Toward the First Optimization Framework for Low-Rank Adaptation

Grigory Malinovsky GRIGORII.MALINOVSKII@KAUST.EDU.SA
KAUST, Saudi Arabia
Umberto Michieli
Samsung R&D Institute, UK
Hasan Abed Al Kader Hammoud
KAUST, Saudi Arabia
Taha Ceritli
Samsung R&D Institute, UK
Hayder Elesedy
Samsung R&D Institute, UK
Mete Ozay
Samsung R&D Institute, UK
Peter Richtárik
KAUST, Saudi Arabia

Abstract

Fine-tuning is a common approach for adapting large foundational models to downstream tasks. With growing model and dataset sizes, parameter-efficient techniques have become crucial. A widely used method is Low-Rank Adaptation (LoRA), which expresses updates as a product of two low-rank matrices. While effective, LoRA often lags behind full-parameter fine-tuning (FPFT), and its optimization theory remains underexplored. We show that LoRA and its extensions, Asymmetric LoRA and Chain of LoRA, face convergence issues. To address this, we propose Randomized Asymmetric Chain of LoRA (RAC-LoRA)—a general framework analyzing convergence rates of LoRA-based methods. Our approach keeps the empirical benefits of LoRA while introducing algorithmic modifications that ensure provable convergence. The framework bridges FPFT and low-rank adaptation, guaranteeing convergence to the FPFT solution with explicit rates. We further provide analysis for smooth non-convex losses under gradient descent, stochastic gradient descent, and federated learning, supported by experiments.

1. Introduction

Many real-world Deep Learning (DL) applications require adapting large pre-trained models to specific tasks [8]. This process, known as fine-tuning, adjusts a model from its pre-trained state to new domains. Fine-tuning is a form of transfer learning, where pre-training knowledge is reused for specific applications [60].

Parameter-Efficient Fine-Tuning. Full-parameter fine-tuning is effective but impractical for modern models with billions of parameters. Parameter-Efficient Fine-Tuning (PEFT) [19] reduces cost by updating only a subset of parameters [51] or adding task-specific modules [65]. PEFT lowers training time, memory requirements, making it a practical choice for large-scale models [18].

1.1. Low-Rank Adaptation (LoRA)

A widely used PEFT method is Low-Rank Adaptation (LoRA) [21], motivated by the low intrinsic dimension of fine-tuning [1, 33]. Instead of updating full weight matrices, LoRA optimizes the product of two trainable low-rank matrices, added to the pre-trained weights: $W = W^0 + \frac{\alpha}{r}BA$, where $W^0 \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times r}$, $A \in \mathbb{R}^{r\times n}$, and $r \ll \min\{m, n\}$. Only A, B are trained, while W^0 stays fixed. The scaling α/r normalizes the update, with A usually Gaussian-initialized and B set to zero; alternatives are explored in Zhu et al. [67]. This approach reduces computation and mitigates overfitting [4], enabling efficient fine-tuning in resource-limited settings [59].

1.2. Chain of LoRA (COLA)

Although efficient, LoRA often underperforms full-parameter fine-tuning (FPFT) [4]. To improve, Xia et al. [64] proposed Chain of LoRA (COLA), which applies multiple LoRA updates iteratively. At each step, a LoRA module is trained, merged with W^0, and reinitialized. After T iterations, the model becomes $W = W^0 + \frac{\alpha}{r}\sum_{t=0}^{T-1} B^t A^t$, with A^t, B^t the low-rank matrices at step t. Unlike a single low-rank update, COLA approximates higher-rank adaptations via a sequence of decompositions, potentially yielding better performance and easier optimization [64].

1.3. Problem Formulation

Supervised learning is typically cast as minimizing a loss function that measures the discrepancy between predictions and outcomes. Here, we focus on fine-tuning, where a pre-trained model is adapted to a new task or dataset with efficient parameter updates. We consider the model-agnostic formulation

$$\min_{\Delta W \in \mathbb{R}^{m\times n}} f(W^0 + \Delta W), \tag{1}$$

where $W^0 \in \mathbb{R}^{m\times n}$ are pre-trained parameters (or those of a single linear layer), and ΔW is the adaptation term. The function $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ denotes the empirical loss on the adaptation dataset. Since $m \times n$ is typically huge in deep learning, ΔW must have special structure to be practical.

1.4. No Reasonable Theory for Low-Rank Adaptation

A solid theoretical understanding of fine-tuning methods with low-rank updates, such as LoRA and COLA, is still missing.

- As noted by Sun et al. [58], the LoRA re-parameterization transforms a *smooth* Lipschitz loss into a *non-smooth* one, creating new challenges in addition to handling low-rank updates. While this hints at difficulties, it does not rule out a proper theory.
- Existing analysis of COLA [64] replaces low-rank optimization over A, B with full-rank optimization of ΔW, ignoring the key low-rank structure and making results largely irrelevant.
- LoRA is highly sensitive to hyper-parameters [29, 32], yet no theory explains this issue.
- Most importantly, COLA can *fail to converge*. In Section 2, we present a simple 3×3 example showing divergence. Thus, COLA is merely a heuristic, and fixing this is an open problem—our focus in this work.

Despite their practical success, LoRA and COLA remain *heuristics* without rigorous theoretical support, raising concerns about their robustness and reliability in settings *beyond* current practice.

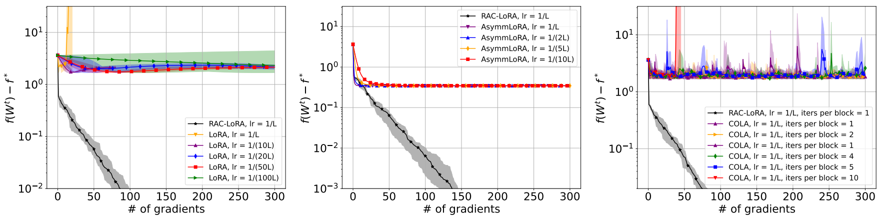

Figure 1: Convergence of LoRA, AsymmLoRA, COLA, and our proposed RAC-LoRA on (2).

2. Shining Some Light on LoRA's Convergence Issues

Most machine learning models are trained by minimizing a loss function using gradient-based methods [52], typically extensions of Gradient Descent (GD) with stochasticity, momentum, or adaptive stepsizes [15, 56]. To study LoRA-style methods, we first analyze their behavior under a GD step. Applying the chain rule, the gradients of the low-rank matrices B, A are $\nabla_{B,A} f(W + \frac{\alpha}{r}BA) = \frac{\alpha}{r}\begin{pmatrix}\nabla B^\top f(W + \frac{\alpha}{r}BA) \\ \nabla f(W + \frac{\alpha}{r}BA)A^\top\end{pmatrix}$, leading to updates $A^+ = A - \eta\frac{\alpha}{r}B^\top \nabla f(W + \frac{\alpha}{r}BA), \qquad B^+ = B - \eta\frac{\alpha}{r}\nabla f(W + \frac{\alpha}{r}BA)A^\top$, where $\eta > 0$ is the step size. Since both A, B are trainable, gradients are multiplied by $B^\top$ and $A^\top$, creating a non-trivial structure that complicates analysis and may disrupt Lipschitz continuity.

Loss of Lipschitz smoothness.

Assumption 1 (Lipschitz Gradient) *f is differentiable, and there exists $L > 0$ such that*

$$\|\nabla f(W) - \nabla f(V)\| \le L\|W - V\|, \quad \forall W, V \in \mathbb{R}^{m\times n},$$

where $\|\cdot\|$ is the Frobenius norm.

Lipschitz continuity of gradients is central for convergence guarantees [48, 57]. However, this property generally fails under LoRA. Even if $f(W)$ is L-smooth, the reparameterized function $f(W^0 + BA)$ is not Lipschitz smooth in variables $\{A, B\}$, as proven in Sun et al. [58, Thm. 2]. This loss of smoothness complicates extending standard gradient methods to LoRA.

Numerical counterexample. We illustrate failure modes of LoRA and COLA using the quadratic

$$f(x) = x^\top Mx + b^\top x, \tag{2}$$

with $d = 9$, $M = \mathrm{Diag}(10, 1, \ldots, 1)$, and $b = (1, \ldots, 1)^\top$. The function is L-smooth with $L = 10$. We represent $x \in \mathbb{R}^9$ as $W \in \mathbb{R}^{3\times 3}$, use rank $r = 1$, and set $\alpha = r$.

Figure 1 shows that with step size $1/L$, both LoRA and COLA diverge, while AsymmLoRA converges to a suboptimal stationary point. With smaller steps, LoRA and COLA converge but still far from optimal. In contrast, our RAC-LoRA converges linearly to the optimum. These results highlight divergence, suboptimal convergence, and step-size sensitivity in existing methods, and demonstrate the reliability of RAC-LoRA.

Algorithm 1 Randomized Asymmetric Chain of LoRA (RAC-LoRA)

1: **Parameters:** pre-trained model $W^0 \in \mathbb{R}^{m \times n}$, rank $r \ll \min\{m, n\}$, learning rate $\gamma > 0$, scaling factor $\alpha > 0$, chain length T, sketch distribution $\mathcal{D}_S^B$ (Option 1) or $\mathcal{D}_S^A$ (Option 2).
2: **for** $t = 0, 1, \ldots, T-1$ **do**
3: Sample a sketch matrix
(Option 1) $B_S^t \sim \mathcal{D}_S^B$ (Option 2) $A_S^t \sim \mathcal{D}_S^A$
4: Using some iterative solver, approximately solve the subproblem
(Option 1) $\hat{A}^t \approx \min_A f(W^t + \frac{\alpha}{r} B_S^t A)$ (Option 2) $\hat{B}^t \approx \min_B f(W^t + \frac{\alpha}{r} B A_S^t)$
5: Apply the update
(Option 1) $W^{t+1} = W^t + \frac{\alpha}{r} B_S^t \hat{A}^t$ (Option 2) $W^{t+1} = W^t + \frac{\alpha}{r} \hat{B}^t A_S^t$
6: **end for**

3. Randomized Asymmetric Chain of LoRA (RAC-LoRA)

To address convergence issues in LoRA, we propose Randomized Asymmetric Chain of LoRA (RAC-LoRA). This method combines asymmetric LoRA with a chain structure to ensure convergence while keeping efficiency. The procedure is summarized in Algorithm 1.

Description of the algorithm. At each block, one matrix is randomly initialized and fixed, while the other is trainable. This avoids optimization in a restricted subspace and reduces convergence to poor stationary points. Two configurations exist: freeze B and train A, or freeze A and train B. We formalize these sampling schemes as follows.

Definition 2 (Left Sketch) *A "left sketch" (rank r) is $\Delta W = \frac{\alpha}{r} B_S \hat{A}$, where $B_S \sim \mathcal{D}_B$ is sampled from a fixed distribution ($n \times r$), and only $\hat{A}$ is trainable.*

Definition 3 (Right Sketch) *A "right sketch" (rank r) is $\Delta W = \frac{\alpha}{r} \hat{B} A_S$, where $A_S \sim \mathcal{D}_A$ is sampled from a fixed distribution ($r \times m$), and only $\hat{B}$ is trainable.*

In both schemes, the trainable matrix is updated for several epochs, effectively training a LoRA block in the chain. Formally:
(Option 1) $\hat{A}^t \approx \min_A f(W^t + \frac{\alpha}{r} B_S^t A)$, (Option 2) $\hat{B}^t \approx \min_B f(W^t + \frac{\alpha}{r} B A_S^t)$.
Finally, the sampled and trained matrices are merged into the model:
(Option 1) $W^{t+1} = W^t + \frac{\alpha}{r} B_S^t \hat{A}^t$, (Option 2) $W^{t+1} = W^t + \frac{\alpha}{r} \hat{B}^t A_S^t$.

3.1. Derivation of the update step

Without loss of generality, let us focus on the Left Sketch scheme (Definition 2). Specifically, for each model in the chain, the update rule is given as follows: $W^{t+1} = W^t + \frac{\alpha}{r} B_S^t \hat{A}^t$. Next, we apply the Lipschitz gradient condition (Assumption 1) to the loss function f:

$$f(U) \leq f(V) + \langle \nabla f(V), U - V \rangle + \frac{L}{2} \|U - V\|_F^2, \quad \forall U, V \in \mathbb{R}^{m \times n}$$

Applying this with $U = W^t$, $V = B_S^t \hat{A}^t$ and $\eta \leq \frac{1}{L}$ leads to

$$f(W^{t+1}) \leq f(W^t) + \langle (B_S^t)^\top \nabla f(W^t), \hat{A}^t \rangle + \frac{1}{2\eta} \langle (B_S^t)^\top B_S^t \hat{A}^t, \hat{A}^t \rangle.$$

Let us minimize the left hand side term in $\hat{A}^t$, when the gradient vanishes: $(B_S^t)^\top \nabla f(W^t) + \frac{1}{\eta}(B_S^t)^\top (B_S^t)\hat{A}^t = 0$. One such solution is given by[1] $\hat{A}^t = -\eta \left((B_S^t)^\top (B_S^t)\right)^\dagger (B_S^t)^\top \nabla f(W^t)$, and his leads to the following gradient update:

$$W^{t+1} = W^t - \frac{\alpha}{r}\eta B_S^t \left((B_S^t)^\top (B_S^t)\right)^\dagger (B_S^t)^\top \nabla f(W^t) = W^t - \gamma H_B^t \nabla f(W^t), \tag{3}$$

where $H_B^t = B_S^t \left((B_S^t)^\top (B_S^t)\right)^\dagger (B_S^t)^\top$ is projection matrix and $\frac{\alpha}{r}\eta = \gamma$. Similarly, we can obtain the update for Right Sketch scheme (Definition 3):

$$W^{t+1} = W^t - \gamma \nabla f(W^t)(A_S^t)^\top \left(A_S^t (A_S^t)^\top\right)^\dagger A_S^t = W^t - \gamma \nabla f(W^t) H_A^t, \tag{4}$$

where $H_A^t = (A_S^t)^\top \left(A_S^t (A_S^t)^\top\right)^\dagger A_S^t$ is also projection matrix. We merge the scaling factor $\frac{\alpha}{r}$ with η into an effective step size γ, simplifying updates. With this, we establish convergence for both standard and stochastic gradient descent.

