OPT2025: 17th Annual Workshop on Optimization for Machine Learning

AdaGrad Meets Muon: Adaptive Stepsizes for Orthogonal Updates

Minxin Zhang MINXINZHANG@MATH.UCLA.EDU
Yuxuan Liu YXLIU@MATH.UCLA.EDU
Hayden Schaeffer HAYDEN@MATH.UCLA.EDU
Department of Mathematics, University of California, Los Angeles

Abstract

The recently proposed optimizer Muon updates weight matrices via orthogonalized momentum and has demonstrated strong empirical success in large language model training. However, it remains unclear how to determine the learning rates for such orthogonalized updates. AdaGrad, by contrast, is a widely used adaptive method that scales stochastic gradients by accumulated past gradients. We propose a new algorithm, AdaGO, which combines a norm-based AdaGrad-type stepsize with an orthogonalized update direction, bringing together the benefits of both approaches. Unlike other adaptive variants of Muon, AdaGO preserves the orthogonality of the update direction, which can be interpreted as a spectral descent direction, while adapting the stepsizes to the optimization landscape by scaling the direction with accumulated past gradient norms. The implementation of AdaGO requires only minimal modification to Muon, with a single additional scalar variable, the accumulated squared gradient norms, to be computed, making it computationally and memory efficient. Optimal theoretical convergence rates are established for nonconvex functions in both stochastic and deterministic settings under standard smoothness and unbiased bounded-variance noise assumptions. Empirical results on CIFAR-10 classification and function regression demonstrate that AdaGO outperforms Muon and Adam.

1. Introduction

The trainable parameters of neural networks, including those in large language models (LLMs), are often arranged as matrices. While widely used optimization algorithms such as stochastic gradient descent (SGD), Adam [15], and their variants treat these parameters as flattened vectors, the recently proposed Muon optimizer [13] explicitly leverages their matrix structure. By updating weight matrices with orthogonalized momentum, Muon has demonstrated superior empirical performance [21, 25]. Nevertheless, a fundamental question remains unresolved: what constitutes an effective learning rate for Muon? More broadly, how should one determine effective learning rates for optimizers that employ orthogonal updates?

Given a matrix $M \in \mathbb{R}^{m\times n}$, its orthogonalization is defined as

$$\mathrm{Orth}\,(M) := \operatorname*{argmin}_{O\in\mathbb{R}^{m\times n}} \left\{ \|O - M\|_F : OO^T = I_m \text{ or } O^TO = I_n \right\},$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Equivalently, if $M = U\Sigma V^T$ is the reduced singular value decomposition (SVD) of M, then $\mathrm{Orth}\,(M) = UV^T$ [4, Proposition 4]. Orthogonalized gradient descent (OGD) can be interpreted as the steepest descent under the spectral norm [5, 10]. The key distinction of Muon from other SGD-based methods is the orthogonalization of the momentum. Since computing the exact orthogonalization is expensive, Muon employs Newton–Schulz

iterations to obtain an efficient approximation in practice [3], whereas theoretical analyses typically assume exact orthogonalization at each iteration [7, 17, 19, 23, 24, 26]. Existing convergence results of Muon generally assume a small constant learning rate; in practice, however, considerable effort is devoted to tuning the learning rate or designing an appropriate learning-rate schedule, as is standard for SGD-based algorithms. However, Muon fundamentally differs from these methods: its orthogonalized updates alter the optimization dynamics in a significant way [3]. This raises a natural question: do the established methods for learning rate selection still apply, or does Muon require a fundamentally different approach? Empirically, when the gradient norm is large at the start of training, employing a large learning rate in Muon produces a rapid initial decrease in training loss but soon leads to plateauing and oscillations as the gradient norm becomes smaller. In contrast, a smaller constant learning rate results in slower convergence yet ultimately achieves a lower final loss. These observations suggest that an adaptively tuned learning rate schedule, informed by the gradients, has the potential to further improve the efficiency of orthogonal updates in the Muon optimizer.

One widely known adaptive SGD method, AdaGrad, adjusts learning rates based on the cumulative history of squared gradients. Originally proposed in [9], the full-matrix variant of AdaGrad scales the update direction using the full outer product of past gradients, whereas the more practical diagonal AdaGrad retains only the diagonal entries of this matrix. A more recent variant, AdaGrad-Norm [30], scales the learning rate by the square root of the accumulated gradient norms. Whereas full-matrix and diagonal AdaGrad adaptively rescale the learning rate for each parameter and thus alter the update direction, AdaGrad-Norm adjusts the stepsize through a single scalar factor while preserving the original stochastic gradient direction. These AdaGrad stepsizes have been extensively studied in the context of standard stochastic gradients [9, 11, 16, 20, 28, 30, 31]. However, their behavior under modified update directions, such as those obtained through orthogonalization, remains largely unexplored. In this work, we bridge this gap by introducing adaptive stepsizes for the orthogonalized directions. Motivated by the strong empirical performance of orthogonalized momentum in the Muon optimizer, we introduce a learning rate schedule that adapts to past gradients while preserving orthogonality in the updates. Specifically, we present a new algorithm, AdaGO, which combines a norm-based AdaGrad stepsize with orthogonalized update directions, and establish theoretical convergence guarantees for nonconvex functions.

1.1. Related work

The convergence of Muon has been analyzed in [17, 19, 23, 26], which show that Muon converges to a stationary point at a rate of $\mathcal{O}(T^{-1/4})$ when using a constant stepsize of magnitude $\mathcal{O}(T^{-3/4})$, where T denotes total number of iterations. The analysis in [24] covers four practical Muon variants, with and without Nesterov momentum and with and without weight decay, and derives the critical batch size. The analysis in [7] interprets Muon as solving a spectral norm constrained problem within the Lion-K framework and establishes convergence to KKT points at a rate that depends on the batch size.

Several adaptive variants of Muon have been proposed. AdaMuon [27] combines Muon with element-wise adaptivity, showing empirical improvements without theoretical convergence guarantees. COSMOS [22] combines SOAP [29] and Muon for memory-efficient LLM training, reporting practical benefits in stability and memory usage, but also lacks convergence guarantees. Shampoo [12] precedes Muon and is equivalent to it when momentum and accumulation are omitted. ASGO

[1] introduces an adaptive one-sided preconditioner, equivalent to Muon when momentum and accumulation are omitted. PolarGrad [18] unifies matrix-aware preconditioned optimizers and proposes polar-decomposition updates that subsume Muon. For Shampoo, ASGO, and PolarGrad, theoretical convergence has been established in convex settings.

