ADOPT: Modified Adam Can Converge with Any β_2 with the Optimal Rate

Shohei Taniguchi
The University of Tokyo
taniguchi@weblab.t.u-tokyo.ac.jp

Keno Harada
The University of Tokyo
keno.harada@weblab.t.u-tokyo.ac.jp

Gouki Minegishi
The University of Tokyo
minegishi@weblab.t.u-tokyo.ac.jp

Yuta Oshima
The University of Tokyo
yuta.oshima@weblab.t.u-tokyo.ac.jp

Seong Cheol Jeong
The University of Tokyo
jeong@weblab.t.u-tokyo.ac.jp

Go Nagahara
The University of Tokyo
nagaharago@weblab.t.u-tokyo.ac.jp

Tomoshi Iiyama
The University of Tokyo
iiyama@weblab.t.u-tokyo.ac.jp

Masahiro Suzuki
The University of Tokyo
masa@weblab.t.u-tokyo.ac.jp

Yusuke Iwasawa
The University of Tokyo
iwasawa@weblab.t.u-tokyo.ac.jp

Yutaka Matsuo
The University of Tokyo
matsuo@weblab.t.u-tokyo.ac.jp

Abstract

Adam is one of the most popular optimization algorithms in deep learning. However, it is known that Adam does not converge in theory unless choosing a hyperparameter, i.e., β_2, in a problem-dependent manner. There have been many attempts to fix the non-convergence (e.g., AMSGrad), but they require an impractical assumption that the gradient noise is uniformly bounded. In this paper, we propose a new adaptive gradient method named ADOPT, which achieves the optimal convergence rate of $\mathcal{O}(1/\sqrt{T})$ with any choice of β_2 without depending on the bounded noise assumption. ADOPT addresses the non-convergence issue of Adam by removing the current gradient from the second moment estimate and changing the order of the momentum update and the normalization by the second moment estimate. We also conduct intensive numerical experiments, and verify that our ADOPT achieves superior results compared to Adam and its variants across a wide range of tasks, including image classification, generative modeling, natural language processing, and deep reinforcement learning. The implementation is available at `https://github.com/iShohei220/adopt`.

1 Introduction

Stochastic optimization algorithms, such as stochastic gradient descent (SGD), play a central role in deep learning. In particular, adaptive gradient methods based on exponential moving averages, such as Adam [Kingma and Ba, 2014], are widely used in practice. Despite the empirical success, it is

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

known that Adam does not converge in theory in general cases. For example, Reddi et al. [2018] show that Adam fails to converge to a correct solution in a simple example where the objective function at time t is given as:

$$f_t(\theta) = \begin{cases} C\theta, & \text{for } t \bmod 3 = 1 \\ -\theta, & \text{otherwise,} \end{cases} \tag{1}$$

where $C > 2$ and $\theta \in [-1, 1]$. In this online optimization setting, Adam converges to a wrong solution (i.e., $\theta = 1$) instead of the true solution (i.e., $\theta = -1$) especially when the hyperparameter β_2 is set to a small value. There have been several attempts to fix the non-convergent behavior of Adam [Reddi et al., 2018, Zou et al., 2019]. For example, AMSGrad [Reddi et al., 2018] ensures the convergence for online convex optimization by making slight modifications to the Adam algorithm. Subsequent studies [Chen et al., 2019, Zhou et al., 2018] show that AMSGrad also converges to a stationary point for smooth nonconvex stochastic optimization problems. However, the convergence proofs rely on the assumption that the gradient noise is uniformly bounded. This assumption is stronger than the one used for the analysis of vanilla SGD [Ghadimi and Lan, 2013, Bertsekas and Tsitsiklis, 2000, Khaled and Richtárik, 2023], where the gradient *variance* is assumed to be uniformly bounded. In fact, the bounded noise assumption is often violated in practice. For example, when Gaussian noise is used in the gradient estimation (e.g., variational autoencoders [Kingma and Welling, 2014] and diffusion models [Ho et al., 2020, Song et al., 2021]), the stochastic gradient is no longer bounded.

Concurrently, Zhou et al. [2019] analyze the non-convergence of Adam in the problem described in Eq. (1) from the perspective of the correlation between the current gradient and the second moment estimate based on the exponential moving average. Specifically, they show that the non-convergence problem can be resolved by excluding the gradient of some recent steps from the calculation of the second moment estimate. Based on the analysis, they propose AdaShift, another variant of Adam. However, their theoretical analysis is limited to a single online convex problem described in Eq. (1), and the convergence of AdaShift for general nonconvex problems is unclear.

More recently, some works have demonstrated that Adam can converge by choosing β_2 in a problem-dependent manner [Shi et al., 2020, Zhang et al., 2022, Wang et al., 2022, Li et al., 2023, Wang et al., 2023]. However, tuning β_2 for each specific problem is troublesome; hence developing algorithms with the problem-independent convergence guarantee is still important to safely apply adaptive gradient methods to a wide range of machine learning problems.

In this paper, we propose an alternative approach to addressing the non-convergence problem of Adam without relying on the choice of β_2 or strong assumptions such as the bounded noise assumption. To derive our algorithm, we first examine the case without momentum, analyzing the convergence bound of RMSprop for general smooth nonconvex optimization problems. Through the analysis, we uncover the fundamental cause of non-convergence, which stems from the correlation between the second moment estimate and the current gradient. This finding aligns with the results demonstrated by Zhou et al. [2019] for online convex optimization. This correlation can be easily eliminated by excluding the current gradient from the second moment estimate.

Subsequently, we extend our findings to the case where momentum is incorporated, as in Adam, and discover that the Adam-style momentum also contributes to non-convergence. To address it, we propose to change the order of the momentum update and the normalization by the second moment estimate. With this small adjustment, we successfully eliminate the non-convergence problem of Adam without relying on a specific hyperparameter choice and the bounded noise assumption. We provide theoretical evidence demonstrating that our derived algorithm, named ADOPT, can achieve convergence with the optimal rate of $\mathcal{O}(1/\sqrt{T})$ for smooth nonconvex optimization.

In our experiments, we begin by assessing the performance of ADOPT in a toy example where Adam typically fails to converge depending on the choice of β_2. This toy example is an extension of the one presented in Eq. (1) by Reddi et al. [2018], but we consider a scenario where AMSGrad is also hard to converge due to the dependence on the bounded noise assumption. Our results demonstrate that ADOPT rapidly converges to the solution, while Adam fails to converge, and AMSGrad exhibits extremely slow convergence. Next, we conduct an experiment using a simple multi-layer perceptron on the MNIST classification task to evaluate the performance of ADOPT in nonconvex optimization. Our findings indicate that ADOPT outperforms existing adaptive gradient methods, including Adam, AMSGrad, and AdaShift. Finally, we evaluate the performance of ADOPT in various practical applications, such as image classification of CIFAR-10 and ImageNet using ResNet [He et al., 2016] and SwinTransformer [Liu et al., 2021], training of deep generative models (NVAE), fine-tuning of language models (LLaMA), and deep reinforcement learning for continuous control. Our empirical results demonstrate that ADOPT achieves superior results over existing algorithms (e.g., Adam) in these practical applications.

2 Preliminary

2.1 Problem Definition

We consider the minimization of the objective function $f : \mathbb{R}^D \to \mathbb{R}$ with respect to the parameter $\boldsymbol{\theta} \in \mathbb{R}^D$. In this context, we focus on first-order stochastic optimization methods, where only the stochastic gradient $\boldsymbol{g}$ is accessible. As the objective f can be nonconvex, the goal is to find a stationary point where $\nabla f(\boldsymbol{\theta}) = 0$ [Blair, 1985, Vavasis, 1995]. In order to analyze the convergence behavior of stochastic optimization algorithms, the following assumptions are commonly employed in the literature:

Assumption 2.1. *The objective function $f(\boldsymbol{\theta})$ is lower-bounded, i.e., $f(\boldsymbol{\theta}) \geq f_{\inf} > -\infty$ for all $\boldsymbol{\theta}$.*

Assumption 2.2. *The stochastic gradient $\boldsymbol{g}_t$ is an unbiased estimator of the objective $f(\boldsymbol{\theta}_{t-1})$, i.e., $\mathbb{E}[\boldsymbol{g}_t] = \nabla f(\boldsymbol{\theta}_{t-1})$ for all $t \geq 1$.*

Assumption 2.3. *The objective function is L-smooth on $\mathbb{R}^D$, i.e., there exists a constant $L > 0$ such that $\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\| \leq L\|\boldsymbol{x} - \boldsymbol{y}\|$ for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^D$.*

Assumption 2.4. *Variance of the stochastic gradient is uniformly bounded , i.e., there exists a constant $\sigma > 0$ such that $\mathbb{E}[\|\boldsymbol{g}_t - \nabla f(\boldsymbol{\theta}_{t-1})\|^2] \leq \sigma^2$.*

For the analysis of adaptive gradient methods (e.g., Adam and AdaGrad), many of previous works [Défossez et al., 2022, Li and Orabona, 2019, Ward et al., 2020, Zou et al., 2018] use a little stronger assumption instead of Assumption 2.4 for ease of proofs:

Assumption 2.5. *The stochastic gradient has a finite second moment, i.e., there exists a constant $G > 0$ such that $\mathbb{E}[\|\boldsymbol{g}_t\|^2] \leq G^2$.*

Assumption 2.5 requires that the true gradient ∇f is also uniformly bounded in addition to the variance of the stochastic gradient $\boldsymbol{g}$. Moreover, the convergence proof of AMSGrad tends to rely on an even stronger assumption as follows [Chen et al., 2019, Zhou et al., 2018].

Assumption 2.6. *The stochastic gradient is uniformly upper-bounded, i.e., there exists a constant $G > 0$ such that $\|\boldsymbol{g}_t\| \leq G$.*

In Assumption 2.6, the gradient noise $\xi_t := \boldsymbol{g}_t - \nabla f$ is assumed to be bounded almost surely in addition to the true graidient ∇f. Note that when Assumption 2.6 holds, Assumption 2.5 is automatically satisfied; hence, Assumption 2.6 is a stronger assumption compared to Assumption 2.5. In this paper, we adopt Assumptions 2.1, 2.2, 2.3 and 2.5 for analysis, because one of our motivations is to address the omission of Assumption 2.6. In the analysis, we derive the upper bound of $\min_t\{\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_t))\|^{4/3}]^{3/2}\}$ to investigate the convergence rate of the stochastic optimization algorithms, which is commonly performed in the literature [Défossez et al., 2022, Zou et al., 2019].

2.2 Review of Stochastic Optimization Algorithms for Nonconvex Objectives

The convergence of the vanilla SGD have been studied extensively in previous works. For smooth nonconvex functions, Ghadimi and Lan [2013] showed that SGD with a constant learning rate converges with an $\mathcal{O}(1/\sqrt{T})$ rate under Assumptions 2.1-2.4 by setting $\alpha_t = \alpha = \Theta(1/\sqrt{T})$, where α_t is a learning rate at the t-th step, and T is a total number of parameter updates. This convergence rate is known to be minimax optimal up to a constant [Drori and Shamir, 2020]. For the diminishing learning rate scheme, the convergence bound of $\mathcal{O}(\log T/\sqrt{T})$ is well-known for $\alpha_t = \alpha/\sqrt{t}$ [Ghadimi and Lan, 2013]. Recently, Wang et al. [2021] have proved that SGD with $\alpha_t = \alpha/\sqrt{t}$ can also achieve the optimal rate $\mathcal{O}(1/\sqrt{T})$ by additionally assuming that the objective f is upper-bounded.

While the vanilla SGD is still one of the most popular choices for stochastic optimization, adaptive gradient methods are dominantly used especially for deep learning. In adaptive gradient methods, the parameter $\boldsymbol{\theta}$ is updated additionally using the second moment estimate $\boldsymbol{v}_t$ in the following form:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \alpha_t \frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}}, \tag{2}$$

where ϵ is a small positive constant. The division between vectors is applied in an element-wise manner, and the addition between a vector $\boldsymbol{a}$ and a scalar b is defined as $(\boldsymbol{a} + b)_i := a_i + b$. In AdaGrad [Duchi et al., 2011], $\boldsymbol{v}_t$ is defined as $\boldsymbol{v}_0 = \mathbf{0}$ and $\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \boldsymbol{g}_t \odot \boldsymbol{g}_t$. In RMSprop [Hinton et al., 2012], an exponential moving average is substituted for the simple summation, i.e., $\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1 - \beta_2) \boldsymbol{g}_t \odot \boldsymbol{g}_t$, where $0 \leq \beta_2 < 1$. Adam [Kingma and Ba, 2014] uses momentum in addition to the second moment estimate to accelerate the convergence as follows:

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1 - \beta_1)\, \boldsymbol{g}_t, \tag{3}$$

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \alpha_t \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}}, \tag{4}$$

where $\boldsymbol{m}_0 = \mathbf{0}$. Here, we omit the bias correction technique used in the original paper for clarity. Unfortunately, RMSprop and Adam are not guaranteed to converge even in a simple convex optimization problem as demonstrated by Reddi et al. [2018], whereas AdaGrad with a constant learning rate is known to converge with an $\mathcal{O}(\log T / \sqrt{T})$ rate under Assupmtions 2.1-2.3 and 2.5 for smooth nonconvex cases [Li and Orabona, 2019, Ward et al., 2020, Zou et al., 2018, Chen et al., 2019, Défossez et al., 2022]. Although the convergence of Adam can be assured by choosing β_2 in a problem-dependent manner [Shi et al., 2020, Zhang et al., 2022, Wang et al., 2022, Li et al., 2023, Wang et al., 2023], it is difficult to know the proper choice of β_2 for each problem before training.