3.2. Convergence results

Our analysis relies on the smallest eigenvalue of the expected projection matrix (Section 3.1). A strictly positive value ensures reliable convergence, so we assume it remains positive.

Assumption 4 *Consider a projection matrix H generated by Left Sketch (Def. 2) or Right Sketch (Def. 3). Assume that sampling distributions $\mathcal{D}_S^B$ and $\mathcal{D}_S^A$ are such that the smallest eigenvalue of the expected projection matrix H generated by sampled matrix is positive: $\lambda_{\min}^H = \lambda_{\min}\left[\mathbb{E}\left[H\right]\right] > 0$.*

The projection matrix has eigenvalues 0 or 1, so its minimum is 0, but the expected projection matrix can have a strictly positive minimum eigenvalue. We require a lower bound on the function.

Remark 5 *Assumption 4 is easily satisfied. Let H be the projection matrix as defined below Equation (4) and assume that the A matrices are drawn from an isotropic distribution (the rows of A are isotropic). Then H is the projection onto the rank of A, which is a subspace of dimension r distributed isotropically in $\mathbb{R}^n$. The matrix $\mathbb{E}[H]$ is then invariant under rotations, so must be a scalar multiple of the identity. By taking traces, one finds that $\mathbb{E}[H] = \frac{r}{n}I$ so $\lambda_{\min}^H = \frac{r}{n}$.*

Assumption 6 *Function f is bounded from below by an infimum $f^\star \in \mathbb{R}$.*

We now present the convergence result for RAC-LoRA with Gradient Descent (GD) updates.

Theorem 7 *Let Assumptions 1 and 4 hold, and let the stepsize satisfy $0 < \gamma \leq \frac{1}{L}$. Then, the iterates of RAC-LoRA (Algorithm 1) with GD updates (Equation 3 or 4) satisfy*

$$\mathbb{E}\left[\left\|\nabla f(\widetilde{W}^T)\right\|^2\right] \leq \frac{2(f(W^0) - f^\star)}{\lambda_{\min}^H \gamma T},$$

where the output $\widetilde{W}^T$ is chosen uniformly at random from $W^0, W^1, \ldots, W^{T-1}$.

1. The dagger notation refers to the Moore-Penrose pseudoinverse.

We obtain sublinear convergence, typical for non-convex problems. To strengthen this, we assume the Polyak-Łojasiewicz (PL) condition, which extends strong convexity to non-convex functions.

Assumption 8 (PL-condition) *Function f satisfies the Polyak-Łojasiewicz (PL) condition with parameter $\mu > 0$ if $\frac{1}{2}\|\nabla f(W)\|^2 \geq \mu(f(W) - f^\star)$ for all $W \in \mathbb{R}^{m\times n}$, where $f^\star = \inf f$, assumed to be finite.*

Next, we establish a convergence rate for RAC-LoRA in the Polyak-Łojasiewicz setting.

Theorem 9 *Let Assumptions 1, 4 and 8 hold, and let the stepsize satisfy $0 < \gamma \leq \frac{1}{L}$. Then, for each $T \geq 0$, the iterates of* RAC-LoRA *(Algorithm 1) with* GD *updates (Equation 3 or 4) satisfy*

$$\mathbb{E}\left[f(W^T)\right] - f^\star \leq \left(1 - \gamma\mu\lambda_{\min}^H\right)^T \left(f(W^0) - f^\star\right).$$

We achieved a linear convergence rate, which is significantly better than previous results; however, this improvement applies to a more limited class of functions. Importantly, we can recover the classical results of GD by setting $\lambda_{\min}^H = 1$, which corresponds to the full-rank scenario.

The comprehensive analysis of different optimizers and their performance across various settings is provided in the appendix, as summarized in Table 1.

References

[1] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020.

[2] Kwangjun Ahn, Chulhee Yun, and Suvrit Sra. SGD with shuffling: Optimal rates without component convexity and large epoch requirements. *Advances in Neural Information Processing Systems*, 33:17526–17535, 2020.

[3] Dimitri P Bertsekas. Incremental proximal methods for large scale convex optimization. *Mathematical Programming*, 129(2):163–195, 2011.

[4] Dan Biderman, Jose Gonzalez Ortiz, Jacob Portes, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, et al. LoRA learns less and forgets less. *arXiv preprint arXiv:2405.09673*, 2024.

[5] Jaeyoung Cha, Jaewook Lee, and Chulhee Yun. Tighter lower bounds for shuffling SGD: Random permutations and beyond. In *International Conference on Machine Learning*, pages 3855–3912. PMLR, 2023.

[6] Zachary Charles and Jakub Konečný. On the outsized importance of learning rates in local update methods. *arXiv preprint arXiv:2007.00878*, 2020.

[7] Yae Jee Cho, Pranay Sharma, Gauri Joshi, Zheng Xu, Satyen Kale, and Tong Zhang. On the convergence of federated averaging with cyclic client participation. In *International Conference on Machine Learning*, pages 5677–5721. PMLR, 2023.

[8] Kenneth Ward Church, Zeyu Chen, and Yanjun Ma. Emerging trends: A gentle introduction to fine-tuning. *Natural Language Engineering*, 27(6):763–778, 2021.

[9] Laurent Condat and Peter Richtárik. Randprox: Primal-dual optimization algorithms with randomized proximal updates. *arXiv preprint arXiv:2207.12891*, 2022.

[10] Laurent Condat, Ivan Agarskỳ, Grigory Malinovsky, and Peter Richtárik. TAMUNA: Doubly accelerated federated learning with local training, compression, and partial participation. *arXiv preprint arXiv:2302.09832*, 2023.

[11] Marina Danilova, Pavel Dvurechensky, Alexander Gasnikov, Eduard Gorbunov, Sergey Guminov, Dmitry Kamzolov, and Innokentiy Shibaev. Recent theoretical advances in non-convex optimization. In *High-Dimensional Optimization and Probability: With a View Towards Data Science*, pages 79–163. Springer, 2022.

[12] Yury Demidovich, Grigory Malinovsky, Igor Sokolov, and Peter Richtárik. A guide through the zoo of biased SGD. *Advances in Neural Information Processing Systems*, 36:23158–23171, 2023.

[13] Margalit R Glasgow, Honglin Yuan, and Tengyu Ma. Sharp bounds for federated averaging (local SGD) and continuous perspective. In *International Conference on Artificial Intelligence and Statistics*, pages 9050–9090. PMLR, 2022.

[14] Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, pages 3556–3564. PMLR, 2021.

[15] Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. In *International Conference on Machine Learning*, pages 5200–5209. PMLR, 2019.

[16] Michał Grudzień, Grigory Malinovsky, and Peter Richtárik. Can 5th generation local training methods support client sampling? yes! In *International Conference on Artificial Intelligence and Statistics*, pages 1055–1092. PMLR, 2023.

[17] Michał Grudzień, Grigory Malinovsky, and Peter Richtárik. Improving accelerated federated learning with compression and importance sampling. *arXiv preprint arXiv:2306.03240*, 2023.

[18] Zeyu Han, Chao Gao, Jinyang Liu, Sai Qian Zhang, et al. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

[19] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.

[20] Samuel Horváth, Maziar Sanjabi, Lin Xiao, Peter Richtárik, and Michael Rabbat. FedShuffle: Recipes for better use of local work in federated learning. *arXiv preprint arXiv:2204.13169*, 2022.

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[22] Prateek Jain, Dheeraj Nagaraj, and Praneeth Netrapalli. SGD without replacement: Sharper rates for general smooth convex functions. *arXiv preprint arXiv:1903.01463*, 2019.

[23] 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.

[24] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[25] Ahmed Khaled and Peter Richtárik. Better theory for SGD in the nonconvex world. *Transactions on Machine Learning Research*, 2022.

[26] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. First analysis of local GD on heterogeneous data. *arXiv preprint arXiv:1909.04715*, 2019.

[27] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 4519–4529. PMLR, 2020.

[28] Ahmed Khaled, Othmane Sebbouh, Nicolas Loizou, Robert M Gower, and Peter Richtárik. Unified analysis of stochastic gradient methods for composite convex and smooth optimization. *Journal of Optimization Theory and Applications*, 199(2):499–540, 2023.

[29] Mikhail Khodak, Renbo Tu, Tian Li, Liam Li, Maria-Florina F Balcan, Virginia Smith, and Ameet Talwalkar. Federated hyperparameter tuning: Challenges, baselines, and connections to weight-sharing. *Advances in Neural Information Processing Systems*, 34:19184–19197, 2021.

[30] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

[31] Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[32] Weirui Kuang, Bingchen Qian, Zitao Li, Daoyuan Chen, Dawei Gao, Xuchen Pan, Yuexiang Xie, Yaliang Li, Bolin Ding, and Jingren Zhou. FederatedScope-LLM: A comprehensive package for fine-tuning large language models in federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5260–5271, 2024.

[33] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*, 2018.

[34] Yinhan Liu. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[35] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[36] Yucheng Lu, Si Yi Meng, and Christopher De Sa. A general analysis of example-selection for stochastic gradient descent. In *International Conference on Learning Representations (ICLR)*, volume 10, 2022.

[37] Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtárik. From local SGD to local fixed-point methods for federated learning. In *International Conference on Machine Learning*, pages 6692–6701. PMLR, 2020.

[38] Grigory Malinovsky and Peter Richtárik. Federated random reshuffling with compression and variance reduction. *arXiv preprint arXiv:2205.03914*, 2022.

[39] Grigory Malinovsky, Kai Yi, and Peter Richtárik. Variance reduced proxskip: Algorithm, theory and application to federated learning. *Advances in Neural Information Processing Systems*, 35:15176–15189, 2022.

[40] Grigory Malinovsky, Samuel Horváth, Konstantin Burlachenko, and Peter Richtárik. Federated learning with regularized client participation. *arXiv preprint arXiv:2302.03662*, 2023.

[41] Grigory Malinovsky, Konstantin Mishchenko, and Peter Richtárik. Server-side stepsizes and sampling without replacement provably help in federated optimization. In *Proceedings of the 4th International Workshop on Distributed Machine Learning*, pages 85–104, 2023.

[42] Grigory Malinovsky, Alibek Sailanbayev, and Peter Richtárik. Random reshuffling with variance reduction: New analysis and better rates. In *Uncertainty in Artificial Intelligence*, pages 1347–1357. PMLR, 2023.

[43] Artavazd Maranjyan, Mher Safaryan, and Peter Richtárik. Gradskip: Communication-accelerated local gradient methods with better computational complexity. *arXiv preprint arXiv:2210.16402*, 2022.

[44] Konstantin Mishchenko, Ahmed Khaled, and Peter Richtárik. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems*, 33: 17309–17320, 2020.

[45] Konstantin Mishchenko, Ahmed Khaled, and Peter Richtárik. Proximal and federated random reshuffling. In *International Conference on Machine Learning*, pages 15718–15749. PMLR, 2022.

[46] Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtárik. Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally! In *International Conference on Machine Learning*, pages 15750–15769. PMLR, 2022.

[47] Angelia Nedić and Dimitri Bertsekas. Convergence rate of incremental subgradient algorithms. *Stochastic optimization: algorithms and applications*, pages 223–264, 2001.

[48] Yu E Nesterov. Introductory lectures on convex optimization. a basic course. 2004.

[49] Lam M Nguyen, Quoc Tran-Dinh, Dzung T Phan, Phuong Ha Nguyen, and Marten Van Dijk. A unified convergence analysis for shuffling-type gradient methods. *Journal of Machine Learning Research*, 22(207):1–44, 2021.

[50] Benjamin Recht and Christopher Ré. Toward a noncommutative arithmetic-geometric mean inequality: Conjectures, case-studies, and consequences. In *Conference on Learning Theory*, pages 11–1. JMLR Workshop and Conference Proceedings, 2012.

[51] Peter Richtárik and Martin Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016.

[52] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[53] Abdurakhmon Sadiev, Dmitry Kovalev, and Peter Richtárik. Communication acceleration of local gradient methods via an accelerated primal-dual algorithm with an inexact prox. *Advances in Neural Information Processing Systems*, 35:21777–21791, 2022.

[54] Abdurakhmon Sadiev, Grigory Malinovsky, Eduard Gorbunov, Igor Sokolov, Ahmed Khaled, Konstantin Burlachenko, and Peter Richtárik. Federated optimization algorithms with random reshuffling and gradient compression. *arXiv preprint arXiv:2206.07021*, 2022.

[55] Itay Safran and Ohad Shamir. Random shuffling beats SGD only after many epochs on ill-conditioned problems. *Advances in Neural Information Processing Systems*, 34:15151–15161, 2021.

[56] A Shapiro and Y Wardi. Convergence analysis of gradient descent stochastic algorithms. *Journal of Optimization Theory and Applications*, 91:439–454, 1996.

[57] Ruo-Yu Sun. Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 8(2):249–294, 2020.

[58] Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in privacy-preserving federated learning. *arXiv preprint arXiv:2403.12313*, 2024.

[59] Zehua Sun, Huanqi Yang, Kai Liu, Zhimeng Yin, Zhenjiang Li, and Weitao Xu. Recent advances in LoRa: A comprehensive survey. *ACM Transactions on Sensor Networks*, 18(4): 1–44, 2022.

[60] Grega Vrbančič and Vili Podgorelec. Transfer learning with adaptive fine-tuning. *IEEE Access*, 8:196197–196211, 2020.

[61] Alex Wang. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[62] Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local SGD better than minibatch SGD? In *International Conference on Machine Learning*, pages 10334–10343. PMLR, 2020.

[63] Blake E Woodworth, Kumar Kshitij Patel, and Nati Srebro. Minibatch vs local sgd for heterogeneous distributed learning. *Advances in Neural Information Processing Systems*, 33: 6281–6292, 2020.