1.2. Contributions and organization

We propose a new algorithm, AdaGO, which combines a norm-based AdaGrad-type stepsize with an orthogonalized update direction, bringing together the benefits of Muon and AdaGrad. Unlike other adaptive variants of Muon, AdaGO preserves the orthogonality of the update direction, while adapts the stepsizes to the optimization landscape. The implementation of AdaGO requires minimal modification to Muon, with a single additional scalar variable, the accumulated squared gradient norms, to be computed, making it computationally and memory efficient. Optimal theoretical convergence rates are established for nonconvex functions in both stochastic and deterministic settings under standard assumptions. Empirical results on CIFAR-10 classification and function regression demostate that AdaGO outperforms Muon and Adam. The rest of paper is organized as follows. We introduce the new algorithm in Section 2, and present the theoretical analysis in Section 3, with proofs deferred to Appendices B–C. Experimental results are reported in Section 4, and Section 5 concludes with a discussion of future directions.

2. AdaGO: A New Algorithm

In this section, we present the new algorithm, AdaGO, combining stepsizes adaptively tuned by past gradients with orthogonalized updates. The details of AdaGO are summarized in Algorithm 1.

At each iteration, AdaGO updates the training parameters by

$$\Theta_t = \Theta_{t-1} - \alpha_t O_t, \quad \text{with } \alpha_t := \max\left\{\epsilon, \eta\frac{\min\{\|G_t\|, \gamma\}}{v_t}\right\},$$

where O_t is the orthogonalized momentum and α_t is an adaptive stepsize. Recall that AdaGrad-Norm accumulates the squared norms of past gradients and scales the stochastic gradient by the reciprocal of the square root of this accumulation, with its convergence rate established under the relatively restrictive assumption that the gradient norms are uniformly bounded [30]. For AdaGO, we remove this assumption and instead accumulate the squared norms clamped by a large constant $\gamma > 0$ to obtain $v_t^2 = \sum_{\tau=0}^{t} \min\{\|G_\tau\|^2, \gamma^2\}$. Empirically, AdaGO performs robustly across a wide range of γ values. To prevent numerical instability from division by small denominators in the stepsize computation, we initialize the accumulator v_t with $v_0 > 0$. Theoretically, Section 3 shows that γ and v_0 appear only in logarithmic terms in the convergence error bounds, and thus have limited impact on performance. Moreover, since the orthogonalized momentum has unit magnitude, we scale O_t by the clamped current gradient norm, i.e., $\min\{\|G_t\|, \gamma\}$. This ensures that the per-iteration update decays to zero as AdaGO converges to a stationary point—a property known as *null gradient consistency*, which is generally desirable for optimization algorithms [18]. In addition, we impose a lower bound $\epsilon > 0$ on the stepsizes, thereby ensuring that AdaGO converges at least as fast as Muon with a small constant stepsize. As shown in the analysis in Section 3, the choice of ϵ depends on the optimization stopping time T. The theoretical results in the following section hold for any choice of matrix norm for the gradients. In practice, however, we use the Frobenius norm of G_t in Lines 6 and 8 of Algorithm 1 for computational efficiency.

Algorithm 1: AdaGO

Require *Learning rate* $\eta > 0$, *momentum* $\mu \in [0, 1)$, *batch size* $\{b_t\}$, $\gamma > 0$, $\epsilon > 0$;
Initialize $M_0 = 0$, $v_0 > 0$;
for $t = 1, 2, \ldots, T$ **do**
  Sample a minibatch of size b_t and compute stochastic gradient $G_t = \nabla \mathcal{L}_t(\Theta_{t-1})$;
  $M_t \leftarrow \mu M_{t-1} + (1-\mu) G_t$;
  $v_t^2 \leftarrow v_{t-1}^2 + \min\{\|G_t\|^2, \gamma^2\}$;
  $O_t \leftarrow \mathrm{Orth}(M_t)$;
  Update parameters $\Theta_t \leftarrow \Theta_{t-1} - \max\{\epsilon, \eta \frac{\min\{\|G_t\|, \gamma\}}{v_t}\} O_t$;
end
Return Θ_T;

3. Convergence Analysis

For the convergence analysis of AdaGO, we impose the standard assumptions that the loss function $\mathcal{L}(\Theta)$ is L-smooth and that the stochastic gradient is an unbiased estimator of the true gradient with bounded variance. With the trainable parameters organized as a matrix $\Theta \in \mathbb{R}^{m \times n}$, the gradient $\nabla \mathcal{L}(\Theta)$ is a matrix of the same dimensions.

Assumption 3.1 *The gradient of $\mathcal{L}(\Theta)$ is Lipschitz continuous, i.e., for arbitrary $\Theta, \Theta' \in \mathbb{R}^{m \times n}$,*

$$\left\|\nabla \mathcal{L}(\Theta) - \nabla \mathcal{L}(\Theta')\right\|_* \leq L \left\|\Theta - \Theta'\right\|_2 \tag{1}$$

for some constant $L > 0$, where $\|\cdot\|_*$ and $\|\cdot\|_2$ denote the nuclear norm and the spectral norm respectively.

Assumption 3.2 *At each iteration t, the stochastic gradient G_t is an unbiased estimate of the true gradient, i.e., $\mathbb{E}[G_t] = \nabla \mathcal{L}(\Theta_{t-1})$, with a uniformly bounded variance*

$$\mathbb{E}\left[\|G_t - \nabla \mathcal{L}(\Theta_{t-1})\|_F^2\right] \leq \frac{\kappa^2}{b_t},$$

where $b_t \geq 1$ is the batch size and $\|\cdot\|_F$ denotes the Frobenius norm.

Note that Assumption 3.1 is equivalent to a more commonly used assumption:

$$\left\|\nabla \mathcal{L}(\Theta) - \nabla \mathcal{L}(\Theta')\right\|_F \leq L' \left\|\Theta - \Theta'\right\|_F \tag{2}$$

for a different Lipschitz constant $L' > 0$. Since OGD is interpreted as the steepest descent under the spectral norm, we assume (1) for the analysis of AdaGO. A detailed discussion on the two equivalent assumptions are presented in [26].

