understanding why adam converges faster than sgd for transformers. *arXiv preprint arXiv:2306.00204*, 2023.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2011.

Yao Shu, Qixin Zhang, Kun He, and Zhongxiang Dai. Refining adaptive zeroth-order optimization at ease. *arXiv preprint arXiv:2502.01014*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1170.

Minhak Song, Liang Zhang, Bingcong Li, Niao He, Michael Muehlebach, and Sewoong Oh. Zeroth-order optimization at the edge of stability. *arXiv preprint arXiv:2604.14669*, 2026.

Akiyoshi Tomihari and Issei Sato. Understanding why adam outperforms sgd: Gradient heterogeneity in transformers. *arXiv preprint arXiv:2502.00213*, 2025.

Bohan Wang, Jingwen Fu, Huishuai Zhang, Nanning Zheng, and Wei Chen. Closing the gap between the upper bound and lower bound of adam's iteration complexity. *Advances in Neural Information Processing Systems*, 36:39006–39032, 2023.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *Journal of Machine Learning Research*, 21(219):1–30, 2020.

Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Haishan Ye and Luo Luo. Why does adaptive zeroth-order optimization work? *arXiv preprint arXiv:2602.01627*, 2026.

Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

Qi Zhang, Yi Zhou, and Shaofeng Zou. Convergence guarantees for RMSProp and adam in generalized-smooth non-convex optimization with affine noise variance. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=QIzRdjIWnS.

Yihua Zhang, Pingzhi Li, Junyuan Hong, Jiaxiang Li, Yimeng Zhang, Wenqing Zheng, Pin-Yu Chen, Jason D Lee, Wotao Yin, Mingyi Hong, et al. Revisiting zeroth-order optimization for memory-efficient llm fine-tuning: A benchmark. In *International Conference on Machine Learning*, pp. 59173–59190. PMLR, 2024a.

Yushun Zhang, Congliang Chen, Tian Ding, Ziniu Li, Ruoyu Sun, and Zhiquan Luo. Why transformers need adam: A hessian perspective. *Advances in neural information processing systems*, 37:131786–131823, 2024b.

Yanjun Zhao, Sizhe Dang, Haishan Ye, Guang Dai, Yi Qian, and Ivor Tsang. Second-order fine-tuning without pain for llms: A hessian informed zeroth-order optimizer. In *International Conference on Learning Representations*, volume 2025, pp. 43496–43520, 2025.

A Related Work

A.1 Zeroth-Order Gradient Estimation

Theoretical foundations of ZO optimization were established in early works such as (Ghadimi & Lan, 2013) and (Nesterov & Spokoiny, 2017), which analyzed convergence properties of ZO-SGD using Gaussian perturbations, or the convergence in the online setting with perturbations sampled uniformly from the unit sphere (Flaxman et al., 2004). Recently, MeZO (Malladi et al., 2023) introduced a memory-efficient variant of ZO-SGD by storing scalar seeds instead of sampled perturbations, significantly reducing memory usage.

Comprehensive benchmarking of ZO methods has been conducted by (Zhang et al., 2024a) to evaluate their empirical performance across tasks, highlighting trade-offs between estimator variance, convergence speed, and memory efficiency.

A.2 Adaptive Zeroth-Order Methods

To enhance convergence and robustness, adaptive ZO algorithms have incorporated techniques inspired by first-order optimization. ZO-AdaMM (Chen et al., 2019) and ZEMA (Nazari et al., 2020) introduced adaptive step sizes based on first and second moment estimates of the gradient approximation. However, these methods did not demonstrate theoretical convergence improvements over ZO-SGD in convex and non-convex constrained settings.

Recent advances have focused on refining the use of momentum and variance reduction. ZO-AdaMU (Jiang et al., 2024) introduced adaptive momentum for perturbations, smoothing both gradient estimates and parameter updates. FZOO (Dang et al., 2025) jointly improved ZO gradient estimation, by combining Rademacher perturbations, and step size adaptation, using the standard deviation of the batch losses. $\mathcal{R}$-AdaZO (Shu et al., 2025) leveraged variance-reduced gradient information embedded in the momentum to refine second moment estimates, leading to improved convergence on synthetic data.

B Synthetic Quadratic Experimental Setup

B.1 Second moment monitoring

Objective functions. The experiment uses a block-structured quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{H}\mathbf{x}$, where $\mathbf{H}$ is a block-diagonal Hessian matrix. For a dimension d, the Hessian is constructed from $\sqrt{d}$ blocks of size $\sqrt{d}$ each. We consider the heterogeneous configuration: each block contains consecutive eigenvalues centered around exponentially increasing values (logarithmically spaced from 1 to 1000). For more details, we refer to Appendix D of Orvieto & Gower (2025) for the case of $d = 9$ that we use below and extend here for larger dimensions. This creates blocks with different curvature, which helps to visualize second moment coordinate-wise variability of Adam. This structure mimics real optimization landscapes where different parameter groups have different sensitivities.

Effect of the problem dimension. In Figure 2a we track the evolution of the second moment $\mathbf{v}_t$ (vector or scalar value for MEAZO) across 3 optimizers: FO-Adam, ZO-Adam and MEAZO. We run the experiment for 5 different problem dimensions: $9, 25, 49, 100$ and 1024. All optimizers use the same step size $1\mathrm{e}{-4}$ until convergence to the same threshold $1\mathrm{e}{-3}$ with $q = 10$ noise samples for the ZO methods. For ZO-Adam, we also monitor $\|\nabla f(\mathbf{x}_t)\|^2 / q$ at iteration t.

Effect of the number of perturbations per step. In this experiment, we train with FO-Adam, ZO-Adam and MEAZO for a fixed step size $1\mathrm{e}{-4}$ on the same $d = 25$ dimensional quadratic heterogeneous quadratic problem until convergence to the same threshold $1\mathrm{e}{-3}$. In Figure 2b we plot the distribution of the second moment values of FO- and ZO-Adam and scalar second moment of MEAZO, at step $t = 5000$ for different values of q. Even if FO- and ZO-Adam's mean($\mathbf{v}_t$) is getting closer, ZO-Adam is not capable of recovering FO-Adam's second moment distribution even for $q = 1000 \times d$, mostly because q is still not large enough and because the ZO gradient is a biased estimate of the true gradient.

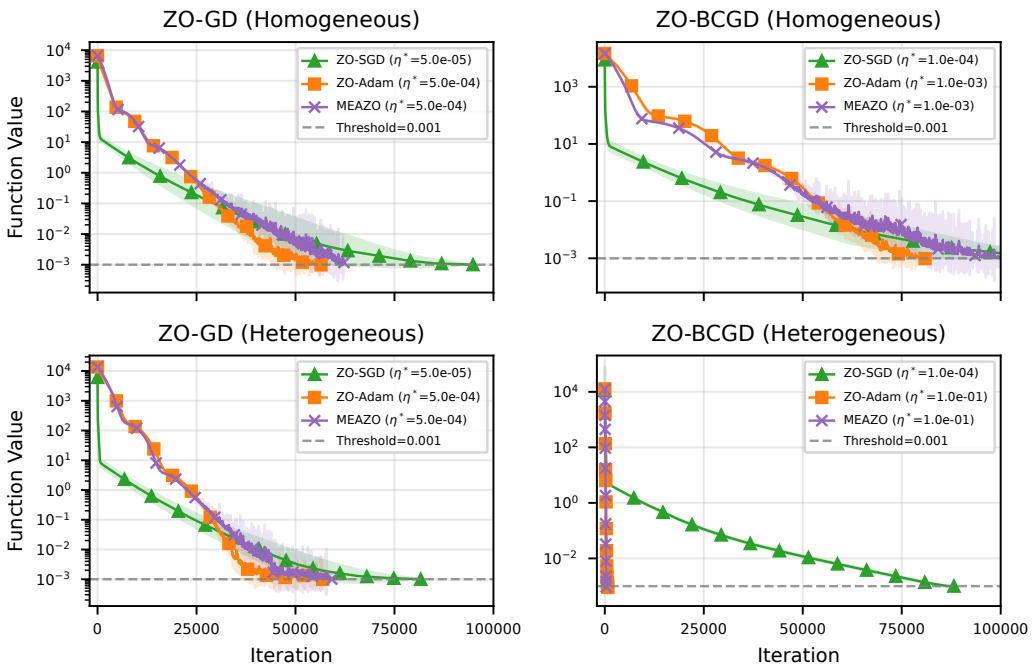


Figure 6: Convergence of ZO methods on block homogeneous (top) vs heterogeneous (bottom) quadratics.

B.2 Homogeneous vs Heterogeneous low-dimensional setting

We exactly reproduce the problem of appendix D of Orvieto & Gower (2025). The only difference being that we do not introduce random batch samples stochasticity by sampling rows of the design matrix. Thus, the only source of randomness lies in the zeroth-order gradients approximation. To find the best step size for each run, we perform step size tuning over the grid $\{1\mathrm{e}{-6}, 5\mathrm{e}{-6} \ldots, 1\mathrm{e}{-1}, 5\mathrm{e}{-1}\}$, with 10 different seeds. For each of the displayed run, we display the average run over the 10 seeds along with a plus/minus standard deviation envelope (in log space). Curves are smoothed over a window of 10 step for better display of ZO-Adam and MEAZO.

On the one hand, in the homogeneous case in Figure 6 (top), one can notice the similar performance of all three optimizers. On the other hand, Figure 6 (bottom) highlights the acceleration of the optimization with MEAZO, resp. ZO-Adam, when rescaling the step size per block, resp. per parameter within a block, in the presence of landscape heterogeneity.

In Figure 7, we show the better robustness around the optimal step size of ZO adaptive methods, *i.e.,* ZO-Adam and MEAZO, compared to ZO-SGD. In the heterogeneous setting, this robustness of ZO adaptive methods is even amplified by applying block-coordinate updates, as shown by way flatter curves for ZO-Adam and MEAZO in Figure 7 (bottom right) compared to Figure 7 (bottom left), than for ZO-SGD.

C LLM Fine-tuning Experimental Setup

C.1 Training Compute Resources and Hyperparameters Details

Unless stated otherwise, all models are trained for $T = 10\,000$ steps with a batch size of 8 on single NVIDIA A100/H100 GPUs equipped with 80GB of memory. We employ a constant step size schedule throughout training, without applying weight decay or gradient norm clipping. For adaptive optimization methods, we use the default hyperparameters: $\beta_1 = 0.9$ and $\beta_2 = 0.999$. For zeroth-order (ZO) optimization methods, we adopt a fixed perturbation scale of $\varepsilon = 1\mathrm{e}{-6}$ and use a single noise sample per mini-batch ($q = 1$).

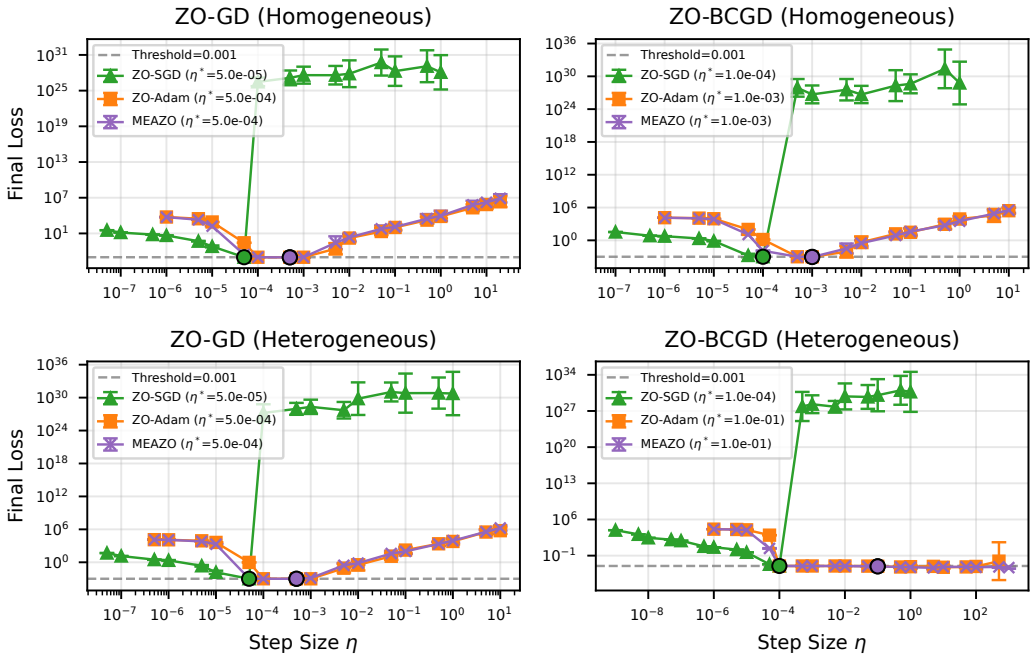


Figure 7: Step size robustness of ZO methods on block homogeneous (top) vs heterogeneous (bottom) quadratics.

C.2 Step Size Sweeps

To ensure a fair comparison across optimization methods, we conduct extensive step size sweeps to identify the optimal step size for each optimizer over two stages.

Coarse grid. We first sweep the step size over a coarse grid: $\eta \in \{1\text{e}{-6}, 5\text{e}{-6}, \ldots, 1\text{e}{-1}\}$, running each configuration across 3 random seeds. We then select the step size that achieves the lowest average validation loss.

Fine grid. Let $\eta^\star$ be the best step size from the coarse sweep and let η^-, η^+ be its adjacent coarse neighbors (extending the coarse grid by one virtual neighbor at the boundary if needed). We then enumerate candidate step sizes with integer mantissas within the bracket(s) $(\eta^-, \eta^\star)$ and $(\eta^\star, \eta^+)$, *i.e.*, values of the form $\eta = m \times 10^k$ with $m \in \{1, \ldots, 9\}$ whose magnitude lies within the bracket. When a bracket crosses a decade boundary, we enumerate the remaining integer-mantissa values in the lower decade and the leading values in the upper decade.

Table 5 presents an estimation of the computational cost for our LoRA finetuning setting.

Table 6 summarizes the best-performing step sizes for LoRA fine-tuning.

Remark C.1. As predicted by our theory, ZO-Adam and MEAZO behave extremely similarly in high dimension and require the same optimal learning rate in almost all cases. We also observe a similar behavior as in FO finetuning: ZO-Adam (and also MEAZO) can use larger learning rates than ZO-SGD as observed in Pan & Li (2023).

C.3 Classification Training Loss

Zhang et al. (2024a) formulate classification for LLM fine-tuning using a multiclass contrastive objective: for each input, the model is paired with *every* verbalized label, computes an average log-probability score over the label tokens, and these scores are used as logits in a softmax cross-entropy loss. This explicitly contrasts the true label against all incorrect ones but requires evaluating C candidate sequences during training.

Model	Dataset	1 Run (h)	1 Run (FLOPs)	Full Tuning (h)	Full Tuning (FLOPs)
Qwen3 0.6B	SST-2	0.88	3.39×10^{14}	47.6	1.83×10^{16}
Qwen3 0.6B	SQuAD	1.15	7.82×10^{15}	62.1	4.22×10^{17}
Qwen3 0.6B	XSum	7.54	2.78×10^{16}	407.2	1.50×10^{18}
Llama-3.2 1B	SST-2	1.11	4.29×10^{15}	60.1	2.31×10^{17}
Llama-3.2 1B	SQuAD	1.66	1.80×10^{16}	89.4	9.73×10^{17}
Llama-3.2 1B	XSum	11.68	5.33×10^{16}	630.6	2.88×10^{18}
Qwen3 8B	SST-2	4.75	2.23×10^{16}	256.5	1.20×10^{18}
Qwen3 8B	SQuAD	8.85	1.09×10^{17}	478.0	5.90×10^{18}
Qwen3 8B	XSum	58.33	2.82×10^{17}	3,149.9	1.52×10^{19}

Table 5: Cost of our dense step-size sweep: $\approx$ 18 candidate step sizes evaluated across 3 seeds, yielding ≈ 54 runs and a total tuning cost of $\approx 54\times$ a single run.

Table 6: Optimal step sizes for 16-rank LoRA fine-tuning (L: Llama-3.2, Q: Qwen-3).