[64] Wenhan Xia, Chengwei Qin, and Elad Hazan. Chain of LoRA: Efficient fine-tuning of language models via residual learning. *arXiv preprint arXiv:2401.04151*, 2024.

[65] Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv preprint arXiv:2312.12148*, 2023.

[66] Chulhee Yun, Shashank Rajput, and Suvrit Sra. Minibatch vs local sgd with shuffling: Tight convergence bounds and beyond. *arXiv preprint arXiv:2110.10342*, 2021.

[67] Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. Asymmetry in low-rank adapters of foundation models. *arXiv preprint arXiv:2402.16842*, 2024.

Table 1: Summary of our theoretical convergence results for RAC-LoRA for solving Problem (1) when using a specific optimizer for approximately solving the subproblem in Step 4. The results for the RAC-LoRA + GD combination are described in Section **??**, while the proofs can be found in Appendix F. The results and proofs for all other combinations can be found in the indicated appendices.

Problem	Fine-tuner	Subproblem Optimizer	Non-convex	PL
(1)	RAC-LoRA	Gradient Descent (GD)	$\mathcal{O}\left(1/T\right)$ Sec. F.1	$\mathcal{O}\left(\exp(-T)\right)$ Sec. F.2
(1)+(6)	RAC-LoRA	Random Reshuffling (RR)	$\mathcal{O}\left(1/T^{\frac{2}{3}}\right)$ Sec. G.1	$\mathcal{O}(1/T^2)$ Sec. G.2
(1)	RAC-LoRA	Stochastic Gradient Descent (SGD)	$\mathcal{O}\left(1/T^{\frac{1}{2}}\right)$ Sec. H.1	$\mathcal{O}\left(1/T\right)$ Sec. H.2
(1)+(12)	Fed-RAC-LoRA	Random Reshuffling (RR)	$\mathcal{O}\left(1/T^{\frac{1}{2}}\right)$ Sec. I.1	$\mathcal{O}\left(1/T\right)$ Sec. I.2

Appendix A. Contributions

To address the aforementioned fundamental issues of LoRA-type heuristics, and to firmly ground the fine-tuning-via-low-rank adaptation line of work in a theoretically sound algorithmic framework, we propose a new generic low-rank adaptation framework for which we coin the name Randomized Asymmetric Chain of LoRA (RAC-LoRA); see Algorithm 1.

- Similarly to COLA [64], our method is iterative: we perform a chain of low-rank updates (see Step 2 in Algorithm 1). In each step of the chain, one matrix (e.g., A) is chosen randomly from a pre-defined distribution, and the other (e.g., B) is trainable (see Step 3 in Algorithm 1). Which of these two update matrices is chosen randomly and which one is trainable is decided a-priori, and hence our method is asymmetric in nature, similarly to AsymmLoRA [67]. We propose two options, depending on which matrix is trainable and which one is chosen randomly: in Option 1, A is trainable, and in Option 2, B is trainable.
- In order to make our framework flexible, we offer a variety of strategies for updating the trainable matrix in each step of the chain. This is possible since in each such step we formulate an auxiliary optimization subproblem in the trainable matrix, and one can thus choose essentially *any optimizer* for approximately solving it (see Step 4 in Algorithm 1). We theoretically analyze several such optimizers within our RAC-LoRA framework, including Gradient Descent (GD) in Appendix F (however, we include and describe the theorems in Section 3.2), Random Reshuffling RR in Appendix G, and Stochastic Gradient Descent (SGD) in Appendix H. In the case of GD and SGD, just a single step of the optimizer is sufficient, and this is what our analysis accounts for. In the case of RR, we apply a single pass over the data in a randomly reshuffled order. See Table 1 for a quick overview. Our analysis applies to the smooth nonconvex regime, in which we prove fast sublinear (i.e., $\mathcal{O}(1/\sqrt{T})$, $\mathcal{O}(1/T)$ or $\mathcal{O}(1/T^2)$) convergence rates to a stationary point, and fast linear (i.e., $\mathcal{O}(\exp(-T))$) rates to the globally optimal solution under the Polyak-Łojasiewicz (PL) condition.

- The update is applied (see Step 5 in Algorithm 1), and the method moves on to the next step of the chain.

Experiments. We apply our method to several machine learning tasks. We start from convex problems with traditional models, such as logistic and linear regression, to provide clear illustrations of our theoretical findings. In addition, we present empirical analyses for multilayer perception (MLP) on MNIST and RoBERTa on the GLUE benchmark tasks [61]. See Appendix B.

Federated Learning. Furthermore, we extend our findings from the simple unstructured problem (1) to the more challenging distributed/federated problem where f has the special form described in (12); there we consider solving a distributed optimization problem via our new Fed-RAC-LoRA method (Algorithm 2). These additional results can be found in Section I. For illustrative purposes, we provide an analysis for RR as the optimizer for the subproblem; see also Table 1. Previous research [58] has shown that using a single learnable matrix in this context provides several key advantages, particularly in terms of preserving privacy, ensuring the correctness of model aggregation, and maintaining stability when adjusting the scaling factor [58]. These benefits are crucial in Federated Learning [31], where data is distributed across multiple clients, and privacy constraints must be upheld while performing model updates. Building on this asymmetric approach, we integrate the concept of chained updates to develop Fed-RAC-LoRA, a more robust and scalable distributed method. Our approach maintains the computational efficiency of the original RAC-LoRA while ensuring rigorous convergence properties in the distributed setting, offering a theoretically sound method for large-scale Federated Learning scenarios.

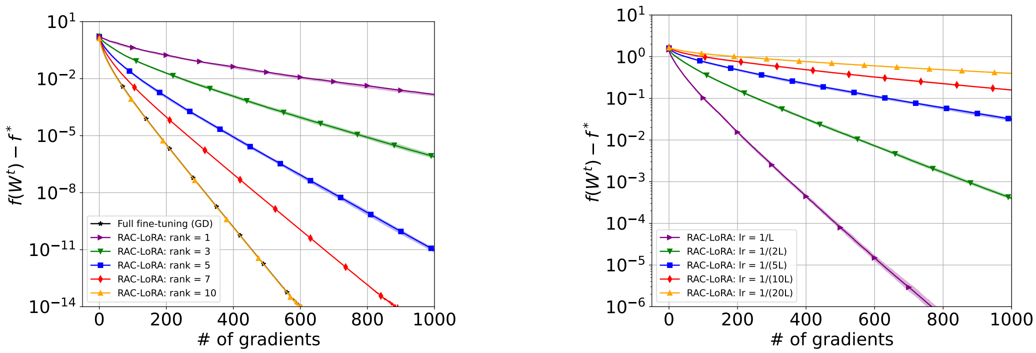

Figure 2: RAC-LoRA convergence with varying ranks and step sizes on a linear regression problem.

Appendix B. Experiments

In this section, we explore the performance of RAC-LoRA as an optimization algorithm in machine learning applications. In Appendix B.1 we validate the theoretical results in convex problems, while in Appendix B.2 we evaluate the method applied to neural networks.

B.1. Convex Optimization Problems

Linear Regression. We conducted our analysis in a controlled setting using linear regression with quadratic regularization and synthetically generated data. Specifically, we utilized 3,000 samples for pre-training the model and 1,000 samples for fine-tuning. In this setup, we have $d = 100$ with weight matrices of size 10×10, and the regularization term is set to 0.0001. As illustrated in Figure 2, the method converges for various ranks and the convergence speed is proportional to $\frac{n}{r}$, and when the rank is set to the full rank, we observe convergence identical to that of FPFT. We remark that COLA would suffer from the same divergence behavior as in Figure 1 on this quadratic problem.

Logistic Regression. Analogous results for logistic regression are shown in Appendix D.

B.2. Non-Convex Optimization Problems

Further experimental results are provided in Appendix E.

B.2.1. RESULTS OF ROBERTA ON NLP TASKS

As in prior work [64, 67], we evaluate low-rank adaptation methods for LLMs using the GLUE dataset [61].

Methodology. We fine-tuned the `roberta-base` model [34] on four of the smallest GLUE tasks to study the behavior of low-rank methods in practical scenarios. For the chained methods, we use a range of values for the number of chains and epochs per chain hyperparameters. In each experiment we used rank 2 for the adaptations and trained using the AdamW optimiser [35] with β parameters 0.9 and 0.999, $\epsilon = 1 \times 10^{-8}$, a learning rate of 4×10^{-4} with linear schedule and a training batch size 8.

Discussion. The results are presented in Table 2. We find that RAC-LoRA performs competitively with other low-rank adaptation methods, but does not outperform Asymmetric LoRA despite having greater capacity. We expect RAC-LoRA to outperform Asymmetric LoRA in settings where there is a benefit to the additional capacity, i.e., those where a full parameter fine tune (FPFT) is much better than Asymmetric LoRA. The performance of the FPFT in Table 2 shows that the selected GLUE tasks do not provide such a setting. Here, a single low-rank adaptation is already enough to obtain performance close to that of FPFT. However, this intuition motivates the experiments in **??** B.2.2 where we intentionally restrict capacity of the adaptations to isolate the effect of the chaining procedure.

Method	# Chains	# Epochs	MRPC	CoLA	RTE	STS-B	Avg
FPFT *	1	30, 80, 80, 40	$90.2_{\pm 0.0}$	$63.6_{\pm 0.0}$	$78.7_{\pm 0.0}$	$91.2_{\pm 0.0}$	80.9
LoRA *			$89.7_{\pm 0.7}$	$63.4_{\pm 1.2}$	$86.6_{\pm 0.7}$	$91.5_{\pm 0.2}$	82.8
LoRA	1	100	$87.7_{\pm 0.2}$	$60.8_{\pm 0.2}$	$75.2_{\pm 1.5}$	$90.2_{\pm 0.1}$	78.5
AsymmLoRA			$86.9_{\pm 0.3}$	$58.7_{\pm 1.0}$	$71.0_{\pm 3.3}$	$90.4_{\pm 0.0}$	76.8
COLA	10	10	$88.0_{\pm 0.8}$	$59.5_{\pm 1.0}$	$72.1_{\pm 0.9}$	$90.7_{\pm 0.2}$	77.6
RAC-LoRA	10	10	$87.0_{\pm 0.7}$	$58.5_{\pm 0.1}$	$72.3_{\pm 1.5}$	$90.3_{\pm 0.0}$	77.0

Table 2: Results with RoBERTa-base for rank 2 on tasks from the GLUE benchmark. *: results taken from the work of [21]. We report Matthews correlation coefficient for COLA, Pearson correlation coefficient for STS-B, and accuracy for the remaining tasks. Results are averaged over 3 seeds and standard deviations are given in the subscript.

B.2.2. RESULTS OF MLPS ON MNIST

In this section, we seek to isolate the effect of the chaining procedure on generalisation performance by restricting the capacity of the low-rank adaptations. This ensures that a single adaptation is not sufficient to reach performance comparable with FPFT, allowing us to explore how chaining adaptations can bridge this gap.

Methodology. We first pre-train a 3-layer MLP on the first five classes (digits 0-4) and then adapt the network using LoRA-based methods for recognizing the remaining five unseen classes (digits 5-9). The model is evaluated solely on these unseen classes[2]. we used rank 1 for the adaptations and trained using the AdamW optimiser [35] with β parameters 0.9 and 0.999, $\epsilon = 1 \times 10^{-8}$, a constant learning rate of 2×10^{-4} and a training batch size 128.

Discussion. Table 3 shows results for MNIST with different ranks and initialization. LoRA reaches around 90% of the accuracy of FPFT leaving some margin for improvement when using the chains. COLA constructs a sequence of LoRA modules, delivering significant accuracy improvements over LoRA due to the chaining procedure. The chaining allows COLA to capture richer features (at the cost of training more parameters). However, both LoRA and COLA lack rigorous convergence guarantees. AsymmLoRA has been shown empirically to approximate the performance of LoRA [58] — but again no convergence result is provided. Our proposed method (RAC-LoRA) enjoys significant accuracy improvements over AsymmLoRA, again due to the chaining procedure. RAC-LoRA leverages a diverse learning process across different LoRA blocks, which intuitively

2. The setup is inspired by https://github.com/sunildkumar/lora_from_scratch/.

allows the model to capture a broader range of features. Crucially, RAC-LoRA comes with convergence guarantees (Theorems 7 and 9). Finally, we note that each iteration of RAC-LoRA requires training only one matrix per LoRA block, while COLA needs training two matrices. This reduction in trainable parameters may offer advantages in resource-constrained settings, such as Federated Learning, where minimizing communication costs is critical.

Appendix C. Conclusion

In this work, we introduced RAC-LoRA, a framework for parameter-efficient fine-tuning that enables interpolation between low-rank adaptation and full parameter fine-tuning. Motivated by the convergence challenges of LoRA, we propose the iterative algorithm RAC-LoRA and provide convergence guarantees across various settings, including gradient descent, stochastic gradient descent, and random reshuffling. We extended this framework to the federated learning setup, where RAC-LoRA has advantages over competing algorithms in terms of communication efficiency. Finally, we validate our theoretical results empirically in both convex problems, such as linear and logistic regression, and non-convex problems, such as MLPs and LLMs, finding that its chaining procedure is advantageous in settings where standard low-rank adaptation approaches (such as LoRA and AsymmLoRA) fail to capture the richness of full-parameter fine-tuning.

Table 3: MLP results on MNIST with rank r and α set to 1. In the case of AsymmLoRA and RAC-LoRA, only the zero-initialized matrix is trained.

