HHS Public Access

Author manuscript

Neurocomputing. Author manuscript; available in PMC 2023 April 07.

Published in final edited form as:

Neurocomputing. 2022 April 07; 481: 333–356. doi:10.1016/j.neucom.2022.01.014.

Calibrating the Adaptive Learning Rate to Improve Convergence of ADAM

Qianqian Tong[1], **Guannan Liang**[1], **Jinbo Bi**[1,*]

[1]Computer Science and Engineering, University of Connecticut, Storrs, CT 06269

Abstract

Adaptive gradient methods (AGMs) have become popular in optimizing the nonconvex problems in deep learning area. We revisit AGMs and identify that the adaptive learning rate (A-LR) used by AGMs varies significantly across the dimensions of the problem over epochs (i.e., anisotropic scale), which may lead to issues in convergence and generalization. All existing modified AGMs actually represent efforts in revising the A-LR. Theoretically, we provide a new way to analyze the convergence of AGMs and prove that the convergence rate of ADAM also depends on its hyper-parameter ϵ, which has been overlooked previously. Based on these two facts, we propose a new AGM by calibrating the A-LR with an activation (*softplus*) function, resulting in the SADAM and SAMSGRAD methods. We further prove that these algorithms enjoy better convergence speed under nonconvex, non-strongly convex, and Polyak-Łojasiewicz conditions compared with ADAM. Empirical studies support our observation of the anisotropic A-LR and show that the proposed methods outperform existing AGMs and generalize even better than S-Momentum in multiple deep learning tasks.

Keywords

ADAM; Deep learning; Adaptive methods; Stochastic methods

1. Introduction

Many machine learning problems can be formulated as the minimization of an objective function f of the form: $\min_{x \in \mathbb{R}^d} f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$, where both f and f_i maybe nonconvex in deep learning. Stochastic gradient descent (SGD), its variants such as SGD with momentum (S-Momentum) [1, 2, 3, 4], and adaptive gradient methods (AGMs) [5, 6, 7] play important roles in deep learning area due to simplicity and wide applicability. In particular, AGMs often exhibit fast initial progress in training and are easy to implement in solving large scale optimization problems. The updating rule of AGMs can be generally written as:

*Corresponding author jinbo.bi@uconn.edu (Jinbo Bi).

$$x_{t+1} = x_t - \frac{\eta_t}{\sqrt{v_t}} \odot m_t, \quad (1)$$

where $\odot$ calculates element-wise product of the first-order momentum m_t and the learning rate (LR) $\frac{\eta_t}{\sqrt{v_t}}$. There is fairly an agreement on how to compute m_t, which is a convex combination of previous m_{t-1} and current stochastic gradient g_t, i.e., $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$, $\beta_1 \in [0,1]$. The LR consists of two parts: the base learning rate (B-LR) η_t is a scalar which can be constant or decay over iterations. In our convergence analysis, we consider the B-LR as constant η. The adaptive learning rate $\frac{1}{\sqrt{v_t}}$, varies adaptively across dimensions of the problem, where $v_t \in \mathbb{R}^d$ is the second-order momentum calculated as a combination of previous and current squared stochastic gradients. Unlike the first-order momentum, the formula to estimate the second-order momentum varies in different AGMs. As the core technique in AGMs, A-LR opens a new regime of controlling LR, and allows the algorithm to move with different step sizes along the search direction at different coordinates.

The first known AGM is ADAGRAD [5] where the second-order momentum is estimated as $v_t = \sum_{i=1}^{t} g_i^2$. It works well in sparse settings, but the A-LR often decays rapidly for dense gradients. To tackle this issue, ADADELTA [7], RMSPROP [8], ADAM [6] have been proposed to use exponential moving averages of past squared gradients, i.e., $v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$, $\beta_2 \in [0,1]$ and calculate the A-LR by $\frac{1}{\sqrt{v_t} + \epsilon}$ where $\epsilon > 0$ is used in case that v_t vanishes to zero. In particular, ADAM has become the most popular optimizer in the deep learning area due to its effectiveness in early training stage. Nevertheless, it has been empirically shown that ADAM generalizes worse than S-Momentum to unseen data and leaves a clear generalization gap [9, 10, 11], and even fails to converge in some cases [12, 13]. AGMs decrease the objective value rapidly in early iterations, and then stay at a plateau whereas SGD and S-Momentum continue to show dips in the training error curves, and thus continue to improve test accuracy over iterations. It is essential to understand what happens to ADAM in the later learning process, so we can revise AGMs to enhance their generalization performance.

Recently, a few modified AGMs have been developed, such as, AMSGRAD [12], YOGI [14], and ADABOUND [13]. AMSGRAD is the first method to theoretically address the non-convergence issue of ADAM by taking the largest second-order momentum estimated in the past iterations, i.e., $v_t = max\{v_{t-1}, \tilde{v}_t\}$ where $\tilde{v}_t = \beta_2 \tilde{v}_{t-1} + (1 - \beta_2) g_t^2$, and proves its convergence in the convex case. The analysis is later extended to other AGMs (such as RMSPROP and AMSGRAD) in nonconvex settings [15, 16, 17, 18]. YOGI claims that the past g_t^2's are forgotten in a fairly fast manner in ADAM and proposes $v_t = v_{t-1} - (1 - \beta_2) sign\left(v_{t-1} - g_t^2\right) g_t^2$ to adjust the decay rate of the A-LR. However, the parameter in the A-LR is adjusted to 10^{-3}, instead of 10^{-8} in the default setting of ADAM, so ϵ dominates the A-LR in later iterations when v_t becomes small and can be responsible for performance improvement. The hyper-parameter has rarely been discussed

previously and our analysis shows that the convergence rate is closely related to ϵ, which is further verified in our experiments. PADAM[1] [19, 15] claims that the A-LR in ADAM and AMSGRAD are "overadapted", and proposes to replace the A-LR updating formula by $1/((v_t)^p + \epsilon)$ where p ∈ (0,1/2] ADABOUND confines the LR to a predefined range by applying), $Clip\left(\frac{\eta}{\sqrt{v_t}}, \eta_l, \eta_r\right)$, where LR values outside the interval $[\eta_l, \eta_r]$ are clipped to the interval edges. However, a more effective way is to softly and smoothly calibrate the A-LR rather than hard-thresholding the A-LR at all coordinates. Our main contributions are summarized as follows:

1. We study AGMs from a new perspective: the range of the A-LR. Through experimental studies, we find that the A-LR is always anisotropic. This anisotropy may lead the algorithm to focus on a few dimensions (those with large A-LR), which may exacerbate generalization performance. We analyze the existing modified AGMs to help explain how they close the generalization gap.
2. Theoretically, we are the first to include hyper-parameter ϵ into the convergence analysis and clearly show that the convergence rate is upper bounded by a $1/\epsilon^2$ term, verifying prior observations that ϵ affects performance of ADAM empirically. We provide a new approach to convergence analysis of AGMs under the nonconvex, non-strongly convex, or Polyak-Łojasiewicz (P-L) condition.
3. Based on the above two results, we propose to calibrate the A-LR using an activation function, particularly we implement the *softplus* function with a hyper-parameter β, which can be combined with any AGM. In this work, we combine it with ADAM and AMSGRAD to form the SADAM and SAMSGRAD methods.
4. We also provide theoretical guarantees of our methods, which enjoy better convergence speed than ADAM and recover the same convergence rate as SGD in terms of the maximum iteration T as $O(1/\sqrt{T})$ rather than the known result: $O(\log(T)/\sqrt{T})$ in [16]. Empirical evaluations show that our methods obviously increase test accuracy, and outperform many AGMs and even S-Momentum in multiple deep learning models.

2. Preliminaries

Notations.

For any vectors $a, b \in \mathbb{R}^d$, we use a ⊙ b for element-wise product, a^2 for element-wise square, $\sqrt{a}$ for element-wise square root, a/b for element-wise division; we use a^k to denote element-wise power of k, and $\|a\|$ to denote its l_2-norm. We use $\langle a, b \rangle$ to denote their inner product, $max\{a, b\}$ to compute element-wise maximum. e is the Euler number, log(·) denotes logarithm function with base e, and $O(\cdot)$ to hide constants which do not rely on the problem parameters.

[1]The PADAM in [19] actually used AMSGRAD, and for clear comparison, we named it PAMSGRAD. In our experiments, we also compared with the ADAM that used the A-LR formula with p, which we named PADAM.

Optimization Terminology.

In convex setting, the optimality gap, $f(x_t) - f^*$, is examined where x_t is the iterate at iteration t, and f^* is the optimal value attained at x^* assuming that f does have a minimum. When $f(x_t) - f^* \leq \delta$, it is said that the method reaches an optimal solution with δ-accuracy. However, in the study of AGMs, the average regret $\frac{1}{T}\sum_{t=1}^{T}\left(f(x_t) - f^*\right)$ (where the maximum iteration number T is pre-specified) is used to approximate the optimality gap to define δ-accuracy. Our analysis moves one step further to examine if $f\left(\frac{1}{T}\sum_{t=1}^{T} x_t\right) - f^* \leq \delta$ by applying Jensen's inequality to the regret.

In nonconvex setting, finding the global minimum or even local minimum is NP-hard, so optimality gap is not examined. Rather, it is common to evaluate if a first-order stationary point has been achieved [20, 12, 14]. More precisely, we evaluate if $E\left[\|\nabla f(x_t)\|^2\right] \leq \delta$ (e.g., in the analysis of SGD [1]). The convergence rate of SGD is $O\left(1/\sqrt{T}\right)$ in both non-strongly convex and nonconvex settings. Requiring $O\left(1/\sqrt{T}\right) \leq \delta$ yields the maximum number of iterations $T = O(1/\delta^2)$. Thus, SGD can obtain a δ-accurate solution in $O(1/\delta^2)$ steps in non-strongly convex and nonconvex settings. Our results recover the rate of SGD and S-Momentum in terms of T.

Assumption 1. *The loss f_i and the objective f satisfy:*

1. ***L-smoothness***. $\forall x, y \in \mathbb{R}^d, \forall i \in \{1, \ldots, n\}, \|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$.
2. ***Gradient bounded***. $\forall x \in \mathbb{R}^d, \forall i \in \{1, \ldots, n\}, \|\nabla f_i(x)\| \leq G,\ G \geq 0$.
3. ***Variance bounded***. $\forall x \in \mathbb{R}^d, t \geq 1,\ E[g_t] = \nabla f(x_t), E\left[\|g_t - \nabla f(x_t)\|^2\right] \leq \sigma^2$.

Definition 1. *Suppose f has the global minimum, denoted as $f^* = f(x^*)$. Then for any $x, y \in \mathbb{R}^d$,*

1. ***Non-strongly convex***. $f(y) \geq f(x) + \nabla f(x)^T(y - x)$.
2. ***Polyak-Łojasiewicz (P-L) condition***. $\exists \lambda > 0$ *such that* $\|\nabla f(x)\|^2 \geq 2\lambda\left(f(x) - f^*\right)$.
3. ***Strongly convex***. $\exists\ \mu > 0$ *such that* $f(y) \geq f(x) + \nabla f(x)^T(y - x) + \frac{\mu}{2}\|y - x\|^2$.

3. Our New Analysis of Adam

First, we empirically observe that Adam has anisotropic A-LR caused by ϵ, which may lead to poor generalization performance. Second, we theoretically show Adam method is sensitive to ϵ, supporting observations in previous work.

3.1. Anisotropic A-LR.

We investigate how the A-LR in Adam varies over time and across problem dimensions, and plot four examples in Figure 1 (more figures in Appendix) where we run Adam to optimize a convolutional neural network (CNN) on the MNIST dataset, and ResNets or DenseNets on

the CIFAR-10 dataset. The curves in Figure 1 exhibit very irregular shapes, and the median value is hardly placed in the middle of the range, the range of A-LR across the problem dimensions is anisotropic for AGMs. As a general trend, the A-LR becomes larger when v_t approaches 0 over iterations. The elements in the A-LR vary significantly across dimensions and there are always some coordinates in the A-LR of AGMs that reach the maximum 10^8 determined by ϵ (because we use $\epsilon = 10^{-8}$ in ADAM).

This anisotropic scale of A-LR across dimensions makes it difficult to determine the B-LR, η. On the one hand, η should be set small enough so that the LR $\frac{\eta}{\sqrt{v_t}+\epsilon}$ is appropriate, or otherwise some coordinates will have very large updates because the corresponding A-LR's are big, likely resulting in performance oscillation [21]. This may be due to that exponential moving average of past gradients is different, hence the speed of m_t diminishing to zero is different from the speed of $\sqrt{v_t}$ diminishing to zero. Besides, noise generated in stochastic algorithms has nonnegligible influence to the learning process. On the other hand, very small η may harm the later stage of the learning process since the small magnitude of m_t multiplying with a small step size (at some coordinates) will be too small to escape sharp local minimal, which has been shown to lead to poor generalization [22, 23, 24]. Further, in many deep learning tasks, stage-wise policies are often taken to decay the LR after several epochs, thus making the LR even smaller. To address the dilemma, it is essential to control the A-LR, especially when stochastic gradients get close to 0.

By analyzing previous modified AGMs that aim to close the generalization gap, we find that all these works can be summarized into one technique: constraining the A-LR, $1/\left(\sqrt{v_t}+\epsilon\right)$, to a reasonable range. Based on the observation of anisotropic A-LR, we propose a more effective way to calibrate the A-LR according to an activation function rather than hard-thresholding the A-LR at all coordinates, empirically improve generalization performance with theoretical guarantees of optimization.

3.2. Sensitive to ϵ.

As a hyper-parameter in AGMs, ϵ is originally introduced to avoid the zero denominator issue when v_t goes to 0, and has never been studied in the convergence analysis of AGMs. However, it has been empirically observed that AGMs can be sensitive to the choice of ϵ in [17, 14]. As shown in Figure 1, a smaller $\epsilon = 10^{-8}$ leads to a wide span of the A-LR across the different dimensions, whereas a bigger $\epsilon = 10^{-3}$ as used in YOGI, reduces the span. To better learn the effect caused by sensitive ϵ, we conduct experiments in multiple datasets and results are shown in Table 1 and 2. The setting of ϵ is the main force causing anisotropy, unsatisfied, there has no theoretical result explains the effect of ϵ on AGMs. Inspired by our observation, we believe that the current convergence analysis for ADAM is not complete if omitting ϵ.

Most of the existing convergence analysis follows the line in [12] to first project the sequence of the iterates into a minimization problem as $x_{t+1} = x_t - \frac{\eta}{\sqrt{v_t}} m_t = \min_x \left\| v_t^{1/4} \left(x - \left(x_t - \frac{\eta}{\sqrt{v_t}} m_t \right) \right) \right\|$, and then examine if $\left\| v_t^{1/4} (x_{t+1} - x^*) \right\|$ decreases over iterations. Hence, ϵ is not discussed in this line of proof because it is

not included in the step size. In our later convergence analysis section, we introduce an important lemma, bounded A-LR, and by using the bounds of the A-LR (specifically, the lower bound μ_1 and upper bound μ_2 both containing ϵ for ADAM), we give a new general framework of prove (details in Appendix) to show the convergence rate for reaching an x that satisfies $E\left[\|\nabla f(x_t)\|^2\right] \leq \delta$ in the nonconvex setting. Then, we also derive the optimality gap from the stationary point in the convex and P-L settings (strongly convex).

Theorem 3.1. ***[Nonconvex]*** *Suppose $f(x)$ is a nonconvex function that satisfies Assumption 1. Let $\eta_t = \eta = O\left(\frac{1}{\sqrt{T}}\right)$,* ADAM *has*

$$\min_{t=1,\ldots,T} E\left[\|\nabla f(x_t)\|^2\right] \leq O\left(\frac{1}{\epsilon^2\sqrt{T}} + \frac{d}{\epsilon T} + \frac{d}{\epsilon^2 T\sqrt{T}}\right).$$

Theorem 3.2. ***[Non-strongly Convex]*** *Suppose $f(x)$ is a convex function that satisfies Assumption 1. Assume that $\forall t$, $E[\|x_t - x^*\| \leq D$, for any $m \neq n$, $E\left[\|x_m - x_n\|\right] \leq D_\infty$, let $\eta_t = \eta = O\left(\frac{1}{\sqrt{T}}\right)$,* ADAM *has convergence rate $f(\bar{x}_t) - f^* \leq O\left(\frac{d}{\epsilon^2\sqrt{T}}\right)$, where $\bar{x}_t = \frac{1}{T}\sum_{t=1}^{T} x_t$.*

Theorem 3.3. ***[P-L Condition]*** *Suppose $f(x)$ has P-L condition (with parameter λ) holds under convex case, satisfying Assumption 1. Let $\eta_t = \eta = O\left(\frac{1}{T^2}\right)$,* ADAM *has the convergence rate: $E\left[f(x_{T+1}) - f^*\right] \leq \left(1 - \frac{2\lambda\mu_1}{T^2}\right)^T E\left[f(x_1) - f^*\right] + O\left(\frac{1}{T}\right)$,*

The P-L condition is weaker than strongly convex, and for the strongly convex case, we also have:

Corollary 3.3.1. ***[Strongly Convex]*** *Suppose $f(x)$ is μ-strongly convex function that satisfies Assumption 1. Let $\eta_t = \eta = O\left(\frac{1}{T^2}\right)$,* ADAM *has the convergence rate:*

$$E\left[f(x_{T+1}) - f^*\right] \leq \left(1 - \frac{2\mu\mu_1}{T^2}\right)^T E\left[f(x_1) - f^*\right] + O\left(\frac{1}{T}\right)$$

This is the first time to theoretically include ϵ into analysis. As expected, the convergence rate of ADAM is highly related with ϵ. A bigger ϵ will enjoy a better convergence rate since ϵ will dominate the A-LR and behaves like SMomentum; A smaller ϵ will preserve stronger "adaptivity", we need to find a better way to control ϵ.

4. The Proposed Algorithms

We propose to use activation functions to calibrate AGMs, and specifically focus on using *softplus* funciton on top of ADAM and AMSGRAD methods.

4.1. Activation Functions Help

Activation functions (such as sigmoid, ELU, tanh) transfer inputs to outputs are widely used in deep learning area. As a well-studied activation function, $softplus(x) = \frac{1}{\beta}\log\left(1 + e^{\beta x}\right)$ is known to keep large values unchanged (behaved like function $y = x$) while smoothing out small values (see Figure 2 (a)). The target magnitude to be smoothed out can be adjusted by a hyper-parameter $\beta \in \mathbb{R}$. In our new algorithms, we introduce $softplus\left(\sqrt{v_t}\right) = \frac{1}{\beta}\log\left(1 + e^{\beta \cdot \sqrt{v_t}}\right)$ to smoothly calibrate the A-LR. This calibration brings the following benefits: (1) constraining extreme large-valued A-LR in some coordinates (corresponding to the small-values in v_t) while keeping others untouched with appropriate β. For the undesirable large values in the A-LR, the *softplus* function condenses them smoothly instead of hard thresholding. For other coordinates, the A-LR largely remains unchanged; (2) removing the sensitive parameter ϵ because the *softplus* function can be lower-bounded by a nonzero number when used on non-negative variables, $softplus(\cdot) \geq \frac{1}{\beta}\log 2$.

After calibrating $\sqrt{v_t}$ with a *softplus* function, the anisotropic A-LR becomes much more regulated (see Figure 3 and Appendix), and we clearly observe improved test accuracy (Figure 2 (b) and more figures in Appendix). We name this method "SADAM" to represent the calibrated ADAM with *softplus* function, here we recommend using *softplus* function but it is not limited to that, and the later theoretical analysis can be easily extended to other activation functions. More empirical evaluations have shown that the proposed methods significantly improve the generalization performance of ADAM and AMSGRAD.

4.2. Calibrated AGMs

With activation function, we develop two new variants of AGMs: SADAM and SAMSGRAD (Algorithms 1 and 2), which are developed based on ADAM and AMSGRAD respectively.

Algorithm 1 SADAM

Input: $x_1 \in \mathbb{R}^d$, learning rate $\{\eta_t\}_{t=1}^{T}$, parameters $0 \leq \beta_1, \beta_2 < 1$, β.

Initialize $m_0 = 0,\ v_0 = 0$

for $t = 1$ to T **do**

Compute stochastic gradient g_t

$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$

$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$

$$x_{t+1} = x_t - \frac{\eta_t}{softplus\left(\sqrt{v_t}\right)} \odot m_t$$

end for

Algorithm 2 SAMSGRAD

Input: $x_1 \in \mathbb{R}^d$, learning rate $\{\eta_t\}_{t=1}^T$, parameters $0 \le \beta_1, \beta_2 < 1$, β.

Initialize $m_0 = 0,\ \tilde{v}_0 = 0$

for $t = 1$ to T **do**

Compute stochastic gradient g_t

$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$

$\tilde{v}_t = \beta_2 \tilde{v}_{t-1} + (1-\beta_2) g_t^2$

$v_t = max\{v_{t-1}, \tilde{v}_t\}$

$$x_{t+1} = x_t - \frac{\eta_t}{softplus\left(\sqrt{v_t}\right)} \odot m_t$$

end for

The key step lies in the way to design the adaptive functions, instead of using the generalized square root function only, we apply $softplus(\cdot)$ on top of the square root of the second-order momentum, which serves to regulate A-LR's anisotropic behavior and replace the tolerance parameter ϵ by the hyper-parameter β used in the *softplus* function.