	XSum			SST-2			SQuAD		
Method	L1B	Q0.6B	Q8B	L1B	Q0.6B	Q8B	L1B	Q0.6B	Q8B
ZO-SGD	1e−4	6e−5	2e−4	1e−4	4e−5	1e−4	9e−5	5e−5	1e−4
ZO-Adam	2e−4	2e−4	4e−4	3e−4	2e−4	6e−4	2e−4	3e−4	4e−4
RAdaZO	5e−5	5e−5	1e−4	6e−5	6e−5	1e−4	7e−5	6e−5	8e−5
MEAZO	2e−4	2e−4	4e−4	3e−4	3e−4	6e−4	2e−4	3e−4	4e−4

In our setup, we instead treat classification (SST-2) as straightforward causal language modeling (CLM). During training, we provide only the ground-truth continuation (the correct verbalized label) and maximize its next-token likelihood (no negative labels and no classification loss). However, at evaluation time, we adopt the same procedure: for each class, we compute the same average log-probability score for its verbalized label and select the label with the highest score. Despite the simpler training objective, we find minimal degradation in final model performance.

C.4 Models

We download all pre-trained models from HuggingFace and load them in full precision (FP32) except for Ministral 14B loaded in BF16. For language tasks we consider Llama-3.2 1B (Grattafiori et al., 2024), Qwen-3 0.6B and 8B (Yang et al., 2025) and Ministral-3 14B Instruct (Liu et al., 2026). For the vision task, we finetune ViT-B/16 (Wu et al., 2020). The training configurations detailed below are summarized in Table 7.

LoRA fine-tuning. For LoRA fine-tuning (Hu et al., 2022), we use a rank of $r = 16$, set the scaling factor to $\alpha = 16$, and disable dropout. LoRA layers are attached to all query (Q) and value (V) projection matrices within each transformer decoder block. Our implementation is based on Hugging Face PEFT's library[4].

Full model fine-tuning. For full model fine-tuning of LLama-3.2 1B on SST-2, we set the batch size to 8 and $q = 1$. For the memory and latency reporting over all other models and datasets, we employ a least demanding setting with batch size set to 1 and $q = 1$.

C.5 Dataset Pre-processing

For all datasets, we utilize their publicly available implementations from HuggingFace. For each model, we format every data point using its corresponding tokenizer's chat template.

[4] https://huggingface.co/docs/peft

Table 7: Summary of training configurations. For all experiments, LoRA uses rank $r = 16$.

Method	Llama-3.2 1B	Qwen-3 0.6B	Qwen-3 8B	Ministral-3 14B	ViT-B/16
Base model					
#Total parameters	1'235'814'400	596'049'920	8'190'735'360	13'945'031'680	85'827'109
LoRA					
#Trainable parameters	1'703'936	2'293'760	7'667'712	11'403'264	618'277
%Trainable parameters	0.1377%	0.3834%	0.0935%	0.0817%	0.7152%

SST-2. The SST-2 dataset (Socher et al., 2013) is pre-divided into training, validation, and test splits. However, since the original test set lacks labels, we reassign the original validation set as our test set. To construct a new validation set, we split the original training data (comprising 67 349 samples) into training and validation subsets using a 95/5 ratio. The chat template employed for training is illustrated in Box 1. For evaluation, we compute the accuracy on the full test set.

Box 1: SST-2 Chat Template

System Prompt

You are a sentiment analysis assistant. You will be given a user provided phrase or sentence, and your job will be to output terrible if it has negative sentiment or great if it has positive sentiment.

User Query

Sentence: `<$sentence>`

Assistant Response

Sentiment: `<$sentiment>`

XSum. The XSum (Narayan et al., 2018) dataset retains its original training, validation, and test splits. Prior to training, we filter out samples that exceed the predefined context length (e.g., 2048 tokens) after tokenization. This filtering process is tokenizer-dependent; using the Llama 3.2 tokenizer, we exclude 1,202 samples across all splits. We then select the first 50,000 training samples and the first 1,000 validation samples.The chat template used for training is shown in Box 2. For evaluation, we compute the ROUGE-L score on the full test set.

SQuAD. The SQuAD dataset (Rajpurkar et al., 2016) includes only training and validation splits. Following the approach used for SST-2, we reassign the validation split as the test set. The original training set contains 87 599 samples, which we divide into training and validation subsets using a 95/5 split. From these, we select the first 50 000 training samples and the first 1000 validation samples. The chat template used for training is shown in Box 3. For evaluation, we compute the F-1 score on the full test set.

Oxford-Pets. To extend our experiments to vision tasks, we use the Oxford-IIIT Pet dataset (Parkhi et al., 2012), which consists of 37 cat and dog breeds with roughly 200 images per class, and evaluate performance on the 37-way fine-grained classification task.

Box 2: XSum Chat Template

System Prompt

You are a writing assistant that helps with summarizing text. You write concise, one-sentence summaries based on a user-provided text or article.

User Query

Article: `<$article>`

Assistant Response

Summary: `<$summary>`

Box 3: SQuAD Chat Template

System Prompt

You are a question answering assistant. You will be given a user-provided context (paragraph or sentence), followed by a question. Your job will be to answer the question based on the paragraph or sentence.

User Query

Title: `<$title>`
Context: `<$context>`
Question: `<$question>`

Assistant Response

Answer: `<$answer>`

D ADDITIONAL EXPERIMENTS

D.1 COORDINATE-WISE VARIABILITY OF THE SECOND MOMENT: FO- VS ZO-ADAM

To further validate our theory, we first study the performance of LoRA fine-tuning of Llama-3.2 1B on XSum (with tuned step sizes). Figure 1 shows that, after proper step size tuning, ZO-SGD matches the performance of ZO-Adam on XSum, despite Adam being the dominant optimizer in FO scenarios. This suggests that coordinate-wise adaptation provides little benefit in ZO settings.

We also examine the second-moment estimates ($\mathbf{v}_t$) maintained by Adam at the final LoRA fine-tuning step ($t = 10000$) of Llama-3.2 1B on the XSum dataset. Table 8 reports the minimum, maximum, mean, standard deviation, and kurtosis of $\mathbf{v}_t$ across all parameters ($d = 1'703'936$). FO-Adam exhibits large variability and high kurtosis, reflecting meaningful per-coordinate differences that justify adaptive scaling. In contrast, ZO-Adam shows nearly uniform values with low variance and kurtosis, confirming that its adaptive mechanism is largely redundant when the gradient signal is isotropic.

Table 8: Statistics of the second moment $\mathbf{v}_t$ at $t = 10000$ for Llama-3.2 1B fine-tuning on XSum (seed 0).

Statistic	ZO-Adam	FO-Adam
Min.	1.39	6.68×10^{-19}
Max.	2.60	3.08×10^{-4}
Mean	1.87	1.02×10^{-6}
Stan. Dev.	1.11×10^{-1}	4.05×10^{-6}
Kurtosis	2.36×10^{-1}	3.23×10^{2}

D.2 Robustness Score Ablations

In Section 4, we introduced the robustness score $A_\tau(\ell)$ (re-stated below for clarity):

Definition D.1 (Robustness Score (Re-stated))**.** For a given training run with step size $\eta \in \mathbb{R}$ for $T \in \mathbb{N}^*$ steps, let $\ell(\eta) \coloneqq \min_{t \le T} \ell_{\text{val}}(\eta, t)$ denote the best validation loss attained when training with step size η, swept over a grid $\eta \in [\eta_{\min}, \eta_{\max}] \subset \mathbb{R}$. For a *ceiling* $\tau > 0$, the step size robustness score is

$$A_\tau(\ell) = \frac{1}{L} \int_{\eta_{\min}}^{\eta_{\max}} \max\big(\tau - \ell(\eta),\, 0\big)\, \mathrm{d} \log_{10} \eta \ , \tag{11}$$

where $L > 0$ is a *loss normalization constant* that puts the score in dimensionless, comparable units.

For each (dataset, model) combination, we chose τ to be the largest value that does not under-approximate any swept method's cup area due to boundary censoring, capped at the shared initial validation loss. This choice guarantees that every reported area in Table 2 is exact rather than a lower bound, but it is only one point on a continuum of reasonable ceilings. Here we ablate two simpler, more naive alternatives and check whether they preserve the paper's qualitative conclusion: adaptive ZO methods (ZO-Adam, MEAZO) are at least as robust as ZO-SGD, and often more so.

Setup shared by both baselines. For a given (dataset, model) combination, let $L = \min_{m,\eta} \ell_m(\eta)$ denote the best loss attained by any swept method, the same normalizer used in (11), and let ℓ_{init} denote the shared, pre-training (step-0) validation loss, identical across methods since no training has occurred yet. Both baselines below fix a *single formula* for τ as a function of a sweep percentage $p \in \{0\%, 10\%, \ldots, 100\%\}$ and apply it identically to every method in a combination. Because the formula is not chosen to avoid censoring, some methods' curves are censored well before others as p grows: whenever $\ell_m(\eta_{\min}) < \tau(p)$ or $\ell_m(\eta_{\max}) < \tau(p)$, the true cup extends past the swept step-size range and the reported area under-estimates the method's true robustness. We shade every such cell in the tables below, and every table is restricted to LoRA fine-tuning with $\ell(\eta)$ taken as each swept step size's best validation loss, matching the scope of Table 2.

Baseline 1: interpolating between L and ℓ_{init}. The first ablation (Table 9) sweeps τ linearly between the best swept loss and the initial validation loss:

$$\tau(p) = L + p \cdot (\ell_{\text{init}} - L), \qquad p \in \{0, 0.1, \ldots, 1\}\,. \tag{12}$$

When L is small relative to ℓ_{init} (as in SST-2, where classification loss falls to $L \approx 0.02$–0.05 while $\ell_{\text{init}} \approx 1$–$8$), the interpolated ceiling grows quickly in absolute terms, and *every* method ends up censored simultaneously well before the sweep completes (by $p = 30\%$ for Qwen-3 0.6B, and by $p = 60\%$ at the latest for every SST-2 combination). Once every curve is censored, the reported areas are all lower bounds computed over the same clipped range, and their relative order can no longer be trusted: for Qwen-3 0.6B on SST-2, ZO-Adam's reported score drops *below* ZO-SGD's from $p = 40\%$ onward (e.g., 130 vs. 140 at $p = 40\%$, 410 vs. 430 at $p = 100\%$), the only combination in either ablation where the adaptive-over-SGD ranking reverses. When L and ℓ_{init} are closer in scale instead (e.g., XSum), the sweep censors methods more gradually and in the order predicted by their robustness: ZO-Adam and MEAZO, whose curves stay near-optimal over a wider step-size range, are censored *before* ZO-SGD (e.g., Qwen-3 8B on XSum, from $p = 20\%$), while their scores remain above ZO-SGD's at every percentage regardless.

Baseline 2: a relative multiple of L. The second ablation (Table 10) instead scales the ceiling multiplicatively above L, independent of ℓ_{init}:

$$\tau(p) = (1 + p)\, L, \qquad p \in \{0, 0.1, \ldots, 1\}\,. \tag{13}$$

Because $\tau(p)$ is now tied to each combination's own L rather than to the (typically much larger) ℓ_{init}, the ceiling grows far more conservatively, and censoring is markedly rarer: SST-2 and SQuAD, censored almost everywhere under Baseline 1, are not censored at all here, and the adaptive-over-SGD ranking never reverses. The only two combinations censored under Baseline 2 are both on XSum: for Llama-3.2 1B all three methods are censored together from $p = 70\%$, since even ZO-SGD's curve there is wide enough that a τ of $1.7 \times L$ ($p = 70\%$) still leaves it under the ceiling

Table 9: Baseline 1 ablation: cup-area robustness score (best-loss) for 16-rank LoRA fine-tuning under a shared, ceiling $\tau(p) = \text{group_min} + p \cdot (\text{init} - \text{group_min})$, swept from $p = 0\%$ (each combo's best swept loss) to $p = 100\%$ (the shared initial validation loss). Each area is divided by that combo's group-min loss. Shaded cells indicate the ceiling at that percentage left- or right-censors that method's curve (the true cup extends beyond the swept LR range, so the reported area is a lower bound).

Method	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
XSum – Llama-3.2 1B (min=1.6271, init=2.7876)											
ZO-SGD	0	0.033	0.1	0.18	0.28	0.39	0.52	0.66	0.82	1	1.2
ZO-Adam	0	0.036	0.11	0.21	0.32	0.44	0.58	0.73	0.9	1.1	1.3
MEAZO	0	0.036	0.11	0.2	0.31	0.44	0.57	0.73	0.89	1.1	1.3
XSum – Qwen-3 0.6B (min=1.8048, init=5.3668)											
ZO-SGD	0	0.14	0.34	0.58	0.85	1.2	1.5	1.8	2.2	2.6	3
ZO-Adam	0	0.19	0.47	0.79	1.1	1.5	2	2.4	2.9	3.4	4
MEAZO	0	0.19	0.48	0.8	1.2	1.5	2	2.4	2.9	3.4	4
XSum – Qwen-3 8B (min=1.1771, init=5.8863)											
ZO-SGD	0	0.26	0.67	1.1	1.7	2.3	2.9	3.6	4.4	5.3	6.3
ZO-Adam	0	0.39	0.98	1.6	2.3	3	3.8	4.6	5.5	6.5	7.6
MEAZO	0	0.39	0.98	1.6	2.3	3	3.8	4.6	5.5	6.5	7.6
SST-2 – Llama-3.2 1B (min=0.0450, init=1.2100)											
ZO-SGD	0	2.4	6.3	11	16	20	25	30	35	40	45
ZO-Adam	0	3	7.7	13	18	24	30	37	44	51	58
MEAZO	0	2.9	7.3	12	18	24	30	36	43	50	58
SST-2 – Qwen-3 0.6B (min=0.0393, init=8.0204)											
ZO-SGD	0	25	56	93	1.4e+02	1.9e+02	2.4e+02	2.8e+02	3.3e+02	3.8e+02	4.3e+02
ZO-Adam	0	27	60	95	1.3e+02	1.7e+02	2.1e+02	2.6e+02	3e+02	3.6e+02	4.1e+02
MEAZO	0	24	56	94	1.4e+02	1.8e+02	2.2e+02	2.7e+02	3.2e+02	3.8e+02	4.3e+02
SST-2 – Qwen-3 8B (min=0.0234, init=7.5706)											
ZO-SGD	0	23	54	93	1.4e+02	1.9e+02	2.6e+02	3.4e+02	4.2e+02	5.1e+02	6.1e+02
ZO-Adam	0	38	86	1.4e+02	2e+02	2.7e+02	3.4e+02	4.1e+02	4.9e+02	5.8e+02	6.8e+02
MEAZO	0	40	88	1.4e+02	2e+02	2.7e+02	3.4e+02	4.2e+02	5e+02	5.8e+02	6.8e+02
SQuAD – Llama-3.2 1B (min=0.2402, init=1.6380)											
ZO-SGD	0	0.42	1.1	2	3	4.1	5.3	6.6	7.9	9.2	10
ZO-Adam	0	0.54	1.4	2.4	3.5	4.8	6.1	7.5	9	11	12
MEAZO	0	0.52	1.4	2.4	3.5	4.7	6	7.4	9	11	12
SQuAD – Qwen-3 0.6B (min=0.2234, init=7.8602)											
ZO-SGD	0	3.5	8.3	14	21	29	38	48	60	72	86
ZO-Adam	0	3.6	8.5	14	21	28	35	43	51	60	70
MEAZO	0	3.6	8.6	15	21	28	36	43	51	60	70
SQuAD – Qwen-3 8B (min=0.1403, init=7.7015)											
ZO-SGD	0	3.7	9.1	16	24	33	42	52	63	76	92
ZO-Adam	0	6.2	14	23	32	42	53	64	76	90	1.1e+02
MEAZO	0	6.3	14	23	32	42	53	64	77	90	1.1e+02

at the sweep boundary; for Qwen-3 8B only ZO-Adam and MEAZO are censored, from $p = 60\%$, since their curves are flatter than ZO-SGD's and a $1.6 \times L$ ceiling is still not tight enough to reach their boundary. In both cases the reported (censored) scores for the affected methods nonetheless remain above ZO-SGD's throughout, since the true, uncensored area can only be larger still.

