HiSo: Efficient Federated Zeroth-Order Optimization via Hessian-Informed Acceleration and Scalar-Only Communication

Zhe Li ZL4063@RIT.EDU
Department of Computing and Information Sciences Ph.D., Rochester Institute of Technology

Bicheng Ying YBC@GOOGLE.COM
Google Inc.

Zidong Liu Z.LIU@COMBOCURVE.COM
ComboCurve Inc.

Chaosheng Dong CHAOSD@AMAZON.COM
Amazon.com Inc.

Haibo Yang HBYCIS@RIT.EDU
Department of Computing and Information Sciences Ph.D., Rochester Institute of Technology

Abstract

Recent Federated Learning (FL) with dimension-free communication greatly reduce communication by transmitting only scalars via zeroth-order stochastic gradient descent (ZO-SGD), making them well-suited for federated fine-tuning of Large Language Models (LLMs). Yet, the high variance in ZO gradient estimation slows convergence. While Hessian information can accelerate convergence, integrating it into FL is challenging due to clients' restrictions on local data and the need to maintain the dimension-free communication. To address this, we first introduce a generalized scalar-only communication FL framework decoupling dimension-free communication from standard ZO-SGD, enabling the integration of advanced optimizers. Based on it, we propose HiSo, a new FL method via Hessian-informed ZO optimization and Scalar-only communication. Specifically, it uses global curvature to accelerate convergence while retaining the minimal communication. Theoretically, we establish convergence guarantees independent of Lipschitz L and model dimension d.

1. Introduction

Federated fine-tuning has been promising for deploying large language models (LLMs) across devices while preserving data privacy [5, 12, 23, 45]. Yet, LLMs' massive scale presents scalability challenges for federated fine-tuning, mainly due to the prohibitive communication cost of transmitting high-dimensional model updates [10, 35]. To mitigate this issue, recent work has proposed using zeroth-order optimization (ZOO) to enable dimension-free communication in FL [18, 25]. In particular, DeComFL [18] encodes uplink and downlink communication using shared random seeds and scalar-only updates, achieving communication independent of model dimension. This is especially attractive for federated LLM fine-tuning, where communication is a dominant bottleneck.

Yet, the practical effectiveness of ZOO-based FL remains limited due to its seriously slow convergence. A key factor is that LLMs often exhibit heterogeneous and anisotropic curvature across their parameter space [2, 14, 39], making it difficult for vanilla ZO-SGD to adaptively scale updates. While prior work has shown that second-order information (e.g., Hessians or their diagonal approximations) can accelerate convergence [11, 14, 40, 46], estimating Hessian approximation and applying such curvature-aware techniques in FL are non-trivial. The challenge becomes even more pronounced in dimension-free communication frameworks, where transmitting any Hessian-related information reintroduces costs that scale with model size - directly contradicting the goal of scalar-only communication. This tension leads to our research question: *Can we accelerate federated ZO fine-tuning while preserving dimension-free communication?* To answer this question,

- We propose a flexible FL framework with scalar-only communication in both uplink and downlink, which supports a broader class of optimization algorithms beyond vanilla ZO-SGD.
- Under this framework, we develop HiSo, a fast federated fine-tuning method via Hessian-informed zeroth-order optimization and Scalar-only communication. It utilizes global Hessian information to speed up convergence while preserving dimension-free communication.
- Theoretically, we propose a novel condition to get a tight estimation of the variance of Hessian-informed ZO gradient under low-effective rank and whitening assumptions. With this treatment, we prove that HiSo can achieve a convergence rate independent of model dimension and function smoothness in non-convex settings, marking the first such result for ZO methods in FL.

2. A Generalized Scalar-Only Communication in FL Framework

2.1. Zeroth-Order SGD and Scalar Representations

We focus on the randomized gradient estimator (RGE) for performing ZO gradient estimation, also called Simultaneous Perturbation Stochastic Approximation [25, 31]. Given a scalar-valued loss function $f(x)$ where $x \in \mathbb{R}^d$, the forward-style RGE is

$$\hat{\nabla} f(x) = \frac{1}{\mu}\big(f(x+\mu u) - f(x)\big)u, \;\; u \sim \mathcal{N}(0, I_d), \tag{1}$$

where u is a perturbation vector from a Gaussian distribution and $\mu > 0$ is a smoothing parameter.

An intriguing attribute of RGE is its efficient representation by only two scalars. First, we introduce a gradient scalar $g := \frac{1}{\mu}(f(x+\mu u) - f(x)) \in \mathbb{R}$ serving as a scaling constant capturing the directional derivative. g can also be explained as an approximate value for the directional gradient. Second, due to the deterministic nature of pseudo-random number generators, the random direction vector $u \in \mathbb{R}^d$ can be uniquely determined by a random seed s. Hence, the estimated gradient $\hat{\nabla} f(x)$ can be expressed by two scalars. Crucially, this compact representation enhances the efficiency of model updates in ZOO frameworks. To illustrate, consider ZO-SGD update rule:

$$x_{R+1} = x_R - \frac{\eta}{\mu}\big(f(x_R + \mu u_R) - f(x_R)\big)u_R = x_R - \eta g_R u_R = \cdots = x_0 - \eta \sum_{r=0}^{R} g_r u_r \tag{2}$$

This implies that, given the initial point x_0, a few number of gradient scalars $\{g_r\}$ and random seeds $\{s_r\}$ are sufficient to reconstruct x_R, irrespective of the dimensionality d of x. This representation will play a crucial role in the dimension-free communication FL algorithm that follows.

2.2. Federated Learning with Dimension-Free Communication

We consider a FL scenario with M clients, each owning a local loss function f_i. The goal is to collaboratively minimize the global loss function across all clients without sharing their private data:

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}) = \min_{\boldsymbol{x}\in\mathbb{R}^d} \frac{1}{M}\sum_{i=1}^{M} f_i(\boldsymbol{x}), \quad \text{ where } f_i(\boldsymbol{x}) := \mathbb{E}\left[F_i(\boldsymbol{x}; \xi_i)\right]. \tag{3}$$

The core of dimension-free communication in FL [18] is using the scalar representation of ZO-SGD to avoid transmitting the full models. Consider the following global model, update rule: $x_{r,\tau}^{(i)}$ is client i's model at the r-th round and τ-th local update step, x_r is the r-th global model:

$$x_{r+1} = \frac{1}{|C_r|}\sum_{i\in C_r} x_{r,\tau}^{(i)} = x_r + \frac{1}{|C_r|}\sum_{i\in C_r}(x_{r,\tau}^{(i)} - x_r) = x_r - \eta\frac{1}{|C_r|}\sum_{i\in C_r}\sum_{k=0}^{\tau-1} g_{r,k}^{(i)} u_{r,k}, \tag{4}$$

where C_r is the set of sampled clients in the r-th round, $u_{r,k}$ are generated by shared random seeds across all clients, ensuring that all clients move along consistent directions. It enables that the global aggregation step in the server is simply computing an average of the gradient scalars: $g_{r,k} = \frac{1}{|C_r|}\sum_{i\in C_r} g_{r,k}^{(i)}$ from the local gradient scalar $g_{r,k}^{(i)} = \big(f_i(x_{r,k}^{(i)} + \mu u_{r,k}) - f_i(x_{r,k}^{(i)})\big)/\mu$.

Uplink. From Eq. (4), sampled clients only transmit $g_{r,k}^{(i)}$ to the server for global aggregation.

Downlink. ZO scalar representation only captures relative updates, so it is crucial that the server and all clients start from the same starting point. To achieve this, a model-reset mechanism is introduced: after local updates, all sampled clients reset the local model to the initial model, which is the global server model by induction. With this mechanism, the downlink can be conceptualized similarly to Eq. (4), with the distinction that clients may miss participation in multiple rounds.

Unlike the standard FL, model reconstruction is used for catching the current global model through global gradient scalars and random seeds from missed rounds. Hence, the server necessitates recording the client's last participation round, historical random seeds, and the global gradient scalars. We show the process in Fig. 1.

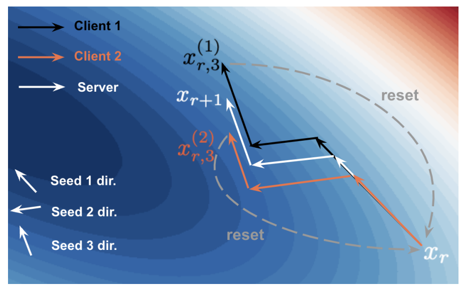

Figure 1: One-round update with 2 sampled clients and 3 local updates. They share the same direction for each local update with different lengths. To arrive x_{r+1} for both clients, it requires **7 steps**: 3 local updates, 1 reset, 3 updates with global values.

2.3. Generalized Scalar-Only Communication in FL

In the work by Li et al. [18], the inherent dependency on ZO-SGD limits its applicability and the full potential of its dimension-free communication framework. One of our key contributions is observing that the crucial element is not the specific choice of ZO-SGD, but rather the basic use of scalar representations. Specifically, by maintaining records of their respective states with the update constructed by these scalar representations, the server and clients can effectively accommodate a wider range of optimization algorithms with dimension-free communication. To address this, we present a generalized formulation allowing for the integration of various optimization techniques. In this framework, communication proceeds as follows: clients send $\{\Delta x_{r,k}^{(i)}\}_{k=1}^{K}$ to server for aggregation, and the server sends the aggregated update Δx_r to clients for reconstruction. The dimension-independent property holds if client-side update $\Delta x_{r,k}^{(i)}$ and the server-side aggregated update Δx_r can be represented by scalars. Note a persistent state may be required to reconstruct Δx_r with r_l as the last participated round.

3. Hessian-informed Scalar-only Communication in FL (HiSo)

3.1. Find a Better Ascent $\Delta x_{r,k}^{(i)}$ Direction

We use the proposed generalized framework to design a novel method superior to ZO-SGD based FL while retaining dimension-free communication. The core challenge in the preceding framework is identifying an effective ascent direction $\Delta x_{r,k}^{(i)}$ that is constructible solely from scalar values and current state. While ZO-SGD meets these requirements, a superior alternative can be found.

Recall that the ZO methods' slow convergence is due to its dependency on random search directions [20]. More specifically, recall the Eq. (1) with $u \sim \mathcal{N}(0, I)$, which uniformly searches all directions in the $\mathbb{R}^d$ space, is the update direction regardless of the scalar g. A natural extension is that we can guide the search direction with an invertible matrix H_r. Suppose H_r is given, the Line 11 in Algorithm 1 can be formulated as the following sub-optimization problem

$$\min_{g\in\mathbb{R}} \ \|\nabla f_i(x_{r,k}^{(i)}) - \Delta x_{r,k}^{(i)}\|_2^2 \quad \text{s.t.}\Delta x_r^{(i)} = g\cdot H_r^{-1/2}u_{r,k},\ \ u_{r,k}\sim\mathcal{N}(0,I_d)\in\mathbb{R}^{d\times 1} \tag{5}$$

It will be clear later why we use this strange $H_r^{-1/2}$ notation instead of H_r directly. Solving the above least-squares problem, we have $g^o = (u_{r,k}^\top H_r^{-1}u_{r,k})^{-1}u_{r,k}^\top H_r^{-1/2}\nabla f_i(x_{r,k}^{(i)})$.

Note $(u^\top H^{-1}u)^{-1}$ is a scalar independent of iterates $\boldsymbol{x}_{r,k}^{(i)}$, so we absorb it into the learning rate. Next, $u_{r,k}^\top H_r^{-1/2}\nabla f_i(x_{r,k}^{(i)}) = \frac{1}{\mu}\big(f_i(x_{r,k}^{(i)}+\mu H_r^{-1/2}u_{r,k}) - f_i(x_{r,k}^{(i)})\big) + \mathcal{O}(\mu)$. Hence, we obtain

$$\boxed{\Delta x_{r,k}^{(i)} = \frac{1}{\mu}\big(f_i(x_{r,k}^{(i)}+\mu H_r^{-1/2}u_{r,k}) - f_i(x_{r,k}^{(i)})\big)H_r^{-1/2}u_{r,k}} \tag{6}$$

Now it should be clear why we use the notation $H_r^{-1/2}$ after we take the expectation of $\Delta x_{r,k}^{(i)}$:

$$\mathbb{E}\,\Delta x_{r,k}^{(i)} \approx \mathbb{E}\,H_r^{-1/2}u_{r,k}u_{r,k}^\top H_r^{-1/2}\nabla f_i(x_{r,k}^{(i)}) = H_r^{-1}\nabla f_i(x_{r,k}^{(i)}) \tag{7}$$

When H_r is well-approximated Hessian matrix, the expectation of gradient descent follows the Newton-style gradient descent [3]. The first-order counterpart of $\Delta x_{r,k}^{(i)}$ is called natural gradient since it can be viewed as a pre-conditioned gradient [1]. Recalling the linear transformation property of Gaussian Distribution, the update equation 6 can be more concisely written as the following form

$$\Delta x_{r,k}^{(i)} = \frac{1}{\mu}[f_i(x_{r,k}^{(i)}+\mu z_{r,k}) - f_i(x_{r,k}^{(i)})]z_{r,k},\quad z_{r,k}\sim\mathcal{N}(0,H_r^{-1}) \tag{8}$$

This also aligns with recent Hessian-Aware ZOO work by Ye et al. [41] and Zhao et al. [46].