The convergence of AdaGO is established in the following theorem, with the proof provided in Appendix B.

Theorem 3.3 *Suppose Assumptions 3.1–3.2 holds. Let $\{\Theta_t\} \subset \mathbb{R}^{m \times n}$ be the sequence of iterates generated by Algorithm 1 and write $\Delta := \mathcal{L}(\Theta_0) - \min_\Theta \mathcal{L}(\Theta)$ and $r := \min\{m, n\}$. If we set $b_t \equiv 1$, $\epsilon = T^{-\frac{3}{4}}$, $1 - \mu = T^{-\frac{1}{2}}$, and $\eta = T^{-\left(\frac{3}{8}+q\right)}$ for arbitrary $q > 0$, then, for large T,*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\nabla \mathcal{L}(\Theta_{t-1})\|_*\right] \leq \mathcal{O}\left(\frac{\Delta + \kappa\sqrt{r} + L}{T^{\frac{1}{4}}} + \frac{L\sqrt{r}}{T^{\frac{1}{4}+q}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right) + 1\right)\right).$$

By [2, Theorem 3], the $\mathcal{O}(T^{-1/4})$ rate established above is the best possible convergence rate for stochastic first-order methods under Assumptions 3.1–3.2.

We also establish the convergence of AdaGO in the deterministic setting without momentum in the following theorem, with the proof given in Appendix C.

Theorem 3.4 *Suppose Assumptions 3.1–3.2 holds. Let $\{\Theta_t\}$ be the sequence of iterates generated by Algorithm 1 using full batch with $\mu = 0$. Write $\Delta := \mathcal{L}(\Theta_0) - \min_\Theta \mathcal{L}(\Theta)$ and $r := \min\{m, n\}$. If $\epsilon = T^{-\frac{1}{2}}$ and $\eta = T^{-q}$ for arbitrary $q > 0$, then, for large T,*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\nabla \mathcal{L}(\Theta_{t-1})\|_*\right] \leq \mathcal{O}\left(\frac{\Delta + L}{\sqrt{T}}\right).$$

As shown in [6, Theorem 2], the $\mathcal{O}(1/\sqrt{T})$ rate established above is the best possible convergence rate for deterministic first-order methods under Assumption 3.1.

4. Experiments

Baselines. We compare our proposed optimizer, AdaGO, against two strong baselines: Adam [15] and Muon [14]. Since Muon and AdaGO are designed specifically for matrix parameters, we use Adam to optimize all scalar and vector parameters in the models. For brevity, we refer to these hybrid methods simply as Muon and AdaGO.

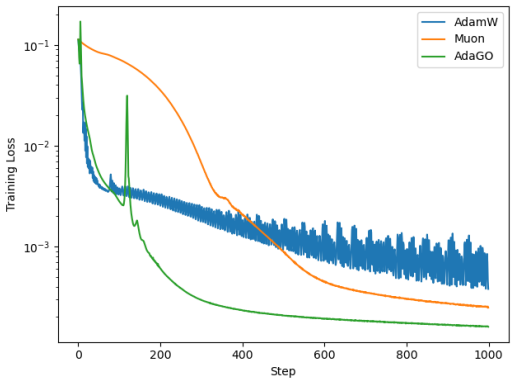
In our experiments, we use standard hyperparameter settings for the baselines. For Adam, we set the momentum coefficients to $\beta_1 = 0.9$ and $\beta_2 = 0.95$. For Muon and AdaGO, the momentum coefficient is set to $\beta = 0.95$. We perform a grid search to find the optimal learning rate η for each optimizer on each task. For AdaGO, we also tune the ϵ hyperparameter. Weight decay is not used.

Datasets and Models. We evaluate the optimizers on two tasks: function regression and image classification on CIFAR-10.

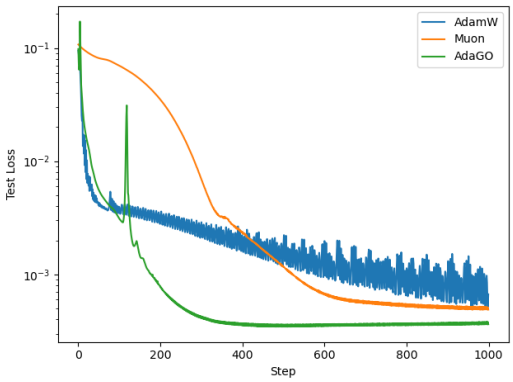
For the function regression task, we generate a dataset by sampling 10,000 points from a Gaussian random field with 50-dimensional input and 50-dimensional output (10% are used as testing data). We use a two-layer MLP with GeLU activation and a hidden dimension of 100 to fit the data. The model is trained for 1000 steps using the mean squared error loss.

For CIFAR-10 classification, we use a convolutional neural network consisting of 3 convolutional layers and 2 fully connected layers. We train the model for 100 epochs using a batch size of 128 and the standard cross-entropy loss. We report both the training loss and the test accuracy.

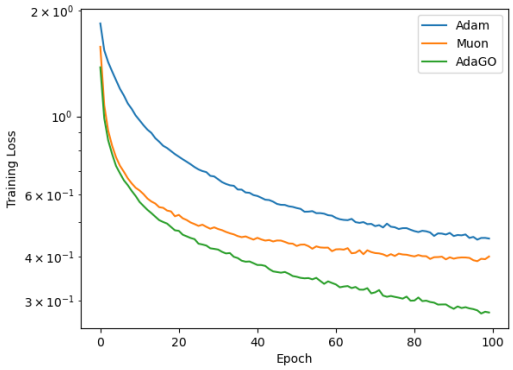
Results. The optimal hyperparameters obtained from sweeping are as follows: for regression tasks, Adam uses a stepsize of 0.01, Muon uses a stepsize of $\eta = 5e{-}3$, while AdaGO uses $\eta = 0.5$ with $\epsilon = 0.005$; for classification tasks, Adam uses $\eta = 3e{-}4$, Muon uses $\eta = 2e{-}3$, and AdaGO uses $\eta = 5e{-}2$ with $\epsilon = 5e{-}4$. Although AdaGO introduces an additional hyperparameter ϵ, both theoretical analysis and empirical observations indicate that its choice is guided by the value of η; specifically, an effective ϵ satisfies $\epsilon < \eta^2$. The regression results (training and test loss) are shown in Figure 1(a)–(b), and the CIFAR-10 results (training loss and test accuracy) appear in Figure 1(c)–(d). These results show that AdaGO consistently outperforms Adam and Muon in both tasks, experimentally demonstrating its superior performance.