In our algorithms, the hyper-parameters are recommended as $\beta_1 = 0.9$, $\beta_2 = 0.999$. For clarity, we omit the bias correction step proposed in the original Adam. However, our arguments and theoretical analysis are applicable to the bias correction version as well [6, 25, 14]. Using the *softplus* function, we introduce a new hyper-parameter β, which performs as a controller to smooth out anisotropic A-LR, and connect the Adam and S-Momentum methods automatically. When β is set to be small, Sadam and SAMSGrad perform similarly to S-Momentum; when β is set to be big, $softplus\left(\sqrt{v_t}\right) = \frac{1}{\beta}\log\left(1 + e^{\beta \cdot \sqrt{v_t}}\right) \approx \frac{1}{\beta}\log\left(e^{\beta \cdot \sqrt{v_t}}\right) = \sqrt{v_t}$, and the updating formula becomes $x_{t+1} = x_t - \frac{\eta_t}{\sqrt{v_t}} \odot m_t$, which is degenerated into the original AGMs. The hyper-parameter β can be well tuned to achieve the best performance for different datasets and tasks. Based on our empirical observations, we recommend to use $\beta = 50$.

As a calibration method, the *softplus* function has better adaptive behavior than simply setting. More precisely, when ϵ is large or β is small, Adam and AMSGrad amount to S-Momentum, but when ϵ is small as commonly suggested 10^{-8} or β is taken large, the two methods are different because comparing Figure 1 and 3 yields that Sadam has more regulated A-LR distribution. The proposed calibration scheme regulates the massive range of A-LR back down to a moderate scale. The median of A-LR in different dimensions is now well positioned to the middle of the 25–75 percentile zone. Our approach opens up a new direction to examine other activation functions (not limited to the *softplus* function) to calibrate the A-LR.

The proposed Sadam and SAMSGrad can be treated as members of a class of AGMs that use the *softplus* (or another suitable activation) function to better adapt the step size. It can

be readily combined with any other AGM, e.g., Rmsrop, Yogi, and PAdam. These methods may easily go back to the original ones by choosing a big β.

5. Convergence Analysis

We first demonstrate an important lemma to highlight that every coordinate in the A-LR is both upper and lower bounded at all iterations, which is consistent with empirical observations (Figure 1), and forms the foundation of our proof.

Lemma 5.1. ***[Bounded A-LR]*** *With Assumption 1, for any* $t \geq 1$, $j \in [1,d]$, $\beta_2 \in [0,1]$, *and in* Adam, β *in* Sadam, *anisotropic A-LR is bounded in AGMs,* Adam *has* (μ_1,μ_2)*-bounded A-LR:*

$$\mu_1 \leq \frac{1}{\sqrt{v_{t,j}} + \epsilon} \leq \mu_2,$$

Sadam *has* (μ_3,μ_4)*-bounded A-LR:*

$$\mu_3 \leq \frac{1}{softplus\left(\sqrt{v_{t,j}}\right)} \leq \mu_4,$$

where $0 < \mu_1 \leq \mu_2$, *and* $0 < \mu_3 \leq \mu_4$

Remark 5.2. *Besides the square root function and* softplus *function, the A-LR calibrated by any positive monotonically increasing function can be bounded. All of the bounds can be shown to be related to* ϵ *or* β *(see* Appendix). *Bounded A-LR is an essential foundation in our analysis, we provide a different way of proof from previous works, and the proof procedure can be easily extended to other gradient methods as long as bounded LR is satisfied.*

Remark 5.3. *These bounds can be applied to all AGMs, including* Adagrad. *In fact, the lower bounds actually are not the same in* Adam *and* Adagrad, *because* Adam *will have smaller* $\sqrt{v_{t,j}}$ *due to moment decay parameter* β_2. *To achieve a unified result, we use the same relaxation to derive the fixed lower bound* μ_1.

We now describe our main results of Sadam (and SAMSGrad) in the nonconvex case, we clearly show that similar to Theorem 3.1, the convergence rate of Sadam is related to the bounds of the A-LR. Our methods have improved the convergence rate of Adam when comparing self-contained parameters ϵ and β.

Theorem 5.4. ***[Nonconvex]*** *Suppose* $f(x)$ *is a nonconvex function that satisfies Assumption 1. Let* $\eta_t = \eta = O\left(\frac{1}{\sqrt{T}}\right)$, Sadam *method has*

$$\min_{t=1,\ldots,T} E\left[\|\nabla f(x_t)\|^2\right] \leq O\left(\frac{\beta^2}{\sqrt{T}} + \frac{d\beta}{T} + \frac{d\beta^2}{T\sqrt{T}}\right).$$

Remark 5.5. *Compared with the rate in Theorem 3.1, the convergence rate of* SADAM *relies on β, which can be a much smaller number ($\beta = 50$ as recommended) than $\frac{1}{\epsilon}$ (commonly $\epsilon = 10^{-8}$ in AGMs), showing that our methods have a better convergence rate than* ADAM. *When β is huge,* SADAM*'s rate is comparable to the classic* ADAM. *When β is small, the convergence rate will be $O\left(\frac{1}{\sqrt{T}}\right)$ which recovers that of SGD* [1].

Corollary 5.5.1. *Treat ϵ or β as a constant, then the* ADAM, SADAM *(and* SAMSGRAD*) methods with fixed L, σ, G, β_1, and $\eta = O\left(\frac{1}{\sqrt{T}}\right)$, have complexity of $O\left(\frac{1}{\sqrt{T}}\right)$, and thus call for $O\left(\frac{1}{\delta^2}\right)$ iterations to achieve δ-accurate solutions.*

Theorem 5.6. ***[Non-strongly Convex]*** *Suppose $f(x)$ is a convex function that satisfies Assumption 1. Assume that $E[\|x_t - x^*\|] \le D$, $\forall t$, and $E\left[\|x_m - x_n\|\right] \le D_\infty$, $\forall\ m \ne n$, let $\eta_t = \eta = O\left(\frac{1}{\sqrt{T}}\right)$,* SADAM *has $f(\bar{x}_t) - f^* \le O\left(\frac{1}{\sqrt{T}}\right)$, where $\bar{x}_t = \frac{1}{T}\sum_{t=1}^{T} x_t$.*

The accurate convergence rate will be $O\left(\frac{d}{\epsilon^2\sqrt{T}}\right)$ for ADAM and $O\left(\frac{d\beta^2}{\sqrt{T}}\right)$ for SADAM with fixed L, σ, G, β_1, D, D_∞. Some works may specify additional sparsity assumptions on stochastic gradients, and in other words, require that $\sum_{t=1}^{T}\sum_{j=1}^{d}\|g_{t,j}\| \ll \sqrt{dT}$ [5, 12, 15, 19] to reduce the order from d to $\sqrt{d}$. Some works may use the element-wise bounds σ_j or G_j, and apply $\sum_{j=1}^{d}\sigma_j = \sigma$, and $\sum_{j=1}^{d} G_j = G$ to hide d. In our work, we do not assume sparsity, so we use σ and G throughout the proof. Otherwise, those techniques can also be used to hide d from our convergence rate.

Corollary 5.6.1. *If ϵ or β is treated as constants, then* ADAM, SADAM *(and* SAMSGRAD*) methods with fixed L, σ, G, β_1, and $\eta = O\left(\frac{1}{\sqrt{T}}\right)$ in the convex case will call for $O\left(\frac{1}{\delta^2}\right)$ iterations to achieve δ-accurate solutions.*

Theorem 5.7. ***[P-L Condition]*** *Suppose $f(x)$ satisfies the P-L condition (with parameter λ) and Assumption 1 in the convex case. Let $\eta_t = \eta = O\left(\frac{1}{T^2}\right)$,* SADAM *has:*

$$E\left[f(x_T + 1) - f^*\right] \le \left(1 - \frac{2\lambda\mu_3}{T^2}\right)^T E\left[f(x_1) - f^*\right] + O\left(\frac{1}{T}\right).$$

Corollary 5.7.1. ***[Strongly Convex]*** *Suppose $f(x)$ is μ-strongly convex function that satisfies Assumption 1. Let $\eta_t = \eta = O\left(\frac{1}{T^2}\right)$,* SADAM *has the convergence rate:*

$$E\left[f(x_T + 1) - f^*\right] \le \left(1 - \frac{2\mu\mu_3}{T^2}\right)^T E\left[f(x_1) - f^*\right] + O\left(\frac{1}{T}\right).$$

In summary, our methods share the same convergence rate as ADAM, and enjoy even better convergence speed if comparing the common values chosen for the parameters ϵ and β. Our convergence rate recovers that of SGD and S-Momentum in terms of T for a small β.

6. Experiments

We compare SADAM and SAMSGRAD against several state-of-the-art optimizers including S-Momentum, ADAM, AMSGRAD, YOGI, PADAM, PAMSGRAD, ADABOUND, and AMSBOUND. More results and architecture details are in Appendix.

Experimental Setup.

We use three datasets for image classifications: MNIST, CIFAR-10 and CIFAR-100 and two datasets for LSTM language models: Penn Treebank dataset (PTB) and the WikiText-2 (WT2) dataset. The MNIST dataset is tested on a CNN with 5 hidden layers. The CIFAR-10 dataset is tested on Residual Neural Network with 20 layers (ResNets 20) and 56 layers (ResNets 56) [9], and DenseNets with 40 layers [11]. The CIFAR-100 dataset is tested on VGGNet [26] and Residual Neural Network with 18 layers (ResNets 18) [9]. The Penn Treebank dataset (PTB) and the WikiText-2 (WT2) dataset are tested on 3-layer LSTM models [27].

We train CNN on the MNIST data for 100 epochs, ResNets/DenseNets on CIFAR-10 for 300 epochs, with a weight decay factor of 5×10^{-4} and a batch size of 128, VGGNet/ResNets on CIFAR-100 for 300 epochs, with a weight decay factor of 0.025 and a batch size of 128 and LSTM language models on 200 epochs. For the CIFAR tasks, we use a fixed multi-stage LR decaying scheme: the B-LR decays by 0.1 at the 150-th epoch and 225-th epoch, which is a popular decaying scheme used in many works [28, 18]. For the language tasks, we use a fixed multi-stage LR decaying scheme: the B-LR decays by 0.1 at the 100-th epoch and 150-th epoch. All algorithms perform grid search for hyperparameters to choose from {10,1,0.1,0.01,0.001,0.0001} for B-LR, {0.9,0.99} for β_1 and {0.99,0.999} for β_2. For algorithm-specific hyper-parameters, they are tuned around the recommended values, such as $p \in \left\{\frac{1}{8}, \frac{1}{16}\right\}$ in PADAM and PAMSGRAD. For our algorithms, β is selected from {10,50,100} in SADAM and SAMSGRAD, though we do observe fine-tuning β can achieve better test accuracy most of time. All experiments on CIFAR tasks are repeated for 6 times to obtain the mean and standard deviation for each algorithm.

Image Classification Tasks.

As a sanity check, experiment on MNIST has been done and its results are in Figure 4, which shows the learning curve for all baseline algorithms and our algorithms on both training and test datasets. As expected, all methods can reach the zero loss quickly, while for test accuracy, our SAMSGRAD shows increase in test accuracy and outperforms competitors within 50 epochs.

Using the PyTorch framework, we first run the ResNets 20 model on CIFAR10 and results are shown in Table 3. The original ADAM and AMSGRAD have lower test accuracy in comparison with S-Momentum, leaving a clear generalization gap exactly same as what is

previously reported. For our methods, SADAM and SAMSGRAD clearly close the gap, and SADAM achieves the best test accuracy among competitors. We further test all methods with CIFAR10 on ResNets 56 with greater network depth, and the overall performance of each algorithm has been improved. For the experiments with DenseNets, we use a DenseNet with 40 layers and a growth rate $k = 12$ without bottleneck, channel reduction, or dropout. The results are reported in the last column of Table 3, SAMSGRAD still achieves the best test performance, and the proposed two methods largely improve the performance of ADAM and AMSGRAD and close the gap with S-Momentum.

Furthermore, two popular CNN architectures: VGGNet [26] and ResNets18 [9] are tested on CIFAR-100 dataset to compare different algorithms. Results can be found in Figure 5 and repeated results are in Appendix. Our proposed methods again perform slightly better than S-Momentum in terms of test accuracy.

LSTM Language Models.

Observing the significant improvements in deep neural networks for image classification tasks, we further conduct experiments on the language models with LSTM. For comparing the efficiency of our proposed methods, two LSTM models over the Penn Treebank dataset (PTB) [29] and the WikiText-2 (WT2) dataset [30] are tested. We present the single-model perplexity results for both our proposed methods and other competitive methods in Figure 6 and our methods achieve both fast convergence and best generalization performance.

In summary, our proposed methods show great efficacy on several standard benchmarks in both training and testing results, and outperform most optimizers in terms of generalization performance.

7. Conclusion

In this paper, we study adaptive gradient methods from a new perspective that is driven by the observation that the adaptive learning rates are anisotropic at each iteration. Inspired by this observation, we propose to calibrate the adaptive learning rates using an activation function, and in this work, we examine *softplus* function. We combine this calibration scheme with ADAM and AMSGRAD methods and empirical evaluations show obvious improvement on their generalization performance in multiple deep learning tasks. Using this calibration scheme, we replace the hyper-parameter ϵ in the original methods by a new parameter β in the *softplus* function. A new mathematical model has been proposed to analyze the convergence of adaptive gradient methods. Our analysis shows that the convergence rate is related to ϵ *or* β, which has not been previously revealed, and the dependence on ϵ *or* β helps us justify the advantage of the proposed methods. In the future, the calibration scheme can be designed based on other suitable activation functions, and used in conjunction with any other adaptive gradient method to improve generalization performance.

Acknowledgments

This work was funded by NSF grants CCF-1514357, DBI-1356655, and IIS1718738 to Jinbo Bi, who was also supported by NIH grants K02-DA043063 and R01-DA037349.

Appendix

A.1. Architecture Used in Our Experiments

Here we mainly introduce the MNIST architecture with Pytorch used in our empirical study, ResNets and DenseNets are well-known architectures used in many works and we do not include details here.

layer	layer setting
F.relu(self.conv1(x))	self.conv1 = nn.Conv2d(1, 6, 5)
F.max pool2d(x, 2, 2)	
F.relu(self.conv2(x))	self.conv2 = nn.Conv2d(6, 16, 5)
x.view(−1, 16*4)	
F.relu(self.fc1(x))	self.fc1 = nn.Linear(16*4*4, 120)
x= F.relu(self.fc2(x))	self.fc2 = nn.Linear(120, 84)
x = self.fc3(x)	self.fc3 = nn.Linear(84, 10)
F.log softmax(x, dim=1)	

B.2. More Empirical Results

In this section, we perform multiply experiments to study the property of anisotropic A-LR exsinting in AGMs and the performance of *softplus* function working on A-LR. We first show the A-LR range of popular ADAM-type methods, then present how the parameter β in SADAM and SAMSGRAD reduce the range of A-LR and improve both training and testing performance.

B.2.1. A-LR Range of AGMs

Besides the A-LR range of ADAM method, which has shown in main paper, we further want to study more other ADAM-type methods, and do experiments focus on AMSGRAD, PADAM, and PAMSGRAD on different tasks (Figure B.2.1, B.2.2, and B.2.3). AMSGRAD also has extreme large-valued coordinates, and will encounter the "small learning rate dilemma" as well as ADAM. With partial parameter p, the value range of A-LR can be largely narrow down, and the maximum range will be reduced around 10^2 with PADAM, and less than 10^2 with PAMSGRAD. This reduced range, avoiding the "small learning rate dilemma", may help us understand what "trick" works on ADAM's A-LR can indeed improve the generalization performance. Besides, the range of A-LR in YOGI, ADABOUND and AMSBOUND will be reduced or controlled by specific ϵ or *clip* function, we don't show more information here.

B.2.2. Parameter β Reduces the Range of A-LR

The main paper has discussed about *softplus* function, and mentions that it does help to constrain large-valued coordinates in A-LR while keep others untouched, here we give more empirical support. No matter how does β set, the modified A-LR will have a reduced range. By setting various β's, we can find an appropriate β that performs the best for specific tasks

on datasets. Besides the results of A-LR range of SADAM on MNIST with different choices of β, we also study SADAM and SAMSGRAD on ResNets 20 and DenseNets.

Here we do grid search to choose appropriate β from {10,50,100,200,500,1000}. In summary, with *softplus* fuction, SADAM and SAMSGRAD will narrow down the range of A-LR, make the A-LR vector more regular, avoiding "small learning rate dilemma" and finally achieve better performance.

B.2.3. Parameter β Matters in Both Training and Testing

After studying existing ADAM-type methods, and effect of different β in adjusting A-LR, we focus on the training and testing accuracy of our *softplus* framework, especially SADAM and SAMSGRAD, with different choices of β.

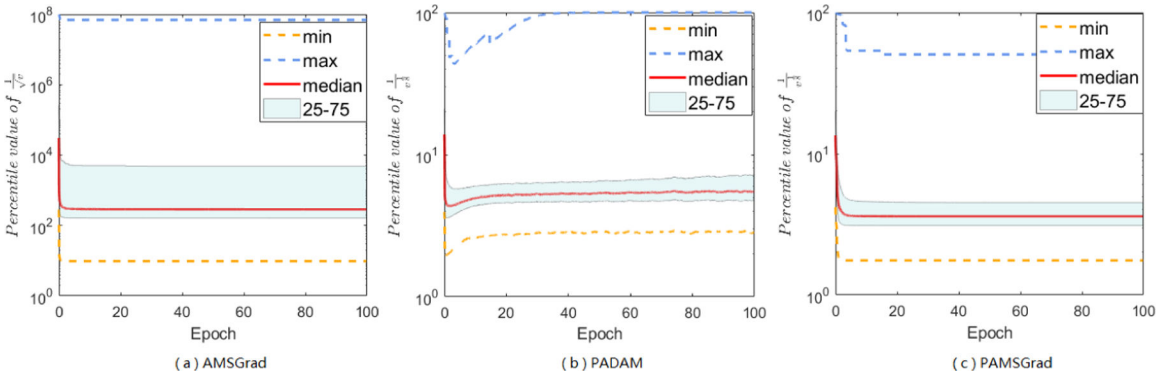

Figure B.2.1:
A-LR range of AMSGRAD (a), PADAM (b), and PAMSGRAD (c) on MNIST.

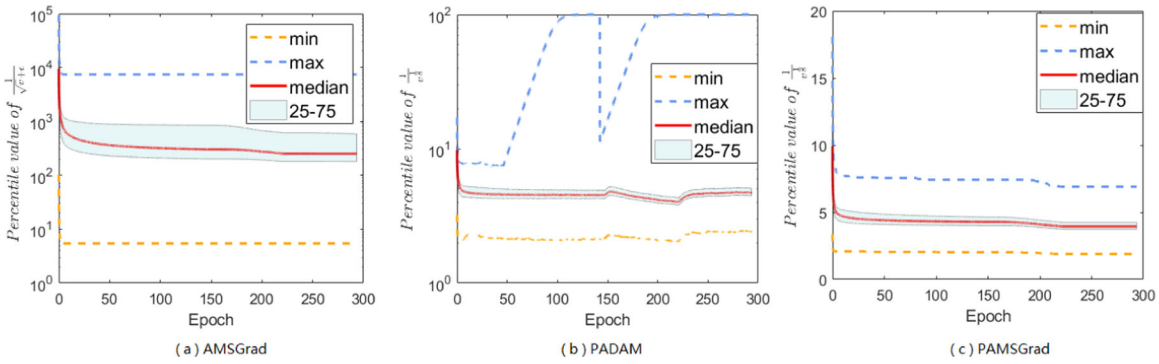

Figure B.2.2:
A-LR range of AMSGRAD (a), PADAM (b), and PAMSGRAD (c) on ResNets 20.

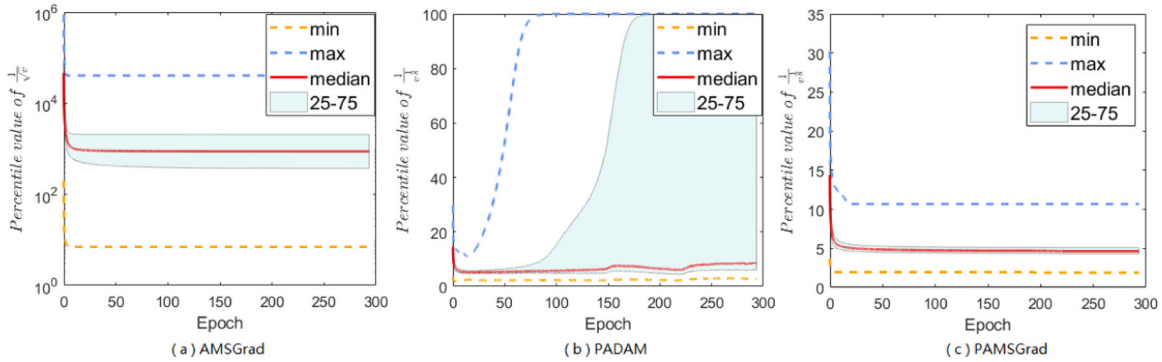

Figure B.2.3:
A-LR range of AMSGRAD (a), PADAM (b), and PAMSGRAD (c) on DenseNets.