Takeaways. A shared ceiling that is allowed to rise all the way to a combination's initial loss (Baseline 1) is the more dangerous choice: it censors every method at once precisely on the tasks where $L \ll \ell_{\text{init}}$ (SST-2), and once every curve is clipped identically, the resulting comparison can invert. A shared ceiling tied only to L (Baseline 2) avoids that failure mode, at the cost of frequently under-estimating, rather than mis-ranking, the most robust methods, since it has no principled upper bound and L itself is not always small enough for $2L$ to reach the sweep boundary. In every uncensored cell of either table, ZO-Adam and MEAZO's cup areas are at least as large as ZO-SGD's, so the paper's central claim "adaptivity improves robustness, not final performance" holds under both

Table 10: Baseline 2 ablation: cup-area robustness score (best-loss) for 16-rank LoRA fine-tuning under a shared, ceiling $\tau(p) = (1 + p) \cdot \text{group_min}$, swept from $p = 0\%$ (each combo's best swept loss) upward as a relative percentage above it. Each area is divided by that combo's group-min loss. Shaded cells indicate the ceiling at that percentage left- or right-censors that method's curve (the true cup extends beyond the swept LR range, so the reported area is a lower bound).

Method	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
XSum – Llama-3.2 1B (min=1.6271, init=2.7876)											
ZO-SGD	0	0.057	0.17	0.3	0.47	0.66	0.9	1.2	1.5	1.8	2.1
ZO-Adam	0	0.064	0.19	0.34	0.53	0.74	0.97	1.2	1.5	1.8	2.1
MEAZO	0	0.062	0.18	0.34	0.52	0.73	0.97	1.2	1.5	1.8	2.1
XSum – Qwen-3 0.6B (min=1.8048, init=5.3668)											
ZO-SGD	0	0.056	0.14	0.23	0.34	0.46	0.59	0.73	0.87	1	1.2
ZO-Adam	0	0.071	0.19	0.33	0.48	0.64	0.81	0.98	1.2	1.4	1.6
MEAZO	0	0.072	0.19	0.33	0.48	0.64	0.81	0.99	1.2	1.4	1.6
XSum – Qwen-3 8B (min=1.1771, init=5.8863)											
ZO-SGD	0	0.032	0.095	0.17	0.26	0.36	0.46	0.56	0.67	0.78	0.89
ZO-Adam	0	0.051	0.15	0.26	0.39	0.53	0.67	0.82	0.98	1.1	1.3
MEAZO	0	0.051	0.14	0.26	0.39	0.53	0.68	0.83	0.98	1.1	1.3
SST-2 – Llama-3.2 1B (min=0.0450, init=1.2100)											
ZO-SGD	0	0.0093	0.035	0.073	0.12	0.18	0.24	0.31	0.39	0.47	0.55
ZO-Adam	0	0.014	0.051	0.11	0.17	0.24	0.33	0.42	0.53	0.63	0.74
MEAZO	0	0.0087	0.046	0.1	0.17	0.24	0.32	0.42	0.52	0.63	0.74
SST-2 – Qwen-3 0.6B (min=0.0393, init=8.0204)											
ZO-SGD	0	0.017	0.057	0.11	0.18	0.24	0.31	0.38	0.46	0.54	0.62
ZO-Adam	0	0.021	0.072	0.14	0.22	0.31	0.4	0.49	0.59	0.69	0.8
MEAZO	0	0.026	0.074	0.14	0.22	0.31	0.4	0.49	0.58	0.68	0.78
SST-2 – Qwen-3 8B (min=0.0234, init=7.5706)											
ZO-SGD	0	0.0027	0.014	0.03	0.05	0.073	0.099	0.13	0.16	0.2	0.23
ZO-Adam	0	0.0042	0.031	0.073	0.13	0.19	0.26	0.32	0.39	0.46	0.53
MEAZO	0	0.013	0.047	0.091	0.15	0.22	0.29	0.37	0.45	0.54	0.64
SQuAD – Llama-3.2 1B (min=0.2402, init=1.6380)											
ZO-SGD	0	0.02	0.076	0.15	0.24	0.33	0.44	0.55	0.66	0.78	0.9
ZO-Adam	0	0.033	0.1	0.2	0.31	0.43	0.57	0.71	0.86	1	1.2
MEAZO	0	0.03	0.097	0.19	0.3	0.41	0.54	0.68	0.82	0.97	1.1
SQuAD – Qwen-3 0.6B (min=0.2234, init=7.8602)											
ZO-SGD	0	0.012	0.054	0.11	0.18	0.26	0.34	0.43	0.52	0.62	0.72
ZO-Adam	0	0.022	0.073	0.14	0.22	0.31	0.41	0.5	0.61	0.71	0.81
MEAZO	0	0.021	0.071	0.14	0.22	0.31	0.4	0.5	0.61	0.71	0.81
SQuAD – Qwen-3 8B (min=0.1403, init=7.7015)											
ZO-SGD	0	0.0081	0.034	0.07	0.12	0.17	0.23	0.29	0.35	0.41	0.47
ZO-Adam	0	0.019	0.07	0.14	0.22	0.31	0.4	0.5	0.6	0.7	0.81
MEAZO	0	0.021	0.068	0.13	0.2	0.29	0.38	0.48	0.59	0.69	0.8

alternative ceilings. The boundary-safe τ from Section 4 is therefore the one choice in this family guaranteed not to under-estimate a method's robustness.

D.3 LoRA Learning and Robustness Curves

Due to space limitations in the main paper, we provide the LoRA fine-tuning loss curves in this appendix. Figures 8 and 9 show the training and validation loss curves, respectively, for different zeroth-order optimization methods across models and tasks using the best step size for each method. Both training and validation losses exhibit the same overall trends: a well-tuned ZO-SGD consistently matches or outperforms adaptive ZO baselines, while MEAZO achieves performance comparable to ZO-Adam. These curves provide additional insight into the optimization dynamics throughout training.

Figure 10 shows that adaptivity (ZO-Adam and MEAZO) substantially strengthens robustness by widening the stable learning-rate window and reducing performance degradation at suboptimal η, particularly for larger step sizes. While a well-tuned ZO-SGD can match or exceed adaptive methods, it remains comparatively brittle. MEAZO is consistently as robust as ZO-Adam in both settings, while reducing memory.

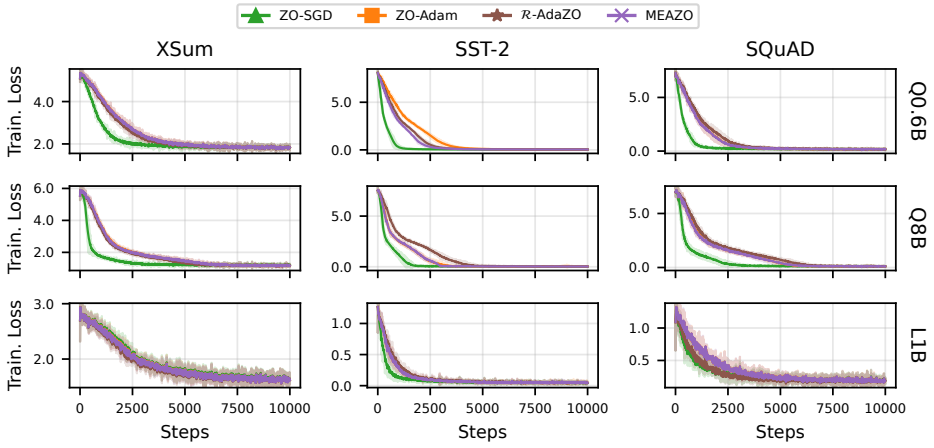


Figure 8: Training loss curves for zeroth-order methods applied to LoRA fine-tuning across models (L: Llama-3.2, Q: Qwen-3) and tasks, using the best step size for each method. Curves show the mean over three seeds, with shaded regions denoting ± 2 standard deviations.

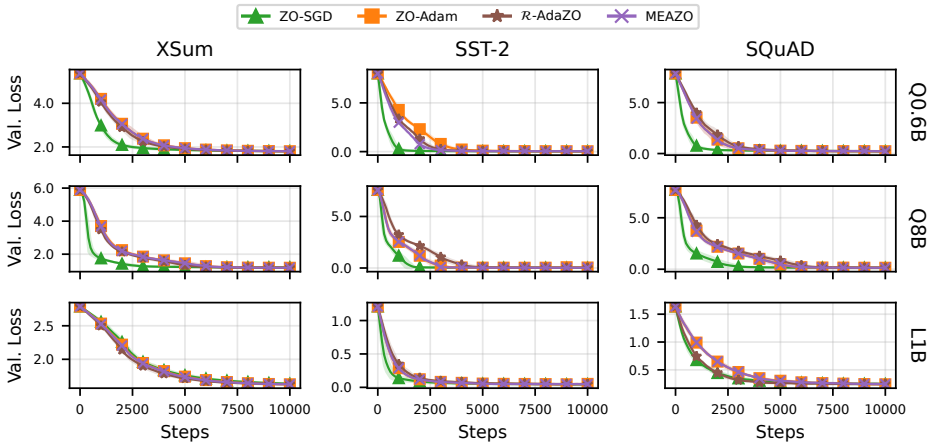


Figure 9: Validation loss curves for zeroth-order methods applied to LoRA fine-tuning across models (L: Llama-3.2, Q: Qwen-3) and tasks, using the best step size for each method. Curves show the mean over three seeds, with shaded regions denoting ± 2 standard deviations.

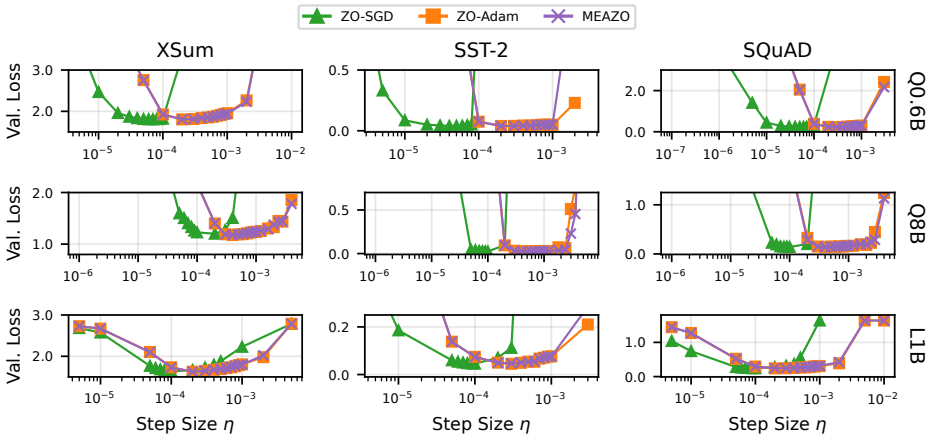
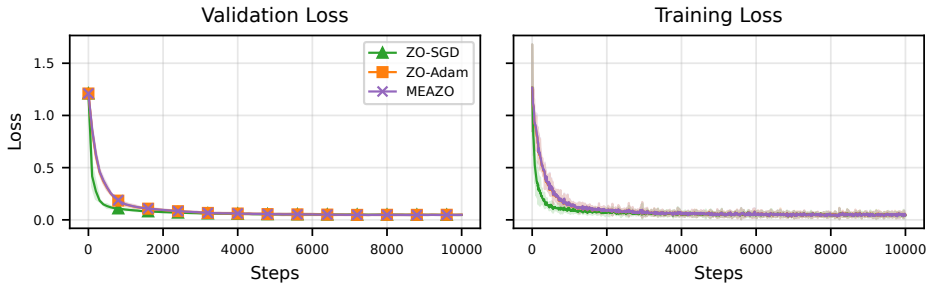


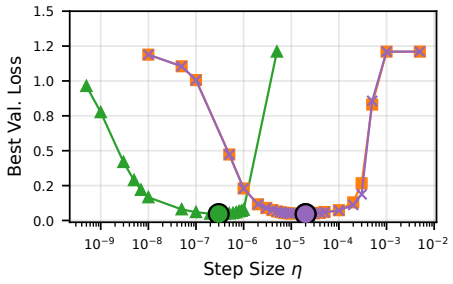
Figure 10: Robustness of ZO-SGD, ZO-Adam, and MEAZO to step size (η) initialization across models when LoRA fine-tuned on various datasets.

D.4 Full Model Fine-tuning

To assess the applicability of MEAZO in memory-intensive settings beyond LoRA or other PEFT methods, we additionally perform here full-model fine-tuning of Llama-3.2 1B on SST-2.



(a) Loss curves



(b) Robustness

Figure 11: Training-loss dynamics and step size robustness for zeroth-order methods on Llama-3.2 1B full model fine-tuning on SST-2. (a) Training and validation loss curves at each method's best step size, showing the mean over three seeds with shaded regions denoting ± 2 standard deviations. (b) Best validation loss as a function of step size η.

Learning and Robustness Curves. We complement the results in Table 4 with the corresponding learning curves (Figure 11a) and robustness curves (Figure 11b). Consistent with expectations, the trends observed in our LoRA fine-tuning experiments persist under full-model fine-tuning, indicating that the properties of ZO adaptive optimizers are not specific to the parameter-efficient setting.

Memory and Latency. While Table 4 presents SST-2 results for Llama-3.2 1B with a batch size of 8 and $q = 1$ over 10 optimization steps (average sequence length: 108.5 tokens), we further characterize the efficiency of the methods through a comprehensive latency and memory analysis across a broader range of model sizes. Measurements are performed in a lightweight setting with batch size 1 and $q = 1$ over 100 steps, using average sequence lengths of 91.7, 263.0, and 616.0 tokens for SST-2, SQuAD, and XSum, respectively. Table 11 shows that MEAZO retains a computational and memory footprint comparable to ZO-SGD while scaling substantially better than ZO-Adam.

Table 11: Peak GPU memory and average per-step latency for full-model fine-tuning with $q = 1$ and batch size 1.

Dataset	Method	L1B		Q0.6B		Q4B		Q8B	
		Peak Mem. (GiB)	Avg. Lat. (s/step)	Peak Mem. (GiB)	Avg. Lat. (s/step)	Peak Mem. (GiB)	Avg. Lat. (s/step)	Peak Mem. (GiB)	Avg. Lat. (s/step)
SST-2	ZO-SGD	10.19	0.14	5.55	0.18	31.96	0.39	68.50	0.73
	ZO-Adam	23.03	0.18	11.63	0.23	75.72	0.55	**OOM**	**OOM**
	MEAZO	10.19	0.15	5.56	0.22	31.95	0.42	68.51	0.75
SQuAD	ZO-SGD	10.19	0.18	5.61	0.21	31.98	0.56	68.52	1.04
	ZO-Adam	23.03	0.23	11.73	0.21	75.62	0.71	**OOM**	**OOM**
	MEAZO	10.19	0.20	5.62	0.22	31.95	0.59	68.52	1.08
XSum	ZO-SGD	10.86	0.31	6.69	0.27	32.24	0.90	68.55	1.63
	ZO-Adam	23.03	0.35	11.65	0.30	75.64	1.06	**OOM**	**OOM**
	MEAZO	10.86	0.32	6.71	0.28	32.27	0.94	68.53	1.68

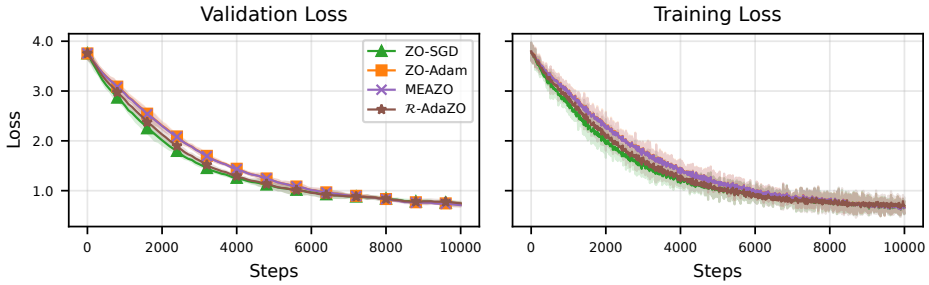
Table 12: Pets accuracy before and after 16-rank LoRA fine-tuning, and a step-size robustness score (cup area under a tau ceiling shared across methods, divided by this table's minimum swept loss). Results are averaged over seeds.