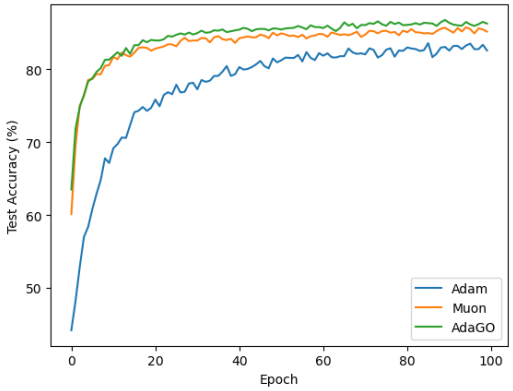
(*a*) Regression: training loss



(*b*) Regression: test loss



(*c*) Classification: training loss



(*d*) Classification: test accuracy

Figure 1: Performance of the optimizers on (a–b) the regression task and (c–d) the CIFAR-10 classification task.

5. Conclusions and Future Work

In this work, we propose AdaGO, a new optimizer that combines a norm-based AdaGrad-type stepsize with an orthogonalized update direction, bringing together the benefits of both Muon and AdaGrad. Unlike other adaptive variants of Muon, AdaGO preserves the orthogonality of the update directions while adapting stepsizes to the optimization landscape. Its implementation requires only minimal modification to Muon, with a single additional scalar variable, the accumulated squared gradient norms, to be computed, making it both computationally and memory efficient. We establish optimal convergence rates for nonconvex functions in both stochastic and deterministic settings under standard assumptions. Experimental results on CIFAR-10 classification and function regression tasks demonstrate the consistent improved performance of AdaGO over Muon and Adam. Future work includes testing AdaGO on LLM training, analyzing the algorithm under weaker assumptions, and developing new adaptive strategies for orthogonalized updates.

References

[1] Kang An, Yuxing Liu, Rui Pan, Yi Ren, Shiqian Ma, Donald Goldfarb, and Tong Zhang. Asgo: Adaptive structured gradient optimization. *arXiv preprint arXiv:2503.20762*, 2025.

[2] Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199(1):165–214, 2023.

[3] Jeremy Bernstein. The modula docs, 2025. URL https://docs.modula.systems/.

[4] Jeremy Bernstein and Laker Newhouse. Old optimizer, new norm: An anthology. *arXiv preprint arXiv:2409.20325*, 2024.

[5] David E Carlson, Edo Collins, Ya-Ping Hsieh, Lawrence Carin, and Volkan Cevher. Preconditioned spectral descent for deep learning. *Advances in neural information processing systems*, 28, 2015.

[6] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020.

[7] Lizhang Chen, Jonathan Li, and Qiang Liu. Muon optimizes under spectral norm constraints. *arXiv preprint arXiv:2506.15054*, 2025.

[8] Ashok Cutkosky and Harsh Mehta. Momentum improves normalized sgd. In *International conference on machine learning*, pages 2260–2268. PMLR, 2020.

[9] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[10] Chen Fan, Mark Schmidt, and Christos Thrampoulidis. Implicit bias of spectral descent and muon on multiclass separable data. *arXiv preprint arXiv:2502.04664*, 2025.

[11] Matthew Faw, Isidoros Tziotis, Constantine Caramanis, Aryan Mokhtari, Sanjay Shakkottai, and Rachel Ward. The power of adaptivity in sgd: Self-tuning step sizes with unbounded gradients and affine variance. In *Conference on Learning Theory*, pages 313–355. PMLR, 2022.

[12] Vineet Gupta, Tomer Koren, and Yoram Singer. Shampoo: Preconditioned stochastic tensor optimization. In *International Conference on Machine Learning*, pages 1842–1850. PMLR, 2018.

[13] K Jordan, Y Jin, V Boza, Y Jiacheng, F Cecista, L Newhouse, and J Bernstein. Muon: An optimizer for hidden layers in neural networks. *URL https://kellerjordan. github. io/posts/muon*, 2024.

[14] Keller Jordan, Yuchen Jin, Vlado Boza, You Jiacheng, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. URL https://kellerjordan.github.io/posts/muon/.

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

[16] Dmitry Kovalev. Sgd with adaptive preconditioning: Unified analysis and momentum acceleration. *arXiv preprint arXiv:2506.23803*, 2025.

[17] Dmitry Kovalev. Understanding gradient orthogonalization for deep learning via non-euclidean trust-region optimization. *arXiv preprint arXiv:2503.12645*, 2025.

[18] Tim Tsz-Kit Lau, Qi Long, and Weijie Su. Polargrad: A class of matrix-gradient optimizers from a unifying preconditioning perspective. *arXiv preprint arXiv:2505.21799*, 2025.

[19] Jiaxiang Li and Mingyi Hong. A note on the convergence of muon. *arXiv preprint arXiv:2502.02900*, 2025.

[20] Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *The 22nd international conference on artificial intelligence and statistics*, pages 983–992. PMLR, 2019.

[21] Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, et al. Muon is scalable for llm training. *arXiv preprint arXiv:2502.16982*, 2025.

[22] Liming Liu, Zhenghao Xu, Zixuan Zhang, Hao Kang, Zichong Li, Chen Liang, Weizhu Chen, and Tuo Zhao. Cosmos: A hybrid adaptive optimizer for memory-efficient training of llms. *arXiv preprint arXiv:2502.17410*, 2025.

[23] Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained lmos. *arXiv preprint arXiv:2502.07529*, 2025.

[24] Naoki Sato, Hiroki Naganuma, and Hideaki Iiduka. Analysis of muon's convergence and critical batch size. *arXiv preprint arXiv:2507.01598*, 2025.

[25] Ishaan Shah, Anthony M Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, Darsh J Shah, et al. Practical efficiency of muon for pretraining. *arXiv preprint arXiv:2505.02222*, 2025.