3.2. Learning Global Curvature without Extra Communication Cost

A follow-up question for the above formulation is how to find this H_r matrix. One plausible approach is, again, utilizing the ZO gradient estimators to approximate directional second derivatives

$$u^\top\nabla^2F(x)u \approx \frac{F(x+\mu u)+F(x-\mu u)-2F(x)}{2\mu^2},\quad u\sim\mathcal{N}(0,I_d) \tag{9}$$

Yet, this method has two limitations: 1) It requires an extra function evaluation per direction and extra communication; 2) forming a full $d\times d$ Hessian is costly and unnecessary. Instead, we only seek a diagonal preconditioner, akin to Adam's per-coordinate scaling [14]. Recall the global update $\Delta x_{r,k}$ approximates the gradient value and can be constructed by scalars only as discussed before. Further, this value is needed for reconstruction. Thus, we have a free variable to approximate the diagonal Hessian by the following proposed rule. We only update the Hessian at the beginning of one communication round with τ-local updates followed by the exponential moving averaging (EMA).

$$\begin{aligned} H_{r+1} = H_{r,\tau} =& (1-\nu)H_{r,\tau-1} + \nu\frac{1}{m}\sum_{i\in S_r}\mathrm{Diag}([\Delta x_{r,\tau}]^2+\epsilon I)\\ &\vdots\\ H_{r,1} =& (1-\nu)H_r + \nu\frac{1}{m}\sum_{i\in S_r}\mathrm{Diag}([\Delta x_{r,0}]^2+\epsilon I), \end{aligned} \tag{10}$$

where ϵ is a small value to make sure that H_{r+1} is strictly positive definite. This Adam-style method, similar to its first-order counterparts [28], has two advantages: 1) the diagonal matrix approximation avoids the d^2 storage cost of the Hessian. 2) the vector $\Delta x_{r,k}$ can be represented by the scalars, so the server and clients can rebuild this global Hessian without extra communication cost.

3.3. Putting Together to Establish HiSo

HiSo is established by substituting the previously determined ascent direction and the global Hessian learning method into our scalars-only communication framework. **To better elucidate the funda-**

mental HiSo with brevity, we use a simplified case where one local update occurs per round ($\tau = 1$). The following equation is for one round update of one client. r_l is the last participated round, $x_r^{(i)}$ is i-th client's model at communication round r and we omit the k for local-update while x_r is the global/server model. The same notation conventions apply for $g_r^{(i)}$, g_r, $\Delta x_r^{(i)}$ and Δx_r.

$$\left.\begin{aligned}&\text{for } t = r_l, \cdots, r-1:\\ &\qquad \Delta x_t = g_t H_t^{-1/2} u_t, \quad u_t \Leftarrow \mathcal{N}(\mathrm{seed_t})\\ &\qquad x_{t+1}^{(i)} = x_t^{(i)} - \eta \Delta x_t\\ &\qquad H_{t+1} = (1-v)H_t + \nu \mathrm{Diag}([\Delta x_t]^2 + \epsilon I)\end{aligned}\right\}\text{(Reconstruct States for the Missing Rounds)}$$

$$\left.\begin{aligned}&\Delta x_r^{(i)} = \frac{1}{\mu}[f_i(x_r^{(i)} + \mu H_r^{-1/2} u_r) - f_i(x_r^{(i)})] H_r^{-1/2} u_r\\ &x_{r+1}^{(i)} = x_r - \eta \Delta x_r^{(i)}\\ &x_{r+1}^{(i)} \Leftarrow x_r \quad \text{(reset)}\end{aligned}\right\}\text{(Client Local Update)}$$

$$\left.\Delta x_r = \frac{1}{|C_r|}\sum_{i\in C_r} \Delta x_r^{(i)} = \left(\frac{1}{|C_r|}\sum_{i\in C_r} g_r^{(i)}\right) H_r^{-1/2} u_r \quad \right\}\text{(Global Aggregation at Server)}$$

4. Performance Analysis

4.1. Hessian, Variance of ZO Gradient, and Low Effective Rank Assumption

We first examine the variance term of ZO gradient. It provides essential insights into Hessian-informed ZO methods.

$$\mathbb{E}\,\|u\|_\Sigma^2 := \mathbb{E}\, u^\mathsf{T} \Sigma u, \quad u \sim \mathcal{N}(0, I_d) \in \mathbb{R}^{d\times 1},$$

where Σ is semi-positive Hessian matrices[1]. The standard L-smoothness assumption implies $\|\Sigma\| \le L$. Thus, the preceding quantity can be upper-bounded as: $\mathbb{E}\,\|u\|_\Sigma^2 \le \|\Sigma\| \cdot \mathbb{E}\,\|u\|^2 \le Ld$.

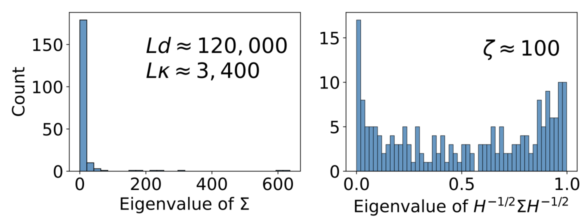

Figure 2: Distribution of the Eigenvalues.

Note that the upper bound derived above can be quite large if the dimension d is large. This dependence on dimensionality is a well-known factor leading to a typically slow convergence rate of ZO methods [24]. Fortunately, this bound only represents a worst-case scenario. Motivated by empirical observations that the Hessian of trained LLMs possesses relatively few eigenvalues significantly far from zero [26, 36, 38], [21] proposed a low-effective rank assumption. This spectral property, where most eigenvalues are concentrated near zero, is illustrated in Fig. 2 (left). To utilize this assumption, we need to treat the variance more carefully: $\mathbb{E}\,\|u\|_\Sigma^2 = \mathrm{Tr}(\Sigma \mathbb{E}\, uu^\mathsf{T}) = L\,\mathrm{Tr}(\Sigma/L) := L\kappa$, where $\kappa = \mathrm{Tr}(\Sigma/L)$ is called the effective rank of Hessian Σ. It is computationally prohibitive to find the exact value of κ, but several previous workers indicate $\kappa \ll d$ [18, 21]. Hence, we get a tighter variance estimation. Utilizing the Hessian approximate matrix, we can further improve this bound. Supposing we have a well approximation matrix H for the Hessian Σ, the weighted Gaussian vector z is sampled from the distribution $\mathcal{N}(0, H^{-1})$. Then, we have

$$\mathbb{E}\,\|z\|_\Sigma^2 = \mathbb{E}\,\mathrm{Tr}(H^{-1/2}\Sigma H^{-1/2} uu^\mathsf{T}) = \mathrm{Tr}(H^{-1/2}\Sigma H^{-1/2}) := \zeta, \tag{11}$$

where we call the quantity ζ as the low whitening rank of Hessian Σ.

1. For a non-convex function, Hessian may contain some negative eigenvalues. One possible choice of Σ can be the absolute eigenvalues of the Hessian.

If H is the perfect approximation of Σ, $\zeta = d$. This case is neither possible in practice nor ideal in LLMs. Recall that only a few eigenvalues of Σ are non-zero, $H \approx \text{Diag}(\Sigma + \epsilon \mathbb{1})$ is a more effective inverse value, which is similar to Wiener filtering in the denoising field [29]. Now we summarize the above into the following definition.

Definition. We call a diagonal matrix H as **a well-approximate matrix of Hessian** Σ if the whitening matrix $\Xi := H^{-1/2}\Sigma H^{-1/2}$ satisfies the following condition:

$$\text{Tr}(\Xi) = \text{Tr}(H^{-1/2}\Sigma H^{-1/2}) \leq \begin{cases} 2d & (L\text{-Smoothness}) \\ \zeta & (\text{Low Effective Rank}) \end{cases}, \tag{12}$$

where ζ is a quantity independent of the dimension d, and the factor 2 is just a safety factor to tolerate the imperfect inverse. The above assumptions and results are summarized in Table 1.

To show the effectiveness of this whitening process, we assume that there are 200 eigenvalues following the log-normal distribution, i.e., $\log(\Sigma) \sim \mathcal{N}(0, 3I)$ to simulate the distribution of Hessian eigenvalues. The simulation in Fig. 2 shows $\zeta \ll L\kappa \ll Ld$. This lays the theoretical foundation for the acceleration of our proposed HiSo.

Assumption	$\mathbb{E}\,\|u\|_\Sigma^2$	$\mathbb{E}\,\|z\|_\Sigma^2$
L-smooth	Ld	$2d$
Low Effective Rank	$L\kappa$	ζ

Table 1: ZO Grad. Variance Upper-Bound

4.2. Convergence Results

Assumption 1 (L-Lipschitz) $\|\nabla F(x) - \nabla F(y)\| \leq L\|x - y\|$.

Assumption 2 (Unbiased Stochastic Gradients with Bounded Variance) $\mathbb{E}\left[\nabla f_i(x;\xi)\right] = \nabla f_i(x)$ *and* $\mathbb{E}\,\|\nabla f_i(x;\xi) - \nabla f_i(x)\|^2 \leq \sigma_s^2$, $\forall x$, *where* ξ *represents a data sample.*

Assumption 3 (Bounded Heterogeneity) *The cost function satisfies* $\|\nabla f_i(x) - \nabla F(x)\| \leq \sigma_G, \forall x$.

Assumption 4 (Bounded Learned Hessian) *The learned Hessian has* $0 < \beta_\ell \leq \|H_r\| \leq \beta_u, \forall r$.

The last assumption is common in Hessian-informed [22, 46] or Adam-style algorithms [14, 28], where the requirement of bounded gradient implies this assumption directly. It is worth noting that, unlike the assumption on Hessian, the parameters β_ℓ and β_u can be easily controlled in the algorithm design by adding the clipping step [19]. This assumption also implies $\beta_u^{-1} \leq \|H_k^{-1}\| \leq \beta_\ell^{-1}$.

Theorem 1 *Under Assumptions 1, 2, 3, and 4, if* $\eta \leq \min\left(\frac{\beta_\ell}{mL}, \frac{1}{8\rho_k}, \frac{\beta_\ell}{4(\tau-1)}\sqrt{\frac{1}{L(d+2)}}\right)$, *the sequence of iterates generated by HiSo satisfies:*

$$\frac{1}{\tau R}\sum_{r=0}^{R-1}\sum_{k=0}^{\tau-1}\mathbb{E}\,\|\nabla F(\bar{x}_{r,k})\|_{H_r^{-1}}^2 \leq \frac{4(F(\bar{x}_1) - F^\star)}{\eta\tau R} + \underbrace{\frac{32\eta(\tau-1)^2 L\bar{\phi}}{\beta_\ell \tau m}(\sigma_G^2 + \sigma_s^2)}_{\text{extra client drift term}} + \frac{16\eta\bar{\rho}}{\beta_\ell m}(\sigma_G^2 + \sigma_s^2) + O(\eta\mu), \tag{13}$$

where $\bar{x}_{r,k} = \frac{1}{M}\sum_{i=1}^M x_{r,k}^{(i)}$, $\bar{\rho} = \frac{1}{\tau R}\sum_r\sum_k(\text{Tr}(H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2}) + 2\|H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2}\|)$, $\Sigma_{r,k}$ *is the Hessian at* $x_{r,k}$ *and* $\bar{\phi} = \frac{1}{R}\sum_r(\text{Tr}(H_r^{-1}) + 2\|H_r^{-1}\|)$. ■

Roughly, $\bar{\rho}$ can be understood as the sum of whitening Hessian eigenvalues and $\bar{\phi}$ as the sum of approximate Hessian eigenvalues. $\bar{\rho}$ includes two parts: 1) $\text{Tr}(H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2})$ is the quantity discussed previously, 2) $\|H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2}\|$ is much smaller than the first term when d is large. The properties of the terms in $\bar{\phi}$ are similar to $\bar{\rho}$.

Corollary 2 (Convergence Rate for HiSo) Suppose the learned global Hessian H_r satisfies the well-approximated condition (12). When $\tau = 1$ and $\eta = \sqrt{m\beta_\ell/\bar{\rho}R}$, HiSo's convergence rate is $\mathcal{O}(\sqrt{d/mR})$. Further, if the Hessian exhibits the low-effective rank property, the rate can be further improved to $\mathcal{O}(\sqrt{\zeta/mR})$ independent of the model dimension d and the Lipschitz condition L.

Acknowledgement

Research reported in this publication was supported by the National Institute Of General Medical Sciences of the National Institutes of Health under Award Number R16GM159671. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

[1] Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2): 251–276, 1998.

[2] Frederik Benzing. Gradient descent on neurons and its link to approximate second-order optimization. In *International Conference on Machine Learning*, pages 1817–1853. PMLR, 2022.

[3] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] Ran Chen, Yuzhe Li, and Tianyou Chai. Multi-objective derivative-free optimization based on hessian-aware gaussian smoothing method. In *2024 IEEE 18th International Conference on Control & Automation (ICCA)*, pages 210–215. IEEE, 2024.

[5] Yae Jee Cho, Luyang Liu, Zheng Xu, Aldi Fahrezi, and Gauri Joshi. Heterogeneous lora for federated fine-tuning of on-device foundation models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12903–12913, 2024.

[6] Wenzhi Fang, Ziyi Yu, Yuning Jiang, Yuanming Shi, Colin N Jones, and Yong Zhou. Communication-efficient stochastic zeroth-order optimization for federated learning. *IEEE Transactions on Signal Processing*, 70:5058–5073, 2022.

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

[8] Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[9] Robert Hönig, Yiren Zhao, and Robert Mullins. Dadaquant: Doubly-adaptive quantization for communication-efficient federated learning. In *International Conference on Machine Learning*, pages 8852–8866. PMLR, 2022.

[10] Ninghui Jia, Zhihao Qu, Baoliu Ye, Yanyan Wang, Shihong Hu, and Song Guo. A comprehensive survey on communication-efficient federated learning in mobile edge environments. *IEEE Communications Surveys & Tutorials*, 2025.

[11] Ruichen Jiang, Ali Kavis, Qiujiang Jin, Sujay Sanghavi, and Aryan Mokhtari. Adaptive and optimal second-order optimistic methods for minimax optimization. *Advances in Neural Information Processing Systems*, 37:94130–94162, 2024.