To fix the non-convergence of Adam without depending on β_2, some researchers have proposed variants of Adam. Reddi et al. [2018] proposed AMSGrad, which substitute $\hat{\boldsymbol{v}}_t$ for $\boldsymbol{v}$ in Eq. (3), where $\hat{\boldsymbol{v}}_0 = \mathbf{0}$ and $\hat{\boldsymbol{v}}_t = \max\{\hat{\boldsymbol{v}}_{t-1}, \boldsymbol{v}_t\}$. The idea behind AMSGrad is that the scaling factor $\sqrt{\hat{\boldsymbol{v}}_t + \epsilon^2}$ should be non-decreasing to ensure the convergence. After Reddi et al. [2018] originally proved the convergence of AMSGrad for online convex optimization, Chen et al. [2019] showed that AMSGrad with $\alpha_t = \alpha / \sqrt{t}$ converges with $\mathcal{O}(\log T / \sqrt{T})$ for nonconvex settings. Zhou et al. [2018] also analyzed the convergence of AMSGrad for nonconvex optimization, and derived the convergence rate of $\mathcal{O}(1/\sqrt{T})$ for a constant learning rate of $\alpha_t = \alpha = \Theta(1/\sqrt{T})$. However, their results depend on Assumption 2.6, which is often violated in practice. For example, variational autoencoders [Kingma and Welling, 2014] and diffusion models [Ho et al., 2020, Song et al., 2021] are typical examples in which Assumption 2.6 does not hold because they utilize unbounded Gaussian noise in the gradient estimation. The cause of requirement for Assumption 2.6 is the max operation in the definition of $\hat{\boldsymbol{v}}_t$. Since the max operation is convex, $\mathbb{E}[\hat{\boldsymbol{v}}_t] \leq \max_t\{\mathbb{E}[\boldsymbol{v}_t]\}$ does not hold; hence Assumption 2.6 is required to upper-bound $\mathbb{E}[\hat{\boldsymbol{v}}_t]$ in their proofs.

Zhou et al. [2019] also tried to fix the non-convergent behavior of Adam. Their proposed AdaShift uses $\boldsymbol{v}_{t-n}$ instead of $\boldsymbol{v}_t$ for the second moment estimate, and calculate the momentum using the latest n gradients as follows:

$$\boldsymbol{m}_t = \frac{\sum_{k=0}^{n-1} \beta_1^k \boldsymbol{g}_{t-k}}{\sum_{k=0}^{n-1} \beta_1^k}, \tag{5}$$

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \alpha_t \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_{t-n} + \epsilon^2}}. \tag{6}$$

In the original paper, some additional techniques (e.g., the block-wise adaptive learning rate) are used, but we omit them for clarity here. Though they give theoretical analysis for a single online convex example, any convergence bounds are not provided for nonconvex cases. More detailed discussion on existing analyses is provided in Appendix A.

3 Analysis: Cause of Non-convergence of Adam and How to Fix It

In this section, to derive an algorithm that can converge with any β_2 without Assumption 2.6, we analyze the cause of non-convergence of Adam, and discuss how it can be eliminated. To start from a simple case, we first analyze the case without momentum. Subsequently, we extend it to the case with momentum and provide a way to fix the convergence issue of Adam.

3.1 Case without Momentum

We first analyze the convergence of RMSprop, which corresponds to the no-momentum case of Adam when we omit the bias correction. For RMSprop, we derive the following convergence bound.

Theorem 3.1. *Under Assumptions 2.1-2.3 and 2.5, the following holds for the RMSprop with a constant learning rate $\alpha_t = \alpha$:*

$$\min_{t=1,\ldots,T}\left\{\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1}))\|^{4/3}\right]^{3/2}\right\} \leq C_1\left(\frac{f_0 - f_{\inf}}{\alpha T} + \frac{C_2}{T}\log\left(1+\frac{G^2}{\epsilon^2}\right) - C_2\log\beta_2\right), \quad (7)$$

where $C_1 = 2\sqrt{G^2+\epsilon^2}$, $C_2 = \frac{\alpha D L}{2(1-\beta_2)} + \frac{2DG}{\sqrt{1-\beta_2}}$, and $f_0 = f(\boldsymbol{\theta}_0)$.

Sketch of proof. By Assumption 2.3, the following holds:

$$\mathbb{E}\left[f(\boldsymbol{\theta}_t)\right] \leq \mathbb{E}\left[f(\boldsymbol{\theta}_{t-1}) + \frac{\alpha^2 L}{2}\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2 - \alpha\nabla f(\boldsymbol{\theta}_{t-1})^\top\left(\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right)\right] \quad (8)$$

Applying Lemmas G.4 and G.6 in the appendix to this, the following inequality is derived:

$$\begin{aligned}
&\mathbb{E}\left[f(\boldsymbol{\theta}_t)\right] \\
&\leq \mathbb{E}\left[f(\boldsymbol{\theta}_{t-1}) + \left(\frac{\alpha^2 L}{2} + 2\alpha G\sqrt{1-\beta_2}\right)\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2 - \frac{\alpha}{2}\nabla f(\boldsymbol{\theta}_{t-1})^\top\left(\frac{\boldsymbol{g}_t}{\sqrt{\tilde{\boldsymbol{v}}_t+\epsilon^2}}\right)\right] \quad (9)\\
&\leq \mathbb{E}\left[f(\boldsymbol{\theta}_{t-1}) + \left(\frac{\alpha^2 L}{2} + 2\alpha G\sqrt{1-\beta_2}\right)\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2\right] - \frac{\alpha}{2}\frac{\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}}, \quad (10)
\end{aligned}$$

where $\tilde{\boldsymbol{v}}_t = \beta_2\boldsymbol{v}_{t-1} + (1-\beta_2)\mathbb{E}[\boldsymbol{g}_t \odot \boldsymbol{g}_t]$. Telescoping this for $t = 1, \ldots, T$ and rearranging the terms, we have

$$\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2} \leq C_1\left(\frac{f(\boldsymbol{\theta}_0) - f_{\inf}}{\alpha} + C_2\log\left(\frac{G^2+\epsilon^2}{\beta_2^T\epsilon^2}\right)\right), \quad (11)$$

where the last inequality holds due to Assumption 2.1 and Lemma G.5. Therefore, the bound in Eq. (7) is derived using $\min_{t=1,\ldots,T}\{\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_{t-1}))\|^{4/3}]^{3/2}\} \leq \sum_{t=1}^{T}\mathbb{E}[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}]^{3/2}/T$. □

A detailed proof is provided in the appendix. When the learning rate α is chosen so that $\alpha = \Theta(1/\sqrt{T})$, the first and second terms on the right hand side of Eq. (7) converge with $\mathcal{O}(1/\sqrt{T})$ and $\mathcal{O}(1/T)$ rates, respectively. However, the last term includes a constant factor in terms of T, which represents the non-convergent behavior of RMSprop in the smooth nonconvex setting. More precisely, RMSprop is guaranteed to converge only to a bounded region around a stationary point, and the size of the bounded region depends on the hyperparameter β_2 and the problem-dependent factors D, G, and L. Therefore, we need to choose β_2 dependently on each problem to make the bounded region adequately small. Since $\lim_{\beta_2\to 1}\log\beta_2/\sqrt{1-\beta_2} = 0$, the size of the bounded region can be made small by setting β_2 to a value close to 1, which aligns with practical observations. However, how close to 1 it should be relies on the problem-dependent factors, which cannot be observed in advance. This result is consistent with recent results of convergence analyses of Adam and RMSprop [Shi et al., 2020, Zhang et al., 2022].

As can be seen from Eqs. (8) and (9), the constant term in Eq. (7) is derived from the last term of Eq. (8). Because $\boldsymbol{g}_t$ and $\boldsymbol{v}_t$ are not statistically independent, this term is first decomposed as in Eq. (9). After the decomposition, $\boldsymbol{g}_t$ and $\tilde{\boldsymbol{v}}_t$ is now conditionally independent given $\boldsymbol{g}_0, \ldots, \boldsymbol{g}_{t-1}$, so Eq. (10) is derived using the following fact:

$$\mathbb{E}\left[\frac{\boldsymbol{g}_t}{\sqrt{\tilde{\boldsymbol{v}}_t+\epsilon^2}}\right] = \mathbb{E}\left[\frac{\nabla f(\boldsymbol{\theta}_{t-1})}{\sqrt{\tilde{\boldsymbol{v}}_t+\epsilon^2}}\right]. \quad (12)$$

This indicates that, if the second moment estimate $\boldsymbol{v}_t$ is designed to be conditionally independent to $\boldsymbol{g}_t$, the constant term in the convergence bound will be removed, because the second term of Eq. (8)

Algorithm 1 ADOPT algorithm

```
Require: Learning rate $\{\alpha_t\}$, initial parameter $\boldsymbol{\theta}_0$
Require: Exponential decay rate $0 \leq \beta_1 < 1, 0 \leq \beta_2 \leq 1$, small constant $\epsilon > 0$
  $\boldsymbol{v}_0 \leftarrow \boldsymbol{g}_0 \odot \boldsymbol{g}_0, \boldsymbol{m}_1 \leftarrow \boldsymbol{g}_1 / \max\left\{\sqrt{\boldsymbol{v}_0}, \epsilon\right\}$
  for $t = 1$ to $T$ do
    $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \alpha_t \boldsymbol{m}_t$
    $\boldsymbol{v}_t \leftarrow \beta_2 \cdot \boldsymbol{v}_{t-1} + (1 - \beta_2)\, \boldsymbol{g}_t \odot \boldsymbol{g}_t$
    $\boldsymbol{m}_{t+1} \leftarrow \beta_1 \cdot \boldsymbol{m}_t + (1 - \beta_1)\, \frac{\boldsymbol{g}_{t+1}}{\max\{\sqrt{\boldsymbol{v}_t}, \epsilon\}}$
  end for
  return $\{\boldsymbol{\theta}_t\}_{t=1}^{T}$
```

can be directly lower-bounded without the decomposition. A simple way to achieve the conditional independence is to substitute $\boldsymbol{v}_{t-1}$ for $\boldsymbol{v}_t$ as a second moment estimate, because $\boldsymbol{v}_{t-1}$ does not have information about $\boldsymbol{g}_t$. This solution is similar to AdaShift, in which $\boldsymbol{v}_{t-n}$ is substituted for $\boldsymbol{v}_t$ as described in Eq. (5). In fact, the modified version of RMSprop is identical to AdaShift with $n = 1$ and $\beta_1 = 0$ except for the additional techniques (e.g., the block-wise adaptive learning rate).

3.2 Case with Momentum

As we have described, RMSprop can be modified to be convergent by removing the current gradient $\boldsymbol{g}_t$ from the second moment estimate $\boldsymbol{v}_t$. However, when we combine adaptive gradient methods with momentum like Adam, the convergence analysis becomes more complicated. Unfortunately, when Adam-style momentum in Eq. (3) is applied, the algorithm does not converge in general even when using $\boldsymbol{v}_{t-1}$ as a second moment estimate instead of $\boldsymbol{v}_t$. This is because the momentum $\boldsymbol{m}_t$ contains all history of the past gradients $\boldsymbol{g}_0, \ldots, \boldsymbol{g}_t$; hence the second moment estimate always correlates with $\boldsymbol{m}_t$. AdaShift prevents this problem by calculating the momentum $\boldsymbol{m}_t$ only using the latest n gradients as described in Eq. (5). In that case, the momentum $\boldsymbol{m}_t$ and the second moment estimate $\boldsymbol{v}_{t-n}$ are conditionally independent, so the convergence can be retained. However, this approach has a trade-off in the choice of n. When n is small, $\boldsymbol{m}_t$ has little information about the past gradients; when n is large, $\boldsymbol{v}_{t-n}$ only has access to the gradient information in the distant past.

To remove this trade-off, instead of truncating the momentum to the latest n steps, we propose to use momentum of the following form:

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1 - \beta_1) \frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_{t-1} + \epsilon^2}}, \tag{13}$$

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \alpha_t \boldsymbol{m}_t. \tag{14}$$

The main difference to the Adam-style momentum in Eq. (3) is the order of update of $\boldsymbol{m}_t$ and the normalization by $\sqrt{\boldsymbol{v}_{t-1} + \epsilon^2}$. In Eq. (3), the normalization is performed after the update of $\boldsymbol{m}_t$, whereas in Eq. (13), the normalization is first applied to the current gradient $\boldsymbol{g}_t$ in advance to the update of $\boldsymbol{m}_t$. In this case, the second moment estimate $\boldsymbol{v}_{t-1}$ is only used to normalize the current gradient $\boldsymbol{g}_t$, so the convergence can be guaranteed. A more detailed convergence analysis is provided in Section 4.

4 Method: Adaptive Gradient Method with the Optimal Convergence Rate

Based on the analysis in the previous section, we propose a new adaptive gradient method named ADOPT (*ADaptive gradient method with the OPTimal convergence rate*). The entire procedure is summarized in Algorithm 4. For a simple discription, we place the update of $\boldsymbol{m}$ after the parameter update in Algorithm 4, but it is equivalent to Eqs. (13) and (14) except that $\max\left\{\sqrt{\boldsymbol{v}}, \epsilon\right\}$ is substitued for $\sqrt{\boldsymbol{v} + \epsilon^2}$. The substitition is applied because we find that it contributes to slightly better performance in practice. We provide an equivalent expression of Algorithm 4 in Algorithm C in the appendix, which is closer to a practical implementation. By this modification, ADOPT can converge with the optimal rate for smooth nonconvex optimization as follows:

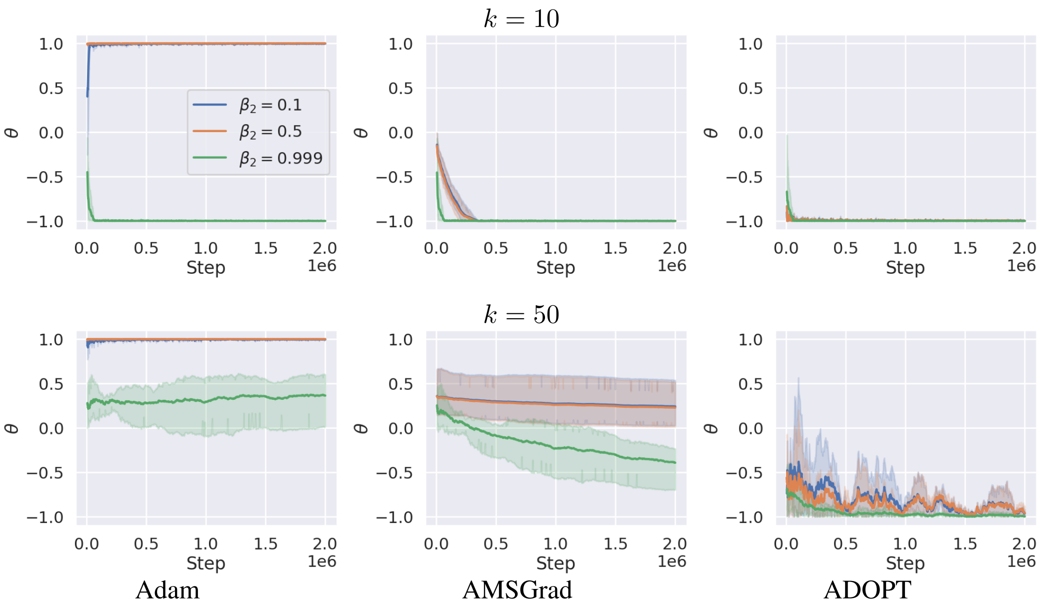


Figure 1: Performance comparison between Adam, AMSGrad and ADOPT in a simple univariate convex optimization problem. The plots show transitions of the parameter value, which should converge to the solution $\theta = -1$.