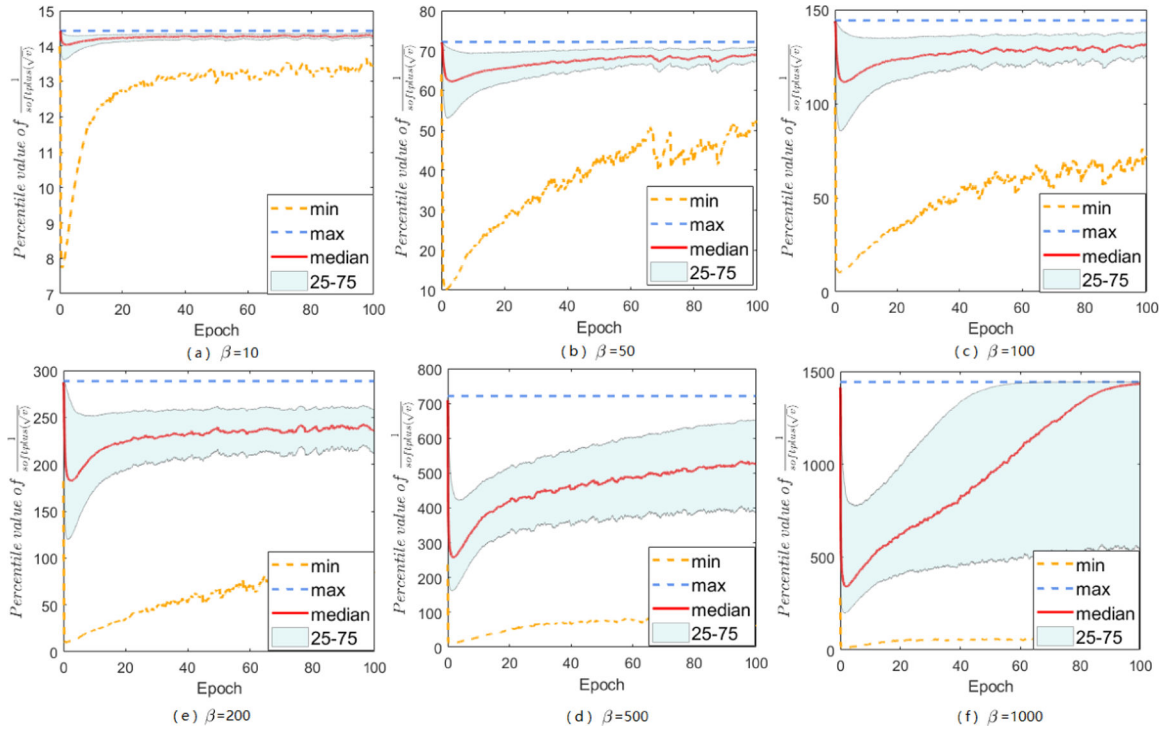

Figure B.2.4:
The range of A-LR: $1/softplus(\sqrt{v_t})$ over iterations for different choices of β. The maximum ranges in all figures are compressed to a reasonable smaller value compared with 10^8.

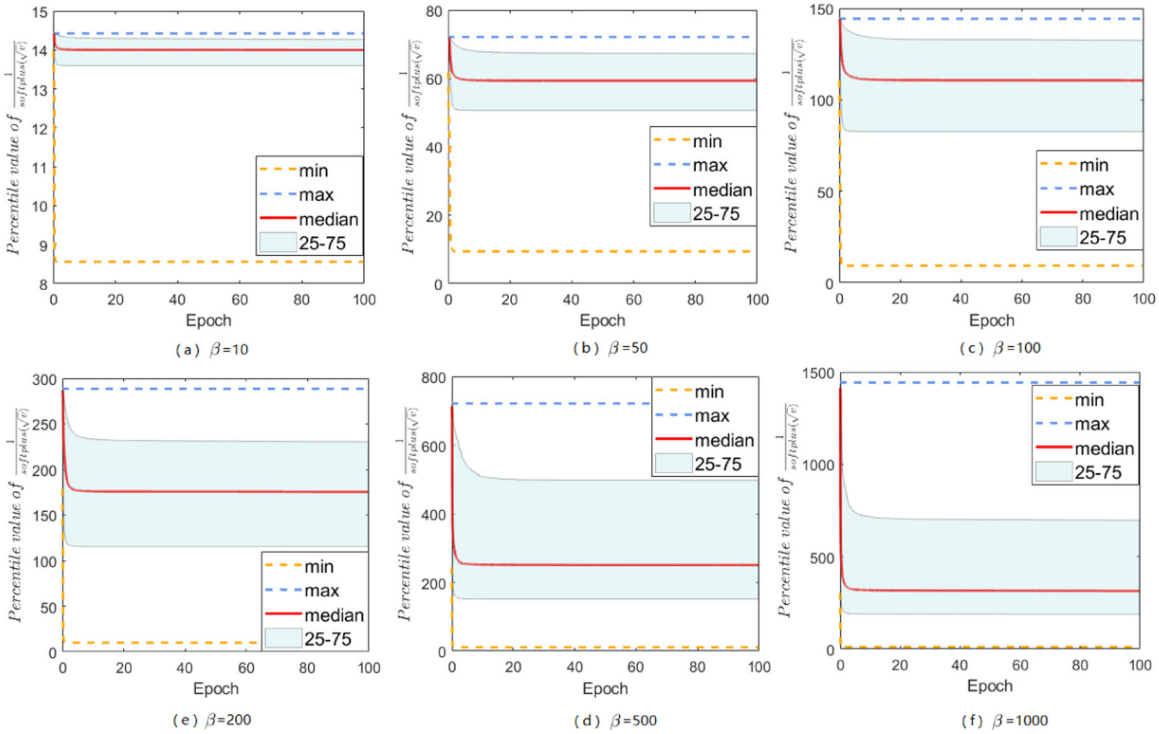

Figure B.2.5:

The range of A-LR: $1/softplus\left(\sqrt{v_t}\right)$, $v_t = max\{v_{t-1}, \tilde{v}_t\}$ over iterations for SAMSGRAD on MNIST with different choice of β. The maximum ranges in all figures are compressed to a reasonable smaller value compared with those of AMSGRAD on MNIST

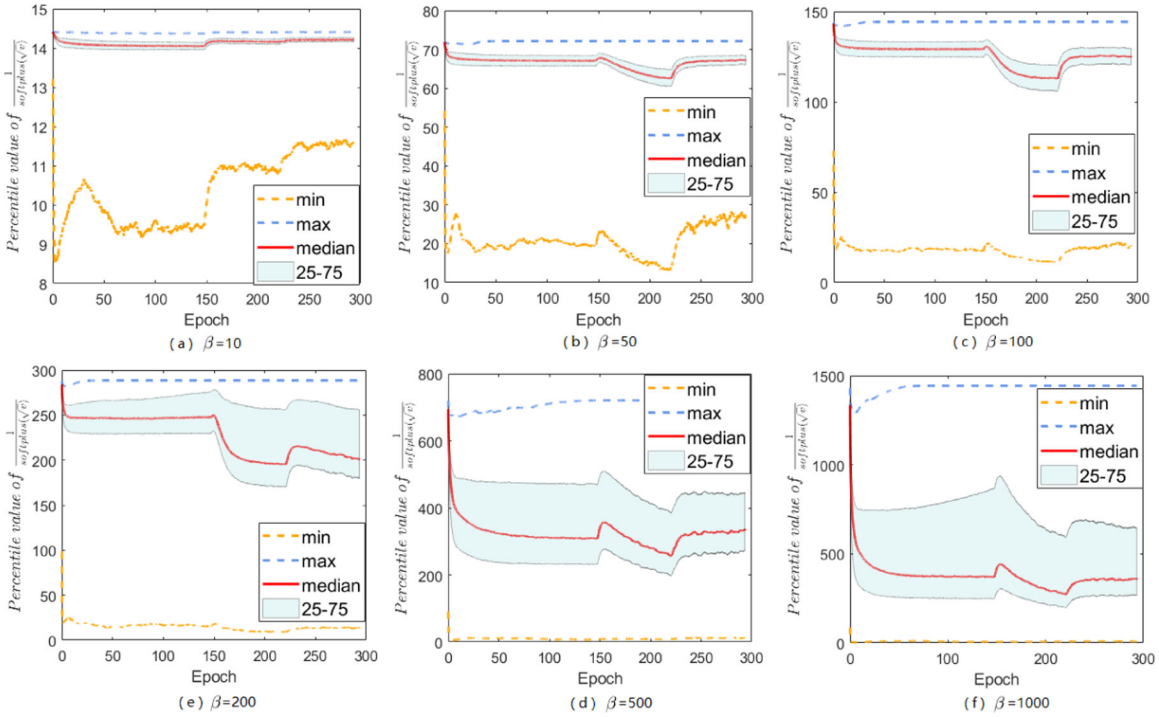

Figure B.2.6:
The range of A-LR: $1/softplus\left(\sqrt{v_t}\right)$ over iterations for SADAM on ResNets 20 with different choices of β.

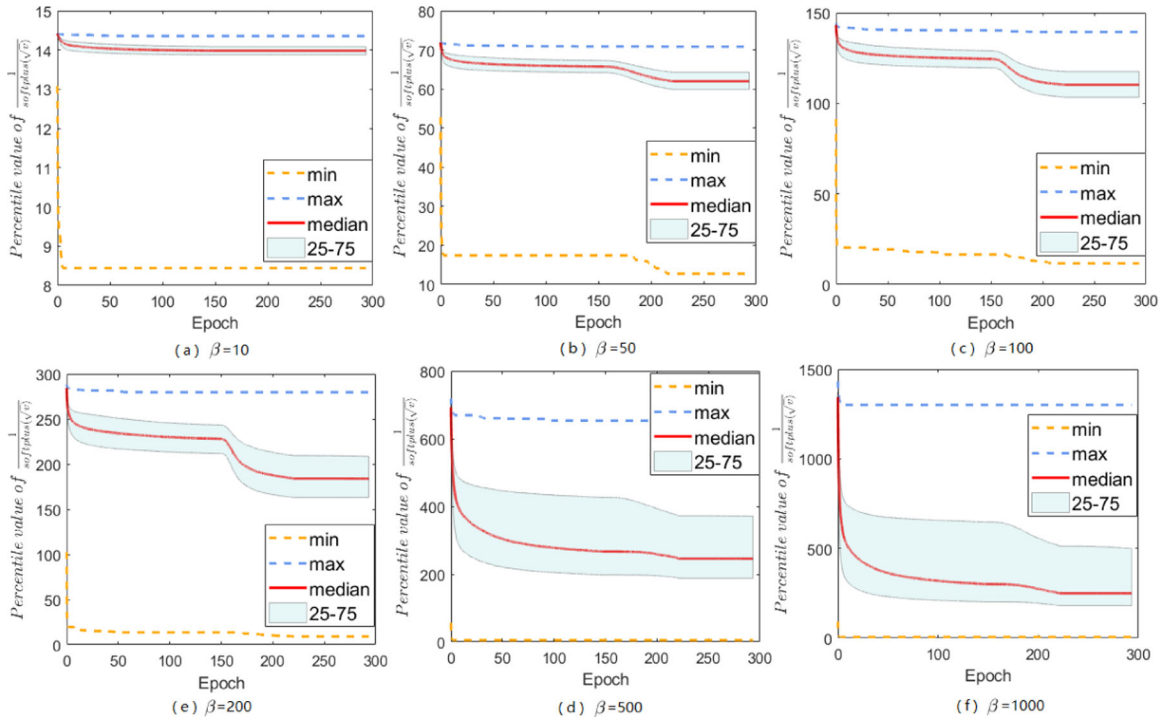

Figure B.2.7:
The range of A-LR: $1/softplus\left(\sqrt{v_t}\right)$, $v_t = max\{v_{t-1}, \tilde{v}_t\}$ over iterations for SAMSGRAD on ResNets 20 with different choices of β.

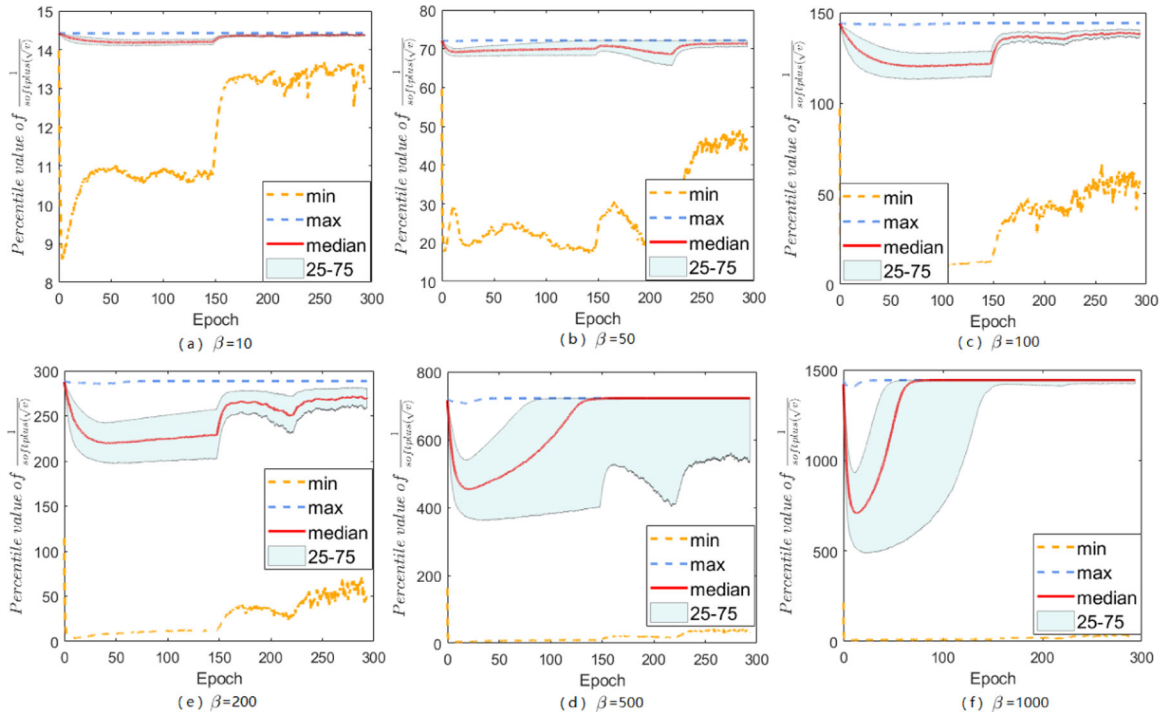

Figure B.2.8:

The range of A-LR: $1/softplus\left(\sqrt{v_t}\right)$ over iterations for SADAM on DenseNets with different choice of β.

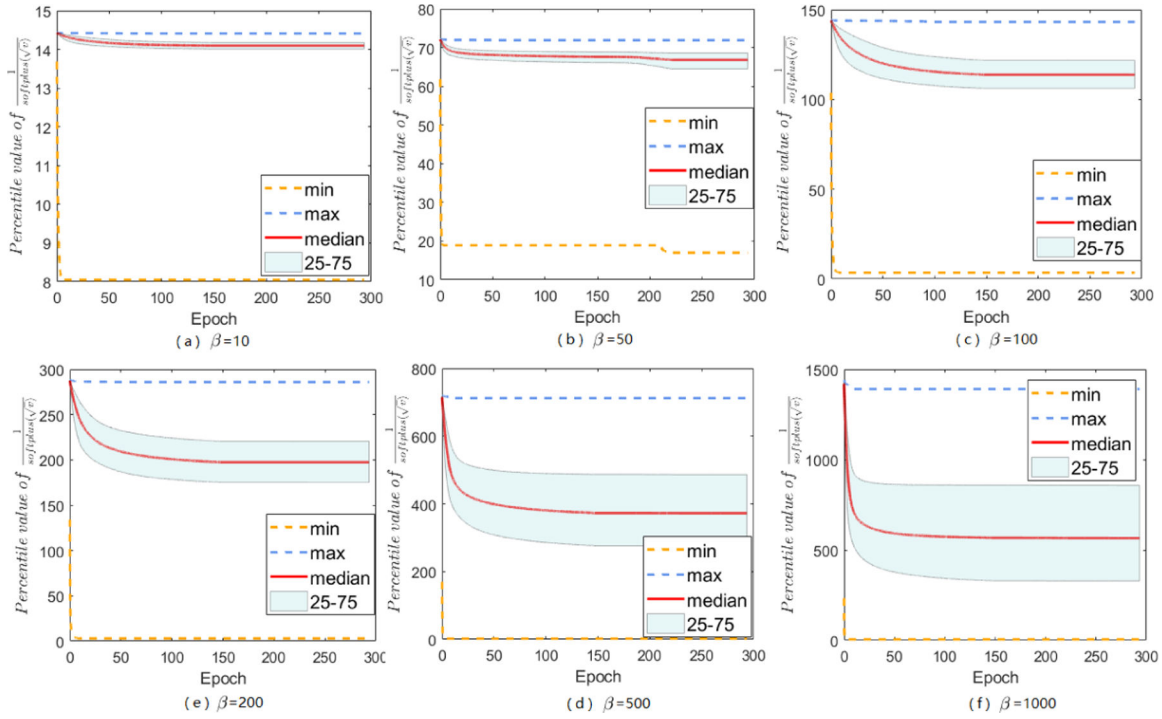

Figure B.2.9:

The range of A-LR: $1/softplus(\sqrt{v_t})$, $v_t = max\{v_{t-1}, \tilde{v}_t\}$ over iterations for SAMSGRAD on DenseNets with different choices of β.

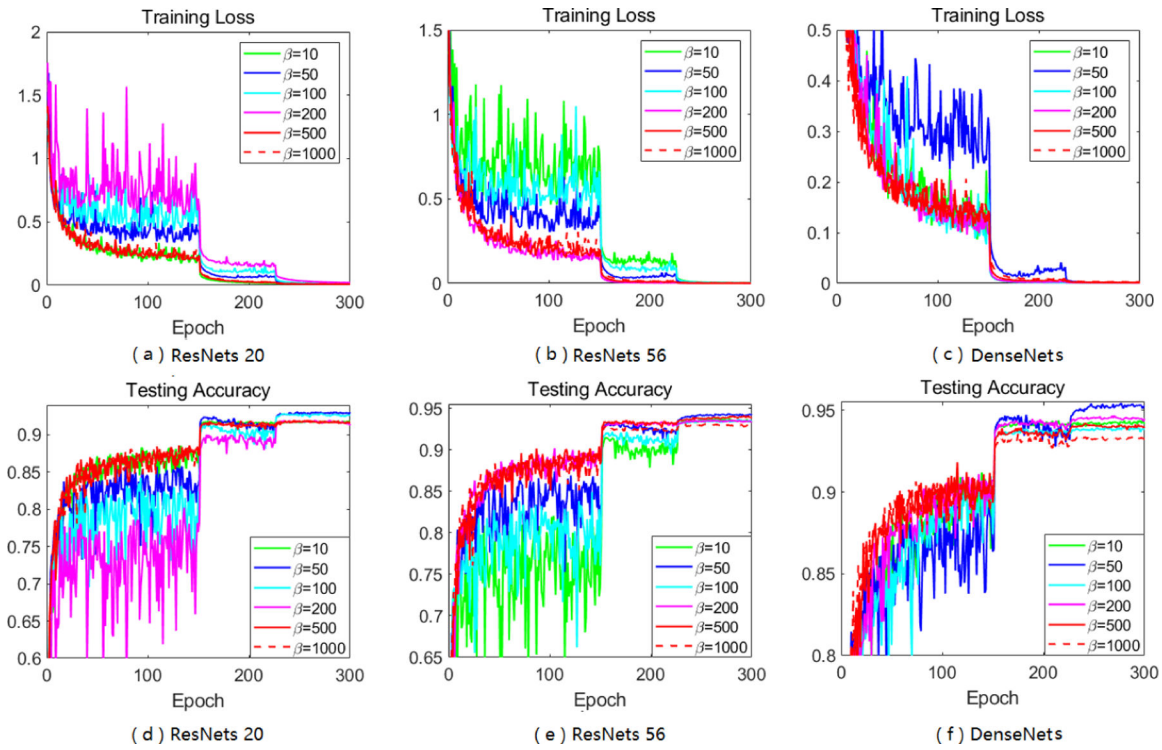

Figure B.2.10:
Performance of SADAM on CIFAR-10 with different choice of β.

