Understanding Adaptivity in Machine Learning Optimization: Theories and Algorithms

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Xiangyi Chen

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Advisor: Dr. Mingyi Hong

May, 2022

Acknowledgements

First and foremost, I would like to express my gratitude to my advisor Prof. Mingyi Hong for the great flexibility he provided in my PhD research. I am very fortunate to have him as my advisor.

Also, I want to thank Prof. Mehmet Akcakaya, Prof. Jie Ding and Prof. Steven Wu for being in my thesis committee and for their very quick response scheduling the defense.

During my PhD studies, I had the opportunities to collaborate with some brilliant and talented researchers. I would like to thank my collaborators Sijia Liu, Pin-Yu Chen, Steven Wu, Ruoyu Sun, Kaidi Xu, Xingguo Li, Songtao Lu, Haoran Sun, Xinwei Zhang, Tiancong Chen. It was my great pleasure to work with them and learn from them. The materials in this dissertation are also results of collaboration with them.

Many thanks to my friends Ziqi Zhou, Ningyuan Liu, Jianjun Yuan, Yingxue Zhou, Burhaneddin Yaman, Yuejun Ma, to name a few, for the cherished time together. I will never forget these memories.

I would like to offer my special thanks to Ping Li for providing me the internship opportunity at Baidu Research. And I would like to thank my internship colleagues at Baidu, Xiaoyun Li, Hang Zhang, Jun-Kun Wang, Yi Hao, Chen Li, Jerry Chee, Fan Zhou, Peng Yang, Weijie Zhao, for the great experience in Summer 2019.

Last but not the least, I would like to thank my family for all their unwavering love and unconditional support in helping me overcome the difficulties encountered.

Abstract

Optimization plays an indispensable role in modern machine learning, due to its necessity in many aspects, especially in model training. Over the past decade, the rapid development of research in deep machine learning models posed many new challenges for machine learning optimization. As a result, designing efficient and robust optimization algorithms remains an active research area within machine learning. In addition, some new and notable optimization algorithms were proposed to tackle the new challenges in model training. An important class of the new algorithms is motivated by the idea of incorporating adaptation into algorithm design, so that the algorithms can adapt to the local geometry of the optimization landscape. However, some of these newly designed algorithms with adaptation are tailored to achieve superior empirical performance for certain classes of optimization problems but are not well understood theoretically. Thus, the performance of these algorithms is less predictable in other domains or applications. In this thesis, we try to build theories for some algorithms with adaptation. In particular, the result of this thesis can be separated into three parts. In the first part, we analyze a class of algorithms with adaptation, which we call Adam-type algorithms, for nonconvex unconstrained optimization. We provide conditions for these algorithms to converge and shed light on design principles for this class of algorithms. In the second part, we extend the previous analysis to zeroth-order constrained/unconstrained optimization and propose an algorithm called ZO-AdaMM, which has superior performance in generating black-box adversarial attacks. In the third part, we study the gradient clipping operation for differentially private SGD. Gradient clipping adds a form of adaptation to SGD that can potentially hurt convergence. We identify regimes where gradient clipping is not an issue and verify the existence of these regimes in practice. Further, we provide a perturbation mechanism to mitigate the adversarial effect caused by gradient clipping.

Contents

List of Figures

Chapter 1

Introduction

Designing algorithms to better exploit the optimization landscape has always been an important research topic in continuous optimization. Many classical optimization algorithms fall into this category, for example, Newton's Method, Heavy Ball Method [Polyak, 1964], Conjugate Gradient Method [Hestenes and Stiefel, 1952]. There is also a vast amount of more recent literature, e.g., Nesterov Momentum [Nesterov, 1983] and Cubic regularized Newton's method [Nesterov and Polyak, 2006]. These algorithms are all shown to improve the convergence rate for certain classes of optimization problems. In modern machine learning, optimization remains to be an active research area. However, modern machine learning optimization problems usually have highly complicated structures, especially optimization problems involving deep neural networks. Such a property makes it very difficult to analyze and exploit the landscape of these optimization problems. Thus, optimization algorithms designed for machine learning in the past decade are more driven or popularized by their empirical performance, instead of having strong theoretical guarantees. A notable class of algorithms in the aforementioned category is adaptive gradient (momentum) methods designed for neural network training. Representative algorithms include Adam [Kingma and Ba, 2014], AdaGrad [Duchi et al., 2011], AdaDelta [Zeiler, 2012], to name a few. A common feature of these algorithms is their adaptation to local geometries by gradually adjusting preconditioning matrices or stepsizes over time. However, due to the lack of theoretical understanding of many of these algorithms, their behavior is less predictable when applied to new problems. Some of these algorithms are even found to be divergent in certain cases [Reddi et al., 2018].

In this work, we take a step toward bridging the gap between theory and practice for some of these algorithms. The following chapters are summarized below.

- In Chapter 2, we study a class of adaptive gradient (momentum) methods (AdaMM) that update the search directions and learning rates simultaneously using past gradients. This class, which we refer to as the "Adam-type", includes the popular algorithms such as Adam [Kingma and Ba, 2014] , AMSGrad [Reddi et al., 2018] , AdaGrad [Duchi et al., 2011]. We develop an analysis framework and a set of mild sufficient conditions that guarantee the convergence of the Adam-type methods, with a convergence rate of order $O(\log T/\sqrt{T})$ for non-convex stochastic optimization. Our convergence analysis applies to a new algorithm called AdaFom (AdaGrad with First Order Momentum). We show that the conditions are essential, by identifying concrete examples in which violating the conditions makes an algorithm diverge. Besides providing one of the first comprehensive analyses for Adam-type methods in the non-convex setting, our results can also help the practitioners easily monitor the progress of algorithms and determine their convergence behavior. These results are included in Chen et al. [2019a].

- In Chapter 3, we extend the analysis approach in Chapter 2 to zeroth-order constrained and unconstrained nonconvex optimization. We propose a zeroth-order adaptive momentum method (ZO-AdaMM), that generalizes adaptive momentum methods to the gradient-free regime. We show that the convergence rate of ZO-AdaMM for nonconvex optimization is roughly a factor of $O(\sqrt{d})$ worse than that of the first-order AdaMM algorithm, where d is problem size. In particular, we provide a deep understanding of why Mahalanobis distance-based projection matters in the convergence of ZO-AdaMM. As a byproduct, our analysis makes the first step toward understanding adaptive learning rate methods for nonconvex constrained optimization. Furthermore, we demonstrate two applications, designing per-image and universal adversarial attacks from black-box neural networks, respectively. We perform experiments on ImageNet and empirically show that ZO-AdaMM converges much faster to a solution of high accuracy compared with some other state-of-the-art zeroth-order optimization methods. These results are parts of Chen et al. [2019b].

- In Chapter 4, we study (differentially) private stochastic gradient descent (SGD) with gradient clipping, an algorithm popularly used for training machine learning models privately. The gradient clipping operation can be viewed as a special form of adaptivity that can potentially hurt convergence in certain regimes. Yet, this algorithm sometimes mysteriously performs well in practice. We first demonstrate how gradient clipping can prevent SGD from converging to a stationary point. We then provide a theoretical analysis that fully quantifies the clipping bias on convergence with a disparity measure between the gradient distribution and geometrically symmetric distribution. Our empirical evaluation further suggests that the gradient distributions along the trajectory of private SGD indeed exhibit symmetric structures that favors convergence. Together, our results provide an explanation for why private SGD with gradient clipping remains effective in practice despite its potential clipping bias. Finally, we develop a new perturbation-based technique that can provably correct the clipping bias even for instances with highly asymmetric gradient distributions. These results are from Chen et al. [2020].

Chapter 2

Convergence analysis of a class of Adam-type algorithms

2.1 Introduction

First-order optimization has witnessed tremendous progress in the last decade, especially to solve machine learning problems [Bottou et al., 2018]. Almost every first-order method obeys the following generic form [Boyd and Vandenberghe, 2004], $\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha_t \mathbf{\Delta}_t$, where $\mathbf{x}_t$ denotes the solution updated at the tth iteration for $t = 1, 2, \ldots, T$, T is the number of iterations, $\mathbf{\Delta}_t$ is a certain (approximate) descent direction, and $\alpha_t > 0$ is some learning rate. The most well-known first-order algorithms are gradient descent (GD) for deterministic optimization [Nesterov, 2013, Cartis et al., 2010] and stochastic gradient descent (SGD) for stochastic optimization [Zinkevich, 2003, Ghadimi and Lan, 2013], where the former determines $\mathbf{\Delta}_t$ using the full (batch) gradient of an objective function, and the latter uses a simpler but more computationally-efficient stochastic (unbiased) gradient estimate.

Recent works have proposed a variety of accelerated versions of GD and SGD [Nesterov, 2013]. These achievements fall into three categories: a) *momentum methods* [Nesterov, 1983, Polyak, 1964, Ghadimi et al., 2015] which carefully design the descent direction $\mathbf{\Delta}_t$; b) *adaptive learning rate methods* [Becker et al., 1988, Duchi et al., 2011, Zeiler, 2012, Dauphin et al., 2015] which determine good learning rates α_t, and c) *adaptive gradient (momentum) methods* that enjoy dual advantages of a) and b). In particular, Adam

[Kingma and Ba, 2014], belonging to the third type of methods, has become extremely popular to solve deep learning problems, e.g., to train deep neural networks. Despite its superior performance in practice, theoretical investigation of Adam-like methods for *non-convex* optimization is still missing.

More recently, the work Reddi et al. [2018] pointed out the convergence issues of Adam even in the convex setting, and proposed AMSGrad, a corrected version of Adam. Although AMSGrad has made a positive step towards understanding the theoretical behavior of adaptive gradient methods, the convergence analysis of Reddi et al. [2018] was still very restrictive because it only works for convex problems, despite the fact that the most successful applications are for non-convex problems. Apparently, there still exists a large gap between theory and practice. To the best of our knowledge, the question that whether adaptive gradient methods such as Adam, AMSGrad, AdaGrad converge for non-convex problems is still open in theory before this research.

Related Work *Momentum methods* take into account the history of first-order information [Nesterov, 2013, 1983, Nemirovskii et al., 1983, Ghadimi and Lan, 2016, Polyak, 1964, Ghadimi et al., 2015, Ochs et al., 2015, Yang et al., 2016, Johnson and Zhang, 2013, Reddi et al., 2016a, Lei et al., 2017]. A well-known method, called Nesterov's accelerated gradient (NAG) originally designed for convex deterministic optimization [Nesterov, 2013, 1983, Nemirovskii et al., 1983], constructs the descent direction $\mathbf{\Delta}_t$ using the difference between the current iterate and the previous iterate. A recent work Ghadimi and Lan [2016] studied a generalization of NAG for non-convex stochastic programming. Similar in spirit to NAG, heavy-ball (HB) methods [Polyak, 1964, Ghadimi et al., 2015, Ochs et al., 2015, Yang et al., 2016] form the descent direction vector through a decaying sum of the previous gradient information. In addition to NAG and HB methods, stochastic variance reduced gradient (SVRG) methods integrate SGD with GD to acquire a hybrid descent direction of reduced variance [Johnson and Zhang, 2013, Reddi et al., 2016a, Lei et al., 2017]. Recently, certain accelerated version of perturbed gradient descent (PAGD) algorithm is also proposed in Jin et al. [2017], which shows the fastest convergence rate among all Hessian free algorithms.

Adaptive learning rate methods accelerate ordinary SGD by using knowledge of the past gradients or second-order information into the current learning rate α_t [Becker et al.,

1988, Duchi et al., 2011, Zeiler, 2012, Dauphin et al., 2015]. In [Becker et al., 1988], the diagonal elements of the Hessian matrix were used to penalize a constant learning rate. However, acquiring the second-order information is computationally prohibitive. More recently, an adaptive subgradient method (i.e., AdaGrad) penalized the current gradient by dividing the square root of averaging of the squared gradient coordinates in earlier iterations [Duchi et al., 2011]. Although AdaGrad works well when gradients are sparse, its convergence is only analyzed in the convex world. Other adaptive learning rate methods include Adadelta [Zeiler, 2012] and ESGD [Dauphin et al., 2015], which lacked theoretical investigation although some convergence improvement was shown in practice.

Adaptive gradient methods update the descent direction and the learning rate simultaneously using knowledge in the past, and thus enjoy dual advantages of momentum and adaptive learning rate methods. Algorithms of this family include RMSProp [Tieleman and Hinton, 2012], Nadam [Dozat, 2016], and Adam [Kingma and Ba, 2014]. Among these, Adam has become the most widely-used method to train deep neural networks (DNNs). Specifically, Adam adopts *exponential* moving averages (with decaying/forgetting factors) of the past gradients to update the descent direction. It also uses inverse of exponential moving average of squared past gradients to adjust the learning rate. The work Kingma and Ba [2014] showed Adam converges with at most $O(1/\sqrt{T})$ rate for convex problems. However, the recent work Reddi et al. [2018] pointed out the convergence issues of Adam even in the convex setting, and proposed a modified version of Adam (i.e., AMSGrad), which utilizes a non-increasing quadratic normalization and avoids the pitfalls of Adam. Although AMSGrad has made a significant progress toward understanding the theoretical behavior of adaptive gradient methods, the convergence analysis of Reddi et al. [2018] only works for convex problems.

After the non-convergence issue of Adam has been raised in [Reddi et al., 2018], there have been a few recent works on proposing new variants of Adam-type algorithms. In the convex setting, reference [Huang et al., 2018] proposed to stabilize the coordinate-wise weighting factor to ensure convergence. Chen and Gu [2018] developed an algorithm that changes the coordinate-wise weighting factor to achieve better generalization performance. Concurrent with this work, several works are trying to understand performance of Adam in non-convex optimization problems. Basu et al. [2018] provided convergence rate of

original Adam and RMSprop under full-batch (deterministic) setting, and Ward et al. [2018] proved convergence rate of a modified version of AdaGrad where coordinate-wise weighting is removed. Furthermore, the work Zhou et al. [2018] provided convergence results for AMSGrad that exhibit a tight dependency on problem dimension compared to Reddi et al. [2018]. The works Zou and Shen [2018] and Li and Orabona [2018] proved that both AdaGrad and its variant (AdaFom) converge to a stationary point with a high probability. The aforementioned works are independent of ours. In particular, our analysis is not only more comprehensive (it covers the analysis of a large family of algorithms in a single framework), but more importantly, it provides insights on how oscillation of stepsizes can affect the convergence rate.

Contributions Our work aims to build the theory to understand the behavior for a *generic* class of adaptive gradient methods for non-convex optimization. In particular, we provide mild sufficient conditions that guarantee the convergence for the Adam-type methods. We summarize our contribution as follows.

• **(Generality)** We consider a class of generalized Adam, referred to as the "Adam-type", and we show for the first time that under suitable conditions about the stepsizes and algorithm parameters, this class of algorithms all converge to first-order stationary solutions of the non-convex problem. This class includes the recently proposed AMSGrad [Reddi et al., 2018], AdaGrad [Duchi et al., 2011], and stochastic heavy-ball methods as well as two new algorithms explained below.

- **(AdaFom)** Adam adds momentum to both the first and the second moment estimate, but this leads to possible divergence [Reddi et al., 2018]. We show that the divergence issue can actually be fixed by a simple variant which adds momentum to only the first moment estimate while using the same second moment estimate as that of AdaGrad, which we call AdaFom (AdaGrad with First Order Moment).

- **(Constant Momemtum)** Our convergence analysis is applicable to the *constant* momentum parameter setting for AMSGrad and AdaFom. The divergence example of Adam given in [Reddi et al., 2018] is for constant momentum parameter, but the convergence analysis of AMSGrad in [Reddi et al., 2018] is for diminishing momentum parameter. This discrepancy leads to a question whether the convergence of

AMSGrad is due to the algorithm form or due to the momentum parameter choice – we show that the constant-momentum version of AMSGrad indeed converges, thus excluding the latter possibility.

• **(Practicality)** The sufficient conditions we derive are simple and easy to check in practice. They can be used to either certify the convergence of a given algorithm for a class of problem instances, or to track the progress and behavior of a particular realization of an algorithm.

• **(Tightness and Insight)** We show the conditions are essential and "tight", in the sense that violating them can make an algorithm diverge. Importantly, our conditions provide insights on how oscillation of a so-called "effective stepsize" (that we define later) can affect the convergence rate of the class of algorithms. We also provide interpretations of the convergence conditions to illustrate why under some circumstances, certain Adam-type algorithms can outperform SGD.

Notations We use $z = x/y$ to denote element-wise division if x and y are both vectors of size d; $x \odot y$ is element-wise product, x^2 is element-wise square if x is a vector, $\sqrt{x}$ is element-wise square root if x is a vector, $(x)_j$ denotes jth coordinate of x, $\|x\|$ is $\|x\|_2$ if not otherwise specified. We use $[N]$ to denote the set $\{1, \cdots, N\}$, and use $O(\cdot)$, $o(\cdot)$, $\Omega(\cdot)$, $\omega(\cdot)$ as standard asymptotic notations.

2.2 Preliminaries and Adam-Type Algorithms

Stochastic optimization is a popular framework for analyzing algorithms in machine learning due to the popularity of mini-batch gradient evaluation. We consider the following generic problem where we are minimizing a function f, expressed in the expectation form as follows

$$\min_{x \in \mathbb{R}^d} f(x) = \mathbb{E}_\xi[f(x;\xi)], \tag{2.1}$$

where ξ is a certain random variable representing randomly selected data sample or random noise.

In a generic first-order optimization algorithm, at a given time t we have access to an unbiased noisy gradient g_t of $f(x)$, evaluated at the current iterate x_t. The noisy gradient is assumed to be bounded and the noise on the gradient at different time t is assumed to be independent. An important assumption that we will make throughout this paper is that the function $f(x)$ is continuously differentiable and has Lipschitz continuous gradient, but could otherwise be a *non-convex* function. The non-convex assumption represents a major departure from the convexity that has been assumed in recent papers for analyzing Adam-type methods, such as [Kingma and Ba, 2014] and [Reddi et al., 2018].

Our work focuses on the generic form of exponentially weighted stochastic gradient descent method presented in Algorithm 1, for which we name as *generalized Adam* due to its resemblance to the original Adam algorithm and many of its variants.

Algorithm 1 Generalized Adam

S0. Initialize $m_0 = 0$ and x_1
for $t = 1, 2, \ldots, T$ **do**
  S1. $m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t$
  S2. $\hat{v}_t = h_t(g_1, g_2, ..., g_t)$
  S3. $x_{t+1} = x_t - \alpha_t m_t / \sqrt{\hat{v}_t}$
end for

In Algorithm 1, α_t is the step size at time t, $\beta_{1,t} > 0$ is a sequence of problem parameters, $m_t \in \mathbb{R}^d$ denotes some (exponentially weighted) gradient estimate, and $\hat{v}_t = h_t(g_1, g_2, ..., g_t) \in \mathbb{R}^d$ takes all the past gradients as input and returns a vector of dimension d, which is later used to inversely weight the gradient estimate m_t. And note that $m_t/\sqrt{\hat{v}_t} \in \mathbb{R}^d$ represents element-wise division. Throughout the paper, we will refer to the vector $\alpha_t/\sqrt{\hat{v}_t}$ as the *effective stepsize*.

We highlight that Generalized Adam includes many well-known algorithms as special cases. We summarize some popular variants of the generalized Adam algorithm in Table 2.1.

Table 2.1: Variants of generalized Adam

$\hat{v}_t$ \ $\beta_{1,t}$	$\beta_{1,t} = 0$	$\beta_{1,t} \leq \beta_{1,t-1}$ $\beta_{1,t} \xrightarrow[t\to\infty]{} b \geq 0$	$\beta_{1,t} = \beta_1$
$\hat{v}_t = 1$	SGD	N/A*	Heavy-ball method
$\hat{v}_t = \frac{1}{t}\sum_{i=1}^{t} g_i^2$	AdaGrad	AdaFom	AdaFom
$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$, $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$	AMSGrad	AMSGrad	AMSGrad
$\hat{v}_t = \beta_2 \hat{v}_{t-1} + (1-\beta_2) g_t^2$	RMSProp	N/A	Adam

* N/A stands for an informal algorithm that was not defined in literature.

We present some interesting findings for the algorithms presented in Table 2.1.

- Adam is often regarded as a "momentum version" of AdaGrad, but it is different from AdaFom which is also a momentum version of AdaGrad [1]. The difference lies in the form of $\hat{v}_t$. Intuitively, Adam adds momentum to both the first and second order moment estimate, while in AdaFom we only add momentum to the first moment estimate and use the same second moment estimate as AdaGrad. These two methods are related in the following way: if we let $\beta_2 = 1 - 1/t$ in the expression of $\hat{v}_t$ in Adam, we obtain AdaFom. We can view AdaFom as a variant of Adam with an increasing sequence of β_2, or view Adam as a variant of AdaFom with exponentially decaying weights of g_t^2. However, this small change has large impact on the convergence: we prove that AdaFom can always converge under standard assumptions (see Corollary 2) , while Adam is shown to possibly diverge [Reddi et al., 2018].

- The convergence of AMSGrad using a fast diminishing $\beta_{1,t}$ such that $\beta_{1,t} \leq \beta_{1,t-1}, \beta_{1,t} \xrightarrow[t\to\infty]{} b, b = 0$ in convex optimization was studied in [Reddi et al., 2018]. However, the convergence of the version with constant β_1 or strictly positive b and the version for non-convex setting are unexplored before our work. We notice that an independent work [Zhou et al., 2018] has also proved the convergence of AMSGrad with constant β_1.

[1] AdaGrad with first order momentum is also studied in [Zou and Shen, 2018] which appeared online after our first version

It is also worth mentioning that Algorithm 1 can be applied to solve the popular "finite-sum" problems whose objective is a sum of n individual cost functions. That is,

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^n f_i(x) := f(x), \tag{2.2}$$

where each $f_i : \mathbb{R}^d \to \mathbb{R}$ is a smooth and possibly non-convex function. If at each time instance the index i is chosen uniformly randomly, then Algorithm 1 still applies, with $g_t = \nabla f_i(x_t)$. It can also be extended to a mini-batch case with $\mathbf{g}_t = \frac{1}{b}\sum_{i\in\mathcal{I}_t} \nabla f_i(\mathbf{x}_t)$, where $\mathcal{I}_t$ denotes the minibatch of size b at time t. It is easy to show that g_t is an unbiased estimator for $\nabla f(x)$.

In the remainder of this paper, we will analyze Algorithm 1 and provide sufficient conditions under which the algorithm converges to first-order stationary solutions with sublinear rate. We will also discuss how our results can be applied to special cases of generalized Adam.

2.3 Convergence Analysis for Generalized Adam

The main technical challenge in analyzing the non-convex version of Adam-type algorithms is that the actually used update directions could no longer be unbiased estimates of the true gradients. Furthermore, an additional difficulty is introduced by the involved form of the adaptive learning rate. Therefore the biased gradients have to be carefully analyzed together with the use of the inverse of exponential moving average while adjusting the learning rate. The existing convex analysis [Reddi et al., 2018] does not apply to the non-convex scenario we study for at least two reasons: first, non-convex optimization requires a different convergence criterion, given by stationarity rather than the global optimality; second, we consider *constant* momentum controlling parameter.

In the following, we formalize the assumptions required in our convergence analysis.

Assumptions

A1.1: f is differentiable and has L-Lipschitz gradient, i.e. $\forall x, y$, $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$. It is also lower bounded, i.e. $f(x^*) > -\infty$ where x^* is an optimal solution.

A1.2: At time t, the algorithm can access a bounded noisy gradient and the true gradient is bounded, i.e. $\|\nabla f(x_t)\| \leq H, \quad \|g_t\| \leq H, \quad \forall t > 1$.

A1.3: The noisy gradient is unbiased and the noise is independent, i.e. $g_t =$

$\nabla f(x_t) + \zeta_t$, $E[\zeta_t] = 0$ and ζ_i is independent of ζ_j if $i \neq j$.

Reference [Reddi et al., 2018] uses a similar (but slightly different) assumption as A2, i.e., the bounded elements of the gradient $\|g_t\|_\infty \leq a$ for some finite a. The bounded norm of $\nabla f(x_t)$ in A2 is equivalent to Lipschitz continuity of f (when f is differentiable) which is a commonly used condition in convergence analysis. This assumption is often satisfied in practice, for example it holds for the finite sum problem (2.2) when each f_i has bounded gradient, and $g_t = \nabla f_i(x_t)$ where i is sampled randomly. A3 is also standard in stochastic optimization for analyzing convergence.

Our main result shows that if the coordinate-wise weighting term $\sqrt{\hat{v}_t}$ in Algorithm 1 is properly chosen, we can ensure the global convergence as well as the sublinear convergence rate of the algorithm (to a first-order stationary solution). First, we characterize how the effective stepsize parameters α_t and $\hat{v}_t$ affect convergence of Adam-type algorithms.

Theorem 1. *Suppose that Assumptions A1.1-A1.3 are satisfied, β_1 is chosen such that $\beta_1 \geq \beta_{1,t}$, $\beta_{1,t} \in [0,1)$ is non-increasing, and for some constant $G > 0$, $\left\|\alpha_t m_t/\sqrt{\hat{v}_t}\right\| \leq G$, $\forall\ t$. Then Generalized Adam yields*

$$
\begin{aligned}
&E\left[\sum_{t=1}^{T} \alpha_t \langle \nabla f(x_t), \nabla f(x_t)/\sqrt{\hat{v}_t}\rangle\right] \qquad (2.3)\\
\leq &E\left[C_1 \sum_{t=1}^{T} \left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2 + C_2 \sum_{t=2}^{T} \left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1 + C_3 \sum_{t=2}^{T-1} \left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right] + C_4
\end{aligned}
$$

where C_1, C_2, C_3 are constants independent of d and T, C_4 is a constant independent of T, the expectation is taken with respect to all the randomness corresponding to $\{g_t\}$.

Further, let $\gamma_t := \min_{j\in[d]} \min_{\{g_i\}_{i=1}^t} \alpha_t/(\sqrt{\hat{v}_t})_j$ denote the minimum possible value of effective stepsize at time t over all possible coordinate and past gradients $\{g_i\}_{i=1}^t$. Then the convergence rate of Generalized Adam is given by

$$
\min_{t\in[T]} E\left[\|\nabla f(x_t)\|^2\right] = O\left(\frac{s_1(T)}{s_2(T)}\right), \qquad (2.4)
$$

where $s_1(T)$ is defined through the upper bound of RHS of (2.3), namely, $O(s_1(T))$, and $\sum_{t=1}^{T} \gamma_t = \Omega(s_2(T))$.

Proof: See Section 2.6.1. **Q.E.D.**

In Theorem 1, $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq G$ is a mild condition. Roughly speaking, it implies that the change of x_t at each each iteration should be finite. As will be evident later,

with $\|g_t\| \leq H$, the condition $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq G$ is automatically satisfied for both AdaGrad and AMSGrad. Besides, instead of bounding the minimum norm of ∇f in (2.4), we can also apply a probabilistic output (e.g., select an output $\mathbf{x}_R$ with probability $p(R=t) = \frac{\gamma_t}{\sum_{t=1}^T \gamma_t}$) to bound $E[\|\nabla f(x_R)\|^2]$ [Ghadimi and Lan, 2013, Lei et al., 2017]. It is worth mentioning that a small number ϵ could be added to $\hat{v}_t$ for ensuring the numerical stability. In this case, our Theorem 1 still holds given the fact the resulting algorithm is still a special case of Algorithm 1. Accordingly, our convergence results for AMSGrad and AdaFom that will be derived later also hold as $\|\alpha_t m_t/(\sqrt{\hat{v}_t}+\epsilon)\| \leq \|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq G$ when ϵ is added to $\hat{v}_t$. We will provide a detailed explanation of Theorem 1 in Section 2.3.1.

Theorem 1 implies a sufficient condition that guarantees convergence of the Adam-type methods: $s_1(T)$ grows slower than $s_2(T)$. We will show in Section 2.3.2 that the rate $s_1(T)$ can be dominated by different terms in different cases, i.e. the non-constant quantities Term A and B below

$$E\Big[\underbrace{\sum_{t=1}^{T}\Big\|\alpha_t g_t/\sqrt{\hat{v}_t}\Big\|^2}_{\text{Term A}} + \underbrace{\sum_{t=2}^{T}\Big\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\Big\|_1}_{\text{Term B}} + \sum_{t=2}^{T-1}\Big\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\Big\|^2\Big] = O(s_1(T)), \qquad (2.5)$$

where the growth of third term at LHS of (2.5) can be directly related to growth of Term B via the relationship between ℓ_1 and ℓ_2 norm or upper boundedness of $(\alpha_t/\sqrt{\hat{v}_t})_j$.

2.3.1 Explanation of convergence conditions

From (2.4) in Theorem 1, it is evident that $s_1(T) = o(s_2(T))$ can ensure proper convergence of the algorithm. This requirement has some important implications, which we discuss below.

• **(The Bounds for $s_1(T)$ and $s_2(T)$)** First, the requirement that $s_1(T) = o(s_2(T))$ implies that $E[\sum_{t=1}^T \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2] = o(\sum_{t=1}^T \gamma_t)$. This is a common condition generalized from SGD. Term A in (2.5) is a generalization of the term $\sum_{t=1}^T \alpha\|a_t g_t\|^2$ for SGD (where $\{\alpha_t\}$ is the stepsize sequence for SGD), and it quantifies possible increase in the objective function brought by higher order curvature. The term $\sum_{t=1}^T \gamma_t$ is the lower bound on the summation of effective stepsizes, which reduces to $\sum_{t=1}^T \alpha_t$ when Generalized Adam is simplified to SGD.

• **(Oscillation of Effective Stepsizes)** Term B in (2.5) characterizes the *oscillation of effective stepsizes* $\alpha_t/\sqrt{\hat{v}_t}$. In our analysis such an oscillation term upper bounds the expected possible ascent in objective induced by skewed update direction $g_t/\sqrt{\hat{v}_t}$ ("skewed" in the sense that $E[g_t/\sqrt{\hat{v}_t}]$ is not parallel with $\nabla f(x_t)$), therefore it cannot be too large. Bounding this term is critical, and to demonstrate this fact, in Section 2.3.2 we show that large oscillation can result in non-convergence of Adam for even simple unconstrained non-convex problems.

• **(Advantage of Adaptive Gradient).** One possible benefit of adaptive gradient methods can be seen from Term A. When this term dominates the convergence speed in Theorem 1, it is possible that proper design of $\hat{v}_t$ can help reduce this quantity compared with SGD (An example is provided in Section 2.4.1 to further illustrate this fact.) in certain cases. Intuitively, adaptive gradient methods like AMSGrad can provide a flexible choice of stepsizes, since $\hat{v}_t$ can have a *normalization effect* to reduce oscillation and overshoot introduced by large stepsizes. At the same time, flexibility of stepsizes makes the hyperparameter tuning of an algorithm easier in practice.

2.3.2 Tightness of the rate bound (2.4)

In the next, we show our bound (2.4) is tight in the sense that there exist problems satisfying Assumption 1 such that certain algorithms belonging to the class of Algorithm 1 can diverge due to the high growth rate of Term A or Term B.

Non-convergence of SGD and ADAM due to effect of Term A

We demonstrate the importance of Term A in this subsection. Consider a simple one-dimensional optimization problem $\min_x f(x)$, with $f(x) = 100x^2$ if $|x| <= b$, and $f(x) = 200b|x| - 100b^2$ if $|x| > b$, where $b = 10$. In Figure 2.1, we show the growth rate of different terms given in Theorem 1, where $\alpha_0 \triangleq 0$, $\alpha_t = 0.01$ for $t \geq 1$, and $\beta_{1,t} = 0, \beta_{2,t} = 0.9$ for both Adam and AMSGrad. We observe that both SGD and Adam are not converging to a stationary solution ($x = 0$), which is because $\sum_{t=1}^{T} \left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2$ grows with the same rate as accumulation of effective stepsizes as shown in the figure. Actually, SGD only converges when $\alpha_t < 0.01$ and our theory provides an perspective of why SGD diverges when $\alpha_t \geq 0.01$. In the example, Adam is also not converging to 0 due to Term A. From our observation, Adam oscillates for any constant stepsize within

$[10^{-4}, 0.1]$ for this problem and Term A always ends up growing as fast as accumulation of effective stepsizes, which implies Adam only converges with diminishing stepsizes even in non-stochastic optimization. In contrast to SGD and Adam, AMSGrad converges in this case since both Term A and Term B grow slower than accumulation of effective stepsizes. For AMSGrad and Adam, $\hat{v}_t$ has a strong normalization effect and it allows the algorithm to use a larger range of α_t. The practical benefit of this flexible choice of stepsizes is easier hyperparameter tuning, which is consistent with the impression of practitioners about the original Adam. We present more experimental results in Section 2.4.1 accompanied with more detailed discussions.