Method	Accuracy (%)	Best η	Robustness
Base	3.65	-	-
ZO-SGD	$78.64_{\pm 1.05}$	2e−4	0.643
ZO-Adam	$79.52_{\pm 0.89}$	7e−4	0.651
$\mathcal{R}$-AdaZO	$76.76_{\pm 1.30}$	2e−4	0.649
MEAZO	$79.55_{\pm 1.07}$	7e−4	0.662

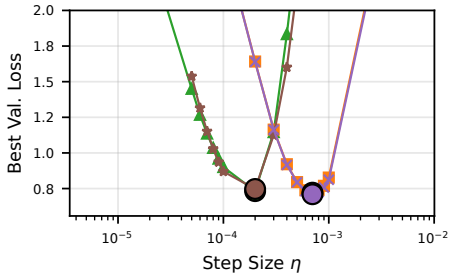
D.5 More Models

We present additional experiments beyond LLM fine-tuning, covering both vision and multimodal settings. In particular, we fine-tune a Vision Transformer (ViT-B/16, ∼86M parameters) on the Oxford-IIIT Pets dataset, and the multimodal Ministral 14B model on SQuAD. For both settings, we follow the same experimental protocol as in our LLM experiments, including three random seeds and the same two-stage step-size sweep procedure.

ViT-B/16 on Oxford-IIIT Pets. We fine-tune ViT-B/16 using LoRA on the Oxford-IIIT Pets dataset. Figure 12 reports the training and validation loss curves at each method's best step size, as well as robustness to step-size perturbations. Table 12 summarizes the final test accuracy for each method alongside the corresponding optimal step size.



(a) Loss curves



(b) Robustness

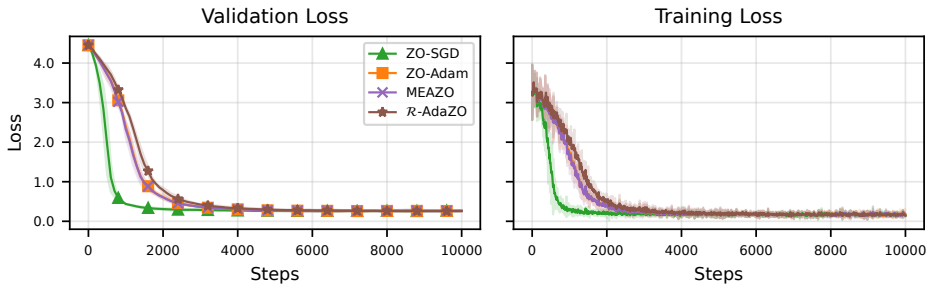
Figure 12: Training-loss dynamics and step-size robustness for zeroth-order methods on ViT-B/16 fine-tuning on Pets. (a) Training and validation loss curves at each method's best step size, showing the mean over three seeds with shaded regions denoting ±2 standard deviations. (b) Best validation loss as a function of step size η.

Ministral 14B on SQuAD. Using the same setup, we fine-tune the multimodal Ministral 14B model on the SQuAD dataset. Figure 13 reports training and validation loss curves at the best step

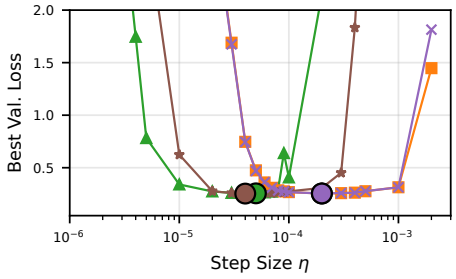
Table 13: SQuAD F1 before and after 16-rank LoRA fine-tuning on Ministral-3 14B, and a step-size robustness score (cup area under a tau ceiling shared across methods, divided by this table's minimum swept loss). Results are averaged over seeds.

Method	F1 (%)	Best η	Robustness
Base	19.85	-	-
ZO-SGD	$85.29_{\pm 0.56}$	5e−5	6.027
ZO-Adam	$85.41_{\pm 0.43}$	2e−4	7.080
$\mathcal{R}$-AdaZO	$85.99_{\pm 0.20}$	4e−5	6.919
MEAZO	$85.75_{\pm 0.09}$	2e−4	6.919

size for each method, as well as robustness to step-size perturbations. Final test performance and optimal step sizes are summarized in Table 13.



(a) Loss curves



(b) Robustness

Figure 13: Training-loss dynamics and step-size robustness for zeroth-order methods on Ministral 14B fine-tuning on SQuAD. (a) Training and validation loss curves at each method's best step size, showing the mean over three seeds with shaded regions denoting ± 2 standard deviations. (b) Best validation loss as a function of step size η.

Conclusion. Overall, the conclusions drawn from our LLM experiments carry over to both the vision and multimodal settings. First, when properly tuned, ZO-SGD performs comparably to adaptive methods. Second, per-coordinate adaptivity offers no tangible benefit: scalar-based MEAZO closely tracks the performance of ZO-Adam. Finally, the primary advantage of adaptivity lies in improved robustness to step-size selection, rather than in per-coordinate scaling itself. This is consistent with the step-size robustness scores reported in Tables 12 and 13: ZO-Adam and MEAZO attain the highest scores, mirroring our LLM findings.

D.6 Impact of β_1 in Adaptive ZO Optimization

In MEAZO (Alg. 1) we employ second moment normalization of the projected gradients only. MEAZO can be viewed as a compressed variant of ZO-Adam with $\beta_1 = 0$, by design. The key quantity is the average projected gradient g (Line 7 in Algorithm 1), which captures directional information.

In the update step (Line 10),

$$\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} - \frac{\eta}{\sqrt{\hat{v} + \zeta}} \left(\frac{1}{q} \sum_{i=1}^{q} \Delta f_i(\mathbf{x}_t; \mathbf{u}_i; \xi) \mathbf{u}_i \right),$$

each noise vector $\mathbf{u}_i$ is weighted by its corresponding projected gradient. This ensures proper directional credit assignment. Replacing the sum with a single smoothed scalar would discard this structure and degrade the update quality.

Figure 14 further supports this design choice: disabling β_1 in ZO-Adam has little to no impact on convergence when fine-tuning Llama 3.2 1B on XSum. This reinforces our decision to avoid smoothed first-moment updates in MEAZO.

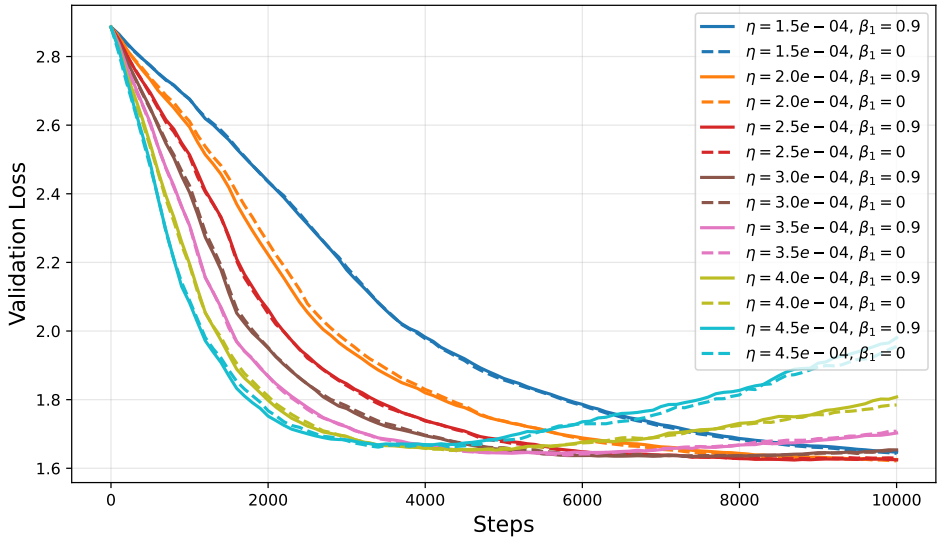


Figure 14: Validation loss curves for ZO-Adam with $\beta_1 = 0$ and $\beta_1 = 0.9$ across different step sizes during fine-tuning of Llama 3.2 1B on XSum.

D.7 Perturbation Distribution Impact

We examine how varying the perturbation distribution used in the ZO gradient estimator in (3) affects performance. Specifically, we consider normal perturbations ($\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$), uniform over the sphere ($\mathbf{u} \sim \text{Uniform}(\mathbb{S})$), Rademacher ($u_i \overset{\text{i.i.d.}}{\sim} \{-1, 1\}$), and ternary perturbations ($u_i \overset{\text{i.i.d.}}{\sim} \{-1, 0, 1\}$). As shown in Fig. 15, across ZO-SGD, ZO-Adam, $\mathcal{R}$-AdaZO, and MEAZO, the choice of perturbation distribution has negligible effect on the optimization trajectory.

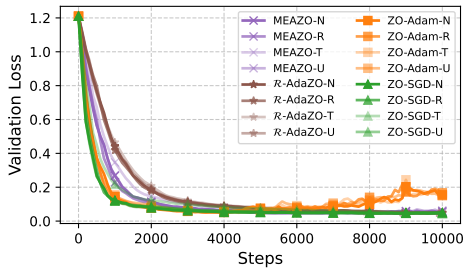


Figure 15: The impact of perturbation distributions on zero-order optimization methods for Llama-3.2 1B on SST-2: **N**ormal, **R**ademacher, **T**ernary, and **U**niform. *Best seen in color.*

D.8 Comparison with FZOO (Dang et al., 2025)

FZOO proposes an alternative mechanism for adapting the step size online. Given a minibatch ξ at iteration t, FZOO draws q Rademacher perturbations $\{\mathbf{u}_i\}_{i=1}^q$, evaluates the q corresponding one-sided perturbed losses $f_i = f(\mathbf{x}_t + \varepsilon \mathbf{u}_i; \xi)$ together with a single shared unperturbed baseline $f_0 = f(\mathbf{x}_t; \xi)$, and forms the update

$$\mathbf{g}_t = \frac{1}{\varepsilon q \sigma_t} \sum_{i=1}^{q} (f_i - f_0)\mathbf{u}_i, \tag{14}$$

where $\sigma_t = \mathsf{std}(\{f_i\}_{i=1}^q)$ is the standard deviation of the q perturbed losses.

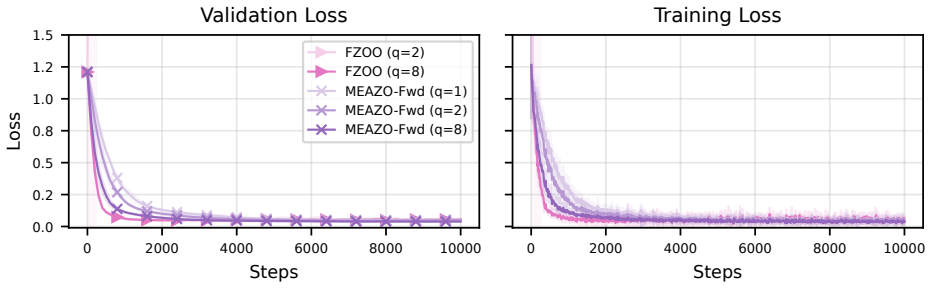
Relative to the central-difference estimator (3) studied in the main body, (14) differs along two independent axes: (i) it uses a *one-sided* (forward) finite difference with a single shared baseline evaluation rather than an antithetic pair per perturbation, so it costs $q+1$ forward passes per step instead of the $2q$ passes used by ZO-SGD, ZO-Adam, and MEAZO; and (ii) it normalizes the step by the *within-batch* standard deviation σ_t, rather than by a cross-step adaptive statistic as in MEAZO. To disentangle which of these two ingredients drives FZOO's behavior, we introduce **MEAZO-Fwd**: a variant of MEAZO that adopts FZOO's Rademacher, shared-baseline, one-sided perturbation scheme (matching its $q+1$-pass forward-difference estimator) while retaining MEAZO's own cross-step adaptive second-moment normalization in place of FZOO's within-batch σ_t. Concretely, MEAZO-Fwd computes

$$\mathbf{g}_t = \frac{1}{\varepsilon q \left(\sqrt{\hat{v}_t} + \zeta\right)} \sum_{i=1}^{q} (f_i - f_0)\mathbf{u}_i, \quad v_t = \beta v_{t-1} + (1-\beta) g_t^2, \quad g_t = \frac{1}{\varepsilon q} \sum_{i=1}^{q} (f_i - f_0), \tag{15}$$

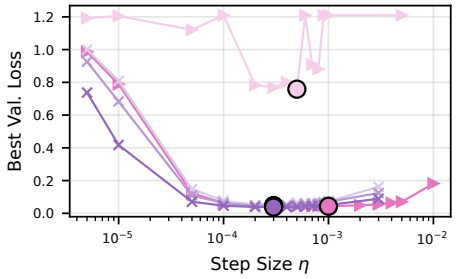
with $\hat{v}_t$ the bias-corrected EMA of g_t^2 (as in MEAZO's update, Section 5). MEAZO-Fwd therefore shares an identical perturbation scheme and forward-pass budget with FZOO, and the two methods differ only in how the step is normalized: σ_t in (14) versus $\sqrt{\hat{v}_t} + \zeta$ in (15). This results in a controlled, apples-to-apples comparison between FZOO's within-batch normalization and MEAZO's cross-step adaptivity.

We compare MEAZO-Fwd and FZOO under our extensive step-size sweeps for LoRA fine-tuning of Llama-3.2 1B on SST-2, for $q \in \{1, 2, 8\}$ (FZOO's default is $q = 8$). Note that FZOO's normalization in (14) is undefined at $q = 1$ (a single perturbed loss has no standard deviation), so that cell is omitted for FZOO; MEAZO-Fwd's cross-step statistic $\hat{v}_t$ carries no such restriction. Figure 16 shows the training and validation loss curves at each method's best step size together with the step-size robustness curves, and Table 3 reports the resulting SST-2 accuracy, best step size, and step-size robustness for every (q, method) pair.

Two trends stand out. First, FZOO is unstable at small q: at $q = 2$ its validation loss diverges and accuracy collapses to near chance, whereas MEAZO-Fwd trains stably at $q = 2$ and, unlike FZOO, remains well defined at $q = 1$. Second, at matched forward-pass budget ($q = 8$), MEAZO-Fwd attains both a slightly higher accuracy and a markedly flatter loss-vs-step-size curve (Figure 16b) than FZOO. This suggests that MEAZO's cross-step adaptive normalization, rather than the Rademacher shared-baseline perturbation scheme it shares with FZOO, is the primary driver of the robustness improvement.



(a) Loss curves



(b) Robustness

Figure 16: Training-loss dynamics and step-size robustness for FZOO and MEAZO-Fwd while LoRA fine-tuning Llama-3.2 1B on SST-2, across $q \in \{1, 2, 8\}$ (FZOO omits $q = 1$; see text). (a) Training and validation loss curves at each (q, method)'s best step size, showing the mean over three seeds with shaded regions denoting ± 2 standard deviations. (b) Best validation loss as a function of step size η.

E General Definitions and Lemmas

Let us state first useful optimization definitions and lemmas needed for our convergence proofs.