[12] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[13] Bumsu Kim, Daniel McKenzie, Hanqin Cai, and Wotao Yin. Curvature-aware derivative-free optimization. *Journal of Scientific Computing*, 103(2):43, 2025.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[15] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[16] Shiwei Li, Wenchao Xu, Haozhao Wang, Xing Tang, Yining Qi, Shijie Xu, weihongluo, Yuhua Li, xiuqiang He, and Ruixuan Li. FedBAT: Communication-efficient federated learning via learnable binarization. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=x2zxPwCkAZ.

[17] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020.

[18] Zhe Li, Bicheng Ying, Zidong Liu, Chaosheng Dong, and Haibo Yang. Achieving dimension-free communication in federated learning via zeroth-order optimization. In *The Thirteenth International Conference on Learning Representations*, 2025.

[19] Hong Liu, Zhiyuan Li, David Hall, Percy Liang, and Tengyu Ma. Sophia: A scalable stochastic second-order optimizer for language model pre-training. *arXiv preprint arXiv:2305.14342*, 2023.

[20] Shaocong Ma and Heng Huang. Revisiting zeroth-order optimization: Minimum-variance two-point estimators and directionally aligned perturbations. In *The Thirteenth International Conference on Learning Representations*, 2025.

[21] 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.

[22] Alessio Maritan, Subhrakanti Dey, and Luca Schenato. Fedzen: Quadratic convergence in zeroth-order federated learning via incremental hessian estimation. In *2024 European Control Conference (ECC)*, pages 2320–2327. IEEE, 2024.

[23] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*, 2023.

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

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

[26] Vardan Papyan. Traces of class/cross-class structure pervade deep learning spectra. *Journal of Machine Learning Research*, 21(252):1–64, 2020.

[27] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

[28] Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konecny, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021.

[29] Ali H Sayed. *Fundamentals of adaptive filtering*. John Wiley & Sons, 2003.

[30] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y 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*, pages 1631–1642, 2013.

[31] James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 1992.

[32] Xiaoxin Su, Yipeng Zhou, Laizhong Cui, John CS Lui, and Jiangchuan Liu. Fed-cvlc: Compressing federated learning communications with variable-length codes. In *IEEE INFOCOM 2024-IEEE Conference on Computer Communications*, pages 601–610. IEEE, 2024.

[33] Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving loRA in privacy-preserving federated learning. In *The Twelfth International Conference on Learning Representations*, 2024.

[34] Yujia Wang, Lu Lin, and Jinghui Chen. Communication-efficient adaptive federated learning. In *International Conference on Machine Learning*, pages 22802–22838. PMLR, 2022.

[35] Yebo Wu, Chunlin Tian, Jingguang Li, He Sun, Kahou Tam, Li Li, and Chengzhong Xu. A survey on federated fine-tuning of large language models. *arXiv preprint arXiv:2503.12016*, 2025.

[36] Yikai Wu, Xingyu Zhu, Chenwei Wu, Annie Wang, and Rong Ge. Dissecting hessian: Understanding common structure of hessian in neural networks. *arXiv preprint arXiv:2010.04261*, 2020.

[37] Haibo Yang, Jia Liu, and Elizabeth S Bentley. Cfedavg: achieving efficient communication and fast convergence in non-iid federated learning. In *2021 19th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)*, pages 1–8. IEEE, 2021.

[38] Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W Mahoney. Pyhessian: Neural networks through the lens of the hessian. In *2020 IEEE international conference on big data (Big data)*, pages 581–590. IEEE, 2020.

[39] Zhewei Yao, Amir Gholami, Sheng Shen, Mustafa Mustafa, Kurt Keutzer, and Michael Mahoney. Adahessian: An adaptive second order optimizer for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 10665–10673, 2021.

[40] Haishan Ye, Zhichao Huang, Cong Fang, Chris Junchi Li, and Tong Zhang. Hessian-aware zeroth-order optimization for black-box adversarial attack. *arXiv preprint arXiv:1812.11377*, 2018.

[41] Haishan Ye, Zhichao Huang, Cong Fang, Chris Junchi Li, and Tong Zhang. Hessian-aware zeroth-order optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[42] Bicheng Ying, Zhe Li, and Haibo Yang. Exact and linear convergence for federated learning under arbitrary client participation is attainable. *arXiv preprint arXiv:2503.20117*, 2025.

[43] Hossein Zakerinia, Shayan Talaei, Giorgi Nadiradze, and Dan Alistarh. Communication-efficient federated learning with data and client heterogeneity. In *International Conference on Artificial Intelligence and Statistics*, pages 3448–3456. PMLR, 2024.

[44] Hualin Zhang, Huan Xiong, and Bin Gu. Zeroth-order negative curvature finding: Escaping saddle points without gradients. *Advances in Neural Information Processing Systems*, 35: 38332–38344, 2022.

[45] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

[46] 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 *The Thirteenth International Conference on Learning Representations*, 2025.

Appendix A. Related Work

Adaptive Gradient Methods & Hessian-Informed Zeroth-Order Optimization. To accelerate first-order FL, adaptive FL algorithms (e.g., FedAdam, FedYogi, FedAdagrad [28]) have been introduced to address the slow convergence in heterogeneous environments. By adaptively adjusting learning rates or applying momentum techniques, these methods significantly outperform vanilla FedAvg in terms of convergence speed and final accuracy. Parallel to this line, recent advances in ZOO have shown its effectiveness in gradient-free learning, especially when gradients are unavailable or expensive to compute. To further enhance convergence speed and stability, several studies [4, 13, 40, 41, 44, 46, 46] proposed Hessian-informed ZOO methods that incorporate second-order information, such as diagonal Hessian approximations, as preconditioning to improve the quality of gradient estimation and reduce variance, which shows the acceleration in centralized settings.

Communication-Efficient Federated Learning & Scalar-Only Communication. Communication efficiency is a critical challenge in FL primarily due to the frequent transmission of high-dimensional model updates between clients and the server [10, 12]. Numerous methods have been proposed to reduce communication overhead in FL, including compression techniques used to reduce the size of transmitted data [9, 16, 32, 34, 37, 43], parameter-efficient methods, such as Low-Rank Adaptation (LoRA) [8, 33] to transmit only a low-rank trainable matrix representing model updates. Moreover, ZOO has also been introduced to the FL context. FedZO [6] integrates ZO-SGD into FL, but its communication heavily relies on the model dimension. DeComFL [18] pioneeringly exploited the intrinsic properties of ZO gradients—specifically, their decomposition into gradient scalars and perturbation vectors determined by random seeds—to achieve dimension-free communication overhead in LLM fine-tuning. Yet, it suffers from slower convergence due to the nature of ZO-SGD.

Appendix B. Conclusion

In this paper, we first present a new federated learning framework that supports scalar-only communication in both uplink and downlink, enabling the integration of a broader class of optimization algorithms beyond vanilla zeroth-order SGD. Building on this foundation, we propose HiSo, a Hessian-informed federated fine-tuning algorithm that leverages diagonal Hessian approximations to accelerate convergence while preserving scalar-only communication efficiency. From a theoretical perspective, we introduce a novel variance characterization for Hessian-informed zeroth-order gradients under a low-effective-rank assumption. This allows us to establish a convergence rate that is independent of both model dimensionality and function smoothness in non-convex settings - a result not previously achieved by any zeroth-order method in federated learning. Our analysis further generalizes the DeComFL framework and extends its theoretical guarantees to support multiple local updates, a critical component in practical FL deployments. Empirically, HiSo consistently outperforms existing baselines, delivering higher test accuracy, up to about $5\times$ faster convergence, and substantially lower communication overhead. These results demonstrate the practical viability and theoretical soundness of unifying curvature-informed optimization with scalar-only communication in federated fine-tuning.

Appendix C. Limitations

The proposed method is currently limited by its treatment of the loss function f_i as a generic one, without considering model-specific module structures. This is in contrast to modern parameter-efficient fine-tuning (PEFT) methods that often exploit properties like low-rank decomposition (e.g., $W = AB^{\mathsf{T}}$, where $A \in \mathbb{R}^{k_1 \times r}$ and $B \in \mathbb{R}^{k_2 \times r}$ and $r \ll k_1, k_2$). It is important to note that this explicit low-rank decomposition is distinct from the 'low effective rank' of the Hessian discussed in this paper. Consequently, there is potential to further refine our approach by designing Hessian information specifically tailored for PEFT methods such as LoRA or GaLore.

Appendix D. Generalized Scalar-Only Communication in Federated Learning

Algorithm 1 Generalized Scalar-Only Communication in Federated Learning

Initialize: learning rate η, local update steps τ, communication rounds R.
Allocate: memory for recording the necessary historical states and client's participation information.
for $r = 0, \cdots, R-1$ **do**
  Server uniformly samples a client set C_r and distributes the shared random seeds $\{s_r\}$.
  for each client $i \in C_r$ **in parallel do**
    Receive the necessary scalar representations of $\{\Delta x_{r'}\}$ from server.
    Reconstruct the $\{\Delta x_{r'}\}$ from the scalars and update state.
    $x_{r,0}^{(i)} = x_{r_l,\tau}^{(i)} - \eta \sum_{r'=r_l}^{r-1} \Delta x_{r'}$ ▷ Equivalent to pull model
    for $k = 0, \cdots, \tau - 1$ **do**
      Find $\Delta x_{r,k}^{(i)}$ that 1) is ascent direction; 2) can be represented by scalars + state;
      $x_{r,k+1}^{(i)} = x_{r,k}^{(i)} - \eta \Delta x_{r,k}^{(i)}$. ▷ Client local update
    end for
    $x_{r,\tau}^{(i)} \Leftarrow x_{r,0}^{(i)}$ reset the model and other necessary states.
    Send the necessary scalar representations of $\{\Delta x_{r,k}^{(i)}\}$ to server. ▷ Equivalent to push model
  end for
  Aggregate the scalar representations of $\{\Delta x_{r,k}^{(i)}\}$ into the ones for the global Δx_r.
end for

Appendix E. Detailed HiSo Algorithm Table

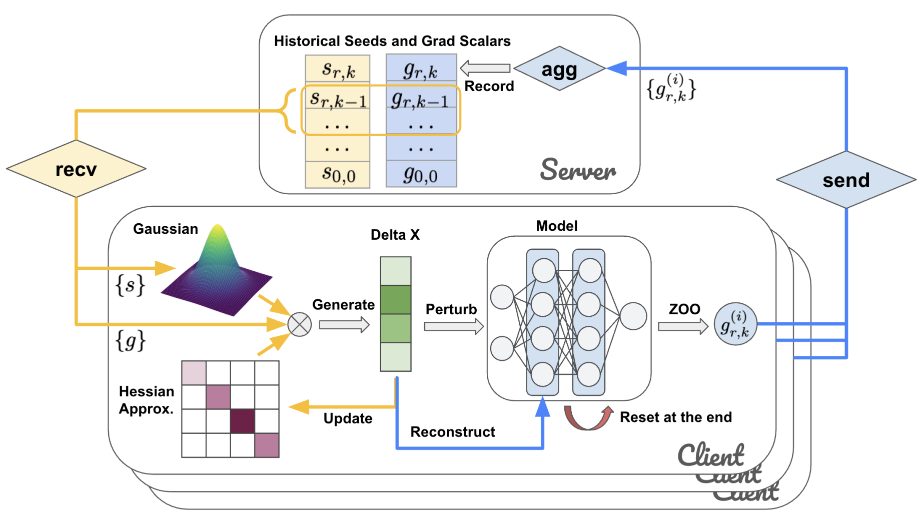

Figure 3: Illustration of HiSo

Although the algorithm listed in the main context is quite complicated, it is simple if we ignore the dimension-free communication property. Mathematically, HiSo is equivalent to the following standard FedAvg style update

$$
\begin{aligned}
&x_{r,0}^{(i)} = x_r && \text{(Receive Model)}\\
&\text{for } k = 0, 1, \cdots, \tau - 1:\\
&\qquad g_{r,k}^{(i)} = \frac{1}{\mu}\big(f_i(x_{r,k}^{(i)} + \mu H_r^{-1/2} u_{r,k}) - f_i(x_{r,k}^{(i)})\big)\\
&\qquad x_{r,k+1}^{(i)} = x_{r,k}^{(i)} - \eta g_{r,k}^{(i)} H_r^{-1/2} u_{r,k} && \text{(Local Update)}\\
&x_{r+1} = \frac{1}{|C_r|}\sum_{i\in C_r} x_{r,\tau}^{(i)} && \text{(Aggregate Model)}\\
&H_{r+1} = (1-\nu)H_r + \nu \mathrm{Diag}([x_{r+1} - x_r]^2 + \epsilon I)
\end{aligned}
$$

With that as reference, we present the full algorithm table for HiSo.