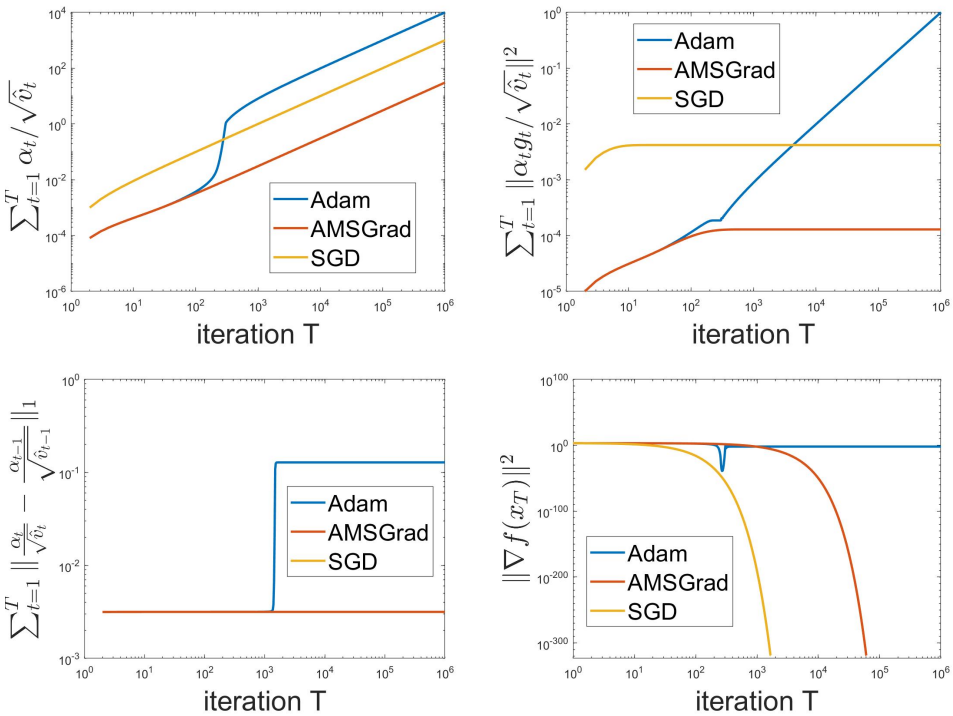


Figure 2.1: A toy example to illustrate effect of Term A on Adam, AMSGrad, and SGD.

Non-convergence of Adam due to effect of Term B

Next, we use an example to demonstrate the importance of the Term B for the convergence of Adam-type algorithms.

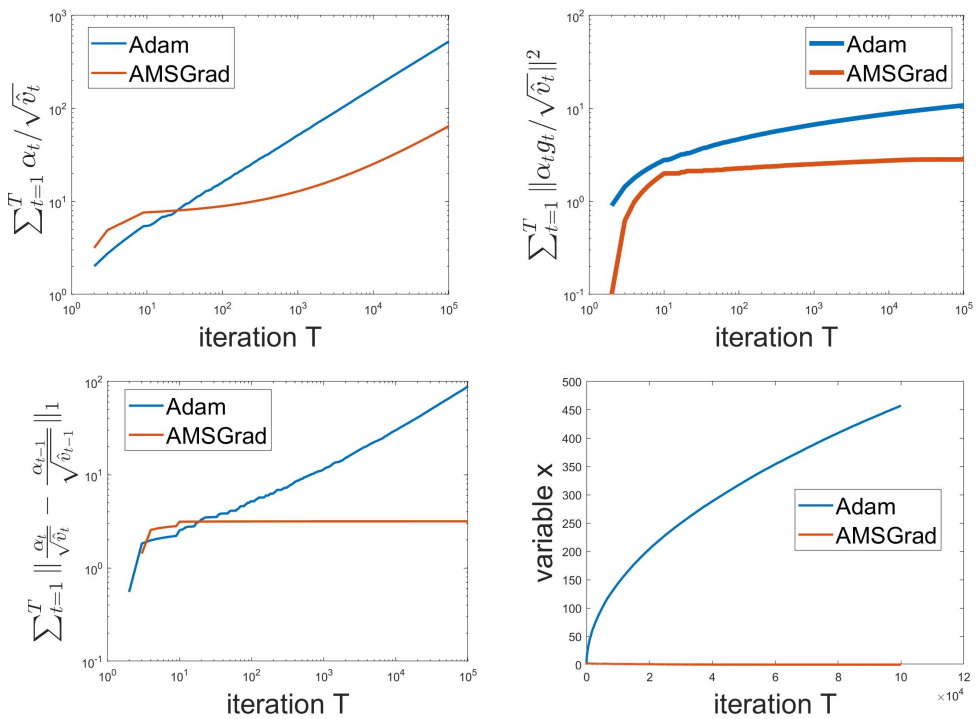


Figure 2.2: A toy example to illustrate effect of Term B on Adam and AMSGrad.

Consider optimization problem $\min_x f(x) = \sum_{i=1}^{11} f_i(x)$ where

$$f_i(x) = \begin{cases} \mathbb{I}[i=1]5.5x^2 + \mathbb{I}[i \neq 1](-0.5x^2), & \text{if } |x| \leq 1 \\ \mathbb{I}[i=1](11|x| - 5.5) + \mathbb{I}[i \neq 1](-|x| + 0.5), & \text{if } |x| > 1 \end{cases} \tag{2.6}$$

and $\mathbb{I}[1=1] = 1, \mathbb{I}[1 \neq 1] = 0$. It is easy to verify that the only point with $\nabla f(x) = 0$ is $x = 0$. The problem satisfies the assumptions in Theorem 1 as the stochastic gradient $g_t = \nabla f_i(x_t)$ is sampled uniformly for $i \in [11]$. We now use the AMSGrad and Adam to optimize x, and the results are given in Figure 2.2, where we set $\alpha_t = 1$, $\beta_{1,t} = 0$, and $\beta_{2,t} = 0.1$. We observe that $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$ in Term B grows with the same rate as $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$ for Adam, where we recall that $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$ is an upper bound of $\sum_{t=1}^{T} \gamma_t$ in Theorem 1. As a result, we obtain $O(s_1(T)/s_2(T)) \neq o(1)$ in (2.4), implying the non-convergence of Adam. Our theoretical analysis matches the empirical results in Figure 2.2. In contrast, AMSGrad converges in Figure 2.2 because of its

smaller oscillation in effective stepsizes, associated with Term B. We finally remark that the importance of the quantity $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$ is also noticed by [Huang et al., 2018]. However, they did not analyze its effect on convergence, and their theory is only for convex optimization.

2.3.3 Convergence of AMSGrad and AdaFom

Theorem 1 provides a general approach for the design of the weighting sequence $\{\hat{v}_t\}$ and the convergence analysis of Adam-type algorithms. For example, SGD specified by Table 2.1 with stepsizes $\alpha_t = 1/\sqrt{t}$ yields $O(\log T/\sqrt{T})$ convergence speed by Theorem 1. Moreover, the explanation on the non-convergence of Adam in [Reddi et al., 2018] is consistent with our analysis in Section 2.3.2. That is, Term B in (2.5) can grow as fast as $s_2(T)$ so that $s_1(T)/s_2(T)$ becomes a constant. Further, we notice that Term A in (2.5) can also make Adam diverge which is unnoticed before. Aside from checking convergence of an algorithm, Theorem 1 can also provide convergence rates of AdaGrad and AMSGrad, which will be given as corollaries later.

Algorithm 2 AMSGrad

(S0). Define $m_0 = 0$, $v_0 = 0$, $\hat{v}_0 = 0$;
for $t = 1, 2, \ldots, T$ **do**
  (S1). $m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t$
  (S2). $v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$
  (S3). $\hat{v}_t = \max\{\hat{v}_{t-1}, v_t\}$
  (S4). $x_{t+1} = x_t - \alpha_t m_t / \sqrt{\hat{v}_t}$
end for

Algorithm 3 AdaFom

(S0). Define $m_0 = 0$, $\hat{v}_0 = 0$;
for $t = 1, 2, \ldots, T$ **do**
  (S1). $m_t = \beta_{1,t} m_{t-1} + (1 - \beta_{1,t}) g_t$
  (S2). $\hat{v}_t = (1 - 1/t)\hat{v}_{t-1} + (1/t) g_t^2$
  (S3). $x_{t+1} = x_t - \alpha_t m_t / \sqrt{\hat{v}_t}$
end for

Our proposed convergence rate of AMSGrad matches the result in [Reddi et al., 2018] for stochastic convex optimization. However, the analysis of AMSGrad in [Reddi et al., 2018] is constrained to diminishing momentum controlling parameter $\beta_{1,t}$. Instead, our analysis is applicable to the more popular *constant* momentum parameter, leading to a more general *non-increasing* parameter setting.

In Corollary 1 and Corollary 2, we derive the convergence rates of AMSGrad and AdaFom, respectively. Note that AdaFom is more general than AdaGrad since when $\beta_{1,t} = 0$, AdaFom becomes AdaGrad.

Corollary 1. *Assume $\exists\, c > 0$ such that $|(g_1)_i| \geq c, \forall i \in [d]$, for AMSGrad (pseudo code in Section 2.6.2) with $\beta_{1,t} \leq \beta_1 \in [0, 1)$ and $\beta_{1,t}$ is non-increasing, $\alpha_t = 1/\sqrt{t}$, we have*

for any T,

$$\min_{t\in[T]} E\left[\|f(x_t)\|^2\right] \leq \frac{1}{\sqrt{T}}\left(Q_1 + Q_2 \log T\right) \tag{2.7}$$

where Q_1 and Q_2 are two constants independent of T.

Proof: See Section 2.6.2. **Q.E.D.**

Corollary 2. *Assume $\exists\, c > 0$ such that $|(g_1)_i| \geq c, \forall i \in [d]$, for AdaFom (Algorithm 4 in Section 2.6.3) with $\beta_{1,t} \leq \beta_1 \in [0,1)$ and $\beta_{1,t}$ is non-increasing, $\alpha_t = 1/\sqrt{t}$, we have for any T,*

$$\min_{t\in[T]} E\left[\|f(x_t)\|^2\right] \leq \frac{1}{\sqrt{T}}(Q_1' + Q_2' \log T) \tag{2.8}$$

where Q_1' and Q_2' are two constants independent of T.

Proof: See Section 2.6.3. **Q.E.D.**

The assumption $|(g_1)_i| \geq c, \forall i$ is a mild assumption and it is used to ensure $\hat{v}_1 \geq r$ for some constant r. It is also usually needed in practice for numerical stability (for AMSGrad and AdaGrad, if $(g_1)_i = 0$ for some i, division by 0 error may happen at the first iteration). In some implementations, to avoid numerical instability, the update rule of algorithms like Adam, AMSGrad, and AdaGrad take the form of $x_{t+1} = x_t - \alpha_t m_t/(\sqrt{\hat{v}_t} + \epsilon)$ with ϵ being a positive number. These modified algorithms still fall into the framework of Algorithm 1 since ϵ can be incorporated into the definition of $\hat{v}_t$. Meanwhile, our convergence proof for Corollary 1 and Corollary 2 can go through without assuming $|(g_1)_i| \geq c, \forall i$ because $\sqrt{\hat{v}_t} \geq \epsilon$. In addition, ϵ can affect the worst case convergence rate by a constant factor in the analysis.

We remark that the derived convergence rate of AMSGrad and AdaFom involves an additional $\log T$ factor compared to the fastest rate of first order methods ($1/\sqrt{T}$). However, such a slowdown can be mitigated by choosing an appropriate stepsize. To be specific, the $\log T$ factor for AMSGrad would be eliminated when we adopt a constant rather than diminishing stepsize, e.g., $\alpha_t = 1/\sqrt{T}$. It is also worth mentioning that our theoretical analysis focuses on the convergence rate of adaptive methods in the worst case for nonconvex optimization. Thus, a sharper convergence analysis that can quantify the benefits of adaptive methods still remains an open question in theory.

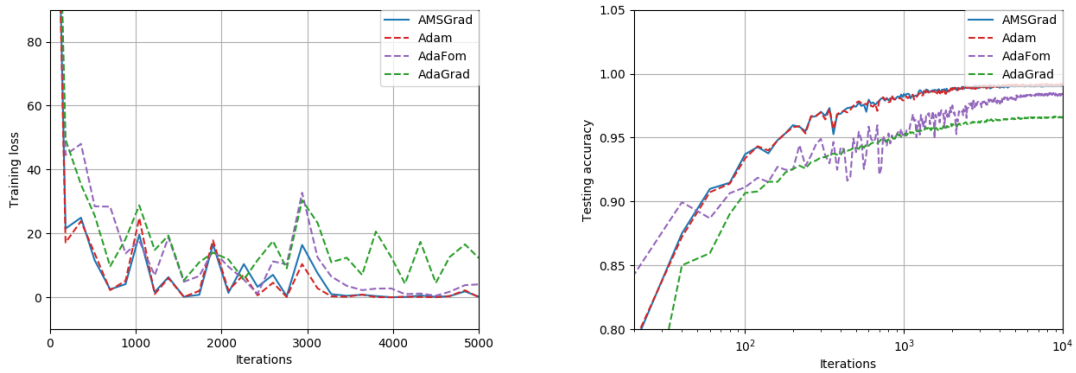
2.4 Empirical performance of Adam-type algorithms

In this section, we compare the empirical performance of Adam-type algorithms, including AMSGrad, Adam, AdaFom and AdaGrad, on training a convolutional neural network (CNN) on MNIST and CIFARNET on CIFAR-10.

In the experiment on MNIST, we consider a convolutional neural network (CNN), which includes 3 convolutional layers and 2 fully-connected layers. In convolutional layers, we adopt filters of sizes $6 \times 6 \times 1$ (with stride 1), $5 \times 5 \times 6$ (with stride 2), and $6 \times 6 \times 12$ (with stride 2), respectively. In both AMSGrad[2] and Adam, we set $\beta_1 = 0.9$ and $\beta_2 = 0.99$. In AdaFom, we set $\beta_1 = 0.9$. We choose 50 as the mini-batch size and the stepsize is choose to be $\alpha_t = 0.0001 + 0.003e^{-t/2000}$.

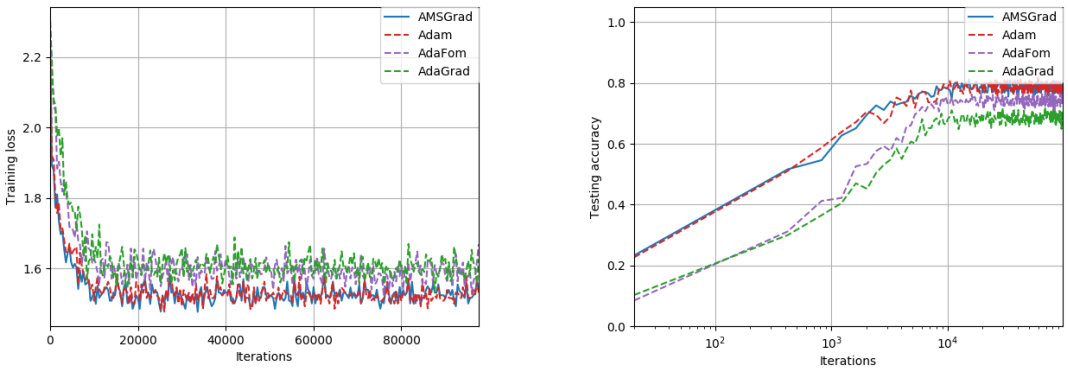
The architecture of the CIFARNET that we are using is as below. The model starts with two convolutional layers with 32 and 64 kernels of size 3 x 3, followed by 2 x 2 max pooling and dropout with keep probability 0.25. The next layers are two convolutional layers with 128 kernels of size 3 x 3 and 2 x 2, respectively. Each of the two convolutional layers is followed by a 2 x 2 max pooling layer. The last layer is a fully connected layer with 1500 nodes. Dropout with keep probability 0.25 is added between the fully connected layer and the convolutional layer. All convolutional layers use ReLU activation and stride 1. The learning rate α_t of Adam and AMSGrad starts with 0.001 and decrease 10 times every 20 epochs. The learning rate of AdaGrad and AdaFom starts with 0.05 and decreases to 0.001 after 20 epochs and to 0.0001 after 40 epochs. These learning rates are tuned so that each algorithm has its best performance.

[2]We customized our algorithms based on the open source code `https://github.com/taki0112/AMSGrad-Tensorflow`.



(a) Training loss versus iterations (b) Testing accuracy versus iterations

Figure 2.3: Comparison of AMSGrad, Adam, AdaFom and AdaGrad under MNIST in training loss and testing accuracy.



(a) Training loss versus iterations (b) Testing accuracy versus iterations

Figure 2.4: Comparison of AMSGrad, Adam, AdaFom and AdaGrad under CIFAR in training loss and testing accuracy.

In Figure 2.3, we present the training loss and the classification accuracy of Adam-type algorithms versus the number of iterations for MNIST. As we can see, AMSGrad performs quite similarly to Adam which confirms the result in [Reddi et al., 2018]. The performance of AdaGrad is worse than other algorithms, because of the lack of momentum and/or the significantly different choice of $\hat{v}_t$. We also observe that the performance of AdaFom lies between AMSGrad/Adam and AdaGrad. This is not surprising, since AdaFom can

be regarded as a momentum version of AdaGrad but uses a simpler adaptive learning rate (independent on β_2) compared to AMSGrad/Adam. In Figure 2.4, we consider to train a larger network (CIFARNET) on CIFAR-10. As we can see, Adam and AMSGrad perform similarly and yield the best accuracy. AdaFom outperforms AdaGrad in both training and testing, which agrees with the results obtained in the MNIST experiment.

2.4.1 Advantages and Disadvantages of Adaptive gradient method

In this section, we provide some additional experiments to demonstrate how specific Adam-type algorithms can perform better than SGD and how SGD can out perform Adam-type algorithms in different situations.

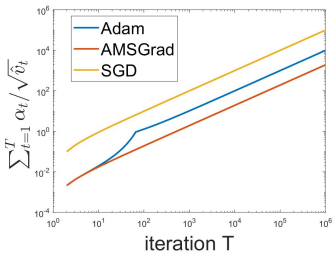
One possible benefit of adaptive gradient methods is the "sparse noise reduction" effect pointed out in Bernstein et al. [2018]. Below we illustrate another possible practical advantage of adaptive gradient methods when applied to solve *non-convex problems*, which we refer to as *flexibility of stepsizes*.

To highlight ideas, let us take AMSGrad as an example, and compare it with SGD. First, in non-convex problems there can be multiple valleys with different curvatures. When using fixed stepsizes (or even a slowly diminishing stepsize), SGD can only converge to local optima in valleys with small curvature while AMSGrad and some other adaptive gradient algorithms can potentially converge to optima in valleys with relative high curvature (this may not be beneficial if one don't want to converge to a sharp local minimum). Second, the flexible choice of stepsizes implies less hyperparameter tuning and this coincides with the popular impression about original Adam.

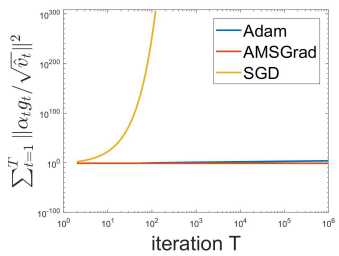
We empirically demonstrate the flexible stepsizes property of AMSGrad using a deterministic quadratic problem. Consider a toy optimization problem $\min_x f(x), f(x) = 100x^2$, the gradient is given by $200x$. For SGD (which reduces to gradient descent in this case) to converge, we must have $\alpha_t < 0.01$; for AMSGrad, $\hat{v}_t$ has a strong normalization effect and it allows the algorithm to use larger α_t's. We show the growth rate of different terms given in Theorem 1 for different stepsizes in Figure 2.5 to Figure 2.8 (where we choose $\beta_{1,t} = 0, \beta_{2,t} = 0.9$ for both Adam and AMSGrad). In Figure 2.5, $\alpha_t = 0.1$ and SGD diverges due to large α_t, AMSGrad converges in this case, Adam is oscillating between two non-zero points. In Figure 2.6, stepsizes α_t is set to 0.01, SGD and Adam are oscillating, AMSGrad converges to 0. For Figure 2.7, SGD converges to 0 and AMSGrad

is converging slower than SGD due to its smaller effective stepsizes, Adam is oscillating. One may wonder how diminishing stepsizes affects performance of the algorithms, this is shown in Figure 2.8 where $\alpha_t = 0.1/\sqrt{t}$, we can see SGD is diverging until stepsizes is small, AMSGrad is converging all the time, Adam appears to get stuck but it is actually converging very slowly due to diminishing stepsizes. This example shows AMSGrad can converge with a larger range of stepsizes compared with SGD.

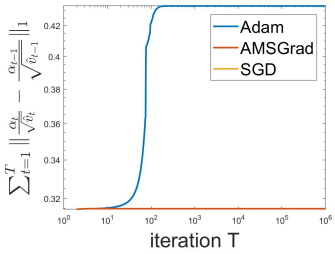
From the figures, we can see that the term $\sum_{t=1}^{T} \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2$ is the key quantity that limits the convergence speed of algorithms in this case. In Figure 2.5, Figure 2.6, and early stage of Figure 2.8, the quantity is obviously a good sign of convergence speed. In Figure 2.7, since the difference of quantity between AMSGrad and SGD is compensated by the larger effective stepsizes of SGD and some problem independent constant, SGD converges faster. In fact, Figure 2.7 provides a case where AMSGrad does not perform well. Note that the normalization factor $\sqrt{\hat{v}_t}$ can be understood as imitating the largest Lipschitz constant along the way of optimization, so generally speaking dividing by this number makes the algorithm converge easier. However when the Lipschitz constant becomes smaller locally around a local optimal point, the stepsizes choice of AMSGrad dictates that $\sqrt{\hat{v}_t}$ does not change, resulting a small effective stepsizes. This could be mitigated by AdaGrad and its momentum variants which allows $\hat{v}_t$ to decrease when g_t keeps decreasing.



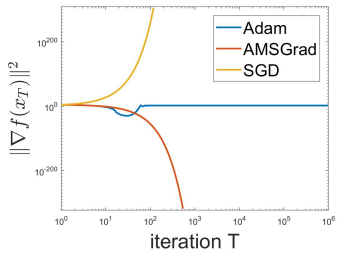
(a) $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$



(b) $\sum_{t=1}^{T} \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2$

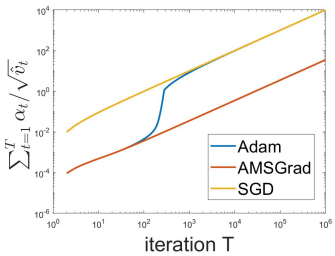


(c) $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$

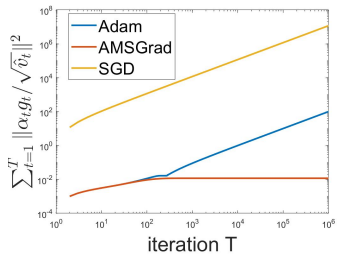


(d) $\|\nabla f(x_T)\|^2$

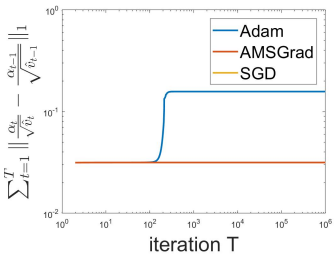
Figure 2.5: Comparison of algorithms with $\alpha_t = 0.1$, we defined $\alpha_0 = 0$



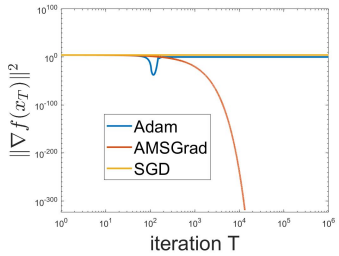
(a) $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$



(b) $\sum_{t=1}^{T} \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2$

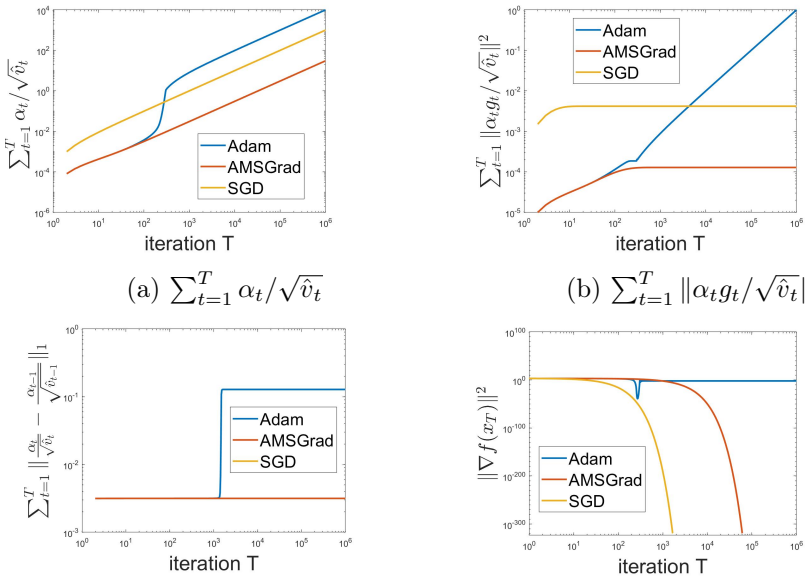


(c) $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$



(d) $\|\nabla f(x_T)\|^2$

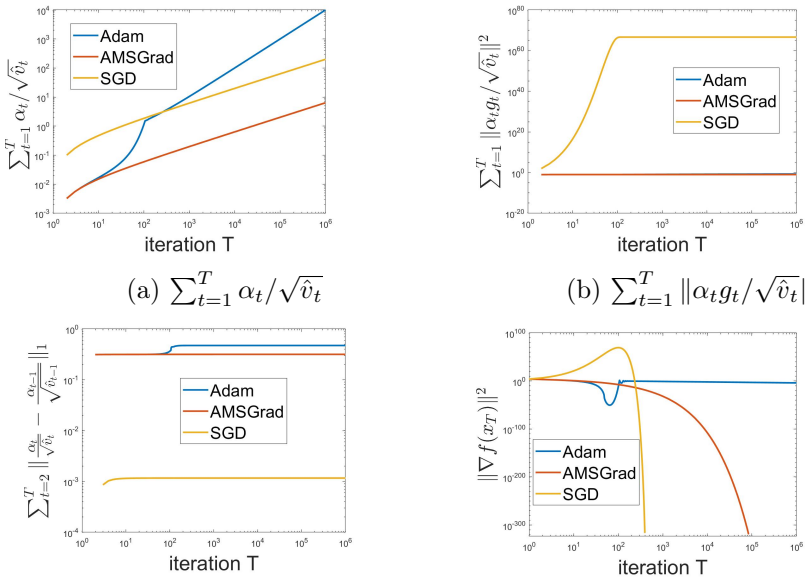
Figure 2.6: Comparison of algorithms with $\alpha_t = 0.01$, we defined $\alpha_0 = 0$



(a) $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$ (b) $\sum_{t=1}^{T} \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2$

(c) $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$ (d) $\|\nabla f(x_T)\|^2$

Figure 2.7: Comparison of algorithms with $\alpha_t = 0.001$, we defined $\alpha_0 = 0$



(a) $\sum_{t=1}^{T} \alpha_t/\sqrt{\hat{v}_t}$ (b) $\sum_{t=1}^{T} \|\alpha_t g_t/\sqrt{\hat{v}_t}\|^2$

(c) $\sum_{t=1}^{T} \|\alpha_t/\sqrt{\hat{v}_t} - \alpha_{t-1}/\sqrt{\hat{v}_{t-1}}\|_1$ (d) $\|\nabla f(x_T)\|^2$

Figure 2.8: Comparison of algorithms with $\alpha_t = 0.1/\sqrt{t}$, we defined $\alpha_0 = 0$

2.5 Discussion

We provided some mild conditions to ensure convergence of a class of Adam-type algorithms, which includes Adam, AMSGrad, AdaGrad, AdaFom, SGD, SGD with momentum as special cases. Apart from providing general convergence guarantees for algorithms, our conditions can also be checked in practice to monitor empirical convergence. We also provide insights on how oscillation of effective stepsizes can affect convergence rate for the class of algorithms which could be beneficial for the design of future algorithms. This chapter focuses on unconstrained non-convex optimization problems, and we well extend the analysis in the next chapter to constrained non-convex optimization problems when designing efficient zeroth order adaptive gradient methods.

2.6 Delayed proofs

2.6.1 Convergence proof for Generalized Adam

In this section, we present the convergence proof of Algorithm 1. We will first give several lemmas prior to proving Theorem 1.

Proof of Auxiliary Lemmas

Lemma 1. *Let $x_0 \triangleq x_1$ in Generalized Adam, consider the sequence*

$$z_t = x_t + \frac{\beta_{1,t}}{1-\beta_{1,t}}(x_t - x_{t-1}),\ \forall t \geq 1. \tag{2.9}$$

Then the following holds true

$$\begin{aligned} z_{t+1} - z_t = &-\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t/\sqrt{\hat{v}_t} \\ &- \frac{\beta_{1,t}}{1-\beta_{1,t}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t g_t/\sqrt{\hat{v}_t}, \quad \forall\ t > 1 \end{aligned}$$

and

$$z_2 - z_1 = -\left(\frac{\beta_{1,2}}{1-\beta_{1,2}} - \frac{\beta_{1,1}}{1-\beta_{1,1}}\right)\alpha_1 m_1/\sqrt{\hat{v}_1} - \alpha_1 g_1/\sqrt{\hat{v}_1}.$$

Proof. [Proof of Lemma 1] By the update rules S1-S3 in Generalized Adam, we have when $t > 1$,

$$\begin{aligned} x_{t+1} - x_t &= -\alpha_t m_t/\sqrt{\hat{v}_t} \\ &\overset{\text{S1}}{=} -\alpha_t(\beta_{1,t} m_{t-1} + (1-\beta_{1,t})g_t)/\sqrt{\hat{v}_t} \\ &\overset{\text{S3}}{=} \beta_{1,t}\frac{\alpha_t}{\alpha_{t-1}}\frac{\sqrt{\hat{v}_{t-1}}}{\sqrt{\hat{v}_t}} \odot (x_t - x_{t-1}) - \alpha_t(1-\beta_{1,t})g_t/\sqrt{\hat{v}_t} \\ &= \beta_{1,t}(x_t - x_{t-1}) + \beta_{1,t}\left(\frac{\alpha_t}{\alpha_{t-1}}\frac{\sqrt{\hat{v}_{t-1}}}{\sqrt{\hat{v}_t}} - 1\right) \odot (x_t - x_{t-1}) - \alpha_t(1-\beta_{1,t})g_t/\sqrt{\hat{v}_t} \\ &\overset{\text{S3}}{=} \beta_{1,t}(x_t - x_{t-1}) - \beta_{1,t}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t(1-\beta_{1,t})g_t/\sqrt{\hat{v}_t}. \end{aligned} \tag{2.10}$$

Since $x_{t+1} - x_t = (1-\beta_{1,t})x_{t+1} + \beta_{1,t}(x_{t+1} - x_t) - (1-\beta_{1,t})x_t$, based on (2.10) we have

$$\begin{aligned}
&(1-\beta_{1,t})x_{t+1} + \beta_{1,t}(x_{t+1} - x_t) \\
=&(1-\beta_{1,t})x_t + \beta_{1,t}(x_t - x_{t-1}) - \beta_{1,t}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t(1-\beta_{1,t})g_t/\sqrt{\hat{v}_t}.
\end{aligned}$$

Divide both sides by $1-\beta_{1,t}$, we have

$$\begin{aligned}
&x_{t+1} + \frac{\beta_{1,t}}{1-\beta_{1,t}}(x_{t+1} - x_t) \\
=&x_t + \frac{\beta_{1,t}}{1-\beta_{1,t}}(x_t - x_{t-1}) - \frac{\beta_{1,t}}{1-\beta_{1,t}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t g_t/\sqrt{\hat{v}_t}. \qquad (2.11)
\end{aligned}$$

Define the sequence

$$z_t = x_t + \frac{\beta_{1,t}}{1-\beta_{1,t}}(x_t - x_{t-1}).$$

Then (2.11) can be written as

$$\begin{aligned}
z_{t+1} = z_t &+ \left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)(x_{t+1} - x_t) \\
&- \frac{\beta_{1,t}}{1-\beta_{1,t}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t g_t/\sqrt{\hat{v}_t} \\
= z_t &- \left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t\sqrt{\hat{v}_t} \\
&- \frac{\beta_{1,t}}{1-\beta_{1,t}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right) \odot m_{t-1} - \alpha_t g_t/\sqrt{\hat{v}_t}, \quad \forall t > 1,
\end{aligned}$$

where the second equality is due to $x_{t+1} - x_t = -\alpha_t m_t/\sqrt{\hat{v}_t}$.

For $t = 1$, we have $z_1 = x_1$ (due to $x_1 = x_0$), and

$$\begin{aligned}
z_2 - z_1 =& x_2 + \frac{\beta_{1,2}}{1-\beta_{1,2}}(x_2 - x_1) - x_1 \\
=& x_2 + \left(\frac{\beta_{1,2}}{1-\beta_{1,2}} - \frac{\beta_{1,1}}{1-\beta_{1,1}}\right)(x_2 - x_1) + \frac{\beta_{1,1}}{1-\beta_{1,1}}(x_2 - x_1) - x_1 \\
=& \left(\frac{\beta_{1,2}}{1-\beta_{1,2}} - \frac{\beta_{1,1}}{1-\beta_{1,1}}\right)(-\alpha_1 m_1/\sqrt{\hat{v}_1}) + \left(\frac{\beta_{1,1}}{1-\beta_{1,1}} + 1\right)(x_2 - x_1) \\
=& \left(\frac{\beta_{1,2}}{1-\beta_{1,2}} - \frac{\beta_{1,1}}{1-\beta_{1,1}}\right)(-\alpha_1 m_1/\sqrt{\hat{v}_1}) + \frac{1}{1-\beta_{1,1}}(-\alpha_1(1-\beta_{1,1})g_1/\sqrt{\hat{v}_1}) \\
=& -\left(\frac{\beta_{1,2}}{1-\beta_{1,2}} - \frac{\beta_{1,1}}{1-\beta_{1,1}}\right)(a_1 m_1/\sqrt{\hat{v}_1}) - \alpha_1 g_1/\sqrt{\hat{v}_1},
\end{aligned}$$

where the forth equality holds due to (S1) and (S3) of Generalized Adam.