Theorem 4.1. *Under Assumptions 2.1-2.3 and 2.5, the following holds for the ADOPT algorithm with a constant learning rate* $\alpha_t = \alpha = \Theta\left(1/\sqrt{T}\right)$*:*

$$\min_{t=1,\ldots,T} \left\{ \mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1}))\|^{4/3} \right]^{3/2} \right\} \leq \mathcal{O}\left(1/\sqrt{T}\right), \tag{15}$$

The detailed proof and related lemmas are provided in the appendix. We also provide the convergence bound for the case of diminishing learning rate (i.e., $\alpha_t = \alpha/\sqrt{t}$) in the appendix, which is closer to practical situations. In that case, ADOPT also converges with the optimal rate.

5 Experiments

In the experiments, we first validate our ADOPT algorithm using a simple toy example in which Adam is known to fail to converge, and confirm our theoretical findings through numerical simulation. Secondly, we run an experiment of training a simple multi-layer perceptron (MLP) for the MNIST dataset to verify the effectiveness of our ADOPT for nonconvex optimization problems. Finally, we evaluate our ADOPT in a wide range of practical applications, including image classification, natural language processing (NLP) tasks, generative modeling, and deep reinforcement learning. Detailed experimental settings are described in the appendix.

Toy problem: We consider a convex optimization problem with an objective $f(\theta) = \theta$ for $\theta \in [-1, 1]$. It is obvious that a solution for the problem is $\theta = -1$. Through the optimization, we only have access to the stochastic objective f_t as follows:

$$f_t(\theta) = \begin{cases} k^2\theta, & \text{with probability } 1/k \\ -k\theta, & \text{with probability } 1 - 1/k \end{cases}, \tag{16}$$

where $k \geq 1$. Because $\mathbb{E}[f_t(\theta)] = f(\theta)$ holds, the stochastic gradient $g_t = \nabla f_t(\theta)$ is an unbiased estimator of the true gradient ∇f regardless of the choice of k, satisfying Assumption 2.2. This problem is equivalent, except for scaling, to the stochastic optimization version of Eq. (1) provided by Reddi et al. [2018] as a case where Adam fails to converge. In this setting, the constant k controls the magnitude of gradient noise. When $k = 1$, it corresponds to the noiseless case where $f_t = f$ with probability 1. As k gets large, stochastic gradient becomes noisy, making G in Assumptions

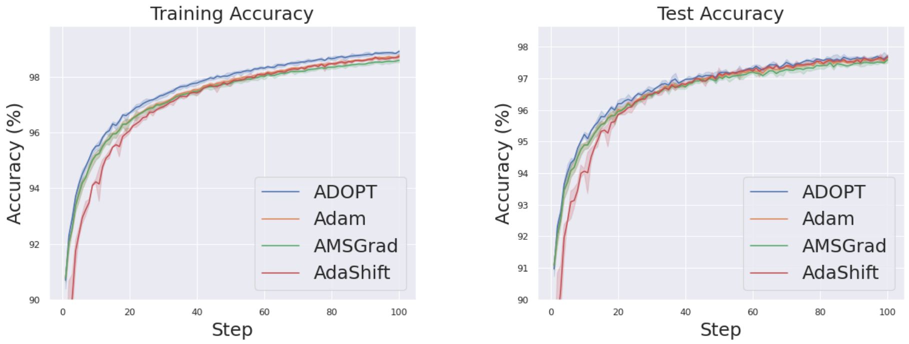


Figure 2: Accuracy for training data (left) and test data(right) in MNIST classification. The error bars show the 95% confidence intervals of three trials.

2.5 and 2.6 large. Therefore, the optimization will be more difficult when k becomes larger. In the experiment, we set $k = 10$ or 50, and compare the robustness of Adam, AMSGrad, and ADOPT for various hyperparameter settings by changing β_2 from $0.1 \sim 0.999$. We set $\beta_1 = 0.9$ for all the algorithms, which is a common choice in practice. We set the learning rate to $\alpha_t = 0.01/\sqrt{1 + 0.01t}$.

The result is shown in Figure 1. It can be seen that, when $k = 10$, Adam fails to converge except for $\beta_2 = 0.999$ while AMSGrad and ADOPT rapidly converge to the correct solution, i.e., $\theta = -1$, with any β_2. In a more extreme case where $k = 50$, Adam fails to converge even with $\beta_2 = 0.999$. This aligns with Theorem 3.1, since, when the gradient noise is large (i.e., G is large), the bounded region of the convergence bound also gets large, leading to divergence of Adam. Moreover, when $k = 50$, it is observed that the convergence of AMSGrad also becomes much slower than ADOPT. In fact, this phenomenon is also consistent with theory. In this problem setting, the second moment $\mathbb{E}[g_t^2]$ is $\mathcal{O}(k^3)$, while the squared norm of the stochastic gradient g_t^2 is $\mathcal{O}(k^4)$. Since the convergence bound of AMSGrad depends on the uniform bound of the stochastic gradient in Assumption 2.6, instead of the second moment in Assumption 2.5, its convergence also deteriorates with the order of g_t^2. Compared to AMSGrad, ADOPT only depends on the second moment bound for its convergence, so it converges much faster than AMSGrad even in such an extreme setting.

We also perform ablation study on how the two algorithmic changes from Adam to ADOPT affect the convergence. The differences between Adam and ADOPT are (1) decorrelation between the second moment estimate and the current gradient, and (2) change of order of momentum update and normalization by the second moment estimate. In this experiment, we remove each algorithmic change from ADOPT, and compare the result in the toy example. We set $k = 50$, and $(\beta_1, \beta_2) = (0.9, 0.999)$, since it is a common hyperparameter choice. The result is shown in Figure 3. It can be observed that ADOPT fails to converge with the exception of either algorithmic change. Therefore, applying both changes is essential to overcome the non-convergence of Adam, which also aligns with theory. These results correspond to the theoretical findings, showing the superiority of ADOPT to Adam and AMSGrad in terms of the convergence speed and its robustness to hyperparameter choices.

MNIST classification: To investigate the effectiveness of ADOPT on nonconvex optimization, we train nonlinear neural networks for MNIST classification tasks, and compare the performance between ADOPT and existing optimization algorithms, such as Adam, AMSGrad and AdaShift. In this experiment, we use a simple MLP with a single hidden layer, and the number of hidden units is set to 784. We set the learning rate to $\alpha_t = \alpha/\sqrt{t}$, and α is tuned in the range of $\{1, 10^{-1}, 10^{-2}, 10^{-3}\}$. We apply weight decay of 1×10^{-4} to prevent over-fitting, and run 10K iterations of parameter updates. Figure 2 shows the learning curves of training and test accuracy. We observe our ADOPT performs slightly better than the others in terms of the convergence speed and the final performance.

Image classification: As a more practical application, we conduct experiments of image classification using real-world image datasets. We first compare ADOPT and Adam in the classification task of the CIFAR-10 dataset using ResNet-18 [He et al., 2016], a widely-used convolutional neural network. We conduct a similar hyperparameter search to the case of MNIST classification. A detailed experimental setting is provided in the appendix. The learning curves of test accuracy are visualized in Figure 4. It can be observed that ADOPT converges a little faster than Adam.

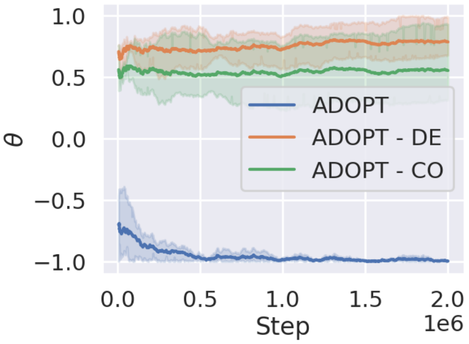


Figure 3: Ablation study of algorithmic changes between Adam and ADOPT. "DE" and CO denote "decorrelation" and "change of order", respectively.

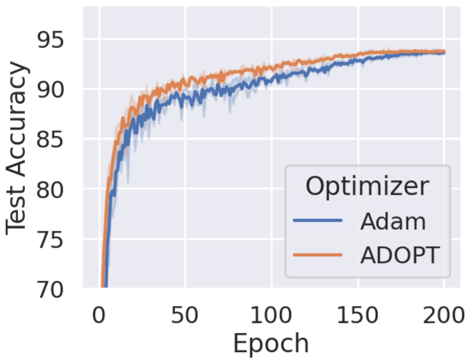


Figure 4: Learning curves of test accuracy for CIFAR-10 classification by ResNet-18 trained with Adam and ADOPT.

Table 1: Top-1 accuracy (%) for ImageNet classification by SwinTransformer.

Epoch	200	300
AdamW	79.29 ± 0.05	81.26 ± 0.04
AMSGrad	78.91 ± 0.03	81.17 ± 0.03
ADOPT	$\mathbf{79.62} \pm 0.03$	$\mathbf{81.50} \pm 0.04$

Table 2: Negative log-likelihood of NVAEs for MNIST density estimation. Lower is better.

Epoch	200	300
Adamax	80.19 ± 0.08	79.41 ± 0.07
ADOPT	$\mathbf{79.02} \pm 0.10$	$\mathbf{78.88} \pm 0.09$

To confirm that our ADOPT works well for modern neural network architectures based on Transformers [Vaswani et al., 2017], we perform an experiment of ImageNet classification using SwinTransformer [Liu et al., 2021]. We follow the official training recipe of Swin Transformer-tiny provided by Torchvision [Paszke et al., 2019a], and fix the training settings except for the optimizer choice. We use AdamW [Loshchilov and Hutter, 2019] as a baseline because it is set as the default official optimizer. We also compare with AMSGrad as another way to fix the non-convergence issue of Adam. Since AdamW uses decoupled weight decay, we also apply it to the other optimizers for fair comparison. We report the top-1 accuracy at 200 and 300 epochs in Tables 1. We observe that ADOPT outperforms AdamW and AMSGrad throughout the training in terms of the test accuracy, demonstrating the effectiveness of ADOPT for this setting.

Generative modeling: We train NVAE [Vahdat and Kautz, 2020] for MNIST using our ADOPT. In the official implementation of NVAE, Adamax [Kingma and Ba, 2014], an infinite-norm variant of Adam, is used as an optimizer, so we use Adamax as a baseline method. We use the exactly the same setting of the official implementation except that the learning rate for ADOPT is set to 2×10^{-4} since the default value 0.01 is too large for ADOPT. We report the negative log-likelihood for test data on Table 2. It is observed that the model trained with ADOPT shows the better likelihood.

Pretraining of large language models: We run a pre-training of GPT-2 [Radford et al., 2019] using the nanoGPT [Karpathy, 2022] code base to compare Adam and ADOPT. We use OpenWebText [Gokaslan and Cohen, 2019] as the training data. Experimental setup conforms to the default settings of nanoGPT except for the selection of the optimizer. We also test a case in which the total batch size was changed from 480 to 96, as a setting where the gradient noise becomes larger. The results are summarized in Figure 5. The most notable finding is that in the small batch size case, Adam causes loss spikes in the early stages of training and fails to converge, while ADOPT is always able to train stably. This is consistent with Adam's theory of non-convergence. As the gradient noise increases, G in Theorem 3.1 also increases, and the constant term in Adam's convergence bounds becomes non-negligible especially when using a large-scale dataset like OpenWebText. As a result, Adam is more likely to fail to train in such cases. Our ADOPT, on the other hand, does not suffer from this problem because it can always guarantee convergence. We also observed that both Adam and ADOPT work well when the batch size is large, but even in this case, ADOPT performs slightly better.

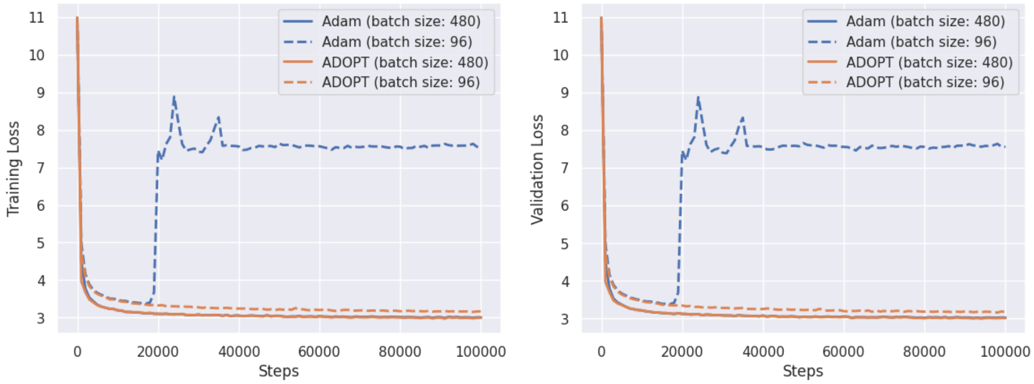


Figure 5: Learning curves of GPT-2 pretraining for training set (left) and validation set (right).