Algorithm 2 Concrete Scalar Representations Communication with States for Federated Learning

Initialize: learning rate η, local update steps K, communication rounds R, clients' participation round $r'_i = 0$.
Allocate: memory for recording the necessary historical states, including historical gradient scalars $\{g\}$, corresponding random seeds $\{s\}$ and clients' last participation round $\{r'\}$.

for $r = 0, 1, \cdots, R-1$ **do**
  Server uniformly samples a client set C_r with cardinality m.
  Server randomly samples a random seed set $\{s_{r,k}\}_{k=0}^{\tau-1}$ and broadcasts it to all sampled clients.
  for each client $i \in C_r$ **in parallel do**
    $\{\{\Delta x_t^{(i)}\}_{k=0}^{\tau-1}\}_{t=r'}^{r-1} = \text{Rebuild}(\{\{s_{t,k}^{(i)}\}_{k=0}^{\tau-1}\}_{t=r'_i}^{r-1}, \{\{g_{t,k}^{(i)}\}_{k=0}^{\tau-1}\}_{t=r'_i}^{r-1})$
    $x_{r,0}^{(i)} = x_{r',0}^{(i)} - \eta \sum_{t=r'}^{r-1}\sum_{k=0}^{\tau-1} \Delta x_{t,k}^{(i)}$
    $\{g_{r,k}^{(i)}\}_{k=0}^{\tau-1} = \text{LocalUpdate}(\{s_{r,k}\}_{k=0}^{\tau-1})$
    Send $\{g_{r,k}^{(i)}\}_{k=0}^{\tau-1}$ back to the server.
  end for
  $\{g_{r,k}\}_{k=0}^{\tau-1} = \left\{\frac{1}{|\mathcal{C}_r|}\sum_{i\in\mathcal{C}_r} g_{r,k}^{(i)}\right\}_{k=0}^{\tau-1}$ ▶ Global gradient scalar aggregation
  $\{\Delta x_{r,k}\}_{k=0}^{\tau-1} = \left\{g_{r,k} H_r^{-1/2} u_{r,k}\right\}_{k=0}^{\tau-1}$ ▶ Global Δ aggregation at server
  Store $\{g_{r,k}\}_{k=0}^{\tau-1}$ and $\{s_{r,k}\}_{k=0}^{\tau-1}$ and update the client's last participation round $r'_i = r$.
  $x_{r+1} = x_r - \eta\sum_{k=0}^{\tau-1}\Delta x_{r,k}$ ▶ (Optional) Global model update
end for

Appendix F. Experiment Detail and Results

F.1. Baseline Selection

We select a broad range of classic baselines to cover both first-order and zeroth-order optimization methods commonly used in FL.

Algorithm 2a Receiving Step for Hessian-Informed ZO Gradient for i-th Client at r-th Round

Function Rebuild($\{\{s_{t,k}\}_{k=0}^{\tau-1}\}_{r=r'}^{r-1}$, $\{\{g_{t,k}\}_{k=0}^{\tau-1}\}_{r=r'}^{r-1}$): ▶ r' is last participation round
  for $t = r', \cdots, r-1$ **do**
    for $k = 0, \cdots, \tau - 1$ **do**
      Utilize the random seed $s_{t,k}$ to produce $u_{t,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
      $\Delta x_{t,k} = g_{t,k} H_t^{-1/2} u_{t,k}$
      $H_{t+1} = (1-\nu) H_t + \nu \mathrm{Diag}([\Delta x_{t,\tau}]^2 + \epsilon I)$
    end for
  end for
  return $\{\{\Delta x_{t,k}\}_{k=0}^{\tau-1}\}_{t=r'}^{r-1}$ ▶ For model reconstruction

Algorithm 2b Sending Step for Hessian-Informed ZO Gradient for i-th Client at r-th Round

Function LocalUpdate($\{s_{r,k}\}_{k=0}^{\tau-1}$):
  for $k = 0, \cdots, \tau - 1$ **do**
    Utilize the random seed $s_{r,k}$ to produce $u_{r,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $g_{r,k}^{(i)} = \frac{1}{\mu}\left[f_i(x_{r,k}^{(i)} + \mu H_r^{-1/2} u_{r,k}) - f_i(x_{i,r}^{(i)})\right]$ ▶ Compute ZO gradient scalar
    $\Delta x_{r,k}^{(i)} = g_{r,k}^{(i)} H_r^{-1/2} u_{r,k}$ ▶ Can be replaced by other representation methods of $\Delta x_{r,k}^{(i)}$
    $x_{r,k+1}^{(i)} = x_{r,k}^{(i)} - \eta \Delta x_{r,k}^{(i)}$ ▶ Update local model
  end for
  $x_{r,\tau}^{(i)} \Leftarrow x_{r,0}^{(i)}$ ▶ Reset the local model and update other necessary states
  return $\{g_{r,k}^{(i)}\}_{k=0}^{\tau-1}$

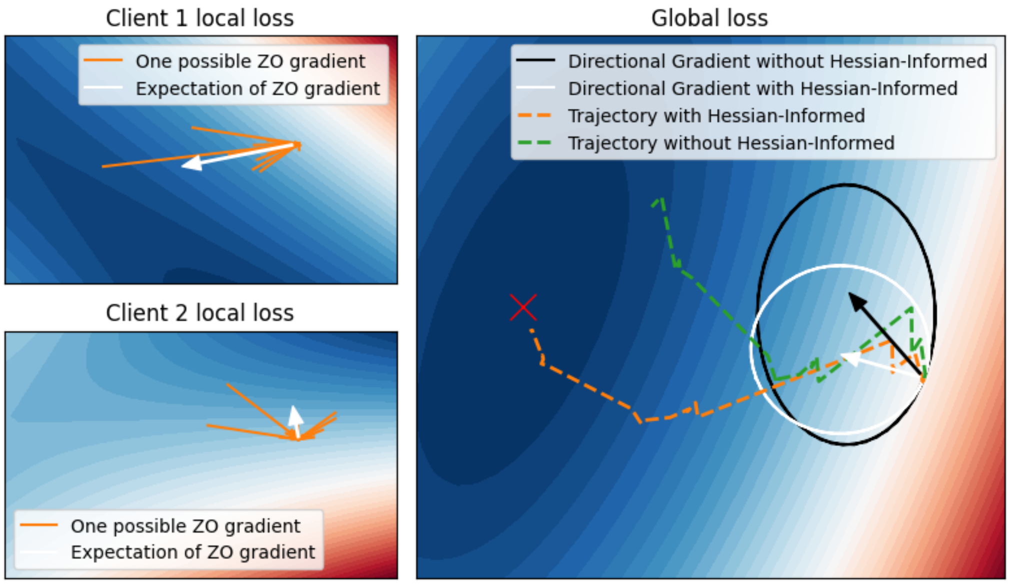

Figure 4: An illustration of Hessian-informed versus regular ZO gradient direction under the FL setting.

First-order methods: FedAvg is the most classic first-order FL algorithm. FedAdam, FedYogi and FedAdagrad are representatives of adaptive gradient-based methods designed to accelerate convergence. All of them are standard baselines widely used in federated optimization literature and practical systems.

Zeroth-order methods: FedZO is the first FL method to incorporate ZO-SGD into client local updates. DeComFL is the first method to achieve dimension-free communication in FL, which also uses ZO-SGD to perform client local updates.

F.2. Experiment Results

The Global Diagonal Hessian Approximation H.

We begin by training a CNN model on MNIST [15] to visualize the learned diagonal Hessian approximation H. To facilitate this, we established a 64-client FL system where data was partitioned non-IID using a Dirichlet distribution ($\alpha = 1$), assigning a unique subset to each client. Each communication round, 8 clients are randomly sampled. Evaluating the Hessian smoothing parameter ν revealed negligible impact on convergence and final accuracy (Fig. 5, left), showing the algorithm's robustness to this hyperparameter. Moreover, Fig. 5 (right) plots each entry of the learned diagonal Hessian values at the end of training. While individual entries may appear stochastic, their overall distribution clearly exhibits a long-tail phenomenon. This observation aligns with the low effective rank assumption discussed in Sec. 4.1. Although computing the exact Hessian is computationally prohibitive, the rapid convergence combined with this observed distribution suggests our strategy effectively approximates relevant Hessian structure.

HiSo is Faster Than DeComFL in Small Model Training Tasks.

In Fig. 5, we evaluate HiSo against the DeComFL baseline, another dimension-free communication FL algorithm. Crucially, the communication cost per round was held identical for both methods to ensure a fair comparison of algorithmic efficiency. Fig. 5 illustrates that, under the same

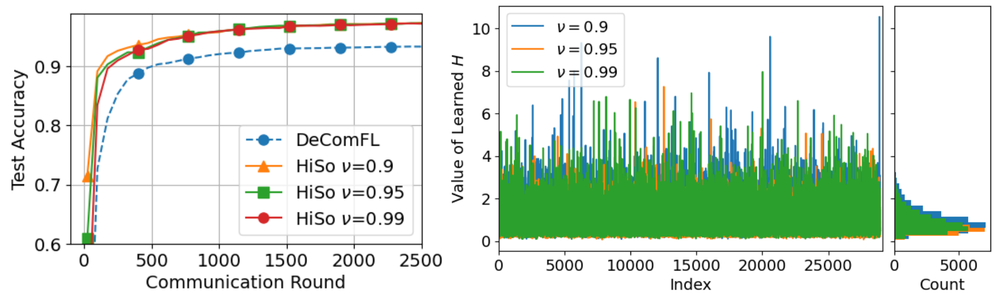

Figure 5: Ablation study of smoothing parameter ν and the distribution of the learned global Hessian H.

communication constraints, our HiSo achieves significantly faster convergence and reaches a superior final performance level compared to DeComFL. For this comparison, both algorithms were tuned using their optimal learning rates. More experiment results are provided in Appendix F.

HiSo can Accelerate Training with Less Communication Cost in LLM Fine-Tuning.

Our FL system consists of 6 clients in total, and 2 clients are uniformly sampled in each round. We execute sentiment classification on SST-2 [30], question matching on QQP, and question answering on SQuAD [27]. As shown in Table 2, HiSo needs less communication rounds required to reach DeComFL's best test accuracy, resulting in lower communication costs: HiSo achieves up to 2× speedup and reduces about 50%-80% communication cost on OPT-350M, a 1.4-2× speedup, saving 29%-50% in communication costs on OPT-1.3B. These results show that HiSo accelerates convergence and reduces communication cost, making it more practical for large-scale federated fine-tuning.

Table 2: HiSo's Acceleration. For DeComFL, we report the total number of communication rounds required to fully converge. For HiSo, we report the number of rounds needed to match DeComFL's best test accuracy, along with the corresponding communication cost. Five perturbations are used.

Model	Method	SST-2			QQP			SQuAD		
		Round	Speedup	Comm. Cost	Round	Speedup	Comm. Cost	Round	Speedup	Comm. Cost
OPT-350M	DeComFL	550	1×	21.56 KB	775	1×	30.35 KB	1350	1×	52.73 KB
	HiSo	275	2×	10.78 KB	425	1.8×	16.64 KB	250	5.4×	9.77 KB
OPT-1.3B	DeComFL	1500	1×	58.59 KB	1125	1×	43.95 KB	350	1×	13.67 KB
	HiSo	1075	1.4×	41.85 KB	750	1.5×	29.30 KB	175	2×	6.84 KB

Comprehensive Performance Comparison on LLM Fine-Tuning Tasks.

Table 3 evaluates a range of federated optimization methods across three LLM scales (e.g., OPT-125M, OPT-350M and OPT-1.3B) on SST-2, QQP, and SQuAD datasets. First-order methods (e.g., FedAvg, FedAdam, FedYogi and FedAdagrad) consistently achieve high test accuracy, but at the cost of extremely large communication volumes, often exceeding hundreds of gigabytes to several terabytes per client. This level of communication overhead is quite challenging and even impractical for real-world federated fine-tuning, especially on edge devices or mobile platforms. For ZO baselines, FedZO's communication cost is still quite high since it is required to transmit d-dimensional update. DeComFL addresses this high communication cost by enabling the scalar-only communication pattern, achieving several orders of magnitude lower communication cost. However, these ZO approaches suffer from limited optimization efficiency and often underperform in accuracy compared with first-order baselines, particularly on large-scale models and complex tasks.

Our proposed method, HiSo, is the first to break this trade-off. It maintains the scalar-only or dimension-free communication paradigm, yet consistently outperforms ZO baselines in test accuracy. For example, on SST-2 with the OPT-1.3B model, HiSo achieves 90.34% test accuracy - slightly lower than FedAdam (92.86%) but with a $10^4\times$ reduction in communication (7.81 KB vs. 0.79 TB). On QQP, HiSo also outperforms all ZO methods across all model sizes, achieving both higher accuracy and dramatically lower bandwidth usage. A similar trend holds on the SQuAD dataset, where HiSo consistently surpasses ZO baselines in F1 score while maintaining kilobyte-level communication. Notably, on OPT-350M and OPT-1.3B, HiSo not only outperforms ZO baselines in test accuracy but also achieves over $100\times$ less communication cost compared to first-order baselines. Moreover, compared with the most related baseline - DeComFL, HiSo achieves higher test accuracy, faster convergence speed, and less communication overhead.

The key to this performance lies in HiSo's Hessian-informed preconditioning and the use of multiple perturbations per round, which together yield more accurate ZO gradient estimates under low-rank curvature. These results demonstrate that, contrary to conventional expectations, it is possible to achieve second-order convergence behavior with near-zero communication overhead—a major step toward practical and scalable federated fine-tuning of LLMs.

Table 3: Performance for LLM Fine-Tuning. 1) We report the total communication cost of the single client during the entire training process until convergence. For SST-2 and QQP datasets, we report test accuracy. For SQuAD dataset, we report the F1 score. 2) The number of perturbations is 5.