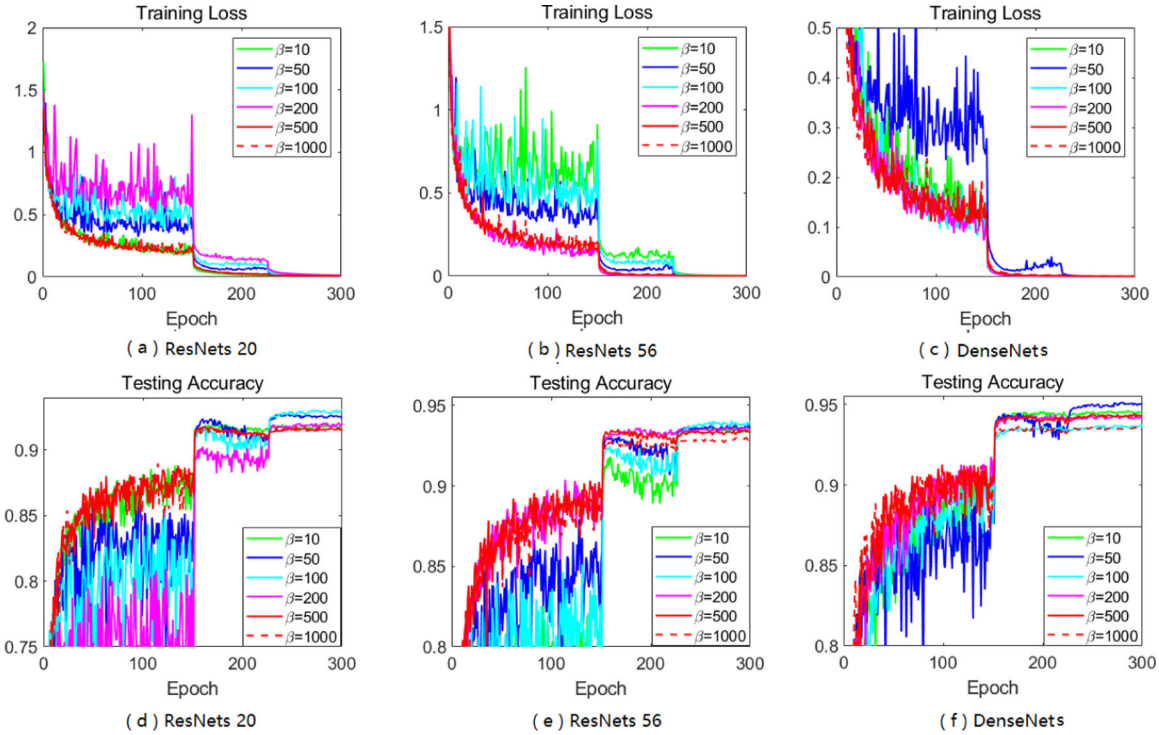

Figure B.2.11:
Performance of SAMSGRAD on CIFAR-10 with different choice of β.

C.3. CIFAR100

Two popular CNN architectures are tested on CIFAR-100 dataset to compare different algorithms: VGGNet [26] and ResNets18 [9]. Besides the figures in main text, we have repeated experiments and show results as follows. Our proposed methods again perform slightly better than S-Momentum in terms of D.4. Theoretical Analysis Details

Table C.3.1:

Test Accuracy(%) of CIFAR100 for VGGNet.

Method	50th epoch	150th epoch	250th epoch	best perfomance
S-Momentum	59.09 ± 2.09	61.25 ± 1.51	76.14 ± 0.12	76.43 ± 0.15
ADAM	60.21 ± 0.81	62.98 ± 0.10	73.81 ± 0.17	74.18 ± 0.15
AMSGRAD	61.00 ± 1.17	63.27 ± 1.18	74.04 ± 0.16	74.26 ± 0.18
PADAM	53.62 ± 1.70	56.02 ± 0.86	75.85 ± 0.20	76.36 ± 0.16
PAMSGRAD	52.49 ± 3.07	57.39 ± 1.40	75.82 ± 0.31	76.26 ± 0.30
ADABOUND	60.27 ± 0.99	60.36 ± 1.71	75.86 ± 0.23	76.10 ± 0.22
AMSBOUND	59.88 ± 0.56	60.11 ± 1.92	75.74 ± 0.23	75.99 ± 0.20
ADAM$^+$	43.59 ± 2.71	44.46 ± 4.39	74.91 ± 0.36	75.58 ± 0.33
AMSGRAD$^+$	44.45 ± 2.83	45.61 ± 3.67	74.85 ± 0.08	75.56 ± 0.24
SADAM	58.59 ± 1.60	61.27 ± 1.67	76.35 ± 0.18	76.64 ± 0.18

Method	50th epoch	150th epoch	250th epoch	best perfomance
SAMSGRAD	59.16 ± 1.20	60.86 ± 0.39	76.27 ± 0.23	76.47 ± 0.26

Table C.3.2:

Test Accuracy(%) of CIFAR100 for ResNets18.

Method	50th epoch	150th epoch	250th epoch	best perfomance
S-Momentum	59.98 ± 1.31	63.32 ± 1.61	77.19 ± 0.36	77.50 ± 0.25
ADAM	63.40 ± 1.42	66.18 ± 1.02	75.68 ± 0.49	76.14 ± 0.24
AMSGRAD	63.16 ± 0.47	66.59 ± 1.42	75.92 ± 0.26	76.32 ± 0.11
PADAM	56.28 ± 0.87	58.71 ± 1.66	77.18 ± 0.21	77.51 ± 0.19
PAMSGRAD	54.34 ± 2.21	58.81 ± 1.95	77.41 ± 0.17	77.67 ± 0.14
ADABOUND	61.13 ± 0.84	64.30 ± 1.84	77.18 ± 0.38	77.50 ± 0.29
AMSBOUND	61.05 ± 1.59	62.04 ± 2.10	77.08 ± 0.19	77.34 ± 0.13
ADAM$^+$	46.5 ± 2.12	48.68 ± 4.06	76.86 ± 0.36	77.19 ± 0.28
AMSGRAD$^+$	49.06 ± 3.23	50.75 ± 2.45	76.58 ± 0.21	76.91 ± 0.12
SADAM	59.00 ± 1.09	62.75 ± 1.03	77.26 ± 0.30	77.61 ± 0.19
SAMSGRAD	59.63 ± 1.27	63.44 ± 1.84	77.31 ± 0.40	77.70 ± 0.31

D.4. Theoretical Analysis Details

We analyze the convergence rate of ADAM and SADAM under different cases, and derive competitive results of our methods. The following table gives an overview of stochastic gradient methods convergence rate under various conditions, in our work we provide a different way of proof compared with previous works and also associate the analysis with hyperparameters of ADAM methods.

D.4.1. Prepared Lemmas

We have a series of prepared lemmas to help with optimization convergence rate analysis, and some of them maybe also used in generalization error bound analysis.

Lemma D.4.1. *For any vectors* $a, b, c \in \mathbb{R}^d$, $< a, b \odot c > \; = \; < a \odot b, c > \; = \; < a \odot \sqrt{b}, c \odot \sqrt{b} >$, *here* $\odot$ *is element-wise product,* $\sqrt{b}$ *is element-wise square root.*

Proof.

$$< a, b \odot c > \; = \; < \begin{pmatrix} a_1 \\ \vdots \\ a_d \end{pmatrix}, \begin{pmatrix} b_1 c_1 \\ \vdots \\ b_d c_d \end{pmatrix} > \; = a_1 b_1 c_1 + \cdots + a_d b_d c_d$$

$$< a \odot b, c > \; = \; < \begin{pmatrix} a_1 b_1 \\ \vdots \\ a_d b_d \end{pmatrix}, \begin{pmatrix} c_1 \\ \vdots \\ c_d \end{pmatrix} > \; = a_1 b_1 c_1 + \cdots + a_d b_d c_d$$

$$< a \odot \sqrt{b}, c \odot \sqrt{b} > \; = \; < \begin{pmatrix} a_1 \sqrt{b_1} \\ \vdots \\ a_d \sqrt{b_d} \end{pmatrix}, \begin{pmatrix} \sqrt{b_1} c_1 \\ \vdots \\ \sqrt{b_d} c_d \end{pmatrix} > \; = a_1 b_1 c_1 + \cdots + a_d b_d c_d$$

□

Lemma D.4.2. *For any vector a, we have.*

$$\left\| a^2 \right\|_\infty \leq \left\| a^2 \right\| . \tag{2}$$

Lemma D.4.3. *For unbiased stochastic gradient, we have*

$$E\left[\left\| g_t \right\|^2 \right] \leq \sigma^2 + G^2 . \tag{3}$$

Proof. From gradient bounded assumption and variance bounded assumption,

$$\begin{aligned} E\left[\left\| g_t \right\|^2 \right] &= E\left[\left\| g_t - \nabla f(x_t) + \nabla f(x_t) \right\|^2 \right] \\ &= E\left[\left\| g_t - \nabla f(x_t) \right\|^2 \right] + \left\| \nabla f(x_t) \right\|^2 \\ &\leq \sigma^2 + G^2 . \end{aligned}$$

□

Lemma D.4.4. *All momentum-based optimizers using first momentum* $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ *will satisfy*

$$E\left[\left\| m_t \right\|^2 \right] \leq \sigma^2 + G^2 . \tag{4}$$

Proof. From the updating rule of first momentum estimator, we can derive

$$m_t = \sum_{i=1}^{t} (1 - \beta_1) \beta_1^{t-i} g_i . \tag{5}$$

Let $\Gamma_t = \sum_{i=1}^{t} \beta_1^{t-i} = \frac{1 - \beta_1^t}{1 - \beta_1}$, by Jensen inequality and Lemma D.4.3,

$$E\left[\|m_t\|^2\right] = E\left[\left\|\sum_{i=1}^{t}(1-\beta_1)\beta_1^{t-i}g_i\right\|^2\right] = \Gamma_t^2 E\left[\left\|\sum_{i=1}^{t}\frac{(1-\beta_1)\beta_1^{t-i}}{\Gamma_t}g_i\right\|^2\right]$$
$$\leq \Gamma_t^2\sum_{i=1}^{t}(1-\beta_1)^2\frac{\beta_1^{t-i}}{\Gamma_t}E\left[\|g_i\|^2\right] \leq \Gamma_t(1-\beta_1)^2\sum_{i=1}^{t}\beta_1^{t-i}\left(\sigma^2+G^2\right)$$
$$\leq \sigma^2+G^2.$$

□

Lemma D.4.5. *Each coordinate of vector* $v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2$ *will satisfy*

$$E\left[v_{t,j}\right] \leq \sigma^2+G^2,$$

where $j \in [1,d]$ *is the coordinate index.*

Proof. From the updating rule of second momentum estimator, we can derive

$$v_{t,j} = \sum_{i=1}^{t}(1-\beta_2)\beta_2^{t-i}g_{i,j}^2 \geq 0. \tag{6}$$

Since the decay parameter $\beta_2 \in [0,1)$, $\sum_{i=1}^{t}(1-\beta_2)\beta_2^{t-i} = 1-\beta_2^t \leq 1$. From Lemma D.4.3,

$$E\left[v_{t,j}\right] = E\left[\sum_{i=1}^{t}(1-\beta_2)\beta_2^{t-i}g_{i,j}^2\right] \leq \sum_{i=1}^{t}(1-\beta_2)\beta_2^{t-i}\left(\sigma^2+G^2\right) \leq \sigma^2+G^2.$$

□

And we can derive the following important lemma:

Lemma D.4.6. ***[Bounded A-LR]*** *For any* $t \geq 1$, $j \in [1,d]$, $\beta_2 \in [0,1]$, *and fixed* ϵ *in* Adam *and* β *defined in softplus function in* Sadam, *the following bounds always hold:*

Adam *has* $(\mu_1,\mu_2)-$ *bounded A-LR:*

$$\mu_1 \leq \frac{1}{\sqrt{v_{t,j}}+\epsilon} \leq \mu_2; \tag{7}$$

Sadam *has* $(\mu_3,\mu_4)-$ *bounded A-LR:*

$$\mu_3 \leq \frac{1}{softplus\left(\sqrt{v_{t,j}}\right)} \leq \mu_4; \tag{8}$$

where $0 < \mu_1 \leq \mu_2$, $0 < \mu_3 \leq \mu_4$. *For brevity, we use* μ_l,μ_u *denoting the lower bound and upper bound respectively, and both* Adam *and* Sadam *will be analysis with the help of* (μ_l,μ_u).

Proof. For ADAM, let $\mu_1 = \frac{1}{\sqrt{\sigma^2 + G^2} + \epsilon}$, $\mu_2 = \frac{1}{\epsilon}$, then we can get the result in (7).

For SADAM, notice that *softplus* $(\cdot)$ is a monotone increasing function, and $\sqrt{v_{t,j}}$ is both upper-bounded and lower-bounded, then we have (8), where $\mu_3 = \frac{1}{\frac{1}{\beta}\log\left(1 + e^{\beta \cdot \sqrt{\sigma^2 + G^2}}\right)}$, $\mu_4 = \frac{1}{\frac{1}{\beta}\log\left(1 + e^{\beta \cdot 0}\right)} = \frac{\beta}{\log 2}$.□

Lemma D.4.7. *Define* $z_t = x_t + \frac{\beta}{1-\beta_1}(x_t - x_{t-1})$, $\forall t \geq 1\ \beta_1 \in [0, 1)$. *Let* $\eta_t = \eta$, *then the following updating formulas hold: Gradient-based optimizer*

$$z_t = x_t, \quad z_{t+1} = z_t - \eta g_t; \tag{9}$$

ADAM *optimizer*

$$z_{t+1} = z_t + \frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}} + \epsilon} - \frac{1}{\sqrt{v_t} + \epsilon}\right) \odot m_{t-1} - \frac{\eta}{\sqrt{v_t} + \epsilon} \odot g_t; \tag{10}$$

SADAM *optimizer*

$$\begin{aligned} z_{t+1} = z_t &+ \frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{softplus\left(\sqrt{v_{t-1}}\right)} - \frac{1}{softplus\left(\sqrt{v_t}\right)}\right) \odot m_{t-1} \\ &- \frac{\eta}{softplus\left(\sqrt{v_t}\right)} \odot g_t\,. \end{aligned} \tag{11}$$

Proof. We consider the ADAM optimizer and let $\beta_1 = 0$, we can easily derive the gradient-based case.

$$z_{t+1} = x_{t+1} + \frac{\beta_1}{1-\beta_1}(x_{t+1} - x_t)$$

$$\begin{aligned} z_{t+1} &= z_t + \frac{1}{1-\beta_1}(x_{t+1} - x_t) - \frac{\beta_1}{1-\beta_1}(x_t - x_{t-1}) \\ &= z_t - \frac{1}{1-\beta_1}\frac{\eta}{\sqrt{v_t} + \epsilon} \odot m_t + \frac{\beta_1}{1-\beta_1}\frac{\eta}{\sqrt{v_{t-1}} + \epsilon} \odot m_{t-1} \\ &= z_t + \frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}} + \epsilon} - \frac{1}{\sqrt{v_t} + \epsilon}\right) \odot m_{t-1} - \frac{\eta}{\sqrt{v_t} + \epsilon} \odot g_t\,. \end{aligned}$$

Similarly, consider the SADAM optimizer:

$$
\begin{aligned}
z_{t+1} &= z_t + \frac{1}{1-\beta_1}(x_{t+1} - x_t) - \frac{\beta_1}{1-\beta_1}(x_t - x_{t-1}) \\
&= z_t - \frac{1}{1-\beta_1}\frac{\eta}{softplus\left(\sqrt{v_t}\right)} \odot m_t + \frac{\beta_1}{1-\beta_1}\frac{\eta}{softplus\left(\sqrt{v_{t-1}}\right)} \odot m_{t-1} \\
&= z_t + \frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{softplus\left(\sqrt{v_{t-1}}\right)} - \frac{1}{softplus\left(\sqrt{v_t}\right)}\right) \odot m_{t-1} - \frac{\eta}{softplus\left(\sqrt{v_t}\right)} \odot g_t .
\end{aligned}
$$

□

Lemma D.4.8. *As defined in Lemma D.4.7, with the condition that* $v_t \geq v_{t-1}$, *i.e.,* AMSGRAD *and* SAMSGRAD, *we can derive the bound of distance of* $\|z_{t+1} - z_t\|^2$ *as follows:*

ADAM *optimizer*

$$
\begin{aligned}
E\left[\|z_{t+1} - z_t\|^2\right] \leq \frac{2\eta^2\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}} + \epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}} + \epsilon}\right)^2\right] \\
+ 2\eta^2\mu_2^2\left(\sigma^2 + G^2\right)
\end{aligned} \tag{12}
$$

SADAM *optimizer*

$$
\begin{aligned}
E\left[\|z_{t+1} - z_t\|^2\right] \leq \frac{2\eta^2\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_{t-1,j}}\right)}\right)^2\right. \\
\left. - \left(\frac{1}{softplus\left(\sqrt{v_{t,j}}\right)}\right)^2\right] \\
+ 2\eta^2\mu_4^2\left(\sigma^2 + G^2\right)
\end{aligned} \tag{13}
$$

Proof. Adam case:

$$
\begin{aligned}
E\left[\|z_{t+1} - z_t\|^2\right] &= E\left[\left\|\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}} + \epsilon} - \frac{1}{\sqrt{v_t} + \epsilon}\right) \odot m_{t-1} - \frac{\eta}{\sqrt{v_t} + \epsilon} \odot g_t\right\|^2\right] \\
&\leq 2E\left[\left\|\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}} + \epsilon} - \frac{1}{\sqrt{v_t} + \epsilon}\right) \odot m_{t-1}\right\|^2\right] + + 2E\left[\left\|\frac{\eta}{\sqrt{v_t} + \epsilon} \odot g_t\right\|\right]^2 \\
&\leq \frac{2\eta^2\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}} + \epsilon} - \frac{1}{\sqrt{v_{t,j}} + \epsilon}\right)^2\right] + 2\eta^2\mu_2^2\left(\sigma^2 + G^2\right) \\
&\leq \frac{2\eta^2\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}} + \epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}} + \epsilon}\right)^2\right] + 2\eta^2\mu_2^2\left(\sigma^2 + G^2\right)
\end{aligned}
$$

The first inequality holds because $\|a-b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$, the second inequality holds because Lemma D.4.3 and D.4.4 and Lemma D.4.6, the third inequality holds because $(a - b)^2 \leq a^2 - b^2$ when $a \geq b$, and in our assumption, we have $v_t \geq v_{t-1}$ holds.

SADAM case:

$$
\begin{aligned}
E\left[\left\|z_{t+1}-z_t\right\|^2\right] &= E\left[\left\|\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{softplus\left(\sqrt{v_t-1}\right)}-\frac{1}{softplus\left(\sqrt{v_t}\right)}\right)\odot m_t-1-\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|^2\right] \\
&\le 2E\left[\left\|\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{softplus\left(\sqrt{v_t-1}\right)}-\frac{1}{softplus\left(\sqrt{v_t}\right)}\right)\odot m_t-1\right\|\right]^2 \\
&+ 2E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|\right]^2 \\
&\le \frac{2\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_{t-1,j}}\right)}-\frac{1}{softplus\left(\sqrt{v_{t,j}}\right)}\right)^2\right] \\
&+ 2\eta^2\mu_4^2\left(\sigma^2+G^2\right) \\
&\le \frac{2\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_{t-1,j}}\right)}\right)^2-\left(\frac{1}{softplus\left(\sqrt{v_{t,j}}\right)}\right)^2\right] \\
&+ 2\eta^2\mu_4^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

Because the *softplus* function is monotone increasing function, therefore, the third inequality holds as well.□

Lemma D.4.9. *As defined in Lemma D.4.7, with the condition that* $v_t \ge v_{t-1}$, *we can derive the bound of the inner product as follows:*

ADAM *optimizer*

$$
-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t),\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right] \le \frac{1}{2}L^2\eta^2\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right); \tag{14}
$$

SADAM *optimizer*

$$
\begin{aligned}
-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t),\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right] &\le \frac{1}{2}L^2\eta^2\mu_4^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) \\
&+\frac{1}{2}\eta^2\mu_4^2\left(\sigma^2+G^2\right).
\end{aligned} \tag{15}
$$

Proof. Since the stochastic gradient is unbiased, then we have $E[g_t] = \nabla f(x_t)$.

ADAM case:

$$
\begin{aligned}
&-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right] \\
&\quad\le \frac{1}{2}E\left[\|\nabla f(z_t)-\nabla f(x_t)\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&\quad\le \frac{L^2}{2}E\left[\|z_t-x_t\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&\quad= \frac{L^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\|x_t-x_t-1\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&\quad= \frac{L^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\left\|\frac{\eta}{\sqrt{v_t-1}+\epsilon}\odot m_t-1\right\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&\quad\le \frac{1}{2}L^2\eta^2\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

The first inequality holds because $\frac{1}{2}a^2+\frac{1}{2}b^2 \ge -<a,b>$, the second inequality holds for L-smoothness, the last inequalities hold due to Lemma D.4.4 and D.4.6.