The proof is now complete. **Q.E.D.**

Without loss of generality, we initialize Generalized Adam as below to simplify our analysis in what follows,

$$\left(\frac{\alpha_1}{\sqrt{\hat{v}_1}} - \frac{\alpha_0}{\sqrt{\hat{v}_0}}\right) \odot m_0 = 0. \tag{2.12}$$

Lemma 2. *Suppose that the conditions in Theorem 1 hold, then*

$$E\left[f(z_{t+1}) - f(z_1)\right] \leq \sum_{i=1}^{6} T_i, \tag{2.13}$$

where

$$T_1 = -E\left[\sum_{i=1}^{t}\langle\nabla f(z_i), \frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)\odot m_{i-1}\rangle\right], \tag{2.14}$$

$$T_2 = -E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i), g_i/\sqrt{\hat{v}_i}\rangle\right], \tag{2.15}$$

$$T_3 = -E\left[\sum_{i=1}^{t}\langle\nabla f(z_i), \left(\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right)\alpha_i m_i/\sqrt{\hat{v}_i}\rangle\right], \tag{2.16}$$

$$T_4 = E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t/\sqrt{v_t}\right\|^2\right], \tag{2.17}$$

$$T_5 = E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)\odot m_{i-1}\right\|^2\right], \tag{2.18}$$

$$T_6 = E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\alpha_i g_i/\sqrt{\hat{v}_i}\right\|^2\right]. \tag{2.19}$$

Proof. [Proof of Lemma 2] By the Lipschitz smoothness of ∇f, we obtain

$$f(z_{t+1}) \leq f(z_t) + \langle\nabla f(z_t), d_t\rangle + \frac{L}{2}\left\|d_t\right\|^2, \tag{2.20}$$

where $d_t = z_{t+1} - z_t$, and Lemma 1 together with (2.12) yield

$$\begin{aligned} d_t = &-\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t/\sqrt{\hat{v}_t} \\ &- \frac{\beta_1}{1-\beta_1}\left(\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right)\odot m_{t-1} - \alpha_t g_t/\sqrt{\hat{v}_t}, \quad \forall t \geq 1. \end{aligned} \tag{2.21}$$

Based on (2.20) and (2.21), we then have

$$\begin{aligned}
E[f(z_{t+1}) - f(z_1)] =& E\left[\sum_{i=1}^{t} f(z_{i+1}) - f(z_i)\right] \\
\leq& E\left[\sum_{i=1}^{t} \langle \nabla f(z_i), d_i\rangle + \frac{L}{2}\|d_i\|^2\right] \\
=& -E\left[\sum_{i=1}^{t} \langle \nabla f(z_i), \frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right) \odot m_{i-1}\rangle\right] \\
& - E\left[\sum_{i=1}^{t} \alpha_i \langle \nabla f(z_i), g_i/\sqrt{\hat{v}_i}\rangle\right] \\
& - E\left[\sum_{i=1}^{t} \langle \nabla f(z_i), \left(\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right) \alpha_i m_i/\sqrt{\hat{v}_i}\rangle\right] \\
& + E\left[\sum_{i=1}^{t} \frac{L}{2}\|d_i\|^2\right] = T_1 + T_2 + T_3 + +E\left[\sum_{i=1}^{t} \frac{L}{2}\|d_i\|^2\right], \quad (2.22)
\end{aligned}$$

where $\{T_i\}$ have been defined in (2.14)-(2.19). Further, using inequality $\|a+b+c\|^2 \leq 3\|a\|^2 + 3\|b\|^2 + 3\|c\|^2$ and (2.22), we have

$$E\left[\sum_{i=1}^{t} \|d_i\|^2\right] \leq T_4 + T_5 + T_6.$$

Substituting the above inequality into (2.22), we then obtain (2.13). **Q.E.D.**

The next series of lemmas separately bound the terms on RHS of (2.13).

Lemma 3. *Suppose that the conditions in Theorem 1 hold, T_1 in* (2.14) *can be bounded as*

$$\begin{aligned}
T_1 =& -E\left[\sum_{i=1}^{t} \langle \nabla f(z_i), \frac{\beta_{1,t}}{1-\beta_{1,t}}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right) \odot m_{i-1}\rangle\right] \\
\leq& H^2 \frac{\beta_1}{1-\beta_1} E\left[\sum_{i=2}^{t}\sum_{j=1}^{d} \left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]
\end{aligned}$$

Proof. [Proof of Lemma 3] Since $\|g_t\| \leq H$, by the update rule of m_t, we have $\|m_t\| \leq H$, this can be proved by induction as below.

Recall that $m_t = \beta_{1,t} m_{t-1} + (1-\beta_{1,t}) g_t$, suppose $\|m_{t-1}\| \le H$, we have

$$\|m_t\| \le (\beta_{1,t} + (1-\beta_{1,t})) \max(\|g_t\|, \|m_{t-1}\|) = \max(\|g_t\|, \|m_{t-1}\|) \le H, \tag{2.23}$$

then since $m_0 = 0$, we have $\|m_0\| \le H$ which completes the induction.

Given $\|m_t\| \le H$, we further have

$$\begin{aligned}
T_1 =& -E\left[\sum_{i=2}^{t} \langle \nabla f(z_i), \frac{\beta_{1,t}}{1-\beta_{1,t}} \left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right) \odot m_{i-1} \rangle\right] \\
\le& E\left[\sum_{i=1}^{t} \|\nabla f(z_i)\| \, \|m_{i-1}\| \left(\frac{1}{1-\beta_{1,t}} - 1\right) \sum_{j=1}^{d} \left| \left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j \right|\right] \\
\le& H^2 \frac{\beta_1}{1-\beta_1} E\left[\sum_{i=1}^{t} \sum_{j=1}^{d} \left| \left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j \right|\right]
\end{aligned}$$

where the first equality holds due to (2.12), and the last inequality is due to $\beta_1 \ge \beta_{1,i}$.

The proof is now complete. **Q.E.D.**

Lemma 4. *Suppose the conditions in Theorem 1 hold. For T_3 in* (2.16)*, we have*

$$\begin{aligned}
T_3 =& -E\left[\sum_{i=1}^{t} \langle \nabla f(z_i), \left(\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right) \alpha_i m_i / \sqrt{\hat{v}_i} \rangle\right] \\
\le& \left(\frac{\beta_1}{1-\beta_1} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right) \left(H^2 + G^2\right)
\end{aligned}$$

Proof. [Proof of Lemma 4]

$$\begin{aligned}
T_3 \le& E\left[\sum_{i=1}^{t} \left|\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right| \frac{1}{2} \left(\|\nabla f(z_i)\|^2 + \|\alpha_i m_i / \sqrt{\hat{v}_i}\|^2\right)\right] \\
\le& E\left[\sum_{i=1}^{t} \left|\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right| \frac{1}{2} \left(H^2 + G^2\right)\right] \\
=& \sum_{i=1}^{t} \left(\frac{\beta_{1,i}}{1-\beta_{1,i}} - \frac{\beta_{1,i+1}}{1-\beta_{1,i+1}}\right) \frac{1}{2} \left(H^2 + G^2\right) \\
\le& \left(\frac{\beta_1}{1-\beta_1} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right) \left(H^2 + G^2\right)
\end{aligned}$$

where the first inequality is due to$\langle a, b\rangle \leq \frac{1}{2}(\|a\|^2 + \|b\|^2)$, the second inequality is using due to upper bound on $\|\nabla f(x_t)\| \leq H$ and $\|\alpha_i m_i/\sqrt{\hat{v}_i}\| \leq G$ given by the assumptions in Theorem 1, the third equality is because $\beta_{1,t} \leq \beta_1$ and $\beta_{1,t}$ is non-increasing, the last inequality is due to telescope sum.

This completes the proof. **Q.E.D.**

Lemma 5. *Suppose the assumptions in Theorem 1 hold. For T_4 in* (2.17)*, we have*

$$\frac{2}{3L}T_4 = E\left[\sum_{i=1}^{t}\left\|\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t/\sqrt{v_t}\right\|^2\right]$$
$$\leq \left(\frac{\beta_1}{1-\beta_1} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2 G^2$$

Proof. [Proof of Lemma 5] The proof is similar to the previous lemma.

$$\begin{aligned}
\frac{2}{3L}T_4 =& E\left[\sum_{i=1}^{t}\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)^2 \|\alpha_t m_t/\sqrt{v_t}\|^2\right] \\
\leq& E\left[\sum_{i=1}^{t}\left(\frac{\beta_{1,t}}{1-\beta_{1,t}} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2 G^2\right] \\
\leq& \left(\frac{\beta_1}{1-\beta_1} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)\sum_{i=1}^{t}\left(\frac{\beta_{1,t}}{1-\beta_{1,t}} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right) G^2 \\
\leq& \left(\frac{\beta_1}{1-\beta_1} - \frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2 G^2
\end{aligned}$$

where the first inequality is due to $\|\alpha_t m_t/\sqrt{v_t}\| \leq G$ by our assumptions, the second inequality is due to non-decreasing property of $\beta_{1,t}$ and $\beta_1 \geq \beta_{1,t}$, the last inequality is due to telescoping sum.

This completes the proof. **Q.E.D.**

Lemma 6. *Suppose the assumptions in Theorem 1 hold. For T_5 in* (2.18)*, we have*

$$\begin{aligned}\frac{2}{3L}T_5 =& E\left[\sum_{i=1}^{t}\left\|\frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)\odot m_{i-1}\right\|^2\right]\\ \leq& \left(\frac{\beta_1}{1-\beta_1}\right)^2 H^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]\end{aligned}$$

Proof. [Proof of Lemma 6]

$$\begin{aligned}\frac{2}{3L}T_5 \leq& E\left[\sum_{i=2}^{t}\left(\frac{\beta_1}{1-\beta_1}\right)^2\sum_{j=1}^{d}\left(\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2(m_{i-1})_j^2\right)\right]\\ \leq& \left(\frac{\beta_1}{1-\beta_1}\right)^2 H^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]\end{aligned}$$

where the fist inequality is due to $\beta_1 \geq \beta_{1,t}$ and (2.12), the second inequality is due to $\|m_i\| < H$.

This completes the proof. **Q.E.D.**

Lemma 7. *Suppose the assumptions in Theorem 1 hold. For T_2 in* (2.15)*, we have*

$$\begin{aligned}T_2 =& -E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i), g_i/\sqrt{\hat{v}_i}\rangle\right]\\ \leq& \sum_{i=2}^{t}\frac{1}{2}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2 + L^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\frac{1}{1-\beta_1}\right)^2 E\left[\sum_{i=1}^{t-1}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right]\\ &+ L^2H^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)^4 E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left|\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right|_j^2\right]\\ &+ 2H^2E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]\\ &+ 2H^2E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right] - E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), \nabla f(x_t)/\sqrt{\hat{v}_i}\rangle\right]. \qquad (2.24)\end{aligned}$$

Proof. [Proof of Lemma 7] Recall from the definition (2.9), we have

$$z_i - x_i = \frac{\beta_{1,i}}{1-\beta_{1,i}}(x_i - x_{i-1}) = -\frac{\beta_{1,i}}{1-\beta_{1,i}}\alpha_{i-1}m_{i-1}/\sqrt{\hat{v}_{i-1}} \tag{2.25}$$

Further we have $z_1 = x_1$ by definition of z_1. We have

$$\begin{aligned} T_2 =& -E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i), g_i/\sqrt{\hat{v}_i}\rangle\right] \\ =& -E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right] - E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i)-\nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right]. \end{aligned} \tag{2.26}$$

The second term of (2.26) can be bounded as

$$\begin{aligned} & -E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i)-\nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right] \\ \leq & E\left[\sum_{i=2}^{t}\frac{1}{2}\|\nabla f(z_i)-\nabla f(x_i)\|^2 + \frac{1}{2}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right] \\ \leq & \frac{L^2}{2}T_7 + \frac{1}{2}E\left[\sum_{i=2}^{t}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right], \end{aligned} \tag{2.27}$$

where the first inequality is because $\langle a, b\rangle \leq \frac{1}{2}\left(\|a\|^2 + \|b\|^2\right)$ and the fact that $z_1 = x_1$, the second inequality is because

$$\|\nabla f(z_i) - \nabla f(x_i)\| \leq L\|z_i - x_i\| = L\|\frac{\beta_{1,t}}{1-\beta_{1,t}}\alpha_{i-1}m_{i-1}/\sqrt{\hat{v}_{i-1}}\|,$$

and T_7 is defined as

$$T_7 = E\left[\sum_{i=2}^{t}\left\|\frac{\beta_{1,i}}{1-\beta_{1,i}}\alpha_{i-1}m_{i-1}/\sqrt{\hat{v}_{i-1}}\right\|^2\right]. \tag{2.28}$$

We next bound the T_7 in (2.28), by update rule $m_i = \beta_{1,i}m_{i-1} + (1-\beta_{1,i}g_i)$, we have

$m_i = \sum_{k=1}^{i}[(\prod_{l=k+1}^{i}\beta_{1,l})(1-\beta_{1,k})g_k]$. Based on that, we obtain

$$
\begin{aligned}
T_7 \leq & \left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\frac{\alpha_{i-1}m_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right] \\
= & \left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\sum_{k=1}^{i-1}\frac{\alpha_{i-1}\left(\prod_{l=k+1}^{i-1}\beta_{1,l}\right)(1-\beta_{1,k})g_k}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right] \\
\leq & 2\left(\frac{\beta_1}{1-\beta_1}\right)^2 \underbrace{E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\sum_{k=1}^{i-1}\frac{\alpha_{k}\left(\prod_{l=k+1}^{i-1}\beta_{1,l}\right)(1-\beta_{1,k})g_k}{\sqrt{\hat{v}_{k}}}\right)_j^2\right]}_{T_8} \\
& + 2\left(\frac{\beta_1}{1-\beta_1}\right)^2 \underbrace{E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\sum_{k=1}^{i-1}\left(\prod_{l=k+1}^{i-1}\beta_{1,l}\right)(1-\beta_{1,k})(g_k)_j\left(\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}-\frac{\alpha_k}{\sqrt{\hat{v}_k}}\right)_j\right)^2\right]}_{T_9}
\end{aligned}
\tag{2.29}
$$

where the first inequality is due to $\beta_{1,t} \leq \beta_1$, the second equality is by substituting expression of m_t, the last inequality is because $(a+b)^2 \leq 2(\|a\|^2 + \|b\|^2)$, and we have introduced T_8 and T_9 for ease of notation.

In (2.29), we first bound T_8 as below

$$
\begin{aligned}
T_8 =& E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\sum_{k=1}^{i-1}\sum_{p=1}^{i-1}\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j\left(\prod_{l=k+1}^{i-1}\beta_{1,k}\right)(1-\beta_{1,k})\left(\frac{\alpha_p g_p}{\sqrt{\hat{v}_p}}\right)_j\left(\prod_{q=p+1}^{i-1}\beta_{1,p}\right)(1-\beta_{1,p})\right] \\
\overset{(i)}{\leq}& E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\sum_{k=1}^{i-1}\sum_{p=1}^{i-1}\left(\beta_1^{i-1-k}\right)\left(\beta_1^{i-1-p}\right)\frac{1}{2}\left(\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2+\left(\frac{\alpha_p g_p}{\sqrt{\hat{v}_p}}\right)_j^2\right)\right] \\
\overset{(ii)}{=}& E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\sum_{k=1}^{i-1}\left(\beta_1^{i-1-k}\right)\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2\sum_{p=1}^{i-1}\left(\beta_1^{i-1-p}\right)\right] \\
\overset{(iii)}{\leq}& \frac{1}{1-\beta_1}E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\sum_{k=1}^{i-1}\left(\beta_1^{i-1-k}\right)\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2\right] \\
\overset{(iv)}{=}& \frac{1}{1-\beta_1}E\left[\sum_{k=1}^{t-1}\sum_{j=1}^{d}\sum_{i=k+1}^{t}\left(\beta_1^{i-1-k}\right)\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2\right] \\
\leq& \left(\frac{1}{1-\beta_1}\right)^2 E\left[\sum_{k=1}^{t-1}\sum_{j=1}^{d}\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2\right] = \left(\frac{1}{1-\beta_1}\right)^2 E\left[\sum_{i=1}^{t-1}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right] \qquad (2.30)
\end{aligned}
$$

where (i) is due to $ab < \frac{1}{2}(a^2+b^2)$ and follows from $\beta_{1,t} \leq \beta_1$ and $\beta_{1,t} \in [0,1)$, (ii) is due to symmetry of p and k in the summation, (iii) is because of $\sum_{p=1}^{i-1}\left(\beta_1^{i-1-p}\right) \leq \frac{1}{1-\beta_1}$, (iv) is exchanging order of summation, and the second-last inequality is due to the similar reason as (iii).

For the T_9 in (2.29), we have

$$
\begin{aligned}
T_9 =& E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\sum_{k=1}^{i-1}\left(\prod_{l=k+1}^{i-1}\beta_{1,k}\right)(1-\beta_{1,k})(g_k)_j\left(\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}-\frac{\alpha_k}{\sqrt{\hat{v}_k}}\right)_j\right)^2\right] \\
\leq& H^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\sum_{k=1}^{i-1}\left(\prod_{l=k+1}^{i-1}\beta_{1,k}\right)\left|\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}-\frac{\alpha_k}{\sqrt{\hat{v}_k}}\right|_j\right)^2\right] \\
\leq& H^2 E\left[\sum_{i=1}^{t-1}\sum_{j=1}^{d}\left(\sum_{k=1}^{i}\beta_1^{i-k}\left|\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_k}{\sqrt{\hat{v}_k}}\right|_j\right)^2\right] \\
\leq& H^2 E\left[\sum_{i=1}^{t-1}\sum_{j=1}^{d}\left(\sum_{k=1}^{i}\beta_1^{i-k}\sum_{l=k+1}^{i}\left|\frac{\alpha_l}{\sqrt{\hat{v}_l}}-\frac{\alpha_{l-1}}{\sqrt{\hat{v}_{l-1}}}\right|_j\right)^2\right] \qquad (2.31)
\end{aligned}
$$

where the first inequality holds due to $\beta_{1,k} < 1$ and $|(g_k)_j| \leq H$, the second inequality holds due to $\beta_{1,k} \leq \beta_1$, and the last inequality applied the triangle inequality. For RHS of (2.31), using Lemma 8 (that will be proved later) with $a_i = \left|\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right|_j$, we further have

$$
\begin{aligned}
T_9 \leq& H^2 E\left[\sum_{i=1}^{t-1}\sum_{j=1}^{d}\left(\sum_{k=1}^{i}\beta_1^{i-k}\sum_{l=k+1}^{i}\left|\frac{\alpha_l}{\sqrt{\hat{v}_l}}-\frac{\alpha_{l-1}}{\sqrt{\hat{v}_{l-1}}}\right|_j\right)^2\right] \\
\leq& H^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)^2 E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left|\frac{\alpha_l}{\sqrt{\hat{v}_l}}-\frac{\alpha_{l-1}}{\sqrt{\hat{v}_{l-1}}}\right|_j^2\right] \qquad (2.32)
\end{aligned}
$$

Based on (2.27), (2.29), (2.30) and (2.32), we can then bound the second term of

(2.26) as

$$
\begin{aligned}
&- E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(z_i)-\nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right]\\
\leq& L^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\left(\frac{1}{1-\beta_1}\right)^2 E\left[\sum_{i=1}^{t-1}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right]\\
&+ L^2H^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)^4 E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left|\frac{\alpha_l}{\sqrt{\hat{v}_l}}-\frac{\alpha_{l-1}}{\sqrt{\hat{v}_{l-1}}}\right|_j^2\right]\\
&+ \frac{1}{2}E\left[\sum_{i=2}^{t}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right]. \qquad (2.33)
\end{aligned}
$$

Let us turn to the first term in (2.26). Reparameterize g_t as $g_t = \nabla f(x_t) + \delta_t$ with $E[\delta_t] = 0$, we have

$$
\begin{aligned}
&E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right]\\
=&E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), (\nabla f(x_i)+\delta_i)/\sqrt{\hat{v}_i}\rangle\right]\\
=&E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), \nabla f(x_i)/\sqrt{\hat{v}_i}\rangle\right] + E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), \delta_i/\sqrt{\hat{v}_i}\rangle\right]. \qquad (2.34)
\end{aligned}
$$

It can be seen that the first term in RHS of (2.34) is the desired descent quantity, the second term is a bias term to be bounded. For the second term in RHS of (2.34), we have

$$
\begin{aligned}
&E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i), \delta_i/\sqrt{\hat{v}_i}\rangle\right]\\
=&E\left[\sum_{i=2}^{t}\langle\nabla f(x_i), \delta_i\odot(\alpha_i/\sqrt{\hat{v}_i}-\alpha_{i-1}/\sqrt{\hat{v}_{i-1}})\rangle\right] + E\left[\sum_{i=2}^{t}\alpha_{i-1}\langle\nabla f(x_i), \delta_i\odot(1/\sqrt{\hat{v}_{i-1}})\rangle\right]\\
&+ E\left[\alpha_1\langle\nabla f(x_1), \delta_1/\sqrt{\hat{v}_1}\rangle\right]\\
\geq&E\left[\sum_{i=2}^{t}\langle\nabla f(x_i), \delta_i\odot(\alpha_i/\sqrt{\hat{v}_i}-\alpha_{i-1}/\sqrt{\hat{v}_{i-1}})\rangle\right] - 2H^2E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right] \qquad (2.35)
\end{aligned}
$$

where the last equation is because given $x_i, \hat{v}_{i-1}$, $E\left[\delta_i \odot (1/\sqrt{\hat{v}_{i-1}})|x_i, \hat{v}_{i-1}\right] = 0$ and $\|\delta_i\| \leq 2H$ due to $\|g_i\| \leq H$ and $\|\nabla f(x_i)\| \leq H$ based on Assumptions A2 and A3. Further, we have

$$
\begin{aligned}
& E\left[\sum_{i=2}^{t}\langle \nabla f(x_i), \delta_t \odot (\alpha_i/\sqrt{\hat{v}_i} - \alpha_{i-1}/\sqrt{\hat{v}_{i-1}})\rangle\right] \\
=& E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}(\nabla f(x_i))_j(\delta_t)_j(\alpha_i/(\sqrt{\hat{v}_i})_j - \alpha_{i-1}/(\sqrt{\hat{v}_{i-1}})_j)\right] \\
\geq& - E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}|(\nabla f(x_i))_j|\,|(\delta_t)_j|\left|(\alpha_i/(\sqrt{\hat{v}_i})_j - \alpha_{i-1}/(\sqrt{\hat{v}_{i-1}})_j)\right|\right] \\
\geq& - 2H^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|(\alpha_i/(\sqrt{\hat{v}_i})_j - \alpha_{i-1}/(\sqrt{\hat{v}_{i-1}})_j)\right|\right] \qquad (2.36)
\end{aligned}
$$

Substituting (2.35) and (2.36) into (2.34), we then bound the first term of (2.26) as

$$
\begin{aligned}
& - E\left[\sum_{i=1}^{t}\alpha_i\langle \nabla f(x_i), g_i/\sqrt{\hat{v}_i}\rangle\right] \\
\leq& 2H^2 E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|(\alpha_i/(\sqrt{\hat{v}_i})_j - \alpha_{i-1}/(\sqrt{\hat{v}_{i-1}})_j)\right|\right] + 2H^2 E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right] \\
& - E\left[\sum_{i=1}^{t}\alpha_i\langle \nabla f(x_i), \nabla f(x_i)/\sqrt{\hat{v}_i}\rangle\right] \qquad (2.37)
\end{aligned}
$$

We finally apply (2.37) and (2.33) to obtain (2.24). The proof is now complete. **Q.E.D.**

Lemma 8. *For $a_i \geq 0$, $\beta \in [0, 1)$, and $b_i = \sum_{k=1}^{i}\beta^{i-k}\sum_{l=k+1}^{i}a_l$, we have*

$$
\sum_{i=1}^{t} b_i^2 \leq \left(\frac{1}{1-\beta}\right)^2\left(\frac{\beta}{1-\beta}\right)^2\sum_{i=2}^{t}a_i^2
$$

Proof. [Proof of Lemma 8] The result is proved by following

$$
\begin{aligned}
&\sum_{i=1}^{t} b_i^2 = \sum_{i=1}^{t}\left(\sum_{k=1}^{i}\beta^{i-k}\sum_{l=k+1}^{i} a_l\right)^2 \\
&\overset{(i)}{=}\sum_{i=1}^{t}\left(\sum_{l=2}^{i}\sum_{k=1}^{l-1}\beta^{i-k}a_l\right)^2 = \sum_{i=1}^{t}\left(\sum_{l=2}^{i}\beta^{i-l+1}a_l\sum_{k=1}^{l-1}\beta^{l-1-k}\right)^2 \\
&\overset{(ii)}{\leq}\left(\frac{1}{1-\beta}\right)^2\sum_{i=1}^{t}\left(\sum_{l=2}^{i}\beta^{i-l+1}a_l\right)^2 = \left(\frac{1}{1-\beta}\right)^2\sum_{i=1}^{t}\left(\sum_{l=2}^{i}\sum_{m=2}^{i}\beta^{i-l+1}a_l\beta^{i-m+1}a_m\right) \\
&\overset{(iii)}{\leq}\left(\frac{1}{1-\beta}\right)^2\sum_{i=1}^{t}\sum_{l=2}^{i}\sum_{m=2}^{i}\beta^{i-l+1}\beta^{i-m+1}\frac{1}{2}\left(a_l^2+a_m^2\right) \\
&\overset{(iv)}{=}\left(\frac{1}{1-\beta}\right)^2\sum_{i=1}^{t}\sum_{l=2}^{i}\sum_{m=2}^{i}\beta^{i-l+1}\beta^{i-m+1}a_l^2 \overset{(v)}{\leq}\left(\frac{1}{1-\beta}\right)^2\frac{\beta}{1-\beta}\sum_{l=2}^{t}\sum_{i=l}^{t}\beta^{i-l+1}a_l^2 \\
&\leq\left(\frac{1}{1-\beta}\right)^2\left(\frac{\beta}{1-\beta}\right)^2\sum_{l=2}^{t}a_l^2
\end{aligned}
$$

where (i) is by changing order of summation, (ii) is due to $\sum_{k=1}^{l-1}\beta^{l-1-k}\leq\frac{1}{1-\beta}$, (iii) is by the fact that $ab\leq\frac{1}{2}(a^2+b^2)$, (iv) is due to symmetry of a_l and a_m in the summation, (v) is because $\sum_{m=2}^{i}\beta^{i-m+1}\leq\frac{\beta}{1-\beta}$ and the last inequality is for similar reason.

This completes the proof. **Q.E.D.**

Proof of Theorem 1

Proof. [Proof of Theorem 1] We combine Lemma 2, Lemma 3, Lemma 4, Lemma 5, Lemma 6, and Lemma 7 to bound the overall expected descent of the objective. First,

from Lemma 2, we have

$$
\begin{aligned}
&E\left[f(z_{t+1}) - f(z_1)\right] \leq \sum_{i=1}^{6} T_i \\
=& -E\left[\sum_{i=1}^{t}\langle \nabla f(z_i), \frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right) \odot m_{i-1}\rangle\right] - E\left[\sum_{i=1}^{t}\alpha_i\langle \nabla f(z_i), g_i/\sqrt{\hat{v}_i}\rangle\right] \\
&- E\left[\sum_{i=1}^{t}\langle \nabla f(z_i), \left(\frac{\beta_{1,i+1}}{1-\beta_{1,i+1}} - \frac{\beta_{1,i}}{1-\beta_{1,i}}\right)\alpha_i m_i/\sqrt{\hat{v}_i}\rangle\right] \\
&+ E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\left(\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}} - \frac{\beta_{1,t}}{1-\beta_{1,t}}\right)\alpha_t m_t/\sqrt{v_t}\right\|^2\right] \\
&+ E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\frac{\beta_{1,i}}{1-\beta_{1,i}}\left(\frac{\alpha_t}{\sqrt{\hat{v}_i}} - \frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right) \odot m_{i-1}\right\|^2\right] + E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\alpha_i g_i/\sqrt{\hat{v}_i}\right\|^2\right]
\end{aligned}
\tag{2.38}
$$

Then from above inequality and Lemma 3, Lemma 4, Lemma 5, Lemma 6, Lemma 7,

we get

$$
\begin{aligned}
&E\left[f(z_{t+1})-f(z_1)\right]\\
\leq& H^2\frac{\beta_1}{1-\beta_1}E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]\\
&+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)\left(H^2+G^2\right)+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2G^2\\
&+\left(\frac{\beta_1}{1-\beta_1}\right)^2H^2E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]+E\left[\sum_{i=1}^{t}\frac{3}{2}L\left\|\alpha_i g_i/\sqrt{\hat{v}_i}\right\|^2\right]\\
&+E\left[\sum_{i=2}^{t}\frac{1}{2}\|\alpha_i g_i/\sqrt{\hat{v}_i}\|^2\right]+L^2\frac{\beta_1}{1-\beta_1}\left(\frac{1}{1-\beta_1}\right)^2E\left[\sum_{k=1}^{t-1}\sum_{j=1}^{d}\left(\frac{\alpha_k g_k}{\sqrt{\hat{v}_k}}\right)_j^2\right]\\
&+L^2H^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)^4E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]\\
&+2H^2E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]\\
&+2H^2E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right]-E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i),\nabla f(x_i)/\sqrt{\hat{v}_i}\rangle\right]
\end{aligned}
$$

By merging similar terms in above inequality, we further have

$$
\begin{aligned}
&E\left[f(z_{t+1})-f(z_1)\right]\\
\leq&\left(H^2\frac{\beta_1}{1-\beta_1}+2H^2\right)E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]\\
&+\left(1+L^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\right)H^2\left(\frac{\beta_1}{1-\beta_1}\right)^2E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]\\
&+\left(\frac{3}{2}L+\frac{1}{2}+L^2\frac{\beta_1}{1-\beta_1}\left(\frac{1}{1-\beta_1}\right)^2\right)E\left[\sum_{i=1}^{t}\left\|\alpha_i g_i/\sqrt{\hat{v}_i}\right\|^2\right]\\
&+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)\left(H^2+G^2\right)+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2G^2\\
&+2H^2E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right]-E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i),\nabla f(x_i)/\sqrt{\hat{v}_i}\rangle\right]
\end{aligned}
\tag{2.39}
$$

Rearranging (2.39), we have

$$
\begin{aligned}
&E\left[\sum_{i=1}^{t}\alpha_i\langle\nabla f(x_i),\nabla f(x_i)/\sqrt{\hat{v}_i}\rangle\right]\\
&\leq\left(H^2\frac{\beta_1}{1-\beta_1}+2H^2\right)E\left[\sum_{i=2}^{t}\sum_{j=1}^{d}\left|\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j\right|\right]+\\
&+\left(1+L^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)\right)^2H^2\left(\frac{\beta_1}{1-\beta_1}\right)^2E\left[\sum_{j=1}^{d}\sum_{i=2}^{t-1}\left(\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right)_j^2\right]\\
&+\left(\frac{3}{2}L+\frac{1}{2}+L^2\frac{\beta_1}{1-\beta_1}\left(\frac{1}{1-\beta_1}\right)^2\right)E\left[\sum_{i=1}^{t}\left\|\alpha_i g_i/\sqrt{\hat{v}_i}\right\|^2\right]\\
&+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)\left(H^2+G^2\right)+\left(\frac{\beta_1}{1-\beta_1}-\frac{\beta_{1,t+1}}{1-\beta_{1,t+1}}\right)^2G^2\\
&+2H^2E\left[\sum_{j=1}^{d}(\alpha_1/\sqrt{\hat{v}_1})_j\right]+E\left[f(z_1)-f(z_{t+1})\right]\\
\leq&E\left[C_1\sum_{i=1}^{t}\left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2+C_2\sum_{i=2}^{t}\left\|\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right\|_1+C_3\sum_{i=2}^{t-1}\left\|\frac{\alpha_i}{\sqrt{\hat{v}_i}}-\frac{\alpha_{i-1}}{\sqrt{\hat{v}_{i-1}}}\right\|^2\right]+C_4
\end{aligned}
$$

where

$$
\begin{aligned}
C_1&\triangleq\left(\frac{3}{2}L+\frac{1}{2}+L^2\frac{\beta_1}{1-\beta_1}\left(\frac{1}{1-\beta_1}\right)^2\right)\\
C_2&\triangleq\left(H^2\frac{\beta_1}{1-\beta_1}+2H^2\right)\\
C_3&\triangleq\left(1+L^2\left(\frac{1}{1-\beta_1}\right)^2\left(\frac{\beta_1}{1-\beta_1}\right)\right)H^2\left(\frac{\beta_1}{1-\beta_1}\right)^2\\
C_4&\triangleq\left(\frac{\beta_1}{1-\beta_1}\right)\left(H^2+G^2\right)+\left(\frac{\beta_1}{1-\beta_1}\right)^2G^2\\
&\quad+2H^2E\left[\|\alpha_1/\sqrt{\hat{v}_1}\|_1\right]+E\left[f(z_1)-f(z^*)\right]
\end{aligned}
$$

and z^* is an optimal of f, i.e. $z^*\in\operatorname{argmin}_z f(z)$.

Using the fact that $(\alpha_i/\sqrt{\hat{v}_i})_j\geq\gamma_i,\forall j$ by definition, inequality (2.4) directly follows. This completes the proof. **Q.E.D.**

2.6.2 Proof of Corollary 1

Proof. [Proof of Corollary 1]

We first bound non-constant terms in RHS of (2.3), which is given by

$$E\left[C_1\sum_{t=1}^{T}\left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2+C_2\sum_{t=2}^{T}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1+C_3\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right]+C_4.$$

For the term with C_1, assume $\min_{j\in[d]}(\sqrt{\hat{v}_1})_j \geq c > 0$ (this is natural since if it is 0, division by 0 error will happen), we have

$$\begin{aligned}
&E\left[\sum_{t=1}^{T}\left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2\right]\\
\leq&E\left[\sum_{t=1}^{T}\|\alpha_t g_t/c\|^2\right]=E\left[\sum_{t=1}^{T}\left\|\frac{1}{\sqrt{t}}g_t/c\right\|^2\right]=E\left[\sum_{t=1}^{T}\left(\frac{1}{c\sqrt{t}}\right)^2\|g_t\|^2\right]\\
\leq&H^2/c^2\sum_{t=1}^{T}\frac{1}{t}\leq H^2/c^2(1+\log T)
\end{aligned}$$

where the first inequality is due to $(\hat{v}_t)_j \geq (\hat{v}_{t-1})_j$, and the last inequality is due to $\sum_{t=1}^{T} 1/t \leq 1+\log T$.