Method	$\mathcal{D}_A$	$\mathcal{D}_B$	**Acc**	**Train Params**
FPFT	-	-	98.0	54,700
LoRA	Gaussian	Zero	83.8	1K
COLA	Gaussian	Zero	92.6	1K
LoRA	Zero	Gaussian	87.0	1K
COLA	Zero	Gaussian	96.2	1K
AsymmLoRA	Gaussian	Zero	62.3	133
RAC-LoRA	Gaussian	Zero	92.0	133
AsymmLoRA	Zero	Gaussian	81.6	912
RAC-LoRA	Zero	Gaussian	96.1	912

Appendix D. Results on Convex Optimization Problems

D.1. Logistic Regression

We performed our analysis in a controlled environment using logistic regression with quadratic regularization on synthetic data. In this configuration, we set $d = 100$, employed weight matrices of size 10×10, and used 2,000 samples, with the regularization term fixed at 0.1. As shown in Figure 3, the method demonstrates convergence across different ranks, and when the rank is set to full rank, we observe convergence that mirrors that of FPFT.

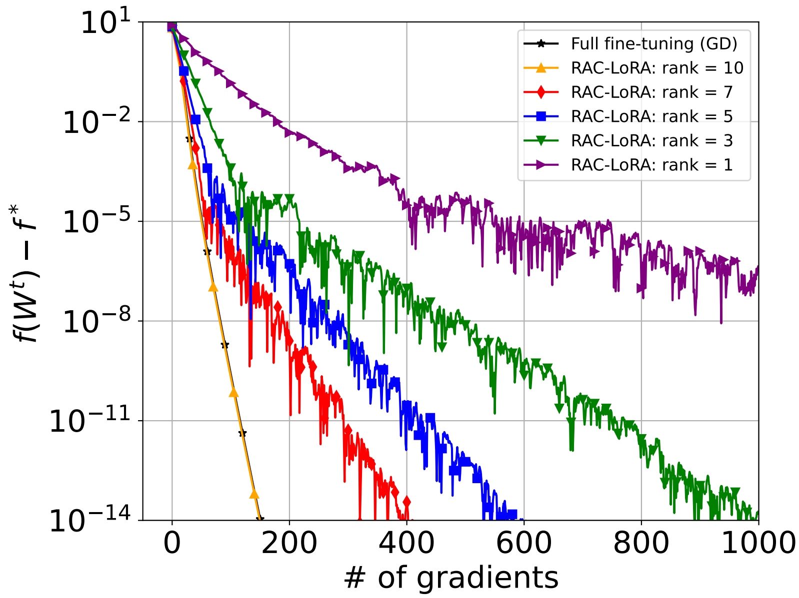

Figure 3: RAC-LoRA convergence with varying ranks and step sizes on a logistic regression problem.

Appendix E. Results on Non-Convex Optimization Problems

E.1. Additional Results of RoBERTa on NLP Tasks

Table 4 reports additional configurations of the number of epochs per chain and the number of chains on the GLUE benchmark. These results further corroborate the discussion in Section B.2.

E.2. Ablation on number of epochs per block in the chains

Convergence proof for RAC-LoRA (Corollary 12 and Corollary 14) states that each LoRA module shall be optimized for one epoch only. However, good approximations can also be obtained using more epochs per block and hence fewer blocks (i.e., fewer parameters), as we show in Table 5 for the case of MLP on MNIST.

Similarly, we plot the training loss curves for RoBERTa-base on the RTE dataset in Figure 4. We observe that all setups reach the same value at convergence with similar speed.

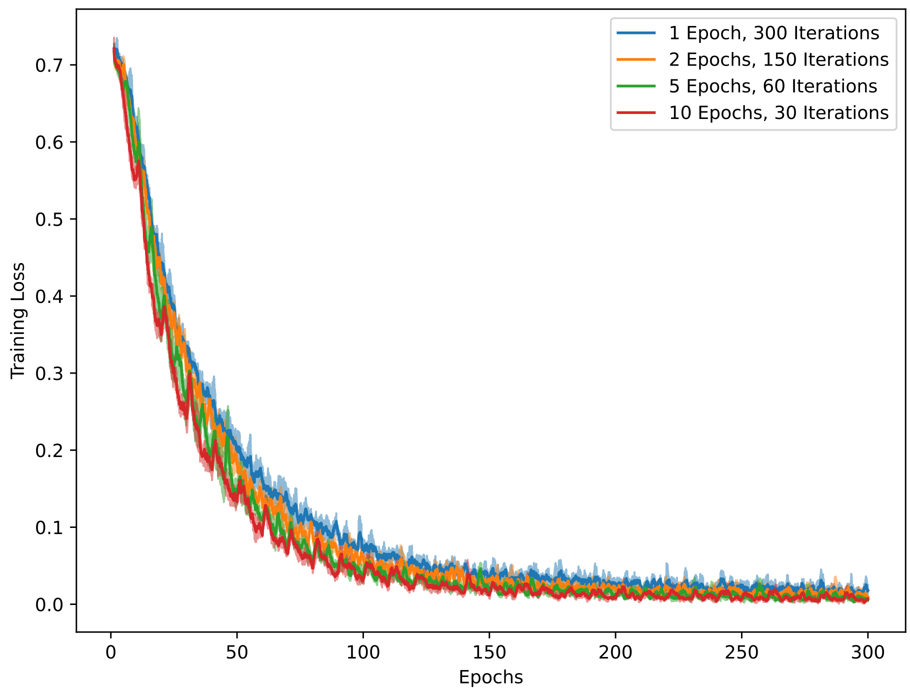

Figure 4: RAC-LoRA training loss curves at a fixed computational budget for varying epochs for each block in the chain. RoBERTa-base with rank 2.

Method	# Chains	# Epochs	MRPC	CoLA	RTE	STS-B	Avg
FPFT *	1	30, 80, 80, 40	$90.2_{\pm 0.0}$	$63.6_{\pm 0.0}$	$78.7_{\pm 0.0}$	$91.2_{\pm 0.0}$	80.9
LoRA *			$89.7_{\pm 0.7}$	$63.4_{\pm 1.2}$	$86.6_{\pm 0.7}$	$91.5_{\pm 0.2}$	82.8
LoRA	1	20	$86.8_{\pm 0.8}$	$58.0_{\pm 0.4}$	$71.4_{\pm 0.7}$	$90.3_{\pm 0.1}$	76.6
AsymmLoRA			$85.5_{\pm 0.5}$	$56.5_{\pm 1.5}$	$69.2_{\pm 0.2}$	$89.6_{\pm 0.1}$	75.2
COLA	2	10	$87.1_{\pm 0.2}$	$58.4_{\pm 1.5}$	$69.9_{\pm 0.9}$	$90.3_{\pm 0.2}$	76.4
	10	2	$84.2_{\pm 1.1}$	$54.2_{\pm 0.4}$	$64.6_{\pm 1.3}$	$89.1_{\pm 0.1}$	73.0
RAC-LoRA	2	10	$85.6_{\pm 1.7}$	$55.3_{\pm 1.2}$	$68.6_{\pm 1.0}$	$89.4_{\pm 0.2}$	74.7
	10	2	$85.4_{\pm 0.4}$	$55.1_{\pm 1.2}$	$65.5_{\pm 0.9}$	$89.3_{\pm 0.1}$	73.8
LoRA	1	50	$88.2_{\pm 0.3}$	$60.1_{\pm 0.4}$	$74.4_{\pm 0.9}$	$90.6_{\pm 0.1}$	78.3
AsymmLoRA			$86.4_{\pm 1.0}$	$57.4_{\pm 0.3}$	$69.9_{\pm 1.8}$	$90.3_{\pm 0.1}$	76.0
COLA	5	10	$87.8_{\pm 1.1}$	$59.3_{\pm 2.1}$	$71.2_{\pm 1.2}$	$90.6_{\pm 0.2}$	77.2
	10	5	$87.7_{\pm 0.5}$	$58.1_{\pm 1.2}$	$70.9_{\pm 0.5}$	$90.2_{\pm 0.2}$	76.7
RAC-LoRA	5	10	$87.2_{\pm 0.6}$	$57.6_{\pm 0.5}$	$70.6_{\pm 0.7}$	$90.2_{\pm 0.1}$	76.4
	10	5	$87.5_{\pm 0.4}$	$57.8_{\pm 1.0}$	$70.3_{\pm 1.2}$	$90.2_{\pm 0.2}$	76.5
LoRA	1	100	$87.7_{\pm 0.2}$	$60.8_{\pm 0.2}$	$75.2_{\pm 1.5}$	$90.2_{\pm 0.1}$	78.5
AsymmLoRA			$86.9_{\pm 0.3}$	$58.7_{\pm 1.0}$	$71.0_{\pm 3.3}$	$90.4_{\pm 0.0}$	76.8
COLA	10	10	$88.0_{\pm 0.8}$	$59.5_{\pm 1.0}$	$72.1_{\pm 0.9}$	$90.7_{\pm 0.2}$	77.6
RAC-LoRA	10	10	$87.0_{\pm 0.7}$	$58.5_{\pm 0.1}$	$72.3_{\pm 1.5}$	$90.3_{\pm 0.0}$	77.0

Table 4: Performance of the methods using RoBERTa-base for rank 2. The experiments are based on 4 tasks from the GLUE benchmark. * denotes the results reported in [21]. We report Matthews correlation coefficient for the CoLA dataset, Pearson correlation coefficient for STS-B, and accuracy for the remaining tasks, with the standard deviations given in the subscript. The results are obtained using 3 random seeds.

	Number of epochs per block					
	1	**2**	**3**	**4**	**5**	**10**
COLA	96.2	95.8	95.9	95.1	95.4	94.5
RAC-LoRA	96.1	95.6	95.6	94.9	94.7	93.9

Table 5: Accuracy at varying epochs for each block in the chained methods (COLA and RAC-LoRA). The setup is the same as in Table 3, with a zero-initialized A matrix and a Gaussian-initialized B matrix. To ensure a fair comparison, the product of the number of epochs per block and the number of blocks is kept constant at 50. The number of trainable parameters for COLA and RAC-LoRA are 1K and 912, respectively.

Appendix F. Analysis of RAC-LoRA with Gradient Descent

F.1. Proof of Theorem 7

The proof is provided for Left Sketch (Definition 2). The result for Right Sketch (Definition 3) can be derived by following the same steps.

Proof We begin by examining the implications of Assumption 1. The relationships between various conditions associated with Assumption 1 are discussed in detail in Nesterov [48].

$$f(W^{t+1}) \leq f(W^t) + \left\langle \nabla f(W^t), W^{t+1} - W^t \right\rangle + \frac{L}{2} \left\| W^{t+1} - W^t \right\|^2$$

Using the update rule $W^{t+1} = W^t - \gamma H_B^t \nabla f(W^t)$ we get

$$\begin{aligned} f(W^{t+1}) \leq & f(W^t) + \left\langle \nabla f(W^t), -\gamma H_B^t \nabla f(W^t) \right\rangle + \frac{L}{2} \left\| -\gamma H_B^t \nabla f(W^t) \right\|^2 \\ \leq & f(W^t) - \gamma \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle + \frac{L}{2}\gamma^2 \left\| H_B^t \nabla f(W^t) \right\|^2 \\ \leq & f(W^t) - \gamma \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle + \frac{L}{2}\gamma^2 \left\langle H_B^t \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle \\ \leq & f(W^t) - \gamma \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle + \frac{L}{2}\gamma^2 \left\langle \nabla f(W^t), (H_B^t)^\top H_B^t \nabla f(W^t) \right\rangle. \end{aligned}$$

Since matrix H_B^t is projection matrix, we have $(H_B^t)^\top H_B^t = (H_B^t)^2 = H_B^t$:

$$f(W^{t+1}) \leq f(W^t) - \gamma \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle + \frac{L}{2}\gamma^2 \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle.$$

Using the fact that $\gamma \leq \frac{1}{L}$ we have

$$f(W^{t+1}) \leq f(W^t) - \frac{\gamma}{2} \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle.$$

Taking expectation we get

$$\begin{aligned} \mathbb{E}\left[f(W^{t+1}) \mid W^t \right] \leq & \mathbb{E}\left[f(W^t) - \frac{\gamma}{2} \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle \mid W^t \right] \\ \leq & f(W^t) - \frac{\gamma}{2} \left\langle \nabla f(W^t), \mathbb{E}\left[H_B^t \right] \nabla f(W^t) \right\rangle \end{aligned}$$

Using an Assumption 4 we have

$$\begin{aligned} \mathbb{E}\left[f(W^{t+1}) \mid W^t \right] \leq & \mathbb{E}\left[f(W^t) - \frac{\gamma}{2} \left\langle \nabla f(W^t), H_B^t \nabla f(W^t) \right\rangle \mid W^t \right] \\ \leq & f(W^t) - \frac{\gamma}{2} \lambda_{\min}^{H_B} \left\| \nabla f(W^t) \right\|^2. \end{aligned}$$

Subtracting $f^\star$ from both sides we get

$$\mathbb{E}\left[f(W^{t+1}) \mid W^t \right] - f^\star \leq f(W^t) - f^\star - \frac{\gamma}{2} \lambda_{\min}^{H_B} \left\| \nabla f(W^t) \right\|^2. \tag{5}$$

Now we can rewrite as

$$\frac{\gamma}{2} \lambda_{\min}^{H_B} \left\| \nabla f(W^t) \right\|^2 \leq \left(f(W^t) - f^\star \right) - \left(\mathbb{E}\left[f(W^{t+1}) \mid W^t \right] - f^\star \right)$$

Taking expectation and using tower property we obtain

$$\frac{\gamma}{2}\lambda_{\min}^{H_B}\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq e^t - e^{t+1},$$

where $e^t = \mathbb{E}\left[f(W^t)\right] - f^\star$. Now we can sum these inequalities together and get

$$\sum_{t=0}^{T-1}\frac{\gamma}{2}\lambda_{\min}^{H_B}\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq \sum_{t=0}^{T-1}\left(e^t - e^{t+1}\right),$$

Using telescoping property of $e^t - e^{t+1}$ we get

$$\sum_{t=0}^{T-1}\frac{\gamma}{2}\lambda_{\min}^{H_B}\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq e^0 - e^T.$$

Once we divide by T we obtain

$$\frac{1}{T}\sum_{t=0}^{T-1}\frac{\gamma}{2}\lambda_{\min}^{H_B}\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq \frac{e^0 - e^T}{T} \leq \frac{e^0}{T}.$$

Finally, we get

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq \frac{2(f(W^0) - f^\star)}{\lambda_{\min}^{H_B}\gamma T}.$$

Applying argument from Danilova et al. [11] we obtain the result for uniformly chosen point.