Similarly, for SADAM, we also have the following result:

$$
\begin{aligned}
&-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right] \\
&\quad\le \frac{1}{2}E\left[\|\nabla f(z_t)-\nabla f(x_t)\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|^2\right] \\
&\quad\le \frac{L^2}{2}E\left[\|z_t-x_t\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|^2\right] \\
&\quad= \frac{L^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\|x_t-x_t-1\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|^2\right] \\
&\quad= \frac{L^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot m_t-1\right\|^2\right]+\frac{1}{2}E\left[\left\|\frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\|^2\right] \\
&\quad\le \frac{1}{2}L^2\eta^2\mu_4^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_4^2\left(\sigma^2+G^2\right).
\end{aligned}
$$

□

D.4.2. ADAM *Convergence in Nonconvex Setting*

Proof. All the analyses hold true under the condition: $v_t \ge v_{t-1}$. From L-smoothness and Lemma D.4.7, we have

$$
\begin{aligned}
f(z_t+1) &\le f(z_t)+\langle \nabla f(z_t), z_t+1-z_t\rangle+\frac{L}{2}\|z_t+1-z_t\|^2 \\
&= f(z_t)+\frac{\eta\beta_1}{1-\beta_1}\left\langle \nabla f(z_t), \left(\frac{1}{\sqrt{v_t-1}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_t-1\right\rangle \\
&\quad-\left\langle \nabla f(z_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle+\frac{L}{2}\|z_t+1-z_t\|^2
\end{aligned}
$$

Take expectation on both sides,

$$\begin{aligned}E[f(z_t+1)-f(z_t)] \le{}& \frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle \nabla f(z_t), \left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}\right\rangle\right] \\ &-E\left[\left\langle \nabla f(z_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right]+\frac{L}{2}E\left[\|z_{t+1}-z_t\|^2\right] \\ ={}& \frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle \nabla f(z_t), \left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}\right\rangle\right] \\ &-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right]-E\left[\left\langle \nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right] \\ &+\frac{L}{2}E\left[\|z_{t+1}-z_t\|^2\right]\end{aligned}$$

Plug in the results from prepared lemmas, then we have,

$$\begin{aligned}E[f(z_t+1)\le f(z_t)] \le{}& \frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle \nabla f(z_t), \left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}\right\rangle\right] \\ &+\frac{1}{2}L^2\eta^2\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right)-E\left[\left\langle \nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right] \\ &+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\left(\sigma^2+G^2\right)\end{aligned}$$

Applying the bound of m_t and $\nabla f(z_t)$,

$$\begin{aligned}E[f(z_t+1)\le f(z_t)] \le{}& \frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}-\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right] \\ &+\frac{1}{2}L^2\eta^2\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right)-E\left[\left\langle \nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle\right] \\ &+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\left(\sigma^2+G^2\right)\end{aligned}$$

By rearranging,

$$\begin{aligned}E\left[\nabla(x_t), \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t\right] \le{}& E[f(z_t)-f(z_t+1)]+\frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}-\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right] \\ &+\frac{1}{2}L^2\eta^2\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right) \\ &+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\left(\sigma^2+G^2\right)\end{aligned}$$

For the LHS above:

$$E\left[\left\langle \nabla f(x_t), \frac{1}{\sqrt{v_t}+\epsilon} \odot g_t \right\rangle\right] \geq E\left[\sum_{\{j \mid \nabla f(x_t)_j g_{t,j} \geq 0\}} \mu_1 \nabla f(x_t)_j g_{t,j} + \sum_{\{j \mid \nabla f(x_t)_j g_{t,j} < 0\}} \mu_2 \nabla f(x_t)_j g_{t,j}\right]$$
$$\geq \sum_{\{j \mid \nabla f(x_t)_j g_{t,j} \geq 0\}} \mu_1 \nabla f(x_t)_j^2 + \sum_{\{j \mid \nabla f(x_t)_j g_{t,j} < 0\}} \mu_2 \nabla f(x_t)_j^2$$
$$\geq \mu_1 \| \nabla f(x_t) \|^2$$

Then we obtain:

$$\eta\mu_1 \| \nabla f(x_t) \|^2 \leq E[f(z_t) - f(z_{t+1})] + \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_{t-1,j}}+\epsilon} - \frac{1}{\sqrt{v_{t,j}}+\epsilon}\right]$$
$$+ \frac{1}{2} L^2 \eta^2 \mu_2^2 \left(\frac{\beta_1}{1-\beta_1}\right)^2 \left(\sigma^2+G^2\right) + \frac{1}{2}\eta^2\mu_2^2\left(\sigma^2+G^2\right)$$
$$+ \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right)$$

Divide $\eta\mu_1$ on both sides:

$$\| \nabla f(x_t) \|^2 \leq \frac{1}{\eta\mu_1} E[f(z_t) - f(z_{t+1})] + \frac{\beta_1}{(1-\beta_1)\mu_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_{t-1,j}}+\epsilon} - \frac{1}{\sqrt{v_{t,j}}+\epsilon}\right]$$
$$+ \frac{1}{2\mu_1} L^2 \eta\mu_2^2 \left(\frac{\beta_1}{1-\beta_1}\right)^2 \left(\sigma^2+G^2\right) + \frac{1}{2\mu_1}\eta\mu_2^2\left(\sigma^2+G^2\right)$$
$$+ \frac{L\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + \frac{L\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)$$

Summing from $t = 1$ to T, where T is the maximum number of iteration,

$$\sum_{t=1}^{T}\left[\| \nabla f(x_t) \|^2\right] \leq \frac{1}{\eta\mu_1} E\left[f(z_1) - f^*\right] + \frac{\beta_1}{(1-\beta_1)\mu_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_{0,j}}+\epsilon} - \frac{1}{\sqrt{v_{T,j}}+\epsilon}\right]$$
$$+ \frac{T}{2\mu_1} L^2 \eta\mu_2^2 \left(\frac{\beta_1}{1-\beta_1}\right)^2 \left(\sigma^2+G^2\right) + \frac{T}{2\mu_1}\eta\mu_2^2\left(\sigma^2+G^2\right)$$
$$+ \frac{L\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{0,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{T,j}}+\epsilon}\right)^2\right] + \frac{TL\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)$$

Since $v_0 = 0$, $\mu_2 = \frac{1}{\epsilon}$, we have

$$\sum_{t=1}^{T}\left[\| \nabla f(x_t) \|^2\right] \leq \frac{1}{\eta\mu_1} E\left[f(z_1) - f^*\right] + \frac{\beta_1 d}{(1-\beta_1)\mu_1} G\sqrt{\sigma^2+G^2}(\mu_2 - \mu_1)$$
$$+ \frac{T}{2\mu_1} L^2 \eta\mu_2^2 \left(\frac{\beta_1}{1-\beta_1}\right)^2 \left(\sigma^2+G^2\right) + \frac{T}{2\mu_1}\eta\mu_2^2\left(\sigma^2+G^2\right)$$
$$+ \frac{L\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1}\left(\mu_2^2 - \mu_1^2\right) + \frac{TL\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)$$

Divided by $\frac{1}{T}$,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\left[\|\nabla f(x_t)\|^2\right] &\le \frac{1}{\eta\mu_1 T}E\left[f(z_1)-f^*\right]+\frac{\beta_1 d}{(1-\beta_1)\mu_1 T}G\sqrt{\sigma^2+G^2}(\mu_2-\mu_1)\\ &+\frac{T}{2\mu_1}L^2\eta\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{1}{2\mu_1}\eta\mu_2^2\left(\sigma^2+G^2\right)\\ &+\frac{L\eta\beta_1^2 d\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1 T}\left(\mu_2^2-\mu_1^2\right)+\frac{L\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)\\ &\le \frac{1}{\eta\mu_1 T}E\left[f(z_1)-f^*\right]+\left(\frac{\beta_1 d}{(1-\beta_1)\mu_1 T}(\mu_2-\mu_1)\right.\\ &\left.+\frac{1}{2\mu_1}L^2\eta\mu_2^2\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\eta\mu_2^2}{2\mu_1}+\frac{L\eta\beta_1^2 d\left(\mu_2^2-\mu_1^2\right)}{(1-\beta_1)^2\mu_1 T}+\frac{L\eta\mu_2^2}{\mu_1}\right)\left(\sigma^2+G^2\right)\end{aligned}$$

The second inequality holds because $G\sqrt{\sigma^2+G^2}\le\sigma^2+G^2$.

Setting $\eta=\frac{1}{\sqrt{T}}$, let $x_0=x_1$, then $z_1=x_1$, $f(z_1)=f(x_1)$ we derive the final result:

$$\begin{aligned}\min_{t=1,\ldots,T}E\left[\|\nabla f(x_t)\|^2\right] &\le \frac{1}{\mu_1\sqrt{T}}E\left[f(x_1)-f^*\right]+\left(\frac{\beta_1 d}{(1-\beta_1)\mu_1 T}(\mu_2-\mu_1)\right.\\ &\left.+\frac{L^2\mu_2^2}{2\mu_1\sqrt{T}}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\mu_2^2}{2\mu_1\sqrt{T}}+\frac{L\beta_1^2 d\left(\mu_2^2-\mu_1^2\right)}{(1-\beta_1)^2\mu_1 T\sqrt{T}}+\frac{L\mu_2^2}{\mu_1\sqrt{T}}\right)\left(\sigma^2+G^2\right)\\ &=\frac{C_1}{\sqrt{T}}+\frac{C_2}{T}+\frac{C_3}{T\sqrt{T}}\end{aligned}$$

where

$$C_1=\frac{1}{\mu_1}\left[f(x_1)-f^*\right]+\left(\frac{L^2\mu_2^2}{2\mu_1}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\mu_2^2}{2\mu_1}+\frac{L\mu_2^2}{\mu_1}\right)\left(\sigma^2+G^2\right)$$

$$C_2=\frac{\beta_1(\mu_2-\mu_1)d}{(1-\beta_1)\mu_1},$$

$$C_3=\frac{L\beta_1^2 d\left(\mu_2^2-\mu_1^2\right)}{(1-\beta_1)\mu_1}.$$

With fixed L,σ,G,β_1, we have $C_1=O\left(\frac{1}{\epsilon^2}\right)$, $C_2=O\left(\frac{d}{\epsilon}\right)$, $C_3=O\left(\frac{d}{\epsilon^2}\right)$.

Therefore,

$$\min_{t=1,\ldots,T} E\left[\|\nabla f(x_t)\|^2\right] \leq O\left(\frac{1}{\epsilon^2\sqrt{T}} + \frac{d}{\epsilon T} + \frac{d}{\epsilon^2 T\sqrt{T}}\right)$$

□

Thus, we get the sublinear convergence rate of ADAM in nonconvex setting, which recovers the well-known result of SGD ([1]) in nonconvex optimization in terms of T.

Remark D.4.10. *The leading item from the above convergence is* $C_1/\sqrt{T}$, ϵ *plays an essential role in the complexity, and we derive a more accurate order* $O\left(\frac{1}{\epsilon^2\sqrt{T}}\right)$. *At present,* ϵ *is always underestimated and considered to be not associated with accuracy of the solution* ([14]). *However, it is closely related with complexity, and with bigger* ϵ, *the computational complexity should be better. This also supports the analysis of* A-LR: $\frac{1}{\sqrt{v_t+\epsilon}}$ *of* ADAM *in our main paper.*

In some other works, people use σ_i or G_i to show all the element-wise bound, and then by applying $\sum_{j=1}^{d}\sigma_i := \sigma$, $\sum_{j=1}^{d} G_i := G$ to hide d in the complexity. Here in our work, we didn't specify write out σ_i or G_i, instead we use σ, G through all the procedure.

D.4.3. SADAM *Convergence in Nonconvex Setting*

As SADAM also has constrained bound pair (μ_3,μ_4), we can learn from the proof of ADAM method, which provides us a general framework of such kind of adaptive methods.

Similar to the ADAM proof, from L-smoothness and Lemma D.4.7, we have

Proof. All the analyses hold true under the condition: $v_t \geq v_{t-1}$. From L-smoothness and Lemma D.4.7, we have

$$\begin{aligned} f(z_{t+1}) &\leq f(z_t) + \langle \nabla f(z_t), z_{t+1} - z_t\rangle + \frac{L}{2}\|z_{t+1} - z_t\|^2 \\ &= f(z_t) + \frac{\eta\beta_1}{1-\beta_1}\left\langle \nabla f(z_t), \left(\frac{1}{softplus\left(\sqrt{v_{t-1}}\right)} - \frac{1}{softplus\left(\sqrt{v_t}\right)}\right) \odot m_{t-1}\right\rangle \\ &\quad - \left\langle \nabla f(z_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)} \odot g_t\right\rangle + \frac{L}{2}\|z_{t+1} - z_t\|^2 \end{aligned}$$

Taking expectation on both sides, and plug in the results from prepared lemmas, then we have,

$$
\begin{aligned}
&E[f(z_t+1)-f(z_t)]\\
&\le \frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle \nabla f(z_t), \left(\frac{1}{softplus\left(\sqrt{v_t-1}\right)}-\frac{1}{softplus\left(\sqrt{v_t}\right)}\right)\odot m_t-1\right\rangle\right]\\
&-E\left[\left\langle \nabla f(z_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]+\frac{L}{2}E\left[\|z_t+1-z_t\|^2\right]\\
&\le \frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle \nabla f(z_t), \left(\frac{1}{softplus\left(\sqrt{v_t-1}\right)}-\frac{1}{softplus\left(\sqrt{v_t}\right)}\right)\odot m_t-1\right\rangle\right]\\
&-E\left[\left\langle \nabla f(z_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]\\
&+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}\right)^2-\left(\frac{1}{softplus\sqrt{v_t,j}}\right)^2\right]+L\eta^2\mu_4^2\left(\sigma^2+G^2\right)\\
&=\frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}-\frac{1}{softplus\sqrt{v_t,j}}\right]\\
&-E\left[\left\langle \nabla f(z_t)-\nabla f(x_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]-E\left[\left\langle \nabla f(x_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]\\
&+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}\right)^2-\left(\frac{1}{softplus\sqrt{v_t,j}}\right)^2\right]+L\eta^2\mu_4^2\left(\sigma^2+G^2\right)\\
&\le \frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}-\frac{1}{softplus\sqrt{v_t,j}}\right]\\
&+\frac{L^2\eta^2\mu_4^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{\eta^2\mu_4^2}{2}\left(\sigma^2+G^2\right)-E\left[\left\langle \nabla f(x_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]\\
&+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}\right)^2-\left(\frac{1}{softplus\sqrt{v_t,j}}\right)^2\right]+L\eta^2\mu_4^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

By rearranging,

$$
\begin{aligned}
&E\left[\left\langle \nabla f(x_t), \frac{\eta}{softplus\left(\sqrt{v_t}\right)}\odot g_t\right\rangle\right]\\
&\le E[f(z_t)-f(z_t+1)]+\frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}-\frac{1}{softplus\left(\sqrt{v_t,j}\right)}\right]\\
&+\frac{L^2\eta^2\mu_4^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{\eta^2\mu_4^2}{2}\left(\sigma^2+G^2\right)\\
&+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_t-1,j}\right)}\right)^2-\left(\frac{1}{softplus\left(\sqrt{v_t,j}\right)}\right)^2\right]+L\eta^2\mu_4^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

For the LHS above:

$$E\left[\left\langle \nabla f(x_t), \frac{1}{softplus\left(\sqrt{v_t}\right)} \odot g_t \right\rangle\right] \geq E\left[\sum_{\left\{j \middle| \nabla f\left(x_{t,j}\right) g_{t,j} \geq 0\right\}} \mu_3 \nabla f\left(x_{t,j}\right) g_{t,j} + \sum_{\left\{j \middle| \nabla f\left(x_{t,j}\right) g_{t,j} < 0\right\}} \mu_4 \nabla f\left(x_{t,j}\right) g_{t,j}\right]$$
$$\geq \sum_{\left\{j \middle| \nabla f\left(x_{t,j}\right) g_{t,j} \geq 0\right\}} \mu_3 \nabla f\left(x_{t,j}\right)^2 + \sum_{\left\{j \middle| \nabla f\left(x_{t,j}\right) g_{t,j} < 0\right\}} \mu_4 \nabla f\left(x_{t,j}\right)^2$$
$$\geq \mu_3 \| \nabla f(x_t) \|^2$$

Then we obtain:

$$\eta\mu_3 \| \nabla f(x_t) \|^2 \leq E[f(z_t) - f(z_t + 1)] + \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2 + G^2} E\left[\sum_{j=1}^{d} \frac{1}{softplus\left(\sqrt{v_t - 1, j}\right)} - \frac{1}{softplus\sqrt{v_{t,j}}}\right]$$
$$+ \frac{L^2\eta^2\mu_4^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2 + G^2\right) + \frac{\eta^2\mu_4^2}{2}\left(\sigma^2 + G^2\right)$$
$$+ \frac{L\eta^2\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_t - 1, j}\right)}\right)^2 - \left(\frac{1}{softplus\sqrt{v_{t,j}}}\right)^2\right] + L\eta^2\mu_4^2\left(\sigma^2 + G^2\right)$$

Divide $\eta\mu_3$ on both sides and then sum from $t = 1$ to T, where T is the maximum number of iteration,

$$\sum_{t=1}^{T}\left[\| \nabla f(x_t) \|^2\right] \leq \frac{1}{\eta\mu_3} E\left[f(z_1) - f^*\right] + \frac{\beta_1}{(1-\beta_1)\mu_3} G\sqrt{\sigma^2 + G^2} E\left[\sum_{j=1}^{d} \frac{1}{softplus\left(\sqrt{v_{0,j}}\right)} - \frac{1}{softplus\left(\sqrt{v_{T,j}}\right)}\right]$$
$$+ \frac{L^2\eta T\mu_4^2}{2\mu_{31}}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2 + G^2\right) + \frac{\eta\mu_4^2 T}{2\mu_3}\left(\sigma^2 + G^2\right)$$
$$+ \frac{L\eta\beta_1^2\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2\mu_3} E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(\sqrt{v_{0,j}}\right)}\right)^2 - \left(\frac{1}{softplus\left(\sqrt{v_{T,j}}\right)}\right)^2\right] + \frac{L\eta\mu_4^2 T\left(\sigma^2 + G^2\right)}{\mu_3}$$

Since, v_0 =0, $\frac{1}{softplus(0)} = \mu_4$, we have

$$\sum_{t=1}^{T}\left[\| \nabla f(x_t) \|^2\right] \leq \frac{1}{\eta\mu_3} E\left[f(z_1) - f^*\right] + \frac{\beta_1 d}{(1-\beta_1)\mu_3} G\sqrt{\sigma^2 + G^2}(\mu_4 - \mu_3)$$
$$+ \frac{L^2\eta T\mu_4^2}{2\mu_3}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2 + G^2\right) + \frac{\eta\mu_4^2 T}{2\mu_3}\left(\sigma^2 + G^2\right)$$
$$+ \frac{L\eta\beta_1^2 d\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2\mu_3}\left(\mu_4^2 - \mu_3^2\right) + \frac{L\eta\mu_4^2 T\left(\sigma^2 + G^2\right)}{\mu_3}$$

Divided by $\frac{1}{T}$,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\left[\|\nabla f(x_t)\|^2\right] &\le \frac{1}{\eta\mu_3 T}E\left[f(z_1)-f^*\right]+\frac{\beta_1 d}{(1-\beta_1)\mu_3 T}G\sqrt{\sigma^2+G^2}(\mu_4-\mu_3)\\ &+\frac{L^2\eta\mu_4^2}{2\mu_3}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{\eta\mu_4^2}{2\mu_3}\left(\sigma^2+G^2\right)\\ &+\frac{L\eta\beta_1^2 d\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_3 T}\left(\mu_4^2-\mu_3^2\right)+\frac{L\eta\mu_4^2\left(\sigma^2+G^2\right)}{\mu_3}\\ &\le \frac{1}{\eta\mu_3 T}E\left[f(z_1)-f^*\right]+\left(\frac{\beta_1 d}{(1-\beta_1)\mu_3 T}(\mu_4-\mu_3)\right.\\ &\left.+\frac{L^2\eta\mu_4^2}{2\mu_3}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\eta\mu_4^2}{2\mu_3}+\frac{L\eta\beta_1^2 d}{(1-\beta_1)^2\mu_3 T}\left(\mu_4^2-\mu_3^2\right)+\frac{L\eta\mu_4^2}{\mu_3}\right)\left(\sigma^2+G^2\right)\end{aligned}$$

Setting $\eta=\frac{1}{\sqrt{T}}$, let $x_0=x_1$, then $z_1=x_1$, $f(z_1)=f(x_1)$ we derive the final result for SADAM method:

$$\begin{aligned}\min_{t=1,\ldots,T}E\left[\|\nabla f(x_t)\|^2\right] &\le \frac{1}{\mu_3\sqrt{T}}E\left[f(x_1)-f^*\right]+\left(\frac{\beta_1 d}{(1-\beta_1)\mu_3 T}(\mu_4-\mu_3)\right.\\ &\left.+\frac{L^2\mu_4^2}{2\mu_3\sqrt{T}}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\mu_4^2}{2\mu_3\sqrt{T}}+\frac{L\beta_1^2 d\left(\mu_4^2-\mu_3^2\right)}{(1-\beta_1)^2\mu_3 T\sqrt{T}}+\frac{L\mu_4^2}{\mu_3\sqrt{T}}\right)\left(\sigma^2+G^2\right)\\ &=\frac{C_1}{\sqrt{T}}+\frac{C_2}{T}+\frac{C_3}{T\sqrt{T}}\end{aligned}$$

where

$$C_1=\frac{1}{\mu_3}\left[f(x_1)-f^*\right]+\left(\frac{L^2\mu_4^2}{2\mu_3}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\mu_4^2}{2\mu_3}+\frac{L\mu_4^2}{\mu_3}\right)\left(\sigma^2+G^2\right)$$

$$C_2=\frac{\beta_1(\mu_4-\mu_3)d}{(1-\beta_1)\mu_3},$$

$$C_3=\frac{L\beta_1^2 d\left(\mu_4^2-\mu_3^2\right)}{(1-\beta_1)^2\mu_3}.$$

With fixed L,σ,G,β_1, we have $C_1=O(\beta^2)$, $C_2=O(d\beta)$, $C_3=O(d\beta^2)$.