Model	Method	SST-2	QQP	SQuAD
OPT-125M	FedAvg	87.63% ± 0.16 (0.15 TB)	61.21% ± 0.37 (0.08 TB)	37.27 ± 0.11 (0.05 TB)
	FedAdam	88.29% ± 0.47 (0.30 TB)	63.18% ± 0.31 (0.06 TB)	37.98 ± 0.20 (0.03 TB)
	FedYogi	88.06% ± 0.33 (0.29 TB)	62.88% ± 0.21 (0.05 TB)	37.66 ± 0.18 (0.04 TB)
	FedAdagrad	85.04% ± 0.51 (0.18 TB)	61.77% ± 0.22 (0.06 TB)	37.29 ± 0.27 (0.04 TB)
	FedZO	84.19% ± 0.22 (0.63 TB)	60.06% ± 0.21 (1.94 TB)	34.03 ± 0.26 (0.14 TB)
	DeComFL	85.21% ± 0.27 (22.92 KB)	60.11% ± 0.19 (32.17 KB)	34.12 ± 0.22 (17.42 KB)
	HiSo (Ours)	85.55% ± 0.21 (14.69 KB)	60.72% ± 0.25 (21.23 KB)	35.26 ± 0.14 (7.12 KB)
OPT-350M	FedAvg	89.79% ± 0.05 (0.58 TB)	63.32% ± 0.13 (0.31 TB)	43.38 ± 0.13 (0.12 TB)
	FedAdam	89.92% ± 0.20 (0.21 TB)	63.28% ± 0.19 (0.28 TB)	45.92 ± 0.14 (0.08 TB)
	FedYogi	89.68% ± 0.29 (0.25 TB)	63.21% ± 0.16 (0.28 TB)	45.01 ± 0.25 (0.09 TB)
	FedAdagrad	87.42% ± 0.09 (0.23 TB)	62.55% ± 0.14 (0.29 TB)	44.49 ± 0.11 (0.09 TB)
	FedZO	86.55% ± 0.23 (0.68 TB)	61.22% ± 0.30 (0.66 TB)	38.14 ± 0.24 (0.38 TB)
	DeComFL	86.72% ± 0.28 (21.56 KB)	60.58% ± 0.16 (30.35 KB)	38.20 ± 0.15 (52.73 KB)
	HiSo (Ours)	87.50% ± 0.22 (17.33 KB)	62.49% ± 0.17 (18.63 KB)	39.13 ± 0.11 (20.51 KB)
OPT-1.3B	FedAvg	90.48% ± 0.35 (0.63 TB)	65.77% ± 0.20 (0.32 TB)	60.39 ± 0.27 (0.41 TB)
	FedAdam	92.86% ± 0.43 (0.79 TB)	64.59% ± 0.53 (1.10 TB)	61.56 ± 0.14 (0.27 TB)
	FedYogi	92.39% ± 0.58 (0.83 TB)	64.44% ± 0.22 (1.12 TB)	61.44 ± 0.19 (0.29 TB)
	FedAdagrad	90.92% ± 0.74 (0.88 TB)	64.05% ± 0.13 (1.08 TB)	60.72 ± 0.23 (0.33 TB)
	FedZO	90.01% ± 0.29 (4.73 TB)	62.91% ± 0.14 (3.53 TB)	57.26 ± 0.17 (1.10 TB)
	DeComFL	90.22% ± 0.10 (58.59 KB)	63.25% ± 0.11 (43.95 KB)	57.14 ± 0.14 (13.67 KB)
	HiSo (Ours)	90.34% ± 0.12 (49.18 KB)	64.20% ± 0.13 (96.67 KB)	57.58 ± 0.07 (7.81 KB)

Appendix G. Main Proof

G.1. Notations

The following proof utilizes matrix and vector notations. A bold symbol, such as $\boldsymbol{x}_k$, generally represents a vector encompassing multiple clients, whereas a normal symbol, such as $x_k^{(i)}$, denotes the value for an individual client. To further lighten the notation for multiple clients and the local cost function, we adopt the following usage:

$$\boldsymbol{x}_k = \begin{bmatrix} x_k^{(1)} & x_k^{(2)} & \cdots & x_k^{(M)} \end{bmatrix} \in \mathbb{R}^{d\times M}, \tag{14}$$

$$\boldsymbol{f}(\boldsymbol{x}_k) = \begin{bmatrix} f_1(x_k^{(1)};\xi_k^{(1)}) & f_2(x_k^{(2)};\xi_k^{(1)}) & \cdots & f_M(x_k^{(M)};\xi_k^{(1)}) \end{bmatrix} \in \mathbb{R}^{1\times M}, \tag{15}$$

$$\nabla\boldsymbol{f}(\boldsymbol{x}_k) = \begin{bmatrix} \nabla f_1(x_k^{(1)};\xi_k^{(1)}) & \nabla f_2(x_k^{(2)};\xi_k^{(1)}) & \cdots & \nabla f_M(x_k^{(M)};\xi_k^{(1)}) \end{bmatrix} \in \mathbb{R}^{d\times M}. \tag{16}$$

where $\nabla f_1(x_k^{(1)};\xi_k^{(1)})$ represent the stochastic gradient evaluated on local cost function f_1 at the point $x_k^{(1)}$. Notice the function value f_i or the gradient ∇f_i applied on the different iterates $\boldsymbol{x}_k^{(i)}$ in above notations. Various vector and matrix norms are used in the proof. For any semi-positive definite matrix Σ, we adopt the following convention in Table 4.

Table 4: Norm Notations in This Paper

Notation	Definition	Comment
$\|x\|_\Sigma^2$	$x^\mathsf{T}\Sigma x$	Mahalanobis (weighted) vector norm, where $x\in\mathbb{R}^d$.
$\|A\|_\Sigma^2$	$\mathrm{Tr}(A^\mathsf{T}\Sigma A)$	Mahalanobis (weighted) matrix norm $A\in\mathbb{R}^{d\times d}$
$\|A\|_2, \|A\|$	$\sigma_{\max}(A)$	Spectrum norm, i.e., largest singular value of A
$\|\boldsymbol{x}\|_F^2$	$\mathrm{Tr}(\boldsymbol{x}^\mathsf{T}\boldsymbol{x})$	Frobenius norm (note $\boldsymbol{x}$ is matrix here)

Remark: While the Frobenius norm can be viewed as a special case of the weighted matrix norm, confusion is unlikely in this paper as we only apply the Frobenius norm to the stacked vector $\boldsymbol{x}$.

Other commonly used constants and symbols are summarized in the following table.

The all-one vector $\mathbb{1} = [1, 1, \cdots, 1]^\mathsf{T} \in \mathbb{R}^{M\times 1}$ and the uniform vector $\mathbb{1}_u = \mathbb{1}/M \in \mathbb{R}^{M\times 1}$ are two common notations we adopted in the rest of the proof. With these symbols, we have the following identity

$$\nabla\boldsymbol{f}(x\mathbb{1}^\mathsf{T})\mathbb{1}_u = \nabla F(x) \in \mathbb{R}^{d\times 1} \tag{17}$$

G.2. Algorithm Reformulation and Main Recursion

To make a concise proof, we first re-write the algorithm into the vector-matrix form as introduced in the previous section. First, to make the convergence proof straightforward, we translate the two-level for-loop structure (outer round loop and inner local update loop) into a single recursion structure. The k-th local update in r-th communication round is equivalent to the $r\tau + k$ iterations. Then, inspired by the work [17, 42], first we notice the Federated Learning algorithm is equivalent if we virtually send the server's model to all clients but keep the aggregation step the same, i.e., only aggregate

Table 5: Notations in This Paper

Notation	Meaning
i	Index of clients
k	Index of iterations
r	Index of communication round and $r = \lfloor k/\tau \rfloor \tau$
τ	The number of local update steps
C_r	Indices set of clients sampled at r-th round
d	Model parameter dimension
m, M	Number of sampled and total clients
f_i, F	Local and global loss function
u, z	A random vector drawing from the standard and weighted Gaussian distributions

the clients' values in C_r. Under this form, we can equivalently reformulate the algorithm into this recursion

$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k - \eta H_k^{-1/2} u_k \frac{\boldsymbol{f}(\boldsymbol{x}_k + \mu H_k^{-1/2} u_k \mathbb{1}^\mathsf{T}) - \boldsymbol{f}(\boldsymbol{x}_k)}{\mu}, \tag{18}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_{k+1} W_k. \tag{19}$$

where $\boldsymbol{x}_k, \boldsymbol{y}_k \in \mathbb{R}^{d\times M}$ is the stacked vectors and W_k represents the communication matrix. Note the single subscript k is for the iteration, which is not the same k in the double subscripts for local update step. The element of $W_k[i,j]$ represents the effective weight that client i to client j at iteration k. If the iteration $k \neq r\tau$, $W_k = I$ – local update step. If $k = r\tau$, W_k becomes some average matrix representing the model average step. More concretely, it is a **column-stochastic** matrix, each column having the same weights and the non-zero elements in each column are the sampled clients in round r. For instance, suppose client $\{0,1,3\}$ sampled in the four clients case, the corresponding W_k are

$$W_k = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \tag{20}$$

Back to the update rule (18) – (19), the following proof is for the general update rule of H_k. Hence, we just need to focus on the property of H_k instead of combining the update rule and revisit it later. We further denote $z_k = H_k^{-1/2} u_k$, $z_k \sim \mathcal{N}(0, H_k^{-1})$ to simplify the update rule:

$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k - \frac{\eta}{\mu} z_k \Big(\boldsymbol{f}(\boldsymbol{x}_k + \mu z_k \mathbb{1}^\mathsf{T}) - \boldsymbol{f}(x_k) \Big), \tag{21}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_{k+1} W_k \tag{22}$$

Because of the shared seeds and Hessians, $\boldsymbol{z}_k$ is a variable that has no client index subscripts. Using directional gradient approximation

$$f(x + \mu z) = f(x) + \mu z^\mathsf{T} \nabla f(x) + \frac{\mu^2}{2} z^\mathsf{T} \left(\int_0^1 \nabla^2 f(x + tz) dt \right) z, \tag{23}$$

the update rule can be concisely written as

$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k - \eta z_k z_k^{\mathsf{T}} \nabla \boldsymbol{f}(\boldsymbol{x}_k) + O(\mu\eta), \tag{24}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_{k+1} W_k, \tag{25}$$

To manage notational complexity and the handling of intricate coefficients, we adopt the $O(\mu\eta)$ notation. Since this paper concentrates on addressing client sampling and local updates in federated learning, the analysis of the zeroth-order approximation error is intentionally simplified. This approach facilitates a clearer understanding of the distinct error sources in the federated setting, without sacrificing proof rigor.

We define the (virtual) centralized iterates $\bar{x}_k := \boldsymbol{x}_k \mathbb{1}_u$ and $\bar{y}_k := \boldsymbol{y}_k \mathbb{1}_u$. The recursion of centralized iterates $\bar{x}_k := \boldsymbol{x}_k \mathbb{1}_u$ is

$$\bar{x}_{k+1} = \boldsymbol{y}_{k+1} W_k \mathbb{1}_u \tag{26}$$

$$= \Big(\boldsymbol{x}_k - \eta z_k z_k^{\mathsf{T}} \nabla \boldsymbol{f}(\boldsymbol{x}_k)\Big) w_k + O(\mu\eta) \tag{27}$$

where we define $w_k := W_k \mathbb{1}_u$. It is straightforward to see that if $k \neq r\tau$, $w_k = \mathbb{1}_u$; if $k = r\tau$, w_k is the random selection vector with each entry having m/M probability to be $1/m$ and 0 otherwise. Hence, we have the following two cases to handle with

$$\boxed{\bar{x}_{k+1} = \begin{cases} \bar{x}_k - \eta z_k z_k^{\mathsf{T}} \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) + O(\mu\eta) & k \neq r\tau, \\ \hat{x}_k - \eta z_k z_k^{\mathsf{T}} \widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) + O(\mu\eta) & k = r\tau. \end{cases}} \tag{28}$$

where we denote

$$\hat{x}_k = \boldsymbol{x}_k w_k, \tag{29}$$

$$\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) = \nabla \boldsymbol{f}(\boldsymbol{x}_k) \mathbb{1}_u = \frac{1}{M} \sum_{i=1}^{M} \nabla f_i(x_k^{(i)}) \in \mathbb{R}^{d \times 1}, \tag{30}$$

$$\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) = \nabla \boldsymbol{f}(\boldsymbol{x}_k) w_k = \frac{1}{m} \sum_{i \in C_r} \nabla f_i(x_k^{(i)}) \in \mathbb{R}^{d \times 1}. \tag{31}$$

Above two centralized recursions will be the main reference the following proof.

G.3. Key Lemmas

G.3.1. LEMMAS ABOUT GAUSSIAN VARIABLES

The rest proof is built on top of the following two fundamental lemmas about the Gaussian distribution.

Lemma 3 (Fourth-Order Moment of Gaussian Vector) *Suppose that the random vector $z \sim \mathcal{N}(0, \Lambda)$ where Λ is a diagonal matrix. For any symmetric matrix W, we have*

$$\mathbb{E}\, z z^{\mathsf{T}} W z z^{\mathsf{T}} = \mathrm{Tr}(W\Lambda) \cdot \Lambda + 2\Lambda W \Lambda. \tag{32}$$

If $u \sim \mathcal{N}(0, I)$, i.e., drawing from a standard Gaussian distribution, we have

$$\mathbb{E}\, u u^{\mathsf{T}} W u u^{\mathsf{T}} = \mathrm{Tr}(W) \cdot I + 2W. \tag{33}$$

Proof Let the matrix $\Psi = z z^{\mathsf{T}} W z z^{\mathsf{T}}$. For each element $i \neq j$,

$$\Psi[i, j] = \mathbb{E}\, z_i z_j \Big(\sum_{i', j'} z_{i'} z_{j'} W[i', j']\Big) = 2\mathbb{E}\, z_i^2 z_j^2 W[i, j] = 2\Lambda_i \Lambda_j W[i, j], \tag{34}$$

where the second equality holds because the zero-mean property of z and z_i is independent of each other. For the diagonal elements,

$$\begin{aligned}\Psi[i,i] =&\mathbb{E}\, z_i^2(\sum_{i',j;} z_{i'}z_{j'}W[i',j']) = \sum_{i'}\mathbb{E}\, z_i^2 z_{i'}^2 W[i',i'] \\ =&\sum_{i'\neq i}\mathbb{E}\, z_i^2\mathbb{E}\, z_{i'}^2 W[i',i'] + \mathbb{E}\, z_i^4 W[i,i] \\ =&\Lambda_i\sum_{i'}\Lambda_{i'}W[i',i'] + 2W[i,i]\Lambda_i^2, \end{aligned} \tag{35}$$

where we utilize the fact that $\mathbb{E}\, z_i^4 = 3\Lambda_i^2$. Lastly, combining the above two results into a concise matrix notation, we establish

$$\Psi = \mathrm{Tr}(W\Lambda)\cdot\Lambda + 2\Lambda W\Lambda \tag{36}$$

For the standard Gaussian distribution case, we just need to substitute $\Lambda = I$ into equation 32. ■

Lemma 4 (Gaussian Smoothed Function) *We define a smooth approximation of objective function f as $f^\mu(\cdot)$ that can be formulated as*

$$f^\mu(x) := \frac{1}{(2\pi)^{\frac{d}{2}}}\int f(x+\mu u)e^{-\frac{1}{2}\|u\|^2}d\boldsymbol{z} = \mathbb{E}\,[f(x+\mu)], \tag{37}$$

where $\mu > 0$ is the smoothing parameter, and $\boldsymbol{z}$ is one n-dimensional standard Gaussian random vector. Then, we have

$$\mathbb{E}\,\frac{f(x+\mu u)-f(x)}{\mu}u = \nabla f^\mu(x), \quad \text{where } u\sim\mathcal{N}(0,I) \tag{38}$$

Above equality implies the ZO gradient is an unbiased estimate of the gradient of the smoothed function f^μ.