■

F.2. Proof of Theorem 9

The proof is provided for Left Sketch (Definition 2). The result for Right Sketch (Definition 3) can be derived by following the same steps.

Proof

We start from the inequality 5:

$$\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq f(W^t) - f^\star - \frac{\gamma}{2}\lambda_{\min}^{H_B}\left\|\nabla f(W^t)\right\|^2.$$

Using PL condition $\left\|\nabla f(W^t)\right\|^2 \geq 2\mu\left(f(W^t) - f^\star\right)$ we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star &\leq f(W^t) - f^\star - \gamma\mu\lambda_{\min}^{H_B}\left(f(W^t) - f^\star\right)\\ &\leq \left(1 - \gamma\mu\lambda_{\min}^{H_B}\right)\left(f(W^t) - f^\star\right).\end{aligned}$$

Once we unroll the recursion we get

$$\mathbb{E}\left[f(W^T)\right] - f^\star \leq \left(1 - \gamma\mu\lambda_{\min}^{H_B}\right)^T\left(f(W^0) - f^\star\right).$$

In order to obtain ε solution we need to take

$$T \geq \mathcal{O}\left(\frac{L}{\mu}\frac{1}{\lambda_{\min}^{H_B}}\log\frac{1}{\varepsilon}\right).$$

■

Appendix G. Analysis of RAC-LoRA with Random Reshuffling

The previous results were obtained using full gradients. However, this approach is impractical in deep learning settings, where calculating full gradients is often infeasible. To analyze stochastic methods, we consider problem (1), where f has the following special structure:

$$f(W^0 + \Delta W) := \frac{1}{N}\sum_{i=1}^{N} f_i(W^0 + \Delta W), \tag{6}$$

where each function f_i represents the individual loss function for one sample and N is total number of datapoints. Next, we analyze a practical variant of stochastic gradient descent (SGD) known as Random Reshuffling (RR), which involves sampling without replacement. In this method, the dataset is shuffled according to a permutation, ensuring that each training sample is used exactly once during each epoch.

RR is a variant of SGD in which each data point is used exactly once per epoch, also known as SGD with sampling without replacement. Many efforts have been made to explain why gradient methods with reshuffling perform so well in practice, across different types of problems. The convergence rates for incremental gradient methods with random reshuffling in convex optimization were first explored by Nedić and Bertsekas [47] and later by Bertsekas [3]. In recent years, a lot of focus has shifted toward strongly convex problems, with studies showing that RR can outperform SGD. For example, Recht and Ré [50] were among the first to analyze this for quadratic least squares problems.

Researchers have also managed to improve results and remove some of the earlier assumptions, such as second-order smoothness, as seen in works by Jain et al. [22], Safran and Shamir [55] and Mishchenko et al. [44]. These studies introduced a new way to account for the random permutation's variance, making it easier to analyze both convex and strongly convex cases. There have even been extensions into non-convex settings, with results under the PL condition [2, 49] and general non-convex smooth cases [36, 42, 44]. More recently, tighter lower bounds for strongly convex and PL functions have been developed [5].

In recent years, there's also been growing interest in applying these reshuffling techniques to distributed and federated learning, which is crucial for training large-scale, decentralized models [7, 20, 38, 40, 41, 45, 54, 66]

To analyze stochastic methods, we need to make assumptions about the variance. The standard assumption is that the variance is bounded:

Assumption 10 *There exist nonnegative constants $\sigma \geq 0$ such that for any $W^t \in \mathbb{R}^{m\times n}$ we have,*

$$\frac{1}{N}\sum_{i=1}^{n}\left\|\nabla f_i\left(W^t\right) - \nabla f\left(W^t\right)\right\|^2 \leq \sigma^2.$$

The proof is provided for Left Sketch (Definition 2). The result for Right Sketch (Definition 3) can be derived by following the same steps.

We consider a method belonging to the class of data permutation methods which is the RR algorithm. In each epoch t of RR, we sample indices $\pi_0, \pi_1, \ldots, \pi_{N-1}$ without replacement from $\{1, 2, \ldots, N\}$, i.e., $\{\pi_0, \pi_1, \ldots, \pi_{N-1}\}$ is a random permutation of the set $\{1, 2 \ldots N\}$ and proceed with N iterates of the form:

$$W^t_{i+1} = W^t_i - \gamma H^t_B \nabla f(W^t_i).$$

We then set $W^{t+1} = W_N^t$, and repeat the process for a total of T LoRA blocks. We can derive the effective step:

$$W^{t+1} = W^t - \gamma H_B^t \sum_{i=0}^{N-1} \nabla f(W_i^t) = W^t - \gamma H_B^t N \hat{g}^t, \tag{7}$$

where $\hat{g}^t = \frac{1}{N}\sum_{i=0}^{N-1} \nabla f(W_i^t)$.

G.1. Analysis of general non-convex setting

Theorem 11 *Suppose that Assumption 1 and Assumption 4 hold. Suppose that a stepsize $\gamma > 0$ is chosen such that $\gamma \leq \frac{1}{2LN}$. We choose the output of the method $\widetilde{W}^T$ uniformly at random from $W^0, W^1, \ldots, W^{T-1}$ Then, the iterate $\widetilde{W}^T$ of RAC-LoRA method (Algorithm 1) with RR updates (Equation 7) satisfy*

$$\mathbb{E}\left[\left\|\nabla f(\widetilde{W}^T)\right\|^2\right] \leq \frac{2}{\gamma N T} \frac{f(W^0) - f^\star}{\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - \frac{1}{4}\lambda_{\max}^H\right)} + \frac{L^2\gamma^2\lambda_{\max}^H N\sigma^2}{\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - \frac{1}{4}\lambda_{\max}^H\right)}.$$

Remark: Notice that if we choose $\gamma = \mathcal{O}(1/T)$, the above result yields the rate $\mathcal{O}(1/T^2)$.

Proof In this context, and in subsequent discussions, the notation $\|\cdot\|$ refers to the Frobenius norm, while $\langle\cdot\rangle$ denotes the inner product associated with the Frobenius norm.

Now we can apply the L-smoothness:

$$\begin{aligned}
f(W^{t+1}) &\leq f(W^t) + \left\langle \nabla f(W^t), W^{t+1} - W^t \right\rangle + \frac{L}{2}\left\|W^{t+1} - W^t\right\|^2 \\
&= f(W^t) + \left\langle \nabla f(W^t), -\gamma H_B^t N \hat{g}^t \right\rangle + \frac{L}{2}\left\|\gamma H_B^t N \hat{g}^t\right\|^2 \\
&= f(W^t) - \gamma N \left\langle \nabla f(W^t), H_B^t \hat{g}^t \right\rangle + \frac{L}{2}\gamma^2 N^2 \left\|H_B^t \hat{g}^t\right\|^2 \\
&= f(W^t) - \frac{\gamma N}{2}\left(\left\|\nabla f(W^t)\right\|^2 + \left\|H_B^t \hat{g}^t\right\|^2 - \left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2\right) + \frac{L}{2}\gamma^2 N^2 \left\|H_B^t \hat{g}^t\right\|^2 \\
&= f(W^t) - \frac{\gamma N}{2}\left(\left\|\nabla f(W^t)\right\|^2 + \left\|H_B^t \hat{g}^t\right\|^2 - \left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2\right) + \frac{L}{2}\gamma^2 N^2 \left\|H_B^t \hat{g}^t\right\|^2 \\
&= f(W^t) - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2 - \frac{\gamma N}{2}\left\|H_B^t \hat{g}^t\right\|^2 (1 - \gamma L N) + \frac{\gamma N}{2}\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2.
\end{aligned}$$

Using $\gamma \leq \frac{1}{LN}$ we get

$$f(W^{t+1}) \leq f(W^t) - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\gamma N}{2}\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2.$$

Let us take expectation and subtract $f^\star$:

$$\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq f(W^t) - f^\star - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\gamma N}{2}\mathbb{E}\left[\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2 \mid W^t\right].$$

Let us consider the last term:

$$\begin{aligned}
&\mathbb{E}\left[\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2 \mid W^t\right] \\
&\quad = \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W^t) - H_B^t\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W_i^t)\right\|^2 \mid W^t\right] \\
&\quad = \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W^t) + H_B^t\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W^t) - H_B^t\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W^t) - H_B^t\frac{1}{N}\sum_{i=0}^{N-1}\nabla f_{\pi_i}(W_i^t)\right\|^2 \mid W^t\right]
\end{aligned}$$

Since $I - H_B^t$ and H_B^t are projection matrices generating perpendicular subspaces we have

$$\begin{aligned}
&\mathbb{E}\left[\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2 \mid W^t\right] \\
&\quad = \mathbb{E}\left[\left\|\left(I - H_B^t\right)\nabla f(W^t)\right\|^2 + \left\|H_B^t\frac{1}{n}\sum_{i=0}^{n-1}\left(\nabla f_{\pi_i}(W^t) - f_{\pi_i}(W_i^t)\right)\right\|^2 \mid W^t\right] \\
&\quad = \mathbb{E}\left[\left\langle\left(I - H_B^t\right)\nabla f(W^t), \left(I - H_B^t\right)\nabla f(W^t)\right\rangle + \left\|H_B^t\frac{1}{N}\sum_{i=0}^{N-1}\left(\nabla f_{\pi_i}(W^t) - f_{\pi_i}(W_i^t)\right)\right\|^2 \mid W^t\right].
\end{aligned}$$

Using the property that H_B^t and $I - H_B^t$ are projection matrices we obtain

$$\begin{aligned}
&\mathbb{E}\left[\left\|\nabla f(W^t) - H^t \hat{g}^t\right\|^2 \mid W^t\right] \\
&\quad \leq \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right]\left\|\nabla f(W^t)\right\|^2 + \mathbb{E}\left[\lambda_{\max}[H^t]L^2\frac{1}{N}\sum_{i=0}^{N-1}\left\|W^t - W_i^t\right\|^2\right].
\end{aligned}$$

Since $\lambda_{\max}[H^t] = 1$ for projections matrix we get

$$\mathbb{E}\left[\left\|\nabla f(W^t) - H_B^t \hat{g}^t\right\|^2 \mid W^t\right] \leq \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\left\|\nabla f(W^t)\right\|^2 + L^2\frac{1}{N}\sum_{i=0}^{N-1}\mathbb{E}\left[\left\|W^t - W_i^t\right\|^2 \mid W^t\right].$$

Now let us consider the last term:

$$\begin{aligned}
\mathbb{E}\left[\left\|W^t - W_k^t\right\|^2\right] &= \gamma^2\mathbb{E}\left[\left\|\sum_{i=0}^{k-1}H_B^t\nabla f_{\pi_i}(W_i^t)\right\|^2 \mid W^t\right] \\
&= \gamma^2\mathbb{E}\left[\left\|\sum_{i=0}^{k-1}H_B^t\left(\nabla f_{\pi_i}(W_i^t) - \nabla f_{\pi_i}(W^t)\right) + \sum_{i=0}^{k-1}H_B^t\nabla f_{\pi_i}(W^t)\right\|^2 \mid W^t\right] \\
&\leq 2\gamma^2 k\mathbb{E}\left[\sum_{i=0}^{k-1}\left(\left\|H_B^t\left(\nabla f_{\pi_i}(W_i^t) - \nabla f_{\pi_i}(W^t)\right)\right\|^2 + 2\gamma^2k^2\left\|H_B^t\nabla f_{\pi_i}(W^t)\right\|^2\right) \mid W^t\right] \\
&\leq 2\gamma^2 k\mathbb{E}\left[\sum_{i=0}^{k-1}\left(\lambda_{\max}\left[H^t\right]\left\|W_i^t - W^t\right\|^2 + 2\gamma^2k^2\left\|H_B^t\nabla f_{\pi_i}(W^t)\right\|^2\right) \mid W^t\right] \\
&\leq 2\gamma^2 k\mathbb{E}\left[\sum_{i=0}^{k-1}\left(\left\|W_i^t - W^t\right\|^2 + 2\gamma^2k^2\lambda_{\max}\left[\mathbb{E}\left[H_B^t\right]\right]\left\|\nabla f_{\pi_i}(W^t)\right\|^2\right) \mid W^t\right].
\end{aligned}$$

Now, we are ready to sum the inequalities. By using $\lambda_{\max}\left[\mathbb{E}\left[H^t\right]\right] = \lambda_{\max}^{H_B}$ and applying Lemma 1 from Mishchenko et al. [44] with Assumption 10, we obtain:

$$\begin{aligned}\sum_{i=0}^{n-1}\mathbb{E}\left[\left\|W^t - W_i^t\right\|^2\right] \leq & \mathbb{E}\left[\sum_{i=0}^{N-1}\left(2\gamma^2 k\sum_{i=0}^{k-1}\left\|W_i^t - W^t\right\|^2 + 2\gamma^2k^2\lambda_{\max}^{H_B}\left\|\nabla f_{\pi_i}(W^t)\right\|^2\right) \mid W^t\right]\\ \leq & \gamma^2L^2N(N-1)\sum_{i=0}^{N-1}\mathbb{E}\left[\left\|W^t - W_k^t\right\|^2\right]\\ & + \frac{1}{3}\gamma^2(N-1)N(2N-1)\lambda_{\max}^{H_B}\left\|\nabla f(W^t)\right\|^2 + \frac{1}{3}\lambda_{\max}^{H_B}\gamma^2N(N+1)\sigma^2.\end{aligned}$$