[26] Wei Shen, Ruichuan Huang, Minhui Huang, Cong Shen, and Jiawei Zhang. On the convergence analysis of muon. *arXiv preprint arXiv:2505.23737*, 2025.

[27] Chongjie Si, Debing Zhang, and Wei Shen. AdaMuon: Adaptive muon optimizer. *arXiv preprint arXiv:2507.11005*, 2025.

[28] Matthew Streeter and H Brendan McMahan. Less regret via online conditioning. *arXiv preprint arXiv:1002.4862*, 2010.

[29] Nikhil Vyas, Depen Morwani, Rosie Zhao, Mujin Kwun, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham Kakade. Soap: Improving and stabilizing shampoo using adam. *arXiv preprint arXiv:2409.11321*, 2024.

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

[31] Dongruo Zhou, Jinghui Chen, Yuan Cao, Ziyan Yang, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.

Appendix A. Useful Lemmas

Lemma A.1 *The Lipschitz continuity of $\nabla\mathcal{L}$ in Assumption 3.1 implies*

$$\mathcal{L}(\Theta') \leq \mathcal{L}(\Theta) + \left\langle \nabla\mathcal{L}(\Theta), \Theta' - \Theta \right\rangle + \frac{L}{2}\left\|\Theta' - \Theta\right\|_2^2. \tag{3}$$

Proof For $s \in [0,1]$, define $h(s) := \mathcal{L}(\Theta + s(\Theta' - \Theta))$. Then

$$\begin{aligned}
\mathcal{L}(\Theta') - \mathcal{L}(\Theta) &= \int_0^1 h'(s)ds \\
&= \int_0^1 \left\langle \nabla\mathcal{L}(\Theta + s(\Theta' - \Theta)), \Theta' - \Theta \right\rangle ds \\
&= \left\langle \nabla\mathcal{L}(\Theta), \Theta' - \Theta \right\rangle + \int_0^1 \left\langle \nabla\mathcal{L}(\Theta + s(\Theta' - \Theta)) - \nabla\mathcal{L}(\Theta), \Theta' - \Theta \right\rangle ds \\
&\leq \left\langle \nabla\mathcal{L}(\Theta), \Theta' - \Theta \right\rangle + \int_0^1 \left\|\nabla\mathcal{L}(\Theta + s(\Theta' - \Theta)) - \nabla\mathcal{L}(\Theta)\right\|_* \left\|\Theta' - \Theta\right\|_2 ds \\
&\leq \left\langle \nabla\mathcal{L}(\Theta), \Theta' - \Theta \right\rangle + \frac{L}{2}\left\|\Theta' - \Theta\right\|_2^2.
\end{aligned}$$

■

Lemma A.2 *For arbitrary nonnegative values, $\{a_t\}_{t=1}^T$, with $a_1 > 0$, it holds that*

$$\sum_{t=1}^{T} \frac{a_t}{\sum_{\tau=1}^{t} a_\tau} \leq \ln\left(\sum_{t=1}^{T} \frac{a_t}{a_1}\right) + 1. \tag{4}$$

Proof A similar result is shown in [30, Lemma 3.2]. For completeness, we include the proof here. Write $S_t := \sum_{\tau=1}^{t} a_\tau$. We first show that

$$\frac{a_t}{S_t} \leq \ln(S_t) - \ln(S_{t-1})$$

for all $t \geq 2$. Indeed, by the Mean Value Theorem, there exists $\xi_t \in [S_{t-1}, S_t]$ such that

$$\ln(S_t) - \ln(S_{t-1}) = \frac{S_t - S_{t-1}}{\xi_t} = \frac{a_t}{\xi_t} \geq \frac{a_t}{S_t}.$$

Hence,

$$\sum_{t=1}^{T} \frac{a_t}{S_t} \leq 1 + \sum_{t=2}^{T} \left(\ln(S_t) - \ln(S_{t-1})\right) = 1 + \ln(S_T) - \ln(S_1) = 1 + \ln\left(\frac{S_T}{a_1}\right),$$

which is exactly (4). ■

Appendix B. Proof of Theorem 3.3

Proof Write

$$\alpha_t := \frac{\min\{\|G_t\|, \gamma\}}{v_t}.$$

Let $\mathbb{E}_t[\cdot] := \mathbb{E}[\cdot|\Theta_{t-1}]$ denote the conditional expectation given the previous iterates $\Theta_0, \cdots, \Theta_{t-1}$. By Lemma A.1,

$$\begin{aligned}
&\mathbb{E}_t\left[\mathcal{L}(\Theta_t) - \mathcal{L}(\Theta_{t-1})\right] \\
\leq&\mathbb{E}_t\left[-\left\langle \nabla\mathcal{L}(\Theta_{t-1}), \max\left\{\epsilon, \eta\alpha_t\right\} O_t\right\rangle\right] + \frac{L}{2}E_t\left[\max\left\{\epsilon, \eta\alpha_t\right\}^2\right] \\
=&\mathbb{E}_t\left[-\left\langle \nabla\mathcal{L}(\Theta_{t-1}) - M_t, \max\left\{\epsilon, \eta\alpha_t\right\} O_t\right\rangle\right] - \mathbb{E}_t\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|M_t\|_*\right] + \frac{L\epsilon^2}{2} + \frac{\eta^2 L}{2}E_t\left[\alpha_t^2\right] \\
\leq&\mathbb{E}_t\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1}) - M_t\|_*\right] - \mathbb{E}_t\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|M_t\|_*\right] + \frac{L\epsilon^2}{2} + \frac{\eta^2 L}{2}E_t\left[\alpha_t^2\right] \\
\leq&\mathbb{E}_t\left[2\max\left\{\epsilon, \eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1}) - M_t\|_*\right] - \mathbb{E}_t\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right] + \frac{L\epsilon^2}{2} + \frac{\eta^2 L}{2}E_t\left[\alpha_t^2\right].
\end{aligned}$$

Then by the law of total expectation,

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right] \\
\leq&\mathcal{L}(\Theta_0) - \mathcal{L}_\star + 2\sum_{t=1}^{T}\mathbb{E}\left[\max\left\{\epsilon, \eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1}) - M_t\|_*\right] + \frac{L\epsilon^2 T}{2} + \frac{\eta^2 L}{2}\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t^2\right] \\
\leq&\mathcal{L}(\Theta_0) - \mathcal{L}_\star + 2\epsilon\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*\right] + 2\eta\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t\|E_t\|_*\right] + \frac{L\epsilon^2 T}{2} + \frac{\eta^2 L}{2}\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t^2\right], \qquad (5)
\end{aligned}$$

where $E_t := M_t - \nabla\mathcal{L}(\Theta_{t-1})$.