Therefore,

$$\min_{t=1,\ldots,T}E\left[\|\nabla f(x_t)\|^2\right]\le O\left(\frac{\beta^2}{\sqrt{T}}+\frac{d\beta}{T}+\frac{d\beta^2}{T\sqrt{T}}\right)$$

□

Thus, we get the sublinear convergence rate of SADAM in nonconvex setting, which is the same order of ADAM and recovers the well-known result of SGD [1] in nonconvex optimization in terms of T.

Remark D.4.11. *The leading item from the above convergence is* $C_1/\sqrt{T}$, β *plays an essential role in the complexity, and a more accurate convergence should be* $O\left(\frac{\beta log\left(1+e^{\beta}\right)}{\sqrt{T}}\right)$. *When* β *is chosen big, this will become* $O\left(\frac{\beta^2}{\sqrt{T}}\right)$, *somehow behave like* ADAM *'s case as* $O\left(\frac{1}{\epsilon^2\sqrt{T}}\right)$, *which also guides us to have a range of* β*; when* β *is chosen small, this will become* $O\left(\frac{1}{\sqrt{T}}\right)$, *the computational complexity will get close to SGD case, and* β *is a much smaller number compared with* $1/\epsilon$, *proving that* SADAM *converges faster. This also supports the analysis of range of A-LR:* $1/softplus\left(\sqrt{v_t}\right)$ *in our main paper.*

D.4.4. Non-strongly Convex

In previous works, convex case has been well-studied in adaptive gradient methods. AMSGRAD and later methods PAMSGRAD both use a projection on minimizing objective function, here we want to show a different way of proof in non-strongly convex case. For consistency, we still follow the construction of sequence $\{z_t\}$.

Starting from convexity:

$$f(y) \geq f(x) + \nabla f(x)^T (y - x).$$

Then, for any $x \in \mathbb{R}^d$, $\forall t \in [1, T]$,

$$\langle \nabla f(x), x_t - x^* \rangle \geq f(x_t) - f^*, \tag{16}$$

where $f^* = f(x^*)$, x^* is the optimal solution.

Proof. ADAM case:

In the updating rule of ADAM optimizer, $x_{t+1} = x_t - \frac{\eta_t}{\sqrt{v_t + \epsilon}} \odot m_t$, setting stepsize to be fixed, $\eta_t = \eta$, and assume $v_t \geq v_{t-1}$ holds. Using previous results,

$$
\begin{aligned}
&E\left[\|z_{t+1}-x^*\|^2\right] \\
&= E\left[\left\|z_t+\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}-\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t-x^*\right\|^2\right] \\
&= E\left[\|z_t-x^*\|^2\right]+E\left[\left\|\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}-\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&+2E\left[\left\langle\frac{\eta\beta_1}{1-\beta_1}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1},z_t-x^*\right\rangle\right]-2E\left[\left\langle\frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t,z_t-x^*\right\rangle\right] \\
&\le E\left[\|z_t-x^*\|^2\right]+2\frac{\eta^2\beta_1^2}{(1-\beta_1)^2}E\left[\left\|\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}\right\|^2\right]+2\eta^2E\left[\left\|\frac{1}{\sqrt{v_t}+\epsilon}\odot g_t\right\|^2\right] \\
&+2\frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1},z_t-x^*\right\rangle\right]-2\eta E\left[\left\langle\frac{1}{\sqrt{v_t}+\epsilon}\odot g_t,z_t-x^*\right\rangle\right] \\
&\le E\left[\|z_t-x^*\|^2\right]+2\frac{\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_t}+\epsilon}\right)^2\right]+2\eta^2\mu_2^2\left(\sigma^2+G^2\right) \\
&+2\frac{\eta\beta_1}{1-\beta_1}E\left[\left\langle\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1},z_t-x^*\right\rangle\right]-2\eta E\left[\left\langle\frac{1}{\sqrt{v_t}+\epsilon}\odot g_t,z_t-x^*\right\rangle\right]
\end{aligned}
$$

The first inequality holds due to $\|a-b\|^2 \le 2\|a\|^2+2\|b\|^2$, the second inequality holds due to Lemma D.4.3, D.4.4, D.4.6.

Since, $< a,b > \ \le \frac{1}{2\eta}a^2+\frac{\eta}{2}b^2$,

$$
\begin{aligned}
&2E\left[\left\langle\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1},z_t-x^*\right\rangle\right] \\
&\le \frac{1}{\eta}E\left[\left\|\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}-\frac{1}{\sqrt{v_t}+\epsilon}\right)\odot m_{t-1}\right\|^2\right]+\eta E\left[\|z_t-x^*\|^2\right] \\
&\le \frac{1}{\eta}\left(\sigma^2+G^2\right)E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right]+\eta E\left[\|z_t-x^*\|^2\right]
\end{aligned}
$$

From the definition of z_t and convexity,

$$\langle \nabla f(x_t), x_t-x^*\rangle \ge f(x_t)-f^* \ge 0$$

$$
\begin{aligned}
&-2\eta E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, z_t - x^*\right\rangle\right] \\
&= -2\eta E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, x_t - x^* + \frac{\beta_1}{1-\beta_1}(x_t - x_{t-1})\right\rangle\right] \\
&= -2\eta E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, x_t - x^*\right\rangle\right] - \frac{2\eta\beta_1}{1-\beta_1}E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, x_t - x_{t-1}\right\rangle\right] \\
&= -2\eta E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, x_t - x^*\right\rangle\right] - \frac{2\eta^2\beta_1}{1-\beta_1}E\left[\left\langle \frac{1}{\sqrt{v_t}+\epsilon}\odot g_t, \frac{1}{\sqrt{v_{t-1}}+\epsilon}\odot m_{t-1}\right\rangle\right] \\
&\le -2\eta\mu_1\langle \nabla f(x_t), x_t - x^*\rangle + \frac{2\eta^2\beta_1\mu_2^2}{(1-\beta_1)}\left(\sigma^2+G^2\right) \\
&\le -2\eta\mu_1\left(f(x_t) - f^*\right) + \frac{2\eta^2\beta_1\mu_2^2}{(1-\beta_1)}\left(\sigma^2+G^2\right)
\end{aligned}
$$

Plugging in previous two inequalities:

$$
\begin{aligned}
&E\left[\|z_{t+1} - x^*\|^2\right] \\
&\le E\left[\|z_t - x^*\|^2\right] + 2\frac{\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t}+\epsilon}\right)^2\right] 2\eta^2\mu_2^2\left(\sigma^2+G^2\right) \\
&+\frac{\beta_1\left(\sigma^2+G^2\right)}{1-\beta_1}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + \frac{\eta^2\beta_1}{1-\beta_1}E\left[\|z_t - x^*\|^2\right] \\
&-2\eta\mu_1\left(f(x_t) - f^*\right) + \frac{2\eta^2\beta_1\mu_2^2}{(1-\beta_1)}\left(\sigma^2+G^2\right)
\end{aligned}
$$

By rearranging:

$$
\begin{aligned}
&2\eta\mu_1\left(f(x_t) - f^*\right) \\
&\le E\left[\|z_t - x^*\|^2\right] - E\left[\|z_{t+1} - x^*\|^2\right] + 2\frac{\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t}+\epsilon}\right)^2\right] \\
&+2\eta^2\mu_2^2\left(\sigma^2+G^2\right) + \frac{\beta_1\left(\sigma^2+G^2\right)}{1-\beta_1}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + \frac{\eta^2\beta_1}{1-\beta_1}E\left[\|z_t - x^*\|^2\right] \\
&+\frac{2\eta^2\beta_1\mu_2^2}{(1-\beta_1)}\left(\sigma^2+G^2\right)
\end{aligned}
$$

Divide $2\eta\mu_1$ on both sides,

$$f(x_t)-f^* \le \frac{1}{2\eta\mu_1}\left(E\left[\|z_t-x^*\|^2\right]-E\left[\|z_{t+1}-x^*\|^2\right]\right)+\frac{\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1}+\epsilon}}\right)^2-\left(\frac{1}{\sqrt{v_t+\epsilon}}\right)^2\right]$$
$$+\frac{\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)+\frac{\beta_1\left(\sigma^2+G^2\right)}{2\eta\mu_1(1-\beta_1)}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}+\epsilon}}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}+\epsilon}}\right)^2\right]$$
$$+\frac{\eta\beta_1}{2\mu_1(1-\beta_1)}E\left[\|z_t-x^*\|^2\right]+\frac{\eta\beta_1\mu_2^2}{(1-\beta_1)\mu_1}\left(\sigma^2+G^2\right)$$

Assume that $\forall t,\ E[\|x_t-x^*\|\le D$, for any $m\neq n$, $E\left[\|x_m-x_n\|\right]\le D_\infty$ hold, then $E[\|z_t-x^*\|^2]$ can be bounded.

$$E\left[\|z_1-x^*\|^2\right]=E\left[\|x_1-x^*\|^2\right]\le D^2 \tag{17}$$

$$\begin{aligned}E\left[\|z_t-x^*\|^2\right]&=E\left[\left\|x_t-x^*+\frac{\beta_1}{1-\beta_1}(x_t-x_{t-1})\right\|^2\right]\\&\le 2E\left[\|x_t-x^*\|^2\right]+\frac{2\beta_1^2}{(1-\beta_1)^2}E\left[\|(x_t-x_{t-1})\|^2\right]\\&\le 2D^2+\frac{2\beta_1^2}{(1-\beta_1)^2}D_\infty^2.\end{aligned} \tag{18}$$

Thus:

$$f(x_t)-f^* \le \frac{1}{2\eta\mu_1}\left(E\left[\|z_t-x^*\|^2\right]-E\left[\|z_{t+1}-x^*\|^2\right]\right)+\frac{\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1}+\epsilon}}\right)^2-\left(\frac{1}{\sqrt{v_t+\epsilon}}\right)^2\right]$$
$$+\frac{\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)+\frac{\beta_1\left(\sigma^2+G^2\right)}{2\eta\mu_1(1-\beta_1)}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}+\epsilon}}\right)^2-\left(\frac{1}{\sqrt{v_{t,j}+\epsilon}}\right)^2\right]$$
$$+\frac{\eta\beta_1 D^2}{\mu_1(1-\beta_1)}+\frac{\eta\beta_1^3 D_\infty^2}{\mu_1(1-\beta_1)^3}+\frac{\eta\beta_1\mu_2^2}{(1-\beta_1)\mu_1}\left(\sigma^2+G^2\right)$$

Summing from $t=1$ to T,

$$\begin{aligned}\sum_{t=1}^{T}\left(f(x_t)-f^*\right) &\le \frac{1}{2\eta\mu_1}\left(E\left[\|z_1-x^*\|^2\right]-E\left[\|z_T-x^*\|^2\right]\right)+\frac{\eta\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_0}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_T}+\epsilon}\right)^2\right]\\ &+\frac{\eta\mu_2^2T}{\mu_1}\left(\sigma^2+G^2\right)+\frac{\beta_1\left(\sigma^2+G^2\right)}{2\eta\mu_1(1-\beta_1)}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{0,j}}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_{T,j}}+\epsilon}\right)^2\right]\\ &+\frac{\eta\beta_1D^2T}{\mu_1(1-\beta_1)}+\frac{\eta\beta_1^3D_\infty^2T}{\mu_1(1-\beta_1)^3}+\frac{\eta\beta_1\mu_2^2T}{(1-\beta_1)\mu_1}\left(\sigma^2+G^2\right)\\ &\le \frac{1}{2\eta\mu_1}D^2+\frac{\eta\beta_1^2d\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1}\left(\mu_2^2-\mu_1^2\right)+\frac{\eta\mu_2^2T}{\mu_1}\left(\sigma^2+G^2\right)+\frac{\beta_1d\left(\sigma^2+G^2\right)}{2\eta\mu_1(1-\beta_1)}\left(\mu_2^2-\mu_1^2\right)\\ &+\frac{\eta\beta_1D^2T}{\mu_1(1-\beta_1)}+\frac{\eta\beta_1^3D_\infty^2T}{\mu_1(1-\beta_1)^3}+\frac{\eta\beta_1\mu_2^2T}{(1-\beta_1)\mu_1}\left(\sigma^2+G^2\right)\end{aligned}$$

The second inequality is based on the fact that, when iteration t reaches the maximum number T, x_t is the optimal solution, $z_T = x^*$.

By Jensen's inequality,

$$\frac{1}{T}\sum_{t=1}^{T}\left(f(x_t)-f^*\right)\ge f(\bar{x}_t)-f^*,$$

where $\bar{x}_t=\frac{1}{T}\sum_{t=1}^{T}x_t$.

Then,

$$\begin{aligned}f(\bar{x}_t)-f^* &\le \frac{D^2}{2\eta\mu_1T}+\frac{\eta\beta_1^2d\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1T}\left(\mu_2^2-\mu_1^2\right)+\frac{\eta\mu_2^2}{\mu_1}\left(\sigma^2+G^2\right)+\frac{\beta_1d\left(\sigma^2+G^2\right)}{2\eta\mu_1(1-\beta_1)T}\left(\mu_2^2-\mu_1^2\right)\\ &+\frac{\eta\beta_1D^2}{\mu_1(1-\beta_1)}+\frac{\eta\beta_1^3D_\infty^2}{\mu_1(1-\beta_1)^3}+\frac{\eta\beta_1\mu_2^2}{(1-\beta_1)\mu_1}\left(\sigma^2+G^2\right)\end{aligned}$$

By plugging the stepsize $\eta=O\left(\frac{1}{\sqrt{T}}\right)$, we complete the proof of Adam in non-strongly convex case.

$$\begin{aligned}f(\bar{x}_t)-f^* &\le \frac{D^2}{2\mu_1\sqrt{T}}+\frac{\beta_1^2d\left(\sigma^2+G^2\right)}{(1-\beta_1)^2\mu_1T\sqrt{T}}\left(\mu_2^2-\mu_1^2\right)+\frac{\mu_2^2}{\mu_1\sqrt{T}}\left(\sigma^2+G^2\right)+\frac{\beta_1d\left(\sigma^2+G^2\right)}{2\mu_1(1-\beta_1)\sqrt{T}}\left(\mu_2^2-\mu_1^2\right)\\ &+\frac{\beta_1D^2}{\mu_1(1-\beta_1)\sqrt{T}}+\frac{\beta_1^3D_\infty^2}{\mu_1(1-\beta_1)^3\sqrt{T}}+\frac{\beta_1\mu_2^2}{(1-\beta_1)\mu_1\sqrt{T}}\left(\sigma^2+G^2\right)\\ &=O\left(\frac{1}{\sqrt{T}}\right)+O\left(\frac{1}{T\sqrt{T}}\right)=O\left(\frac{1}{\sqrt{T}}\right).\end{aligned}$$

□

Remark D.4.12. *The leading item of convergence order of* ADAM *should be* $O\left(\frac{\widetilde{C}}{\sqrt{T}}\right)$, *where* $\widetilde{C} = \frac{D^2}{2\mu_1} + \frac{\mu_2^2}{\mu_1}\left(\sigma^2 + G^2\right) + \frac{\beta_1 d\left(\sigma^2 + G^2\right)}{2\mu_1(1-\beta_1)}\left(\mu_2^2 - \mu_1^2\right) + \frac{\beta_1 D^2}{\mu_1(1-\beta_1)} + \frac{\beta_1^3 D_\infty^2}{\mu_1(1-\beta_1)^3} + \frac{\beta_1 \mu_2^2}{(1-\beta_1)\mu_1}\left(\sigma^2 + G^2\right)$. *With fixed* $L, \sigma, G, \beta_1, D, D_\infty, \widetilde{C} = O\left(\frac{d}{\epsilon^2}\right)$, *which also contains as well as dimension d, here with bigger* ϵ, *the order should be better, this also supports the discussion in our main paper.*

The analysis of SADAM is similar to ADAM, by replacing the bounded pairs (μ_1,μ_2) with (μ_3,μ_4), we briefly give convergence result below.

Proof. SADAM case:

$$f(\bar{x}_t) - f^* \leq \frac{D^2}{2\eta\mu_3 T} + \frac{\eta\beta_1^2 d\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2\mu_3 T}\left(\mu_4^2 - \mu_3^2\right) + \frac{\eta\mu_4^2}{\mu_3}\left(\sigma^2 + G^2\right) + \frac{\beta_1 d\left(\sigma^2 + G^2\right)}{2\eta\mu_3(1-\beta_1)T}\left(\mu_4^2 - \mu_3^2\right) + \frac{\eta\beta_1 D^2}{\mu_3(1-\beta_1)} + \frac{\eta\beta_1^3 D_\infty^2}{\mu_3(1-\beta_1)^3} + \frac{\eta\beta_1\mu_4^2}{(1-\beta_1)\mu_3}\left(\sigma^2 + G^2\right)$$

By plugging the stepsize $\eta = O\left(\frac{1}{\sqrt{T}}\right)$, we get the convergence rate of SADAM in non-strongly convex case.

$$\begin{aligned} f(\bar{x}_t) - f^* &\leq \frac{D^2}{2\mu_3\sqrt{T}} + \frac{\beta_1^2 d\left(\sigma^2 + G^2\right)}{(1-\beta_1)^2\mu_3 T\sqrt{T}}\left(\mu_4^2 - \mu_3^2\right) + \frac{\mu_4^2}{\mu_3\sqrt{T}}\left(\sigma^2 + G^2\right) + \frac{\beta_1 d\left(\sigma^2 + G^2\right)}{2\mu_3(1-\beta_1)\sqrt{T}}\left(\mu_4^2 - \mu_3^2\right) \\ &+ \frac{\beta_1 D^2}{\mu_3(1-\beta_1)\sqrt{T}} + \frac{\beta_1^3 D_\infty^2}{\mu_3(1-\beta_1)^3\sqrt{T}} + \frac{\beta_1\mu_4^2}{(1-\beta_1)\mu_3\sqrt{T}}\left(\sigma^2 + G^2\right) \\ &= O\left(\frac{1}{\sqrt{T}}\right) + O\left(\frac{1}{T\sqrt{T}}\right) = O\left(\frac{1}{\sqrt{T}}\right). \end{aligned}$$

For brevity,

$$f(\bar{x}_t) - f^* = O\left(\frac{1}{\sqrt{T}}\right).$$

□

Remark D.4.13. *The leading item of convergence order of* SADAM *should be* $O\left(\frac{\widetilde{C}}{\sqrt{T}}\right)$, *where* $\widetilde{C} = \frac{D^2}{2\mu_3} + \frac{\mu_4^2 d}{\mu_3}\left(\sigma^2 + G^2\right) + \frac{\beta_1 d\left(\sigma^2 + G^2\right)}{2\mu_3(1-\beta_1)}\left(\mu_4^2 - \mu_3^2\right) + \frac{\beta_1 D^2}{\mu_3(1-\beta_1)} + \frac{\beta_1^3 D_\infty^2}{\mu_3(1-\beta_1)^3} + \frac{\beta_1\mu_4^2}{(1-\beta_1)\mu_3}\left(\sigma^2 + G^2\right)$. *With fixed* $L, \sigma, G, \beta_1, D, D_\infty$, $\widetilde{C} = O\left(d\beta log\left(1 + e^{\beta}\right)\right) = O\left(d\beta^2\right)$, *with small* β, *the* SADAM *will be similar to SGD convergence rate, and* β *is a much smaller number compared with* $1/\epsilon$, *proving that* SADAM *method perfoms better than* ADAM *in terms of convergence rate.*

D.4.5. P-L Condition

Suppose that strongly convex assumption holds, we can easily deduce the P-L condition (see Lemma D.4.14), which shows that P-L condition is much weaker than strongly convex condition. And we further prove the convergence of Adam-type optimizer (Adam and Sadam) under the P-L condition in non-strongly convex case, which can be extended to the strongly convex case as well.

Lemma D.4.14. *Suppose that f is continuously diffentiable and strongly convex with parameter γ. Then f has the unique minimizer, denoted as $f^* = f(x^*)$. Then for any $x \in \mathbb{R}^d$, we have*

$$\|\nabla f(x)\|^2 \geq 2\gamma\big(f(x) - f^*\big).$$

Proof. From strongly convex assumption,

$$\begin{aligned} f^* &\geq f(x) + \nabla f(x)^T(x^* - x) + \frac{\gamma}{2}\|x^* - x\|^2 \\ &\geq f(x) + \min_{\xi}\Big(\nabla f(x)^T\xi + \frac{\gamma}{2}\|\xi\|^2\Big) \\ &= f(x) - \frac{1}{2\gamma}\|\nabla f(x)\|^2 \end{aligned}$$

Letting $\xi = x^* - x$, when $\xi = -\frac{\nabla f(x)}{\gamma}$, the quadratic function can achieve its minimum.□

We restate our theorems under PL condition.