Definition E.1 (L-smoothness)**.** A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be *L-smooth* if its gradient is L-Lipschitz continuous, that is,

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d.$$

Equivalently, for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2. \tag{16}$$

Lemma E.2 (Lemma 1.2.3 of Nesterov (2013))**.** *If a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is L-*smooth*, then for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,*

$$|f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle| \leq \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2. \tag{17}$$

Lemma E.3 (ZO Projected Gradient upper bound)**.** *Let $\mathbf{x}, \mathbf{u} \in \mathbb{R}^d$ and $\varepsilon > 0$. If a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is L-*smooth*, then*

$$\langle \nabla f(\mathbf{x}), \mathbf{u} \rangle - \frac{L\varepsilon}{2}\|\mathbf{u}\|^2 \leq \frac{f(\mathbf{x} + \varepsilon\mathbf{u}) - f(\mathbf{x} - \varepsilon\mathbf{u})}{2\varepsilon} \leq \langle \nabla f(\mathbf{x}), \mathbf{u} \rangle + \frac{L\varepsilon}{2}\|\mathbf{u}\|^2, \tag{18}$$

which implies that

$$\left| \frac{f(\mathbf{x} + \varepsilon\mathbf{u}) - f(\mathbf{x} - \varepsilon\mathbf{u})}{2\varepsilon} \right| \leq |\langle \nabla f(\mathbf{x}), \mathbf{u} \rangle| + \frac{L\varepsilon}{2}\|\mathbf{u}\|^2. \tag{19}$$

Proof. Applying (16) with $\mathbf{y} := \mathbf{x} + \varepsilon\mathbf{u}$ leads to:

$$f(\mathbf{x} + \varepsilon\mathbf{u}) \leq f(\mathbf{x}) + \varepsilon\langle \nabla f(\mathbf{x}), \mathbf{u} \rangle + \frac{L\varepsilon^2}{2}\|\mathbf{u}\|^2.$$

Similarly, applying (17) with $\mathbf{y} := \mathbf{x} - \varepsilon\mathbf{u}$ and multiplying the left-hand side by -1 gives

$$-f(\mathbf{x} - \varepsilon\mathbf{u}) \leq -f(\mathbf{x}) + \varepsilon\langle \nabla f(\mathbf{x}), \mathbf{u} \rangle + \frac{L\varepsilon^2}{2}\|\mathbf{u}\|^2.$$

Summing up the above two equations and dividing by 2ε proves to (18):

$$\frac{f(\mathbf{x} + \varepsilon\mathbf{u}) - f(\mathbf{x} - \varepsilon\mathbf{u})}{2\varepsilon} \leq \langle \nabla f(\mathbf{x}), \mathbf{u} \rangle + \frac{L\varepsilon}{2}\|\mathbf{u}\|^2. \tag{20}$$

One can also apply the proved inequality to $\mathbf{u} \leftarrow -\mathbf{u}$ and multiply it by -1, which proves the left-hand side inequality of:

$$\langle \nabla f(\mathbf{x}), \mathbf{u} \rangle - \frac{L\varepsilon}{2}\|\mathbf{u}\|^2 \leq \frac{f(\mathbf{x} + \varepsilon\mathbf{u}) - f(\mathbf{x} - \varepsilon\mathbf{u})}{2\varepsilon} \leq \langle \nabla f(\mathbf{x}), \mathbf{u} \rangle + \frac{L\varepsilon}{2}\|\mathbf{u}\|^2. \tag{21}$$

Applying the fact that for any $A, M \in \mathbb{R}$ and $B \geq 0$, $A - B \leq M \leq A + B \implies |M| \leq |A| + B$ to inequalities in (21) concludes the proof. □

Lemma E.4 (Smoothing - G-Lipschitzness Transfer)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be a G-Lipschitz function. Then its smoothed version $F_\varepsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{D}}\big[F(\mathbf{x} + \varepsilon\mathbf{u})\big]$ is also G-Lipschitz.*

Proof. Since F is G-Lipschitz, for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ we have

$$|F(\mathbf{x}) - F(\mathbf{y})| \leq G\|\mathbf{x} - \mathbf{y}\|.$$

For the smoothed function F_ε, consider

$$|F_\varepsilon(\mathbf{x}) - F_\varepsilon(\mathbf{y})| = |\mathbb{E}_{\mathbf{u}\sim\mathcal{D}}\left[F(\mathbf{x} + \varepsilon\mathbf{u}) - F(\mathbf{y} + \varepsilon\mathbf{u})\right]|.$$

By Jensen's inequality and the Lipschitz property of F,

$$|F_\varepsilon(\mathbf{x}) - F_\varepsilon(\mathbf{y})| \leq \mathbb{E}_{\mathbf{u}\sim\mathcal{D}}\left[|F(\mathbf{x} + \varepsilon\mathbf{u}) - F(\mathbf{y} + \varepsilon\mathbf{u})|\right] \leq \mathbb{E}_{\mathbf{u}\sim\mathcal{D}}\left[G\|\mathbf{x} - \mathbf{y}\|\right] = G\|\mathbf{x} - \mathbf{y}\|.$$

Hence F_ε is G-Lipschitz. □

Lemma E.5 (Smoothing - L-smoothness Transfer)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function. Then its smoothed version $F_\varepsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{D}}\big[F(\mathbf{x}+\varepsilon\mathbf{u})\big]$ is also L-smooth.*

Remark E.6. The above lemma can be found in eq (12) of Nesterov & Spokoiny (2017) for normal perturbation, Lemma 4.1 (a) of Gao et al. (2018) for uniform spherical perturbation.

Proof. By Leibniz's rule (differentiation under the expectation), we have

$$\nabla F_\varepsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}}\big[\nabla F(\mathbf{x}+\varepsilon\mathbf{u})\big].$$

For any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,

$$\|\nabla F_\varepsilon(\mathbf{x}) - \nabla F_\varepsilon(\mathbf{y})\| = \Big\|\mathbb{E}_{\mathbf{u}}\big[\nabla F(\mathbf{x}+\varepsilon\mathbf{u}) - \nabla F(\mathbf{y}+\varepsilon\mathbf{u})\big]\Big\|.$$

Applying Jensen's inequality (since $\|\cdot\|$ is convex), we get

$$\|\nabla F_\varepsilon(\mathbf{x}) - \nabla F_\varepsilon(\mathbf{y})\| \le \mathbb{E}_{\mathbf{u}}\big[\|\nabla F(\mathbf{x}+\varepsilon\mathbf{u}) - \nabla F(\mathbf{y}+\varepsilon\mathbf{u})\|\big]. \tag{a}$$

Since F is L-smooth, its gradient is L-Lipschitz, so

$$\|\nabla F(\mathbf{x}+\varepsilon\mathbf{u}) - \nabla F(\mathbf{y}+\varepsilon\mathbf{u})\| \le L\|\mathbf{x}-\mathbf{y}\|. \tag{b}$$

Combining (a) and (b), we obtain

$$\|\nabla F_\varepsilon(\mathbf{x}) - \nabla F_\varepsilon(\mathbf{y})\| \le L\|\mathbf{x}-\mathbf{y}\|.$$

Hence, F_ε is also L-smooth. □

Lemma E.7 (Smoothing - Function Error)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, and let $\mathcal{P}$ be any centered probability distribution. Define its smoothed version as*

$$F_\varepsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[F(\mathbf{x}+\varepsilon\mathbf{u})\right], \quad \forall \mathbf{x} \in \mathbb{R}^d.$$

Then, for any $\mathbf{x} \in \mathbb{R}^d$,

$$|F_\varepsilon(\mathbf{x}) - F(\mathbf{x})| \le \frac{\varepsilon^2}{2} L \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\mathbf{u}\|^2\right]. \tag{22}$$

Proof. Let $\mathbf{x}, \mathbf{u} \in \mathbb{R}^d$ and $\varepsilon > 0$, by L-smoothness of F one has

$$F(\mathbf{x}+\varepsilon\mathbf{u}) - F(\mathbf{x}) \le \varepsilon\langle\nabla F(\mathbf{x}), \mathbf{u}\rangle + \frac{\varepsilon^2}{2} L \|\mathbf{u}\|^2$$

$$\implies \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[F(\mathbf{x}+\varepsilon\mathbf{u}) - F(\mathbf{x})\right] \le \varepsilon\langle\nabla F(\mathbf{x}), \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\mathbf{u}\right]\rangle + \frac{\varepsilon^2}{2} L \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\mathbf{u}\|^2\right]$$

$$\overset{\mathcal{P}\text{ centered}}{\iff} F_\varepsilon(\mathbf{x}) - F(\mathbf{x}) \le \frac{\varepsilon^2}{2} L \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\mathbf{u}\|^2\right]. \tag{23}$$

By re-applying the above reasoning by inverting $\mathbf{x}$ and $\mathbf{x}+\varepsilon\mathbf{u}$ gives the symmetric bound. □

Lemma E.8 (Smoothing - Gradient Error)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, and let $\mathcal{P}$ be any probability distribution supported on a bounded set in $\mathbb{R}^d$. Define its smoothed version as*

$$F_\varepsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[F(\mathbf{x}+\varepsilon\mathbf{u})\right], \quad \forall \mathbf{x} \in \mathbb{R}^d.$$

Then, for any $\mathbf{x} \in \mathbb{R}^d$,

$$\|\nabla F_\varepsilon(\mathbf{x}) - \nabla F(\mathbf{x})\| \le \varepsilon L \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\mathbf{u}\|\right]. \tag{24}$$

Remark E.9. A similar proof scheme is used in Berahas et al. (2022) to prove similar bounds for normal, eq. (2.10), and uniform distributions, eq. (2.35).

Proof. We have

$$\begin{aligned}
\|\nabla F_\varepsilon(\mathbf{x}) - \nabla F(\mathbf{x})\| &= \|\nabla \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[F(\mathbf{x}+\varepsilon\mathbf{u})\right] - \nabla F(\mathbf{x})\| \\
&\overset{\text{Leibniz's}}{=} \|\mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\nabla F(\mathbf{x}+\varepsilon\mathbf{u})\right] - \nabla F(\mathbf{x})\| \\
&= \|\mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\nabla F(\mathbf{x}+\varepsilon\mathbf{u}) - \nabla F(\mathbf{x})\right]\| \\
&\overset{\text{Jensen's}}{\le} \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\nabla F(\mathbf{x}+\varepsilon\mathbf{u}) - \nabla F(\mathbf{x})\|\right] \\
&\overset{L\text{-smoothness}}{\le} \varepsilon L \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[\|\mathbf{u}\|\right].
\end{aligned}$$

□

Lemma E.10 (Smoothing - Stochastic Gradient Variance Bound)**.** *Let $F : \mathbb{R}^d \to \mathbb{R}$ be the population loss defined as*

$$F(\mathbf{x}) = \mathbb{E}_{\xi\sim\mathcal{D}}\left[f(\mathbf{x};\xi)\right] \quad \forall \mathbf{x} \in \mathbb{R}^d, \tag{25}$$

where each individual sample loss $f(.;\xi)$. Let $\mathcal{P}$ be any probability distribution supported on a bounded set in $\mathbb{R}^d$. Let us recalled the definition of the smoothed version of F

$$F_\varepsilon(\mathbf{x}) := \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[F(\mathbf{x}+\varepsilon\mathbf{u})\right], \quad \forall \mathbf{x} \in \mathbb{R}^d, \tag{26}$$

and similarly for $f(.,\xi)$

$$f_\varepsilon(\mathbf{x};\xi) := \mathbb{E}_{\mathbf{u}\sim\mathcal{P}}\left[f(\mathbf{x}+\varepsilon\mathbf{u};\xi)\right], \quad \forall \mathbf{x} \in \mathbb{R}^d, \forall \xi. \tag{27}$$

Assume that, for all $\mathbf{x} \in \mathbb{R}^d$, the sample loss gradient is an unbiased estimate of the the full loss gradient[5], and that there exists $\sigma > 0$ such that, its variance is bounded by σ^2, i.e.,

$$\begin{aligned}&\mathbb{E}_\xi\left[\nabla f(\mathbf{x};\xi)\right] = \nabla F(\mathbf{x}),\\ &\mathbb{E}_\xi\left[\left\|\nabla f(\mathbf{x};\xi) - \nabla F(\mathbf{x})\right\|^2\right] \le \sigma^2.\end{aligned} \tag{28}$$

Then, for any $\mathbf{x} \in \mathbb{R}^d$, the smoothed sample loss gradient is also an unbiased estimator of the smoothed full loss and its variance of is also bounded by σ^2, i.e.,

$$\mathbb{E}_\xi\left[\nabla f_\varepsilon(\mathbf{x};\xi)\right] = \nabla F_\varepsilon(\mathbf{x}), \tag{29}$$

$$\mathbb{E}_\xi\left[\left\|\nabla f_\varepsilon(\mathbf{x};\xi) - \nabla F_\varepsilon(\mathbf{x})\right\|^2\right] \le \sigma^2. \tag{30}$$

Proof. Let $\mathbf{x} \in \mathbb{R}^d$. We have that the mean of the smoothed sample loss equals

$$\begin{aligned}\mathbb{E}_\xi\left[\nabla f_\varepsilon(\mathbf{x};\xi)\right] &\overset{(27)}{=} \mathbb{E}_\xi\left[\nabla \mathbb{E}_{\mathbf{u}}\left[f(\mathbf{x}+\varepsilon\mathbf{u};\xi)\right]\right] \overset{\text{Leibniz' rule}}{=} \nabla \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_\xi\left[f(\mathbf{x}+\varepsilon\mathbf{u};\xi)\right]\right]\\ &\overset{(25)}{=} \nabla \mathbb{E}_{\mathbf{u}}\left[F(\mathbf{x}+\varepsilon\mathbf{u})\right] \overset{(26)}{=} \nabla F_\varepsilon(\mathbf{x}).\end{aligned}$$

We also have that the variance of the smoothed sample loss equals

$$\begin{aligned}\mathbb{E}_\xi\left[\left\|\nabla f_\varepsilon(\mathbf{x};\xi) - \nabla F_\varepsilon(\mathbf{x})\right\|^2\right] &= \mathbb{E}_\xi\left[\left\|\nabla \mathbb{E}_{\mathbf{u}}\left[f(\mathbf{x}+\varepsilon\mathbf{u};\xi) - F(\mathbf{x}+\varepsilon\mathbf{u})\right]\right\|^2\right]\\ &\overset{\text{Leibniz' rule}}{=} \mathbb{E}_\xi\left[\left\|\mathbb{E}_{\mathbf{u}}\left[\nabla f(\mathbf{x}+\varepsilon\mathbf{u};\xi) - \nabla F(\mathbf{x}+\varepsilon\mathbf{u})\right]\right\|^2\right]\\ &\overset{\text{Jensen's ineq.}}{\le} \mathbb{E}_\xi\left[\mathbb{E}_{\mathbf{u}}\left[\left\|\nabla f(\mathbf{x}+\varepsilon\mathbf{u};\xi) - \nabla F(\mathbf{x}+\varepsilon\mathbf{u})\right\|^2\right]\right]\\ &\overset{(28)}{\le} \sigma^2\end{aligned}$$

□

E.1 Uniform Perturbations

In this section we will derive results for the *q-uniform samples centered ZO estimator*:

$$\hat{\nabla} f_\varepsilon^q(\mathbf{x}) = \frac{d}{q}\sum_{i=1}^{q}\frac{f(\mathbf{x}+\varepsilon\mathbf{u}_i) - f(\mathbf{x}-\varepsilon\mathbf{u}_i)}{2\varepsilon}\mathbf{u}_i, \tag{31}$$

where $\mathbf{u}_i \sim \text{Uniform}(\mathbb{S})$ with $\mathbb{S} = \{\mathbf{u} \in \mathbb{R}^d : \|\mathbf{u}\| = 1\}$ being the unit sphere of dimensions d and $\varepsilon > 0$.

We now state useful and known results.

Definition E.11 (Uniform Smoothed Function)**.** Let us denote by f_ε, the smoothed function over the unit ball $\mathbb{B}$ of dimensions d, that is for all $\mathbf{x} \in \mathbb{R}^d$

$$f_\varepsilon(\mathbf{x}) := \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}\left[f(\mathbf{x}+\varepsilon\mathbf{v})\right]. \tag{32}$$

[5] If stochastic losses are regular enough, *e.g.,* $\mathbb{E}_\xi\left[\|\nabla f(\mathbf{x};\xi)\|\right] < \infty$ such that dominated convergence theorem can be applied, then $\mathbb{E}_\xi\left[\nabla f(\mathbf{x};\xi)\right] = \nabla F(\mathbf{x})$.

Remark E.12. This is the same definition as in eq. (4.1) of Shalev-Shwartz (2011) and eq. (50) of Gao et al. (2018).

Lemma E.13 (Unbiased ZO Gradient Estimate of the Smoothed Function)**.** *The ZO estimator in (31) provides an unbiased estimate of the gradient of the smoothed function f_ε defined in (32):*

$$\mathbb{E}_{\mathcal{U}_q \sim (\mathbb{S})^q} \left[\hat{\nabla} f_\varepsilon^q(\mathbf{x}) \right] = \nabla f_\varepsilon(\mathbf{x}), \tag{33}$$

where $\mathcal{U}_q := \{\mathbf{u}_1, \ldots, \mathbf{u}_q\}$ denotes the q random vectors sampled i.i.d. from the unit sphere $\mathbb{S}$.