Using $\gamma \leq \frac{1}{2LN}$ we get

$$\begin{aligned}\sum_{i=0}^{n-1}\mathbb{E}\left[\left\|W^t - W_i^t\right\|^2\right] & \leq \frac{4}{3}\left(1-\gamma^2L^2N(N-1)\right)\sum_{i=0}^{N-1}\mathbb{E}\left[\left\|W^t - W_i^t\right\|^2\right]\\ & \leq \frac{4}{3}\left(\frac{1}{3}\gamma^2(N-1)N(2N-1)\lambda_{\max}^{H_B}\left\|\nabla f(W^t)\right\|^2 + \frac{1}{3}\lambda_{\max}^{H_B}\gamma^2N(N+1)\sigma^2\right)\\ & \leq \gamma^2n^3\lambda_{\max}^{H_B}\left\|\nabla f(W^t)\right\|^2 + \gamma^2\lambda_{\max}^{H_B}N^2\sigma^2\end{aligned}$$

Plugging to the previous bound we obtain:

$$\begin{aligned}\mathbb{E}\left[\left\|\nabla f(W^t) - H^t\hat{g}^t\right\|^2 \mid W^t\right] \leq & \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\left\|\nabla f(W^t)\right\|^2 + L^2\gamma^2N^2\lambda_{\max}^{H_B}\left\|\nabla f(W^t)\right\|^2\\ & + L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Now we have the following

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq & f(W^t) - f^\star - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2\\ & + \frac{\gamma N}{2}\left(\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\left\|\nabla f(W^t)\right\|^2 + L^2\gamma^2N^2\lambda_{\max}^{H_B}\left\|\nabla f(W^t)\right\|^2\right)\\ & + \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Using $\gamma \leq \frac{1}{2LN}$ we get

$$\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq f(W^t) - f^\star - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right) \tag{8}$$

$$+ \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2. \tag{9}$$

After rearranging the terms, we have

$$\begin{aligned}\frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right) \leq & \left(f(W^t) - f^\star\right) - \left(\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star\right)\\ & + \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Next, we have

$$\begin{aligned}\left\|\nabla f(W^t)\right\|^2 \leq & \frac{2}{\gamma N}\frac{1}{\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)}\left(\left(f(W^t)-f^\star\right)-\left(\mathbb{E}\left[f(W^{t+1})\mid W^t\right]-f^\star\right)\right)\\ &+\frac{2}{\gamma N}\frac{1}{\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)}\frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Using telescoping property and taking expectation we get

$$\begin{aligned}\frac{1}{T}\sum_{t=0}^{T-1}\left\|\nabla f(W^t)\right\|^2 \leq & \frac{2}{\gamma NT}\frac{f(W^0)-f^\star}{\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)}\\ &+\frac{L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2}{\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)}.\end{aligned}$$

Applying argument from Danilova et al. [11] we obtain the result for uniformly chosen point. ■

Corollary 12 *Suppose that Assumption 1 and Assumption 4 hold. Suppose that a stepsize $\gamma > 0$ is chosen such that $\gamma \leq \frac{1}{2LN}$. Let the updates have a form of several gradient steps (variance $\sigma^2 = 0$) We choose the output of the method $\widetilde{W}^T$ uniformly at random from $W^0, W^1, \ldots, W^{T-1}$ Then, the iterate $\widetilde{W}^T$ of* RAC-LoRA *method (Algorithm 1) with several* GD *updates (Equation 3) satisfy*

$$\mathbb{E}\left[\left\|\nabla f(\widetilde{W}^T)\right\|^2\right] \leq \frac{2}{\gamma NT}\frac{f(W^0)-f^\star}{\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H}\right)}.$$

Given that the step size is divided by the number of gradient steps allocated for each LoRA block, employing multiple gradient steps for a single LoRA block does not provide any significant benefits. This observation suggests that a single gradient step is adequate for each LoRA block. Therefore, in practical applications, it is more advantageous to utilize only one epoch per LoRA block within the training chain. This approach not only streamlines the training process but also optimizes computational efficiency, allowing for more effective resource allocation without compromising the performance of the model.

G.2. Analysis of Polyak-Łojasiewicz setting

Next, we establish the convergence rate for the Polyak-Łojasiewicz setting (Assumption 8).

Theorem 13 *Suppose that Assumption 1, Assumption 8 and Assumption 4 hold. Suppose that a stepsize $\gamma \geq 0$ is chosen such that $\gamma \leq \frac{1}{2NL}$. Then, the iterates of* RAC-LoRA *method (Algorithm 1) with* RR *updates (Equation 7) satisfy*

$$\begin{aligned}\mathbb{E}\left[f(W^T)-f^\star\right] \leq & \left(1-\gamma N\mu\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)\right)^T\mathbb{E}\left[f(W^0)-f^\star\right]\\ &+\frac{L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2}{2\left(1-\lambda_{\max}\left[\mathbb{E}\left[I-H_B^t\right]\right]-\frac{1}{4}\lambda_{\max}^{H_B}\right)}.\end{aligned}$$

Proof
We start from Equation 8:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq & f(W^t) - f^\star \\ & - \frac{\gamma N}{2}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right) \\ & + \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Using PL condition $\left\|\nabla f(W^t)\right\|^2 \geq 2\mu\left(f(W^t) - f^\star\right)$ we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq & f(W^t) - f^\star \\ & - \gamma N\mu\left(f(W^t) - f^\star\right)\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right) \\ & + \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

Taking full expectation we obtain:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \leq & \left(1 - \gamma N\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right)\right)\mathbb{E}\left[f(W^t) - f^\star\right] \\ & + \frac{\gamma N}{2}L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2.\end{aligned}$$

After unrolling the recursion we obtain

$$\begin{aligned}\mathbb{E}\left[f(W^{T}) - f^\star\right] \leq & \left(1 - \gamma N\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right)\right)^T\mathbb{E}\left[f(W^0) - f^\star\right] \\ & + \frac{L^2\gamma^2\lambda_{\max}^{H_B}N\sigma^2}{2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right)}.\end{aligned}$$

This finishes the proof. ■

Corollary 14 *Suppose that Assumption 1, Assumption 8 and Assumption 4 hold. Let the updates have a form of several gradient steps (variance $\sigma^2 = 0$) Suppose that a stepsize $\gamma \geq 0$ is chosen such that $\gamma \leq \frac{1}{2NL}$. Then, the iterates of* RAC-LoRA *method (Algorithm 1) with several* GD *updates (Equation 3) satisfy*

$$\mathbb{E}\left[f(W^{T}) - f^\star\right] \leq \left(1 - \gamma N\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - \frac{1}{4}\lambda_{\max}^{H_B}\right)\right)^T\mathbb{E}\left[f(W^0) - f^\star\right].$$

Since the step size is divided by the number of gradient steps for each LoRA block, using multiple gradient steps does not offer significant advantages. Thus, a single gradient step per LoRA block is sufficient. Practically, it is more efficient to use only one epoch per LoRA block in the training chain.

Appendix H. Analysis of RAC-LoRA with SGD under the arbitrary data sampling paradigm

In the previous section, we introduced the Random Reshuffling (RR) method, where each data point is used exactly once during each epoch, also known as sampling without replacement. This method has demonstrated strong empirical performance across various optimization tasks. However, in this section, we shift our focus to the RAC-LoRA framework, where Stochastic Gradient Descent (SGD) is applied with a more general, arbitrary data sampling procedure, allowing for broader flexibility in how data is selected and used during training.

The analysis of general sampling schemes in SGD has garnered significant attention in the literature, particularly in understanding its impact on convergence rates and optimization performance across different problem classes. For strongly convex functions, general sampling methods have been rigorously studied in works such as Gower et al. [15], which provide detailed convergence guarantees and bounds. In the case of general convex optimization problems, Khaled et al. [28] offer a thorough analysis of the performance of SGD under various sampling strategies. Furthermore, for non-convex settings, both Khaled and Richtárik [25] and Demidovich et al. [12] have explored how general sampling procedures influence the convergence behavior and optimization efficiency of SGD, shedding light on its applicability to a wide range of machine learning tasks.

In the following sections, we build on these foundational studies to examine how the flexibility of general sampling in the RAC-LoRA framework can lead to improved convergence in certain scenarios, while also maintaining robust performance across different convexity settings.

To conduct this analysis, we introduce a general assumption that extends the standard assumptions presented in Khaled and Richtárik [25].

The proof is provided for Right Sketch (Definition 3). The result for Left Sketch (Definition 2) can be derived by following the same steps.

Assumption 15 (Expected smoothness) *The second moment of the stochastic gradient satisfies*

$$\mathbb{E}\left[\|g(W)\|^2\right] \leq 2A_1\left(f(W) - f^{\inf}\right) + B_1 \cdot \|\nabla f(W)\|^2 + C_1$$

for some $A, B, C \geq 0$ and all $W \in \mathbb{R}^{m\times n}$.

Now we can also do stochastic analysis. Let us consider the SGD update for LoRA method:

$$\Delta W = \frac{\alpha}{r}\hat{B}A_S,$$

$$W^{t+1} = W^t + \frac{\alpha}{r}\hat{B}^t A_S^t, \qquad \hat{B}^t = -\gamma g(W^t)(A_S^t)^\top\left(A_S^t(A_S^t)^\top\right)^\dagger$$

Now we have

$$W^{t+1} = W^t - \gamma g(W^t)(A_S^t)^\top\left(A_S^t(A_S^t)^\top\right)^\dagger A_S^t = W^t - \gamma g(W^t)H_A^t. \tag{10}$$

H.1. Analysis of general non-convex setting

Theorem 16 *Suppose that Assumption 1 and Assumption 4 hold. Suppose that a stepsize $\gamma > 0$ is chosen such that $\gamma \leq \min\left[1/\sqrt{LA_1\lambda_{\max}^{H}T}, 1/\left(LB_1\frac{\lambda_{\max}^{H_A}}{\lambda_{\min}^{H_A}}\right)\right]$. Then, the iterate W^T of* RAC-LoRA *method (Algorithm 1) with* SGD *updates (Equation 10) satisfy*

$$\min_{0\leq t\leq T-1}\mathbb{E}\left[\left\|\nabla f(W^T)\right\|^2\right] \leq \frac{6}{\lambda_{\min}^{H_A}\gamma T}\left(f(W^0)-f^\star\right) + LC_1\gamma\frac{\lambda_{\max}^{H_A}}{\lambda_{\min}^{H_A}}.$$

Remark: Notice that if we choose $\gamma = \mathcal{O}(1/\sqrt{T})$, the above result yields the rate $\mathcal{O}(1/\sqrt{T})$.

Proof We start from L-smoothness:

$$\begin{aligned}
f(W^{t+1}) \leq& f(W^t) + \left\langle \nabla f(W^t), W^{t+1}-W^t\right\rangle + \frac{L}{2}\left\|W^{t+1}-W^t\right\|^2\\
=& f(W^t) + \left\langle \nabla f(W^t), -\gamma g(W^t)H_A^t\right\rangle + \frac{L}{2}\left\|-\gamma g(W^t)H_A^t\right\|^2\\
=& f(W^t) - \gamma\left\langle \nabla f(W^t), g(W^t)H_A^t\right\rangle + \frac{L}{2}\left\|-\gamma g(W^t)H_A^t\right\|^2.
\end{aligned}$$

Let us take conditional expectation:

$$\begin{aligned}
\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \gamma\mathbb{E}\left[\left\langle \nabla f(W^t), g(W^t)H_A^t\right\rangle \mid W^t\right]\\
&+ \frac{L}{2}\mathbb{E}\left[\left\|-\gamma g(W^t)H_A^t\right\|^2 \mid W^t\right].
\end{aligned}$$

Using that $g(W^t)$ and H_A^t are independent, so we have

$$\begin{aligned}
\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \gamma\left\langle \nabla f(W^t), \mathbb{E}\left[g(W^t)\right]\mathbb{E}\left[H_A^t\right]\right\rangle\\
&+ \frac{L}{2}\mathbb{E}\left[\left\|-\gamma g(W^t)H_A^t\right\|^2 \mid W^t\right]\\
\leq& f(W^t) - \gamma\left\langle \nabla f(W^t), \mathbb{E}\left[g(W^t)\right]\mathbb{E}\left[H_A^t\right]\right\rangle\\
&+ \gamma^2\frac{L}{2}\mathbb{E}\left[\left\langle g(W^t)H_A^t, g(W^t)H_A^t\right\rangle \mid W^t\right]\\
\leq& f(W^t) - \gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\left\|\nabla f(W^t)\right\|^2\\
&+ \gamma^2\frac{L}{2}\mathbb{E}\left[\left\langle g(W^t)H_A^t, g(W^t)H_A^t\right\rangle \mid W^t\right].
\end{aligned}$$

Using the property of projection matrix H_A^t, we have

$$\begin{aligned}
\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\left\|\nabla f(W^t)\right\|^2\\
&+ \gamma^2\frac{L}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\mathbb{E}\left[\left\|g(W^t)\right\|^2\right].
\end{aligned}$$

Now we need to use assumption on stochastic gradients. We will use the most general assumption: ABC – assumption:

$$\mathbb{E}\left[\|g(W^t)\|^2\right] \leq 2A_1(f(W^t)-f^\star) + B_1\left\|\nabla f(W^t)\right\|^2 + C_1.$$

Now we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq& f(W^t) - f^\star - \gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\left\|\nabla f(W^t)\right\|^2 \\ &+\gamma^2\frac{L}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\left(2A_1(f(W^t)-f^\star) + B_1\left\|\nabla f(W^t)\right\|^2 + C_1.\right).\end{aligned}$$

Combining these terms together we get

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq& \left(f(W^t) - f^\star\right)\left(1+\gamma^2 A_1 L\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right) \\ &- \gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\left\|\nabla f(W^t)\right\|^2\left(1-\gamma\frac{L}{2}\frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]}B_1\right) \\ &+ \gamma^2\frac{L}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]C_1. \end{aligned} \tag{11}$$

Using condition on stepsize: $1-\gamma\frac{LB_1}{2}\frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]} \geq \frac{1}{2}$ we get

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq& \left(f(W^t) - f^\star\right)\left(1+\gamma^2 A_1 L\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right) \\ &- \frac{1}{2}\gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\left\|\nabla f(W^t)\right\|^2 \\ &+ \gamma^2\frac{L}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]C_1.\end{aligned}$$