Theorem D.4.15. *Suppose $f(x)$ satisfies Assumption 1 and PL condition (with parameter λ) in non-strongly convex case and $v_t \geq v_{t-1}$. Let $\eta_t = \eta = O\big(\frac{1}{T}\big)$,*

Adam *and* Sadam *have convergence rate*

$$E\big[f(x_t) - f^*\big] \leq O\Big(\frac{1}{T}\Big).$$

Proof. Adam case:

Starting from L-smoothness, and borrowing the previous results we already have

$$\begin{aligned} E[f(z_t+1) - f(z_t)] \leq{}& \frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_{t-1,j}}+\epsilon} - \frac{1}{\sqrt{v_{t,j}}+\epsilon}\right] \\ &+ \frac{L^2\eta^2\mu_2^2}{2}\Big(\frac{\beta_1}{1-\beta_1}\Big)^2\big(\sigma^2+G^2\big) + \frac{\eta^2\mu_2^2}{2}\big(\sigma^2+G^2\big) - E\left\langle\nabla f(x_t), \frac{\eta}{\sqrt{v_t}+\epsilon}\odot g_t\right\rangle \\ &+ \frac{L\eta^2\beta_1^2\big(\sigma^2+G^2\big)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\Big(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\Big)^2 - \Big(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\Big)^2\right] + L\eta^2\mu_2^2\big(\sigma^2+G^2\big) \end{aligned}$$

$$E\left\langle \nabla f(x_t), \frac{1}{\sqrt{v_t}+\epsilon} \odot g_t \right\rangle \geq \mu_1 \| \nabla f(x_t) \|^2$$

Therefore, we get:

$$\begin{aligned} E[f(z_t+1) - f(z_t)] \leq & \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_{t-1,j}}+\epsilon} - \frac{1}{\sqrt{v_{t,j}}+\epsilon}\right] \\ & + \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) + \frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right) - \eta\mu_1\| \nabla f(x_t)\|^2 \\ & + \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right) \end{aligned}$$

From P-L condition assumption,

$$\begin{aligned} E[f(z_t+1)] \leq & E[f(z_t)] + \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_{t-1,j}}+\epsilon} - \frac{1}{\sqrt{v_{t,j}}+\epsilon}\right] \\ & + \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) + \frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right) - 2\lambda\eta\mu_1 E\left[f(x_t) - f^*\right] \\ & + \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{t-1,j}}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_{t,j}}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right) \end{aligned}$$

From convexity,

$$\begin{aligned} f(z_t+1) &\geq f(x_t+1) + \frac{\beta_1}{1-\beta_1} < \nabla f(x_t+1), x_t+1-x_t > \\ &= f(x_t+1) + \frac{\beta_1}{1-\beta_1} < \nabla f(x_t+1), \frac{\eta}{\sqrt{v_t}+\epsilon} \odot m_t > \end{aligned}$$

From L-smoothness,

$$f(z_t) \leq f(x_t) + \frac{\beta_1}{1-\beta_1} < \nabla f(x_t), x_t - x_t - 1 > + \frac{L}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 \| x_t - x_t - 1\|^2.$$

Then we can obtain

$$
\begin{aligned}
&E[f(x_t+1)] + \frac{\beta_1}{1-\beta_1} E\left[< \nabla f(x_t+1), \frac{\eta}{\sqrt{v_t}+\epsilon} \odot m_t > \right] \\
&\le E[f(x_t)] + \frac{\beta_1}{1-\beta_1} E[< \nabla f(x_t), x_t - x_t - 1 >] + \frac{L}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\|x_t - x_t - 1\|^2\right] \\
&+ \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_t-1,j}+\epsilon} - \frac{1}{\sqrt{v_t,j}+\epsilon}\right] \\
&+ \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) + \frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right) - 2\lambda\eta\mu_1 E\left[f(x_t) - f^*\right] \\
&+ \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right) \\
&= E[f(x_t)] + \frac{\beta_1}{1-\beta_1} E\left[< \nabla f(x_t), \frac{\eta}{\sqrt{v_t-1}+\epsilon} \odot m_t - 1 > \right] + \frac{L\eta^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\left\|\frac{1}{\sqrt{v_t-1}+\epsilon} \odot m_t - 1\right\|^2\right] \\
&+ \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_t-1,j}+\epsilon} - \frac{1}{\sqrt{v_t,j}+\epsilon}\right] \\
&+ \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) + \frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right) - 2\lambda\eta\mu_1 E\left[f(x_t) - f^*\right] \\
&+ \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

By rearranging,

$$
\begin{aligned}
E[f(x_t+1)] \le E[f(x_t)] &+ \frac{\beta_1\eta}{1-\beta_1}\left(E\left[< \nabla f(x_t) m \frac{1}{\sqrt{v_t-1}+\epsilon} \odot m_t - 1 > \right] - E\left[< \nabla f(x_t+1), \frac{1}{\sqrt{v_t}+\epsilon} \odot m_t > \right]\right) \\
&+ \frac{L\eta^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\left\|\frac{1}{\sqrt{v_t-1}+\epsilon} \odot m_t - 1\right\|^2\right] + \frac{\eta\beta_1}{1-\beta_1} G\sqrt{\sigma^2+G^2} E\left[\sum_{j=1}^{d} \frac{1}{\sqrt{v_t-1,j}+\epsilon} - \frac{1}{\sqrt{v_t,j}+\epsilon}\right] \\
&+ \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right) + \frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right) - 2\lambda\eta\mu_1 E\left[f(x_t) - f^*\right] \\
&+ \frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

From the fact $\pm < a, b > \ \le \frac{1}{2}a^2 + \frac{1}{2}b^2$, and Lemma D.4.1, D.4.4,

$$
\begin{aligned}
E\left[< \nabla f(x_t), \frac{1}{\sqrt{v_t-1}+\epsilon} \odot m_t - 1 > \right] &= E\left[< \nabla f(x_t+1) \odot \sqrt{\frac{1}{\sqrt{v_t-1}+\epsilon}}, m_t \odot \sqrt{\frac{1}{\sqrt{v_t-1}+\epsilon}} > \right] \\
&\le \frac{G^2\mu_2}{2} + \frac{\left(\sigma^2+G^2\right)\mu_2}{2} \le \left(\sigma^2+G^2\right)\mu_2
\end{aligned}
$$

Similar,

$$-E\left[<\nabla f(x_t+1), \frac{1}{\sqrt{v_t+\epsilon}}\odot m_t>\right] = -E\left[<\nabla f(x_t+1)\odot\sqrt{\frac{1}{\sqrt{v_t-1}+\epsilon}}, m_t\odot\sqrt{\frac{1}{\sqrt{v_t-1}+\epsilon}}>\right]$$
$$\leq \frac{G^2\mu_2}{2}+\frac{\left(\sigma^2+G^2\right)\mu_2}{2}\leq\left(\sigma^2+G^2\right)\mu_2$$

Then,

$$E[f(x_t+1)]\leq E[f(x_t)]+\frac{2\beta_1\eta\mu_2}{1-\beta_1}\left(\sigma^2+G^2\right)+\frac{L\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)$$
$$+\frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_t-1,j}+\epsilon}-\frac{1}{\sqrt{v_t,j}+\epsilon}\right]$$
$$+\frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right)-2\lambda\eta\mu_1E\left[f(x_t)-f^*\right]$$
$$+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\left(\sigma^2+G^2\right)$$

$$E\left[f(x_t+1)-f^*\right]\leq(1-2\lambda\eta\mu_1)E\left[f(x_t)-f^*\right]+\frac{2\beta_1\eta\mu_2}{1-\beta_1}\left(\sigma^2+G^2\right)+\frac{L\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)$$
$$+\frac{\eta\beta_1}{1-\beta_1}G\sqrt{\sigma^2+G^2}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_t-1,j}+\epsilon}-\frac{1}{\sqrt{v_t,j}+\epsilon}\right]$$
$$+\frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\sigma^2+G^2\right)+\frac{\eta^2\mu_2^2}{2}\left(\sigma^2+G^2\right)$$
$$+\frac{L\eta^2\beta_1^2\left(\sigma^2+G^2\right)}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\left(\sigma^2+G^2\right)$$
$$\leq(1-2\lambda\eta\mu_1)E\left[f(x_t)-f^*\right]+\left(\frac{2\beta_1\eta\mu_2}{1-\beta_1}+\frac{L\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\right.$$
$$+\frac{\eta\beta_1}{1-\beta_1}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_t-1,j}+\epsilon}-\frac{1}{\sqrt{v_t,j}+\epsilon}\right]+\frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2+\frac{\eta^2\mu_2^2}{2}$$
$$\left.+\frac{L\eta^2\beta_1^2}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2-\left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right]+L\eta^2\mu_2^2\right)\left(\sigma^2+G^2\right)$$

The last inequality holds because $G\sqrt{\sigma^2+G^2}\leq\sigma^2+G^2$.

Let

$$\theta=1-2\lambda\eta\mu_1$$

$$\Theta_t = \left(\frac{2\beta_1\eta\mu_2}{1-\beta_1} + \frac{L\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta\beta_1}{1-\beta_1}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_t-1,j}+\epsilon} - \frac{1}{\sqrt{v_t,j}+\epsilon}\right] + \frac{L^2\eta^2\mu_2^2}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta^2\mu_2^2}{2} + \frac{L\eta^2\beta_1^2}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_t-1,j}+\epsilon}\right)^2 - \left(\frac{1}{\sqrt{v_t,j}+\epsilon}\right)^2\right] + L\eta^2\mu_2^2\right)\left(\sigma^2+G^2\right)$$

then we have

$$E\left[f(x_t+1)-f^*\right] \le \theta E\left[f(x_t)-f^*\right] + \Theta_t .$$

Let $\Phi_t = E\left[f(x_t)-f^*\right]$, then $\Phi_1 = E\left[f(x_1)-f^*\right]$,

$$\begin{aligned}\Phi_{t+1} &\le \theta\Phi_t + \Theta_t \le \theta^2\Phi_{t-1} + \theta\Theta_{t-1} + \Theta_t \\ &\cdots \\ &\le \theta^t\Phi_1 + \theta^{t-1}\Theta_1 + \cdots + \theta\Theta_{t-1} + \Theta_t \\ &\overset{\theta<1}{\le} \theta^t\Phi_1 + \Theta_1 + \cdots + \Theta_{t-1} + \Theta_t .\end{aligned}$$

Let $t = T$,

$$\begin{aligned}\Phi_{T+1} &\le \theta^T\Phi_1 + \Theta_1 + \cdots + \Theta_{T-1} + \Theta_T \\ &\le \theta^T\Phi_1 + \left(\frac{2\beta_1\eta\mu_2 T}{1-\beta_1} + \frac{L\eta^2\mu_2^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta\beta_1}{1-\beta_1}E\left[\sum_{j=1}^{d}\frac{1}{\sqrt{v_{0,j}+\epsilon}} - \frac{1}{\sqrt{v_{T,j}+\epsilon}}\right]\right. \\ &\quad + \frac{L^2\eta^2\mu_2^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta^2\mu_2^2 T}{2} \\ &\quad \left. + \frac{L\eta^2\beta_1^2}{(1-\beta_1)^2}E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{v_{0,j}+\epsilon}}\right)^2 - \left(\frac{1}{\sqrt{v_{T,j}+\epsilon}}\right)^2\right] + L\eta^2\mu_2^2 T\right)\left(\sigma^2+G^2\right) \\ &\le \theta^T\Phi_1 + \left(\frac{2\beta_1\eta\mu_2 T}{1-\beta_1} + \frac{L\eta^2\mu_2^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta\beta_1 d}{1-\beta_1}(\mu_2-\mu_1) + \frac{L^2\eta^2\mu_2^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\right. \\ &\quad \left. + \frac{\eta^2\mu_2^2 T}{2} + \frac{L\eta^2\beta_1^2 d}{(1-\beta_1)^2}\left(\mu_2^2-\mu_1^2\right) + L\eta^2\mu_2^2 T\right)\left(\sigma^2+G^2\right) \\ &= \theta^T\Phi_1 + O(\eta T) + O\left(\eta^2 T\right) + O(\eta) + O\left(\eta^2\right)\end{aligned}$$

From the above inequality, η should be set less than $O\left(\frac{1}{T}\right)$ to ensure all items in the RHS small enough.

Set $\eta = \frac{1}{T^2}$, then $\theta = 1 - 2\lambda\eta\mu_1 = 1 - \frac{2\lambda\mu_1}{T^2}$

$$\begin{aligned}\Phi_{T+1} &= \theta^T \Phi_1 + O\left(\frac{1}{T}\right) + O\left(\frac{1}{T^3}\right) + O\left(\frac{1}{T^2}\right) + O\left(\frac{1}{T^4}\right) \\ &= \theta^T \Phi_1 + O\left(\frac{1}{T}\right) \to 0\end{aligned}$$

With appropriate η, we can derive the convergence rate under P-L condition (strongly convex) case.

The proof of SADAM is exactly same as ADAM, by replacing the bounded pairs (μ_1,μ_2) with (μ_3,μ_4), and we can also get:

$$\begin{aligned}\Phi_{T+1} &\le \theta^T \Phi_1 + \Theta_1 + \cdots + \Theta_{T-1} + \Theta_T \\ &\le \theta^T \Phi_1 + \left(\frac{2\beta_1\eta\mu_4 T}{1-\beta_1} + \frac{L\eta^2\mu_4^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta\beta_1}{1-\beta_1} E\left[\sum_{j=1}^{d} \frac{1}{softplus\left(v_{0,j}\right)} - \frac{1}{softplus\left(v_{T,j}\right)}\right]\right. \\ &+ \frac{L^2\eta^2\mu_4^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta^2\mu_4^2 T}{2} \\ &+ \left.\frac{L\eta^2\beta_1^2}{(1-\beta_1)^2} E\left[\sum_{j=1}^{d}\left(\frac{1}{softplus\left(v_{0,j}\right)}\right)^2 - \left(\frac{1}{softplus\left(v_{T,j}\right)}\right)^2\right] + L\eta^2\mu_4^2 T\right)\left(\sigma^2 + G^2\right) \\ &\le \theta^T \Phi_1 + \left(\frac{2\beta_1\eta\mu_4 T}{1-\beta_1} + \frac{L\eta^2\mu_4^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2 + \frac{\eta\beta_1 d}{1-\beta_1}(\mu_4 - \mu_3) + \frac{L^2\eta^2\mu_4^2 T}{2}\left(\frac{\beta_1}{1-\beta_1}\right)^2\right. \\ &+ \left.\frac{\eta^2\mu_4^2 T}{2} + \frac{L\eta^2\beta_1^2 d}{(1-\beta_1)^2}\left(\mu_4^2 - \mu_3^2\right) + L\eta^2\mu_4^2 T\right)\left(\sigma^2 + G^2\right) \\ &= \theta^T \Phi_1 + O(\eta T) + O\left(\eta^2 T\right) + O(\eta) + O\left(\eta^2\right)\end{aligned}$$

By setting appropriate η, we can also prove the SADAM converges under PL condition (and strongly convex).

Set $\eta = O\left(\frac{1}{T^2}\right)$,

$$E\left[f(x_{T+1}) - f^*\right] \le \left(1 - \frac{2\lambda\mu_3}{T^2}\right)^T E\left[f(x_1) - f^*\right] + O\left(\frac{1}{T}\right).$$

Overall, we have proved ADAM algorithm and SADAM in all commonly used conditions, our designed algorithms always enjoy the same convergence rate compared with ADAM, and even get better results with appropriate choice of β defined in *softplus* function. The proof procedure can be easily extended to other adaptive gradient algorithms, and theoretical results support the discussion and experiments in our main paper.

References

[1]. Ghadimi S, Lan G, Stochastic first-and zeroth-order methods for nonconvex stochastic programming, SIAM Journal on Optimization 23 (4) (2013) 2341–2368.

[2]. Wright SJ, Nocedal J, Numerical optimization, Springer Science 35 (6768) (1999) 7.

[3]. Wilson AC, Recht B, Jordan MI, A lyapunov analysis of momentum methods in optimization, arXiv preprint arXiv:1611.02635

[4]. Yang T, Lin Q, Li Z, Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization, arXiv preprint arXiv:1604.03257

[5]. Duchi J, Hazan E, Singer Y, Adaptive subgradient methods for online learning and stochastic optimization, Journal of Machine Learning Research 12 (Jul) (2011) 2121–2159.

[6]. Kingma DP, Ba J, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980

[7]. Zeiler MD, Adadelta: an adaptive learning rate method, arXiv preprint arXiv:1212.5701

[8]. Tieleman T, Hinton G, Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural networks for machine learning 4 (2) (2012) 26–31.

[9]. He K, Zhang X, Ren S, Sun J, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[10]. Zagoruyko S, Komodakis N, Wide residual networks, arXiv preprint arXiv:1605.07146

[11]. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[12]. Reddi SJ, Kale S, Kumar S, On the convergence of adam and beyond

[13]. Luo L, Xiong Y, Liu Y, Sun X, Adaptive gradient methods with dynamic bound of learning rate, arXiv preprint arXiv:1902.09843

[14]. Zaheer M, Reddi S, Sachan D, Kale S, Kumar S, Adaptive methods for nonconvex optimization, in: Advances in Neural Information Processing Systems, 2018, pp. 9815–9825.

[15]. Zhou D, Tang Y, Yang Z, Cao Y, Gu Q, On the convergence of adaptive gradient methods for nonconvex optimization, arXiv preprint arXiv:1808.05671

[16]. Chen X, Liu S, Sun R, Hong M, On the convergence of a class of adam-type algorithms for non-convex optimization, arXiv preprint arXiv:1808.02941

[17]. De S, Mukherjee A, Ullah E, Convergence guarantees for rmsprop and adam in non-convex optimization and an empirical comparison to nesterov acceleration

[18]. Staib M, Reddi SJ, Kale S, Kumar S, Sra S, Escaping saddle points with adaptive gradient methods, arXiv preprint arXiv:1901.09149

[19]. Chen J, Gu Q, Closing the generalization gap of adaptive gradient methods in training deep neural networks, arXiv preprint arXiv:1806.06763

[20]. Reddi SJ, Hefny A, Sra S, Poczos B, Smola AJ, On variance reduction in stochastic gradient descent and its asynchronous variants, in: Advances in Neural Information Processing Systems, 2015, pp. 2647–2655.

[21]. Kleinberg R, Li Y, Yuan Y, An alternative view: When does sgd escape local minima?, in: International Conference on Machine Learning, 2018, pp. 2703–2712.

[22]. Keskar NS, Mudigere D, Nocedal J, Smelyanskiy M, Tang PTP, On large-batch training for deep learning: Generalization gap and sharp minima, arXiv preprint arXiv:1609.04836

[23]. Chaudhari P, Choromanska A, Soatto S, LeCun Y, Baldassi C, Borgs C, Chayes J, Sagun L, Zecchina R, Entropy-sgd: Biasing gradient descent into wide valleys, arXiv preprint arXiv:1611.01838

[24]. Li H, Xu Z, Taylor G, Studer C, Goldstein T, Visualizing the loss landscape of neural nets, in: Advances in Neural Information Processing Systems, 2018, pp. 6389–6399.

[25]. Dozat T, Incorporating nesterov momentum into adam

[26]. Simonyan K, Zisserman A, Very deep convolutional networks for largescale image recognition, arXiv preprint arXiv:1409.1556

[27]. Merity S, Keskar NS, Socher R, Regularizing and Optimizing LSTM Language Models, arXiv preprint arXiv:1708.02182

[28]. Keskar NS, Socher R, Improving generalization performance by switching from adam to sgd, arXiv preprint arXiv:1712.07628

[29]. Mikolov T, Karafiát M, Burget L, Černockỳ J, Khudanpur S, Recurrent neural network based language model, in: Eleventh annual conference of the international speech communication association, 2010.