Proof. This is a direct consequence of Lemma 1 of Flaxman et al. (2004) or of Lemma 4.4 of Shalev-Shwartz (2011) applied to $\mathbf{u}$ and $-\mathbf{u}$. □

F Omitted Proofs and Derivations

F.1 Proof of Proposition 3.1

We provide the proof of Proposition 3.1 (restated below):

Proposition (Restated)**.** *Let $f : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, and let $\hat{\nabla} f_\varepsilon^q(\mathbf{x})$ denote the q-sample ZO estimator in (3), where $\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ for all $i \in [q] := \{1, \dots, q\}$. Then in the $\varepsilon \to 0$ limit we have:*

$$\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right] \approx \frac{1}{q}\left(\|\nabla f(\mathbf{x})\|^2 \mathbf{1}_d + \nabla f(\mathbf{x})^2\right) + \nabla f(\mathbf{x})^2, \tag{34}$$

where $(\cdot)^2$ stands for the element-wise power of 2 and $\mathbb{E}_{\mathbf{u}}$ denotes the expectation over $\mathbf{u}_1, \dots, \mathbf{u}_q$.

Proof. We first note that from Lemma E.3, we have $\forall \mathbf{u} \in \mathbb{R}^d$:

$$\lim_{\varepsilon \to 0} \frac{f(\mathbf{x} + \varepsilon \mathbf{u}) - f(\mathbf{x} - \varepsilon \mathbf{u})}{2\varepsilon} = \mathbf{u}^\top \nabla f(\mathbf{x}). \tag{35}$$

Then, in the $\varepsilon \to 0$ limit we have:

$$\hat{\nabla} f_\varepsilon^q(\mathbf{x}) = \frac{1}{q}\sum_{i=1}^q \frac{f(\mathbf{x} + \varepsilon \mathbf{u}_i) - f(\mathbf{x} - \varepsilon \mathbf{u}_i)}{2\varepsilon}\mathbf{u}_i \approx \frac{1}{q}\sum_{i=1}^q \left(\mathbf{u}_i^\top \nabla f(\mathbf{x})\right)\mathbf{u}_i = \frac{1}{q}\sum_{i=1}^q \left(\mathbf{g}^\top \mathbf{u}_i\right)\mathbf{u}_i \tag{36}$$

where $\mathbf{g} = \nabla f(\mathbf{x})$ and we drop the dependency on $\mathbf{x}$ for clarity. Taking the expectation of the k^{th} element squared, we get:

$$\left[\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right]\right]_k \overset{(36)}{\approx} \frac{1}{q^2}\mathbb{E}_{\mathbf{u}}\left[\left(\sum_{i=1}^q \left(\mathbf{g}^\top \mathbf{u}_i\right)\mathbf{u}_i\right)^2\right] \tag{37}$$

$$= \frac{1}{q^2}\mathbb{E}_{\mathbf{u}}\left[\sum_{i=1}^q \left(\mathbf{g}^\top \mathbf{u}_i\right)^2 u_{i,k}^2 + \sum_{i=1}^q \sum_{j \neq i} (\mathbf{g}^\top \mathbf{u}_i) u_{i,k} (\mathbf{g}^\top \mathbf{u}_j) u_{j,k}\right] \tag{38}$$

$$= \frac{1}{q^2}\sum_{i=1}^q \mathbb{E}_{\mathbf{u}}\left[\left(\mathbf{g}^\top \mathbf{u}_i\right)^2 u_{i,k}^2\right] + \frac{1}{q^2}\sum_{i=1}^q \sum_{j \neq i} \mathbb{E}_{\mathbf{u}}\left[(\mathbf{g}^\top \mathbf{u}_i) u_{i,k} (\mathbf{g}^\top \mathbf{u}_j) u_{j,k}\right] \tag{39}$$

Expanding the first term with $\mathbf{v} := \mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$:

$$\begin{aligned}
\mathbb{E}_{\mathbf{v}}\left[\left(\mathbf{g}^\top \mathbf{v}\right)^2 v_k^2\right] &= \mathbb{E}_{\mathbf{v}}\left[\left(\sum_{l=1}^d g_l v_l\right)^2 v_k^2\right] \\
&= \mathbb{E}_{\mathbf{v}}\left[\left(\sum_{l=1}^d g_l^2 v_l^2 + \sum_{l \neq s} g_l g_s v_l v_s\right) v_k^2\right] \\
&= \sum_{l=1}^d g_l^2 \mathbb{E}_{\mathbf{v}}\left[v_l^2 v_k^2\right] + \underbrace{\sum_{l \neq s} g_l g_s \mathbb{E}_{\mathbf{v}}\left[v_l v_s v_k^2\right]}_{\text{vanishes since odd moments = 0}} \\
&= g_k^2 \mathbb{E}_{\mathbf{v}}\left[v_k^4\right] + \sum_{l \neq k} g_l^2 \mathbb{E}_{\mathbf{v}}\left[v_l^2\right] \mathbb{E}_{\mathbf{v}}\left[v_k^2\right] \\
&= g_k^2 \cdot 3 + \sum_{l \neq k} g_l^2 \cdot 1 \\
&= 3g_k^2 + \sum_{l \neq k} g_l^2 \\
&= \sum_{l=1}^d g_l^2 + 2g_k^2 \\
&= \|\mathbf{g}\|^2 + 2g_k^2.
\end{aligned}$$

For the second term, with $\mathbf{v} \coloneqq \mathbf{u}_i$, $\mathbf{w} \coloneqq \mathbf{u}_j, i \neq j$:

$$\begin{aligned}
\mathbb{E}_{\mathbf{v},\mathbf{w}}\left[(\mathbf{g}^\top \mathbf{v})v_k(\mathbf{g}^\top \mathbf{w})w_k\right] &= \mathbb{E}_{\mathbf{v}}\left[(\mathbf{g}^\top \mathbf{v})v_k\right] \cdot \mathbb{E}_{\mathbf{w}}\left[(\mathbf{g}^\top \mathbf{w})w_k\right] \quad (\mathbf{v} \perp \mathbf{w}) \\
&= \Big(\sum_{a=1}^{d} g_a \mathbb{E}_{\mathbf{v}}\left[v_a v_k\right]\Big) \cdot \Big(\sum_{b=1}^{d} g_b \mathbb{E}_{\mathbf{w}}\left[w_b w_k\right]\Big) \\
&= \Big(\sum_{a=1}^{d} g_a \delta_{a,k}\Big) \cdot \Big(\sum_{b=1}^{d} g_b \delta_{b,k}\Big) \\
&= g_k^2.
\end{aligned}$$

Plugging back into (37):

$$\begin{aligned}
\left[\mathbb{E}_{\mathbf{u}}\left[\hat{\mathbf{g}}_q^2(\mathbf{x})\right]\right]_k &\approx \frac{1}{q^2}\Big(q(\|\mathbf{g}\|^2 + 2g_k^2) + q(q-1)g_k^2\Big) \\
&= \frac{1}{q^2}\Big(q\|\mathbf{g}\|^2 + 2qg_k^2 + q(q-1)g_k^2\Big) \\
&= \frac{\|\mathbf{g}\|^2}{q} + \frac{2g_k^2}{q} + \frac{(q-1)g_k^2}{q} \\
&= \frac{\|\mathbf{g}\|^2}{q} + \frac{(q+1)g_k^2}{q} \\
&= \frac{1}{q}\left(\|\mathbf{g}\|^2 + g_k^2\right) + g_k^2.
\end{aligned}$$

Since $\mathbf{g} = \nabla f(\mathbf{x})$, this matches the proposition in vector form:

$$\boxed{\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right] \approx \frac{1}{q}\left(\|\nabla f(\mathbf{x})\|^2 \mathbf{1}_d + \nabla f(\mathbf{x})^2\right) + \nabla f(\mathbf{x})^2.}$$

□

F.2 Proof of Proposition F.1

We provide the proof of the following porpositio for the uniform perturbation case:

Proposition F.1 (Uniform). *Let $f : \mathbb{R}^d \to \mathbb{R}$ be an L-smooth function, and let $\hat{\nabla} f_\varepsilon^q(\mathbf{x})$ denote the q-sample ZO estimator in (3), where $\mathbf{u}_i \sim$ Uniform$(\mathbb{S})$ for all $i \in [q] \coloneqq \{1, \ldots, q\}$. Then in the $\varepsilon \to 0$ limit we have:*

$$\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right] \approx \frac{d\left(\|\nabla f(\mathbf{x})\|^2 \mathbf{1}_d + 2\nabla f(\mathbf{x})^2\right)}{q(d+2)} + \frac{q-1}{q}\nabla f(\mathbf{x})^2 \tag{40}$$

where $(\cdot)^2$ stands for the element-wise power of 2 and $\mathbb{E}_{\mathbf{u}}$ denotes the expectation over $\mathbf{u}_1, \ldots, \mathbf{u}_q$.

Proof. Let $\mathbf{u}_i \sim$ Uniform$(\mathbb{S})$ be sampled from the unit sphere in $\mathbb{R}^d$. By Lemma E.3 and in the $\varepsilon \to 0$ limit, the q-sample ZO estimator becomes:

$$\hat{\nabla} f_\varepsilon^q(\mathbf{x}) \approx \frac{d}{q}\sum_{i=1}^{q}(\mathbf{g}^\top \mathbf{u}_i)\mathbf{u}_i, \tag{41}$$

where $\mathbf{g} = \nabla f(\mathbf{x})$ for brevity.

Taking the expectation of the k^{th} element squared, we get:

$$\begin{aligned}
\left[\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right]\right]_k &\approx \frac{d^2}{q^2}\mathbb{E}_{\mathbf{u}}\left[\Big(\sum_{i=1}^{q}(\mathbf{g}^\top \mathbf{u}_i)\mathbf{u}_i\Big)^2\right] \\
&= \frac{d^2}{q^2}\left(\sum_{i=1}^{q}\mathbb{E}_{\mathbf{u}}\left[(\mathbf{g}^\top \mathbf{u}_i)^2 u_{i,k}^2\right] + \sum_{i\neq j}\mathbb{E}_{\mathbf{u}}\left[(\mathbf{g}^\top \mathbf{u}_i)u_{i,k}(\mathbf{g}^\top \mathbf{u}_j)u_{j,k}\right]\right).
\end{aligned}$$

For $\mathbf{v} \sim \text{Uniform}(\mathbb{S})$, the moments are:

$$\mathbb{E}[v_i^2] = \frac{1}{d}, \quad \mathbb{E}[v_i^4] = \frac{3}{d(d+2)}, \quad \mathbb{E}[v_i^2 v_j^2] = \frac{1}{d(d+2)}.$$

Thus:

$$\mathbb{E}_{\mathbf{v}}\left[(\mathbf{g}^\top \mathbf{v})^2 v_k^2\right] = g_k^2 \cdot \frac{3}{d(d+2)} + \sum_{l \neq k} g_l^2 \cdot \frac{1}{d(d+2)} = \frac{\|\mathbf{g}\|^2 + 2g_k^2}{d(d+2)}.$$

For two independent samples $\mathbf{v}, \mathbf{w}$:

$$\mathbb{E}_{\mathbf{v},\mathbf{w}}\left[(\mathbf{g}^\top \mathbf{v}) v_k (\mathbf{g}^\top \mathbf{w}) w_k\right] = \Big(\sum_a g_a \mathbb{E}[v_a v_k]\Big)^2 = (g_k/d)^2 = \frac{g_k^2}{d^2}.$$

Plugging back:

$$\left[\mathbb{E}_{\mathbf{u}}\left[\hat{\mathbf{g}}_q^2(\mathbf{x})\right]\right]_k \approx \frac{d^2}{q^2}\Big(q \cdot \frac{\|\mathbf{g}\|^2 + 2g_k^2}{d(d+2)} + q(q-1) \cdot \frac{g_k^2}{d^2}\Big).$$

Simplify:

$$= \frac{d(\|\mathbf{g}\|^2 + 2g_k^2)}{q(d+2)} + \frac{(q-1)g_k^2}{q}.$$

And we obtain the vector form:

$$\boxed{\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f_\varepsilon^q(\mathbf{x})^2\right] \approx \frac{d\left(\|\nabla f(\mathbf{x})\|^2 \mathbf{1}_d + 2\nabla f(\mathbf{x})^2\right)}{q(d+2)} + \frac{q-1}{q}\nabla f(\mathbf{x})^2.}$$

□

G MEAZO'S CONVERGENCE PROOF

G.1 ASSUMPTIONS

Assumption G.1 (Uniform perturbations independent from the mini-batch)**.** For each iteration $t = 1, \dots, T$, the mini-batch ξ_t and the perturbation $\mathbf{u}_t$ are sampled independently. Moreover, the perturbations are uniformly sampled over the unit $\mathbb{R}^d$ sphere, that is

$$\mathbf{u}_t \sim \mathrm{Unif}(\mathbb{S}^{d-1}), \qquad \forall t \in \mathbb{N}^*, \tag{42}$$

where $\mathbb{S}^{d-1} = \{u \in \mathbb{R}^d : \|u\| = 1\}$. The perturbations satisfy

$$\mathbb{E}[\mathbf{u}_t] = 0, \qquad \mathbb{E}[\mathbf{u}_t \mathbf{u}_t^\top] = \frac{1}{d} I_d, \qquad \|\mathbf{u}_t\|^2 = 1. \tag{43}$$

Assumption G.2 (Lower boundedness of the objective)**.** There exists $F_\star > -\infty$ such that

$$F(\mathbf{x}) \geq F_\star, \qquad \forall \mathbf{x} \in \mathbb{R}^d. \tag{44}$$

Consequently, the smoothed objective is also lower bounded, that is

$$F_\varepsilon(\mathbf{x}) := \mathbb{E}_{\mathbf{v} \sim \mathrm{Unif}(\mathbb{B}^d)} \left[F(\mathbf{x} + \varepsilon \mathbf{v}) \right] \geq F_\star, \tag{45}$$

where $\mathbb{B}^d := \{\mathbf{v} \in \mathbb{R}^d : \|\mathbf{v}\| \leq 1\}$.

Assumption G.3 (Smoothness)**.** For every ξ, the stochastic objective $f(\cdot; \xi)$ is L-smooth:

$$\|\nabla f(\mathbf{x}; \xi) - \nabla f(\mathbf{y}; \xi)\| \leq L \|\mathbf{x} - \mathbf{y}\| . \tag{46}$$

Consequently, F and F_ε are L-smooth.

Let $q = 1$, and define the unscaled directional finite difference

$$\delta_t := \frac{f(\mathbf{x}_t + \varepsilon \mathbf{u}_t; \xi_t) - f(\mathbf{x}_t - \varepsilon \mathbf{u}_t; \xi_t)}{2\varepsilon} . \tag{47}$$

The projected gradient for a uniform perturbation is then

$$g_t := d\delta_t = d \frac{f(\mathbf{x}_t + \varepsilon \mathbf{u}_t; \xi_t) - f(\mathbf{x}_t - \varepsilon \mathbf{u}_t; \xi_t)}{2\varepsilon} . \tag{48}$$

Assumption G.4 (Bounded directional finite differences)**.** There exists a dimension-independent constant $B > 0$ such that, almost surely,

$$|\delta_t| \leq B, \qquad \forall t \in \mathbb{N}^* . \tag{49}$$

Let us also define the projected gradient bound with explicit dimension dependency

$$R_d := \sqrt{\zeta} + dB . \tag{50}$$

Then, by (48),

$$0 \leq g_t^2 \leq d^2 B^2 = \left(R_d - \sqrt{\zeta} \right)^2 , \qquad \forall t \in \mathbb{N}^* . \tag{51}$$

Comparison with assumptions in the literature. The uniform sphere sampling assumption is standard as in Flaxman et al. (2004); Nazari et al. (2020); Shu et al. (2025) and empirically (Appendix D.7) we observe that the choice of perturbation distribution does not impact model performance. Extending the analysis to Gaussian perturbations is left for future work.

Smoothness and boundedness of the gradients are standard in the adaptive FO optimization literature (Chen et al., 2019; Ward et al., 2020; Zaheer et al., 2018). On a side note, relaxing assumptions for the convergence of adaptive methods in the FO setting is an active research topic Wang et al. (2023); Hong & Lin (2024); Zhang et al. (2025) going beyond the scope of this work. In particular, (Défossez et al., 2020) assume a uniform bound on the stochastic gradients in the first-order setting. Our Assumption G.4 plays the analogous role in the zeroth-order setting: bounding the unscaled directional finite difference δ_t controls the scalar projected gradient $g_t = d\delta_t$, and therefore the adaptive denominator and second-moment estimate. The induced bound $R_d := \sqrt{\zeta} + dB$ makes the dimension dependence explicit. ZO-AdaMM Chen et al. (2019) similarly assumes smoothness and uniform (infinite) norm bounded gradient. $\mathcal{R}$-AdaZO Shu et al. (2025) instead assumes bounded function values and variance and derives dimension-dependent second moment bounds for the ZO estimator.