Proof See the proof in [7, 25]. ■

G.3.2. VARIANCE LEMMA FOR SAMPLING NOISE

Before we present the main proof, we first bound the variance of $\widehat{\nabla\boldsymbol{f}}(\boldsymbol{x}_k)$.

Lemma 5 *Suppose f_i is L-smooth and the local cost functions satisfy the data heterogeneity assumption σ_G^2. For any semi-positive definite matrix Σ, the variance of the sampled gradient $\widehat{\nabla\boldsymbol{f}}(\boldsymbol{x}_k)$ satisfies:*

$$\mathbb{E}\,\|\widehat{\nabla\boldsymbol{f}}(\boldsymbol{x}_k)\|_\Sigma^2 \leq 2\|\nabla F(\bar{x}_k)\|_\Sigma^2 + \frac{2}{m}\|\Sigma\|(\sigma_G^2+\sigma_s^2) + \frac{2L^2}{M}\|\Sigma\|\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2, \tag{39}$$

where m is the number of sampled clients per round and M is the total number of clients.

Proof For any semi-positive matrix Σ, we have

$$\mathbb{E}\,\|\widehat{\nabla\boldsymbol{f}}(\boldsymbol{x}_k)\|_\Sigma^2 \leq 2\mathbb{E}\,\|\widehat{\nabla\boldsymbol{f}}(\bar{x}_k\mathbb{1}^\mathsf{T})\|_\Sigma^2 + 2\mathbb{E}\,\|\widehat{\nabla\boldsymbol{f}}(\boldsymbol{x}_k) - \widehat{\nabla\boldsymbol{f}}(\bar{x}_k\mathbb{1}^\mathsf{T})\|_\Sigma^2 \tag{40}$$

where the inequality utilizes Jensen's inequality.

Next, noticing that the variance identity for any weighted distance $\|\cdot\|_\Sigma$ satisfies

$$\begin{aligned}\mathbb{E}\,\|\bar{x}_k - \mathbb{E}\,\bar{x}_k\|_\Sigma^2 =&\mathbb{E}\,\|\bar{x}_k\|_\Sigma^2 - \mathbb{E}\,(\bar{x}_k^\mathsf{T}\Sigma\mathbb{E}\,\bar{x}_k) - \mathbb{E}\,(\mathbb{E}\,\bar{x}_k^\mathsf{T})\Sigma\bar{x}_k + \|\mathbb{E}\,\bar{x}_k\|_\Sigma^2 \\ =&\mathbb{E}\,\|\bar{x}_k\|_\Sigma^2 - \|\mathbb{E}\,\bar{x}_k\|_\Sigma^2 \end{aligned} \tag{41}$$

Combining with the fact that $\mathbb{E}_{w_k}\widehat{\nabla \boldsymbol{f}}(\bar{x}_k\mathbb{1}^\top) = \nabla F(\bar{x}_k)$, we establish

$$\mathbb{E}\,\|\widehat{\nabla \boldsymbol{f}}(\bar{x}_k\mathbb{1}^\top)\|_\Sigma^2 = \mathbb{E}\,\|\widehat{\nabla \boldsymbol{f}}(\bar{x}_k\mathbb{1}^\top) - \nabla F(\bar{x}_k)\|_\Sigma^2 + \|\nabla F(\bar{x}_k)\|_\Sigma^2 \tag{42}$$

The first term in the above equality can be further bounded through the data heterogeneity assumption that

$$\begin{aligned}\mathbb{E}\,\|\widehat{\nabla \boldsymbol{f}}(\bar{x}_k\mathbb{1}^\top) - \nabla F(\bar{x}_k)\|_\Sigma^2 =& \frac{1}{m^2}\mathbb{E}\,\Big\|\sum_{i\in C_r}\Big(\nabla f_i(\bar{x}_k;\xi_k) - \nabla F(\bar{x}_k)\Big)\Big\|_\Sigma^2\\ =& \frac{1}{mM}\sum_{i=1}^{M}\|\nabla f_i(\bar{x}_k;\xi_k) - \nabla F(\bar{x}_k)\|_\Sigma^2\\ \le& \frac{1}{m}\|\Sigma\|(\sigma_G^2+\sigma_s^2)\end{aligned} \tag{43}$$

where the second equality holds since the zero-mean property. Substituting the above results back to equation 40, we arrive

$$\begin{aligned}\mathbb{E}\,\|\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|_\Sigma^2 \le& 2\|\nabla F(\bar{x}_k)\|_\Sigma^2 + \frac{2}{m}\|\Sigma\|(\sigma_G^2+\sigma_s^2) + 2\mathbb{E}\,\|\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) - \widehat{\nabla \boldsymbol{f}}(\bar{x}_k\mathbb{1}^\top)\|_\Sigma^2\\ \le& 2\|\nabla F(\bar{x}_k)\|_\Sigma^2 + \frac{2}{m}\|\Sigma\|(\sigma_G^2+\sigma_s^2) + 2L^2\|\Sigma\|\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\top\|_F^2/M\end{aligned} \tag{44}$$

where we applied the $L-$ Lipschitz condition and Jensen's inequality in the last step. ∎

G.4. Descent Lemma

Lemma 6 *When $\eta \le \left\{\frac{\beta_\ell}{mL}, \frac{1}{8\rho_k}\right\}$, the virtual centralized iterates $\bar{x}_k$ of one round satisfy*

$$\begin{aligned}\mathbb{E}\,F(\bar{x}_{(r+1)\tau+1}) \le& \mathbb{E}\,F(\bar{x}_{r\tau+1}) - \frac{\eta}{4}\sum_{j=r\tau+1}^{(r+1)\tau}\|\nabla F(\bar{x}_j)\|_{H_r^{-1}}^2 + O(\eta^2\mu)\\ &+ \frac{4\tau\eta^2}{\beta_\ell m}\sum_{j=r\tau+1}^{(r+1)\tau}\rho_k(\sigma_G^2+\sigma_s^2) + \frac{2L}{mM}\sum_{j=r\tau+1}^{(r+1)\tau}\|\boldsymbol{x}_j - \bar{x}_j\mathbb{1}^\top\|_F^2\end{aligned} \tag{45}$$

where $\rho_k = \mathrm{Tr}(H_k^{-1/2}\Sigma_k H_k^{-1/2}) + 2\|H_k^{-1/2}\Sigma_k H_k^{-1/2}\|$. ∎

Proof. Recall there are two random variables in the main recursion Eq. (28), one is the ZO random direction z_k and the other is the client sampling vector w_k. First, taking the conditional expectation over w_k, we have

$$\mathbb{E}_{w_k}\bar{x}_{k+1} = \bar{x}_k - \eta z_k z_k^\top\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) + \mathcal{O}(\eta\mu) \tag{46}$$

for any iteration k. Then, taking conditional expectation over z_k, we have

$$\mathbb{E}\,\bar{x}_{k+1} = \bar{x}_k - \eta H_k^{-1}\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) + \mathcal{O}(\eta\mu) \tag{47}$$

As a result of Assumption 1, there is a semi-positive definite matrix $\Sigma_y \preceq L\cdot I_d$ such that the global loss function satisfies

$$F(x) \le F(y) + \langle\nabla F(y), x-y\rangle + \frac{1}{2}(x-y)^\top\Sigma_y(x-y). \tag{48}$$

Hence, we have

$$F(\bar{x}_{k+1}) \le F(\bar{x}_k) + \langle\nabla F(\bar{x}_k), \bar{x}_{k+1}-\bar{x}_k\rangle + \frac{1}{2}(\bar{x}_{k+1}-\bar{x}_k)^\top\Sigma_k(\bar{x}_{k+1}-\bar{x}_k) \tag{49}$$

Now, substituting Eq. (28) into the above expansion and taking the conditional expectation, we will establish the following two cases.

Local Update Iteration:

When the iteration k is not the communication iteration, i.e. $k \neq r\tau$, we have

$$\begin{aligned} \mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \eta \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\top H_k^{-1} \nabla F(\bar{x}_k) + O(\eta^2 \mu) \\ & + \eta^2 \mathbb{E}\, [\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\top z_k z_k^\top \Sigma_k z_k z_k^\top \widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)] \end{aligned} \tag{50}$$

First, we focus on the cross term

$$\begin{aligned} -\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\top H_k^{-1} \nabla F(\bar{x}_k) =& - \nabla F(\bar{x}_k)^\top H_k^{-1} \nabla F(\bar{x}_k) + (\nabla F(\bar{x}_k) - \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k))^\top H_k^{-1} \nabla F(\bar{x}_k) \\ \leq & - \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + \frac{1}{2} \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + \frac{1}{2} \|\nabla F(\bar{x}_k) - \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|^2_{H_k^{-1}} \\ =& - \frac{1}{2} \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + \frac{1}{2} \|\nabla F(\bar{x}_k) - \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|^2_{H_k^{-1}} \end{aligned} \tag{51}$$

Because of Assumption 4, we have $\beta_u^{-1} \leq \|H_k^{-1}\| \leq \beta_\ell^{-1}$, which implies

$$\begin{aligned} \frac{1}{2} \|\nabla F(\bar{x}_k) - \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|^2_{H_k^{-1}} \leq & \frac{1}{2\beta_\ell} \|\nabla F(\bar{x}_k) - \overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|^2 \\ \leq & \frac{1}{2\beta_\ell N} \sum_{i=1}^{M} \|\nabla f_i(\bar{x}_k) - \nabla f_i(x_k^{(i)})\|^2 \\ =& \frac{L^2}{2\beta_\ell N} \|\boldsymbol{x}_k - \bar{x}_k \mathbb{1}^\top\|_F^2 \end{aligned} \tag{52}$$

Substituting back, we have

$$\begin{aligned} \mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \frac{\eta}{2} \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + \frac{\eta L^2}{2\beta_\ell N} \|\boldsymbol{x}_k - \bar{x}_k \mathbb{1}^\top\|_F^2 \\ & + \eta^2 \underbrace{\mathbb{E}\, [\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\top z_k z_k^\top \Sigma_k z_k z_k^\top \widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)]}_{:=Q} \end{aligned} \tag{53}$$

Next, the key is this quadratic term. Leveraging Lemma 3, we establish

$$\begin{aligned} Q =& \mathbb{E}_{w_k} \left(\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\top \left(\operatorname{Tr}(\Sigma_k H_k^{-1}) H_k^{-1} + 2 H_k^{-1} \Sigma_k H_k^{-1} \right) \widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k) \right) \\ \leq & (\operatorname{Tr}(\Sigma_k H_k^{-1}) + 2\|H^{-1/2} \Sigma_k H^{-1/2}\|) \mathbb{E}_{w_k} \|\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\|^2_{H_k^{-1}} \end{aligned} \tag{54}$$

where we utilize the following inequality in the last step

$$\|x\|^2_{H_k^{-1} \Sigma_k H_k^{-1}} = \operatorname{Tr}(H_k^{-1/2} x x^\top H_k^{-1/2} H_k^{-1/2} \Sigma_k H_k^{-1/2}) \leq \|H_k^{-1/2} \Sigma_k H_k^{-1/2}\| \|x\|^2_{H_k^{-1}}.$$

For simplicity, we introduce the matrix $\Xi_k = H_k^{-1/2} \Sigma_k H_k^{-1/2}$. Plugging the previous sampling noise variance result (44), we establish

$$Q \leq (\operatorname{Tr}(\Xi_k) + 2\|\Xi_k\|) \Big(2 \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + \frac{2}{\beta_\ell m} (\sigma_G^2 + \sigma_s^2) + \frac{2L^2}{\beta_\ell M} \|\boldsymbol{x}_k - \bar{x}_k \mathbb{1}^\top\|_F^2 / M \Big) \tag{55}$$

This $\operatorname{Tr}(\Xi_k) + 2\|\Xi_k\|$ is the key quantity that we will encounter repeatedly. To further reduce the notation, we denote $\rho_k = \operatorname{Tr}(\Xi_k) + 2\|\Xi_k\|$ Combining all the above results, we have

$$\begin{aligned} \mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \Big(\frac{\eta}{2} - 2\eta^2 \rho_k \Big) \|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + O(\eta^2 \mu) \\ & + \Big(\frac{\eta L^2}{2\beta_\ell M} + \frac{2\eta^2 L^2 \rho_k}{\beta_\ell M} \Big) \|\boldsymbol{x}_k - \bar{x}_k \mathbb{1}^\top\|_F^2 + \frac{2\eta^2 \rho_k}{\beta_\ell m} (\sigma_G^2 + \sigma_s^2) \end{aligned} \tag{56}$$

When $\eta \leq \frac{1}{4\rho_k}$, the coefficients can be simplified into

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \frac{\eta}{4}\|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + O(\eta^2\mu) \\ & + \frac{\eta L^2}{\beta_\ell M}\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2 + \frac{2\eta^2\rho_k}{\beta_\ell m}(\sigma_G^2 + \sigma_s^2)\end{aligned} \tag{57}$$