[30]. Bradbury J, Merity S, Xiong C, Socher R, Quasi-recurrent neural networks, arXiv preprint arXiv:1611.01576

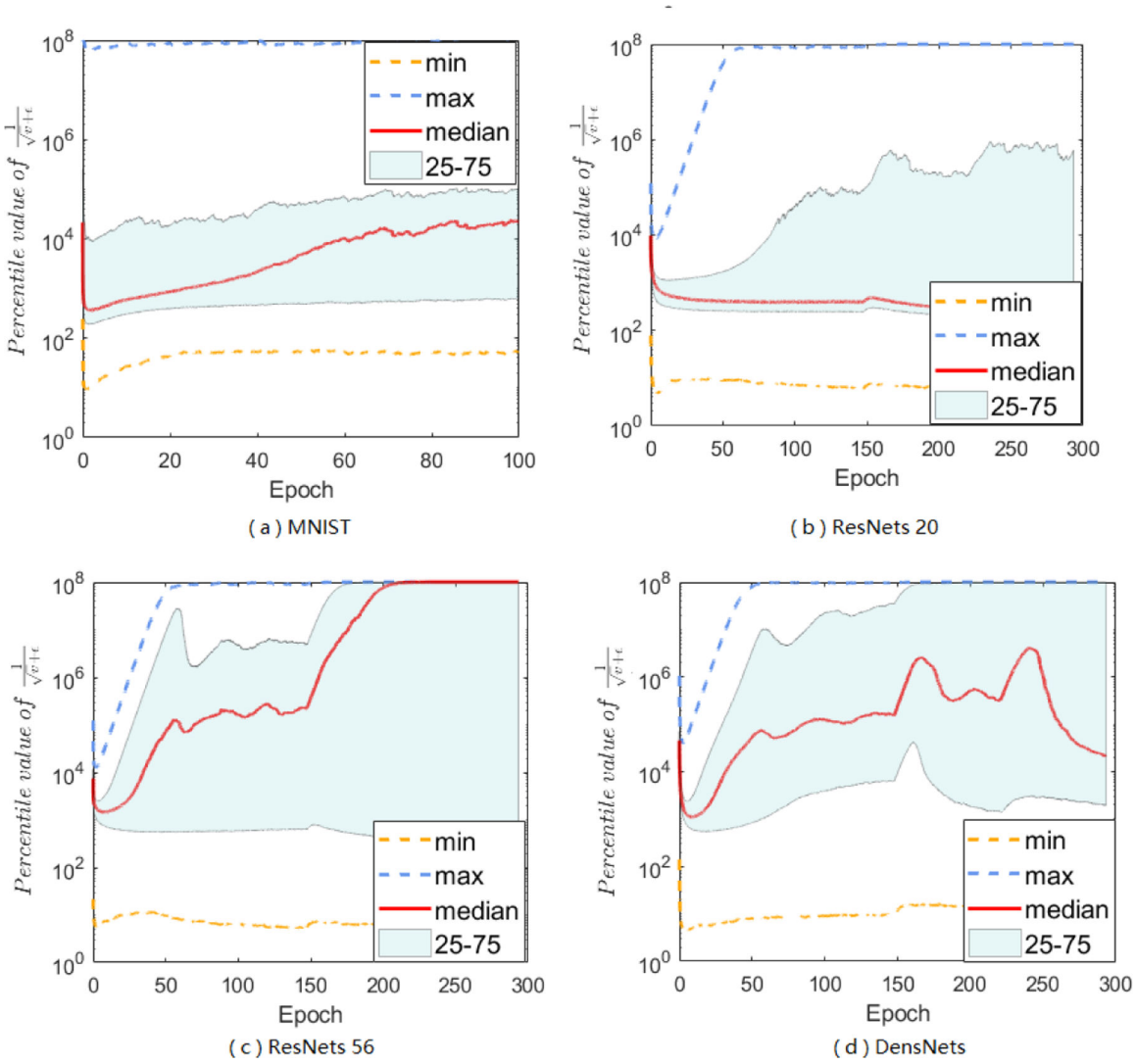

Figure 1:

Range of the A-LR in ADAM over iterations in four settings: (a) CNN on MNIST, (b) ResNet20 on CIFAR-10, (d) ResNet56 on CIFAR-10, (d) DenseNets on CIFAR-10. We plot the min, max, median, and the 25 and 75 percentiles of the A-LR across dimensions (the elements in $\frac{1}{\sqrt{v_t}+\epsilon}$)

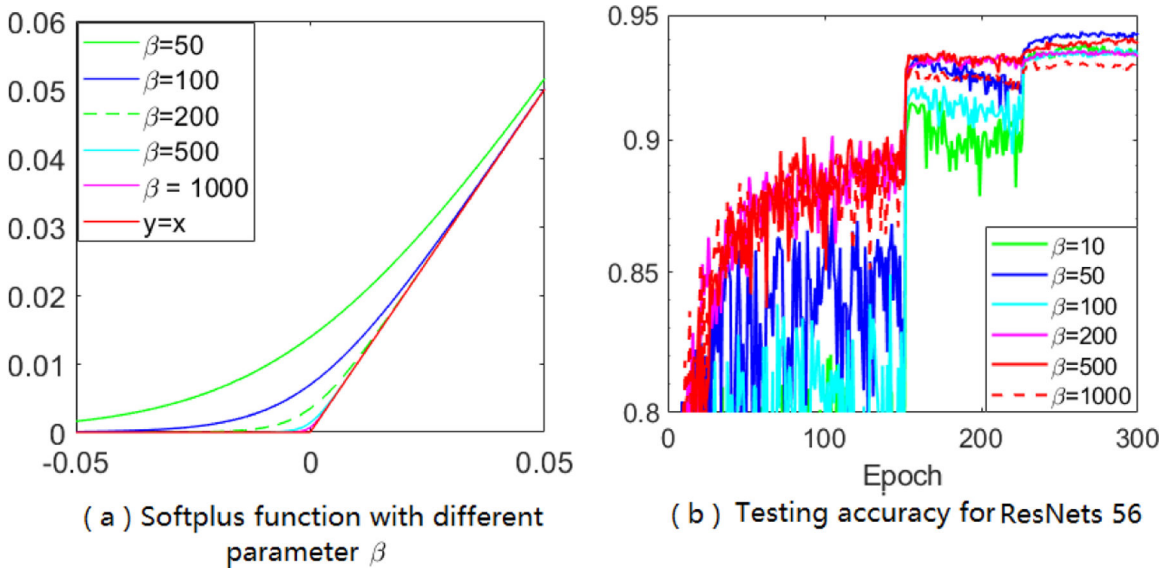

Figure 2:
Behavior of the softplus function, and the test performance of our SADAM algorithm.

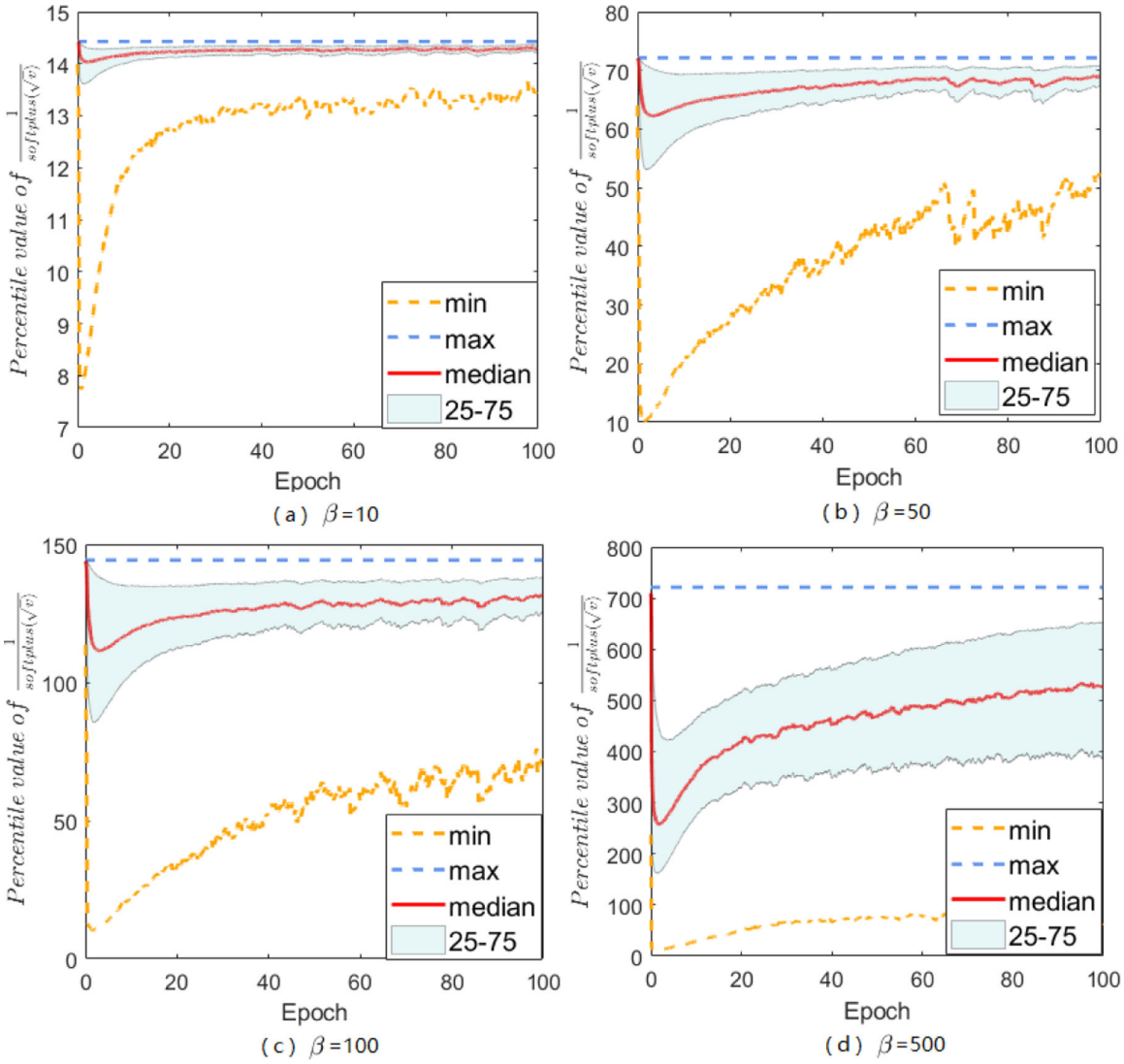

Figure 3:
Behavior of the A-LR in the SADAM method with different choices of β (CNN on the MNIST data).

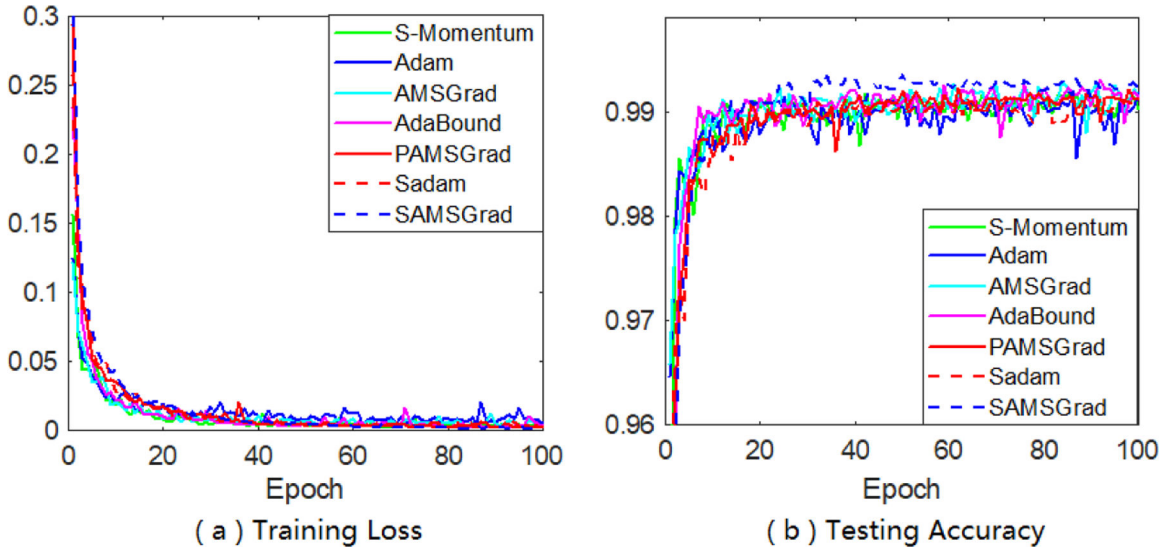

Figure 4:
Training loss and test accuracy on MNIST.

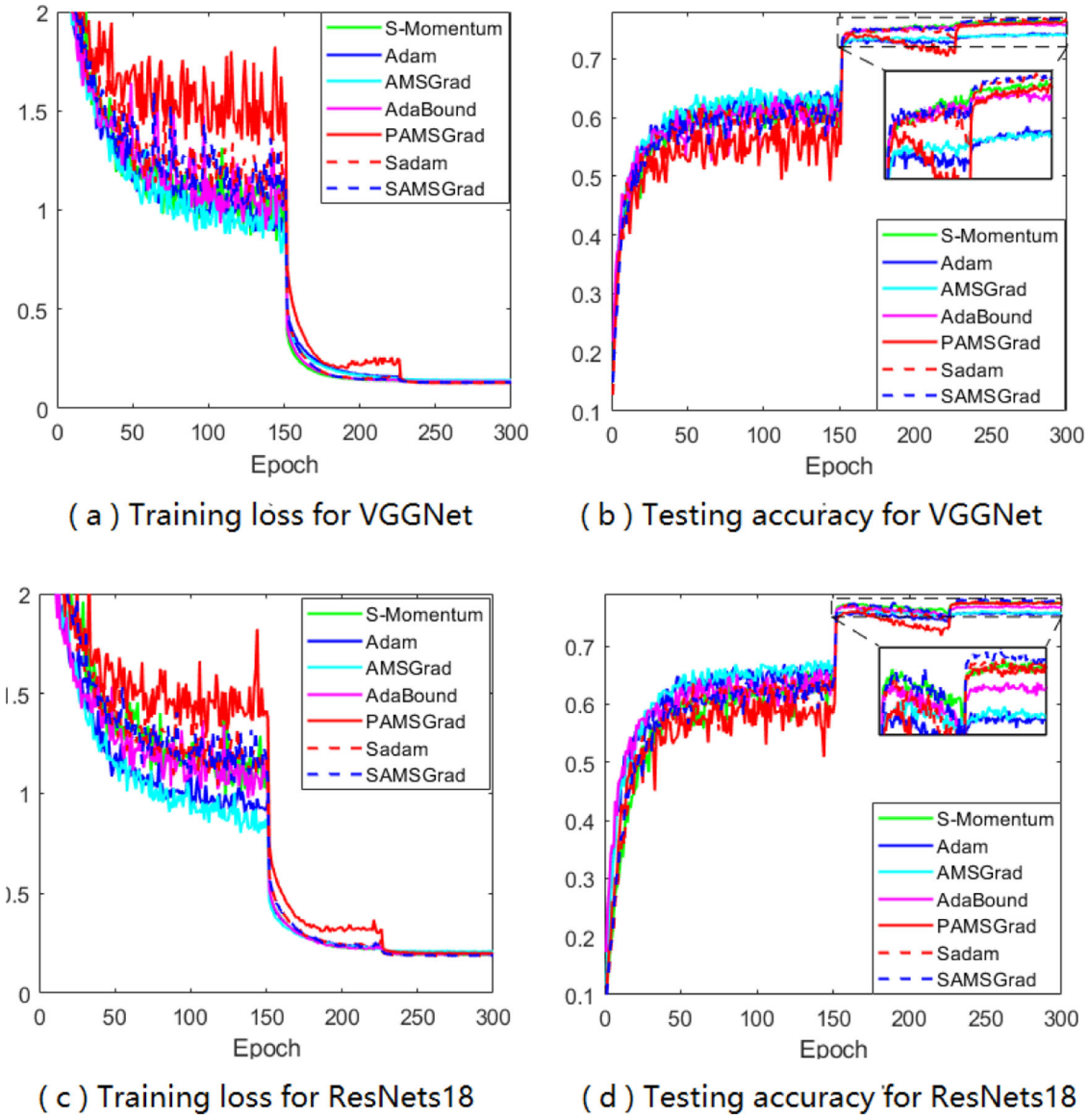

Figure 5:
Training loss and test accuracy of two CNN architectures on CIFAR-100.

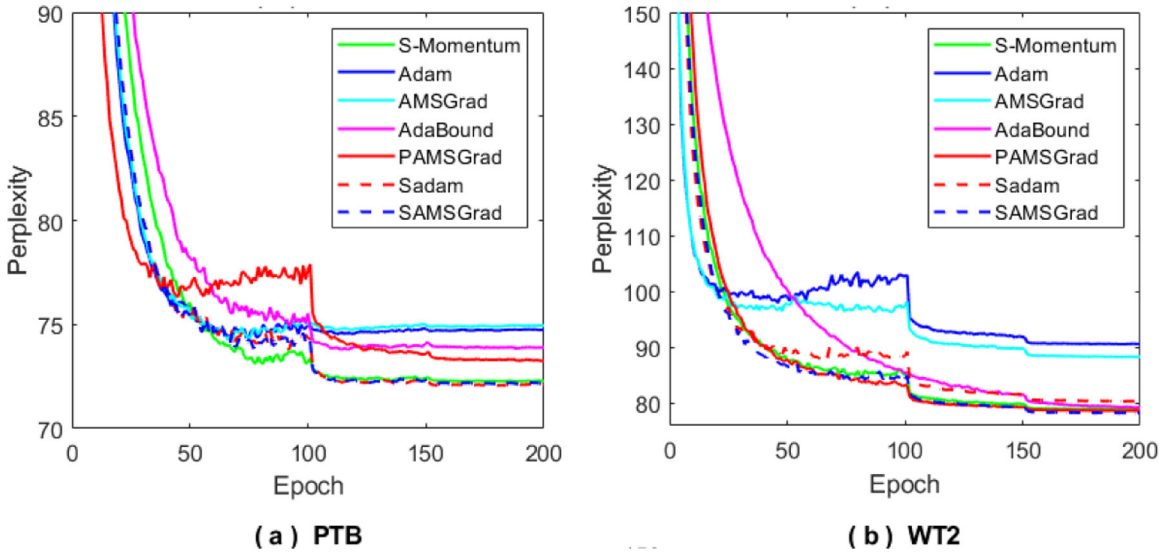

Figure 6:
Perplexity curves on the test set on 3-layer LSTM models over PTB and WT2 datasets

Table 1:

Test Accuracy(%) of ADAM for different ϵ.

ϵ	ResNets 20	ResNets 56	DenseNets	ResNet 18	VGG
10^{-1}	92.51 ± 0.13	94.29 ± 0.10	94.78 ± 0.19	77.21 ± 0.26	76.05 ± 0.27
10^{-2}	92.88 ± 0.21	94.15 ± 0.17	94.35 ± 0.10	76.64 ± 0.24	75.69 ± 0.16
10^{-4}	92.03 ± 0.21	93.62 ± 0.18	94.15 ± 0.12	76.19 ± 0.20	74.45 ± 0.19
10^{-6}	92.99 ± 0.22	93.56 ± 0.15	94.24 ± 0.24	76.09 ± 0.20	74.20 ± 0.33
10^{-8}	91.68 ± 0.12	92.82 ± 0.09	93.32 ± 0.06	76.14 ± 0.24	74.18 ± 0.15

Table 2:

Test Accuracy(%) of AMSGRAD for different ϵ.

ϵ	ResNets 20	ResNets 56	DenseNets	ResNet 18	VGG
10^{-1}	92.80 ± 0.22	94.12 ± 0.07	94.92 ± 0.10	77.26 ± 0.30	75.84 ± 0.16
10^{-2}	92.89 ± 0.07	94.20 ± 0.18	94.43 ± 0.22	76.23 ± 0.26	75.37 ± 0.18
10^{-4}	91.85 ± 0.10	93.50 ± 0.14	94.02 ± 0.18	76.30 ± 0.31	74.44 ± 0.16
10^{-6}	91.98 ± 0.23	93.54 ± 0.16	94.17 ± 0.10	76.14 ± 0.16	74.17 ± 0.28
10^{-8}	91.70 ± 0.12	93.10 ± 0.11	93.71 ± 0.05	76.32 ± 0.11	74.26 ± 0.18

Table 3:

Test Accuracy(%) of CIFAR-10 for ResNets 20, ResNets 56 and DenseNets.

Method	B-LR	ϵ	β	ResNets 20	ResNets 56	DenseNets
S-Momentum [9, 11]	-	-	-	91.25	93.03	94.76
ADAM [14]	10^{-3}	10^{-3}	-	92.56 ± 0.14	93.42 ± 0.16	93.35 ± 0.21
YOGI [14]	10^{-2}	10^{-3}	-	92.62 ± 0.17	93.90 ± 0.21	94.38 ± 0.26
S-Momentum	10^{-1}	-	-	92.73 ± 0.05	94.11 ± 0.15	95.03 ± 0.15
ADAM	10^{-3}	10^{-8}	-	91.68 ± 0.12	92.82 ± 0.09	93.32 ± 0.06
AMSGRAD	10^{-3}	10^{-8}	-	91.7 ± 0.12	93.10 ± 0.11	93.71 ± 0.05
PADAM	10^{-1}	10^{-8}	-	92.7 ± 0.10	94.12 ± 0.12	95.06 ± 0.06
PAMSGRAD	10^{-1}	10^{-8}	-	92.74 ± 0.12	94.18 ± 0.06	**95.21** ± 0.10
ADABOUND	10^{-2}	10^{-8}	-	91.59 ± 0.24	93.09 ± 0.14	94.16 ± 0.10
AMSBOUND	10^{-2}	10^{-8}	-	91.76 ± 0.16	93.08 ± 0.09	94.03 ± 0.11
ADAM$^+$	10^{-1}	0.013	-	92.89 ± 0.13	92.24 ± 0.10	94.54 ± 0.13
AMSGRAD$^+$	10^{-1}	0.013	-	**92.95** ± 0.17	**94.32** ± 0.10	94.58 ± 0.18
SADAM	10^{-2}	-	50	**93.01** ± 0.16	94.26 ± 0.10	95.19 ± 0.18
SAMSGRAD	10^{-2}	-	50	92.88 ±0.10	**94.32** ± 0.18	**95.31** ± 0.15