Finetuning of large language models: We finetune the pretrained LLaMA-7B on 52K instruction-following data provided by Stanford Alpaca and compare the performance between the default optimizer (Adam) and our ADOPT under the exactly same experimental setting. For evaluation, we use Multi-task Language Understanding (MMLU) Benchmark [Hendrycks et al., 2021], which is widely used to assess the performance of large language models. The MMLU score for LLaMA-7B without finetuning is 35.1. After fine-tuned via instruction-following using the baseline implementation with Adam, the score improves to 41.2. When we substitute ADOPT for Adam, the score even improves to 42.13. The detailed score comparison for each task is summarized in Figure 7 in the appendix. Other experimental results, including deep RL experiments, and detailed experimental settings are also provided in the appendix.

6 Conclusion

In this paper, we demystified the fundamental cause of divergence of adaptive gradient methods based on the exponential moving average, such as Adam and RMSprop, in general smooth nonconvex optimization problems, and demonstrate a way to fix the issue, proposing a new optimizer named ADOPT. Not only does ADOPT converge with the optimal rate without depending on a hyperparameter choice in theory, but ADOPT demonstrates better performance in a wide range of pracital applications.

We expect that this work will serve as a bridge between theory and practice in the research of adaptive gradient methods. Since ADOPT can be safely applied to many machine learning problems without careful tuning of hyperparameters, it can be expected to improve the training stability and the model performance in practice by substituting it for the existing adaptive gradient methods (e.g., Adam).

One of the limitations of our analysis is that it still relies on the assumption that the second moment of stochastic gradient is uniformly bounded (i.e., Assumption 2.5). Although this assumption is weaker than the bounded stochastic gradient assumption (i.e., Assumption 2.6), it would be more desirable to relax it to the bounded *variance* assumption (i.e., Assumption 2.4), which is often adopted in the analysis of the vanilla SGD [Ghadimi and Lan, 2013]. For Adam, a recent work by Wang et al. [2023] have derived a problem-dependent convergence bound which achieves the $\mathcal{O}(1/\sqrt{T})$ rate without Assumption 2.5. Their proof techniques may help to relax our assumptions in the proof of Theorem 4.1, which we leave as future work.

From a broader perspective, adaptive gradient methods like Adam have been widely used even for the training of large-scale foundation models (e.g., large language models). Although such models can be useful for people, their negative aspects, such as concerns about copyright infringement, are not negligible. Researchers needs to deeply recognize and understand such social impacts of machine learning algorithms.

References

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

Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=ryQu7f-RZ`.

Fangyu Zou, Li Shen, Zequn Jie, Weizhong Zhang, and Wei Liu. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11127–11135, 2019.

Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=H1x-x309tm`.

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

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

Dimitri P Bertsekas and John N Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM Journal on Optimization*, 10(3):627–642, 2000.

Ahmed Khaled and Peter Richtárik. Better theory for SGD in the nonconvex world. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=AU4qHN2VkS`. Survey Certification.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=PxTIG12RRHS`.

Zhiming Zhou, Qingru Zhang, Guansong Lu, Hongwei Wang, Weinan Zhang, and Yong Yu. Adashift: Decorrelation and convergence of adaptive learning rate methods. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=HkgTkhRcKQ`.

Naichen Shi, Dawei Li, Mingyi Hong, and Ruoyu Sun. Rmsprop converges with proper hyperparameter. In *International Conference on Learning Representations*, 2020.

Yushun Zhang, Congliang Chen, Naichen Shi, Ruoyu Sun, and Zhi-Quan Luo. Adam can converge without any modification on update rules. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=l5UNyaHqFdO`.

Bohan Wang, Yushun Zhang, Huishuai Zhang, Qi Meng, Zhi-Ming Ma, Tie-Yan Liu, and Wei Chen. Provable adaptivity in adam. *arXiv preprint arXiv:2208.09900*, 2022.

Haochuan Li, Ali Jadbabaie, and Alexander Rakhlin. Convergence of adam under relaxed assumptions. *arXiv preprint arXiv:2304.13972*, 2023.

Bohan Wang, Jingwen Fu, Huishuai Zhang, Nanning Zheng, and Wei Chen. Closing the gap between the upper bound and lower bound of adam's iteration complexity. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

Charles Blair. Problem complexity and method efficiency in optimization (as nemirovsky and db yudin). *Siam Review*, 27(2):264, 1985.

Stephen A Vavasis. Complexity issues in global optimization: a survey. *Handbook of global optimization*, pages 27–41, 1995.

Alexandre Défossez, Leon Bottou, Francis Bach, and Nicolas Usunier. A simple convergence proof of adam and adagrad. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL `https://openreview.net/forum?id=ZPQhzTSWA7`.

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.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *The Journal of Machine Learning Research*, 21(1):9047–9076, 2020.

Fangyu Zou, Li Shen, Zequn Jie, Ju Sun, and Wei Liu. Weighted adagrad with unified momentum. *arXiv preprint arXiv:1808.03408*, 2018.

Yoel Drori and Ohad Shamir. The complexity of finding stationary points with stochastic gradient descent. In *International Conference on Machine Learning*, pages 2658–2667. PMLR, 2020.

Xiaoyu Wang, Sindri Magnússon, and Mikael Johansson. On the convergence of step decay step-size for stochastic optimization. *Advances in Neural Information Processing Systems*, 34:14226–14238, 2021.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011. URL `http://jmlr.org/papers/v12/duchi11a.html`.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Lecture 6e rmsprop: Divide the gradient by a running average of its recent magnitude, 2012. URL `https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf`.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019a.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in neural information processing systems*, 33:19667–19679, 2020.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Andrej Karpathy. NanoGPT. https://github.com/karpathy/nanoGPT, 2022.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Jeff Haochen and Suvrit Sra. Random shuffling beats sgd after finite epochs. In *International Conference on Machine Learning*, pages 2624–2633. PMLR, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019b. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/haarnoja18b.html.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi: 10.1109/IROS.2012.6386109.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *The Journal of Machine Learning Research*, 22(1):12348–12355, 2021.

A Detailed Relationships to Existing Analyses

In this section, we discuss the relationships between our analysis and existing ones on the convergence of Adam-like optimizers in smooth nonconvex optimization problems. Tables 3 and 4 are a summary of comparisons between them in terms of their problem settings and derived convergence rates.

Zhang et al. [2022] focus on convergence of Adam in the finite sum problem, where the objective has a following form:

$$f(\boldsymbol{\theta}) = \sum_{i=1}^{n} f_i(\boldsymbol{\theta}). \tag{17}$$

f_i is, for example, a loss function for i-th training sample. Although many deep learning problems can be formulated as a finite sum problem, training of the variational autoencoders (VAEs) or diffusion models is out of the finite-sum problem, since their objective is formulated as an infinite sum (i.e., an expectation over continuous variables). Moreover, they assume the stochastic gradient $\boldsymbol{g}$ is L-Lipschitz, whereas we only assume true gradient ∇f is L-Lipschitz. They also assume a growth condition as follows:

$$\mathbb{E}\left[\|\boldsymbol{g}_t\|^2\right] \leq G_0^2 + G_1^2 \|\nabla f(\boldsymbol{\theta}_{t-1})\|^2. \tag{18}$$

This growth condition is weaker than our Assumption 2.5. Assumption 2.5 is a special case of the growth condition where $G_1 = 0$. Their derived convergence rate has a constant factor of $\mathcal{O}(G_0)$; hence the strong growth condition (i.e., $G_0 = 0$) is required to assure convergence. Moreover, to assure convergence, one needs to choose sufficiently large β_2, which has to be tuned in a problem-dependent manner.

Wang et al. [2022] also focus on convergence of Adam in the finite sum problem, but they relax the L-Lipschitz condition on $\boldsymbol{g}$ to the (L_0, L_1)-Lipschitz condition. They also assume the growth condition in Eq. (18), and their convergence rate has the same order with Zhang et al. [2022], so it still requires the strong growth condition (i.e., $G_0 = 0$) to assure convergence. The condition of β_2 is also similar to Zhang et al. [2022].

Li et al. [2023] consider Adam's convergence on general smooth nonconvex problems. Similar to Wang et al. [2022], they use (L_0, L_ρ)-Lipschitz condition on the true gradient ∇f. They also assume that the gradient noise is almost surely bounded:

$$\|\boldsymbol{g} - \nabla f\| \leq \sigma \tag{19}$$

The relationship between this assumption and our Assumption 2.5 is a little complicated. Assumption 2.5 is equivalent to a combination of Assumption 2.4 and the following assumption:

Assumption A.1. *The true gradient is uniformly bounded, i.e., there exist constants G and σ such that $\|\nabla f(\boldsymbol{\theta})\|^2 \leq G^2 - \sigma^2$ and $0 < \sigma \leq G$.*

The bounded noise assumption of Eq. (19) is strictly stronger than Assumption 2.4, but they do not assume the bounded true gradient (i.e., Assumption A.1). The bounded noise assumption is often violated in practice (e.g., training of VAEs), because the gradient is often estimated using unbounded noise (i.e., Gaussian noise). Their convergence rate $\mathcal{O}(1/\sqrt{T})$ is better than Zhang et al. [2022] and Wang et al. [2022], while it still requires constraints on the hyperparameters, which have to be chosen in a problem-dependent manner.

Défossez et al. [2022] analyzes the convergence of Adam under exactly the same assumptions with ours, and they derive the $\mathcal{O}(\log T/\sqrt{T})$ rate, which is worse than our ADOPT's convergence rate. Moereover, to assure the convergence, β_2 has to be chosen dependently on the total number of iterations T.

Wang et al. [2023] analyzes the convergence of Adam under Assumptions 2.1-2.4, and they derive the $\mathcal{O}(1/\sqrt{T})$ rate. However, to assure the convergence, β_2 has to be chosen dependently on the total number of iterations T as in Défossez et al. [2022].

Chen et al. [2019] and **Zhou et al. [2018]** analyze the convergence of AMSGrad for general smooth nonconvex problems, and derive the convergence rate of $\mathcal{O}(\log T/\sqrt{T})$ and $\mathcal{O}(1/\sqrt{T})$, respectively. However, to guarantee the convergence, the stochastic gradient $\boldsymbol{g}$ has to be bounded almost surely

	Algorithm	Problem	Smoothness	Gradient Growth
Zhang et al. [2022]	Adam	Finite sum	L-Lipschitz $\boldsymbol{g}$	$\mathbb{E}[\|g\|^2] \leq G_0^2 + G_1^2 \|\nabla f\|^2$
Wang et al. [2022]	Adam	Finite sum	(L_0, L_1)-Lipschitz $\boldsymbol{g}$	$\mathbb{E}[\|g\|^2] \leq G_0^2 + G_1^2 \|\nabla f\|^2$
Li et al. [2023]	Adam	General	(L_0, L_ρ)-Lipschitz ∇f	$\|\boldsymbol{g} - \nabla f\| \leq \sigma$
Défossez et al. [2022]	Adam	General	L-Lipschitz ∇f	$\mathbb{E}[\|\boldsymbol{g}\|^2] \leq G^2$
Wang et al. [2023]	Adam	General	L-Lipschitz ∇f	$\mathbb{E}[\|\boldsymbol{g} - \nabla f\|^2] \leq G^2$
Chen et al. [2019]	AMSGrad	General	L-Lipschitz ∇f	$\|\boldsymbol{g}\| \leq G$
Zhou et al. [2018]	AMSGrad	General	L-Lipschitz ∇f	$\|\boldsymbol{g}\| \leq G$
Ours	ADOPT	General	L-Lipschitz ∇f	$\mathbb{E}[\|\boldsymbol{g}\|^2] \leq G^2$

Table 3: Comparison of the problem settings between our analysis and other existing works.

	Constraints	Convergence
Zhang et al. [2022]	$\beta_1 < \sqrt{\beta_2}, \beta_2 \geq \gamma(n)$	$\mathcal{O}(\log T/\sqrt{T}) + \mathcal{O}(G_0)$
Wang et al. [2022]	$\beta_1 < \sqrt{\beta_2}, \delta(\beta_2) = \mathcal{O}(1/G_1)$	$\mathcal{O}(\log T/\sqrt{T}) + \mathcal{O}(G_0)$
Li et al. [2023]	$\beta_1 < \sqrt{\beta_2}, \beta_1 \leq c(L_0, L_\rho, G)$	$\mathcal{O}(1/\sqrt{T})$
Défossez et al. [2022]	$\beta_1 < \sqrt{\beta_2}, 1 - \beta_2 = \Theta(1/T)$	$\mathcal{O}(\log T/\sqrt{T})$
Wang et al. [2023]	$\beta_1 \leq \sqrt{\beta_2} - 8\left(1 - \beta_2\right) \beta_2^{-2}, 1 - \beta_2 = \Theta(1/T)$	$\mathcal{O}(1/\sqrt{T})$
Chen et al. [2019]	$\beta_1 < \sqrt{\beta_2}$	$\mathcal{O}(\log T/\sqrt{T})$
Zhou et al. [2018]	$\beta_1 < \sqrt{\beta_2}$	$\mathcal{O}(1/\sqrt{T})$
Ours	-	$\mathcal{O}(1/\sqrt{T})$

Table 4: Comparison of the convergence rate and imposed constraints on the hyperparameters between our analysis and other existing works. Please refer to the original papers for the definitions of γ and c.

(Assumption 2.6), which is often violated in practice. In addition, the hyperparameter β_1 and β_2 should be chosen satisfying $\beta_1 < \sqrt{\beta_2}$. This constraint is relatively minor compared to the constraint imposed in the analyses of Adam, since it can be satisfied in a problem-independent manner.

B With-Replacement vs. Without-Replacement

In the optimization of finite-sum problems, practitioners often use *without-replacement sampling*, which is also known as *random shuffling*, to obtain stochastic gradient. In this case, the stochastic gradient has a small bias due to the lack of replacement, so Assumption 2.2 is violated. However, the vanilla SGD is known to converge with the without-replacement strategy [Haochen and Sra, 2019], and some of the analyses of Adam also adopt without-replacement sampling [Zhang et al., 2022, Wang et al., 2022].