Communication Iteration:

When the iteration k is the communication iteration, i.e. $k \neq r\tau$, we have

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \eta\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\mathsf{T} H_k^{-1}\nabla F(\bar{x}_k) + O(\eta^2\mu) \\ & + \mathbb{E}\left(\hat{x}_k - \bar{x}_k - \eta\eta z_k z_k^\mathsf{T}\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\right)^\mathsf{T}\Sigma_k\left(\hat{x}_k - \bar{x}_k - \eta\eta z_k z_k^\mathsf{T}\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)\right) \\ \leq & F(\bar{x}_k) - \eta\overline{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\mathsf{T} H_k^{-1}\nabla F(\bar{x}_k) + O(\eta^2\mu) \\ & + 2\mathbb{E}\,(\hat{x}_k - \bar{x}_k)^\mathsf{T}\Sigma_k(\hat{x}_k - \bar{x}_k) + 2\eta^2\mathbb{E}[\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)^\mathsf{T} z_k z_k^\mathsf{T}\Sigma_k z_k z_k^\mathsf{T}\widehat{\nabla \boldsymbol{f}}(\boldsymbol{x}_k)]\end{aligned} \tag{58}$$

Next, we notice that

$$\mathbb{E}\,(\hat{x}_k - \bar{x}_k)^\mathsf{T}\Sigma_k(\hat{x}_k - \bar{x}_k) \leq L\mathbb{E}\,\|\hat{x}_k - \bar{x}_k\|^2 = \frac{L}{mM}\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2 \tag{59}$$

Utilizing previously established result Eq. (56), we have

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \left(\frac{\eta}{2} - 4\eta^2\rho_k\right)\|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + O(\eta^2\mu) \\ & + \left(\frac{L}{m} + \frac{\eta L^2}{2\beta_\ell} + \frac{4\eta^2 L^2}{\beta_\ell M}\rho_k\right)\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2 + \frac{4\eta^2\rho_k}{\beta_\ell m}(\sigma_G^2 + \sigma_s^2)\end{aligned} \tag{60}$$

When $\eta \leq \frac{1}{8\rho_k}$, the coefficients can be simplified into

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \frac{\eta}{4}\|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + O(\eta^2\mu) \\ & + \left(\frac{L}{mM} + \frac{\eta L^2}{\beta_\ell M}\right)\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2 + \frac{4\eta^2\rho_k}{\beta_u m}(\sigma_G^2 + \sigma_s^2)\end{aligned} \tag{61}$$

We further require the learning rate $\eta \leq \frac{\beta_\ell}{mL}$ to establish

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{k+1}) \leq & F(\bar{x}_k) - \frac{\eta}{4}\|\nabla F(\bar{x}_k)\|^2_{H_k^{-1}} + O(\eta^2\mu) \\ & + \frac{2L}{mM}\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2 + \frac{4\eta^2\rho_k}{\beta_\ell m}(\sigma_G^2 + \sigma_s^2)\end{aligned} \tag{62}$$

Combining Two into One Round:

Combining the above two results and iterating from $k = r\tau + 1$ to $k = (r+1)\tau$, we establish

$$\begin{aligned}\mathbb{E}\, F(\bar{x}_{(r+1)\tau+1}) \leq & \mathbb{E}\, F(\bar{x}_{r\tau+1}) - \frac{\eta}{4}\sum_{j=r\tau+1}^{(r+1)\tau}\|\nabla F(\bar{x}_j)\|^2_{H_r^{-1}} + O(\eta^2\mu) \\ & + \frac{4\tau\eta^2\rho_k}{\beta_\ell m}(\sigma_G^2 + \sigma_s^2) + \frac{2L}{mM}\sum_{j=r\tau+1}^{(r+1)\tau}\|\boldsymbol{x}_j - \bar{x}_j\mathbb{1}^\mathsf{T}\|_F^2\end{aligned} \tag{63}$$

where we can absorb the coefficients on the consensus term $\|\boldsymbol{x}_j - \bar{x}_j\mathbb{1}^\mathsf{T}\|_F^2$ into $2L/mM$ since above we already require the learning rate $\eta \leq \frac{\beta_\ell}{mL}$. Also, we replace H_k by H_r since it is not updated within one communication round. ∎

G.5. Consensus Lemma

Lemma 7 *When* $\eta \leq \frac{\beta_\ell}{4(\tau-1)}\sqrt{\frac{1}{L(d+2)}}$, *the sum of the consensus error of one round is bounded by the following term*

$$\frac{1}{\tau}\sum_{k=r\tau+1}^{(r+1)\tau}\mathbb{E}\,\|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\top\|_F^2 \leq 4\eta^2(\tau-1)^2 M\beta_\ell^{-1}\|\Phi_r\|(\sigma_G^2+\sigma_s^2) + O(\eta^2\mu^2) \tag{64}$$

where $\Phi_r := \mathrm{Tr}(H_r^{-1}) + 2H_r^{-1}$. ■

Proof. The consensus residual is defined as

$$\|\boldsymbol{x}_{k+1} - \bar{x}_{k+1}\mathbb{1}^\top\|_F^2 = \|\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\top - \eta(z_k z_k^\top \nabla \boldsymbol{f}(\boldsymbol{x}_k) - z_k z_k^\top \nabla \boldsymbol{f}(\boldsymbol{x}_k)\mathbb{1}_u\mathbb{1}^\top) + O(\eta\mu)\|_F^2 \tag{65}$$

If $k = r\tau$, all clients have the same value. Hence, we can expand the difference $\boldsymbol{x}_k - \bar{x}_k\mathbb{1}^\top$ up to $k = r\tau$ and arrive at

$$\begin{aligned}
&\|\boldsymbol{x}_{k+1} - \bar{x}_{k+1}\mathbb{1}^\top\|_F^2\\
=&\left\|\eta\sum_{j=r\tau+1}^{k}\left(z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j) - z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j)\mathbb{1}_u\mathbb{1}^\top\right) + O(\eta\mu)\right\|_F^2\\
\leq&(\tau-1)\sum_{j=r\tau+1}^{k}\eta^2\|z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j) - z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j)\mathbb{1}_u\mathbb{1}^\top\|_F^2 + O(\eta^2\mu^2)
\end{aligned} \tag{66}$$

where we utilize Jensen's inequality in the above step. Next, we focus on the term in the summation

$$\begin{aligned}
&\|z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j) - z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j)\mathbb{1}_u\mathbb{1}^\top\|_F^2\\
\leq&4\|z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j) - z_j z_j^\top \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top)\|_F^2 + 2\|z_j z_j^\top \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top) - z_j z_j^\top \nabla F(\bar{x}_j\mathbb{1}^\top)\mathbb{1}^\top\|_F^2\\
&+4\|z_j z_j^\top \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top)\mathbb{1}_u\mathbb{1}^\top - z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j)\mathbb{1}_u\mathbb{1}^\top\|_F^2\\
\leq&8\|z_j z_j^\top \nabla \boldsymbol{f}(\boldsymbol{x}_j) - z_j z_j^\top \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top)\|_F^2 + 2\|z_j z_j^\top \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top) - z_j z_j^\top \nabla F(\bar{x}_j\mathbb{1}^\top)\mathbb{1}^\top\|_F^2
\end{aligned} \tag{67}$$

where we utilize the identity that $\nabla \boldsymbol{F}(\bar{x}_j\mathbb{1}^\top) = \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top)\mathbb{1}_u$. Recall that

$$\mathbb{E}\,z_j z_j^\top z_j z_j^\top = \mathrm{Tr}(H_r^{-1})H_r^{-1} + 2H_r^{-2} := \Phi_r H_r^{-1} \tag{68}$$

where r is the corresponding round for the iteration j. Notice $\|\Phi_r\| \leq (d+2)/\beta_\ell$, which is not a tight bound though. Hence, taking the expectation with respect to z_j, we establish

$$\begin{aligned}
&\mathbb{E}\,\|\boldsymbol{x}_{k+1} - \bar{x}_{k+1}\mathbb{1}^\top\|_F^2\\
\leq&8\eta^2(\tau-1)\sum_{j=r\tau+1}^{k}\|\nabla \boldsymbol{f}(\boldsymbol{x}_j) - \nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top)\|_{\Phi_r H_r^{-1}}^2\\
&+2\eta^2(\tau-1)\sum_{j=r\tau+1}^{k}\|\nabla \boldsymbol{f}(\bar{x}_j\mathbb{1}^\top) - \nabla F(\bar{x}_j\mathbb{1}^\top)\mathbb{1}^\top\|_{\Phi_r H_r^{-1}}^2 + O(\eta^2\mu^2)\\
\leq&8\eta^2(\tau-1)L\beta_\ell^{-1}\|\Phi_r\|\sum_{j=r\tau+1}^{k}\|\boldsymbol{x}_j - \bar{x}_j\mathbb{1}^\top\|_F^2 + 2\eta^2(\tau-1)^2 M\beta_\ell^{-1}\|\Phi_r\|(\sigma_G^2+\sigma_s^2) + O(\eta^2\mu^2)
\end{aligned} \tag{69}$$

Lastly, we just need to take another summation over k from $r\tau$ to $(r+1)\tau-2$. Recall that $\|\boldsymbol{x}_{r\tau+1} - \bar{x}_{r\tau+1}\mathbb{1}^\top\|_F^2 = 0$. After rearranging and utilizing the fact that $\sum_{k=r\tau}^{(r+1)\tau-2}\sum_{j=r\tau+1}^{k} a_j \leq$

$(\tau-1)\sum_{k=r\tau+1}^{(r+1)\tau} a_k$ for any nonnegative value a_k, we have

$$\left(1-8\eta^2(\tau-1)^2 L\beta_\ell^{-1}\|\Phi_r\|\right)\frac{1}{\tau}\sum_{k=r\tau+1}^{(r+1)\tau}\|\mathbb{E}\,\|\boldsymbol{x}_k-\bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2$$
$$\leq 2\eta^2(\tau-1)^2 M\beta_\ell^{-1}\|\Phi_r\|(\sigma_G^2+\sigma_s^2)+O(\eta^2\mu^2) \tag{70}$$

After restricting η to force $1-8\eta^2(\tau-1)^2 L\beta_\ell^{-1}\|\Phi_r\|<1/2$, we establish this lemma. ■

A special case is the local update steps $\tau=1$. In this case, we don't need any consensus error since the models are all synchronized. We can simply discard the term $\mathbb{E}\,\|\boldsymbol{x}_k-\bar{x}_k\mathbb{1}^\mathsf{T}\|_F^2$ in the descent lemma.

G.6. Convergence Proof of Theorem 1

Proof: We are now ready to present the convergence theorem, which simply combines the consensus lemma and the descent lemma above then taking the double exepection.

$$\mathbb{E}\left[F(\bar{x}_{(r+1)\tau+1})\right] \leq \mathbb{E}\left[F(\bar{x}_{r\tau+1})\right]-\frac{\eta}{4}\sum_{j=r\tau}^{(r+1)\tau-1}\mathbb{E}\,\|\nabla F(\bar{x}_j)\|^2_{H_r^{-1}}+O(\eta^2\mu)$$
$$+\frac{4\tau\eta^2\rho_k}{\beta_\ell m}(\sigma_G^2+\sigma_s^2)+\frac{8\eta^2(\tau-1)^2 L}{\tau m}\sum_{j=r\tau}^{(r+1)\tau-1}\|\Phi_r\|(\sigma_G^2+\sigma_s^2) \tag{71}$$

Expanding the summations and re-arranging terms, we obtain

$$\frac{1}{\tau R}\sum_{j=1}^{\tau R}\mathbb{E}\,\|\nabla F(\bar{x}_j)\|^2_{H_r^{-1}} \leq \frac{4(F(\bar{x}_1)-F^\star)}{\eta\tau R}+\frac{16\eta\bar{\rho}}{\beta_\ell m}(\sigma_G^2+\sigma_s^2)+\frac{32\eta(\tau-1)^2L\bar{\phi}}{\beta_\ell\tau m}(\sigma_G^2+\sigma_s^2)$$
$$+\mathcal{O}(\eta\mu), \tag{72}$$

where

$$\bar{\rho} = \frac{1}{K}\sum_{k=0}^{K}\rho_k=\frac{1}{K}\sum_{k=0}^{K}(\mathrm{Tr}(\Xi_k)+2\|\Xi_k\|) \tag{73}$$
$$= \frac{1}{K}\sum_{k=0}^{K}(\mathrm{Tr}(H_k^{-1/2}\Sigma_k H_k^{-1/2})+2\|H_k^{-1/2}\Sigma_k H_k^{-1/2}\|) \tag{74}$$
$$\bar{\phi} = \frac{1}{R}\sum_r\|\Phi_r\|=\frac{1}{R}\sum_r(\mathrm{Tr}(H_r^{-1})+2\|H_r^{-1}\|) \tag{75}$$

Combining all learning rate requirements, we have

$$\eta \leq \min\left(\frac{\beta_\ell}{mL},\frac{1}{8\rho_k},\frac{\beta_\ell}{4(\tau-1)}\sqrt{\frac{1}{L(d+2)}}\right) \tag{76}$$

Lastly, translating the above result back to the two-level k and r indexing, we establish Theorem 1.