Using tower property of expectation we obtain

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \leq& \mathbb{E}\left[f(W^t) - f^\star\right]\left(1+\gamma^2 A_1 L\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right) \\ &- \frac{1}{2}\gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \\ &+ \gamma^2\frac{L}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]C_1.\end{aligned}$$

Let us define $\delta^t = \mathbb{E}\left[f(W^t) - f^\star\right]$ and $r^t = \mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right]$, after reshuffling of terms we obtain

$$\frac{1}{2}\gamma\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\mathbb{E}\left[\left\|\nabla f(W^t)\right\|^2\right] \leq \left(1+\gamma^2 A_1 L\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right)\delta^t - \delta^{t+1} + \gamma^2\frac{LC_1}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right].$$

Let use fix $w^{-1} > 0$ and define $w^t = \frac{w^{t-1}}{1+L\gamma^2 A\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}$ for all $t \geq 0$. Multiplying by $\frac{w^t}{\gamma}$,

$$\frac{1}{2}w^t r^t\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right] \leq \frac{w^t}{\gamma}\left(1+\gamma^2 A_1 L\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right)\delta^t - \frac{w^t}{\gamma}\delta^{t+1} + \gamma\frac{LC_1}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right].$$

Now we obtain

$$\frac{1}{2}w^t r^t\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right] \leq \frac{w^{t-1}}{\gamma}\delta^t - \frac{w^t}{\gamma}\delta^{t+1} + \gamma\frac{LC_1}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]w^t.$$

Summing up both sides as $t = 0, 1, \ldots, T-1$ we have,

$$\frac{1}{2}\sum_{t=0}^{T-1} w^t r^t \lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right] \leq \frac{w^{-1}}{\gamma}\delta^0 - \frac{w^{T-1}}{\gamma}\delta^T + \gamma\frac{LC_1}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\sum_{t=0}^{T-1} w^t$$
$$\leq \frac{w^{-1}}{\gamma}\delta^0 + \gamma\frac{LC_1}{2}\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\sum_{t=0}^{T-1} w^t.$$

Let us define $W^T = \sum_{t=0}^{T-1} w^t$. Dividing both sides by W^T we have,

$$\frac{1}{2}\min_{0\leq t\leq T-1} r^t \leq \frac{1}{W^T}\sum_{t=0}^{T-1} w^t r^t \leq \frac{w^{-1}}{W^T}\frac{\delta^0}{\gamma}\frac{1}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]} + \frac{LC_1\gamma}{2}\frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]}.$$

Note that,

$$W^T = \sum_{t=0}^{T-1} w^t \geq \sum_{t=0}^{T-1}\min_{0\leq i\leq T-1} w^i = Tw^{T-1} = \frac{Tw^{-1}}{\left(1 + L\gamma^2 A\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right)^T}.$$

Using this we get

$$\frac{1}{2}\min_{0\leq t\leq T-1} r^t \leq \frac{\left(1 + L\gamma^2 A_1\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right)^T}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\gamma T}\delta^0 + \frac{LC_1\gamma}{2}\frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]}.$$

Using the fact that $1 + x \leq \exp(x)$, we have that

$$\left(1 + L\gamma^2 A_1\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right)^T \leq \exp\left(L\gamma^2 A_1\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]T\right) \leq \exp(1) \leq 3$$

where the second inequality holds because $\gamma \leq 1/\sqrt{LA_1\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]T}$ by assumption. Substituting we get,

$$\min_{0\leq t\leq T-1} r^t \leq \frac{6}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\gamma T}\left(f(W^0) - f^\star\right) + LC_1\gamma\frac{\lambda_{\max}^{H_A}}{\lambda_{\min}^{H_A}}.$$

■

H.2. Analysis of Polyak-Łojasiewicz setting

In this section we provide analysis of RAC-LoRA method with general SGD update under Polyak-Łojasiewicz condition (Assumption 8).

Theorem 17 *Suppose that Assumption 1, Assumption 8 and Assumption 4 hold. Suppose that a stepsize* $\gamma \geq 0$ *is chosen such that* $\gamma \leq \min\left[\frac{\mu}{2A_1 L\frac{\lambda_{\max}^{H_A}}{\lambda_{\min}^{H_A}}}, 1/\left(LB_1\frac{\lambda_{\max}^{H_A}}{\lambda_{\min}^{H_A}}\right)\right]$. *Then, the iterates of RAC-LoRA method (Algorithm 1) with SGD updates (Equation 10) satisfy*

$$\mathbb{E}\left[f(W^T)\right] - f^\star \leq \left(1 - \gamma\mu\lambda_{\min}^{H}\right)^T\left(f(W^0) - f^\star\right).$$

Proof

We start from 11:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \le& \left(f(W^t) - f^\star\right)\left(1 + \gamma^2 A_1 L \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right) \\ &- \gamma \lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right] \left\|\nabla f(W^t)\right\|^2 \left(1 - \gamma \frac{L}{2} \frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]} B_1\right) \\ &+ \gamma^2 \frac{L}{2} \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right] C_1.\end{aligned}$$

Using $\left(1 - \gamma \frac{L}{2} \frac{\lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]}{\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]} B_1\right) \ge \frac{3}{4}$ and PL condition we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \le& \left(f(W^t) - f^\star\right)\left(1 - \frac{3}{2}\gamma\mu\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right] + \gamma^2 A_1 L \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right]\right) \\ &+ \gamma^2 \frac{L}{2} \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right] C_1.\end{aligned}$$

Using that $L A_1 \gamma \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right] \le \frac{\mu}{2} \lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]$ we obtain

$$\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \le \left(f(W^t) - f^\star\right)\left(1 - \gamma\mu\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\right) + \gamma^2 \frac{L}{2} \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right] C_1.$$

Taking full expectation and using tower property we obtain:

$$\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \le \mathbb{E}\left[f(W^t) - f^\star\right]\left(1 - \gamma\mu\lambda_{\min}\left[\mathbb{E}\left[H_A^t\right]\right]\right) + \gamma^2 \frac{L}{2} \lambda_{\max}\left[\mathbb{E}\left[H_A^t\right]\right] C_1.$$

Once we unroll the recursion we obtain

$$\mathbb{E}\left[f(W^T) - f^\star\right] \le \mathbb{E}\left[f(W^0) - f^\star\right]\left(1 - \gamma\mu\lambda_{\min}^{H_A}\right)^T + \gamma \frac{L}{2\mu\lambda_{\min}^{H_A}} \lambda_{\max}^{H_A} C_1.$$

■

Algorithm 2 Federated Randomized Asymmetric Chain of LoRA (Fed-RAC-LoRA)

Parameters: initial pre-trained model W^0, rank r, learning rate $\gamma > 0$, scaling factor α, server stepsize $\beta > 0$ number of modules in chain T, sample distribution $\mathcal{D}_S^B$ or $\mathcal{D}_S^A$.
for $t = 0, 1, \ldots, T-1$ **do**
  Sample a subset (cohort) of clients S^t
    (Option 1) Sample a matrix B_S^t (Option 2) Sample a matrix A_S^t
  Send the model W^t and fixed matrix (Option 1) B_S^t or (Option 2) A_S^t to clients
  for $m \in S^t$ **do**
    Solve subproblem

$$\text{(Option 1) } \hat{A}_m^t \approx \min_A f_m(W^t + \frac{\alpha}{r} B_S^t A) \qquad \text{(Option 2) } \hat{B}_m^t \approx \min_B f_m(W^t + \frac{\alpha}{r} B A_S^t)$$

    Send the updates to server (Option 1) $\hat{A}_m^t$ or (Option 2) $\hat{B}_m^t$
  end for
  Merge the updates

$$\text{(Option 1)} \quad W^{t+1} = W^t + \beta \frac{\alpha}{r} B_S^t \frac{1}{C} \sum_{m \in S^t} \hat{A}_m^t$$

$$\text{(Option 2)} \quad W^{t+1} = W^t + \beta \frac{\alpha}{r} \frac{1}{C} \sum_{m \in S^t} \hat{B}_m^t A_S^t$$

end for

Appendix I. Federated Learning setting

We consider the main optimization problem (1), with f having the double finite-sum structure

$$f(W^0 + \Delta W) := \frac{1}{M} \sum_{m=1}^{M} \underbrace{\frac{1}{N} \sum_{i=1}^{N} f_{m,i}(W^0 + \Delta W)}_{f_m(W^0 + \Delta W)}, \tag{12}$$

where M is the total number of clients and N is the number of data points on each client. In the context of Federated Learning, each client maintains its own local loss function f_m, which also follows a finite-sum structure, reflecting the client's local data. This formulation captures the decentralized nature of the learning process, where each client performs computations based on their local dataset.

Federated Learning (FL) [23, 31] is a distributed machine learning framework that enables multiple devices or clients to collaboratively train a shared model without sending their raw data to a central server. In contrast to traditional machine learning, where data is centralized for model training, Federated Learning allows each client to train a local model using its own data. The clients then share only the updated model parameters with a central server or aggregator. The server aggregates these updates to form a new global model, which is then redistributed to the clients for further iterations of the process [31]. Local Training (LT) is a key component of Federated Learning (FL), in which each participating client conducts several local optimization steps before synchronizing their model parameters with the central server.

The analysis of LT marked a significant advancement by eliminating the need for data homogeneity assumptions, as demonstrated by Khaled et al. [26, 27]. However, later studies by Woodworth et al. [63] and Glasgow et al. [13] revealed that LocalSGD (also known as FedAvg) has no communication complexity advantage over MinibatchSGD in heterogeneous data settings. Additionally, Malinovskiy et al. [37] analyzed LT methods for general fixed-point problems, while Koloskova et al. [30] explored decentralized aspects of LT.

Although removing the data homogeneity requirement was a major breakthrough, the results were somewhat discouraging, as they indicated that LT-enhanced GD, or LocalGD, exhibits a sublinear convergence rate, which is worse than the linear convergence rate of vanilla GD [62]. The impact of server-side step sizes was further explored by Malinovsky et al. [41] and Charles and Konečný [6].

Subsequent LT methods aimed to achieve linear convergence by addressing client drift, which had hindered earlier approaches. Scaffold, introduced by Karimireddy et al. [24], was the first to successfully mitigate client drift and achieve a linear convergence rate. Similar methods were later proposed by Gorbunov et al. [14]. Although this was a significant breakthrough, these methods still have slightly higher or equal communication complexity compared to vanilla GD.

Mishchenko et al. [46] recently introduced the ProxSkip method, a simple yet effective approach to Local Training that achieves provable communication acceleration in the smooth strongly convex regime, even with heterogeneous data. In a follow-up article, Malinovsky et al. [39] expanded on ProxSkip, presenting a broad variance reduction framework. Condat and Richtárik [9] further applied ProxSkip to complex splitting schemes involving the sum of three operators in a forward-backward setting. Additionally, Sadiev et al. [53] and Maranjyan et al. [43] improved the computational complexity of ProxSkip while preserving its communication efficiency. Condat et al. [10] introduced accelerated Local Training methods allowing client sampling based on ProxSkip, while Grudzień et al. [16, 17] proposed an accelerated method using the RandProx approach with primal and dual updates.

In practice, Federated Learning faces a fundamental challenge: it is often infeasible for all clients to communicate and aggregate updates with the central server simultaneously due to limitations such as network bandwidth, client availability, or resource constraints. Therefore, rather than requiring all clients to participate in every round of communication, we adopt a strategy in which only a randomly selected subset of clients is involved in each aggregation step. This approach relies on uniform sampling of the clients, ensuring that the selection process is unbiased over time.

The method operates as follows: in each communication round, the central server sends the current global model, denoted by W^t, along with the sampled matrix, to the clients chosen to participate in the current cohort. Each client in this cohort trains a local learnable matrix using an optimization algorithm (e.g., stochastic gradient descent) based on their local data. After completing the local updates, the clients send their computed updates (i.e., changes in model parameters) back to the central server.

Once the server receives these updates, it aggregates them (e.g., by averaging the updates) to produce an updated global model. In addition to the aggregation, the server may perform an additional server-side update step to further refine the model before broadcasting it in the next round. This iterative process of local training, communication, and aggregation continues until convergence is achieved or a predefined stopping criterion is met.

The proof is provided for Left Sketch (Definition 2). The result for Right Sketch (Definition 3) can be derived by following the same steps.

For local optimzier we use Random Reshuffling, where the effective step has a form:

$$W^t_{m,i+1} = W^t_{m,i} - \gamma H^t_B \nabla f_{m,i}(W^t_{m,i}) \tag{13}$$

The server-side step looks like $W^{t+1} = W^t - \tilde{\eta} H^t_B \frac{1}{C} \sum_{m\in S^t} \hat{A}^t_m$.