Unfortunately, we find that our ADOPT has a counter example, in which ADOPT fails to converge when using without-replacement sampling. For example, when we consider minimizing $f(\theta) = \sum_{i=1}^{3} f_i(\theta)$, where $\theta \in [-1, 1]$, $f_1(\theta) = 1.9\theta$ and $f_2(\theta) = f_2(\theta) = -\theta$, it can be easily observed that ADOPT with $\beta_1 = \beta_2 = 0$ fails to converge to the correct solution, i.e., $\theta = 1$.

This non-convergence can be easily avoided by using the with-replacement strategy. Moreover, the difference between with- and without-replacement sampling becomes negligible when n in the finite-sum $\sum_{i=1}^{n} f_i$ is large enough; hence it does not affect the practical performance very much. In fact, our experiments except for the toy example are performed using without-replacement sampling, but divergent behaviors are not observed. If one applies ADOPT to problems where the difference seems severe (e.g., when training with a small dataset), we recommend to use with-replacement sampling instead of random shuffling for stable training. When one uses PyTorch [Paszke et al., 2019b] for the implementation, for example, with-replacement sampling can be easily applied by specifying `replacemnet=True` for `torch.utils.data.RandomSampler`, and feeding it to the `sampler` argument of `torch.utils.data.DataLoader`.

C Another Expression of ADOPT

Algorithm 2 Alternative representation of ADOPT algorithm

```
Require: Learning rate $\{\alpha_t\}$, initial parameter $\boldsymbol{\theta}_0$
Require: Exponential decay rate $0 \leq \beta_1 < 1, 0 \leq \beta_2 \leq 1$, small constant $\epsilon > 0$
  $\boldsymbol{v}_0 \leftarrow \boldsymbol{g}_0 \odot \boldsymbol{g}_0$
  for $t = 1$ to $T$ do
    if $t = 1$ then
      $\boldsymbol{m}_t \leftarrow \boldsymbol{g}_t / \max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}$
    else
      $\boldsymbol{m}_t \leftarrow \beta_1 \cdot \boldsymbol{m}_{t-1} + (1 - \beta_1)\, \boldsymbol{g}_t / \max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}$
    end if
    $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \alpha_t \boldsymbol{m}_t$
    $\boldsymbol{v}_t \leftarrow \beta_2 \cdot \boldsymbol{v}_{t-1} + (1 - \beta_2)\, \boldsymbol{g}_t \odot \boldsymbol{g}_t$
  end for
  return $\{\boldsymbol{\theta}_t\}_{t=1}^T$
```

D Recommendation of Hyperparameter Settings for ADOPT

We experimentally find that our ADOPT works similarly to Adam when the same hyperparameters are used, but ϵ should be set to a little larger value (e.g., 1×10^{-6}) for ADOPT compared to Adam, in which ϵ is set to 1×10^{-8} by default. Our recommendation of the hyperparameter settings for ADOPT is provided in Table 5.

β_1	0.9
β_2	0.9999
ϵ	1×10^{-6}

Table 5: Recommended hyperparameters for the ADOPT algorithm

E Theorems

Theorem E.1. *Under Assumptions 2.1, 2.2, 2.3, and 2.5, if the objective f is upper-bounded by $f_{\sup}$, the following holds for the ADOPT algorithm with a learning rate $\alpha_t = \alpha/\sqrt{t}$:*

$$\begin{aligned}
&\min_{t=1,\ldots,T} \left\{ \mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_t)\|^{4/3} \right]^{3/2} \right\} \\
&\leq \frac{3\sqrt{\max\{G^2, 1\} + \epsilon^2}}{2\left((T+1)^{3/2} - 1\right)} \left(\frac{f_{\sup} - f_{\inf}}{\alpha} (T+1) + \left(\frac{\sqrt{2}\alpha\beta_1 G^2 L}{\epsilon^2 (1-\beta_1)} + \frac{\alpha G^2 L}{2\epsilon^2} \right) T \right) \\
&\quad + \frac{3\sqrt{\max\{G^2, 1\} + \epsilon^2}}{2\left((T+1)^{3/2} - 1\right)} \left(\frac{2\sqrt{2}\beta_1 G^2}{\epsilon (1-\beta_1)} \left(\sqrt{T+1} - 1 \right) + \frac{\alpha \beta_1^2 G^2 L}{\epsilon (1-\beta_1)^2} \frac{T}{T+1} \right) \\
&\quad + \frac{3\sqrt{\max\{G^2, 1\} + \epsilon^2}}{2\left((T+1)^{3/2} - 1\right)} \left(\frac{2\alpha\beta_1^2 G^2 L}{\epsilon (1-\beta_1)^2} + \frac{\alpha^2 \beta_1 G^2 L}{\sqrt{2}\epsilon^2 (1-\beta_1)} \right) \log(T+1). \qquad (20)
\end{aligned}$$

F Proofs

Proof of Theorems 4.1 and E.1. We define $\boldsymbol{\phi}_t$ for $t \geq 1$ as follows:

$$\boldsymbol{\phi}_t = \frac{1}{1-\beta_1} \boldsymbol{\theta}_t - \frac{\beta_1}{1-\beta_1} \boldsymbol{\theta}_{t-1}. \qquad (21)$$

We also define $\boldsymbol{\phi}_0 = \boldsymbol{\theta}_0$. By Assumption 2.3, the following holds for $t \geq 1$:

$$\begin{aligned}
f\left(\boldsymbol{\phi}_t\right) &\leq f\left(\boldsymbol{\phi}_{t-1}\right) + \nabla f\left(\boldsymbol{\phi}_{t-1}\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right) + \frac{L}{2}\left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|^2 &(22)\\
&= f\left(\boldsymbol{\phi}_{t-1}\right) + \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\\
&\quad + \left(\nabla f\left(\boldsymbol{\phi}_{t-1}\right) - \nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right) + \frac{L}{2}\left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|^2 &(23)\\
&\leq f\left(\boldsymbol{\phi}_{t-1}\right) + \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\\
&\quad + \left\|\nabla f\left(\boldsymbol{\phi}_{t-1}\right) - \nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\| \left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\| + \frac{L}{2}\left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|^2 &(24)\\
&\leq f\left(\boldsymbol{\phi}_{t-1}\right) + \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\\
&\quad + L\left\|\boldsymbol{\phi}_{t-1} - \boldsymbol{\theta}_{t-1}\right\| \left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\| + \frac{L}{2}\left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|^2, &(25)
\end{aligned}$$

where the second inequality holds due to the Cauchy-Schwarz inequality, and the last inequality holds due to Assumption 2.3. By taking the expectation, the following holds:

$$\begin{aligned}
\mathbb{E}\left[f\left(\boldsymbol{\phi}_t\right)\right] &\leq \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] + \mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\right]\\
&\quad + L\mathbb{E}\left[\left\|\boldsymbol{\phi}_{t-1} - \boldsymbol{\theta}_{t-1}\right\| \left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|\right] + \frac{L}{2}\mathbb{E}\left[\left\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right\|^2\right] &(26)\\
&\leq \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] + \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1\left(1 - \beta_1^{t-1}\right)G^2}{\left(1 - \beta_1\right)\epsilon} - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1 - \beta_2^T\right)G^2 + \epsilon^2}} &(27)\\
&\quad + \frac{\alpha_{t-1}\left(\alpha_{t-1} - \alpha_t\right)\beta_1^2\left(1 - \beta_1^{t-1}\right)G^2 L}{\epsilon^2\left(1 - \beta_1\right)^2} + \frac{\alpha_t\alpha_{t-1}\beta_1\sqrt{1 - \beta_1^{t-1}}G^2 L}{\left(1 - \beta_1\right)\epsilon^2}\\
&\quad + \frac{\left(\alpha_{t-1} - \alpha_t\right)^2\beta_1^2\left(1 - \beta_1^{t-1}\right)G^2 L}{2\left(1 - \beta_1\right)^2\epsilon^2}\\
&\quad + \frac{\alpha_t^2 G^2 L}{2\epsilon^2} + \frac{\alpha_t\left(\alpha_{t-1} - \alpha_t\right)\beta_1\sqrt{1 - \beta_1^{t-1}}G^2 L}{2\left(1 - \beta_1\right)\epsilon^2}.
\end{aligned}$$

When $\alpha_t = \alpha$, the following holds:

$$\begin{aligned}
\mathbb{E}\left[f\left(\boldsymbol{\phi}_t\right)\right] &\leq \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] - \alpha\frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1 - \beta_2^T\right)G^2 + \epsilon^2}} + \frac{\alpha^2\beta_1\sqrt{1 - \beta_1^{t-1}}G^2 L}{\left(1 - \beta_1\right)\epsilon^2} + \frac{\alpha^2 G^2 L}{2\epsilon^2} &(28)\\
&\leq \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] - \alpha\frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}}{\sqrt{\left(1 - \beta_2^T\right)G^2 + \epsilon^2}} + \frac{\alpha^2\left(1 + \beta_1\right)G^2 L}{2\left(1 - \beta_1\right)\epsilon^2}. &(29)
\end{aligned}$$

Telescoping it for $t = 1, \ldots, T$, we have

$$\begin{aligned}
\mathbb{E}\left[f\left(\boldsymbol{\phi}_T\right)\right] &\leq f\left(\boldsymbol{\theta}_0\right) - \alpha\frac{\sum_{t=1}^T \mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}}{\sqrt{\left(1 - \beta_2^T\right)G^2 + \epsilon^2}} + \frac{\alpha^2\left(1 + \beta_1\right)G^2 LT}{2\left(1 - \beta_1\right)\epsilon^2} &(30)\\
&\leq f\left(\boldsymbol{\theta}_0\right) - \alpha\frac{\sum_{t=1}^T \mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}}{\sqrt{\left(1 - \beta_2^T\right)G^2 + \epsilon^2}} + \frac{\alpha^2\left(1 + \beta_1\right)G^2 LT}{2\left(1 - \beta_1\right)\epsilon^2} &(31)
\end{aligned}$$

By rearranging the terms, we have

$$\begin{aligned}
&\min_{t=1,\ldots,T}\left\{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}\right\} \\
&\leq \frac{\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}}{T} && (32)\\
&\leq \sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}\left(\frac{f\left(\boldsymbol{\theta}_0\right)-f_{\inf}}{\alpha T}+\frac{\alpha\left(1+\beta_1\right)G^2L}{2\left(1-\beta_1\right)\epsilon^2}\right) && (33)\\
& && (34)
\end{aligned}$$

When $\alpha_t = \alpha/\sqrt{t}$, the following holds for $t \geq 2$:

$$\begin{aligned}
\alpha_{t-1}-\alpha_t &= \alpha\left(\frac{1}{\sqrt{t-1}}-\frac{1}{\sqrt{t}}\right) && (35)\\
&= \frac{\alpha\left(\sqrt{t}-\sqrt{t-1}\right)}{\sqrt{t\left(t-1\right)}} && (36)\\
&= \frac{\alpha}{\sqrt{t\left(t-1\right)}\left(\sqrt{t}+\sqrt{t-1}\right)} && (37)\\
&\leq \frac{\alpha}{2\left(t-1\right)^{3/2}} && (38)\\
&\leq \frac{\sqrt{2}\alpha}{t^{3/2}}. && (39)
\end{aligned}$$

This also holds for $t = 1$ by defining $\alpha_0 = \alpha$. Applying it to Eq. (27), we have

$$\begin{aligned}
\mathbb{E}\left[f\left(\boldsymbol{\phi}_t\right)\right] \leq\ & \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] + \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1\left(1-\beta_1^{t-1}\right)G^2}{\left(1-\beta_1\right)\epsilon} - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}} \\
& + \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2\left(1-\beta_1^{t-1}\right)G^2L}{\epsilon^2\left(1-\beta_1\right)^2} + \frac{\alpha_t\alpha_{t-1}\beta_1\sqrt{1-\beta_1^{t-1}}G^2L}{\left(1-\beta_1\right)\epsilon^2} \\
& + \frac{\left(\alpha_{t-1}-\alpha_t\right)^2\beta_1^2\left(1-\beta_1^{t-1}\right)G^2L}{2\left(1-\beta_1\right)^2\epsilon^2} + \frac{\alpha_t^2G^2L}{2\epsilon^2} \\
& + \frac{\alpha_t\left(\alpha_{t-1}-\alpha_t\right)\beta_1\sqrt{1-\beta_1^{t-1}}G^2L}{2\left(1-\beta_1\right)\epsilon^2} \qquad (40) \\
\leq\ & \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] + \frac{\sqrt{2}\alpha\beta_1\left(1-\beta_1^{t-1}\right)G^2}{t^{3/2}\left(1-\beta_1\right)\epsilon} - \frac{\alpha}{\sqrt{t}}\frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}} \\
& + \frac{2\alpha^2\beta_1^2G^2L}{\epsilon^2\left(1-\beta_1\right)^2t^2} + \frac{\sqrt{2}\alpha^2\beta_1G^2L}{\left(1-\beta_1\right)\epsilon^2t} + \frac{\alpha^2\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2t^3} + \frac{\alpha^2G^2L}{2\epsilon^2t} \\
& + \frac{\alpha^2\beta_1G^2L}{\sqrt{2}\left(1-\beta_1\right)\epsilon^2t^2} \qquad (41) \\
=\ & \mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)\right] - \frac{\alpha}{\sqrt{t}}\frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}} \\
& + \frac{\alpha^2\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{2\left(1-\beta_1\right)\epsilon^2}\cdot t^{-1} + \frac{\sqrt{2}\alpha\beta_1G^2}{\left(1-\beta_1\right)\epsilon}\cdot t^{-\frac{3}{2}} \\
& + \frac{\alpha^2\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{\sqrt{2}\left(1-\beta_1\right)^2\epsilon^2}\cdot t^{-2} + \frac{\alpha^2\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2}\cdot t^{-3}. \qquad (42)
\end{aligned}$$