G.2 Main result

Let us recall MEAZO's update rule (7), for $q = 1$ and uniform perturbation at $t \in \mathbb{N}^*$:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \alpha_t \mathbf{G}_t \ , \tag{52}$$

where

$$\alpha_t := \frac{1}{\sqrt{\hat{v}_t + \zeta}} \in \mathbb{R} \ , \tag{53}$$

$$\mathbf{G}_t := \Delta f_\varepsilon(\mathbf{x}_t; \mathbf{u}_t, \xi_t)\mathbf{u}_t \overset{(48)}{=} g_t \mathbf{u}_t \in \mathbb{R}^d \ . \tag{54}$$

Let us also recall the update of the scalar second moment estimate

$$v_t = \beta v_{t-1} + (1-\beta) g_t^2 \in \mathbb{R} \ . \tag{55}$$

So one gets

$$v_t = (1-\beta) \sum_{i=1}^{t} \beta^{t-i} g_i^2 \in \mathbb{R} \ , \tag{56}$$

and after bias correction

$$\hat{v}_t = \frac{v_t}{1-\beta^t} = \frac{1-\beta}{1-\beta^t} \sum_{i=1}^{t} \beta^{t-i} g_i^2 \in \mathbb{R} \ . \tag{57}$$

Theorem G.5 (Convergence of MEAZO)**.** *Suppose that Assumptions G.1, G.2, G.3 and G.4 hold. Let $q = 1$, $\eta > 0$, $\varepsilon > 0$, and $\beta \in (0,1)$. Let $\{\mathbf{x}_t\}_{t\in[T]}$ be the iterates generated by Algorithm 1, then*

$$\begin{aligned} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F(\mathbf{x}_t)\right\|^2 \le \frac{4R_d\left[F(\mathbf{x}_1) - F_\star\right]}{\eta T} + \frac{4R_d L \varepsilon^2}{\eta T} \\ + \left(\frac{8R_d^2}{\sqrt{1-\beta}} + \frac{2R_d L \eta}{1-\beta}\right)\left[\frac{1}{T}\log\left(1 + \frac{R_d^2}{(1-\beta)\zeta}\right) - \log(\beta)\right] + 2L^2\varepsilon^2 \ . \end{aligned} \tag{58}$$

Proof. By the uniform function-value smoothing bound,

$$F_\varepsilon(\mathbf{x}_1) - F_\star \le F(\mathbf{x}_1) - F_\star + L\varepsilon^2.$$

Moreover, the smoothing-gradient bias gives

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F(\mathbf{x}_t)\right\|^2 \le \frac{2}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 + 2L^2\varepsilon^2.$$

Substituting these two inequalities into (89) yields (58). □

Corollary G.6 (Dimension-dependent convergence rate)**.** *Under the assumptions of Theorem 5.5, choose*

$$\eta = \frac{\eta_0}{\sqrt{T}}, \qquad \varepsilon^2 = \frac{\varepsilon_0^2}{\sqrt{T}}, \qquad \beta = 1 - \frac{1}{T} \ , \tag{59}$$

where $\eta_0, \varepsilon_0 > 0$ and $T \ge 2$. Then

$$\begin{aligned} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F(\mathbf{x}_t)\right\|^2 \le \frac{4R_d\left[F(\mathbf{x}_1) - F_\star\right]}{\eta_0\sqrt{T}} + \frac{4R_d L\varepsilon_0^2}{\eta_0 T} + \frac{2L^2\varepsilon_0^2}{\sqrt{T}} \\ + \frac{8R_d^2 + 2R_d L\eta_0}{\sqrt{T}}\left[\log\left(1 + \frac{R_d^2 T}{\zeta}\right) + 2\right] \ . \end{aligned} \tag{60}$$

Since $R_d = \sqrt{\zeta} + dB = \mathcal{O}(d)$ for dimension-independent B and ζ, it follows that

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F(\mathbf{x}_t)\right\|^2 = \mathcal{O}\left(\frac{d^2 \log\left(1 + d^2 T/\zeta\right)}{\sqrt{T}}\right) = \widetilde{\mathcal{O}}\left(\frac{d^2}{\sqrt{T}}\right) \ , \tag{61}$$

where $\widetilde{\mathcal{O}}$ hides logarithmic factors in d, T, and ζ^{-1}.

Proof. Substituting (59) into (58), using $-\log(1-x) \le x/(1-x)$ for $0 < x < 1$ to get that

$$-T \log\left(1 - \frac{1}{T}\right) \le 2 \ , \tag{62}$$

and collecting terms gives (60). Finally, substituting $R_d = \sqrt{\zeta} + dB = \mathcal{O}(d)$ yields (61). □

G.3 Technical lemmas

Lemma G.7 (Sum of ratios with the denominator being the sum of past numerators, Lemma 5.2 of Défossez et al. (2020))**.** *Let $\beta \in (0, 1]$ and let $(a_t)_{t\ge1}$ be nonnegative. Define $b_t := \sum_{j=1}^{t} \beta^{t-j} a_j$. Then, for every $\rho > 0$ and $T \ge 1$,*

$$\sum_{t=1}^{T} \frac{a_t}{\rho + b_t} \le \log\left(1 + \frac{b_T}{\rho}\right) - T \log(\beta) \ . \tag{63}$$

The following lemma is the scalar MEAZO analogue of the predictable-denominator argument used to control adaptive linear terms. It directly controls the complete conditional expectation in (66), without introducing any shorthand for $\nabla F_\varepsilon(\mathbf{x}_t)$.

Lemma G.8 (Conditional unbiasedness of the ZO estimator)**.** *Conditionally on $\mathcal{F}_{t-1}$, let ξ_t be drawn from the distribution defining F, and let $\mathbf{u}_t$ be independent and uniformly distributed on $\mathbb{S}^{d-1}$. If the relevant expectations are finite, then*

$$\mathbb{E}_{t-1}[\mathbf{G}_t] := \mathbb{E}_{\xi_t, \mathbf{u}_t}\left[\mathbf{G}_t \mid \mathcal{F}_{t-1}\right] = \nabla F_\varepsilon(\mathbf{x}_t). \tag{64}$$

Proof. Conditionally on $\mathcal{F}_{t-1}$, the iterate $\mathbf{x}_t$ is fixed. Taking expectation first over ξ_t and using the objective definition $F(\mathbf{x}) = \mathbb{E}_\xi[f(\mathbf{x}; \xi)]$ turns $\mathbf{G}_t$ into the centered ZO estimator of F. Taking expectation over the i.i.d. uniform perturbations and applying the standard sphere–ball smoothing identity then gives

$$\mathbb{E}_{t-1}[\mathbf{G}_t] = \frac{d}{\varepsilon} \mathbb{E}_{u \sim \mathrm{Unif}(\mathbb{S}^{d-1})}\left[F(\mathbf{x}_t + \varepsilon u) u\right] = \nabla F_\varepsilon(\mathbf{x}_t). \tag{65}$$

□

Lemma G.9 (Control of the adaptive linear term)**.** *Define $\overline{g_t^2} := \mathbb{E}_{t-1}[g_t^2] = \mathbb{E}_{\xi_t, \mathbf{u}_t}\left[g_t^2 \mid \mathcal{F}_{t-1}\right]$ and $\widetilde{v}_t := \beta v_{t-1} + (1-\beta)\overline{g_t^2}$. Let $\widehat{\widetilde{v}}_t := \widetilde{v}_t/(1-\beta^t)$. Under Assumption G.4 and Lemma G.8, for every $t \in \mathbb{N}^*$ and $i \in [d]$,*

$$\mathbb{E}_{t-1}\left[\frac{\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}}{\sqrt{\widehat{v}_t + \zeta}}\right] \ge \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{2\sqrt{\widehat{\widetilde{v}}_t + \zeta}} - 2R_d \sqrt{\frac{1-\beta}{1-\beta^t}} \mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t + \zeta}\right] \ . \tag{66}$$

Consequently,

$$\boxed{\mathbb{E}_{t-1}\left[\alpha_t \left\langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \right\rangle\right] \ge \frac{1}{2R_d} \left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 - 2R_d \sqrt{\frac{1-\beta}{1-\beta^t}} \mathbb{E}_{t-1}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \ .} \tag{67}$$

Proof. Fix $i \in [d]$, and define the following coordinate notations:

$$\nabla_i F_\varepsilon(\mathbf{x}_t) := [\nabla F_\varepsilon(\mathbf{x}_t)]_i \in \mathbb{R}, \qquad G_{t,i} := [\mathbf{G}_t]_i \in \mathbb{R} \ .$$

We then have by Lemma G.8,

$$\nabla_i F_\varepsilon(\mathbf{x}_t) = \mathbb{E}_{t-1}[G_{t,i}] \ . \tag{68}$$

Let us define $\overline{g_t^2} := \mathbb{E}_{t-1}[g_t^2]$, the predictable ($\mathcal{F}_{t-1}$-measurable) second moment estimate $\widetilde{v}_t := \beta v_{t-1} + (1-\beta)\overline{g_t^2}$ and its bias-corrected version $\widehat{\widetilde{v}}_t := \widetilde{v}_t/(1-\beta^t)$. Since $\nabla_i F_\varepsilon(\mathbf{x}_t)$ and $\widehat{\widetilde{v}}_t$ are $\mathcal{F}_{t-1}$-measurable, and since (68), we have

$$\mathbb{E}_{t-1}\left[\frac{\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}}{\sqrt{\widehat{v}_t + \zeta}}\right] = \frac{\nabla_i F_\varepsilon(\mathbf{x}_t)^2}{\sqrt{\widehat{\widetilde{v}}_t + \zeta}} + A_{t,i} \ , \tag{69}$$

where

$$A_{t,i} := \mathbb{E}_{t-1}\left[\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}\left(\frac{1}{\sqrt{\widehat{v}_t+\zeta}} - \frac{1}{\sqrt{\widehat{\widehat{v}}_t+\zeta}}\right)\right] \quad . \tag{70}$$

The goal is now to upper bound $|A_{t,i}|$ term, in order to finally lower bound it.

Let us define the intended denominator

$$D_t := \sqrt{\widehat{v}_t+\zeta}\sqrt{\widehat{\widehat{v}}_t+\zeta}\left(\sqrt{\widehat{v}_t+\zeta}+\sqrt{\widehat{\widehat{v}}_t+\zeta}\right) \tag{71}$$

and reorganize the following difference

$$\frac{1}{\sqrt{\widehat{v}_t+\zeta}} - \frac{1}{\sqrt{\widehat{\widehat{v}}_t+\zeta}} = \frac{\widehat{\widehat{v}}_t - \widehat{v}_t}{D_t} = \frac{1-\beta}{1-\beta^t}\frac{\overline{g_t^2}-g_t^2}{D_t} = c_t\frac{\overline{g_t^2}-g_t^2}{D_t} \quad .$$

where we introduced $c_t := (1-\beta)/(1-\beta^t)$ to make notation easier. Therefore,

$$A_{t,i} = c_t\mathbb{E}_{t-1}\left[\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}\frac{\overline{g_t^2}-g_t^2}{D_t}\right] \quad . \tag{72}$$

Since g_t^2 and $\overline{g_t^2}$ are both non-negative, a simple triangle inequality gives us

$$\left|\overline{g_t^2} - g_t^2\right| \leq \overline{g_t^2} + g_t^2 \quad .$$

Then, we can write $|A_{t,i}|$ into an addition of two terms

$$|A_{t,i}| \overset{(72)}{\leq} c_t\mathbb{E}_{t-1}\left[\frac{|\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}|\overline{g_t^2}}{D_t}\right] + c_t\mathbb{E}_{t-1}\left[\frac{|\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}|g_t^2}{D_t}\right] \quad . \tag{73}$$

i) For the term containing $\overline{g_t^2}$. We simplify the denominator using

$$\begin{aligned} &\sqrt{\widehat{v}_t+\zeta} + \sqrt{\widehat{\widehat{v}}_t+\zeta} \geq \sqrt{\widehat{\widehat{v}}_t+\zeta} \\ \implies D_t &\overset{(71)}{\geq} \sqrt{\widehat{v}_t+\zeta}\left(\widehat{\widehat{v}}_t+\zeta\right) \iff \frac{1}{D_t} \leq \frac{1}{\sqrt{\widehat{v}_t+\zeta}\left(\widehat{\widehat{v}}_t+\zeta\right)} \quad . \end{aligned} \tag{74}$$

ii) For the term containing g_t^2. We simplify the denominator using

$$\begin{aligned} &\sqrt{\widehat{v}_t+\zeta} + \sqrt{\widehat{\widehat{v}}_t+\zeta} \geq \sqrt{\widehat{v}_t+\zeta} \\ \implies D_t &\overset{(71)}{\geq} (\widehat{v}_t+\zeta)\sqrt{\widehat{\widehat{v}}_t+\zeta} \iff \frac{1}{D_t} \leq \frac{1}{(\widehat{v}_t+\zeta)\sqrt{\widehat{\widehat{v}}_t+\zeta}} \quad . \end{aligned} \tag{75}$$

Consequently, by combining (73), (74) and (75) we get

$$|A_{t,i}| \leq \mathbb{E}_{t-1}\left[\underbrace{c_t\frac{|\nabla_i F_\varepsilon(\mathbf{x}_t)||G_{t,i}|\overline{g_t^2}}{\sqrt{\widehat{v}_t+\zeta}\left(\widehat{\widehat{v}}_t+\zeta\right)}}_{=:\kappa_{t,i}} + \underbrace{c_t\frac{|\nabla_i F_\varepsilon(\mathbf{x}_t)||G_{t,i}|g_t^2}{(\widehat{v}_t+\zeta)\sqrt{\widehat{\widehat{v}}_t+\zeta}}}_{=:\rho_{t,i}}\right] \quad , \tag{76}$$

Like in Défossez et al. (2020) and Ward et al. (2020), let us use the following version of Young's inequality to bound $\kappa_{t,i}$ and $\rho_{t,i}$:

$$xy \leq \frac{\lambda}{2}x^2 + \frac{y^2}{2\lambda}, \quad \forall x, y \in \mathbb{R}, \lambda > 0 \quad . \tag{77}$$

Upper bounding $\kappa_{t,i}$. Let us choose

$$x := \frac{|\nabla_i F_\varepsilon(\mathbf{x}_t)|}{\sqrt{\widehat{\widehat{v}}_t + \zeta}}, \quad y := \frac{c_t |G_{t,i}| \overline{g_t^2}}{\sqrt{\widehat{v}_t + \zeta}\sqrt{\widehat{\widehat{v}}_t + \zeta}}, \quad \lambda := \frac{\sqrt{\widehat{\widehat{v}}_t + \zeta}}{2} \ .$$

Applying Young's inequality (77) gives

$$\kappa_{t,i} = xy \le \frac{\lambda}{2}x^2 + \frac{1}{2\lambda}y^2 = \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t + \zeta}} + \frac{c_t^2 G_{t,i}^2 \left(\overline{g_t^2}\right)^2}{\left(\widehat{v}_t + \zeta\right)\left(\widehat{\widehat{v}}_t + \zeta\right)^{3/2}} \ . \tag{78}$$

Since ζ and v_{t-1} are non-negative,

$$\widehat{\widehat{v}}_t + \zeta \ge \widehat{\widehat{v}}_t := \frac{1}{1-\beta^t}\beta v_{t-1} + \frac{1-\beta}{1-\beta^t}\overline{g_t^2} \ge c_t \overline{g_t^2} \ , \tag{79}$$

we can deduce a simplification of the second term in (78)

$$\frac{c_t^2 \left(\overline{g_t^2}\right)^2}{\left(\widehat{\widehat{v}}_t + \zeta\right)^{3/2}} \le \sqrt{c_t \overline{g_t^2}} \ .$$

Therefore,

$$\kappa_{t,i} \overset{(78)}{\le} \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t + \zeta}} + \sqrt{c_t \overline{g_t^2}} \frac{G_{t,i}^2}{\widehat{v}_t + \zeta} \ .$$