For the term with C_2, we have

$$\begin{aligned}
&E\left[\sum_{t=2}^{T}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1\right]=E\left[\sum_{j=1}^{d}\sum_{t=2}^{T}\left(\frac{\alpha_{t-1}}{(\sqrt{\hat{v}_{t-1}})_j}-\frac{\alpha_t}{(\sqrt{\hat{v}_t})_j}\right)\right]\\
=&E\left[\sum_{j=1}^{d}\left(\frac{\alpha_1}{(\sqrt{\hat{v}_1})_j}-\frac{\alpha_T}{(\sqrt{\hat{v}_T})_j}\right)\right]\leq E\left[\sum_{j=1}^{d}\frac{\alpha_1}{(\sqrt{\hat{v}_1})_j}\right]\leq d/c
\end{aligned}\tag{2.40}$$

where the first equality is due to$(\hat{v}_t)_j \geq (\hat{v}_{t-1})_j$ and $\alpha_t \leq \alpha_{t-1}$, and the second equality is due to telescope sum.

For the term with C_3, we have

$$\begin{aligned}
&E\left[\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right]\\
\leq &E\left[\frac{1}{c}\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1\right]\\
\leq &d/c^2
\end{aligned}$$

where the first inequality is due to $|(\alpha_t/\sqrt{\hat{v}_t}-\alpha_{t-1}/\sqrt{\hat{v}_{t-1}})_j|\leq 1/c$.

Then we have for AMSGRAD,

$$\begin{aligned}
&E\left[C_1\sum_{t=1}^{T}\left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2+C_2\sum_{t=2}^{T}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1+C_3\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right]+C_4\\
\leq &C_1H^2/c^2(1+\log T)+C_2d/c+C_3(d/c)^2+C_4
\end{aligned}\tag{2.41}$$

Now we lower bound the effective stepsizes, since $\hat{v}_t$ is exponential moving average of g_t^2 and $\|g_t\|\leq H$, we have $(\hat{v}_t)_j\leq H^2$, we have

$$\alpha/(\sqrt{\hat{v}_t})_j\geq\frac{1}{H\sqrt{t}}$$

And thus

$$E\left[\sum_{t=1}^{T}\alpha_i\langle\nabla f(x_t),\nabla f(x_t)/\sqrt{\hat{v}_t}\rangle\right]\geq E\left[\sum_{t=1}^{T}\frac{1}{H\sqrt{t}}\|\nabla f(x_t)\|^2\right]\geq\frac{\sqrt{T}}{H}\min_{t\in[T]}E\left[\|\nabla f(x_t)\|^2\right]\tag{2.42}$$

Then by (2.3), (2.41) and (2.42), we have

$$\frac{1}{H}\sqrt{T}\min_{t\in[T]}E\left[\|\nabla f(x_t)\|^2\right]\leq C_1H^2/c^2(1+\log T)+C_2d/c+C_3d/c^2+C_4$$

which is equivalent to

$$\begin{aligned}
&\min_{t\in[T]} E\left[\|\nabla f(x_t)\|^2\right] \\
\leq &\frac{H}{\sqrt{T}}\left(C_1 H^2/c^2(1+\log T) + C_2 d/c + C_3 d/c^2 + C_4\right) \\
= &\frac{1}{\sqrt{T}}\left(Q_1 + Q_2 \log T\right)
\end{aligned}$$

One more thing is to verify the assumption $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq G$ in Theorem 1, since $\alpha_{t+1}/(\sqrt{\hat{v}_{t+1}})_j \leq \alpha_t/(\sqrt{\hat{v}_t})_j$ and $\alpha_1/(\sqrt{\hat{v}_1})_j \leq 1/c$ in the algorithm, we have $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq \|m_t\|/c \leq H/c$.

This completes the proof. **Q.E.D.**

2.6.3 Proof of Corollary 2

Proof. [Proof of Corollary 2]

The proof is similar to proof for Corollary 1, first let's bound RHS of (2.3) which is

$$E\left[C_1 \sum_{t=1}^{T} \left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2 + C_2 \sum_{t=2}^{T} \left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1 + C_3 \sum_{t=2}^{T-1} \left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}} - \frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right] + C_4$$

We recall from Table 2.1 that in AdaGrad, $\hat{v}_t = \frac{1}{t}\sum_{i=1}^{t} g_t^2$. Thus, when $\alpha_t = 1/\sqrt{t}$, we obtain $\alpha_t/\sqrt{\hat{v}_t} = 1/\sum_{i=1}^{t} g_i^2$. We assume $\min_{j\in[d]} |(g_1)_j| \geq c > 0$, which is equivalent to $\min_{j\in[d]}(\sqrt{\hat{v}_1})_j \geq c > 0$ (a requirement of the AdaGrad). For C_1 term we have

$$\begin{aligned}
&E\left[\sum_{t=1}^{T} \left\|\alpha_t g_t/\sqrt{\hat{v}_t}\right\|^2\right] = E\left[\sum_{t=1}^{T} \left\|\frac{g_t}{\sqrt{\sum_{i=1}^{t} g_i^2}}\right\|^2\right] = E\left[\sum_{j=1}^{d}\sum_{t=1}^{T} \frac{(g_t)_j^2}{\sum_{i=1}^{t}(g_i)_j^2}\right] \\
\leq &E\left[\sum_{j=1}^{d}\left(1 - \log((g_1)_j^2) + \log\sum_{t=1}^{T}(g_t)_j^2\right)\right] \leq d(1 - \log(c^2) + 2\log H + \log T)
\end{aligned}$$

where the third inequality used Lemma 9 and the last inequality used $\|g_t\| \leq H$ and $\min_{j\in[d]} |(g_1)_j| \geq c > 0$.

For C_2 term we have

$$E\left[\sum_{t=2}^{T}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1\right]=E\left[\sum_{j=1}^{d}\sum_{t=2}^{T}\left(\frac{1}{\sqrt{\sum_{i=1}^{t-1}(g_i)_j^2}}-\frac{1}{\sqrt{\sum_{i=1}^{t}(g_i)_j^2}}\right)\right]$$
$$=E\left[\sum_{j=1}^{d}\left(\frac{1}{\sqrt{(g_1)_j^2}}-\frac{1}{\sqrt{\sum_{i=1}^{T}(g_i)_j^2}}\right)\right]\leq d/c$$

For C_3 term we have

$$\begin{aligned}&E\left[\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|^2\right]\\ \leq&E\left[\frac{1}{c}\sum_{t=2}^{T-1}\left\|\frac{\alpha_t}{\sqrt{\hat{v}_t}}-\frac{\alpha_{t-1}}{\sqrt{\hat{v}_{t-1}}}\right\|_1\right]\\ \leq&d/c^2\end{aligned}$$

where the first inequality is due to $|(\alpha_t/\sqrt{\hat{v}_t}-\alpha_{t-1}/\sqrt{\hat{v}_{t-1}})_j|\leq 1/c$.

Now we lower bound the effective stepsizes $\alpha_t/(\sqrt{\hat{v}_t})_j$,

$$\frac{\alpha_t}{(\sqrt{\hat{v}_t})_j}=\frac{1}{\sqrt{\sum_{i=1}^{t}(g_i)_j^2}}\geq\frac{1}{H\sqrt{t}},$$

where we recall that $\alpha_t=1/\sqrt{t}$ and $\|g_t\|\leq H$. Following the same argument in the proof of Corollary 1 and the previously derived upper bounds, we have

$$\frac{\sqrt{T}}{H}\min_{t\in[T]}E\left[\|\nabla f(x_t)\|^2\right]\leq C_1d(1-\log(c^2)+2\log H+\log T)+C_2d/c+C_3d/c^2+C_4$$

which yields

$$\begin{aligned}&\min_{t\in[T]}E\left[\|\nabla f(x_t)\|^2\right]\\ \leq&\frac{H}{\sqrt{T}}\left(C_1d(1-\log(c^2)+2\log H+\log T)+C_2d/c+C_3d/c^2+C_4\right)\\ =&\frac{1}{\sqrt{T}}\left(Q_1'+Q_2'\log T\right)\end{aligned}$$

The last thing is to verify the assumption $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq G$ in Theorem 1, since $\alpha_{t+1}/(\sqrt{\hat{v}_{t+1}})_j \leq \alpha_t/(\sqrt{\hat{v}_t})_j$ and $\alpha_1/(\sqrt{\hat{v}_1})_j \leq 1/c$ in the algorithm, we have $\|\alpha_t m_t/\sqrt{\hat{v}_t}\| \leq \|m_t\|/c \leq H/c$.

This completes the proof. **Q.E.D.**

Lemma 9. *For $a_t \geq 0$ and $\sum_{i=1}^t a_i \neq 0$, we have*

$$\sum_{t=1}^{T} \frac{a_t}{\sum_{i=1}^t a_i} \leq 1 - \log a_1 + \log \sum_{i=1}^{T} a_i.$$

Proof. [Proof of Lemma 9] We will prove it by induction. Suppose

$$\sum_{t=1}^{T-1} \frac{a_t}{\sum_{i=1}^t a_i} \leq 1 - \log a_1 + \log \sum_{i=1}^{T-1} a_i,$$

we have

$$\sum_{t=1}^{T} \frac{a_t}{\sum_{i=1}^t a_i} = \frac{a_T}{\sum_{i=1}^T a_i} + \sum_{t=1}^{T-1} \frac{a_t}{\sum_{i=1}^t a_i} \leq \frac{a_T}{\sum_{i=1}^T a_i} + 1 - \log a_1 + \log \sum_{i=1}^{T-1} a_i.$$

Applying the definition of concavity to $\log(x)$, with $f(z) \triangleq \log(z)$, we have $f(z) \leq f(z_0) + f'(z_0)(z - z_0)$, then substitute $z = x - b, z_0 = x$, we have $f(x-b) \leq f(x) + f'(x)(-b)$ which is equivalent to $\log(x) \geq \log(x-b) + b/x$ for $b < x$, using $x = \sum_{i=1}^T a_i$, $b = a_T$, we have

$$\log \sum_{i=1}^{T} a_i \geq \log \sum_{i=1}^{T-1} a_i + \frac{a_T}{\sum_{i=1}^T a_i}$$

and then

$$\sum_{t=1}^{T} \frac{a_t}{\sum_{i=1}^t a_i} \leq \frac{a_T}{\sum_{i=1}^T a_i} + 1 - \log a_1 + \log \sum_{i=1}^{T-1} a_i \leq 1 - \log a_1 + \log \sum_{i=1}^{T} a_i.$$

Now it remains to check first iteration. We have

$$\frac{a_1}{a_1} = 1 \leq 1 - \log(a_1) + \log(a_1) = 1$$

This completes the proof. **Q.E.D.**

Chapter 3

Zeroth order optimization with adaptive gradient

3.1 Introduction

In this chapter, we will take a step on developing zeroth-order adaptive gradient (momentum) methods, with an application in black-box adversarial attacks. The development of gradient-free optimization methods has become increasingly important to solve many machine learning problems in which explicit expressions of the gradients are expensive or infeasible to obtain [Liu et al., 2018b, Sahu et al., 2018, Feurer et al., 2015, Kotthoff et al., 2017, Chen et al., 2017, Ilyas et al., 2018a, Tu et al., 2018]. *Zeroth-Order (ZO)* optimization methods, one type of gradient-free optimization methods, mimic first-order (FO) methods but approximate the full gradient (or stochastic gradient) through random gradient estimates, given by the difference of function values at random query points [Nesterov and Spokoiny, 2015, Ghadimi and Lan, 2013]. Compared to Bayesian optimization, derivative-free trust region methods, genetic algorithms and other types of gradient-free methods [Shahriari et al., 2016, Conn et al., 2009a, Whitley, 1994, Conn et al., 2009b], ZO optimization has two main advantages: a) ease of implementation, via slight modification of commonly-used gradient-based algorithms, and b) comparable convergence rates to first-order algorithms.

Due to the stochastic nature of ZO optimization, which arises from both data sampling and random gradient estimation, existing ZO methods suffer from large variance of

the noisy gradient compared to FO stochastic methods [Liu et al., 2019]. In practice, this causes poor convergence performance and/or function query efficiency. To partially mitigate these issues, ZO sign-based SGD (ZO-signSGD) was proposed by Liu et al. [2019] with the rationale that taking the sign of random gradient estimates (i.e., normalizing gradient estimates elementwise) as the descent direction improves the robustness of gradient estimators to stochastic noise. Although ZO-signSGD has faster convergence speed than many existing ZO algorithms, it is only guaranteed to converge to a neighborhood of a solution. In the FO setting, taking the sign of a stochastic gradient as the descent direction gives rise to signSGD [Bernstein et al., 2018]. The use of sign of stochastic gradients also appears in adaptive gradient (momentum) methods (AdaMM) such as Adam [Kingma and Ba, 2014], RMSProp [Tieleman and Hinton, 2012], AMSGrad [Reddi et al., 2018], Padam [Chen and Gu, 2018], and AdaFom [Chen et al., 2019a]. Indeed, it has been suggested by Balles and Hennig [2018] that AdaMM enjoy dual advantages of sign descent and variance adaption.

Considering the motivation of ZO-signSGD and the success of AdaMM in FO optimization, one question arises: Can we generalize AdaMM to the ZO regime? To answer this question, we develop the zeroth-order adaptive momentum method (ZO-AdaMM) and analyze its convergence properties in nonconvex settings for both constrained and constrained optimization.

Contributions *Theoretically*, for nonconvex optimization, we show that ZO-AdaMM is roughly a factor of $O(\sqrt{d})$ worse than that of the FO AdaMM algorithm, where d is the number of optimization variables. We also show that the *Euclidean* projection based AdaMM-type methods could suffer non-convergence issues for constrained optimization. This highlights the necessity of *Mahalanobis distance* based projection. And we establish the Mahalanobis distance based convergence analysis, which makes the first step toward understanding adaptive learning rate methods for nonconvex constrained optimization.

Practically, we formalize the experimental comparison of ZO-AdaMM with 6 state-of-the-art ZO algorithms in the application of black-box adversarial attacks to generate both per-image and universal adversarial perturbations. Our proposal could provide an experimental benchmark for future studies on ZO optimization.

Related work Many types of ZO algorithms have been developed, and their convergence rates have been rigorously studied under different problem settings. We highlight some recent works as below. For unconstrained stochastic optimization, ZO stochastic gradient descent (ZO-SGD) [Ghadimi and Lan, 2013] and ZO stochastic coordinate descent (ZO-SCD) [Lian et al., 2016] were proposed, which have $O(\sqrt{d}/\sqrt{T})$ convergence rate, where T is the number of iterations. Compared to FO stochastic algorithms, ZO optimization suffers a slowdown dependent on the variable dimension d, e.g., $O(\sqrt{d})$ for ZO-SGD and ZO-SCD. In Duchi et al. [2015], the tightness of the dimension-dependent factor $O(\sqrt{d})$ has been proved in the framework of ZO stochastic mirror descent (ZO-SMD). In order to further improve the iteration complexity of ZO algorithms, the technique of variance reduction was applied to ZO-SGD and ZO-SCD, leading to ZO stochastic variance reduced algorithms with an improved convergence rate in T, namely, $O(d/T)$ [Liu et al., 2018c, Gu et al., 2016, Liu et al., 2018a]. This improvement is aligned with ZO gradient descent (ZO-GD) for deterministic nonconvex programming [Nesterov and Spokoiny, 2015]. Moreover, ZO versions of proximal SGD (ProxSGD) [Ghadimi et al., 2016], Frank-Wolfe (FW) [Balasubramanian and Ghadimi, 2018, Sahu et al., 2018, Chen et al., 2018], and online alternating direction method of multipliers (OADMM) [Liu et al., 2018b, Gao et al., 2014] have been developed for constrained optimization. Aside from the recent works on ZO algorithms mentioned before, there is rich literature in derivative-free optimization (DFO). Traditional DFO methods can be classified into direct search-based methods and model-based methods. Both the two types of methods are mostly iterative methods. The difference is that direct search-based methods refine their search directions based on the queried function values directly, while a model-based method builds a model that approximates the function to be optimized and updates the search direction based on the model. Representative methods developed in DFO literature include NOMAD [Le Digabel, 2011, Audet and Dennis Jr, 2006], PSWarm [Vaz and Vicente, 2009], Cobyla [Powell, 1994], and BOBYQA [Powell, 2009]. More comprehensive discussions on DFO methods can be found in Rios and Sahinidis [2013], Audet and Hare [2017].

3.2 Preliminaries: Gradient Estimation via ZO Oracle

The ZO gradient estimate of a function f is constructed by the forward difference of two function values at a random unit direction:

$$\hat{\nabla} f(\mathbf{x}) = (d/\mu)[f(\mathbf{x} + \mu\mathbf{u}) - f(\mathbf{x})]\mathbf{u}, \tag{3.1}$$

where $\mathbf{u}$ is a random vector drawn uniformly from the sphere of a unit ball, and $\mu > 0$ is a small step size, known as the smoothing parameter. In many existing work such as [Nesterov and Spokoiny, 2015, Ghadimi and Lan, 2013], the random direction vector $\mathbf{u}$ was drawn from the standard Gaussian distribution. Here the use of uniform distribution ensures that the ZO gradient estimate (3.1) is defined in a bounded space rather than the whole real space required for Gaussian. As will be evident later, the boundedness of random gradient estimates is one of important conditions in the convergence analysis of ZO-AdaMM.

The rationale behind the ZO gradient estimate (3.1) is that although it is a biased approximation to the true gradient of f, it is *unbiased* to the gradient of the randomized smoothing version of f with parameter μ [Duchi et al., 2015, Liu et al., 2018c, Gao et al., 2014], i.e.,

$$f_\mu(\mathbf{x}) = \mathbb{E}_{\mathbf{u} \sim U_{\mathrm{B}}}[f(\mathbf{x} + \mu\mathbf{u})], \tag{3.2}$$

where $\mathbf{u} \sim U_{\mathrm{B}}$ denotes the uniform distribution over the unit Euclidean ball B. We review properties of the smoothing function (3.2) and connections to the ZO gradient estimator (3.1) in Section 3.7.1.

3.3 AdaMM from First to Zeroth Order

Consider a stochastic optimization problem of the generic form

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) = \mathbb{E}_{\boldsymbol{\xi}}[f(\mathbf{x}; \boldsymbol{\xi})], \tag{3.3}$$

where $\mathbf{x} \in \mathbb{R}^d$ are optimization variables, $\mathcal{X}$ is a closed convex set, f is a differentiable (possibly nonconvex) objective function, and $\boldsymbol{\xi}$ is a certain random variable that captures environmental uncertainties. In problem (3.3), if $\boldsymbol{\xi}$ obeys a uniform distribution built on

empirical samples $\{\boldsymbol{\xi}_i\}_{i=1}^n$, then we recover a finite-sum formulation with the objective function $f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n f(\mathbf{x};\boldsymbol{\xi}_i)$.

First-order AdaMM in terms of AMSGrad [Reddi et al., 2018]. We specify the algorithmic framework of AdaMM by AMSGrad [Reddi et al., 2018], a modified version of Adam [Kingma and Ba, 2014] with convergence guarantees for nonconvex optimization. In the algorithm, the descent direction $\mathbf{m}_t$ is given by an exponential moving average of the past gradients. The learning rate r_t is adaptively penalized by a square root of exponential moving averages of squared past gradients. It has been proved in [Reddi et al., 2018, Chen et al., 2019a, Zhou et al., 2018, Phuong and Phong, 2019] that AdaMM can reach $O(1/\sqrt{T})$[1] convergence rate. Here we omit its possible dependency on d for simplicity, but more accurate analysis will be provided later in Section 3.4.

Algorithm 4 ZO-AdaMM

Input: $\mathbf{x}_1 \in \mathcal{X}$, step sizes $\{\alpha_t\}_{t=1}^T$, $\beta_{1,t}, \beta_2 \in (0,1]$, and set $\mathbf{m}_0$, $\mathbf{v}_0$ and $\hat{\mathbf{v}}_0$
for $t = 1, 2, \ldots, T$ **do**
  let $\hat{\mathbf{g}}_t = \hat{\nabla} f_t(\mathbf{x}_t)$ by (3.1), $f_t(\mathbf{x}_t) := f(\mathbf{x}_t;\boldsymbol{\xi}_t)$
  $\mathbf{m}_t = \beta_{1,t}\mathbf{m}_{t-1} + (1-\beta_{1,t})\hat{\mathbf{g}}_t$
  $\mathbf{v}_t = \beta_2\mathbf{v}_{t-1} + (1-\beta_2)\hat{\mathbf{g}}_t^2$
  $\hat{\mathbf{v}}_t = \max(\hat{\mathbf{v}}_{t-1}, \mathbf{v}_t)$, and $\hat{\mathbf{V}}_t = \mathrm{diag}(\hat{\mathbf{v}}_t)$
  $\mathbf{x}_{t+1} = \Pi_{\mathcal{X},\sqrt{\hat{\mathbf{v}}_t}}(\mathbf{x}_t - \alpha_t \hat{\mathbf{V}}_t^{-1/2}\mathbf{m}_t)$
end for

ZO-AdaMM. By integrating AdaMM with the random gradient estimator (3.1), we obtain ZO-AdaMM in Algorithm 4. Here the square root, the square, the maximum, and the division operators are taken elementwise. Also, $\Pi_{\mathcal{X},\mathbf{H}}(\mathbf{a})$ denotes the projection operation under Mahalanobis distance with respect to $\mathbf{H}$, i.e., $\mathrm{argmin}_{\mathbf{x}\in\mathcal{X}} \|\sqrt{\mathbf{H}}(\mathbf{x}-\mathbf{a})\|_2^2$. If $\mathcal{X} = \mathbb{R}^d$, the projection step simplifies to $\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha_t\hat{\mathbf{V}}_t^{-1/2}\mathbf{m}_t$. Clearly, $\alpha_t\hat{\mathbf{V}}_t^{-1/2}$ and $\mathbf{m}_t$ can be interpreted as the adaptive learning rate and the momentum-type descent direction, which adopt exponential moving averages as follows,

$$\mathbf{m}_t = \sum_{j=1}^t \left[\left(\prod_{k=1}^{t-j}\beta_{1,t-k+1}\right)(1-\beta_{1,j})\hat{\mathbf{g}}_j\right], \ \mathbf{v}_t = (1-\beta_2)\sum_{j=1}^t(\beta_2^{t-j}\hat{\mathbf{g}}_j^2). \tag{3.4}$$

Here we assume that $\mathbf{m}_0 = \mathbf{0}$, $\mathbf{v}_0 = \mathbf{0}$ and $0^0 = 1$ by convention, and let $\hat{\mathbf{g}}_t = \hat{\nabla} f_t(\mathbf{x}_t)$ by

[1] In the paper, we could omit $\log(T)$ in Big O notation.

(3.1) with $f_t(\mathbf{x}_t) := f(\mathbf{x}_t; \boldsymbol{\xi}_t)$.

Motivation and rationale behind ZO-AdaMM. First, gradient normalization helps noise reduction in ZO optimization as shown by Ilyas et al. [2018a], Liu et al. [2019]. In the similar spirit, ZO-AdaMM also normalizes the descent direction $\mathbf{m}_t$ by $\sqrt{\hat{\mathbf{v}}_t}$. In the extreme case of $\beta_{1,t} = \beta_2 \to 0$ and $\hat{\mathbf{v}}_t = \mathbf{v}_t$, ZO-AdaMM could reduce to ZO-signSGD Liu et al. [2019] (ignoring the max operation in line 6) since $\hat{\mathbf{V}}_t^{-1/2}\mathbf{m}_t = \mathbf{m}_t/\sqrt{\mathbf{v}_t} = \hat{\mathbf{g}}_t/\sqrt{\hat{\mathbf{g}}_t^2} = \text{sign}(\hat{\mathbf{g}}_t)$ known from (3.4). However, the downside of ZO-signSGD is its worse convergence accuracy than ZO-SGD, i.e., it only converges to a neighborhood of a stationary point even for unconstrained optimization. Compared to ZO-signSGD, ZO-AdaMM is able to cover ZO-SGD as a special case when $\beta_{1,t} = 0$, $\beta_2 = 1$, $\mathbf{v}_0 = \mathbf{1}$ and $\hat{\mathbf{v}}_0 \leq \mathbf{1}$ from Algorithm 1. Thus, we hope that with appropriate choices of $\beta_{1,t}$ and β_2, ZO-AdaMM could enjoy dual advantages of ZO-signSGD and ZO-SGD. Another motivation comes from the possible presence of time-dependent gradient priors [Ilyas et al., 2018c]. Given this, the use of past gradients in momentum also helps noise reduction.

Why is ZO-AdaMM difficult to analyze? The convergence analysis of ZO-AdaMM becomes significantly more challenging than existing ZO methods due to the involved coupling among stochastic sampling, ZO gradinet estimation, momentum, adaptive learning rate, and projection operation. In particular, the use of Mahalanobis distance in projection step plays a key role on convergence guarantees. And the conventional variance bound on ZO gradient estimates is insufficient to analyze the convergence of ZO-AdaMM due to the use of adaptive learning rate. In the next sections, we will carefully study the convergence of ZO-AdaMM under different settings.

3.4 Convergence Analysis of ZO-AdaMM

In this section, we begin by providing a deep understanding on the importance of Mahalanobis distance used in ZO-AdaMM (Algorithm 4), and then introduce the Mahalanobis distance based convergence analysis for both unconstrained and constrained nonconvex optimization. Our analysis makes the first step toward understanding adaptive learning rate methods for nonconvex constrained optimization. Throughout the section, we make

the following assumptions.

A2.1: $f_t(\cdot) := f(\cdot; \boldsymbol{\xi}_t)$ has L_g-Lipschitz continuous gradient, where $L_g > 0$.

A2.2: f_t has η-bounded stochastic gradient $\|\nabla f_t(\mathbf{x})\|_\infty \leq \eta$.

3.4.1 Importance of Mahalanobis distance based projection operation

Recall from Algorithm 4 that ZO-AdaMM takes the projection operation $\Pi_{\mathcal{X}, \sqrt{\hat{\mathbf{V}}_t}}(\cdot)$ onto the constraint set $\mathcal{X}$ under Mahalanobis distance with respect to (w.r.t.) $\hat{\mathbf{V}}_t$. In some recent adversarial learning algorithms Kurakin et al. [2016], Ilyas et al. [2018b], the Euclidean projection $\Pi_{\mathcal{X}}(\cdot)$ was used in both FO and ZO AdaMM-type methods rather than the Mahalanobis distance based projection in Algorithm 4. However, such an implementation could lead to *non-convergence*: Proposition 1 shows the non-convergence issue of Algorithm 4 using the Euclidean projection operation when solving a simple linear program subject to ℓ_1-norm constraint. This is an important point which is ignored in design of many algorithms on adversarial training Madry et al. [2017].

Proposition 1. *Consider the following problem*

$$\underset{\mathbf{x}=[x_1,x_2]^T}{\text{minimize}} \; -2x_1 - x_2; \quad \textit{subject to} \;\; |x_1 + x_2| \leq 1, \tag{3.5}$$

then Algorithm 4, initialized by $\mathbf{x} = [0.5, 0.5]^T$*, using the Euclidean projection* $\Pi_{\mathcal{X}}(\cdot)$ *converges to a fixed point* $[0.5, 0.5]^T$ *rather than a stationary point of* (3.5).

Proof: *The proof investigates a special case of Algorithm 4, projected signSGD; See Section 3.7.2.*

Proposition 1 indicates that replacing the Mahalanobis distance based projection in Algorithm 4 with Euclidean projection will lead to a divergent algorithm, highlighting the importance of using Mahalanobis distance. However, the use of Mahalanobis distance based projection complicates the convergence analysis, especially in constrained optimization. Accordingly, we define a Mahalanobis based convergence measure that can simplify the analysis and can be converted into the traditional convergence measure.

Let $\mathbf{x}^+ = \mathbf{x}_{t+1}$, $\mathbf{x}^- = \mathbf{x}_t$, $\mathbf{g} = \mathbf{m}_t$, $\omega = \alpha_t$ and $\mathbf{H} = \hat{\mathbf{V}}_t^{1/2}$, the projection step of Algorithm 4 can be written in the generic form

$$\mathbf{x}^+ = \underset{\mathbf{x} \in \mathcal{X}}{\text{argmin}} \{\langle \mathbf{g}, \mathbf{x} \rangle + (1/\omega) D_{\mathbf{H}}(\mathbf{x}, \mathbf{x}^-)\}, \tag{3.6}$$

where $D_{\mathbf{H}}(\mathbf{x}, \mathbf{x}^-) = \|\mathbf{H}^{1/2}(\mathbf{x} - \mathbf{x}^-)\|^2/2$ gives the Mahalanobis distance w.r.t. $\mathbf{H}$, and $\|\cdot\|$ denotes ℓ_2 norm. Based on (3.6), the concept of *gradient mapping* Ghadimi et al. [2016] is given by

$$P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega) := (\mathbf{x}^- - \mathbf{x}^+)/\omega. \tag{3.7}$$

The gradient mapping $P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega)$ yields a natural interpretation: a projected version of $\mathbf{g}$ at the point $\mathbf{x}^-$ given the learning rate ω, yielding $\mathbf{x}^+ = \mathbf{x}^- - \omega P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^{-1}, \mathbf{g}, \omega)$. We note that different from Ghadimi et al. [2016], Reddi et al. [2016b], the gradient mapping in (3.7) is defined on the projection under the Mahalanobis distance $D_{\mathbf{H}}(\cdot, \cdot)$ rather than the Euclidean distance.

With the aid of (3.7), we propose the Mahalanobis distance based convergence measure for ZO-AdaMM:

$$\|\mathcal{G}(\mathbf{x}_t)\|^2 := \|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f(\mathbf{x}_t), \alpha_t)\|^2. \tag{3.8}$$

If $\mathcal{X} = \mathbb{R}^d$, then the convergence measure (3.8) reduces to

$$\|\hat{\mathbf{V}}_t^{-1/4} \nabla f(\mathbf{x}_t)\|^2, \tag{3.9}$$

which corresponds to the squared Euclidean norm of gradient in a linearly transformed coordinate system $\mathbf{y}_t = \hat{\mathbf{V}}_t^{1/4}\mathbf{x}_t$. As will be evident later, the measure (3.9) can be transformed to the conventional measure $\|\nabla f(\mathbf{x}_t)\|^2$ for unconstrained optimization.

We remark that Mahalanobis (M-) distance facilitates our convergence analysis in an equivalently transformed space, over which the analysis can be generalized from the conventional projected gradient descent framework. To get intuition, let us consider a simpler first-order case with the $\mathbf{x}$-descent step given by Algorithm 4 as $\beta_{1,t} = 0$ and $\mathcal{X} = \mathbb{R}^d$: $\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha \hat{\mathbf{V}}_t^{-1/2} \nabla f(\mathbf{x}_t)$. Note that the ZO case is more involved but follows the same intuition. Upon defining $\mathbf{y}_t \triangleq \hat{\mathbf{V}}_t^{1/4}\mathbf{x}_t$, the $\mathbf{x}$-update can then be rewritten as the update rule in $\mathbf{y}$: $\mathbf{y}_{t+1} = \mathbf{y}_t - \alpha \hat{\mathbf{V}}_t^{-1/4} \nabla f(\mathbf{x}_t)$. Since $\nabla_{\mathbf{y}_t} f(\mathbf{x}_t) = (\frac{\partial \mathbf{x}_t}{\partial \mathbf{y}_t})^T \nabla f(\mathbf{x}_t) = \hat{\mathbf{V}}_t^{-1/4} \nabla f(\mathbf{x}_t)$, the $\mathbf{y}$-update, $\mathbf{y}_{t+1} = \mathbf{y}_t - \alpha \nabla_{\mathbf{y}} f(\mathbf{x}_t)$, obeys the gradient descent framework. In the constrained case, a similar but more involved analysis can be made, showing that the *M-projection in the* $\mathbf{x}$*-coordinate system* is *equivalent* to the *Euclidean projection in the* $\mathbf{y}$*-coordinate system* which makes projected gradient descent applicable to the update in $\mathbf{y}$. By contrast, the direct use of *Euclidean projection in the* $\mathbf{x}$*-coordinate system* leads to *divergence* in ZO-AdaMM (Proposition 1).

3.4.2 Unconstrained nonconvex optimization

We next demonstrate the convergence analysis of ZO-AdaMM for unconstrained nonconvex optimization. In Proposition 2, we begin by exploring the relationship between the convergence measure (3.9) and ZO gradient estimates; *See Section 3.7.2 for proof.*

Proposition 2. *Suppose that* ***A2.1-A2.2*** *hold and let* $\mathcal{X} = \mathbb{R}^d$, $\hat{\mathbf{v}}_0^{1/2} \geq c\mathbf{1}$, $f_\mu(\mathbf{x}_1) - \min_{\mathbf{x}} f_\mu(\mathbf{x}) \leq D_f$, $\beta_{1,t} = \beta_1$, $\gamma := \beta_1/\beta_2 < 1$, $\mu = 1/\sqrt{Td}$, *and* $\alpha_t = 1/\sqrt{Td}$ *in Algorithm 4, then ZO-AdaMM yields*

$$\mathbb{E}\left[\left\|\hat{\mathbf{V}}_R^{-1/4}\nabla f(\mathbf{x}_R)\right\|^2\right] \leq \frac{L_g^2}{2c}\frac{d}{T} + 2D_f\frac{\sqrt{d}}{\sqrt{T}} + \frac{L_g(4+5\beta_1^2)(1-\beta_1)}{2(1-\beta_1)^2(1-\beta_2)(1-\gamma)}\frac{\sqrt{d}}{\sqrt{T}} + \frac{2}{c}\mathbb{E}\left[2\eta^2 + \frac{\eta\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\}}{1-\beta_1}\right]\frac{d}{T}, \tag{3.10}$$

where $\mathbf{x}_R$ *is picked uniformly randomly from* $\{\mathbf{x}_t\}_{t=1}^T$, *and* $\hat{\mathbf{g}}_t = \hat{\nabla} f_t(\mathbf{x}_t)$ *by* (3.1).