Multiplying t to the both sides and rearranging the terms, we have

$$\begin{aligned}
& \frac{\sqrt{t}\,\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}} \\
& \leq \frac{\mathbb{E}\left[f\left(\boldsymbol{\phi}_{t-1}\right)-f\left(\boldsymbol{\phi}_t\right)\right]}{\alpha}\cdot t + \frac{\alpha\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{2\left(1-\beta_1\right)\epsilon^2} + \frac{\sqrt{2}\beta_1G^2}{\left(1-\beta_1\right)\epsilon}\cdot t^{-\frac{1}{2}} \qquad (43) \\
& + \frac{\alpha\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{\sqrt{2}\left(1-\beta_1\right)^2\epsilon^2}t^{-1} + \frac{\alpha\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2}t^{-2} \qquad (44)
\end{aligned}$$

$$
\begin{aligned}
&\sum_{t=1}^{T} \frac{\sqrt{t}\,\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}} \\
&\leq \frac{f(\boldsymbol{\phi}_0) - Tf(\boldsymbol{\phi}_T) + \sum_{t=1}^{T-1} f(\boldsymbol{\phi}_t)}{\alpha} + \frac{\alpha\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2LT}{2\left(1-\beta_1\right)\epsilon^2} \\
&\quad + \frac{\sqrt{2}\beta_1 G^2}{\left(1-\beta_1\right)\epsilon}\sum_{t=1}^{T} t^{-\frac{1}{2}} + \frac{\alpha\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{\sqrt{2}\left(1-\beta_1\right)^2\epsilon^2}\sum_{t=1}^{T} t^{-1} + \frac{\alpha\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2}\sum_{t=1}^{T} t^{-2} \qquad (45) \\
&\leq \frac{f_{\sup}-f_{\inf}}{\alpha}T + \frac{\alpha\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2LT}{2\left(1-\beta_1\right)\epsilon^2} \\
&\quad + \frac{\sqrt{2}\beta_1 G^2}{\left(1-\beta_1\right)\epsilon}\left(1+\int_1^T t^{-\frac{1}{2}}dt\right) + \frac{\alpha\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{\sqrt{2}\left(1-\beta_1\right)^2\epsilon^2}\left(1+\int_1^T t^{-1}dt\right) \\
&\quad + \frac{\alpha\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2}\left(1+\int_1^T t^{-2}dt\right) \qquad (46) \\
&\leq \frac{f_{\sup}-f_{\inf}}{\alpha}T + \frac{\alpha\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2LT}{2\left(1-\beta_1\right)\epsilon^2} + \frac{\sqrt{2}\beta_1 G^2}{\left(1-\beta_1\right)\epsilon}\left(2\sqrt{T}-1\right) \\
&\quad + \frac{\alpha\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)G^2L}{\sqrt{2}\left(1-\beta_1\right)^2\epsilon^2}\left(1+\log T\right) + \frac{\alpha\beta_1^2G^2L}{\left(1-\beta_1\right)^2\epsilon^2}\left(2-\frac{1}{T}\right) \qquad (47)
\end{aligned}
$$

Therefore, the following bound is derived.

$$
\begin{aligned}
&\min_{t=1,\dots,T}\left\{\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2}\right\} \\
&\leq \frac{\sum_{t=1}^{T}\sqrt{t}\,\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2}}{\sum_{t=1}^{T}\sqrt{t}} \qquad (48) \\
&\leq \frac{\sum_{t=1}^{T}\sqrt{t}\,\mathbb{E}\left[\|\nabla f(\boldsymbol{\theta}_{t-1})\|^{4/3}\right]^{3/2}}{\int_0^T\sqrt{t}dt} \qquad (49) \\
&\leq \frac{3C_T\left(f_{\sup}-f_{\inf}\right)}{2\alpha}\frac{1}{\sqrt{T}} + \frac{3\alpha\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)C_TG^2L}{4\left(1-\beta_1\right)\epsilon^2\sqrt{T}} + \frac{3\beta_1C_TG^2}{\sqrt{2}\left(1-\beta_1\right)\epsilon}\left(\frac{2}{T}-\frac{1}{T^{3/2}}\right) \\
&\quad + \frac{3\alpha\beta_1\left(1+\left(2\sqrt{2}-1\right)\beta_1\right)C_TG^2L}{2\sqrt{2}\left(1-\beta\right)^2\epsilon^2}\left(\frac{1}{T^{3/2}}+\frac{\log T}{T^{3/2}}\right) + \frac{3\alpha\beta_1^2C_TG^2L}{2\left(1-\beta_1\right)^2\epsilon^2}\left(\frac{2}{T^{3/2}}-\frac{1}{T^{5/2}}\right), \qquad (50)
\end{aligned}
$$

where $C_T = \sqrt{\left(1-\beta_2^T\right)G^2+\epsilon^2}$.

□

G Lemmas

Lemma G.1. *For all $\boldsymbol{\theta} \in \mathbb{R}^D$ and $t \geq 1$, the following holds*

$$\|\nabla f(\boldsymbol{\theta}_{t-1})\| \leq G. \qquad (51)$$

Proof.

$$\|\nabla f(\boldsymbol{\theta}_{t-1})\| = \sqrt{\|\mathbb{E}[\boldsymbol{g}_t]\|^2} \tag{52}$$

$$\leq \sqrt{\mathbb{E}\left[\|\boldsymbol{g}_t\|^2\right]} \tag{53}$$

$$\leq G. \tag{54}$$

The first inequality holds because $\mathbb{E}[(\boldsymbol{g}_t)_i]^2 \leq \mathbb{E}[(\boldsymbol{g}_t)_i^2]$, and the second inequality holds due to Assumption 2.5. □

Lemma G.2. *For all $\boldsymbol{\theta} \in \mathbb{R}^D$ and $t \geq 1$, the following holds*

$$\mathbb{E}[\|\boldsymbol{g}_t\|] \leq G \tag{55}$$

Proof.

$$\mathbb{E}[\|\boldsymbol{g}_t\|] \leq \mathbb{E}\left[\|\boldsymbol{g}_t\|^2\right]^{1/2} \tag{56}$$

$$\leq G, \tag{57}$$

where the first inequality holds due to the Hölder's inequality and the second one holds due to Assumption 2.5. □

Lemma G.3. *For the RMSprop algorithm, the following holds for $t \geq 1$:*

$$\mathbb{E}\left[\sum_{i=1}^{D} (\boldsymbol{v}_t)_i\right] \leq \left(1 - \beta_2^t\right) G^2 \tag{58}$$

Proof.

$$\mathbb{E}\left[\sum_{i=1}^{D} (\boldsymbol{v}_t)_i\right] = \mathbb{E}\left[(1-\beta_2) \sum_{i=1}^{D} \sum_{k=1}^{t} \beta_2^{t-k} (\boldsymbol{g}_k)_i^2\right] \tag{59}$$

$$\leq (1-\beta_2) G^2 \sum_{k=1}^{t} \beta_2^{t-k} \tag{60}$$

$$= \left(1 - \beta_2^t\right) G^2. \tag{61}$$

□

Lemma G.4. *For the RMSprop algorithm, the following holds:*

$$\begin{aligned}&\mathbb{E}\left[\nabla f(\boldsymbol{\theta}_{t-1})^\top \left(\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}}\right)\right] \\ &\geq \frac{1}{2}\mathbb{E}\left[\nabla f(\boldsymbol{\theta}_{t-1})^\top \left(\frac{\boldsymbol{g}_t}{\sqrt{\tilde{\boldsymbol{v}}_t + \epsilon^2}}\right)\right] - 2G\sqrt{1-\beta_2}\,\mathbb{E}\left[\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}}\right\|^2\right]\end{aligned} \tag{62}$$

Proof.

$$\mathbb{E}\left[\nabla f(\boldsymbol{\theta}_{t-1})^\top \left(\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}}\right)\right] = \sum_{i=1}^{D} \mathbb{E}\left[\frac{(\nabla f(\boldsymbol{\theta}_{t-1}))_i (\boldsymbol{g}_t)_i}{\sqrt{(\boldsymbol{v}_t)_i + \epsilon^2}}\right] \tag{63}$$

We define $\tilde{\boldsymbol{v}}_t$ as follows:

$$\tilde{\boldsymbol{v}}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)\,\mathbb{E}[\boldsymbol{g}_t \odot \boldsymbol{g}_t] \tag{64}$$

Using this, the following holds:

$$\begin{align}
&\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}}\right] \nonumber\\
&=\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\mathbb{E}\left[\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\left(\frac{1}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}}-\frac{1}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right)\right] \tag{65}\\
&=\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\mathbb{E}\left[\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\left(\frac{1}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}}-\frac{1}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right)\right] \tag{66}\\
&\geq\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]-\mathbb{E}\left[\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\left(\frac{1}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}}-\frac{1}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right)\right|\right], \tag{67}
\end{align}$$

where the last inequality holds due to $A \geq -|A|$. For the second term, the following holds:

$$\begin{align}
&\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\left(\frac{1}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}}-\frac{1}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right)\right| \nonumber\\
&=\left(1-\beta_2\right)\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\frac{\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]-\left(\boldsymbol{g}_t\right)_i^2}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}\left(\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}+\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}\right)}\right| \tag{68}\\
&\leq\left(1-\beta_2\right)\left(\frac{\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\right|\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\left(\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2\right)}+\frac{\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\right|\left(\boldsymbol{g}_t\right)_i^2}{\left(\left(\boldsymbol{v}_t\right)_i+\epsilon^2\right)\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right), \tag{69}
\end{align}$$

where the last inequality holds due to the triangle inequality. For the first term, the following holds:

$$\begin{align}
&\mathbb{E}\left[\frac{\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\right|\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}{\sqrt{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\left(\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2\right)}\right] \nonumber\\
&\leq\frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\left(1-\beta_2\right)\mathbb{E}\left[\frac{\left(\boldsymbol{g}_t\right)_i^2\,\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]^2}{\left(\left(\boldsymbol{v}_t\right)_i+\epsilon^2\right)\left(\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2\right)^{3/2}}\right] \tag{70}\\
&\leq\frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\mathbb{E}\left[\frac{\left(\boldsymbol{g}_t\right)_i^2\sqrt{\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}}{\sqrt{1-\beta_2}\left(\left(\boldsymbol{v}_t\right)_i+\epsilon^2\right)}\right] \tag{71}\\
&\leq\frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\frac{G}{\sqrt{1-\beta_2}}\mathbb{E}\left[\frac{\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\right] \tag{72}
\end{align}$$

The first inequality is derived using the following fact:

$$\forall\lambda>0, x, y\in\mathbb{R}, xy\leq\frac{\lambda}{2}x^2+\frac{y^2}{2\lambda}. \tag{73}$$

For the second term of Eq. (69), the following holds:

$$\begin{aligned}
&\mathbb{E}\left[\frac{\left|\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i\left(\boldsymbol{g}_t\right)_i\right|\left(\boldsymbol{g}_t\right)_i^2}{\left(\left(\boldsymbol{v}_t\right)_i+\epsilon^2\right)\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right] && (74)\\
&\leq \frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\frac{\left(\boldsymbol{g}_t\right)_i^2}{\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}\right]+\left(1-\beta_2\right)\mathbb{E}\left[\frac{\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}{\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\frac{\left(\boldsymbol{g}_t\right)_i^4}{\left(\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2\right)^2}\right] && (75)\\
&\leq \frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\mathbb{E}\left[\frac{\sqrt{\mathbb{E}\left[\left(\boldsymbol{g}_t\right)_i^2\right]}\left(\boldsymbol{g}_t\right)_i^2}{\sqrt{1-\beta_2}\left(\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2\right)}\right] && (76)\\
&\leq \frac{1}{\left(1-\beta_2\right)}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{4\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]+\frac{G}{\sqrt{1-\beta_2}}\mathbb{E}\left[\frac{\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\right] && (77)
\end{aligned}$$

The first inequality is derived using Eq. (73).

Putting these inequalities together, the following is derived:

$$\begin{aligned}
&\mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top\left(\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right)\right]\\
&\geq \sum_{i=1}^{D}\mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{2\sqrt{\left(\tilde{\boldsymbol{v}}_t\right)_i+\epsilon^2}}\right]-2G\sqrt{1-\beta_2}\mathbb{E}\left[\frac{\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\right] && (78)\\
&\geq \frac{1}{2}\mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top\left(\frac{\boldsymbol{g}_t}{\sqrt{\tilde{\boldsymbol{v}}_t+\epsilon^2}}\right)\right]-2G\sqrt{1-\beta_2}\mathbb{E}\left[\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2\right]. && (79)
\end{aligned}$$

□

Lemma G.5. *For the RMSprop algorithm, the following holds:*

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2\right]\leq D\left(\log\left(1+\frac{\left(1-\beta_2^T\right)G^2}{\epsilon^2}\right)-T\log\beta_2\right) \tag{80}$$

Proof.