By Cauchy-Schwarz inequality and Lemma A.2, the third term on the right side of the above inequality satisfies

$$\begin{aligned}
\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t\|E_t\|_*\right] \leq&\sum_{t=1}^{T}\sqrt{\mathbb{E}\left[\|E_t\|_*^2\right]\mathbb{E}\left[\alpha_t^2\right]} \\
\leq&\sqrt{\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*^2\right]}\sqrt{\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t^2\right]} \\
\leq&\sqrt{\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*^2\right]}\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right) + 1}. \qquad (6)
\end{aligned}$$

Now write $\tilde{E}_t := G_t - \nabla\mathcal{L}(\Theta_{t-1})$ for $t \geq 1$. Then

$$\begin{aligned}
M_{t+1} =&\mu M_t + (1-\mu)G_{t+1} \\
=&\mu\left(E_t + \nabla\mathcal{L}(\Theta_{t-1})\right) + (1-\mu)\left(\tilde{E}_{t+1} + \nabla\mathcal{L}(\Theta_t)\right) \\
=&\nabla\mathcal{L}(\Theta_t) + \mu\left(\nabla\mathcal{L}(\Theta_{t-1}) - \nabla\mathcal{L}(\Theta_t)\right) + \mu E_t + (1-\mu)\tilde{E}_{t+1}.
\end{aligned}$$

Hence, for $t \geq 0$,

$$E_{t+1} = \mu \left(\nabla \mathcal{L}(\Theta_{t-1}) - \nabla \mathcal{L}(\Theta_t) \right) + \mu E_t + (1-\mu)\tilde{E}_{t+1}.$$

A recursive formula can be derived as in the proof of [8, Theorem]:

$$E_{t+1} = \mu^t E_1 + (1-\mu) \sum_{\tau=0}^{t-1} \mu^\tau \tilde{E}_{t+1-\tau} + \mu \sum_{\tau=0}^{t-1} \mu^\tau \left(\nabla \mathcal{L}(\Theta_{t-\tau-1}) - \nabla \mathcal{L}(\Theta_{t-\tau}) \right). \tag{7}$$

Assuming a minibatch of size $b > 0$ is sampled independently at each iteration, it follows that

$$\begin{aligned}
\mathbb{E}\left[\|E_{t+1}\|_* \right] \leq & \mu^t \mathbb{E}\left[\|E_1\|_* \right] + (1-\mu)\mathbb{E}\left[\left\| \sum_{\tau=0}^{t-1} \mu^\tau \tilde{E}_{t+1-\tau} \right\|_* \right] + \mu L \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\max\{\epsilon, \eta \alpha_{t-\tau}\} \right] \\
\leq & \frac{\mu^t \kappa \sqrt{r}}{\sqrt{b}} + (1-\mu)\sqrt{r}\mathbb{E}\left[\left\| \sum_{\tau=0}^{t-1} \mu^\tau \tilde{E}_{t+1-\tau} \right\|_F \right] + \mu L \epsilon \frac{1-\mu^t}{1-\mu} + \mu \eta L \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\alpha_{t-\tau} \right] \\
\leq & \frac{\mu^t \kappa \sqrt{r}}{\sqrt{b}} + (1-\mu)\sqrt{r}\mathbb{E}\left[\left\| \sum_{\tau=0}^{t-1} \mu^\tau \tilde{E}_{t+1-\tau} \right\|_F^2 \right]^{\frac{1}{2}} + \mu L \epsilon \frac{1-\mu^t}{1-\mu} + \mu \eta L \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\alpha_{t-\tau} \right] \\
\leq & \frac{\mu^t \kappa \sqrt{r}}{\sqrt{b}} + (1-\mu)\frac{\kappa\sqrt{r}}{\sqrt{b}} \sqrt{\sum_{\tau=0}^{t-1} \mu^{2\tau}} + \mu L \epsilon \frac{1-\mu^t}{1-\mu} + \mu \eta L \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\alpha_{t-\tau} \right] \\
= & \frac{\mu^t \kappa \sqrt{r}}{\sqrt{b}} + (1-\mu)\frac{\kappa\sqrt{r}}{\sqrt{b}} \sqrt{\frac{1-\mu^{2t}}{1-\mu^2}} + \mu L \epsilon \frac{1-\mu^t}{1-\mu} + \mu \eta L \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\alpha_{t-\tau} \right].
\end{aligned}$$

Therefore,

$$\begin{aligned}
\sum_{t=0}^{T} \mathbb{E}\left[\|E_{t+1}\|_* \right] \leq & \frac{\kappa\sqrt{r}}{(1-\mu)\sqrt{b}} + T\kappa\sqrt{\frac{r(1-\mu)}{b}} + \frac{T\mu L\epsilon}{1-\mu} + \mu\eta L \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} \mu^\tau \mathbb{E}\left[\alpha_{t-\tau} \right] \\
\leq & \frac{\kappa\sqrt{r}}{(1-\mu)\sqrt{b}} + T\kappa\sqrt{\frac{r(1-\mu)}{b}} + \frac{T\mu L\epsilon}{1-\mu} + \frac{\mu\eta L}{1-\mu} \sum_{t=1}^{T} \mathbb{E}\left[\alpha_t \right] \\
\leq & \frac{\kappa\sqrt{r}}{(1-\mu)\sqrt{b}} + T\kappa\sqrt{\frac{r(1-\mu)}{b}} + \frac{T\mu L\epsilon}{1-\mu} + \frac{\mu\eta L}{1-\mu}\sqrt{T} \left(\sum_{t=1}^{T} \mathbb{E}\left[\alpha_t^2 \right] \right)^{\frac{1}{2}} \\
\leq & \frac{\kappa\sqrt{r}}{(1-\mu)\sqrt{b}} + T\kappa\sqrt{\frac{r(1-\mu)}{b}} + \frac{T\mu L\epsilon}{1-\mu} + \frac{\mu\eta L}{1-\mu}\sqrt{T} \left(\ln\left(\frac{\gamma^2}{v_0^2} T \right) + 1 \right).
\end{aligned} \tag{8}$$