Proposition 2 implies that the convergence rate of ZO-AdaMM has a dependency on ZO gradient estimates in terms of $G_{\text{zo}} := \max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\}$. Moreover, if we consider the first order AdaMM in Chapter 2 in which the ZO gradient estimate $\hat{\mathbf{g}}_t$ is replaced with the stochastic gradient, then one can simply assume $\max_{t\in[T]}\{\|\mathbf{g}_t\|_\infty\}$ to be a dimension-independent constant under **A2.2**. However, in the ZO setting, G_{zo} is no longer independent of d. For example, it could be directly bounded by $\|\hat{\nabla} f(\mathbf{x})\|_2 \leq (d/\mu)\|f(\mathbf{x}+\mu\mathbf{u}) - f(\mathbf{x})\|_2 \leq dL_c$ under the following assumption:

A2.3: f_t is L_c-Lipschitz continuous.

In Proposition 3, we show that the dimension-dependency of G_{zo} can be further improved by using sphere concentration results; *See Section 3.7.2 for proof.*

Proposition 3. *Under* ***A2.3***, $\max\{d, T\} \geq 3$, *and given* $\delta \in (0,1)$, *then with probability at least* $1-\delta$,

$$\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\} \leq 2L_c\sqrt{d\log(dT/\delta)}. \tag{3.11}$$

Here we provide some insights on Proposition 3. Since the unit random vector used to define $\hat{\mathbf{g}}_t$ is uniformly sampled on a sphere, $\|\hat{\mathbf{g}}_t\|_\infty$ can be improved to $O(\sqrt{d})$ with high probability. This is a tight bound since when the function difference is a constant, the lower bound satisfies $\|\hat{\mathbf{g}}_t\|_\infty = \Omega(\sqrt{d})$ by sphere concentration. It is also

not surprising that our bound (3.11) grows with T since we bound the maximum $\|\hat{\mathbf{g}}_t\|_\infty$ over T realizations with high probability. The time-dependence is required to compensate the growth of the probability that there exists an estimate with the extreme ℓ_∞ value versus time. Note that as long as T has polynomial rather than exponential dependency on d, we then always have $\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\} = O(\sqrt{d\log(d)})$. Based on Proposition 2 and Proposition 3, the convergence rate of ZO-AdaMM is provided by Theorem 2. See Section 3.7.2 for proof.

Theorem 2. *Suppose that **A2.1** and **A2.3** hold. Given parameter settings in Proposition 2 and 3, then with probability at least* $1 - 1/(T\sqrt{d})$*, ZO-AdaMM yields*

$$\mathbb{E}\left[\left\|\hat{\mathbf{V}}_R^{-1/4}\nabla f(\mathbf{x}_R)\right\|^2\right] = O\left(\sqrt{d}/\sqrt{T} + d^{1.5}/T\right). \tag{3.12}$$

We can also extend the convergence rate of ZO-AdaMM in Theorem 2 using the measure $\mathbb{E}[\|\nabla f(\mathbf{x}_R)\|^2]$. Since $\hat{V}_{t,ii}^{-1/2} \geq 1/\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\}$ (by the update rule), we obtain from (3.11) that

$$\mathbb{E}\left[\|\nabla f(\mathbf{x}_R)\|^2\right] \leq 2L_c\sqrt{d\log(dT/\delta)}\mathbb{E}\left[\left\|\hat{\mathbf{V}}_R^{-1/4}\nabla f(\mathbf{x}_R)\right\|^2\right]. \tag{3.13}$$

Theorem 2, together with (3.13), implies $O(d/\sqrt{T} + d^2/T)$ convergence rate of ZO-AdaMM under the conventional measure. We remark that compared to the first order rate $O(\sqrt{d}/\sqrt{T} + d/T)$ [Zhou et al., 2018] of AdaMM for unconstrained nonconvex optimization under **A2.1**-**A2.2**, ZO-AdaMM suffers $O(\sqrt{d})$ and $O(d)$ slowdown on the rate term $O(1/\sqrt{T})$ and $O(1/T)$, respectively. This dimension-dependent slowdown is similar to ZO-SGD versus SGD shown by [Ghadimi and Lan, 2013]. We also remark that compared to FO-AdaMM, ZO-AdaMM requires additional **A2.3** to bound the ℓ_∞ norm of ZO gradient estimates.

3.4.3 Constrained nonconvex optimization

To analyze ZO-AdaMM in a general constrained case, one needs to handle the coupling effects from all three factors: momentum, adaptive learning rate, and projection operation. Here we focus on addressing the coupling issue in the last two factors, which yields our results on ZO-AdaMM at $\beta_{1,t} = 0$. This is equivalent to the ZO version of RMSProp Tieleman and Hinton [2012] with Reddi's convergence fix in Reddi et al. [2018]. When

the momentum factor comes into play, the scenario becomes much more complicated. We leave the answer to the general case $\beta_{1,t} \neq 0$ for future research. Even for SGD with momentum, we are not aware of any successful convergence analysis for stochastic constrained nonconvex optimization.

It is known from SGD Ghadimi et al. [2016] that the presence of projection induces a stochastic bias (independent of iteration number T) for constrained nonconvex optimization. In Theorem 3, we show that the same challenge holds for ZO-AdaMM. Thus, one has to adopt the variance reduced gradient estimator, which induces higher querying complexity than the estimator (3.1); *See Section 3.7.2 for proof.*

Theorem 3. *Suppose that* ***A2.1-A2.2*** *hold,* $\hat{\mathbf{v}}_0^{1/2} \geq c\mathbf{1}$, $f_\mu(\mathbf{x}_1) - \min_{\mathbf{x}} f_\mu(\mathbf{x}) \leq D_f$, $\alpha_t = \alpha \leq \frac{c}{L_g}$, $\mu = \frac{1}{\sqrt{Td}}$, *and* $\beta_{1,t} = 0$ *in Algorithm 4, then the convergence rate of ZO-AdaMM under* (3.8) *satisfies*

$$\mathbb{E}[\|\mathcal{G}(\mathbf{x}_R)\|^2] \leq \frac{6D_f}{\alpha T} + \frac{3L_g^2 d}{4cT} + \frac{6\eta^2}{c^4 T}(\max_{t\in[T]} \mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] + d\eta^2) + \frac{3c+9}{c} \max_{t\in[T]} \mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2],$$

where $\mathbf{x}_R$ *is picked uniformly randomly from* $\{\mathbf{x}_t\}_{t=1}^T$, $\mathcal{G}(\mathbf{x})$ *has been defined in* (3.8), *and* f_μ *is the smoothing function of* f *defined in* (3.2).

Theorem 3 implies that regardless of the number of iterations T, ZO-AdaMM only converges to a solution's neighborhood whose size is determined by the variance of ZO gradient estimates $\max_{t\in[T]} \mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2]$. To make this term diminishing, we consider the following variance reduced gradient estimator built on multiple stochastic samples and random direction vectors Liu et al. [2019],

$$\hat{\mathbf{g}}_t = \frac{1}{bq} \sum_{j\in\mathcal{I}_t} \sum_{i=1}^{q} \hat{\nabla} f(\mathbf{x}_t; \mathbf{u}_{i,t}, \boldsymbol{\xi}_j), \quad \hat{\nabla} f(\mathbf{x}_t; \mathbf{u}_{i,t}, \boldsymbol{\xi}_j) := \frac{d[f(\mathbf{x}_t + \mu \mathbf{u}_{i,t}; \boldsymbol{\xi}_j) - f(\mathbf{x}_t; \boldsymbol{\xi}_j)]}{\mu} \mathbf{u}_{i,t}, \quad (3.14)$$

where $\mathcal{I}_t$ is a mini-batch containing b stocahstic samples at time t, and $\{\mathbf{u}_{i,t}\}_{i=1}^q$ are q random direction vectors at time t. We present the variance of (3.14) in Lemma 1, whose proof is induced from [Liu et al., 2019, Proposition 2] by using $\|\nabla f_t\|_2^2 \leq d\|\nabla f_t\|_\infty^2 = d\eta^2$ in **A2.2**.

Lemma 1. *Suppose that* ***A2.1-A2.2*** *hold, then for* $\mu \leq 1/\sqrt{d}$, *the variance of* (3.14)

yields

$$\mathbb{E}\left[\|\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)\|_2^2\right] = O\left(d/b + d^2/q\right). \tag{3.15}$$

Based on Lemma 1, the rate of ZO-AdaMM in Theorem 3 becomes $\mathbb{E}[\|\mathcal{G}(\mathbf{x}_R)\|^2] = O(d/T + d/b + d^2/q)$. Note that if **A2.3** holds, then the dimension-dependency can be improved by $O(d)$ factor based on Lemma 1. To the best of our knowledge, even in the FO case we are not aware of existing convergence rate analysis on adaptive learning rate methods for nonconvex contrained optimization.

Comparison with other ZO methods Since the existing convergence analysis for different ZO methods is built on different problem settings and assumptions. The direct comparison over the convergence rates might not be fair enough. Thus, in Table 3.1 we compare ZO-AdaMM with others ZO methods from 4 perspectives: a) the type of gradient estimator, b) the setting of smoothing parameter μ, c) convergence rate, and d) function query complexity.

Table 3.1 shows that for unconstrained nonconvex optimization, the convergence of ZO-AdaMM achieves worse dependency on d than ZO-SGD Ghadimi and Lan [2013], ZO-SCD Lian et al. [2016] and ZO-signSGD Liu et al. [2019]. However, it has milder choice of μ than ZO-SGD, less query complexity than ZO-SCD, and no T-independent convergence bias compared to ZO-signSGD. Also, for constrained nonconvex optimization, ZO-AdaMM yields the similar rate to ZO-ProxSGD Ghadimi et al. [2016], which also implies ZO projected SGD (ZO-PSGD). We also highlight that at the first glance, ZO-AdaMM has a worse d-dependency (regardless of choice of μ) than ZO-SGD. However, even in the FO setting, AdaMM has an extra $O(\sqrt{d})$ dependency in the worst case due to the effect of (coordinate-wise) gradient normalization when bounding the distance of two consecutive updates. Thus, in addition to comparing with different ZO methods, Table 3.1 also summarizes the convergence performance of FO AdaMM. Note that our rate yields $O(\sqrt{d})$ slowdown compared to FO AdaMM though bounding ZO gradient estimate norm requires stricter assumption.

Table 3.1: Summary of convergence rate and query complexity of various ZO algorithms given T iterations.

Method	Assumptions	Gradient estimator	Smoothing parameter μ	Rate	Query
ZO-SGD Ghadimi and Lan [2013]	NC[1], UCons[1], **A2.1**, **A2.3**[2]	GauGE[1]	$O\left(\frac{1}{d\sqrt{T}}\right)$	$O\left(\frac{\sqrt{d}}{\sqrt{T}}+\frac{d}{T}\right)$	$O\left(T\right)$
ZO-SCD Lian et al. [2016]	NC, UCons, **A2.1**, **A2.3**[2]	CooGE[1]	$O\left(\frac{1}{\sqrt{T}}+\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{\sqrt{d}}{\sqrt{T}}+\frac{d}{T}\right)$	$O\left(dT\right)$
ZO-signSGD Liu et al. [2019]	NC, UCons, **A2.1**, **A2.3**	sign-UniGE[1]	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O(\frac{\sqrt{d}}{\sqrt{T}}+\frac{\sqrt{d}}{\sqrt{b}}+\frac{d}{\sqrt{bq}})$[3]	$O\left(bqT\right)$
ZO-ProxSGD / ZO-PSGD Ghadimi et al. [2016]	NC, Cons[4], **A2.1**, **A2.3**	GauGE	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O\left(\frac{d^2}{qT}+\frac{d}{q}\right)$	$O\left(qT\right)$
ZO-SMD Duchi et al. [2015]	C, Cons, **A2.3**	GauGE/UniGE	$O\left(\frac{1}{dt}\right)$	$O\left(\frac{\sqrt{d}}{\sqrt{T}}\right)$	$O(T)$
AdaMM Chen et al. [2019a], Zhou et al. [2018]	NC, UCons, **A2.1**, **A2.2**	SGE[1]	n/a	$O\left(\frac{\sqrt{d}}{\sqrt{T}}+\frac{d}{T}\right)$	n/a
AdaMM Reddi et al. [2018], Chen and Gu [2018], Phuong and Phong [2019]	C, Cons, **A2.2**	SGE	n/a	$O\left(\frac{d}{\sqrt{T}}\right)$	n/a
ZO-AdaMM	NC, UCons, **A2.1**, **A2.3**	UniGE	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O\left(\frac{d}{\sqrt{T}}+\frac{d^2}{T}\right)$	$O\left(T\right)$
ZO-AdaMM	NC, Cons, **A2.1**, **A2.3** $\beta_{1,t}=0$	UniGE	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O\left(\frac{d}{T}+\frac{1}{b}+\frac{d}{q}\right)$	$O\left(bqT\right)$

[1] *Abbreviations.* NC: Nonconvex; UCons: Unconstrained; GauGE: Gaussian random vector based gradient estimate; UniGE: Uniform random vector based gradient estimate; CooGE: Coordinate-wise gradient estimate; SGE: stochastic (first-order) gradient estimate

[2] Assumption of bounded variance of stochastic gradients is implied from **A2.3**. [3] Convergence of ZO-signSGD is measured by $\mathbb{E}[\|\nabla f(\mathbf{x}_T)\|_2]$ rather than its square used in other algorithms for nonconvex optimization.

3.5 Applications to Black-Box Adversarial Attacks

In this section, we demonstrate the effectiveness of ZO-AdaMM by experiments on generating black-box adversarial examples. Our experiments will be performed on Inception V3 Szegedy et al. [2016] using ImageNet Deng et al. [2009]. Here we focus on two types of black-box adversarial attacks: *per-image* adversarial perturbation Xu et al. [2019] and *universal* adversarial perturbation against multiple images Chen et al. [2017], Ilyas et al. [2018a], Suya et al. [2017], Cheng et al. [2018]. For each type of attack, we allow both constrained and unconstrained optimization problem settings. We compare our propos ed ZO-AdaMM method with 6 existing ZO algorithms: ZO-SGD, ZO-SCD and ZO-signSGD for unconstrained optimization, and ZO-PSGD, ZO-SMD and ZO-NES for constrained optimization. The first 5 methods have been summarized in Table 3.1, and ZO-NES refers to the black-box attack generation method in Ilyas et al. [2018a], which applies a projected version of ZO-signSGD using natural evolution strategy (NES) based random gradient estimator. In our experiments, every method takes the same number of queries per iteration. Accordingly, the total query complexity is consistent with the number of iterations.

3.5.1 Experiment setup

It is known that DNN-based image classifiers are vulnerable to adversarial examples—one can carefully craft images with imperceptible perturbations (a.k.a. adversarial

perturbations or adversarial attacks) that can fool image classifiers even under a *black box* threat model, where details of the model are unknown to the attacker Chen et al. [2017], Ilyas et al. [2018a], Suya et al. [2017], Cheng et al. [2018].

We focus on two problem settings of black-box adversarial attacks: per-image adversarial perturbation and universal adversarial perturbation. Let $(\mathbf{x}, t)$ denote a legitimate image $\mathbf{x}$ with the true label $t \in \{1, 2, \ldots, K\}$, where K is the total number of image classes. And let $\mathbf{x}' = \mathbf{x} + \boldsymbol{\delta}$ denote an adversarial example, where $\boldsymbol{\delta}$ is the adversarial perturbation. Our goal is to design $\boldsymbol{\delta}$ for a single image $\mathbf{x}$ or multiple images $\{\mathbf{x}_i\}_{i=1}^M$. Spurred by Carlini and Wagner [2017], we consider the optimization problem

$$\begin{array}{ll} \underset{\boldsymbol{\delta}}{\text{minimize}} & \frac{\lambda}{M}\sum_{i=1}^M f(\mathbf{x}_i + \boldsymbol{\delta}) + \|\boldsymbol{\delta}\|_2^2 \\ \text{subject to} & (\mathbf{x}_i + \boldsymbol{\delta}) \in [-0.5, 0.5]^d, \forall i, \end{array} \tag{3.16}$$

where $f(\mathbf{x}_0 + \boldsymbol{\delta})$ denotes the (black-box) attack loss function, $\lambda > 0$ is a regularization parameter that strikes a balance between minimizing the attack loss and the ℓ_2 distortion, and we normalize the pixel values to $[-0.5, 0.5]^d$. In problem (3.16), we specify the loss function for untargeted attack Carlini and Wagner [2017], $f(\mathbf{x}') = \max\{Z(\mathbf{x}')_t - \max_{j \neq t} Z(\mathbf{x}')_j, -\kappa\}$, where $Z(\mathbf{x}')_k$ denotes the prediction score of class k given the input $\mathbf{x}'$, and the parameter $\kappa > 0$ governs the gap between the confidence of the predicted label and the true label t. In experiments, we choose $\kappa = 0$, and the attack loss f reaches the minimum value 0 as the perturbation succeeds to fool the neural network.

In problem (3.16), if $M = 1$, then it becomes our first task to find per-image adversarial perturbations. If $M > 1$, then the problem corresponds to the task of finding universarial adversarial perturbations to M images. Problem (3.16) yields a constrained formulation for the design of black-box adversarial attacks. Since some ZO algorithms are designed only for unconstrained optimization (see Table 3.1), we also consider the unconstrained version of problem (3.16) Liu et al. [2018c],

$$\begin{array}{ll} \underset{\mathbf{w} \in \mathbb{R}^d}{\text{minimize}} & \frac{\lambda}{M}\sum_{i=1}^M \left[f\left(0.5\tanh(\tanh^{-1}(2\mathbf{x}_i) + \mathbf{w})\right) \right. \\ & \left. + \|0.5\tanh(\tanh^{-1}(2\mathbf{x}_i) + \mathbf{w}) - \mathbf{x}_i\|_2^2 \right], \end{array} \tag{3.17}$$

where $\mathbf{w} \in \mathbb{R}^d$ are optimization variables, and we eliminate the inequality constraint in (3.16) by leveraging $0.5\tanh(\tanh^{-1}(2\mathbf{x}_i) + \mathbf{w}) = \mathbf{x}_i + \boldsymbol{\delta} \in [-0.5, 0.5]^d$.

The experiments of generating black-box adversarial examples will be performed on Inception V3 Szegedy et al. [2016] under the dataset ImageNet Deng et al. [2009]. We

will compare the proposed ZO-AdaMM method with 6 existing ZO algorithms, ZO-SGD Ghadimi and Lan [2013], ZO-SCD Lian et al. [2016] and ZO-signSGD Liu et al. [2019] for unconstrained optimization, and ZO-PSGD Ghadimi et al. [2016], ZO-SMD Duchi et al. [2015] and ZO-NES Ilyas et al. [2018a] for constrained optimization. The first 5 methods have been summarized in Table 3.1, and ZO-NES refers to the black-box attack generation method in Ilyas et al. [2018a], which applies a projected version of ZO-signSGD using natural evolution strategy (NES) based random gradient estimator. In all the aforementioned ZO algorithms, we adopt the random gradient estimator (3.14) and set $b = 1$ and $q = 10$ so that every method takes the same query cost per iteration. Accordingly, the total query complexity is consistent with the number of iterations.

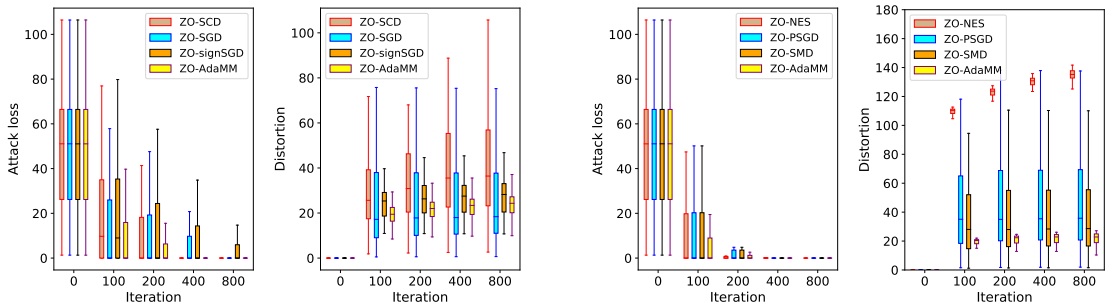
Per-image adversarial perturbation In Fig. 3.1, we present the attack loss and the resulting ℓ_2-distortion against iteration numbers for solving both unconstrained and constrained adversarial attack problems, namely, (3.17) and (3.16) ($M = 1$ and $\lambda = 10$), over 100 randomly selected images. In ZO-AdaMM, we set $\mathbf{v}_0 = \hat{\mathbf{v}}_0 = 10^{-5}$, $\mathbf{m}_0 = \mathbf{0}$, $\beta_{1t} = \beta_1 = 0.9$, $\beta_2 = 0.3$ and $T = 1000$. Here the exponential moving average parameters (β_1, β_2) are exhaustively searched over $\{01, 0.3, 0.5, 0.7, 0.9\}^2$. In ZO-AdaMM, we also choose a decaying learning rate $\alpha_t = \alpha/\sqrt{t}$ with $\alpha = 0.01$. For fair comparison, we use the decaying strategy for all other ZO algorithms, and we determine the best choice of α by greedy search over the interval $[10^{-4}, 10^{-2}]$.

In this set of experiments, every algorithm is initialized by zero perturbation. Thus, as the iteration increases, the attack loss decreases until it converges to 0 (indicating successful attack) while the distortion could increase. At this sense, the best attack performance should correspond to the best tradeoff between the fast convergence to 0 attack loss and the low distortion power (evaluated by ℓ_2 norm). As we can see, ZO-AdaMM consistently outperforms other ZO methods in terms of the fast convergence of attack loss and relatively small perturbation. We also note that ZO-signSGD and ZO-NES have poor convergence accuracy in terms of either large attack loss or large distortion at final iterations. This is not surprising, since it has been shown in Liu et al. [2019] that ZO-signSGD only converges to a neighborhood of a solution, and ZO-NES can be regarded as a Euclidean projection based ZO-signSGD, which could induce convergence issues shown by Prop. 1. More detailed statistics are shown in Table

3.2.

Problem	Methods	ASR	Ave. iters (1st succ.)	$\|\boldsymbol{\delta}_t\|_2^2$ (1st succ.)	Final $\|\boldsymbol{\delta}_T\|_2^2$
(3.17)	ZO-SCD	78%	240	57.88	57.51
	ZO-SGD	78%	**159**	38.36	37.85
	ZO-signSGD	74%	179	**23.00**	28.52
	ZO-AdaMM	**81**%	173	28.58	**28.20**
(3.16)	ZO-NES	**82**%	229	82.78	84.41
	ZO-PSGD	78%	**112**	60.32	58.10
	ZO-SMD	76%	198	35.08	35.05
	ZO-AdaMM	78%	197	**23.77**	**23.72**

Table 3.2: Performance of per-image attack over 100 images under $T = 1000$ iterations, where ASR represents attack success rate, and the distortion $\|\boldsymbol{\delta}\|_2^2$ is averaged over successful attacks only.



(a) unconstrained setting (b) constrained setting

Figure 3.1: The attack loss and adversarial distortion v.s. iterations. Each box represents results from 100 images.

Universal adversarial perturbation We now focus on designing a universal adversarial perturbation using the constrained attack problem formulation. Here we attack $M = 100$ random selected images from ImageNet. In Fig. 3.2, we present the attack loss as well as the ℓ_2 norm of universal perturbation at different iteration numbers. As we can see, compared with the other ZO algorithms, ZO-AdaMM has the fastest convergence speed to reach the smallest adversarial perturbation (namely, strongest universal attack). Moreover, in Table 3.3 we present detailed attack success rate and ℓ_2 distortion over $T = 40000$ iterations. Consistent with Fig. 3.2, ZO-AdaMM achieves highest success rate with lowest distortion.

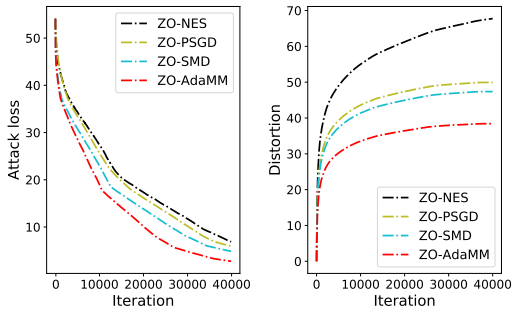


Figure 3.2: Attack loss and distortion of universal attack.

Methods	Attack success rate	Final $\|\boldsymbol{\delta}_T\|_2^2$
ZO-NES	74%	67.74
ZO-PSGD	78%	49.92
ZO-SMD	79%	47.36
ZO-AdaMM	**84%**	**38.40**

Table 3.3: Summary of attack success rate and eventual ℓ_2 distortion for universal attack against 100 images under $T = 40000$ iterations.

3.6 Conclusion

In this paper, we propose ZO-AdaMM, the first effort to integrate adaptive momentum methods with ZO optimization. In theory, we show that ZO-AdaMM has convergence guarantees for nonconvex constrained optimization. Compared with (first-order) AdaMM, it suffers a slowdown factor of $O(\sqrt{d})$. Particularly, we establish a new Mahalanobis distance based convergence measure whose necessity and importance are provided in characterizing the convergence behavior of ZO-AdaMM on nonconvex constrained problems. To demonstrate the utility of the algorithm, we show the superior performance of ZO-AdaMM for designing adversarial examples from black-box neural networks. Compared with 6 state-of-the-art ZO methods, ZO-AdaMM has the fastest empirical convergence to strong black-box adversarial attacks that require the minimum distortion strength.

3.7 Delayed Results and Proofs

3.7.1 Smoothing Function and Random Gradient Estimate

Lemma 2. *a) Relationship between f_μ and f: If f is convex, then f_μ is convex. If f is L_c-Lipschitz continuous, then f_μ is L_c-Lipschitz continuous. Moreover for any $\mathbf{x} \in \mathbb{R}^d$,*

$$|f_\mu(\mathbf{x}) - f(\mathbf{x})| \leq L_c \mu. \tag{3.18}$$

If f has L_g-Lipschitz continuous gradient, then f_μ has L_g-Lipschitz continuous gradient. Moreover for any $\mathbf{x} \in \mathbb{R}^d$,

$$|f_\mu(\mathbf{x}) - f(\mathbf{x})| \leq L_g \mu^2/2 \tag{3.19}$$

$$\|\nabla f_\mu(\mathbf{x}) - \nabla f(\mathbf{x})\|_2^2 \leq \mu^2 d^2 L_g^2/4. \tag{3.20}$$

b) Statistical properties of $\hat{\nabla} f$: For any $\mathbf{x} \in \mathbb{R}^d$,

$$\mathbb{E}_{\mathbf{u}}\left[\hat{\nabla} f(\mathbf{x})\right] = \nabla f_\mu(\mathbf{x}). \tag{3.21}$$

If f has L_g-Lipschitz continuous gradient, then

$$\mathbb{E}_{\mathbf{u}}\left[\|\hat{\nabla} f(\mathbf{x})\|_2^2\right] \leq 2d\|\nabla f(\mathbf{x})\|_2^2 + \mu^2 L_g^2 d^2/2. \tag{3.22}$$

Proof: We refer readers to [Gao et al., 2014, Lemma 4.1] for the detailed proof of a)-b) except the Lipschitz continuity of f_μ and (3.18). Suppose that f is L_c-Lipschitz continuous, based on the definition of f_μ in (3.2), we obtain

$$|f_\mu(\mathbf{x}) - f_\mu(\mathbf{y})| \leq \frac{1}{\alpha(d)} \int_B |f(\mathbf{x} + \mu\mathbf{u}) - f(\mathbf{y} + \mu\mathbf{u})| d\mathbf{u} \leq L_c \|\mathbf{x} - \mathbf{y}\|_2,$$

where $\alpha(d)$ denotes the volume of the unit ball B in $\mathbb{R}^d$.

Moreover, we prove (3.18) as below.

$$|f_\mu(\mathbf{x}) - f(\mathbf{x})| = \left|\frac{1}{\alpha(d)} \int_B f(\mathbf{x} + \mu\mathbf{u}) - f(\mathbf{x}) d\mathbf{u}\right| \leq \frac{\mu L_c}{\alpha(d)} \int_B \|\mathbf{u}\|_2 d\mathbf{u} = \frac{\mu L_c d}{d+1} \leq \mu L_c,$$

where the first equality holds due to (3.2), Jensen's inequality and Lipschitz continuity

of f, and the last equality holds since $(1/\alpha(d))\int_B \|\mathbf{u}\|_2^p d\mathbf{u} = \frac{n}{n+p}$ [Gao et al., 2014, Lemma 6.3.a]. **Q.E.D.**

In Lemma 2, it is clear from (3.20) and (3.21) that the ZO gradient estimate (3.1) becomes unbiased to the true gradient ∇f only when $\mu \to 0$. However, if μ is too small, then the difference of empirical function values is also too small to represent the function differential Lian et al. [2016], Liu et al. [2018c]. Thus, the tolerance on the smoothing parameter μ is an important factor to indicate the convergence performance of ZO optimization methods. It is also known from (3.22) that regardless of the value of μ, the variance of the ZO gradient estimate is always proportional to the dimension d. This is one of reasons for the dimension-dependent slowdown in convergence of ZO optimization methods. This also introduces technical difficulties for analyzing the effect of adaptive learning rate on the convergence of ZO-AdaMM in nonconvex optimization.

3.7.2 Proof for convergence analysis

Proof of Proposition 1

Let us consider a special case of Algorithm 4 with the average ZO gradient estimate $\hat{\nabla} f(\mathbf{x}) = \frac{d}{q\mu}\sum_{i=1}^q \{[f(\mathbf{x}+\mu\mathbf{u}_i) - f(\mathbf{x})]\mathbf{u}_i\}$ under $\beta_{1,t} = \beta_2 \to 0$, $\mu \to 0$ and $q \to \infty$. The conditions of $\beta_{1,t} = \beta_2 \to 0$ enables Algorithm 4 to reduce to ZO-signSGD in Liu et al. [2019], and the conditions of $\mu \to 0$ and $q \to \infty$ makes the ZO gradient estimate unbiased to $\nabla f(\mathbf{x})$ and its variance close to 0 [Liu et al., 2019, Proposition 2]. As a result, we obtain $\hat{\mathbf{g}}_t \to \nabla f(\mathbf{x}_t)$, and Algorithm 1 becomes signSGD Bernstein et al. [2018],

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X},\mathbf{I}}(\mathbf{x}_t - \alpha_t \text{sign}(\nabla f(\mathbf{x}_t))) \tag{3.23}$$

where $\text{sign}(x) = 1$ if $x > 0$ and -1 if $x < 0$, and it is taken elementwise for a vector argument.

Let $f(\mathbf{x}) = -2x_1 - x_2$ in (3.5). We then run (3.23) at $x_1 = x_2 = 0.5$, which yields

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}([0.5, 0.5]^T - \alpha_t[-1,-1]^T) = \Pi_X([0.5+\alpha_t, 0.5+\alpha_t]^T) = [0.5, 0.5]^T, \tag{3.24}$$

where $\mathcal{X}$ encodes the constraint $|x_1 + x_2| \le 1$.

It is clear that the updating rule (3.24) will converge to $\mathbf{x} = [0.5, 0.5]^T$ regardless

of the choice of α_t. The remaining question is whether or not it is a stationary point. Recall that a point $\mathbf{x}^*$ is a stationary point if it satisfies the following conditions:

$$\langle \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \geq 0, \, \forall \mathbf{x} \in \mathcal{X}. \tag{3.25}$$

Since the gradient at $[0.5, 0.5]^T$ is $[-2, -1]^T$, and the inequality (3.25) at $\mathbf{x} = [0.6, 0.4]^T \in X$ does *not* hold, given by $\langle [-2, -1]^T, [0.6, 0.4]^T - [0.5, 0.5]^T \rangle = -0.1 < 0$. This implies that $\mathbf{x}^* = [0.5, 0.5]^T$ is *not* a stationary point of problem (3.5).

Next, we apply the Mhalanobis distance $\hat{\mathbf{V}}_t = \text{diag}(\nabla f(\mathbf{x}_t)^2)$ to (3.23),

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}, \hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t - \alpha_t \text{sign}(\nabla f(\mathbf{x}_t))) = \Pi_{\mathcal{X}, \hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t - \alpha_t \hat{\mathbf{V}}_t^{-1/2} \nabla f(\mathbf{x}_t)). \tag{3.26}$$

Similar to (3.23), we then consider the impact of fixed point $\mathbf{x}_{t+1} = \mathbf{x}_t$ on (3.26). By the definition of projection operator, we have

$$\mathbf{x}_t = \underset{\mathbf{x} \in \mathcal{X}}{\text{argmin}} \, \|\hat{\mathbf{V}}^{1/4}(\mathbf{x} - \mathbf{x}_t + \alpha_t \hat{\mathbf{V}}_t^{-1/2} \nabla f(\mathbf{x}_t)))\| \tag{3.27}$$

The optimality condition of (3.27) is given by

$$\langle \hat{\mathbf{V}}^{1/2}(\mathbf{x}_t - \mathbf{x}_t + \alpha_t \hat{\mathbf{V}}_t^{-1/2} \nabla f(\mathbf{x}_t)), \mathbf{x} - \mathbf{x}_t \rangle \geq 0, \, \forall x \in X,$$

which reduces to

$$\langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle \geq 0, \, \forall \mathbf{x} \in X. \tag{3.28}$$

It thus means that $\mathbf{x}_t$ is a stationary point by (3.25). **Q.E.D.**

Proof of Proposition 2

Before proving the main result Proposition 2, we first prove a few auxiliary lemmas.