G.6.1. CONVERGENCE RATE

To establish the convergence rate, we distinguish two scenarios – the local update $\tau = 1$ and the local update $\tau > 1$. When $\tau = 1$, the rate becomes much simpler

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\,\|\nabla F(\bar{x}_{r,0})\|^2_{H_r^{-1}} \leq \frac{4(F(\bar{x}_1)-F^\star)}{\eta R} + \frac{16\eta\bar{\rho}}{\beta_\ell m}(\sigma_G^2+\sigma_s^2) + \mathcal{O}(\eta\mu), \tag{77}$$

When the communication round R is sufficiently large and the ZO smoothing parameter μ is sufficiently small, we choose the learning rate $\eta = \sqrt{\frac{m\beta_\ell}{\bar{\rho}R}}$, which leads to the following rate:

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\,\|\nabla F(\bar{x}_{r,0})\|^2_{H_r^{-1}} = \mathcal{O}\left(\sqrt{\frac{\bar{\rho}}{mR}}\right) \tag{78}$$

Based on the Table 1, we can establish the following four rates based on the conditions:

1. H_r is a well-approximated one with L-smoothness assumption, then the rate is $\mathcal{O}\left(\sqrt{\frac{d}{mR}}\right)$.
2. H_r is a well-approximated one with low effective rank, then the rate is $\mathcal{O}\left(\sqrt{\frac{\zeta}{mR}}\right)$.
3. DeComFL Case: No Hessian information is learned, i.e., $H_k \equiv I$, with L-smoothness assumption, then the rate is $\mathcal{O}\left(\sqrt{\frac{Ld}{mR}}\right)$.
4. DeComFL Case: No Hessian information is learned, i.e., $H_k \equiv I$, with low effective rank, then the rate is $\mathcal{O}\left(\sqrt{\frac{L\kappa}{mR}}\right)$.

For the local update $\tau > 1$ case, we choose the learning rate $\eta = \min\left(\sqrt{\frac{m\beta_\ell}{\tau\bar{\rho}R}}, \sqrt{\frac{m\beta_\ell}{\tau\bar{\phi}R}}\right)$. Then we obtain the following rate

$$\frac{1}{\tau R}\sum_{r=0}^{R-1}\sum_{k=0}^{\tau-1}\mathbb{E}\,\|\nabla F(\bar{x}_{r,k})\|^2_{H_r^{-1}} = \underbrace{\mathcal{O}\left(\sqrt{\frac{\bar{\rho}}{\tau mR}}\right)}_{\text{descent residue}} + \underbrace{\mathcal{O}\left(\sqrt{\frac{\tau\bar{\phi}}{mR}}\right)}_{\text{consensus residue}} \tag{79}$$

where the second extra term comes from the client model diverging in the local update steps.

Similarly, we can establish the four rates based on the assumption. Here we focus on the low effective rank case since it reveals the difference between DeComFL and HiSo.

When $H_r \equiv I$, we have $\bar{\phi} = d + 2$ and $\bar{\rho} \leq L\kappa$. Therefore, we establish the following rate for DeComFL rate:

$$\mathcal{O}\left(\sqrt{\frac{L\kappa}{\tau mR}}\right) + \mathcal{O}\left(\sqrt{\frac{\tau d}{mR}}\right) \tag{80}$$

Here we can see that even if $\bar{\rho}$ can be tighter bounded by low-effective rank, the convergence rate still depends on d.

In contrast, if H_r well-approximates the Hessian Σ with the low effective rank, we establish the convergence rate for HiSo is

$$\mathcal{O}\left(\sqrt{\frac{\zeta}{\tau mR}}\right) + \mathcal{O}\left(\sqrt{\frac{\tau\kappa}{mR}}\right) \tag{81}$$

Now, if we compare Eq. (80) with Eq. (81), we can tell that HiSo is still capable of being independent of Lipschitz L and model dimension d; meanwhile, DeComFL cannot. This probably explains why the original paper [18] cannot provide the proof for the dimension-free rate with $\tau > 1$. Of course, Eq. (80) is just an upper bound for the worst-case scenario. The practical performance may not be pessimistic as the bound indicates.

Corollary 8 (Convergence Rate for DeComFL) Note that DeComFL [18] can be regarded as a special case of HiSo with $H_r \equiv I, \forall r$ and $\beta_\ell = \beta_u = 1$. Therefore, we can immediately recover the convergence rate of DeComFL with $\tau = 1$ is $\mathcal{O}(\sqrt{Ld/mR})$ with standard assumptions or $\mathcal{O}(\sqrt{L\kappa/mR})$ with the extra low-effective rank phenomenon.

Corollary 9 (Convergence Rate for $\tau > 1$ case) When the local update step $\tau > 1$, the difference between HiSo and DeComFL becomes bigger. Under the well-approximate and low whitening rank scenario, the convergence rate of HiSo is $\mathcal{O}(\sqrt{\zeta/\tau mR}) + \mathcal{O}(\sqrt{\tau\kappa/mR})$, still independent of the model dimension d and Lipschitz condition L; meanwhile, DeComFL becomes dependent on d again. This resolved the previous open question that DeComFL [18] cannot provide the convergence rate with a low-effective rank assumption when $\tau > 1$. See Appendix G.6.1 for details.

Appendix H. Multi-Perturbation Version

Following our detailed examination of ZO-gradient variance, it is evident that reducing this variance is crucial for enhancing the performance of ZO-based methods. In this context, **multi-perturbation sampling in ZO-SGD can be viewed as analogous to mini-batching in standard SGD**, where multiple samples are used to improve the quality of the gradient estimate.

In terms of HiSo, the multi-perturbation version is simply replacing the finding $\Delta x_{r,k}^{(i)}$ step by the following:

$$\boxed{\begin{aligned} &\text{for } p = 0, 1 \cdots, P-1: \\ &\qquad u_{r,k,p} \sim \mathcal{N}(0, I) \\ &\qquad g_{r,k,p}^{(i)} = \frac{1}{\mu}[f_i(x_{r,k}^{(i)} + \mu H_r^{-1/2} u_{r,k,p}) - f_i(x_{r,k}^{(i)})] \\ &\Delta x_{r,k}^{(i)} = H_r^{-1/2} \frac{1}{P} \sum_{p=0}^{P-1} g_{r,k,p}^{(i)} u_{r,k,p} \end{aligned}} \tag{82}$$

Notice for the multi-perturbation version, we need to transmit P random seeds to generate p random vector $u_{r,k,p}$. Moreover, P local gradient scalars $g_{r,k,p}^{(i)}$ are required to be communicated as well.

At the server side, the aggregation step now is required to average P values separately:

$$\boxed{\Delta x_{r,k} = \frac{1}{\tau |C_r|} \sum_{i \in C_r} \sum_{k=0}^{\tau-1} \Delta x_{r,k}^{(i)} = \frac{1}{\tau} \sum_{k=0}^{\tau-1} \left[\frac{1}{P} \sum_{p=0}^{P-1} \underbrace{\left(\frac{1}{|C_r|} \sum_{i \in C_r} g_{r,k,p}^{(i)} \right)}_{:= g_{r,k,p}} H_r^{-1/2} u_{r,k,p} \right]} \tag{83}$$

Notice we can switch the order of summation in above equations because $u_{r,k,p}$ is common among all clients. This aggregated gradient scalar $g_{r,k,p}$ stands for the r-th round, k-th local update, and p-th perturbation. P gradient scalars together with P random seeds are sufficient to reconstruct the global $\Delta x_{r,k}$. For the reconstruction step, everything is the same.

H.1. Performance Analysis

Theorem 10 (Multi-Perturbation Version) *Under Assumptions 1, 2, 3 and 4, if* $\eta \leq \min\left(\frac{\beta_\ell}{mL}, \frac{1}{8\rho_{k,P}}, \frac{\beta_\ell}{4(\tau-1)}\sqrt{\frac{1}{L(d+2)}}\right)$ *the sequence of iterates generated by HiSo with* P *perturbations satisfies:*

$$\frac{1}{\tau R}\sum_{r=0}^{R-1}\sum_{k=0}^{\tau-1}\mathbb{E}\,\|\nabla F(\bar{x}_{r,k})\|^2_{H_r^{-1}} \leq \frac{4(F(\bar{x}_1)-F^\star)}{\eta\tau R} + \underbrace{\frac{32\eta(\tau-1)^2L\bar{\phi}_P}{\beta_\ell\tau m}(\sigma_G^2+\sigma_s^2)}_{\text{extra client drift term}} + \frac{16\eta\bar{\rho}_P}{\beta_\ell m}(\sigma_G^2+\sigma_s^2) + \mathcal{O}(\eta\mu), \tag{84}$$

where

$$\bar{\rho}_P = \frac{1}{\tau R}\sum_r\sum_k\left(\frac{1}{P}\operatorname{Tr}(H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2}) + (\frac{1}{P}+1)\|H_r^{-1/2}\Sigma_{r,k}H_r^{-1/2}\|\right) \tag{85}$$

$$\bar{\phi}_P = \frac{1}{R}\sum_r\left(\frac{1}{P}\operatorname{Tr}(H_r^{-1}) + (\frac{1}{P}+1)\|H_r^{-1}\|\right) \tag{86}$$

and the rest of the quantities are the same as Theorem 1. ■

Proof: In this case, the algorithm formulation can be written as

$$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k - \eta\frac{1}{P}\sum_{p=1}^{P} z_{k,p}z_{k,p}^{\mathsf{T}}\nabla \boldsymbol{f}(\boldsymbol{x}_k;\xi_k) + O(\mu\eta), \tag{87}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{y}_{k+1}W_k, \tag{88}$$

Notice there are three sources of the randomness – random direction z, gradient noise coming from ξ_km and the sampling randomness W_k. They are independent of each other, so we can treat them one by one separately. It is straightforward to verify that the mean is unchanged

$$\mathbb{E}\,\frac{1}{P}\sum_{p=1}^{P} z_{k,p}z_{k,p}^{\mathsf{T}}\nabla \boldsymbol{f}(\boldsymbol{x}_k;\xi_k) = H_k^{-1}\nabla \boldsymbol{f}(\boldsymbol{x}_k) \tag{89}$$

Next, noting $\{z_{k,p}\}_p$ is independent and identically distributed, utilizing lemma 3 we establish

$$\begin{aligned}
&\frac{1}{P^2}\sum_{p'=1}^{P}\sum_{p=1}^{P}\mathbb{E}\,z_{k,p}z_{k,p}^{\mathsf{T}}\Sigma_k z_{k,p'}z_{k,p'}^{\mathsf{T}}\\
=&\frac{P^2-P}{P^2}H_k^{-1}\Sigma_kH_k^{-1} + \frac{1}{P^2}\sum_{p=1}^{P}\mathbb{E}\,z_{k,p}z_{k,p}^{\mathsf{T}}\Sigma_k z_{k,p}z_{k,p}^{\mathsf{T}}\\
=&\frac{P-1}{P}H_k^{-1}\Sigma_kH_k^{-1} + \frac{1}{P}(\operatorname{Tr}(\Sigma_kH_k^{-1})H_k^{-1} + 2H_k^{-1}\Sigma_kH_k^{-1})\\
=&\frac{1}{P}\operatorname{Tr}(\Sigma_kH_k^{-1})H_k^{-1} + \left(\frac{1}{P}+1\right)H_k^{-1}\Sigma_kH_k^{-1}
\end{aligned} \tag{90}$$

Recall that this quantity ρ_k of the single perturbation case is

$$\rho_k = \operatorname{Tr}(H_k^{-1/2}\Sigma_kH_k^{-1/2}) + 2\|H_k^{-1/2}\Sigma_kH_k^{-1/2}\|^2$$

The multi-perturbation version one will become

$$\rho_{k,P} = \frac{1}{P}\operatorname{Tr}(H_k^{-1/2}\Sigma_kH_k^{-1/2}) + \left(\frac{1}{P}+1\right)\|H_k^{-1/2}\Sigma_kH_k^{-1/2}\|^2 \approx \frac{1}{P}\rho_k$$

Recall that the first term in ρ_k is typically much bigger than the second one. Hence, $\rho_{k,P} \approx \rho_k/P$ as we expect that multi-perturbation will decrease the variance of the random search direction.

Besides, it is a similar case applied to quantity:

$$\frac{1}{P^2}\sum_{p'=1}^{P}\sum_{p=1}^{P}\mathbb{E}\, z_{k,p}z_{k,p}^{\mathsf{T}}z_{k,p'}z_{k,p'} = \frac{1}{P}\operatorname{Tr}(H_k^{-1})H_k^{-1} + \left(\frac{1}{P}+1\right)H_k^{-1}H_k^{-1} \tag{91}$$

So that the multi-perturbation version of $\phi_{r,P}$ will become

$$\phi_{r,P} = \frac{1}{P}\operatorname{Tr}(H_r^{-1}) + \left(\frac{1}{P}+1\right)\|H_r^{-1}\|^2 \approx \frac{1}{P}\phi_r$$

Notice we just need to update the Eq. (54) with the result of Eq. (90). After some calculations and simplification, we establish the result of Theorem 10.

H.2. Convergence Rate

Notice the relationship $\rho_{k,P} \approx \rho_k/P$, we can immediately establish that for $\tau = 1$ the convergence rate of HiSo is $\mathcal{O}\left(\sqrt{\frac{\bar{\rho}_P}{mR}}\right)$. Further, under the well-approximated Hessian assumption, we can establish the dimension-free rate

$$\frac{1}{R}\sum_{r=0}^{R-1}\|\nabla F(\bar{x}_{r,0})\|^2_{H_r^{-1}} = \mathcal{O}\left(\sqrt{\frac{\zeta}{mPR}}\right) \tag{92}$$

When $\tau > 1$, we have $\mathcal{O}\left(\sqrt{\frac{\bar{\rho}}{\tau mR}}\right) + \mathcal{O}\left(\sqrt{\frac{\tau\bar{\phi}}{mR}}\right)$. Further, under the well-approximated Hessian assumption, we can establish the dimension-free rate

$$\frac{1}{\tau R}\sum_{r=0}^{R-1}\sum_{k=0}^{\tau-1}\|\nabla F(\bar{x}_{r,k})\|^2_{H_r^{-1}} = \mathcal{O}\left(\sqrt{\frac{\zeta}{\tau mPR}}\right) + \mathcal{O}\left(\sqrt{\frac{\tau\kappa}{mPR}}\right) \tag{93}$$