Also, by (7),

$$\begin{aligned}\mathbb{E}\left[\|E_{t+1}\|_F^2\right] =&\mu^{2t}\mathbb{E}\left[\|E_1\|_F^2\right] + \mu^{t+1}\sum_{\tau=0}^{t-1}\mu^\tau\mathbb{E}\left[\langle E_1, \nabla\mathcal{L}(\Theta_{t-\tau-1}) - \nabla\mathcal{L}(\Theta_{t-\tau})\rangle\right]\\ &+ (1-\mu)^2\sum_{\tau=0}^{t-1}\mu^{2\tau}\mathbb{E}\left[\left\|\tilde{E}_{t+1-\tau}\right\|_F^2\right] + \mu^2\mathbb{E}\left[\left\|\sum_{\tau=0}^{t-1}\mu^\tau\left(\nabla\mathcal{L}(\Theta_{t-\tau-1}) - \nabla\mathcal{L}(\Theta_{t-\tau})\right)\right\|_F^2\right]\\ \leq&\frac{\mu^{2t}\kappa^2}{b} + L\mu^{t+1}\sum_{\tau=0}^{t-1}\mu^\tau\mathbb{E}\left[\|E_1\|_F\max\{\epsilon, \eta\alpha_{t-\tau}\}\right]\\ &+ (1-\mu)^2\frac{\kappa^2}{b}\sum_{\tau=0}^{t-1}\mu^{2\tau} + \mu^2\mathbb{E}\left[\left(\sum_{\tau=0}^{t-1}\mu^\tau\left\|\nabla\mathcal{L}(\Theta_{t-\tau-1}) - \nabla\mathcal{L}(\Theta_{t-\tau})\right\|_F\right)^2\right]\end{aligned}$$

Then by Cauchy-Schwarz inequality,

$$\begin{aligned}\mathbb{E}\left[\|E_{t+1}\|_F^2\right] \leq&\frac{\mu^{2t}\kappa^2}{b} + \frac{L\kappa\epsilon\mu^{t+1}}{\sqrt{b}}\frac{1-\mu^t}{1-\mu} + \frac{\eta L\kappa\mu^{t+1}}{\sqrt{b}}\sum_{\tau=0}^{t-1}\mu^\tau\sqrt{\mathbb{E}\left[\alpha_{t-\tau}^2\right]} + \frac{\kappa^2}{b}\frac{(1-\mu)(1-\mu^{2t})}{1+\mu}\\ &+ L^2\mu^2\epsilon^2\left(\frac{1-\mu^{2t}}{1-\mu^2}\right) + \eta^2L^2\mu^2\left(\sum_{\tau=0}^{t-1}\mu^{2\tau}\right)\mathbb{E}\left[\sum_{\tau=0}^{t-1}\alpha_{t-\tau}^2\right]\\ \leq&\frac{\mu^{2t}\kappa^2}{b} + \frac{L\kappa\epsilon\mu^{t+1}}{\sqrt{b}}\frac{1-\mu^t}{1-\mu} + \frac{\eta L\kappa\mu^{t+1}}{\sqrt{b}}\left(\sqrt{\frac{1-\mu^{2t}}{1-\mu^2}}\right)\sqrt{\mathbb{E}\left[\sum_{\tau=0}^{t-1}\alpha_{t-\tau}^2\right]} + \frac{\kappa^2}{b}\frac{(1-\mu)(1-\mu^{2t})}{1+\mu}\\ &+ L^2\mu^2\epsilon^2\left(\frac{1-\mu^{2t}}{1-\mu^2}\right) + \eta^2L^2\mu^2\left(\frac{1-\mu^{2t}}{1-\mu^2}\right)\mathbb{E}\left[\sum_{\tau=0}^{t-1}\alpha_{t-\tau}^2\right]. \qquad (9)\end{aligned}$$

Applying [30, Lemma 3.2] gives

$$\sum_{t=1}^{T}\sum_{\tau=0}^{t-1}\alpha_{t-\tau}^2 = \sum_{t=1}^{T}\sum_{\tau=1}^{t}\alpha_\tau^2 = \sum_{t=1}^{T}(T-t+1)\alpha_t^2 \leq T\sum_{t=1}^{T}\alpha_t^2 \leq T\ln\left(\left(\frac{\gamma^2}{v_0^2}T\right)+1\right).$$

Also note that

$$\sum_{t=1}^{T}\frac{(1-\mu)(1-\mu^{2t})}{1+\mu} = \frac{1-\mu}{1+\mu}T - \frac{\mu^2}{(1+\mu)^2}(1-\mu^{2T}) \leq \frac{1-\mu}{1+\mu}T.$$

Therefore, it follows from (9) that

$$\begin{aligned}\sum_{t=0}^{T}\mathbb{E}\left[\|E_{t+1}\|_F^2\right] \leq & \frac{1}{1-\mu^2}\frac{\kappa^2}{b}+\frac{L\kappa\epsilon}{(1-\mu)^2\sqrt{b}}+\frac{1-\mu}{1+\mu}\frac{\kappa^2 T}{b}+\frac{\eta^2L^2\mu^2}{1-\mu^2}T\left(\ln\left(\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)\right)\\ &+\frac{L^2\mu^2\epsilon^2T}{1-\mu^2}+\frac{\eta L\kappa}{\sqrt{b}\sqrt{1-\mu^2}}\sqrt{\left(\sum_{t=1}^{T}\mu^{2t+2}\right)\left(\sum_{t=1}^{T}\sum_{\tau=0}^{t-1}\alpha_{t-\tau}^2\right)}\\ \leq & \frac{1}{1-\mu^2}\frac{\kappa^2}{b}+\frac{L\kappa\epsilon}{(1-\mu)^2\sqrt{b}}+\frac{1-\mu}{1+\mu}\frac{\kappa^2 T}{b}+\frac{\eta^2L^2\mu^2}{1-\mu^2}T\left(\ln\left(\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)\right)\\ &+\frac{L^2\mu^2\epsilon^2T}{1-\mu^2}+\frac{\mu\eta L\kappa}{(1-\mu^2)\sqrt{b}}\sqrt{T\ln\left(\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)}. \end{aligned} \tag{10}$$