Lemma 10. *Given $\{\mathbf{x}_t\}$ from Algorithm 4, consider the sequence*

$$\mathbf{z}_t = \mathbf{x}_t + \frac{\beta_1}{1 - \beta_1}(\mathbf{x}_t - \mathbf{x}_{t-1}), \; \forall t \geq 1, \tag{3.29}$$

where let $\mathbf{x}_0 := \mathbf{x}_1$. *Then for* $\beta_{1,t} = \beta_1$ *and* $\mathcal{X} = \mathbb{R}^d$, $\forall t > 1$

$$\begin{aligned}
&\mathbf{z}_{t+1} - \mathbf{z}_t \\
&= -\frac{\beta_1}{1-\beta_1}\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)\mathbf{m}_{t-1} - \alpha_t \hat{\mathbf{V}}_t^{-1/2}\hat{\mathbf{g}}_t
\end{aligned}$$

and

$$\mathbf{z}_2 - \mathbf{z}_1 = -\alpha_1 \hat{\mathbf{g}}_1 / \sqrt{\hat{\mathbf{v}}_1}.$$

Proof of Lemma 10: The proof follows from Lemma 6.1 in Chen et al. [2019a] by setting $\beta_{1,t} = \beta_1$.

Lemma 11. *By ZO-AdaMM update rule, we have*

$$\mathbb{E}[f_\mu(\mathbf{z}_{t+1}) - f_\mu(\mathbf{z}_1)] \leq \sum_{t=1}^{T} E\left[\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle\right] + \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2}\sum_{t=1}^{T} E\left[\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2\right] \tag{3.30}$$

Proof of Lemma 11: By smoothness of function f, we can have

$$\begin{aligned}
&f_\mu(\mathbf{z}_{t+1}) - f_\mu(\mathbf{z}_t) \\
\leq &\langle \nabla f_\mu(\mathbf{z}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{L_g}{2}\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 \\
= &\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{L_g}{2}\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 + \langle \nabla f_\mu(\mathbf{z}_t) - \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle \\
\leq &\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{L_g}{2}\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 + \frac{1}{2}(\frac{1}{L_g}\|\nabla f_\mu(\mathbf{z}_t) - \nabla f_\mu(\mathbf{x}_t)\|^2 + L_g\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2) \\
\leq &\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + L_g\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 + \frac{1}{2}L_g\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
= &\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + L_g\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 + \frac{1}{2}L_g\|\frac{\beta_1}{1-\beta_1}(\mathbf{x}_t - \mathbf{x}_{t-1})\|^2
\end{aligned} \tag{3.31}$$

Further, by (3.29), we have

$$\mathbf{z}_{t+1} - \mathbf{z}_t = \frac{1}{1-\beta_1}(\mathbf{x}_{t+1} - \mathbf{x}_t) + \frac{\beta_1}{1-\beta_1}(\mathbf{x}_t - \mathbf{x}_{t-1})$$

and thus

$$\|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 \leq \frac{2}{(1-\beta_1)^2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \frac{2\beta_1^2}{(1-\beta_1)^2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 \tag{3.32}$$

Substituting (3.32) into (3.31), we get

$$\begin{aligned}
&f_\mu(\mathbf{z}_{t+1}) - f_\mu(\mathbf{z}_t) \\
\leq&\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{2L_g}{(1-\beta_1)^2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \frac{2\beta_1^2 L_g}{(1-\beta_1)^2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 \\
&+ \frac{1}{2}L_g\frac{\beta_1^2}{(1-\beta_1)^2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 \\
=&\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{2L_g}{(1-\beta_1)^2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \frac{5L_g\beta_1^2}{2(1-\beta_1)^2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2
\end{aligned} \tag{3.33}$$

Summing t from 1 to T and take expectation, we get

$$\begin{aligned}
&\mathbb{E}[f_\mu(\mathbf{z}_{t+1}) - f_\mu(\mathbf{z}_1)] \\
\leq&\mathbb{E}\Bigg[\sum_{t=1}^{T}\Bigg(\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle + \frac{2L_g}{(1-\beta_1)^2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \frac{5L_g\beta_1^2}{2(1-\beta_1)^2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2\Bigg)\Bigg] \\
\leq&\sum_{t=1}^{T}\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle\right] + \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2}\sum_{t=1}^{T}\mathbb{E}\left[\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2\right]
\end{aligned}$$

Q.E.D.

Lemma 12. *Assume $\|\hat{g}_t\|_\infty \leq G_{zo}$, $\forall t \in [T]$ and $m_0 = 0$, By ZO-AdaMM update rule, we have*

$$\begin{aligned}
\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle] \leq&\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\sum_{i=1}^{d}\frac{\alpha_1}{\sqrt{\hat{\mathbf{v}}_{0,i}}}\right] \\
&- \sum_{t=1}^{T}\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t\hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t)\rangle\right].
\end{aligned} \tag{3.34}$$

Proof of Lemma 12: By Lemma 10, we have

$$
\begin{aligned}
&\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t \rangle \\
=&\langle \nabla f_\mu(\mathbf{x}_t), -\frac{\beta_1}{1-\beta_1}\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)\mathbf{m}_{t-1} - \alpha_t \hat{\mathbf{V}}_t^{-1/2}\hat{\mathbf{g}}_t \rangle \\
=&\langle \nabla f_\mu(\mathbf{x}_t), -\frac{\beta_1}{1-\beta_1}\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)\mathbf{m}_{t-1} \rangle - \langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\hat{\mathbf{g}}_t \rangle, \qquad (3.35)
\end{aligned}
$$

and

$$
\begin{aligned}
&\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\hat{\mathbf{g}}_t \rangle \\
=&\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t) \rangle + \langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \\
=&\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t) \rangle + \langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1} \hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \\
&+ \langle \nabla f_\mu(\mathbf{x}_t), \left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle. \qquad (3.36)
\end{aligned}
$$

Substitute (3.36) into (3.35), we have

$$
\begin{aligned}
&\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t \rangle \\
\leq&\langle \nabla f_\mu(\mathbf{x}_t), -\frac{\beta_1}{1-\beta_1}\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right) \odot \mathbf{m}_{t-1} \rangle \\
&- \langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t) \rangle - \langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1} \hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \\
&- \langle \nabla f_\mu(\mathbf{x}_t), \left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \\
=&\langle \nabla f_\mu(\mathbf{x}_t), -\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)\frac{\mathbf{m}_t}{1-\beta_1} \rangle \\
&- \langle \nabla f_\mu(\mathbf{x}_t), -\left(\alpha_t \hat{\mathbf{V}}_t^{-1/2} - \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}\right)\nabla f_\mu(\mathbf{x}_t) \rangle \\
&- \langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t) \rangle - \langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1} \hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \\
\leq&(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2)\sum_{i=1}^{d}\left|\frac{\alpha_{t-1}}{\sqrt{\hat{\mathbf{v}}_{t-1,i}}} - \frac{\alpha_t}{\sqrt{\hat{\mathbf{v}}_{t,i}}}\right| \\
&- \langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t) \rangle - \langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1} \hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)) \rangle \qquad (3.37)
\end{aligned}
$$

where the last inequality follows from the assumption that $\hat{\mathbf{V}}_t = \mathrm{diga}(\hat{\mathbf{v}}_t)$, $\|\nabla f_\mu(\mathbf{x}_t)\|_\infty \leq \eta$ and $\|\hat{\mathbf{g}}_t\|_\infty \leq G_{zo}$.

The upper bound on $\|\mathbf{m}_t\|_\infty$ can be proved by a simple induction. Recall that

$\mathbf{m}_t = \beta_{1,t}\mathbf{m}_{t-1} + (1-\beta_{1,t})\hat{\mathbf{g}}_t$, suppose $\|\mathbf{m}_{t-1}\| \leq G_{zo}$, we have

$$\begin{aligned}\|\mathbf{m}_t\|_\infty \leq &(\beta_{1,t} + (1-\beta_{1,t}))\max(\|\hat{\mathbf{g}}_t\|_\infty, \|\mathbf{m}_{t-1}\|_\infty) \\ = &\max(\|\hat{\mathbf{g}}_t\|_\infty, \|\mathbf{m}_{t-1}\|_\infty) \leq G_{zo}. \end{aligned} \tag{3.38}$$

Then since $\mathbf{m}_0 = 0$, we have $\|\mathbf{m}_0\| \leq G_{zo}$, which completes the induction.

Sum t from 1 to T and take expectation over randomness of $\hat{\mathbf{g}}_t$, we have

$$\begin{aligned}&\sum_{t=1}^{T}\mathbb{E}[\langle\nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t\rangle] \\ \leq&\mathbb{E}\left[\sum_{t=1}^{T}\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\sum_{i=1}^{d}\left|\frac{\alpha_{t-1}}{\sqrt{\hat{\mathbf{v}}_{t-1,i}}} - \frac{\alpha_t}{\sqrt{\hat{\mathbf{v}}_{t,i}}}\right|\right] \\ &-\sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla f_\mu(\mathbf{x}_t), \alpha_t\hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t)\rangle\right] - \sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle\right] \\ \leq&\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\sum_{i=1}^{d}\frac{\alpha_1}{\sqrt{\hat{\mathbf{v}}_{0,i}}}\right] - \sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla f_\mu(\mathbf{x}_t), \alpha_t\hat{\mathbf{V}}_t^{-1/2}\nabla f_\mu(\mathbf{x}_t)\rangle\right]\end{aligned}$$

where the last inequality follows from following facts.

1. Since $\hat{\mathbf{v}}_t = \max(\hat{\mathbf{v}}_{t-1}, \mathbf{v}_t)$, we know $\hat{\mathbf{v}}_t$ is non-decreasing. Given the fact that α_t is non-increasing (by our choice), we have $\alpha_{t-1}/\hat{\mathbf{v}}_{t-1,i} - \alpha_t/\hat{\mathbf{v}}_{t,i} \geq 0$. Thus, following inequality holds.

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{d}\left|\frac{\alpha_{t-1}}{\sqrt{\hat{\mathbf{v}}_{t-1,i}}} - \frac{\alpha_t}{\sqrt{\hat{\mathbf{v}}_{t,i}}}\right|\right] = \mathbb{E}\left[\sum_{i=1}^{d}\sum_{t=1}^{T}\left|\frac{\alpha_{t-1}}{\sqrt{\hat{\mathbf{v}}_{t-1,i}}} - \frac{\alpha_t}{\sqrt{\hat{\mathbf{v}}_{t,i}}}\right|\right] \\ =&\mathbb{E}\left[\sum_{i=1}^{d}\sum_{t=1}^{T}\left(\frac{\alpha_{t-1}}{\sqrt{\hat{\mathbf{v}}_{t-1,i}}} - \frac{\alpha_t}{\sqrt{\hat{\mathbf{v}}_{t,i}}}\right)\right] \leq \mathbb{E}\left[\sum_{i=1}^{d}\frac{\alpha_1}{\sqrt{\hat{\mathbf{v}}_{0,i}}}\right]\end{aligned} \tag{3.39}$$

2. We have $\mathbb{E}[\hat{\mathbf{g}}_t|\hat{\mathbf{g}}_{1:t-1}] = \nabla f_\mu(\mathbf{x}_t)$ by the assumption that $\mathbb{E}[\hat{\mathbf{g}}_t] = \nabla f_\mu(\mathbf{x}_t)$ and the noise on $\hat{\mathbf{g}}_t$ is independent of $\hat{\mathbf{g}}_{1:t-1}$. Thus, the following holds

$$\mathbb{E}\left[\langle\nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle\right] = 0 \tag{3.40}$$

Q.E.D.

Lemma 13. *Assume $\gamma := \beta_1/\beta_2 < 1$, ZO-AdaMM yields*

$$\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \leq \alpha_t^2 d \frac{1-\beta_1}{1-\beta_2} \frac{1}{1-\gamma} \tag{3.41}$$

Comment: This is an important lemma for ZO-AdaMM, it shows the squared update quantity is not dependent on size of stochastic gradient, thus giving a tighter dependency on d compared with [Reddi et al., 2018].

Proof of Lemma 13: By the update rule, we have

$$\begin{aligned}
&\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 = \alpha_t^2 \left\| \frac{\mathbf{m}_t}{\sqrt{\hat{\mathbf{v}}_t}} \right\|^2 \\
\leq &\alpha_t^2 \sum_{i=1}^{d} \frac{((1-\beta_1)\sum_{j=0}^{t-1} \beta_1^{t-j} \hat{\mathbf{g}}_{j,i})^2}{(1-\beta_2)\sum_{j=0}^{t-1} \beta_2^{t-j} \hat{\mathbf{g}}_{j,i}^2} \leq \alpha_t^2 \sum_{i=1}^{d} \frac{(1-\beta_1)^2 (\sum_{j=0}^{t-1} \beta_1^{t-j})(\sum_{j=0}^{t-1} \beta_1^{t-j} \hat{\mathbf{g}}_{j,i}^2)}{(1-\beta_2)\sum_{j=0}^{t-1} \beta_2^{t-j} \hat{\mathbf{g}}_{j,i}^2} \\
\leq &\alpha_t^2 \sum_{i=1}^{d} \frac{(1-\beta_1)\sum_{j=0}^{t-1} \beta_1^{t-j} \hat{\mathbf{g}}_{j,i}^2}{(1-\beta_2)\sum_{j=0}^{t-1} \beta_2^{t-j} \hat{\mathbf{g}}_{j,i}^2} \leq \alpha_t^2 \sum_{i=1}^{d} \sum_{j=0}^{t-1} \frac{(1-\beta_1)\beta_1^{t-j} \hat{\mathbf{g}}_{j,i}^2}{(1-\beta_2)\beta_2^{t-j} \hat{\mathbf{g}}_{j,i}^2} \\
\leq &\alpha_t^2 d \sum_{j=0}^{t-1} \frac{1-\beta_1}{1-\beta_2} \gamma^{t-j} \leq \alpha_t^2 d \frac{1-\beta_1}{1-\beta_2} \frac{1}{1-\gamma}
\end{aligned}$$

where the second inequality is due to Cauchy-Schwarz and $\gamma = \beta_1/\beta_2 < 1$.

Q.E.D.

Proof of Proposition 2: Substitute (3.41) and (3.34) into (3.30), we get

$$\begin{aligned}
&\mathbb{E}[f_\mu(\mathbf{z}_{t+1}) - f_\mu(\mathbf{z}_1)] \\
\leq &\sum_{t=1}^{T} \mathbb{E}[\langle \nabla f_\mu(\mathbf{x}_t), \mathbf{z}_{t+1} - \mathbf{z}_t \rangle] + \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2} \sum_{t=1}^{T} \mathbb{E}[\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2] \\
\leq &\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2 \right) \left\| \frac{\alpha_1}{\sqrt{\hat{\mathbf{v}}_0}} \right\|_1 \right] - \sum_{t=1}^{T} \mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2} \nabla f_\mu(\mathbf{x}_t) \rangle \right] \\
&+ \sum_{t=1}^{T} \alpha_t^2 d \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2} \frac{1-\beta_1}{1-\beta_2} \frac{1}{1-\gamma}
\end{aligned} \tag{3.42}$$

Rearrange and assume $f_\mu(\mathbf{z}_1) - \min_{\mathbf{z}} f_\mu(\mathbf{z}) \leq D_f$, we get

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_t \hat{\mathbf{V}}_t^{-1/2} \nabla f_\mu(\mathbf{x}_t)\rangle\right] \\
\leq & D_f + \mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\left\|\frac{\alpha_1}{\sqrt{\hat{\mathbf{v}}_0}}\right\|_1\right] + \sum_{t=1}^{T} \alpha_t^2 d \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2}\frac{1-\beta_1}{1-\beta_2}\frac{1}{1-\gamma}
\end{aligned} \tag{3.43}$$

Set $\alpha_t = \alpha = 1/\sqrt{Td}$ and divide both sides by $T\alpha$, uniformly randomly pick R from 1 to T,

$$\begin{aligned}
&\mathbb{E}\left[\|\hat{\mathbf{V}}_R^{-1/2}\nabla f_\mu(x_R)\|^2\right] = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|V_t^{-1/2}\nabla f_\mu(\mathbf{x}_t)\|^2\right] \\
\leq & \frac{D_f}{T\alpha} + \frac{1}{T}\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\left\|\frac{1}{\sqrt{\hat{\mathbf{v}}_0}}\right\|_1\right] + \alpha d \frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2}\frac{1-\beta_1}{1-\beta_2}\frac{1}{1-\gamma} \\
= & \frac{\sqrt{d}}{\sqrt{T}} D_f + \frac{1}{T}\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + \eta^2\right)\left\|\frac{1}{\sqrt{\hat{\mathbf{v}}_0}}\right\|_1\right] + \frac{\sqrt{d}}{\sqrt{T}}\frac{4L_g + 5L_g\beta_1^2}{2(1-\beta_1)^2}\frac{1-\beta_1}{1-\beta_2}\frac{1}{1-\gamma}
\end{aligned} \tag{3.44}$$

Since $\hat{\mathbf{V}}_{0,ii}^{1/2} \geq c$, $\forall i \in [d]$. By Lemma 2, we have

$$\|\hat{\mathbf{V}}_t^{-1/4}(\nabla f_\mu(x) - \nabla f_\mu(x))\|^2 \leq \frac{\mu^2 d^2 L^2}{4c} \tag{3.45}$$

Then we can easily adapt (3.44) to

$$\begin{aligned}
\mathbb{E}\left[\|\hat{\mathbf{V}}_t^{-1/4}\nabla f(x_R)\|^2\right] \leq & \frac{\mu^2 d^2 L^2}{2c} + 2\frac{\sqrt{d}}{\sqrt{T}} D_f + 2\frac{1}{T}\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + 2\eta 2\right)\left\|\frac{1}{\sqrt{\hat{\mathbf{v}}_0}}\right\|_1\right] \\
& + \frac{\sqrt{d}}{\sqrt{T}}\frac{4L_g + 5L_g\beta_1^2}{(1-\beta_1)^2}\frac{1-\beta_1}{1-\beta_2}\frac{1}{1-\gamma}
\end{aligned}$$

Substituting into μ finishes the proof. **Q.E.D.**

Proof of Proposition 3

Upon defining $G_{\text{zo},i} := \max_{t\in[T]} |\hat{g}_{t,i}|$

by [Dasgupta and Gupta, 2003, Lemma 2.2], for a vector $\mathbf{u}$ sampled from a unit sphere

in $\mathbb{R}^d$, we have for any $i \in [d]$,

$$P[|u_i| \geq \sqrt{\xi/d}] \leq \exp\left((1 - \xi + \log \xi)/2\right). \tag{3.46}$$

Let $\xi = 4 \log \frac{dT}{\delta}$, and by the assumption of $\max(d, T) \geq 3$ we have $1 + \log \xi \leq \xi/2$. Thus, we obtain from (3.46) that

$$P[|u_i| \geq \sqrt{\xi/d}] \leq \exp(-\xi/4) = \exp(-\log(dT/\delta)) = \delta/dT. \tag{3.47}$$

Recall that the ZO gradient estimate $\hat{\mathbf{g}}_t$ is given by the form

$$\hat{\nabla} f(\mathbf{x}) = (d/\mu)[f(\mathbf{x} + \mu\mathbf{u}) - f(\mathbf{x})]\mathbf{u}. \tag{3.48}$$

By Lipschitz of f under **A2.2**, the ith coordinate of the ZO gradient estimate (3.48) is upper bounded by $dL_c|u_i|$. Since $\mathbf{u}$ is drawn uniformly randomly from a unit sphere, by (3.47) we have

$$P[dL_c|u_i| \geq L_c\sqrt{\xi d}] \leq \delta/dT. \tag{3.49}$$

Also, since $|\hat{g}_{t,i}| \leq dL_c|u_i|$, based on (3.49) we obtain that

$$P[|\hat{g}_{t,i}| \geq L_c\sqrt{\xi d}] \leq P[dL_c|u_i| \geq L_c\sqrt{\xi d}] \leq \delta/dT. \tag{3.50}$$

Substituting $\xi = 4 \log \frac{dT}{\delta}$ into (3.50), we have

$$P[|\hat{g}_{t,i}| \geq 2L_c\sqrt{d \log(dT/\delta)}] \leq \delta/dT \tag{3.51}$$

Then by the union bound and (3.51), we have

$$\begin{aligned} &P[|\hat{g}_{t,i}| \geq 2L_c\sqrt{d \log(dT/\delta)},\ \forall i, t] \\ \leq &\sum_{t \in [T]} \sum_{i \in [d]} P[|\hat{g}_{t,i}| \geq 2L_c\sqrt{d \log(dT/\delta)}] \leq dT(\delta/dT) = \delta, \end{aligned}$$

which implies the inequality (3.11). **Q.E.D.**

Proof of Theorem 2

The idea is to prove a similar result as Proposition 2 conditioned on the event in Proposition 3 ($\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\} \leq 2L_c\sqrt{d\log(dT/\delta)}$). Thus, the proof follows the same flow as Proposition 2. The difference is that (3.40) does not hold conditioned on the event and more efforts are need to bound the corresponding term in (3.40). Denote the event that $\max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\} \leq 2L_c\sqrt{d\log(dT/\delta)}$ to be $U(\delta)$, we need to upper bound

$$\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle | U(\delta)\right] \tag{3.52}$$

where $\mathbb{E}[\cdot|U(\delta)]$ is conditional expectation conditioned on $U(\delta)$.

By Proposition 3, we know $P(U(\delta)) \geq 1-\delta$ and using the fact that $E[\cdot|A] = \frac{E[\cdot]-E[\cdot|A^c]P(A^c)}{P(A)}$ for any event A and its complimentary event A^c, we have

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle | U(\delta)\right] \\
\leq &\frac{\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle\right]}{1-\delta} \\
&+ \frac{\delta\left|\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle | U(\delta)^c\right]\right|}{1-\delta}
\end{aligned} \tag{3.53}$$

and further we have

$$\begin{aligned}
&\left|\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle | U(\delta)^c\right]\right| \\
\leq &d\frac{\alpha_{t-1}}{c}(\eta^2 + \eta\max_{t\in[T]}\|\hat{g}_t\|_\infty) \\
\leq &d\frac{\alpha_{t-1}}{c}(\eta^2 + \eta d L_c)
\end{aligned} \tag{3.54}$$

where the first inequality is due to $\|\nabla f_\mu(x_t)\|_\infty \leq \eta$ and $\hat{v}_{t-1}^{1/2} \geq \hat{\mathbf{v}}_0^{1/2} \geq c\mathbf{1}$, the second inequality is due to (3.1) and Lipschitz continuity of $f(\mathbf{x};\boldsymbol{\xi})$.

Using the fact that $\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle\right] = 0$ proved in in (3.40)

and set $\delta = 1/Td^{0.5}$, we have for $T \geq 2$

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla f_\mu(\mathbf{x}_t), \alpha_{t-1}\hat{\mathbf{V}}_{t-1}^{-1/2}(\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t))\rangle | U(1/Td^{0.5})\right] \\
\leq & 2\frac{1}{Td^{0.5}} d \frac{\alpha_{t-1}}{c}(\eta^2 + \eta d L_c) = 2\frac{d^{1.5}}{T}\frac{\alpha_{t-1}}{c}\eta L_c + 2\frac{d^{0.5}}{T}\frac{\alpha_{t-1}}{c}\eta^2
\end{aligned} \tag{3.55}$$

Replacing (3.40) with (3.55) and going through the rest of the proof of Proposition (2), one can finally get

$$\begin{aligned}
&\mathbb{E}\left[\|\hat{\mathbf{V}}_t^{-1/4}\nabla f(x_R)\|^2 \big| U(1/Td^{0.5})\right] \\
\leq & \frac{\mu^2 d^2 L^2}{2c} + 2\frac{\sqrt{d}}{\sqrt{T}} D_f + 2\frac{1}{T}\mathbb{E}\left[\left(\frac{\eta G_{zo}}{1-\beta_1} + 2\eta 2\right)\left\|\frac{1}{\sqrt{\hat{\mathbf{v}}_0}}\right\|_1 \Big| U(1/Td^{0.5})\right] \\
& + \frac{\sqrt{d}}{\sqrt{T}}\frac{4L_g + 5L_g\beta_1^2}{(1-\beta_1)^2}\frac{1-\beta_1}{1-\beta_2}\frac{1}{1-\gamma} + 2\frac{d^{1.5}}{T}\frac{\eta L_c}{c} + 2\frac{d^{0.5}}{T}\frac{\eta^2}{c}.
\end{aligned}$$

Since in the event of $U(1/Td^{0.5})$, we have

$$G_{zo} = \max_{t\in[T]}\{\|\hat{\mathbf{g}}_t\|_\infty\} \leq 2L_c\sqrt{d\log(d^{1.5}T^2)} = 2L_c\sqrt{d}\sqrt{1.5\log d + 2\log T}. \tag{3.56}$$

Substituting the above inequality into (3.56), we get the desired result. **Q.E.D.**

Proof of Theorem 3

To proceed into proof of Theorem 3, we give a few technical lemmas for the properties of (3.7).

Lemma 14. *For any symmetric* $\mathbf{H} \succeq 0, \mathbf{g}, \omega$*, we have*

$$\langle \mathbf{g}, P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega)\rangle \geq \|\mathbf{H}^{1/2}P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega)\|^2 \tag{3.57}$$

Proof of Lemma 14: By definition of $\mathbf{x}^+$, the optimality condition of (3.6) is

$$\langle \mathbf{g} + \frac{1}{\omega}\mathbf{H}(\mathbf{x}^+ - \mathbf{x}^-), \mathbf{x} - \mathbf{x}^+\rangle \geq 0 \quad \forall \mathbf{x} \in \mathcal{X}$$

Thus

$$\langle \mathbf{g} + \frac{1}{\omega}\mathbf{H}(\mathbf{x}^+ - x), \mathbf{x} - \mathbf{x}^+ \rangle \geq 0$$

which can be rearranged to

$$\langle \mathbf{g}, P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega) \rangle = \frac{1}{\omega}\langle \mathbf{g}, x - \mathbf{x}^+ \rangle \geq \frac{1}{\omega^2}\langle \mathbf{H}(x - \mathbf{x}^+), x - \mathbf{x}^+ \rangle = \|\mathbf{H}^{1/2} P_{\mathcal{X},\mathbf{H}}(\mathbf{x}^-, \mathbf{g}, \omega)\|^2$$

This completes the proof. **Q.E.D.**

Lemma 15. *Let* $\mathbf{x}_1^+$ *and* $\mathbf{x}_2^+$ *be given by* (3.6) *with* $\mathbf{g}$ *replaced by* $\mathbf{g}_1$ *and* $\mathbf{g}_2$*, with* $H \succ 0$*, we have*

$$\|\mathbf{x}_1^+ - \mathbf{x}_2^+\| \leq \frac{\omega}{\lambda_{\min}(\mathbf{H})}\|\mathbf{g}_1 - \mathbf{g}_2\| \tag{3.58}$$

$$\|\mathbf{H}^{1/2}(\mathbf{x}_1^+ - \mathbf{x}_2^+)\| \leq \omega\|\mathbf{H}^{-1/2}(\mathbf{g}_1 - \mathbf{g}_2)\|. \tag{3.59}$$

where $\lambda_{\min}(\mathbf{H})$ *is the minimum eigenvalue of* $\mathbf{H}$*.*

Proof of Lemma 15: By definition of $\mathbf{x}^+$, the optimality condition of (3.6) is

$$\langle \mathbf{g} + \frac{1}{\omega}\mathbf{H}(\mathbf{x}^+ - \mathbf{x}^-), \mathbf{x} - \mathbf{x}^+ \rangle \geq 0 \quad \forall \mathbf{x} \in \mathcal{X}$$

Thus, we have

$$\langle \mathbf{g}_1 + \frac{1}{\omega}\mathbf{H}(\mathbf{x}_1^+ - \mathbf{x}^-, \mathbf{x}_2^+ - \mathbf{x}_1^+ \rangle \geq 0$$

$$\langle \mathbf{g}_2 + \frac{1}{\omega}\mathbf{H}(\mathbf{x}_2^+ - \mathbf{x}^-, \mathbf{x}_1^+ - \mathbf{x}_2^+ \rangle \geq 0$$

Summing up the above two inequalities, we get

$$\langle \mathbf{g}_1 - \mathbf{g}_2, \mathbf{x}_2^+ - \mathbf{x}_1^+ \rangle \geq \frac{1}{\omega}\langle \mathbf{H}(\mathbf{x}_2^+ - \mathbf{x}_1^+), \mathbf{x}_2^+ - \mathbf{x}_1^+ \rangle \tag{3.60}$$

By Cauchy-Schwarz inequality, we get

$$\begin{aligned}\|\mathbf{g}_1-\mathbf{g}_2\|\|\mathbf{x}_2^+-\mathbf{x}_1^+\| \geq \langle \mathbf{g}_1-\mathbf{g}_2, \mathbf{x}_2^+-\mathbf{x}_1^+\rangle \geq& \frac{1}{\omega}\langle \mathbf{H}(\mathbf{x}_2^+-\mathbf{x}_1^+), \mathbf{x}_2^+-\mathbf{x}_1^+\rangle \\ \geq& \frac{1}{\omega}\lambda_{\min}(\mathbf{H})\|\mathbf{x}_2^+-\mathbf{x}_1^+\|^2\end{aligned}$$

which gives (3.58).

Further, by (3.60) and Cauchy-Schwarz, we also have

$$\begin{aligned}&\|\mathbf{H}^{-1/2}(\mathbf{g}_1-\mathbf{g}_2)\|\|\mathbf{H}^{1/2}(\mathbf{x}_2^+-\mathbf{x}_1^+)\| \geq \langle \mathbf{g}_1-\mathbf{g}_2, \mathbf{x}_2^+-\mathbf{x}_1^+\rangle \\ \geq& \frac{1}{\omega}\langle \mathbf{H}(\mathbf{x}_2^+-\mathbf{x}_1^+), \mathbf{x}_2^+-\mathbf{x}_1^+\rangle = \frac{1}{\omega}\|\mathbf{H}^{1/2}(\mathbf{x}_2^+-\mathbf{x}_1^+)\|^2\end{aligned}$$

which gives (3.59). This completes the proof. **Q.E.D.**

The following lemma characterizes the difference between projected points if different distance matrices are used in ZO-AdaMM.

Lemma 16. *Assume* $\mathbf{V}_t^{1/2} \geq c\mathbf{I}$*, ZO-AdaMM yields*

$$\left\|(P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t) - P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t))\right\|^2 \leq \sum_{i=1}^d v_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^4}\eta^2. \tag{3.61}$$

Proof of Lemma 16: Recall the optimality condition of (3.6) is

$$\langle \mathbf{g} + \frac{1}{\omega}\mathbf{H}(\mathbf{x}^+ - \mathbf{x}^-, \mathbf{x} - \mathbf{x}^+\rangle \geq 0 \quad \forall \mathbf{x} \in \mathcal{X} \tag{3.62}$$

Let us define

$$\begin{aligned}\mathbf{x}_t^* &\triangleq \mathbf{x}_t - \alpha_t P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t) \\ \tilde{\mathbf{x}}_t^* &\triangleq \mathbf{x}_t - \alpha_t P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t).\end{aligned}$$

By optimality condition (3.62), we have

$$\langle \nabla f_\mu(\mathbf{x}_t) + \frac{1}{\alpha_t}\hat{\mathbf{V}}_t^{1/2}(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t), \mathbf{x}_t^* - \tilde{\mathbf{x}}_t^* \rangle \geq 0$$
$$\langle \nabla f_\mu(\mathbf{x}_t) + \frac{1}{\alpha_t}\hat{\mathbf{V}}_{t-1}^{1/2}(\mathbf{x}_t^* - \mathbf{x}_t), \tilde{\mathbf{x}}_t^* - \mathbf{x}_t^* \rangle \geq 0$$

Summing the above up, we get

$$\langle \hat{\mathbf{V}}_t^{1/2}(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t) - \hat{\mathbf{V}}_{t-1}^{1/2}(\mathbf{x}_t^* - \mathbf{x}_t), \mathbf{x}_t^* - \tilde{\mathbf{x}}_t^* \rangle \geq 0$$

which is equivalent to

$$\begin{aligned}&\langle (\hat{\mathbf{V}}_t^{1/2} - \hat{\mathbf{V}}_{t-1}^{1/2})(\mathbf{x}_t^* - \mathbf{x}_t), \mathbf{x}_t^* - \tilde{\mathbf{x}}_t^* \rangle \\ &\quad + \langle \hat{\mathbf{V}}_t^{1/2}(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t^*), \mathbf{x}_t^* - \tilde{\mathbf{x}}_t^* \rangle \geq 0.\end{aligned}$$

Further rearranging, we have

$$\langle (\hat{\mathbf{V}}_t^{1/2} - \hat{\mathbf{V}}_{t-1}^{1/2})(\mathbf{x}_t^* - \mathbf{x}_t), \mathbf{x}_t^* - \tilde{\mathbf{x}}_t^* \rangle \geq \|\hat{\mathbf{V}}_t^{1/4}(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t^*)\|^2 \geq c\|(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t^*)\|^2$$

which implies (by using Cauchy-Swartz on the left hand side and then squaring both sides)

$$\begin{aligned}c^2\|(\tilde{\mathbf{x}}_t^* - \mathbf{x}_t^*)\|^2 \leq &\|(\hat{\mathbf{V}}_t^{1/2} - \hat{\mathbf{V}}_{t-1}^{1/2})(\mathbf{x}_t^* - \mathbf{x}_t)\|^2 = \sum_{i=1}^d (\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})^2(\hat{x}_{t,i}^* - x_{t,i})^2 \\ &\overset{(a)}{\leq} \sum_{i=1}^d \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\|\hat{x}_t^* - x_t\|^2 \overset{(b)}{\leq} \sum_{i=1}^d \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^2}\alpha_t^2\|\nabla f_\mu(x_t)\|^2 \\ &\leq \sum_{i=1}^d \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^2}\alpha_t^2\eta^2 \qquad (3.63)\end{aligned}$$

where (a) is due to $\hat{v}_{t,i}^{1/2} \geq \hat{v}_{t-1,i}^{1/2}$ and (b) is due to Lemma 15 by treating $\mathbf{g}_1 = \nabla f_\mu(\mathbf{x}_t)$, $\mathbf{g}_2 = 0$, $\mathbf{x}^- = \mathbf{x}_t$, $H = \hat{\mathbf{V}}_t^{1/2}$. Substituting (3.7) into LHS of the above inequality and rearrange, we get (3.61). This completes the proof. **Q.E.D.**

Now we are ready to prove our main theorem.