Taking conditional expectation and using Assumption G.4 implying that $\sqrt{\overline{g_t^2}} = \sqrt{\mathbb{E}_{t-1}[g_t^2]} \le R_d - \sqrt{\zeta} \le R_d$ gives us

$$\mathbb{E}_{t-1}[\kappa_{t,i}] \le \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t + \zeta}} + R_d \sqrt{\frac{1-\beta}{1-\beta^t}} \, \mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t + \zeta}\right] \ . \tag{80}$$

Upper bounding $\rho_{t,i}$. Let us choose

$$x := \frac{|\nabla_i F_\varepsilon(\mathbf{x}_t)| \, |g_t|}{\sqrt{\widehat{\widehat{v}}_t + \zeta}}, \quad y := \frac{c_t |G_{t,i}| |g_t|}{\widehat{v}_t + \zeta}, \quad \lambda := \frac{\sqrt{\widehat{\widehat{v}}_t + \zeta}}{2\overline{g_t^2}} \ .$$

When $\overline{g_t^2} > 0$, we have $xy = \rho_{t,i}$. Applying Young's inequality (77) gives

$$\rho_{t,i} \le \frac{\lambda}{2}x^2 + \frac{1}{2\lambda}y^2 = \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2 g_t^2}{4\overline{g_t^2}\sqrt{\widehat{\widehat{v}}_t + \zeta}} + \frac{c_t^2 \overline{g_t^2} G_{t,i}^2 g_t^2}{\sqrt{\widehat{\widehat{v}}_t + \zeta}\left(\widehat{v}_t + \zeta\right)^2} \ .$$

Taking conditional expectation and using $\mathbb{E}_{t-1}[g_t^2] =: \overline{g_t^2}$ yields

$$\mathbb{E}_{t-1}[\rho_{t,i}] \le \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t + \zeta}} + \frac{c_t^2 \overline{g_t^2}}{\sqrt{\widehat{\widehat{v}}_t + \zeta}} \mathbb{E}_{t-1}\left[\frac{G_{t,i}^2 g_t^2}{\left(\widehat{v}_t + \zeta\right)^2}\right] \ . \tag{81}$$

Since ζ and v_{t-1} are non-negative,

$$\widehat{v}_t + \zeta \ge \widehat{v}_t := \frac{1}{1-\beta^t}\beta v_{t-1} + \frac{1-\beta}{1-\beta^t} g_t^2 \ge c_t g_t^2$$

we have

$$\frac{c_t^2 g_t^2}{\left(\widehat{v}_t+\zeta\right)^2} \leq \frac{c_t}{\widehat{v}_t+\zeta} \ .$$

Therefore, we can simplify the term inside the conditional expectation in (81)

$$\mathbb{E}_{t-1}[\rho_{t,i}] \leq \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t+\zeta}} + \frac{c_t\overline{g_t^2}}{\sqrt{\widehat{\widehat{v}}_t+\zeta}}\mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t+\zeta}\right] \ . \tag{82}$$

Moreover, since $\widehat{\widehat{v}}_t+\zeta \geq c_t\overline{g_t^2}$ as shown in (79), we obtain

$$\frac{c_t\overline{g_t^2}}{\sqrt{\widehat{\widehat{v}}_t+\zeta}} \leq \sqrt{c_t\overline{g_t^2}} \ .$$

Using the above to simplify the factor in front of the conditional expectation in (82) and using $\sqrt{\overline{g_t^2}} = \sqrt{\mathbb{E}_{t-1}[g_t^2]} \leq R_d - \sqrt{\zeta} \leq R_d$ finally gives

$$\mathbb{E}_{t-1}[\rho_{t,i}] \leq \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{4\sqrt{\widehat{\widehat{v}}_t+\zeta}} + R_d\sqrt{c_t}\,\mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t+\zeta}\right] \ . \tag{83}$$

Note that if $\overline{g_t^2}=0$, then $g_t=0$ conditionally almost surely, and hence $\rho_{t,i}=0$.

Difference with Défossez et al. (2020) in the choice of x, y, λ for bounding ρ. Unlike in the coordinate-wise adaptive setting of Adam, the stochastic quantity driving the scalar second moment estimate is g_t^2, whereas the i-th coordinate of the MEAZO estimator is $G_{t,i} = g_t u_{t,i}$. We adapt the application of Young's inequality so that the remainder retains the factor $G_{t,i}^2$. This distinction is essential when summing over the coordinates, since

$$\sum_{i=1}^{d} G_{t,i}^2 = \|\mathbf{G}_t\|^2 = g_t^2 \ .$$

Consequently, the coordinate-wise remainders collapse to a single scalar adaptive ratio instead of producing an additional factor d. Beside working with the smoothed objective F_ε, *this is the only structural modification needed to adapt the argument of Défossez et al. (2020) to the scalar second moment estimate used by MEAZO.*

Combining the bounds to lower bound $A_{t,i}$. Combining the bounds on $\kappa_{t,i}$ and $\rho_{t,i}$ gives

$$|A_{t,i}| \overset{(76)}{\leq} \mathbb{E}_{t-1}[\kappa_{t,i}+\rho_{t,i}] \leq \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{2\sqrt{\widehat{\widehat{v}}_t+\zeta}} + 2R_d\sqrt{c_t}\,\mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t+\zeta}\right] \ .$$

So we can deduce the following lower bound on $A_{t,i}$

$$A_{t,i} \geq -\frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{2\sqrt{\widehat{\widehat{v}}_t+\zeta}} - 2R_d\sqrt{c_t}\,\mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t+\zeta}\right] \ . \tag{84}$$

Recall the decomposition of (69)

$$\mathbb{E}_{t-1}\left[\frac{\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}}{\sqrt{\widehat{v}_t+\zeta}}\right] = \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{\sqrt{\widehat{\widehat{v}}_t+\zeta}} + A_{t,i} \ .$$

Substituting (84) into (G.3) yields

$$\boxed{\mathbb{E}_{t-1}\left[\frac{\nabla_i F_\varepsilon(\mathbf{x}_t) G_{t,i}}{\sqrt{\widehat{v}_t+\zeta}}\right] \geq \frac{\left(\nabla_i F_\varepsilon(\mathbf{x}_t)\right)^2}{2\sqrt{\widehat{\widehat{v}}_t+\zeta}} - 2R_d\sqrt{c_t}\,\mathbb{E}_{t-1}\left[\frac{G_{t,i}^2}{\widehat{v}_t+\zeta}\right]} \ . \tag{85}$$

Summing (85) over $i \in [d]$ and using the fact that the adaptive denominator is scalar gives

$$\sum_{i=1}^{d} \frac{(\nabla_i F_\varepsilon(\mathbf{x}_t))^2}{2\sqrt{\widehat{\widehat{v}}_t + \zeta}} = \frac{\|\nabla F_\varepsilon(\mathbf{x}_t)\|^2}{2\sqrt{\widehat{\widehat{v}}_t + \zeta}} \ . \tag{86}$$

Moreover, since $\mathbf{G}_t = g_t \mathbf{u}_t$ and $\|\mathbf{u}_t\| = 1$ from Assumption G.1, we have $\sum_{i=1}^{d} G_{t,i}^2 = \|\mathbf{G}_t\|^2 = g_t^2$, and therefore

$$\mathbb{E}_{t-1}\left[\alpha_t \langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \rangle\right] \geq \frac{\|\nabla F_\varepsilon(\mathbf{x}_t)\|^2}{2\sqrt{\widehat{\widehat{v}}_t + \zeta}} - 2R_d \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}_{t-1}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \ . \tag{87}$$

Under Assumption G.4, we further have $\widehat{\widehat{v}}_t + \zeta \leq R_d^2$. Indeed, Assumption G.4 yields

$$\widehat{\widehat{v}}_t + \zeta \leq \left(R_d - \sqrt{\zeta}\right)^2 + \zeta \leq R_d^2 \ ,$$

where the last inequality follows from $R_d \geq \sqrt{\zeta}$. Therefore,

$$\frac{1}{2\sqrt{\widehat{\widehat{v}}_t + \zeta}} \geq \frac{1}{2R_d} \ .$$

and consequently

$$\boxed{\mathbb{E}_{t-1}\left[\alpha_t \langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \rangle\right] \geq \frac{1}{2R_d} \|\nabla F_\varepsilon(\mathbf{x}_t)\|^2 - 2R_d \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}_{t-1}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right]} \ . \tag{88}$$

□

G.4 Convergence of the smoothed objective

Theorem G.10 (Convergence of the smoothed objective)**.** *Suppose that Assumptions G.1, G.2, G.3 and G.4 hold. Let $q = 1$, $\eta > 0$, $\varepsilon > 0$, and $\beta \in (0, 1)$. Let $\{\mathbf{x}_t\}_{t \in [T]}$ be the iterates generated by Algorithm 1, then*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\|\nabla F_\varepsilon(\mathbf{x}_t)\|^2 \leq \frac{2R_d[F_\varepsilon(\mathbf{x}_1) - F_\star]}{\eta T} + \left(\frac{4R_d^2}{\sqrt{1-\beta}} + \frac{R_d L \eta}{1-\beta}\right)\left[\frac{1}{T}\log\left(1 + \frac{R_d^2}{(1-\beta)\zeta}\right) - \log(\beta)\right] \ . \tag{89}$$

Proof. We divide the proof in several bold steps.

(i) L-Smoothness descent for the smoothed objective. By Assumption G.3, F_ε is L-smooth. Using (52),

$$\begin{aligned} F_\varepsilon(\mathbf{x}_{t+1}) &\leq F_\varepsilon(\mathbf{x}_t) + \langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{L}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\ &= F_\varepsilon(\mathbf{x}_t) - \eta\alpha_t \langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \rangle + \frac{L\eta^2}{2}\alpha_t^2 \|\mathbf{G}_t\|^2 \ . \end{aligned} \tag{90}$$

(ii) Control of the adaptive quadratic term. From (53) and (54),

$$\boxed{\alpha_t^2 \|\mathbf{G}_t\|^2 = \frac{\|g_t \mathbf{u}_t\|^2}{(\sqrt{\hat{v}_t + \zeta})^2} \overset{(43)\ \&\ \zeta > 0}{=} \frac{g_t^2}{\hat{v}_t + \zeta}} \ , \tag{91}$$

Then

$$\alpha_t^2 \|\mathbf{G}_t\|^2 \overset{\hat{v}_t \geq v_t}{\leq} \frac{g_t^2}{v_t + \zeta} = \frac{1}{1-\beta}\left[\frac{a_t}{b_t + \rho}\right]$$

where we identify

$$a_t := g_t^2, \qquad b_t := \sum_{j=1}^{t} \beta^{t-j} g_j^2 \overset{(56)}{=} \frac{v_t}{1-\beta}, \qquad \rho := \frac{\zeta}{1-\beta} \ . \tag{92}$$

Applying Lemma G.7 to the sequences in (92) gives

$$\sum_{t=1}^{T} \alpha_t^2 \left\|\mathbf{G}_t\right\|^2 \le \frac{1}{1-\beta} \left[\log\left(1 + \frac{v_T}{\zeta}\right) - T\log(\beta)\right]. \tag{93}$$

Assumption G.4 and (56) imply $v_T \le (R_d - \sqrt{\zeta})^2 \left(1-\beta^T\right) \le (R_d - \sqrt{\zeta})^2$. Therefore, after taking expectation,

$$\boxed{\sum_{t=1}^{T} \mathbb{E}\left[\alpha_t^2 \left\|\mathbf{G}_t\right\|^2\right] \le \frac{1}{1-\beta} \left[\log\left(1 + \frac{(R_d - \sqrt{\zeta})^2}{\zeta}\right) - T\log(\beta)\right].} \tag{94}$$

(iii) Control of the adaptive linear term. The remaining term in (90) is $\alpha_t \langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \rangle$. Applying (67) of Lemma G.9 gives

$$\boxed{\mathbb{E}_{t-1}\left[\alpha_t \left\langle \nabla F_\varepsilon(\mathbf{x}_t), \mathbf{G}_t \right\rangle\right] \ge \frac{1}{2R_d} \left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 - 2R_d \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}_{t-1}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \ .} \tag{95}$$

To control the cumulative remainder without discarding the bias-correction factor, observe that

$$\begin{aligned} \sqrt{\frac{1-\beta}{1-\beta^t}}\, \frac{g_t^2}{\widehat{v}_t + \zeta} &= \frac{\sqrt{(1-\beta)(1-\beta^t)}\, g_t^2}{v_t + (1-\beta^t)\zeta} \\ &\le \frac{\sqrt{1-\beta}\, g_t^2}{v_t + (1-\beta)\zeta} = \frac{1}{\sqrt{1-\beta}} \frac{a_t}{b_t + \zeta} \ , \end{aligned} \tag{96}$$

where we used $1-\beta^t \le 1$ and that $1-\beta^t \ge 1-\beta$. We reuse the definitions of a_t and b_t from (92). Applying Lemma G.7, using $v_T \le (R_d - \sqrt{\zeta})^2$, and taking expectations gives

$$\boxed{\sum_{t=1}^{T} \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \le \frac{1}{\sqrt{1-\beta}} \left[\log\left(1 + \frac{(R_d - \sqrt{\zeta})^2}{(1-\beta)\zeta}\right) - T\log(\beta)\right] \ .} \tag{97}$$

(iv) Summation of the descent inequality. Taking conditional expectation in (90) and applying (95) gives

$$\begin{aligned} \frac{\eta}{2R_d} \left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 \le F_\varepsilon(\mathbf{x}_t) - \mathbb{E}_{t-1}[F_\varepsilon(\mathbf{x}_{t+1})] + 2R_d\eta \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}_{t-1}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \\ + \frac{L\eta^2}{2} \mathbb{E}_{t-1}[\alpha_t^2 \left\|\mathbf{G}_t\right\|^2] \ . \end{aligned} \tag{98}$$

Taking total expectation and summing from step $t = 1$ to T, we obtain

$$\begin{aligned} \frac{\eta}{2R_d} \sum_{t=1}^{T} \mathbb{E}\left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 \le F_\varepsilon(\mathbf{x}_1) - \mathbb{E}[F_\varepsilon(\mathbf{x}_{T+1})] + 2R_d\eta \sum_{t=1}^{T} \sqrt{\frac{1-\beta}{1-\beta^t}}\, \mathbb{E}\left[\frac{g_t^2}{\widehat{v}_t + \zeta}\right] \\ + \frac{L\eta^2}{2} \sum_{t=1}^{T} \mathbb{E}[\alpha_t^2 \left\|\mathbf{G}_t\right\|^2] \ . \end{aligned} \tag{99}$$

By Assumption G.2, $F_\varepsilon(\mathbf{x}_{T+1}) \ge F_\star$. Applying (97) and (94) gives

$$\begin{aligned} \frac{\eta}{2R_d} \sum_{t=1}^{T} \mathbb{E}\left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 \le F_\varepsilon(\mathbf{x}_1) - F_\star + \frac{2R_d\eta}{\sqrt{1-\beta}} \left[\log\left(1 + \frac{(R_d - \sqrt{\zeta})^2}{(1-\beta)\zeta}\right) - T\log(\beta)\right] \\ + \frac{L\eta^2}{2(1-\beta)} \left[\log\left(1 + \frac{(R_d - \sqrt{\zeta})^2}{\zeta}\right) - T\log(\beta)\right] \ . \end{aligned} \tag{100}$$

Moreover, since $(R_d - \sqrt{\zeta})^2 \leq R_d^2$ and $1 - \beta \leq 1$, both logarithmic terms of (100) are bounded by

$$\log\left(1 + \frac{R_d^2}{(1-\beta)\zeta}\right) \quad .$$

Dividing by $\eta T/(2R_d)$ and factoring this common logarithmic term yields a bound with the same structure as in Theorem 2 of Défossez et al. (2020)

$$\boxed{\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left\|\nabla F_\varepsilon(\mathbf{x}_t)\right\|^2 \leq \frac{2R_d[F_\varepsilon(\mathbf{x}_1) - F_\star]}{\eta T} + \left(\frac{4R_d^2}{\sqrt{1-\beta}} + \frac{R_d L \eta}{1-\beta}\right)\left[\frac{1}{T}\log\left(1 + \frac{R_d^2}{(1-\beta)\zeta}\right) - \log(\beta)\right].}$$

□