Then by (5) and (6),

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}\left[\max\left\{\epsilon,\eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\\ \leq&\mathcal{L}(\Theta_0)-\mathcal{L}_\star+2\eta\sqrt{\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*^2\right]}\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1}+2\epsilon\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*\right]+\frac{L}{2}\epsilon^2T+\frac{\eta^2L}{2}\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t^2\right]\\ \leq&\mathcal{L}(\Theta_0)-\mathcal{L}_\star+2\eta\sqrt{r\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_F^2\right]}\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1}+2\epsilon\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*\right]+\frac{L\epsilon^2T}{2}+\frac{\eta^2L}{2}\sum_{t=1}^{T}\mathbb{E}\left[\alpha_t^2\right],\end{aligned}$$

where $r:=\min\{m,n\}$. Combining the above with (8) and (10) gives

$$\begin{aligned}&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\\ \leq&\frac{1}{T\epsilon}\sum_{t=1}^{T}\mathbb{E}\left[\max\left\{\epsilon,\eta\alpha_t\right\}\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\\ \leq&\frac{\Delta}{\epsilon T}+\frac{L\epsilon}{2}+\frac{\eta^2L}{2\epsilon T}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)+\frac{2}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_*\right]+\frac{2\eta}{T\epsilon}\sqrt{r\sum_{t=1}^{T}\mathbb{E}\left[\|E_t\|_F^2\right]}\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1}\\ \leq&\frac{\Delta}{\epsilon T}+\frac{L\epsilon}{2}+\frac{\eta^2L}{2\epsilon T}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)+2\left(\frac{\kappa\sqrt{r}}{(1-\mu)T\sqrt{b}}+\kappa\sqrt{\frac{r(1-\mu)}{b}}+\frac{\mu L\epsilon}{1-\mu}\right.\\ &\left.+\frac{\mu\eta L}{(1-\mu)\sqrt{T}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)\right)+\frac{2\eta\sqrt{r}}{\sqrt{T}\epsilon}\left(\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1}\right)\left(\frac{\sqrt{L\kappa\epsilon}}{(1-\mu)\sqrt{T}b^{1/4}}+\frac{L\mu\epsilon}{\sqrt{1-\mu^2}}\right.\\ &\left.+\frac{\kappa}{\sqrt{bT}\sqrt{1-\mu^2}}+\sqrt{\frac{1-\mu}{1+\mu}}\frac{\kappa}{\sqrt{b}}+\frac{\eta L\mu}{\sqrt{1-\mu^2}}\sqrt{\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1}+\frac{1}{(bT)^{1/4}}\sqrt{\frac{\mu\eta L\kappa}{1-\mu^2}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)^{1/4}\right).\end{aligned} \tag{11}$$

In particular, if choosing $b=1$, $\epsilon=T^{-\frac{3}{4}}$, $1-\mu=T^{-\frac{1}{2}}$, and $\eta=T^{-\left(\frac{3}{8}+q\right)}$ for arbitrary $q>0$, then by (11),

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\\
\leq&\frac{\Delta}{T^{1/4}}+\frac{L}{2T^{3/4}}+\frac{L}{2T^{1+2q}}\left(\ln T+1\right)+2\left(\frac{\kappa\sqrt{r}}{T^{1/2}}+\frac{\kappa\sqrt{r}}{T^{1/4}}+\frac{L}{T^{1/4}}+\frac{L}{T^{3/8+q}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)\right)\\
&+2\left(\sqrt{\ln T+1}\right)\left(\frac{\sqrt{L\kappa r}}{T^{1/2+q}}+\frac{L\sqrt{r}}{T^{5/8+q}}+\frac{2\kappa\sqrt{r}}{T^{3/8+q}}+\frac{L\sqrt{r}}{T^{1/4+q}}\sqrt{\ln T+1}+\frac{\sqrt{rL\kappa}}{T^{5/16+3q/2}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)^{\frac{1}{4}}\right)\\
=&\mathcal{O}\left(\frac{\Delta+\kappa\sqrt{r}+L}{T^{1/4}}+\frac{L\sqrt{r}}{T^{1/4+q}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right)\right),
\end{aligned}$$

for large T. The proof is thus completed. ■

Appendix C. Proof of Theorem 3.4

Proof By Lemma A.1,

$$\begin{aligned}
\mathcal{L}(\Theta_t)-\mathcal{L}(\Theta_{t-1})\leq&-\left\langle\nabla\mathcal{L}(\Theta_{t-1}),\max\{\epsilon,\eta\alpha_t\}O_t\right\rangle+\frac{L}{2}\max\{\epsilon^2,\eta^2\alpha_t^2\}\\
\leq&-\max\{\epsilon,\eta\alpha_t\}\left\|\nabla\mathcal{L}(\Theta_{t-1})\right\|_*+\frac{L\epsilon^2}{2}+\frac{\eta^2L}{2}\alpha_t^2.
\end{aligned}$$

Hence, by Lemma A.2,

$$\sum_{t=1}^{T}\max\{\epsilon,\eta\alpha_t\}\left\|\nabla\mathcal{L}(\Theta_{t-1})\right\|_*\leq\Delta+\frac{L\epsilon^2T}{2}+\frac{\eta^2L}{2}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right).$$

It follows that

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\left\|\nabla\mathcal{L}(\Theta_{t-1})\right\|_*\leq&\frac{1}{T\epsilon}\sum_{t=1}^{T}\max\{\epsilon,\eta\alpha_t\}\left\|\nabla\mathcal{L}(\Theta_{t-1})\right\|_*\\
\leq&\frac{\Delta}{T\epsilon}+\frac{L\epsilon}{2}+\frac{\eta^2L}{2T\epsilon}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right).
\end{aligned}$$

In particular, if choosing $\epsilon=T^{-\frac{1}{2}}$ and $\eta=T^{-q}$ for arbitrary $q>0$, then

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\leq\frac{2\Delta+L}{2\sqrt{T}}+\frac{L}{2T^{2q+1/2}}\left(\ln\left(\frac{\gamma^2}{v_0^2}T\right)+1\right).$$

For large $T>0$,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla\mathcal{L}(\Theta_{t-1})\|_*\right]\leq\mathcal{O}\left(\frac{\Delta+L}{\sqrt{T}}\right)$$

The proof is thus completed. ■