Proof of Theorem 3:

We start with standard decent lemma in nonconvex optimization. By Lipschitz smoothness of f_μ, we have

$$f_\mu(\mathbf{x}_{t+1}) \leq f_\mu(\mathbf{x}_t) - \alpha_t \langle \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle + \frac{L}{2}\alpha_t^2 \|P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2. \tag{3.64}$$

We need to upper bound RHS of the above inequality and split out a descent quantity.

$$\begin{aligned}
&- \langle \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle \\
=& - \langle \hat{\mathbf{g}}_t, P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle + \langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle \\
\leq& - \|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2 + \langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle
\end{aligned} \tag{3.65}$$

where the inequality is by Lemma (14) and some simple substitutions.

Further, for the last term in RHS of (3.65) we have

$$\begin{aligned}
&\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle \\
&= \left.\begin{aligned} &+\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle \\ &-\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle \end{aligned}\right\} A \\
&\left.\begin{aligned} &+\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle \\ &-\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle \end{aligned}\right\} B \\
&+ \underbrace{\langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle}_{C}
\end{aligned} \tag{3.66}$$

Next, we bound the three terms in RHS of (3.66).

Let's bound term A first, with the assumption $\hat{\mathbf{V}}^{1/2} \geq c\mathbf{I}$, by Lemma 15, (3.7) and Cauchy-Schwartz inequality, we have:

$$A = \langle \hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t) - P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle \leq \frac{1}{c}\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2 \tag{3.67}$$

Now let's bound term C, because $\mathbb{E}[\hat{\mathbf{g}}_t] = \nabla f_\mu(\mathbf{x}_t)$ and the noise in $\hat{\mathbf{g}}_t$ is independent

of $\nabla f_\mu(\mathbf{x}_t)$ and $\hat{\mathbf{V}}_{t-1}$, we have

$$\mathbb{E}[\langle \nabla f_\mu(\mathbf{x}_t) - \hat{\mathbf{g}}_t, P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)\rangle] = 0 \tag{3.68}$$

Substituting the above bounds for A and C, into (3.66) and (3.65), using Young's inequality on term B, we have

$$\begin{aligned} &- \mathbb{E}[\langle \nabla f_\mu(\mathbf{x}_t), P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\rangle] \\ \leq &- \mathbb{E}[\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2] + \frac{1}{c}\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] + \frac{1}{2}\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] + \frac{1}{2}\mathbb{E}[B_2] \end{aligned} \tag{3.69}$$

where we define

$$B_2 := \left\| (P_{\mathcal{X},\hat{\mathbf{V}}_{t-1}^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t) - P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t)) \right\|^2.$$

What remains is to bound the term $B2$ which is given by Lemma 16.

Combining (3.64), (3.69), (3.61), we have

$$\begin{aligned} \mathbb{E}[f_\mu(\mathbf{x}_{t+1})] \leq &\mathbb{E}[f_\mu(\mathbf{x}_t)] - \alpha_t \mathbb{E}[\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2] + \alpha_t(\frac{1}{c} + \frac{1}{2})\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] \\ &+ \alpha_t \frac{1}{2}\mathbb{E}[\sum_{i=1}^d \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^4}\eta^2] + \frac{L}{2}\alpha_t^2 \mathbb{E}\left[\frac{1}{c^2}\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2\right] \end{aligned} \tag{3.70}$$

which can be rearranged into

$$\begin{aligned} &(\alpha_t - \frac{L}{2c^2}\alpha_t^2)\mathbb{E}[\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2] \\ \leq &\mathbb{E}[f_\mu(\mathbf{x}_t)] - \mathbb{E}[f_\mu(\mathbf{x}_{t+1})] + \alpha_t(\frac{1}{c} + \frac{1}{2})\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] \\ &+ \alpha_t \frac{1}{2}\mathbb{E}\left[\sum_{i=1}^d \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^4}\eta^2\right]. \end{aligned} \tag{3.71}$$

In addition, we have

$$\begin{aligned}
&\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f(\mathbf{x}_t), \alpha_t)\|^2 \leq 3\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2 \\
&\qquad + 3\,\|\hat{\mathbf{V}}_t^{1/4}(P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t) - P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f(\mathbf{x}_t), \alpha_t))\|^2 \\
&\qquad + 3\|\hat{\mathbf{V}}_t^{1/4}(P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t) - P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f_\mu(\mathbf{x}_t), \alpha_t))\|^2 \\
&\leq 3\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \hat{\mathbf{g}}_t, \alpha_t)\|^2 + \frac{3}{c}\|\nabla f_\mu(\mathbf{x}_t) - \nabla f(\mathbf{x}_t)\|^2 + \frac{3}{c}\|\hat{\mathbf{g}}_t - \nabla f_\mu(\mathbf{x}_t)\|^2 \qquad (3.72)
\end{aligned}$$

where the second inequality is by (3.7) and Lemma (15)

Combining (3.72) and (3.71), we have

$$\begin{aligned}
&\left(\alpha_t - \frac{L}{2c^2}\alpha_t^2\right)\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t, \nabla f(\mathbf{x}_t), \alpha_t)\|^2 \\
&\leq 3(\mathbb{E}[f_\mu(\mathbf{x}_t)] - \mathbb{E}[f_\mu(\mathbf{x}_{t+1})]) + (3\alpha_t(\frac{1}{c} + \frac{1}{2}) + \frac{3}{c}(\alpha_t - \frac{L}{2c^2}\alpha_t^2))\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] \\
&\quad + \frac{3}{2}\alpha_t \mathbb{E}[\sum_{i=1}^{d} \hat{v}_{t,i}^{1/2}(\hat{v}_{t,i}^{1/2} - \hat{v}_{t-1,i}^{1/2})\frac{1}{c^4}\eta^2] + \frac{3}{c}(\alpha_t - \frac{L}{2c^2}\alpha_t^2)\|\nabla f_\mu(\mathbf{x}_t) - \nabla f(\mathbf{x}_t)\|^2 \qquad (3.73)
\end{aligned}$$

Summing over t from 1 to T, setting $\alpha_t = \alpha$, and dividing both sides by $T(\alpha - \frac{L_g\alpha^2}{2c^2})$, we get

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\|\hat{\mathbf{V}}_t^{1/4} P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t \nabla f(\mathbf{x}_t), \alpha_t)\|^2] \\
&\leq \frac{3}{T(\alpha - \frac{L_g\alpha^2}{2c})}(\mathbb{E}[f_\mu(\mathbf{x}_1)] - \mathbb{E}[f_\mu(x_{T+1})]) + \left(\frac{3\alpha(c+2)}{2Tc(\alpha - \frac{L_g\alpha^2}{2c})} + \frac{3}{Tc}\right)\sum_{t=1}^{T}\mathbb{E}[\|\hat{\mathbf{g}}_t - f_\mu(\mathbf{x}_t)\|^2] \\
&\quad + \frac{3\alpha}{2T(\alpha - \frac{L_g\alpha^2}{2c})}\mathbb{E}[\sum_{i=1}^{d}\hat{v}_{T,i}]\frac{1}{c^4}\eta^2 + \frac{3}{Tc}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f_\mu(\mathbf{x}_t) - \nabla f(\mathbf{x}_t)\|^2]. \qquad (3.74)
\end{aligned}$$

Choose $\alpha \leq \frac{c}{L}$, we have

$$\alpha - \frac{L_g\alpha^2}{2c} = \alpha\left(1 - \frac{L_g\alpha}{2c}\right) \geq \alpha(1 - \frac{1}{2}) = \frac{\alpha}{2} \qquad (3.75)$$

and (3.74) becomes

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\hat{\mathbf{V}}_t^{1/4}P_{\mathcal{X},\hat{\mathbf{V}}_t^{1/2}}(\mathbf{x}_t\nabla f(\mathbf{x}_t),\alpha_t)\|^2\right]\\
\leq&\frac{6}{T\alpha}D_f+\frac{1}{T}(\frac{9}{c}+3)\sum_{t=1}^{T}\mathbb{E}\left[\|\hat{\mathbf{g}}_t-f_\mu(\mathbf{x}_t)\|^2\right]+\frac{3}{T}\frac{1}{c^4}\eta^2\mathbb{E}\left[\sum_{i=1}^{d}\hat{v}_{T,i}\right]+\frac{3}{c}\frac{\mu^2d^2L_g^2}{4}
\end{aligned}\tag{3.76}$$

where we defined $D_f := \mathbb{E}[f_\mu(\mathbf{x}_1)] - \min_x f_\mu(x)$ and used the fact that $\|\nabla f_\mu(\mathbf{x}_t) - \nabla f(\mathbf{x}_t)\|^2 \leq \frac{\mu^2d^2L_g^2}{4}$ by Lemma 2.

Further, we have

$$\begin{aligned}
\mathbb{E}\left[\sum_{i=1}^{d}\hat{v}_{T,i}\right]&=\mathbb{E}\left[\sum_{i=1}^{d}\max_{t\in[T]}(1-\beta_2)\sum_{k=1}^{t}\beta_2^{t-k}\hat{g}_{k,i}^2\right]\\
\leq&\mathbb{E}\left[d\max_{t\in[T]}(1-\beta_2)\sum_{k=1}^{t}\beta_2^{t-k}\|\hat{g}_k\|_\infty\right]\\
\leq&\mathbb{E}\left[d\max_{t\in[T]}\|\hat{g}_t\|_\infty\right]
\end{aligned}\tag{3.77}$$

where the last inequality holds since $\sum_{k=1}^{T}\beta_2^{T-k} \leq 1/(1-\beta_2)$.

Uniformly randomly picking R from 1 to T and substituting (3.77) into (3.76) finishes the proof. **Q.E.D.**

Chapter 4

Understanding gradient clipping in private SGD

4.1 Introduction

In this chapter, we study gradient clipping, a different form of adaptivity. It is shown in Zhang et al. [2020] that GD with gradient clipping achieves faster convergence when the problems satisfies a special smoothness assumption. However, for SGD, gradient clipping could create estimation bias on gradient, preventing convergence. In this chapter, we focus on understanding the adversarial effect of gradient clipping, which is of great importance to further understand the performance of (differentially) private SGD in practice.

Many modern applications of machine learning rely on datasets that may contain sensitive personal information, including medical records, browsing history, and geographic locations. To protect the private information of individual citizens, many machine learning systems now train their models subject to the constraint of differential privacy [Dwork et al., 2006], which informally requires that no individual training example has a significant influence on the trained model. To achieve this formal privacy guarantee, one of the most popular training methods, especially for deep learning, is *differentially private stochastic gradient descent* (DP-SGD) [Bassily et al., 2014, Abadi et al., 2016b, Song et al., 2013]. At a high level, DP-SGD is a simple modification of SGD that makes each step differentially private with the *Gaussian mechanism*: at each iteration t, it

first computes a gradient estimate g_t based on a random subsample, and then updates the model using a noisy gradient $\tilde{g}_t = g_t + \eta$, where η is a noise vector drawn from a multivariate Gaussian distribution.

Despite the simple form of DP-SGD, there is a major disparity between its theoretical analysis and practical implementation. The formal privacy guarantee of Gaussian mechanism requires that the per-coordinate standard deviation of the noise vector η scales linearly with the ℓ_2 sensitivity of the gradient estimate g_t—that is, the maximal change on g_t in ℓ_2 distance if by changing a single example. To bound the ℓ_2-sensitivity, existing theoretical analyses typically assume that the loss function is L-Lipschitz in the model parameters, and the constant L is known to the algorithm designer for setting the noise rate [Bassily et al., 2014, Wang and Xu, 2019]. Since this assumption implies that the gradient of each example has ℓ_2 norm bounded by L, any gradient estimate from averaging over the gradients of m examples has ℓ_2-sensitivity bounded by L/m. However, in many practical settings, especially those with deep learning models, such Lipschitz constant or gradient bounds are not a-priori known or even computable (since it involves taking the worst case over both examples and pairs of parameters). In practice, the bounded ℓ_2-sensitivity is ensured by *gradient clipping* [Abadi et al., 2016b] that shrinks an individual gradient whenever its ℓ_2 norm exceeds certain threshold c. More formally, given any gradient g on a simple example and a clipping threshold c, the gradient clipping does the following

$$\mathrm{clip}(g, c) = g \cdot \min\left(1, \frac{c}{\|g\|}\right). \tag{4.1}$$

The clipping operation can be viewed as a special form of adaptivity, and it is shown in Zhang et al. [2020] that SGD with gradient clipping can be provably faster than SGD under certain assumptions. However, the clipping operation in general can create a substantial bias in the update direction, leading to divergence. To illustrate this clipping bias, consider the following two optimization problems even without the privacy constraint.

Example 1. Consider optimizing $f(x) = \frac{1}{3}\sum_{i=1}^{3}\frac{1}{2}(x - a_i)^2$ over $x \in \mathbb{R}$, where $a_1 = a_2 = -3$ and $a_3 = 9$. Since the gradient $\nabla f(x) = x - 1$, the optimum is $x^* = 1$. Now suppose we run SGD with gradient clipping with a threshold of $c = 1$. At the optimum,

the gradients for all three examples are clipped and the expected clipped gradient is 1/3, which leads the parameter to move away from x^*.

Example 2. Let $f(x) = \frac{1}{2}\sum_{i=1}^{2}\frac{1}{2}(x-a_i)^2$, where $a_1 = -3$ and $a_2 = 3$. The minimum of f is achieved at $x^* = 0$, where the expected clipped gradient is also 0. However, SGD with clipped gradients and $c = 1$ may never converge to x^* since the expected clipped gradients are all 0 for any $x \in [-2, 2]$, which means all these points are "stationary" for the algorithm.

Both examples above show that clipping bias can prevent convergence in the worst case. Existing analyses on gradient clipping quantify this clipping bias either with 1) the difference between clipped and unclipped gradients [Pichapati et al., 2019], or 2) the fraction of examples with gradient norms exceeding the clip threshold c [Zhang et al., 2020]. These approaches suggest that a small clip threshold will lead to large clipping bias and worsen the training performance of DP-SGD. However, in practice, DP-SGD often remains effective even with a *small* clip threshold [Beaulieu-Jones et al., 2019, Bu et al., 2019], which indicates a gap in the current theoretical understanding of gradient clipping.

4.1.1 Our results

We study the effects of gradient clipping on SGD and DP-SGD and provide:

Symmetricity-based analysis. We characterize the clipping bias on the convergence to stationary points through the geometric structure of the gradient distribution. To isolate the clipping effects, we first analyze the non-private SGD with gradient clipping (but without Gaussian perturbation), with the following key analysis steps. **1)** We first show that the inner product $\mathbb{E}[\langle \nabla f(x_t), g_t \rangle]$ goes to zero in SGD, where $\nabla f(x)$ denotes the true gradient and g_t denotes a clipped stochastic gradient. **2)** We then show that when the gradient distribution is symmetric, inner product upper bounds a constant re-scaling of $\|\nabla f(x_t)\|$, and so SGD minimizes the gradient norm. **3)** We quantify the clipping bias via a coupling between the gradient distribution and a nearby symmetric distribution and express it as a disparity measure (that resembles the Wasserstein distance) between the two distributions. As a result, when the gradient distributions are near-symmetric or

when the clipping bias favors convergence, the clipped gradient remains aligned with the true gradient, even if clipping aggressively shrinks almost all the sample gradients.

Theoretical and empirical evaluation of DP-SGD. Building on the previous SGD analysis, we obtain a similar convergence guarantee on DP-SGD with gradient clipping. Importantly, we are able to prove such convergence guarantee even *without* Lipschitzness of the loss function, which is often required for DP-SGD analyses. We also provide extensive empirical studies to investigate the gradient distributions of DP-SGD across different epoches on two real datasets. To visualize the symmetricity of the gradient distributions, we perform multiple random projections on the gradients and examine the two-dimensional projected distributions. Our results suggest that the gradient distributions in DP-SGD quickly exhibit symmetricity, despite the asymmetricity at initialization.

Gradient correction mechanism. Finally, we provide a simple modification to DP-SGD that can mitigate the clipping bias. We show that perturbing the gradients *before* clipping can provably reduce the clipping bias for any gradient distribution. The pre-clipping perturbation does not by itself provide privacy guarantees, but can trade-off the clipping bias with higher variance.

4.1.2 Related work

The divergence caused by the clipping bias was also studied by prior work. In Pichapati et al. [2019], an adaptive gradient clipping method is analyzed and the divergence is characterized by a bias depending on the difference between the clipped and unclipped gradients. However, they study a different variant of clipping that bounds the ℓ_∞ norm of the gradient instead of ℓ_2 norm; the latter, which we study in this paper, is the more commonly used clipping operation [Abadi et al., 2016b,a]. In Zhang et al. [2020], the divergence is characterized by a bias depending on the clipping probability. These results suggest that, the clipping probability as well as the bias are *inversely* proportional to the size of the clipping threshold. For example, small clipping threshold results in large bias in the gradient estimation, which can potentially lead to worse training and generalization performance. Thakkar et al. [2019] provides another adaptive gradient clipping heuristic

that sets the threshold based on a privately estimated quantile, which can be viewed as minimizing the clipping probability.

4.2 Convergence of SGD with clipped gradient

In this section, we analyze convergence of SGD with clipped gradient, but without the Gaussian perturbation. This simplification is useful for isolating the clipping bias. Consider the standard stochastic optimization formulation

$$\min_{x \in \mathbb{R}^d} f(x) = \mathbb{E}_{s \sim D}[f(x, s)], \tag{4.2}$$

where $x \in \mathbb{R}^d$ is the optimization variable; D denotes the underlying distribution over the examples s. In the next section, we will instantiate D as the empirical distribution over the private dataset. We assume that the algorithm is given access to a stochastic gradient oracle: given any iterate x_t of SGD, the oracle returns $\nabla f(x_t) + \xi_t$, where ξ_t is independent noise with zero mean. In addition, we assume $f(x)$ is G-smooth, i.e. $\|\nabla f(x) - \nabla f(y)\| \leq G\|x - y\|, \forall x, y$. At each iteration t, SGD with gradient clipping performs the following update:

$$x_{t+1} = x_t - \alpha \text{clip}(\nabla f(x_t) + \xi_t, c) := x_t - \alpha g_t, \tag{4.3}$$

where $g_t := \text{clip}(\nabla f(x_t) + \xi_t, c)$ denotes the realized clipped gradient.

To carry out the analysis of iteration (4.3), we first note that the standard convergence analysis for SGD-type method consists of two main steps:

S1) Show that the term $\mathbb{E}[\langle \nabla f(x_t), g_t \rangle]$ diminishes to zero.

S2) Show that the aforementioned quantity is proportional to $\|\nabla f(x_t)\|^2$ or $c\|\nabla f(x_t)\|$, indicating that the size of gradient also decreases to zero.

In our analysis below, we will see that showing the first step is relatively easy, while the main challenge is to show that the second step holds true. Our first result is given below.

Theorem 4. *Let G be the Lipschitz constant of ∇f such that $\|\nabla f(x) - \nabla f(y)\| \leq G\|x - y\|, \forall x, y$. For SGD with gradient clipping of threshold c, if we set $\alpha = \frac{1}{\sqrt{T}}$, we*

have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla f(x_t), g_t\rangle\right] \leq \frac{D_f}{\sqrt{T}} + \frac{G}{2\sqrt{T}}c^2, \tag{4.4}$$

where $D_f := f(x_1) - \min_x f(x)$.

Note that for SGD without clipping, we have $\mathbb{E}[\langle\nabla f(x_t), g_t\rangle] = \|\nabla f(x_t)\|^2$, so the convergence can be readily established. However, when clipping is applied, the expectation is different but if we have $\mathbb{E}[\langle\nabla f(x_t), g_t\rangle]$ being positive, or have it to scale with $\|\nabla f(x_t)\|$, we can still establish a convergence guarantee. However, the divergence examples (Example 1 and 2) indicate proving this second step requires additional conditions. Now we study a geometric condition that is observed empirically.

4.2.1 Symetricity-Based Analysis on Gradient Distribution

Let $p_t(\xi_t)$ be the probability density function of ξ_t and $\tilde{p}_t(\xi_t)$ is an arbitrary distribution. To quantify the clipping bias, we start the analysis with the following decomposition:

$$\mathbb{E}_{\xi_t\sim p}[\langle\nabla f(x_t), g_t\rangle] = \mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle] + \underbrace{\int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle(p_t(\xi_t)-\tilde{p}_t(\xi_t))d\xi_t}_{:=b_t}. \tag{4.5}$$

In (4.5), we can choose $\tilde{p}(\xi_t)$ to be some "nice" distribution that can effectively relate $\mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle]$ to $\|\nabla f(x_t)\|^2$ and the remaining term will be treated as the bias. This way of splitting ensures that when the gradients follow a "nice" distribution, the bias will diminish with the distance between p and $\tilde{p}_t$. More precisely, we want to find a distribution $\tilde{p}$ such that $\mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle]$ is lower bounded by norm sqaured of the true gradient and thus convergence can be ensured.

A straightforward "nice" distribution will be $\langle\nabla f(x_t), g_t\rangle \geq \Omega(\|\nabla f(x_t)\|_2^2),\ \forall g_t$, i.e. all stochastic gradients are positively aligned with the true gradient. This may be satisfied when the gradient is large and the noise ξ is bounded. However, when the gradient is small, it is hard to argue that this can still be true in general. Specifically, in the training of neural nets, the cosine similarities between many stochastic gradients and the true gradient (i.e. $\cos(\nabla f(x_t), \nabla f(x_t)+\xi_t)$) can be negative, which implies that this

assumption does not hold (see Figure 4.3 in Section 4.4).

Although Figure 4.3 seems to exclude the *ideal* distribution, we observe that the distribution of cosine of the gradients appears to be *symmetric*. Will such a "symmetricity" property help define the "nice" distribution for gradient clipping? If so, how to characterizes the performance of gradient clipping in this situation? In the following result, we rigorously answer to these questions.

Theorem 5. *Assume $\tilde{p}(\cdot)$ is a symmetric distribution satisfying $\tilde{p}(\xi_t) = \tilde{p}(-\xi_t)$, $\forall\, \xi_t \in \mathbb{R}^d$. Then gradient clipping with threshold c has the following properties:*

1. *If* $\|\nabla f(x_t)\| \leq \frac{3}{4}c$, *then* $\mathbb{E}_{\xi_t\sim\tilde{p}}[\langle \nabla f(x_t), g_t\rangle] \geq \|\nabla f(x_t)\|^2 \mathbb{P}_{\xi_t\sim\tilde{p}}\left(\|\xi_t\| < \frac{c}{4}\right)$;
2. *If* $\|\nabla f(x_t)\| > \frac{3}{4}c$, *then* $\mathbb{E}_{\xi_t\sim\tilde{p}}[\langle \nabla f(x_t), g_t\rangle] \geq \frac{3\cdot c}{4}\|\nabla f(x_t)\| \mathbb{P}_{\xi_t\sim\tilde{p}}\left(\|\xi_t\| < \frac{c}{4}\right)$.

Theorem 5 states that when the noise distribution is symmetric, gradient clipping will keep the expected clipped gradients positively aligned with the true gradient. This is the desired property that can guarantee convergence. Combining Theorem 5 with Theorem 4, we have Corollary 3 to fully characterize the convergence behavior of SGD with gradient clipping.

Corollary 3. *Consider the SGD algorithm with gradient clipping given in* (4.3)*. Set $\alpha = \frac{1}{\sqrt{T}}$, and choose $\tilde{p}(\cdot)$ as a symetric distribution satisfying $\tilde{p}_t(\xi_t) = \tilde{p}_t(-\xi_t)$, $\forall\, \xi \in \mathbb{R}^d$. Then the following holds:*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{P}_{\xi_t\sim\tilde{p}_t}\left(\|\xi_t\| < \frac{c}{4}\right) \min\left\{\|\nabla f(x_t)\|, \frac{3}{4}c\right\} \|\nabla f(x_t)\| \leq \frac{D_f}{\sqrt{T}} + \frac{G}{2\sqrt{T}}c^2 - \frac{1}{T}\sum_{t=1}^{T} b_t, \tag{4.6}$$

where we have defined $b_t := \int \langle \nabla f(x_t), clip(\nabla f(x_t) + \xi_t, c)\rangle (p_t(\xi_t) - \tilde{p}_t(\xi_t)) d\xi_t$.

The above result suggests that, as long as the probabilities $\mathbb{P}_{\xi_t\sim\tilde{p}_t}\left(\|\xi_t\| < \frac{c}{4}\right)$ are bounded away from 0 and the symmetric distributions $\tilde{p}_t$ are close approximations to p_t (small bias b_t),[1] then gradient norm goes to 0. Moreover, when $\|\xi_t\|$ is drawn from a sub-gaussian distribution with constant variance, the probability does not diminish with

[1]Both Theorem 5 and Corollary 3 hold under a more relaxed condition of $\tilde{p}(\xi) = \tilde{p}(-\xi)$ for ξ with ℓ_2 norm exceeding $c/4$.

the dimension. This is consistent with the observations in recent work of Li et al. [2020], Gur-Ari et al. [2018] on deep learning training, and we also provide our own empirical evaluation on the probability term in the Appendix. Note if the bias is negative and very large, the bound on the rhs will not be meaningful. Therefore, it is useful to further study properties of such bias term. In the next section, we will discuss how large the bias term can be for a few choices of p and $\tilde{p}$. It turns out that the accumulation of b_t can help in some cases. In addition, one can extend the convergence results to some special non-symmetric distributions.

4.2.2 Beyond symmetric distributions

Theorem 5 and Corollary 3 suggest that as long as the distribution p is sufficiently close to a symmetric distribution $\tilde{p}$, the convergence bias expressed as $\sum_{t=1}^{T} b_t$ will be small. We now show that our bias decomposition result enables us to analyze the effect of the bias even for some highly asymmetric distributions. Note that when $b_t \geq 0$, the bias in fact helps convergence according Corollary 3.

We now provide three examples where b_t can be non-negative. Therefore, near-symmetricity is not a necessary condition for convergence, and our symmetricity-based analysis remains an effective tool to establish convergence for a broad class of distributions.

Positively skewed. Suppose p is positively skewed, that is, $p(\xi) \geq p(-\xi)$, for all ξ with $\langle \xi, \nabla f(x) \rangle > 0$. With such distributions, the stochastic gradients tend to be positively aligned with the true gradient. If one chooses $\tilde{p}(\xi_t) = \frac{1}{2}(p(\xi_t) + p(-\xi_t))$, the bias b_t can be written as

$$\int_{\xi_t \in \{\xi : \langle \xi, \nabla f(x_t) \rangle > 0\}} \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c) - \mathrm{clip}(\nabla f(x_t) - \xi_t, c) \rangle \left(\frac{1}{2}(p(\xi_t) - p(-\xi_t)) \right) d\xi_t,$$

which is always positive since $\langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c) - \mathrm{clip}(\nabla f(x_t) - \xi_t, c) \rangle \geq 0$. Substituting into (4.5), we have $\mathbb{E}_{\xi_t \sim p}[\langle \nabla f(x_t), g_t \rangle]$ strictly larger than $\mathbb{E}_{\xi_t \sim \tilde{p}}[\langle \nabla f(x_t), g_t \rangle]$, which means the positive skewness helps the convergence (we want $\mathbb{E}_{\xi_t \sim p}[\langle \nabla f(x_t), g_t \rangle]$ as large as possible).

Mixture of symmetric. The distribution of stochastic gradient $\nabla f(x_t) + \xi_t$ is a mixture of two symmetric distributions p_0 and p_1 with mean 0 and v respectively. Such a distribution might be possible when most of samples are well classified. In this case,

even though the distribution of ξ_t is not symmetric, one can apply similar argument of Theorem 5 to the component with mean v, and the zero mean component yield a bias 0. In particular, let w_0 be the probability that $\nabla f(x_t) + \xi_t$ is drawn from p_0. One can choose $\tilde{p} = p - w_0 p_0$ which is the component symmetric over v. The bias become

$$\int \langle \nabla f(x_t), \text{clip}(\nabla f(x_t) + \xi_t, c) \rangle \, w_0 p_0(\xi_t) d\xi_t = 0. \tag{4.7}$$

This is because $p_0(\xi_t)$ corresponds to a zero mean symmetric distribution of $\nabla f(x_t) + \xi_t$, and the fact that if $\nabla f(x_t) + \xi_t$ follows a symmetric distribution centered at 0, so does $\text{clip}(\nabla f(x_t) + \xi_t, c)$ for any $c > 0$. Note that despite $\tilde{p} = p - w_0 p_0$ is not a distribution since $\int \tilde{p}(\xi_t) = 1 - w_0$, Theorem 5 can still be applied with everything on r.h.s. of inequalities multiplied by $1 - w_0$ because one can apply Theorem 5 to distribution $\tilde{p}(\xi_t)/(1 - w_0)$ and then scale everything down.

Mixture of symmetric or positively skewed. If p is a mixture of multiple symmetric or positively skewed distributions, one can split the distributions into multiple ones and use their individual properties. That is, one can easily establish convergence guarantee for p being a mixture of m spherical distributions with mean $u_1, ..., u_m$ and $\langle f(x_t), u_i \rangle \geq 0, \forall i \in [m]$ as in the following theorem.

Theorem 6. *Given m distributions with the pdf of the ith distribution being $p_i(\xi) = \phi_i(\|\xi - u_i\|)$ for some function ϕ_i. If $\nabla f(x_t) = \sum_{i=1}^m w_i u_i$ for some $w_i \geq 0, \sum_{i=1}^m w_i = 1$. Define a mixture of these distributions with zero mean as below:*

$$p'(\xi) = \sum_{i=1}^m w_i p_i(\xi - \nabla f(x_t)).$$

If $\langle u_i, \nabla f(x_t) \rangle \geq 0, \forall i \in [m]$, we have

$$\mathbb{E}_{\xi_t \sim p'} \left[\langle \nabla f(x_t), g_t \rangle \right] \geq \|\nabla f(x_t)\| \sum_{i=1}^m w_i \min\left(\|u_i\|, \frac{3}{4} c \right) \cos(\nabla f(x_t), u_i) \mathbb{P}_{\xi_t \sim p_i} \left(\|\xi_t\| < \frac{c}{4} \right) \geq 0.$$

Besides these examples of favorable biases above, there are also many cases where b_t can be negative and lead to a convergence gap, such as negatively skewed distributions or multimodal distributions with highly imbalanced modes. We have illustrated possible distributions in our divergence examples (Examples 1 and 2). In such cases, one should

expect that clipping has an adversarial impact on the convergence guarantee. However, as we also show in Section 4.4, the gradient distributions on real datasets tend to be symmetric, so their clipping biases are small.

4.3 DP-SGD with Gradient Clipping

We now extend the results above to analyze the overall convergence DP-SGD with gradient clipping. To match up with the setting in Section 4.2, we consider the distribution D to be the empirical distribution over a private dataset S of n examples $\{s_1, \ldots, s_n\}$, and so $f(x) = \frac{1}{n}\sum_{i=1}^{n} f(x, s_i)$. For any iterate $x_t \in \mathbb{R}^d$ and example s_i, let $\xi_{t,i} = \nabla f(x_t, s_i) - \nabla f(x_t)$ denote the gradient noise on the example, and p_t denote the distribution over $\xi_{t,i}$. At each iteration t, DP-SGD performs:

$$x_{t+1} = x_t - \alpha \left(\left(\frac{1}{|S_t|} \sum_{i \in S_t} \text{clip}(\nabla f(x_t) + \xi_{t,i}, c) \right) + Z_t \right), \tag{4.8}$$

where S_t is a random subsample of S (with replacement)[2] and $Z_t \sim \mathcal{N}(0, \sigma^2 I)$ is the noise added for privacy. We first recall the privacy guarantee of the algorithm below:

Theorem 7 (Privacy (Theorem 1 in Abadi et al. [2016b]))**.** *There exist constants u and v so that given the number of iterations T, for any $\epsilon \leq uq^2T$, where $q = \frac{|S_t|}{n}$, DP-SGD with gradient clipping of threshold c is (ϵ, δ)-differentially private for any $\delta > 0$, if $\sigma^2 \geq v\frac{c^2 T \ln\left(\frac{1}{\delta}\right)}{n^2\epsilon^2}$.*

By accounting for the sub-sampling noise and Gaussian perturbation in DP-SGD, we obtain the following convergence guarantee, where we further bound the clipping bias term b_t with the Wasserstein distance between the gradient distribution and a coupling symmetric distribution.