Let us formulate nesesary assumptions

Assumption 18 (Functional dissimilarity) *The variance at the optimum in the non-convex regime is defined as*

$$\Delta^\star \overset{def}{=} f^\star - \frac{1}{M}\sum_{m=1}^{M} f^\star_m$$

where $f^\star_m = \inf_W f_m(W)$ *and* $f^\star = \inf_W f(W)$*. For each device* m*, the variance at the optimum is defined as*

$$\Delta^\star_m \overset{def}{=} f^\star - \frac{1}{n}\sum_{i=1}^{n} f^\star_{m,i}$$

where $f^\star_{m,i} = \inf_W f_{m,i}(W)$

I.1. Analysis of general non-convex setting

Theorem 19 *Suppose that Assumption 1 and Assumption 4 hold. Suppose that stepsizes* $\gamma, \tilde{\eta} > 0$ *is chosen such that* $\gamma n \leq \tilde{\eta} \leq \frac{1-\lambda^{H_B}_{\min}}{4L}$*. Then, the iterate* W^T *of* Fed-RAC-LoRA *method (Algorithm 2) with* RR *updates (Equation 13) satisfy*

$$\begin{aligned}
\min_{t=0,\ldots,T-1} \mathbb{E}\left[\left\|\nabla f\left(W^t\right)\right\|^2\right] \leq & \frac{4\left(1 + 4\tilde{\eta}L^3\gamma^2N^2 + 2L^2\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\right)^T}{\lambda_{\max}\left[\mathbb{E}\left[I - H^t_B\right]\right]\tilde{\eta}T}\left(f(W^0) - f^\star\right) \\
&+ \frac{8\gamma^2NL^3}{\lambda_{\max}\left[\mathbb{E}\left[I - H^t_B\right]\right]}\left(\frac{1}{M}\sum_{m=1}^{M}\Delta^*_m + N\Delta^*\right) \\
&+ \frac{8L^2\tilde{\eta}}{\lambda_{\max}\left[\mathbb{E}\left[I - H^t_B\right]\right]}\frac{M-C}{C\max\{M-1,1\}}\Delta^*.
\end{aligned}$$

Proof

We start from L-smoothness:

$$\begin{aligned}
f(W^{t+1}) \leq & f(W^t) + \left\langle \nabla f(W^t), W^{t+1} - W^t \right\rangle + \frac{L}{2}\left\|W^{t+1} - W^t\right\|^2 \\
\leq & f(W^t) - \left\langle \nabla f(W^t), \tilde{\eta}\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H^t \nabla f^{\pi^t_{m,i}}_m\left(W^t_{m,i}\right)\right\rangle \\
&+ \frac{L}{2}\left\|\tilde{\eta}\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H^t \nabla f^{\pi^t_{m,i}}_m(W^t_{m,i})\right\|^2.
\end{aligned}$$

Now we take expectation with respect to sampling:

$$
\begin{aligned}
\mathbb{E}_{S^t}\left[f(W^{t+1})\right] \leq & f(W^t) - \tilde{\eta}\mathbb{E}_{S^t}\left[\left\langle \nabla f(W^t), \frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}\left(W_{m,i}^t\right)\right\rangle\right] \\
& + \frac{L}{2}\tilde{\eta}^2\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \\
\leq & f(W^t) - \tilde{\eta}\left\langle \nabla f(W^t), \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}\left(W_{m,i}^t\right)\right\rangle \\
& + \frac{L}{2}\tilde{\eta}^2\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right].
\end{aligned}
$$

Using $2\langle a, b\rangle = \|a+b\|^2 - \|a\|^2 - \|b\|^2$, we have

$$
\begin{aligned}
\mathbb{E}_{S^t}\left[f(W^{t+1})\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 - \frac{\tilde{\eta}}{2}\left\|\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2 \\
& + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t) - \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \\
\leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t) - \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right].
\end{aligned}
$$

Now we need to add and subtract $H_B^t \nabla f(W^t)$:

$$\begin{aligned}
\mathbb{E}_{S^t}\left[f(W^{t+1})\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 \\
& + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t) - H_B^t \nabla f(W^t) + H_B^t \nabla f(W^t) - \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2 \mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \\
\leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 \\
& + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\left(I - H_B^t\right) + \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W^t) - \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2 \mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \\
\leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 \\
& + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\left(I - H_B^t\right) + \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \left(\nabla f_m^{\pi_{m,i}^t}(W^t) - \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2 \mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right].
\end{aligned}$$

Since $H_B^t(I - H_B^t) = 0$ we obtain

$$\begin{aligned}
\mathbb{E}_{S^t}\left[f(W^{t+1})\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\|\nabla f(W^t)\|^2 \\
& + \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\left(I - H_B^t\right)\right\|^2 + \frac{\tilde{\eta}}{2}\left\|\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \left(\nabla f_m^{\pi_{m,i}^t}(W^t) - \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right)\right\|^2 \\
& + \frac{L}{2}\tilde{\eta}^2 \mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right].
\end{aligned}$$

Now we take conditional expectation and use tower property:

$$\begin{aligned}
\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\mathbb{E}\left[\|\nabla f(W^t)\left(I - H_B^t\right)\|^2 \mid W^t\right] \\
& + \frac{\tilde{\eta}}{2}\mathbb{E}\left[\left\|\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1} H_B^t \left(\nabla f_m^{\pi_{m,i}^t}(W^t) - \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right)\right\|^2 \mid W^t\right] \\
& + \frac{L}{2}\tilde{\eta}^2 \mathbb{E}\left[\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1} H_B^t \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \mid W^t\right].
\end{aligned}$$

Next, we use eigenvalues to obtain bounds:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\|\nabla f(W^t)\|^2 \\ & + \frac{\tilde{\eta}}{2}\mathbb{E}\left[\lambda_{\max}\left[H_B^t\right]\left\|\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1}\left(\nabla f_m^{\pi_{m,i}^t}(W^t) - \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right)\right\|^2 \mid W^t\right] \\ & + \frac{L}{2}\tilde{\eta}^2\mathbb{E}\left[\mathbb{E}_{S^t}\left[\lambda_{\max}\left[H_B^t\right]\left\|\frac{1}{Cn}\sum_{m\in S^t}\sum_{i=0}^{N-1}\nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \mid W^t\right].\end{aligned}$$

Since $\lambda_{\max}\left[H_B^t\right] = 1$ we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\|\nabla f(W^t)\|^2 \\ & + \frac{\tilde{\eta}}{2}\mathbb{E}\left[\left\|\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1}\left(\nabla f_m^{\pi_{m,i}^t}(W^t) - \nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right)\right\|^2 \mid W^t\right] \\ & + \frac{L}{2}\tilde{\eta}^2\mathbb{E}\left[\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1}\nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \mid W^t\right].\end{aligned}$$

Using Lemma 5 from Malinovsky et al. [41] we have

$$\begin{aligned}&\frac{L}{2}\tilde{\eta}^2\mathbb{E}\left[\mathbb{E}_{S^t}\left[\left\|\frac{1}{CN}\sum_{m\in S^t}\sum_{i=0}^{N-1}\nabla f_m^{\pi_{m,i}^t}(W_{m,i}^t)\right\|^2\right] \mid W^t\right] \\ &\leq L^3\tilde{\eta}^2\mathbb{E}\left[\frac{1}{Mn}\sum_{m=1}^{M}\sum_{i=0}^{N-1}\left\|W_{m,i}^t - W^t\right\|^2 \mid W^t\right] \\ &\quad + L\tilde{\eta}^2\left\|\nabla f\left(W^t\right)\right\|^2 \\ &\quad + L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right)\end{aligned}$$

Using this bound and L-smoothness for the term in second line we obtain:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\|\nabla f(W^t)\|^2 \\ & + \frac{\tilde{\eta}}{2}L^2\mathbb{E}\left[\frac{1}{Mn}\sum_{m=1}^{M}\sum_{i=0}^{n-1}\left\|W^t - W_{m,i}^t\right\|^2 \mid W^t\right] \\ & + L^3\tilde{\eta}^2\mathbb{E}\left[\frac{1}{Mn}\sum_{m=1}^{M}\sum_{i=0}^{n-1}\left\|W^t - W_{m,i}^t\right\|^2 \mid W^t\right] + L\tilde{\eta}^2\left\|\nabla f\left(W^t\right)\right\|^2 \\ & + L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right)\end{aligned}$$

Since $\tilde{\eta} \leq \frac{1}{2L}$ we get

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] &\leq f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\|\nabla f(W^t)\|^2 \\ &+ \tilde{\eta}L^2\mathbb{E}\left[\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1}\left\|W^t - W_{m,i}^t\right\|^2 \mid W^t\right] + L\tilde{\eta}^2\left\|\nabla f\left(W^t\right)\right\|^2 \\ &+ L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Using lemma 6 from (cite) we obtain

$$\begin{aligned}\frac{1}{MN}\sum_{m=1}^{M}\sum_{i=0}^{N-1}\mathbb{E}\left[\left\|W^t - W_{m,i}^t\right\|^2 \mid W^t\right] &\leq 4\gamma^2N^2L\left(f\left(W^t\right) - f^*\right) \\ &+2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*.\end{aligned}$$

Plugging this bound we obtain

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 + \frac{\tilde{\eta}}{2}\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\|\nabla f(W^t)\|^2 \\ &+ \tilde{\eta}L^2\left(4\gamma^2N^2L\left(f\left(W^t\right) - f^*\right) + 2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) \\ &+ L\tilde{\eta}^2\left\|\nabla f\left(W^t\right)\right\|^2 \\ &+ L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Next, we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right] - 2L\tilde{\eta}\right) \\ &+ \tilde{\eta}L^2\left(4\gamma^2N^2L\left(f\left(W^t\right) - f^*\right) + 2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) \\ &+ L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Using $\tilde{\eta} \leq \frac{1-\lambda_{\max}\left[\mathbb{E}\left[H_B^t\right]\right]}{4L}$ we get

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq& f(W^t) - \frac{\tilde{\eta}}{4}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\right) \\ &+ \tilde{\eta}L^2\left(4\gamma^2N^2L\left(f\left(W^t\right) - f^*\right) + 2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) \\ &+ L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Next, we subtract $f^\star$ from both sides:

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] - f^\star \leq & f(W^t) - f^\star - \frac{\tilde{\eta}}{4}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\right) \\ & + \tilde{\eta}L^2\left(4\gamma^2N^2L\left(f\left(W^t\right) - f^*\right) + 2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) \\ & + L\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Taking full expectation we obtain

$$\begin{aligned}&\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \leq \mathbb{E}\left[f(W^t) - f^\star\right]\left(1 + 4\tilde{\eta}L^3\gamma^2N^2 + 2L^2\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\right) \\ &- \frac{\tilde{\eta}}{4}\left\|\nabla f(W^t)\right\|^2\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\right) \\ &+ \tilde{\eta}L^2\left(2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) + 2L^2\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\Delta^*.\end{aligned}$$

Next, we apply lemma from Khaled and Richtárik [25] and obtain

$$\begin{aligned}\min_{t=0,\ldots,T-1}\mathbb{E}\left[\left\|\nabla f\left(W^t\right)\right\|^2\right] \leq & \frac{4\left(1 + 4\tilde{\eta}L^3\gamma^2N^2 + 2L^2\tilde{\eta}^2\frac{M-C}{C\max\{M-1,1\}}\right)^T}{\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]\tilde{\eta}T}\left(f(W^0) - f^\star\right) \\ & + \frac{8\gamma^2NL^3}{\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]}\left(\frac{1}{M}\sum_{m=1}^{M}\Delta_m^* + N\Delta^*\right) \\ & + \frac{8L^2\tilde{\eta}}{\lambda_{\max}\left[\mathbb{E}\left[I - H_B^t\right]\right]}\frac{M-C}{C\max\{M-1,1\}}\Delta^*.\end{aligned}$$

■

I.2. Analysis of Polyak-Łojasiewicz setting

Theorem 20

Suppose that Assumption 1, Assumption 8 and Assumption 4 hold. Suppose that stepsizes $\gamma, \tilde{\eta} > 0$ is chosen such that $\gamma n \leq \tilde{\eta} \leq \frac{1-\lambda_{\min}^{H_B}}{4L}$. Then, the iterate W^T of Fed-RAC-LoRA *method (Algorithm 2) with* RR *updates (Equation 13) satisfy*

$$\begin{aligned}&\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \leq \left(f(W^0) - f^\star\right)\left(1 - \tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)\right)^T \\ &+ \frac{\tilde{\eta}L^2}{\tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)}\left(2\gamma^2N^2L\Delta^* + 2\gamma^2NL\frac{1}{M}\sum_{m=1}^{M}\Delta_m^*\right) \\ &+ \frac{L\tilde{\eta}^2}{\tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)}\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Proof We start from

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \frac{\tilde{\eta}}{2}\left\|\nabla f(W^t)\right\|^2 \left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 2L\tilde{\eta}\right) \\ & + \tilde{\eta}L^2 \left(4\gamma^2 N^2 L \left(f\left(W^t\right) - f^*\right) + 2\gamma^2 N^2 L\Delta^* + 2\gamma^2 N L \frac{1}{M}\sum_{m=1}^{M} \Delta_m^*\right) \\ & + L\tilde{\eta}^2 \frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Using Assumption8 we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \tilde{\eta}\mu\left\|\nabla f(W^t)\right\|^2 \left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 2L\tilde{\eta}\right)\left(f(W^t) - f^\star\right) \\ & + \tilde{\eta}L^2 \left(4\gamma^2 n^2 L \left(f\left(W^t\right) - f^*\right) + 2\gamma^2 N^2 L\Delta^* + 2\gamma^2 N L \frac{1}{M}\sum_{m=1}^{M} \Delta_m^*\right) \\ & + L\tilde{\eta}^2 \frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

Using the stepsize $\gamma \leq \frac{1}{4nL}$ we have

$$\begin{aligned}\mathbb{E}\left[f(W^{t+1}) \mid W^t\right] \leq & f(W^t) - \tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)\left(f(W^t) - f^\star\right) \\ & + \tilde{\eta}L^2 \left(2\gamma^2 N^2 L\Delta^* + 2\gamma^2 N L \frac{1}{M}\sum_{m=1}^{M} \Delta_m^*\right) \\ & + L\tilde{\eta}^2 \frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

After unrolling the recursion we obtain

$$\begin{aligned}&\mathbb{E}\left[f(W^{t+1}) - f^\star\right] \leq \left(f(W^0) - f^\star\right)\left(1 - \tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)\right)^T \\ & + \frac{\tilde{\eta}L^2}{\tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)}\left(2\gamma^2 N^2 L\Delta^* + 2\gamma^2 N L \frac{1}{M}\sum_{m=1}^{M} \Delta_m^*\right) \\ & + \frac{L\tilde{\eta}^2}{\tilde{\eta}\mu\left(1 - \lambda_{\max}\left[\mathbb{E}\left[I - H^t\right]\right] - 3L\tilde{\eta}\right)}\frac{M-C}{C\max\{M-1,1\}}\left(2L\left(f\left(W^t\right) - f^\star\right) + 2L\Delta^*\right).\end{aligned}$$

■