$$\left\|\frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t+\epsilon^2}}\right\|^2=\sum_{i=1}^{D}\frac{\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2} \tag{81}$$

$$\begin{aligned}
\frac{\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2} &= \frac{1}{1-\beta_2}\frac{\left(1-\beta_2\right)\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2} && (82)\\
&\leq -\frac{1}{1-\beta_2}\log\left(1-\frac{\left(1-\beta_2\right)\left(\boldsymbol{g}_t\right)_i^2}{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}\right) && (83)\\
&= \frac{1}{1-\beta_2}\log\left(\frac{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}{\beta_2\left(\boldsymbol{v}_{t-1}\right)_i+\epsilon^2}\right) && (84)\\
&= \frac{1}{1-\beta_2}\left(\log\left(\frac{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}{\left(\boldsymbol{v}_{t-1}\right)_i+\epsilon^2}\right)+\log\left(\frac{\left(\boldsymbol{v}_{t-1}\right)_i+\epsilon^2}{\beta_2\left(\boldsymbol{v}_{t-1}\right)_i+\epsilon^2}\right)\right) && (85)\\
&\leq \frac{1}{1-\beta_2}\left(\log\left(\frac{\left(\boldsymbol{v}_t\right)_i+\epsilon^2}{\left(\boldsymbol{v}_{t-1}\right)_i+\epsilon^2}\right)-\log\beta_2\right) && (86)
\end{aligned}$$

$$\sum_{t=1}^{T} \frac{(\boldsymbol{g}_t)_i^2}{(\boldsymbol{v}_t)_i + \epsilon^2} \leq \frac{1}{1-\beta_2} \left(\log \left(\frac{(\boldsymbol{v}_T)_i + \epsilon^2}{\epsilon^2} \right) - T \log \beta_2 \right) \tag{87}$$

$$\leq \frac{1}{1-\beta_2} \left(\log \left(1 + \frac{\left(1-\beta_2^T\right) G^2}{\epsilon^2} \right) - T \log \beta_2 \right) \tag{88}$$

$$\sum_{t=1}^{T} \mathbb{E} \left[\left\| \frac{\boldsymbol{g}_t}{\sqrt{\boldsymbol{v}_t + \epsilon^2}} \right\|^2 \right] \leq \sum_{i=1}^{D} \mathbb{E} \left[\sum_{t=1}^{T} \frac{(\boldsymbol{g}_t)_i^2}{(\boldsymbol{v}_t)_i + \epsilon^2} \right] \tag{89}$$

$$\leq \frac{1}{1-\beta_2} \sum_{i=1}^{D} \mathbb{E} \left[\log \left(1 + \frac{(\boldsymbol{v}_T)_i}{\epsilon^2} \right) \right] - \frac{DT \log \beta_2}{1-\beta_2} \tag{90}$$

$$\leq \sum_{i=1}^{D} \log \left(1 + \frac{\mathbb{E}\left[(\boldsymbol{v}_T)_i\right]}{\epsilon^2} \right) - \frac{DT \log \beta_2}{1-\beta_2} \tag{91}$$

$$\leq \frac{D}{1-\beta_2} \left(\log \left(1 + \frac{\left(1-\beta_2^T\right) G^2}{\epsilon^2} \right) - T \log \beta_2 \right) \tag{92}$$

□

Lemma G.6. *For the RMSprop algorithm, the following holds:*

$$\mathbb{E} \left[\nabla f \left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\frac{\boldsymbol{g}_t}{\sqrt{\beta_2 \tilde{\boldsymbol{v}}_t + \epsilon^2}} \right) \right] \geq \frac{\mathbb{E}\left[\left\| \nabla f \left(\boldsymbol{\theta}_{t-1}\right) \right\|^{4/3} \right]^{3/2}}{\sqrt{\left(1-\beta_2^t\right) G^2 + \epsilon^2}} \tag{93}$$

Proof.

$$\begin{aligned}
&\mathbb{E} \left[\nabla f \left(\boldsymbol{\theta}_{t-1}\right)^\top \left(\frac{\boldsymbol{g}_t}{\sqrt{\tilde{\boldsymbol{v}}_t + \epsilon^2}} \right) \right] \\
&= \sum_{i=1}^{D} \mathbb{E} \left[\frac{\left(\nabla f \left(\boldsymbol{\theta}_{t-1}\right)\right)_i \cdot (\boldsymbol{g}_t)_i}{\sqrt{(\tilde{\boldsymbol{v}}_t)_i + \epsilon^2}} \right] \\
&= \sum_{i=1}^{D} \mathbb{E} \left[\frac{\left(\nabla f \left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{\sqrt{\beta_2 \left(\boldsymbol{v}_{t-1}\right)_i + \epsilon^2}} \right] \\
&\geq \mathbb{E} \left[\frac{\left\| \nabla f \left(\boldsymbol{\theta}_{t-1}\right) \right\|^2}{\sqrt{\sum_{i=1}^{D} (\tilde{\boldsymbol{v}}_t)_i + \epsilon^2}} \right] \\
&\geq \frac{\mathbb{E}\left[\left\| \nabla f \left(\boldsymbol{\theta}_{t-1}\right) \right\|^{4/3} \right]^{3/2}}{\sqrt{\mathbb{E}\left[\sum_{i=1}^{D} (\tilde{\boldsymbol{v}}_t)_i \right] + \epsilon^2}} \\
&\geq \frac{\mathbb{E}\left[\left\| \nabla f \left(\boldsymbol{\theta}_{t-1}\right) \right\|^{4/3} \right]^{3/2}}{\sqrt{\left(1-\beta_2^t\right) G^2 + \epsilon^2}}.
\end{aligned} \tag{94}$$

The second equality holds due to Assumption 2.2. The first inequality holds because $(\tilde{\boldsymbol{v}}_t)_i \geq 0$ for all $i = 1, \ldots, D$. The second inequality holds due to the Hölder's inequality. The last inequality holds due to Lemma G.3. □

Lemma G.7. *For the ADOPT algorithm, the following holds for* $t \geq 1$*:*

$$\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1} = \frac{\left(\alpha_{t-1} - \alpha_t\right) \beta_1}{1-\beta_1} \boldsymbol{m}_{t-1} - \alpha_t \frac{\boldsymbol{g}_t}{\max\left\{ \sqrt{\boldsymbol{v}_{t-1}}, \epsilon \right\}}, \tag{95}$$

where we define $\alpha_0 = \alpha$*.*

Proof. For $t = 1$, the following holds by definition:

$$\boldsymbol{\phi}_1 - \boldsymbol{\phi}_0 = \frac{1}{1-\beta_1}\boldsymbol{\theta}_1 - \left(\frac{\beta_1}{1-\beta_1} + 1\right)\boldsymbol{\theta}_0 \tag{96}$$

$$= \frac{1}{1-\beta_1}\left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right) \tag{97}$$

$$= -\frac{\alpha_1 \boldsymbol{g}_1}{\max\left\{\sqrt{\boldsymbol{v}_0}, \epsilon\right\}}. \tag{98}$$

For $t \geq 2$, the following holds:

$$\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1} = \frac{1}{1-\beta_1}\left(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}\right) - \frac{\beta_1}{1-\beta_1}\left(\boldsymbol{\theta}_{t-1} - \boldsymbol{\theta}_{t-2}\right) \tag{99}$$

$$= \frac{1}{1-\beta_1}\left(\alpha_{t-1}\beta_1 \boldsymbol{m}_{t-1} - \alpha_t \boldsymbol{m}_t\right) \tag{100}$$

$$= \frac{1}{1-\beta_1}\left(\alpha_{t-1}\beta_1 \boldsymbol{m}_{t-1} - \alpha_t\left(\beta_1 \boldsymbol{m}_{t-1} + \left(1-\beta_1\right)\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right)\right) \tag{101}$$

$$= \frac{1}{1-\beta_1}\left(\left(\alpha_{t-1} - \alpha_t\right)\beta_1 \boldsymbol{m}_{t-1} - \alpha_t\left(1-\beta_1\right)\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right) \tag{102}$$

$$= \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1}{1-\beta_1}\boldsymbol{m}_{t-1} - \alpha_t \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \tag{103}$$

□

Lemma G.8. *For the ADOPT algorithm, the following holds for* $t \geq 1$*:*

$$\boldsymbol{\phi}_{t-1} - \boldsymbol{\theta}_{t-1} = -\frac{\alpha_{t-1}\beta_1}{1-\beta_1}\boldsymbol{m}_{t-1}. \tag{104}$$

Proof. For $t = 1$, Eq. (104) holds obviously because $\boldsymbol{\phi}_0 = \boldsymbol{\theta}_0$ and $\boldsymbol{m}_0 = \boldsymbol{0}$. For $t \geq 2$, the following holds:

$$\boldsymbol{\phi}_{t-1} - \boldsymbol{\theta}_{t-1} = \left(\frac{1}{1-\beta_1} - 1\right)\boldsymbol{\theta}_{t-1} - \frac{\beta_1}{1-\beta_1}\boldsymbol{\theta}_{t-2} \tag{105}$$

$$= \frac{\beta_1}{1-\beta_1}\left(\boldsymbol{\theta}_{t-1} - \boldsymbol{\theta}_{t-2}\right) \tag{106}$$

$$= -\frac{\alpha_{t-1}\beta_1}{1-\beta_1}\boldsymbol{m}_{t-1}. \tag{107}$$

□

Lemma G.9. *For the ADOPT algorithm, the following holds for* $t \geq 1$*:*

$$\begin{aligned} &\mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top\left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\right] \\ &\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1\left(1 - \beta_1^{t-1}\right)G^2}{\left(1-\beta_1\right)\sqrt{\beta_2^{t-2} + \epsilon^2}} - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^t\right)G^2 + \epsilon^2}}. \end{aligned} \tag{108}$$

Proof.

$$\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top\left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)$$

$$= \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1}{1-\beta_1}\nabla f\left(\boldsymbol{\theta}_{t-1}\right))^\top \boldsymbol{m}_{t-1} - \alpha_t \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \tag{109}$$

$$\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1}{1-\beta_1}\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right))\right\|\left\|\boldsymbol{m}_{t-1}\right\| - \alpha_t \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \tag{110}$$

$$\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)\beta_1 G}{1-\beta_1}\left\|\boldsymbol{m}_{t-1}\right\| - \alpha_t \nabla f\left(\boldsymbol{\theta}_{t-1}\right)^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}. \tag{111}$$

By taking the expectation for both sides, the following holds:

$$
\begin{aligned}
&\mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^{\top} \cdot \left(\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\right)\right] \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \mathbb{E}\left[\nabla f\left(\boldsymbol{\theta}_{t-1}\right)^{\top} \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right] && (112) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \sum_{i=1}^{D} \mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i \cdot \left(\boldsymbol{g}_t\right)_i}{\max\left\{\sqrt{\left(\boldsymbol{v}_{t-1}\right)_i}, \epsilon\right\}}\right] && (113) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \sum_{i=1}^{D} \mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{\max\left\{\sqrt{\left(\boldsymbol{v}_{t-1}\right)_i}, \epsilon\right\}}\right] && (114) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \sum_{i=1}^{D} \mathbb{E}\left[\frac{\left(\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right)_i^2}{\sqrt{\left(\boldsymbol{v}_{t-1}\right)_i + \epsilon^2}}\right] && (115) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \mathbb{E}\left[\frac{\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|^2}{\sqrt{\sum_{i=1}^{D}\left(\boldsymbol{v}_{t-1}\right)_i + \epsilon^2}}\right] && (116) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\mathbb{E}\left[\sum_{i=1}^{D}\left(\boldsymbol{v}_{t-1}\right)_i\right] + \epsilon^2}} && (117) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1 G}{1-\beta_1}\mathbb{E}\left[\left\|\boldsymbol{m}_{t-1}\right\|\right] - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^t\right)G^2 + \epsilon^2}} && (118) \\
&\leq \frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1\left(1-\beta_1^{t-1}\right)G^2}{\left(1-\beta_1\right)\sqrt{\beta_2^{t-2}+\epsilon^2}} - \alpha_t \frac{\mathbb{E}\left[\left\|\nabla f\left(\boldsymbol{\theta}_{t-1}\right)\right\|_i^{4/3}\right]^{3/2}}{\sqrt{\left(1-\beta_2^t\right)G^2 + \epsilon^2}}. && (119)
\end{aligned}
$$

□

Lemma G.10. *For the ADOPT algorithm, the following holds for* $t \geq 0$*:*

$$
\mathbb{E}\left[\sum_{i=1}^{D}\left(\boldsymbol{v}_t\right)_i\right] \leq \left(1-\beta_2^t\right)G^2. \tag{120}
$$

Proof.

$$
\begin{aligned}
\mathbb{E}\left[\sum_{i=1}^{D}\left(\boldsymbol{v}_t\right)_i\right] &= \mathbb{E}\left[\left(1-\beta_2\right)\sum_{i=1}^{D}\sum_{k=1}^{t}\beta_2^{t-k}\left(\boldsymbol{g}_{k-1}\right)_i^2\right] && (121) \\
&\leq \left(1-\beta_2\right)G^2\sum_{k=1}^{t}\beta_2^{t-k} && (122) \\
&= \left(1-\beta_2^t\right)G^2. && (123)
\end{aligned}
$$

□

Lemma G.11. *For the ADOPT algorithm, the following holds for* $0 \leq t \leq T$*.*

$$
\mathbb{E}\left[\left\|\boldsymbol{m}_t\right\|^2\right] \leq \frac{G^2}{\epsilon^2}. \tag{124}
$$

Proof.

$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{m}_t\|^2\right] \\
&= \mathbb{E}\left[\left\|\beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right\|^2\right] && (125) \\
&= \mathbb{E}\left[\beta_1^2 \|\boldsymbol{m}_{t-1}\|^2 + (1-\beta_1)^2 \left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right\|^2 + 2\beta_1 (1-\beta_1) \boldsymbol{m}_{t-1}^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right] && (126) \\
&\le \mathbb{E}\left[\beta_1 \|\boldsymbol{m}_{t-1}\|^2 + (1-\beta_1)\left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right\|^2\right] && (127) \\
&\le \mathbb{E}\left[\beta_1 \|\boldsymbol{m}_{t-1}\|^2 + \frac{1-\beta_1}{\epsilon^2}\|\boldsymbol{g}_t\|^2\right] && (128) \\
&\le \mathbb{E}\left[\frac{1-\beta_1}{\epsilon^2}\sum_{k=1}^{t}\beta_1^{t-k}\|\boldsymbol{g}_k\|^2\right] && (129) \\
&\le \frac{(1-\beta_1) G^2}{\epsilon^2}\sum_{k=1}^{t}\beta_1^{t-k} && (130) \\
&\le \frac{(1-\beta_1^t) G^2}{\epsilon^2} && (131) \\
&\le \frac{G^2}{\epsilon^2}. && (132)
\end{aligned}$$

First inequality is derived using the following fact:

$$\forall \lambda > 0, \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d, \boldsymbol{x}^\top \boldsymbol{y} \le \frac{\lambda}{2}\|\boldsymbol{x}\|^2 + \frac{1}{2\lambda}\|\boldsymbol{y}\|^2 \tag{133}$$

By setting $\lambda = (1-\beta_1)/\beta_1, \boldsymbol{x} = \beta_1 \boldsymbol{m}_{t-1}, \boldsymbol{y} = (1-\beta_1)\boldsymbol{g}_t / \max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}$, we obtain

$$2\beta_1(1-\beta_1)\boldsymbol{m}_{t-1}^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \le \beta_1(1-\beta_1)\left(\|\boldsymbol{m}_{t-1}\|^2 + \left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}}\right\|^2\right) \tag{134}$$

Injecting it into Eq. (126), we obtain Eq. (127).

□

Lemma G.12. *For the ADOPT algorithm, the following holds for* $t \ge 0$.

$$\mathbb{E}\left[\|\boldsymbol{m}_t\|\right] \le \frac{G}{\epsilon} \tag{135}$$

Proof.

$$
\begin{aligned}
\mathbb{E}\left[\|\boldsymbol{m}_t\|\right] &= \mathbb{E}\left[\left\|(1-\beta_1)\sum_{k=1}^{t}\beta_1^{t-k}\frac{\boldsymbol{g}_k}{\max\left\{\sqrt{\boldsymbol{v}_{k-1}},\epsilon\right\}}\right\|\right] && (136)\\
&\leq (1-\beta_1)\sum_{k=1}^{t}\beta_1^{t-k}\mathbb{E}\left[\left\|\frac{\boldsymbol{g}_k}{\max\left\{\sqrt{\boldsymbol{v}_{k-1}},\epsilon\right\}}\right\|\right] && (137)\\
&\leq (1-\beta_1)\sum_{k=1}^{t}\frac{\beta_1^{t-k}}{\epsilon}\mathbb{E}\left[\|\boldsymbol{g}_k\|\right] && (138)\\
&\leq \frac{1-\beta_1}{\epsilon}\sum_{k=1}^{t}\beta_1^{t-k}\mathbb{E}\left[\|\boldsymbol{g}_k\|^2\right]^{1/2} && (139)\\
&\leq \frac{(1-\beta_1)\,G}{\epsilon}\sum_{k=1}^{t}\beta_1^{t-k} && (140)\\
&= \frac{(1-\beta_1^t)\,G}{\epsilon} && (141)\\
&\leq \frac{G}{\epsilon}. && (142)
\end{aligned}
$$

□

Lemma G.13. *For the ADOPT algorithm, the following holds for* $t \geq 1$*:*

$$
\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{\phi}_{t-1}-\boldsymbol{\theta}_{t-1}\|\,\|\boldsymbol{\phi}_t-\boldsymbol{\phi}_{t-1}\|\right]\\
&\leq \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2\left(1-\beta_1^{t-1}\right)G^2}{\epsilon^2\left(1-\beta_1\right)^2} + \frac{\alpha_t\alpha_{t-1}\beta_1\sqrt{1-\beta_1^{t-1}}G^2}{\epsilon^2\left(1-\beta_1\right)}. && (143)
\end{aligned}
$$

Proof.

$$
\begin{aligned}
&\|\boldsymbol{\phi}_{t-1}-\boldsymbol{\theta}_{t-1}\|\,\|\boldsymbol{\phi}_t-\boldsymbol{\phi}_{t-1}\|\\
&= \left\|-\frac{\alpha_{t-1}\beta_1}{1-\beta_1}\boldsymbol{m}_{t-1}\right\|\left\|\frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1}{1-\beta_1}\boldsymbol{m}_{t-1}-\alpha_t\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}},\epsilon\right\}}\right\| && (144)\\
&\leq \frac{\alpha_{t-1}\beta_1}{1-\beta_1}\|\boldsymbol{m}_{t-1}\|\left(\frac{\left(\alpha_{t-1}-\alpha_t\right)\beta_1}{1-\beta_1}\|\boldsymbol{m}_{t-1}\|+\alpha_t\left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}},\epsilon\right\}}\right\|\right) && (145)\\
&\leq \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2}{\left(1-\beta_1\right)^2}\|\boldsymbol{m}_{t-1}\|^2+\frac{\alpha_t\alpha_{t-1}\beta_1}{1-\beta_1}\|\boldsymbol{m}_{t-1}\|\left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}},\epsilon\right\}}\right\|. && (146)
\end{aligned}
$$

Taking the expectation yields:

$$
\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{\phi}_{t-1}-\boldsymbol{\theta}_{t-1}\|\,\|\boldsymbol{\phi}_t-\boldsymbol{\phi}_{t-1}\|\right]\\
&\leq \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2}{\left(1-\beta_1\right)^2}\mathbb{E}\left[\|\boldsymbol{m}_{t-1}\|^2\right]+\frac{\alpha_t\alpha_{t-1}\beta_1}{1-\beta_1}\mathbb{E}\left[\|\boldsymbol{m}_{t-1}\|\left\|\frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}},\epsilon\right\}}\right\|\right] && (147)\\
&\leq \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2}{\left(1-\beta_1\right)^2}\mathbb{E}\left[\|\boldsymbol{m}_{t-1}\|^2\right]+\frac{\alpha_t\alpha_{t-1}\beta_1}{\left(1-\beta_1\right)\epsilon}\mathbb{E}\left[\|\boldsymbol{m}_{t-1}\|\,\|\boldsymbol{g}_t\|\right] && (148)\\
&\leq \frac{\alpha_{t-1}\left(\alpha_{t-1}-\alpha_t\right)\beta_1^2\left(1-\beta_1^{t-1}\right)G^2}{\epsilon^2\left(1-\beta_1\right)^2}+\frac{\alpha_t\alpha_{t-1}\beta_1\sqrt{1-\beta_1^{t-1}}G^2}{\left(1-\beta_1\right)\epsilon^2}. && (149)
\end{aligned}
$$

□

Lemma G.14. *For the ADOPT algorithm, the following holds for* $t \geq 1$*:*

$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\|^2\right] \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2 \left(1 - \beta_1^{t-1}\right) G^2}{\left(1 - \beta_1\right)^2 \epsilon^2} + \frac{\alpha_t^2 G^2}{\epsilon^2} + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1 \sqrt{1 - \beta_1^{t-1}} G^2}{\left(1 - \beta_1\right) \epsilon^2}. \qquad (150)
\end{aligned}$$

Proof.

$$\begin{aligned}
&\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\|^2 \\
&= \left\| \frac{\left(\alpha_{t-1} - \alpha_t\right) \beta_1}{1 - \beta_1} \boldsymbol{m}_{t-1} - \alpha_t \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \right\|^2 \qquad (151) \\
&= \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2}{\left(1 - \beta_1\right)^2} \|\boldsymbol{m}_{t-1}\|^2 + \alpha_t^2 \left\| \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \right\|^2 \\
&\quad - \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1}{1 - \beta_1} \boldsymbol{m}_{t-1}^\top \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \qquad (152) \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2}{\left(1 - \beta_1\right)^2} \|\boldsymbol{m}_{t-1}\|^2 + \frac{\alpha_t^2}{\epsilon^2} \|\boldsymbol{g}_t\|^2 \\
&\quad + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1}{1 - \beta_1} \|\boldsymbol{m}_{t-1}\| \left\| \frac{\boldsymbol{g}_t}{\max\left\{\sqrt{\boldsymbol{v}_{t-1}}, \epsilon\right\}} \right\| \qquad (153) \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2}{\left(1 - \beta_1\right)^2} \|\boldsymbol{m}_{t-1}\|^2 + \frac{\alpha_t^2}{\epsilon^2} \|\boldsymbol{g}_t\|^2 + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1}{\left(1 - \beta_1\right) \epsilon} \|\boldsymbol{m}_{t-1}\| \|\boldsymbol{g}_t\|. \qquad (154)
\end{aligned}$$

Taking the expectation yields:

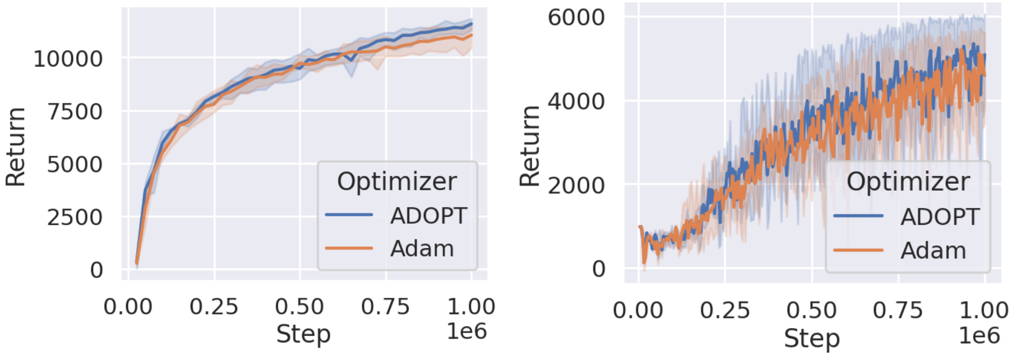
$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}\|^2\right] \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2}{\left(1 - \beta_1\right)^2} \mathbb{E}\left[\|\boldsymbol{m}_{t-1}\|^2\right] + \frac{\alpha_t^2}{\epsilon^2} \mathbb{E}\left[\|\boldsymbol{g}_t\|^2\right] + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1}{\left(1 - \beta_1\right) \epsilon} \mathbb{E}\left[\|\boldsymbol{m}_{t-1}\| \|\boldsymbol{g}_t\|\right] \qquad (155) \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2 \left(1 - \beta_1^{t-1}\right) G^2}{\left(1 - \beta_1\right)^2 \epsilon^2} + \frac{\alpha_t^2 G^2}{\epsilon^2} + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1}{\left(1 - \beta_1\right) \epsilon} \mathbb{E}\left[\|\boldsymbol{m}_{t-1}\| \|\boldsymbol{g}_t\|\right] \qquad (156) \\
&\leq \frac{\left(\alpha_{t-1} - \alpha_t\right)^2 \beta_1^2 \left(1 - \beta_1^{t-1}\right) G^2}{\left(1 - \beta_1\right)^2 \epsilon^2} + \frac{\alpha_t^2 G^2}{\epsilon^2} + \frac{\alpha_t \left(\alpha_{t-1} - \alpha_t\right) \beta_1 \sqrt{1 - \beta_1^{t-1}} G^2}{\left(1 - \beta_1\right) \epsilon^2}. \qquad (157)
\end{aligned}$$

□

H Additional Experiments

Deep reinforcement learning:

We train reinforcement learning (RL) agents using the soft actor crtitic algorithm [Haarnoja et al., 2018] with ADOPT for the optimizer. As a benchmark, we use a continuous control tasks of HalfCheetah-v4 on MuJoCo simulator [Todorov et al., 2012]. For comparison to ADOPT, Adam is used as a baseline optimizer. We follow the hyperparameter settings recommended by Stable-Baselines3 [Raffin et al., 2021], and just change the choice of an optimizer. The result is shown in Figure 6. The error bars indicate 95% confidence intervals of three trials. We observe slight performance improvement by using ADOPT instead of Adam.



HalfCheetah-v4 **Ant-v4**

Figure 6: Performance comparison between Adam and ADOPT in reinforcement learning.

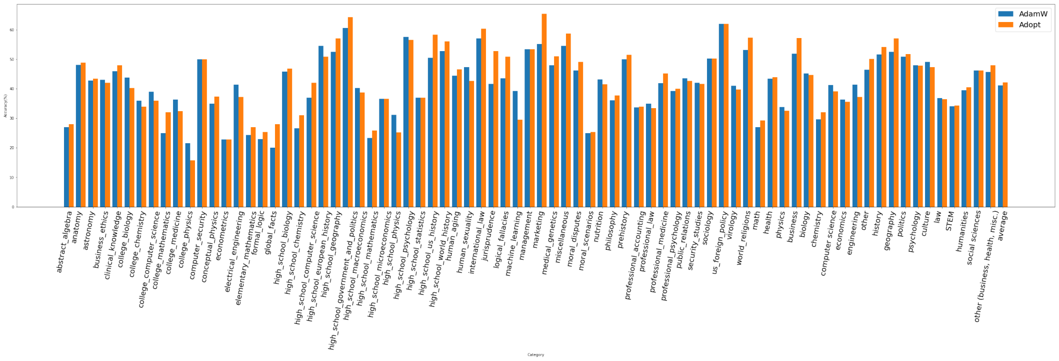


Figure 7: Comparison of MMLU scores for LLaMA-7B finetuned via instruction following using AdamW and ADOPT.

I Details of Experimental Setups

I.1 Code

Our implementation for the experiment is available at `https://github.com/iShohei220/adopt`.

I.2 Total amount of compute

We run our experiments mainly on cloud GPU instances with 8× A100. It took approximately 320 hours for our experiments in total.

I.3 License of Assets

Datasets: The MNIST database is downloaded from `http://yann.lecun.com/exdb/mnist`, which is license-free. The terms of access for the ImageNet database is provided at `https://www.image-net.org/download`. The dataset of Stanford Alpaca is CC BY NC 4.0 (allowing only non-commercial use).

Pretrained models: The pretrained model of LLaMA is provided under GNU General Public License v3.0.

Simulator: MuJoCo is provided under Apache License 2.0.

Code: Our implementation of ImageNet classification is based on the Torchvision's official training recipe provided at `https://github.com/UiPath/torchvision/tree/master/references/classification`. Torchvision is provided under BSD 3-Clause License. We use the official implementation of NVAE provided at `https://github.com/NVlabs/NVAE`, whose license is described at `https://github.com/NVlabs/NVAE/blob/master/LICENSE`.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our main contribution is to demystify the cause of non-convergence of Adam, which is clearly written in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are described in the last section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Assumptions and proofs are provided in Section 2 and the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Experimental settings are provided in Section 5 and the appendix. We also share the implementation of the experiment.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Data and code are provided in the appendix.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Detailed experimental settings are provided in Section 5 and the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Error bars are reported in all the figures and tables.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Computational resources used in our experiments are reported in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We confirmed that our research conforms with the Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discussed them in the last section of the paper.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: They are provided both in the main paper and the appendix.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.