Theorem 8 (Convergence)**.** *Suppose $x \in \mathbb{R}^d$, let $m = |S_t|$, and let $\tilde{p}_t$ be a symmetric distribution with $\tilde{p}_t(\xi_t) = \tilde{p}_t(-\xi_t)$, $\forall\ \xi_t \in \mathbb{R}^d$. For DP-SGD with gradient clipping, set*

$$\alpha = \frac{\sqrt{D_f d \ln(\frac{1}{\delta})}}{n\epsilon c\sqrt{L}}.$$

[2] Alternatively, subsampling with replacement [Wang et al., 2019] and Poisson subsampling [Zhu and Wang, 2019] have also been proposed.

Then there exist u and v such that for any $\epsilon \leq u\frac{m^2}{n^2}T$, $\sigma^2 = v\frac{c^2 T \ln\left(\frac{1}{\delta}\right)}{n^2\epsilon^2}$, we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{P}_{\xi_t\sim\tilde{p}}\left(\|\xi_t\| < \frac{c}{4}\right)h_c(\|\nabla f(x_t)\|) \leq \left(\frac{1}{2}v + \frac{3}{2}\right)\frac{c\times\sqrt{D_f G d\ln(\frac{1}{\delta})}}{n\epsilon} + \frac{1}{T}\sum_{t=1}^{T}W_{\nabla f(x_t),c}(\tilde{p}_t, p_t),$$

where $h_c(y) := \min(y^2, \frac{3}{4}cy)$; $D_f := f(x_1) - \min_x f(x)$; $W_{v,c}(p,p')$ is the Wasserstein distance between p and p' with metric function

$$d_{\nabla f(x_t),c}(a,b) := |\langle\nabla f(x_t), \mathit{clip}(\nabla f(x_t) + a, c)\rangle - \langle\nabla f(x_t), \mathit{clip}(v+b,c)\rangle|.$$

Remark on the Wasserstein distance. In (4.6), it is clear that the convergence bias b_t can be bounded by the total variation distance between p_t and $\tilde{p}_t$ or some similar distance between distributions such as the one below

$$\begin{aligned}-b_t &= \int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle(\tilde{p}_t(\xi_t) - p_t(\xi_t))d\xi_t \\ &\leq c\cdot\|\nabla f(x_t)\|\int|p_t(\xi_t) - \tilde{p}_t(\xi_t)|d\xi_t. \end{aligned} \tag{4.9}$$

However, the above bound (or the total variation distance) becomes trivial when p_t is the empirical distribution over a finite sample, because it is always 2 (always 1 for the total variation distance) when $\tilde{p}$ is continuous. In addition, the bias is hard to interpret without further transformation. This is why we bound b_t by the Wasserstein distance as follows:

$$\begin{aligned}-b_t &= \int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle(\tilde{p}(\xi_t) - p(\xi_t))d\xi_t \\ &= \int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle\tilde{p}(\xi_t)d\xi_t - \int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t', c)\rangle p(\xi_t')d\xi_t' \\ &= \int\int(\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle - \langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t', c)\rangle)\gamma(\xi_t, \xi_t')d\xi_t d\xi_t' \\ &\leq \int\int|\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle - \langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t', c)\rangle|\gamma(\xi_t, \xi_t')d\xi_t d\xi_t', \end{aligned} \tag{4.10}$$

where $\gamma(\cdot,\cdot)$ is any joint distribution with marginal $\tilde{p}(\cdot)$ and $p(\cdot)$. Thus, we have

$$-b_t \leq \inf_{\gamma\in\Gamma(\tilde{p},p)}\int\int|\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t, c)\rangle - \langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + \xi_t', c)\rangle|\gamma(\xi_t, \xi_t')d\xi_t d\xi_t',$$

where $\Gamma(\tilde{p}, p)$ is the set of all couplings with marginals $\tilde{p}$ and p on the two factors, respectively. If we define the distance function

$$d_{y,c}(a, b) := |\langle y, \text{clip}(y + a, c)\rangle - \langle y, \text{clip}(y + b, c)\rangle|.$$

Then we have

$$-b_t \leq \inf_{\gamma \in \Gamma(\tilde{p}, p)} \int \int d_{\nabla f(x_t), c}(\xi_t, \xi_t')\gamma(\xi_t, \xi_t') d\xi_t d\xi_t'. \tag{4.11}$$

The right hand side (r.h.s.) of the above inequality is the Wasserstein distance defined on the distance function $d_{\nabla f(x_t), c}$. It converges to the distance between the population distribution of gradient and $\tilde{p}$ with n being large since the empirical distribution will be similar to the population distribution.

Thus, if the population distribution of gradient is approximate symmetric, the bias term tends to be small. In addition, the distance function is uniformly bounded by $\|\nabla f(x)\|c$ which makes it is more favorable than ℓ_2 distance. Compared with the expression of b_t in Corollary 3, the Wasserstein distance is easier to interpret when $\tilde{p}$ is discrete.

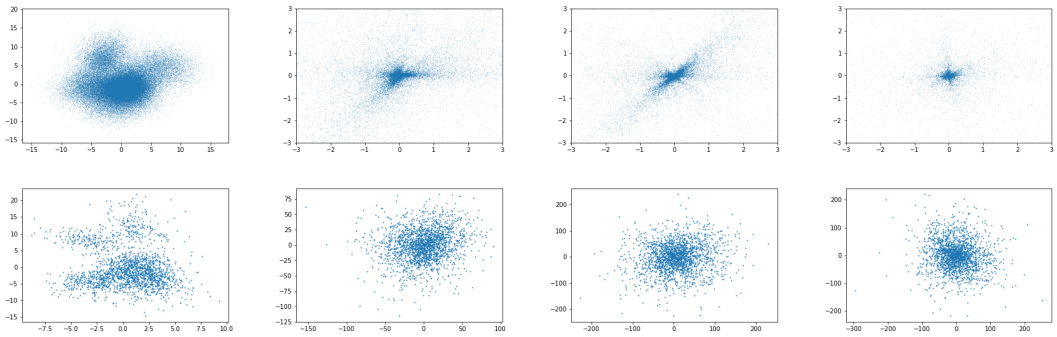
4.4 Experiments

In this section, we investigate whether the gradient distributions of DP-SGD are approximate symmetric in practice. However, since the gradient distributions are high-dimensional, certifying symmetricity is in general intractable. We instead consider two simple proxy measures and visualizations.

Setup. We run DP-SGD implemented in Tensorflow [3] on two popular datasets MNIST [LeCun et al., 2010] and CIFAR-10 [Krizhevsky and Hinton, 2009]. For MNIST, we train a CNN with two convolution layers with 16 4×4 kernels followed by a fully connected layer with 32 nodes. We use DP-SGD to train the model with $\alpha = 0.15$, and a batchsize of 128. For CIFAR-10, we train a CNN with two convolutional layers with 2×2 max pooling of stride 2 followed by a fully connected layer, all using ReLU activation, each layer uses a dropout rate of 0.5. The two convolution layers respectively has 32 and

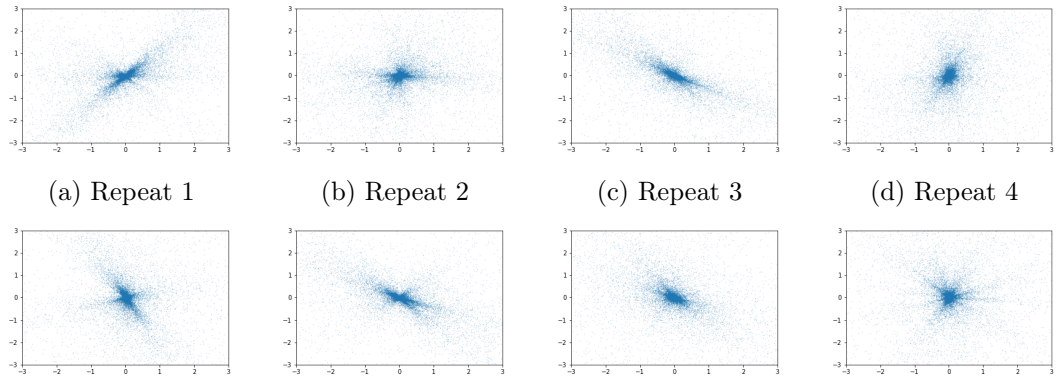
[3] https://github.com/tensorflow/privacy/tree/master/tutorials

64 kernels, each of size $3\times$ 3; further the fully connected layer has $1,500$ nodes. We use $\alpha = 0.001$ and decrease it by 10 times every 20 epochs. The clip norm of both experiments is set to be $c = 1$ and the noise multiplier is 1.1.



(a) Epoch 0 (b) Epoch 5 (c) Epoch 10 (d) Epoch 59

Figure 4.1: Gradient distributions on MNIST (top row) and CIFAR10 (bottom row) at the end of different epochs (indexed by columns). The gradients for epoch 0 are computed at initialization (before training).



(a) Repeat 1 (b) Repeat 2 (c) Repeat 3 (d) Repeat 4

(e) Repeat 5 (f) Repeat 6 (g) Repeat 7 (h) Repeat 8

Figure 4.2: Gradient distributions on MNIST at the end of epoch 9 projected using different random matrices.

4.4.1 Visualization with random projections.

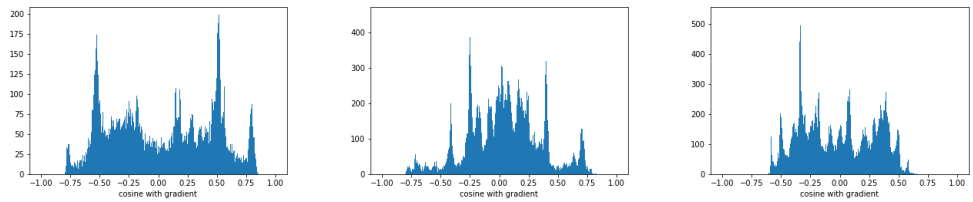
We visualize the gradient distribution by projecting the gradient to a two-dimensional space using random Gaussian matrices. Note that given any symmetric distribution,

its two-dimensional projection remains symmetric for any projection matrix. On the contrary, if for all projection matrix, the projected gradient distribution is symmetric, the original gradient distribution should also be symmetric. We repeat the projection using different randomly generated matrices and visualize the induced distributions.

We can see that on both datasets, the gradient distribution is non-symmetric before training (Epoch 0), but over the epochs, the gradient distributions become increasingly symmetric. The distribution of gradients on MNIST at the end of epoch 9 projected to a random two-dimensional space using different random matrices is shown in Figure 4.2. It can be seem that the approximate symmetric property holds for all 8 realizations. We provide many more visualizations from different realized random projections across different epochs in the Appendix.

4.4.2 Symmetricity of angles.

We also measure the cosine similarities between per-sample stochastic gradients and the true gradient. We observe that the cosine similarities between per-sample stochastic gradients and the true gradient, defined as $\cos(\nabla f(x_t) + \xi_{t,i}, \nabla f(x_t))$, is approximate symmetric around 0 as shown in the histograms in Figure 4.3.



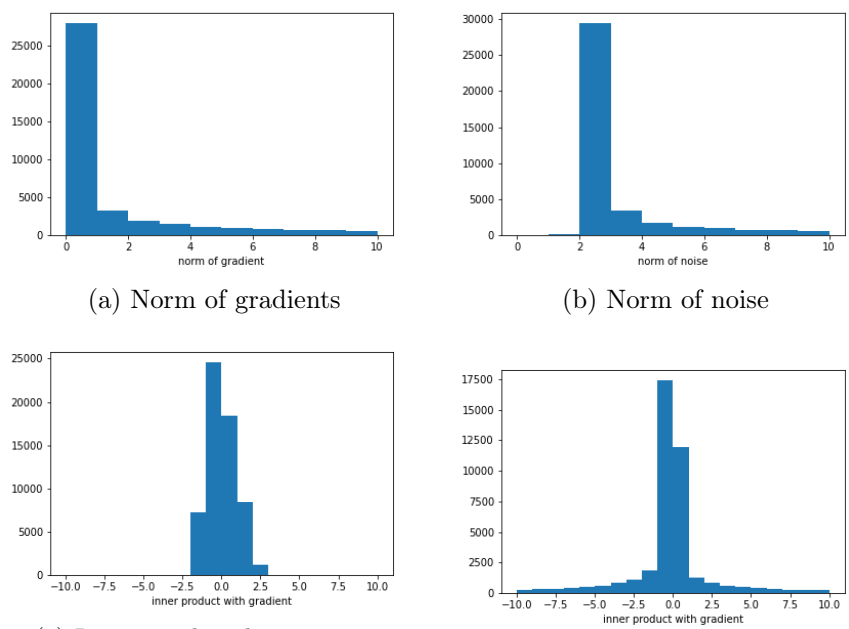
(a) Epoch 4 (b) Epoch 10 (c) Epoch 59

Figure 4.3: Histogram of cosine between stochastic gradients and the true gradient at the end of different epochs.

4.4.3 Evaluation on the probability term.

In this section, we evaluate the probability term in Corollary 3 using a few statistics of the empirical gradient distribution on MNIST. Specifically, at the end of different epochs, we plot histogram of norm of stochastic gradient and norm of noise, along with the inner product between stochastic gradient (and clipped stochastic gradient) and the

true gradient. The results are shown in Figure 4.4 –4.6. One observation is that the norm of stochastic gradients is concentrated around 0 while having a heavy tail. The noise distribution is concentrated around some positive value with a heavy tail, the mode of the noise actually corresponds to the approximate 0 norm mode of stochastic gradients. As the training progresses, the norm of stochastic gradients and the norm of noise are approaching 0. We set clipping threshold to be 1 in the experiment, so actually the probability $\mathbb{P}(\|\xi_t\| \leq \frac{1}{4}c)$ is 0 for the empirical distribution p. When we use a distribution $\tilde{p}$ with $\mathbb{P}(\|\xi_t\| \leq \frac{1}{4}c) \geq l$ for some value $l > 0$ to approximate p, this approximation indeed can create an approximation bias. However, the bias may not be too large since the mode of the norm of noise is not too much bigger than $\frac{c}{4}$. Furthermore, in Corollary 3 and Theorem 5, we actually can change $\mathbb{P}_{\xi_t \sim \tilde{p}}(\|\xi_t\| \leq \frac{1}{4}c)$ to $\mathbb{P}_{\xi_t \sim \tilde{p}}(\|\xi_t\| \leq zc)$ with any $z < 1$ and simultaneously change the $\frac{3}{4}c$ to $(1-z)c$ to make the probability term larger.



(a) Norm of gradients

(b) Norm of noise

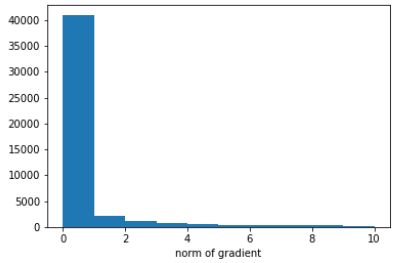
(c) Inner product between true gradient and clipped stochastic gradients

(d) Inner product between true gradient and stochastic gradients

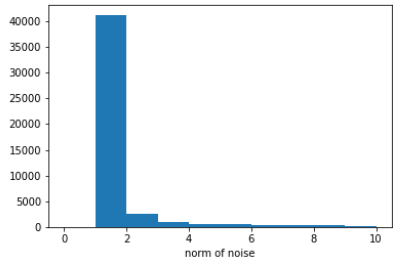
Figure 4.4: Distribution of different statistics at epoch 3.

Despite the discussions above, the distribution of the norm of stochastic gradients and the norm of the noise combined with the 2d visualization experiments implies the

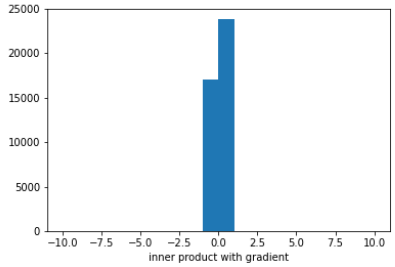
noise on gradient might follow a mixture of distributions with each component being approximate symmetric. Specifically, one component may correspond to an approximate zero mean distribution of stochastic gradients. Intuitively this can be true since each class of data may corresponds to a few variations of stochastic gradients and the gradient noise is observed to be low rank in Li et al. [2020]. We have some discussions in Section 4.2.2 to explain how convergence can be achieved in the cases of symmetric distribution mixtures but it may not be the complete explanation here. Further exploration of gradient distribution in practice might be an important question.



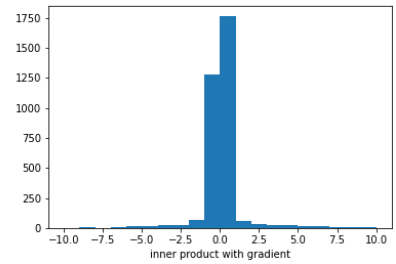
(a) Norm of gradients



(b) Norm of noise

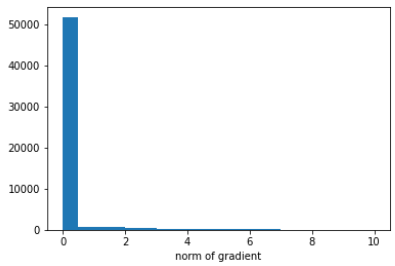


(c) Inner product between true gradient and clipped stochastic gradients

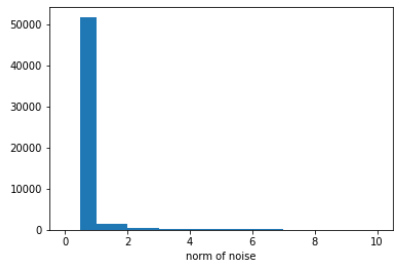


(d) Inner product between true gradient and stochastic gradients

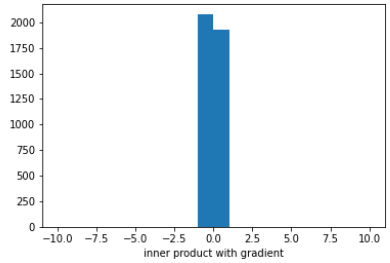
Figure 4.5: Distribution of different statistics at epoch 9.



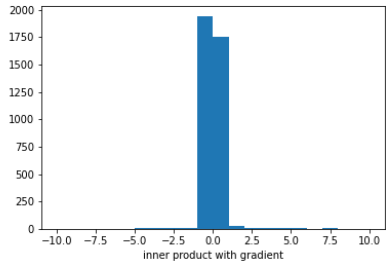
(a) Norm of gradients



(b) Norm of noise



(c) Inner product between true gradient and clipped stochastic gradients

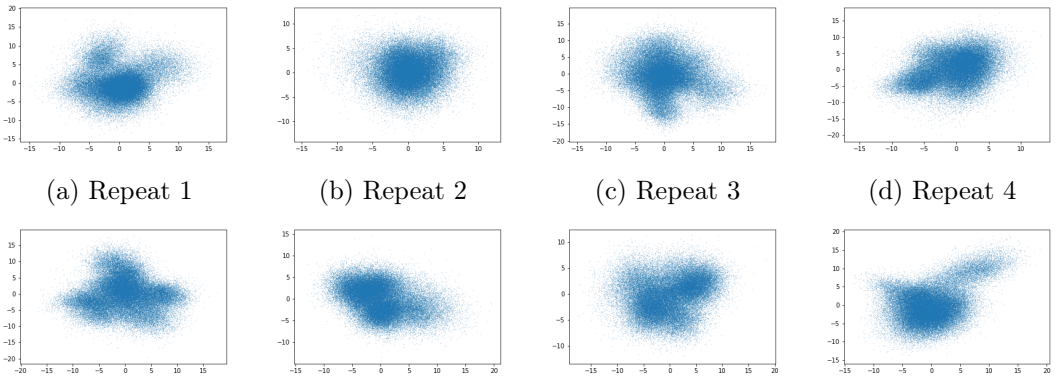


(d) Inner product between true gradient and stochastic gradients

Figure 4.6: Distribution of different statistics at epoch 59.

4.4.4 Repetition of random projection

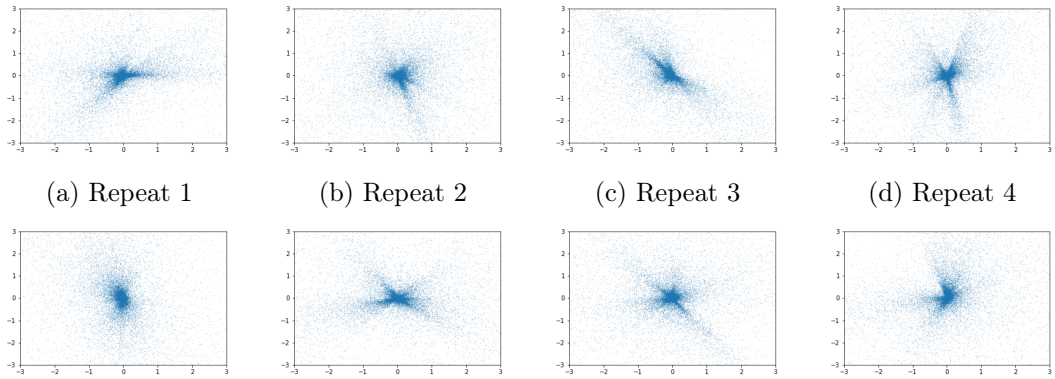
In this section, we show the projection of stochastic gradients into 2d spaces described in Section 4.4 for different projection matrices in Figure 4.7 –4.10. It can be seen that as the training progresses, the gradient distribution in 2d space tends to be increasingly more symmetric.



(a) Repeat 1 (b) Repeat 2 (c) Repeat 3 (d) Repeat 4

(e) Repeat 5 (f) Repeat 6 (g) Repeat 7 (h) Repeat 8

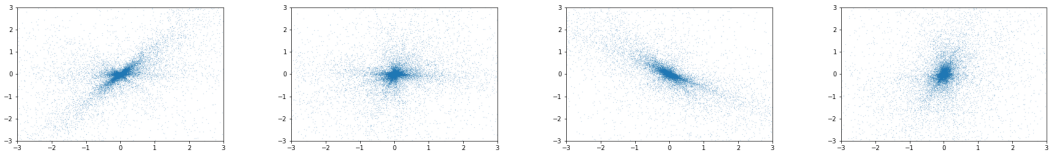
Figure 4.7: Distribution of gradients on MNIST after epochs 0 projected using different random matrices.



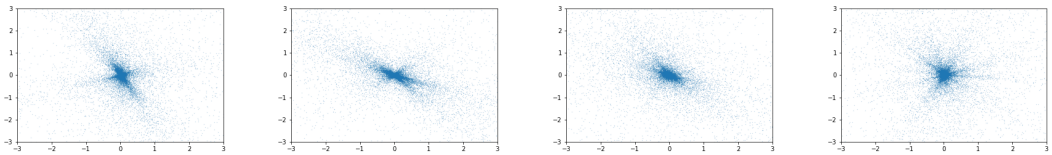
(a) Repeat 1 (b) Repeat 2 (c) Repeat 3 (d) Repeat 4

(e) Repeat 5 (f) Repeat 6 (g) Repeat 7 (h) Repeat 8

Figure 4.8: Distribution of gradients on MNIST after epochs 3 projected using different random matrices.

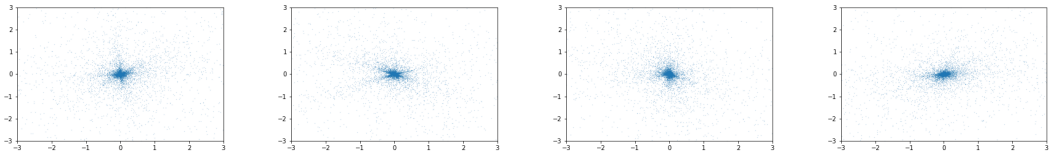


(a) Repeat 1 (b) Repeat 2 (c) Repeat 3 (d) Repeat 4

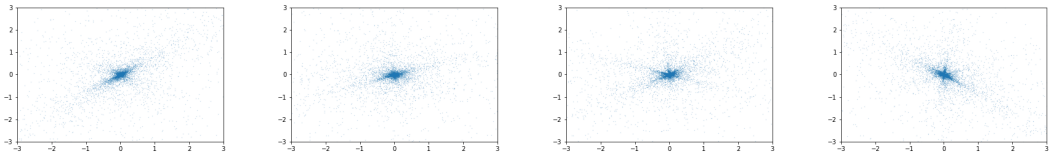


(e) Repeat 5 (f) Repeat 6 (g) Repeat 7 (h) Repeat 8

Figure 4.9: Distribution of gradients on MNIST after epochs 9 projected using different random matrices.



(a) Repeat 1 (b) Repeat 2 (c) Repeat 3 (d) Repeat 4



(e) Repeat 5 (f) Repeat 6 (g) Repeat 7 (h) Repeat 8

Figure 4.10: Distribution of gradients on MNIST after epochs 59 projected using different random matrices.

4.5 Mitigating Clipping Bias with Perturbation

From previous analyses, SGD with gradient clipping and DP-SGD have good convergence performance when the gradient noise distribution is approximately symmetric or when the gradient bias favors convergence (e.g., mixture of symmetric distributions with aligned mean). Although in practice, gradient distributions do exhibit (approximate) symmetry (see Sec. 4.4), it would be desirable to have tools to handle situations where the clipping bias does not favor convergence. Now we provide an approach to decrease the bias. If one adds some Gaussian noise before clipping, i.e.

$$g_t = \text{clip}(\nabla f(x_t) + \xi_t + k\zeta_t, c), \zeta_t \sim \mathcal{N}(0, I), \tag{4.12}$$

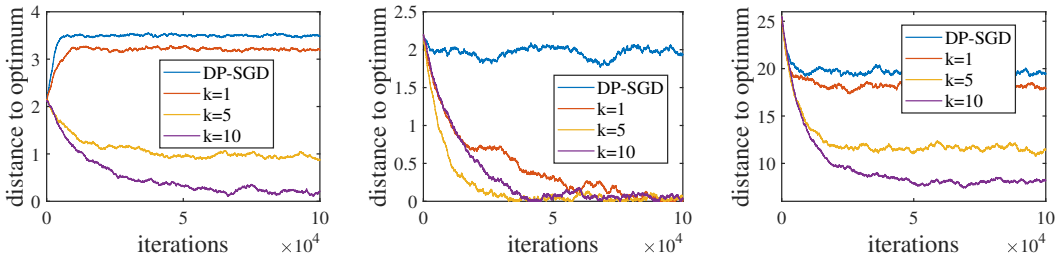
we can prove $|b_t| = O\left(\frac{\sigma_{\xi_t}^2}{k^2}\right)$ as in Theorem 9.

Theorem 9. *Let $g_t = clip(\nabla f(x_t) + \xi_t + k\zeta_t, c)$ and $\zeta_t \sim \mathcal{N}(0, I)$. Then gradient clipping algorithm has following properties:*

$$\mathbb{E}_{\xi_t \sim p, \zeta_t}[\langle \nabla f(x_t), g_t \rangle] \geq \|\nabla f(x_t)\| \min\left\{\|\nabla f(x_t)\|, \frac{3}{4}c\right\} \mathbb{P}\left(\|k\zeta_t\| < \frac{c}{4}\right) - \|\nabla f(x_t)\| O\left(\frac{\sigma_{\xi_t}^2}{k^2}\right), \tag{4.13}$$

where $\sigma_{\xi_t}^2$ is the variance of the gradient noise ξ_t.

More discussion can be found in the Appendix. By adding the noise, one trades off bias with variance. Larger noise makes the algorithm converges possibly slower but better. This trick can be helpful when the gradient distribution is not favorable. To verify its effect in practice, we run SGD with gradient clipping on a few unfavorable problems including examples in Section 4.1 and a new high dimensional example. For the new example, we minimize the function $f(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\|x - z_i\|^2$ with $n = 10000$. Each z_i is drawn from a mixture of isotropic Gaussian with 3 components of dimension 10. The covariance matrix of all components is I and the means of the 3 components are drawn from $\mathcal{N}(0, 36I)$, $\mathcal{N}(0, 4I)$, $\mathcal{N}(0, I)$, respectively. We set $\alpha = 0.1$ for the new examples and $\alpha = 0.001$ for the examples in Section 4.1. Figure **??** shows $\|x_t - \text{argmin}_x f(x)\|$ versus t. We can see that SGD with gradient clipping converges to non-optimal points as predicted by theory. In contrast, pre-clipping perturbation ensures convergence.



(a) Example 1 (b) Example 2 (b) Synthetic data

4.6 Conclusion

In this paper, we provide a theoretical analysis on the effect of gradient clipping in SGD and private SGD. We provide a new way to quantify the clipping bias by coupling the gradient distribution with a geometrically symmetric distribution. Combined with our empirical evaluation showing that gradient distribution of private SGD follows some symmetric structure along the trajectory, these results provide an explanation why gradient clipping works in practice. We also provide a perturbation-based technique to reduce the clipping bias even for adversarial instances.

4.7 Delayed Results and Proofs

4.7.1 Proof of Theorem 4

By smoothness assumption, we have

$$f(x_{t+1}) \leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{G}{2}\|x_{t+1} - x_t\|^2. \tag{4.14}$$

Then, by update rule and the fact that $\|g_t\| \leq c$, we have

$$\begin{aligned} f(x_{t+1}) \leq & f(x_t) - \alpha\langle \nabla f(x_t), g_t \rangle + \frac{G\alpha^2}{2}\|g_t\|^2 \\ \leq & f(x_t) - \alpha\langle \nabla f(x_t), g_t \rangle + \frac{G\alpha^2 c^2}{2}. \end{aligned} \tag{4.15}$$

Take expectation, sum over t from 1 to T, divide both sides by $T\alpha$, rearranging and substituting into $\alpha = \frac{1}{\sqrt{T}}$, we get

$$\begin{aligned} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(x_t), g_t \rangle] \leq & \frac{1}{T\alpha}(f(x_1) - f(x_{T+1})) + \frac{G\alpha c^2}{2} \\ \leq & \frac{1}{\sqrt{T}}\mathbb{E}[f(x_1) - f(x_{T+1})] + \frac{Gc^2}{2\sqrt{T}} \\ \leq & \frac{1}{\sqrt{T}}D_f + \frac{Gc^2}{2\sqrt{T}}, \end{aligned} \tag{4.16}$$

where $D_f = f(x_1) - \min_x f(x)$. **Q.E.D.**

4.7.2 Proof of Theorem 5

In the proof, we assume $\xi_t \sim \tilde{p}_t$, and we omit subscript of $\mathbb{P}$ and $\mathbb{E}$ to simplify notations. Further, we will denote $\bar{g}_t \triangleq \nabla f(x_t)$.

When gradient is small

Let us first consider the case with $\|\bar{g}_t\| \leq \frac{3}{4}c$.

Denote B to be the event that $\|\bar{g}_t + \xi_t\| \leq c$ and $\|\bar{g}_t - \xi_t\| \leq c$, we have $\mathbb{P}(B) \geq \mathbb{P}(\|\xi_t\| \leq \frac{c}{4})$. Define $D = \{\xi : \|\bar{g}_t + \xi_t\| > c \text{ or } \|\bar{g}_t - \xi_t\| > c\}$. Taking an expectation

conditioning on x_t, we have

$$\begin{aligned}\mathbb{E}[\langle \bar{g}_t, g_t\rangle] =& \langle \bar{g}_t, \mathbb{E}[\text{clip}(\bar{g}_t + \xi_t, c)]\rangle \\ =& \left\langle \bar{g}_t, \mathbb{E}\left[\text{clip}(\bar{g}_t + \xi_t, c)\middle| B\right]\right\rangle \mathbb{P}(B) + \left\langle \bar{g}_t, \int_D \text{clip}(\bar{g}_t + \xi_t, c)\tilde{p}(\xi_t)d\xi_t\right\rangle \\ \geq& \|\bar{g}_t\|^2 \mathbb{P}\left(\|\xi_t\| \leq \frac{c}{4}\right) + \underbrace{\left\langle \bar{g}_t, \int_D \text{clip}(\bar{g}_t + \xi_t, c)\tilde{p}(\xi_t)d\xi_t\right\rangle}_{T_1},\end{aligned}$$

where the last inequality is due to $\text{clip}(\bar{g}_t + \xi_t, c) = \bar{g}_t + \xi_t$ when B happens and $\mathbb{P}(B) \geq \mathbb{P}(\|\xi_t\| \leq \frac{c}{4})$ and the assumption that $\tilde{p}(\xi) = \tilde{p}(-\xi)$.

Now we need to analyze T_1. We have

$$\begin{aligned}T_1 =& \frac{1}{2}\left(\left\langle \bar{g}_t, \int_D \text{clip}(\bar{g}_t + \xi_t, c)\tilde{p}(\xi_t)d\xi_t\right\rangle + \left\langle \bar{g}_t, \int_D \text{clip}(\bar{g}_t - \xi_t, c)\tilde{p}(\xi_t)d\xi_t\right\rangle\right) \\ =& \frac{1}{2}\left\langle \bar{g}_t, \int_D \left(\text{clip}(\bar{g}_t + \xi_t, c) + \text{clip}(\bar{g}_t - \xi_t, c)\right)\tilde{p}(\xi_t)d\xi_t\right\rangle \\ =& \frac{1}{2}\|\bar{g}_t\| \times \int_D \underbrace{\left(\|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t + \xi_t) + \|\text{clip}(\bar{g}_t - \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t - \xi_t)\right)}_{T_2(\xi_t)} \tilde{p}(\xi_t)d\xi_t,\end{aligned} \tag{4.17}$$

where the last equality is because $\langle a, b\rangle = \|a\|\|b\| \cos(a, b)$ for any vector a, b, and that the clipping operation keeps directions.

Now we will show that $T_2(\xi_t) \geq 0$. Towards this end, let us consider three cases.

Case I. Suppose $||\bar{g}_t + \xi_t|| \geq c$ and $||\bar{g}_t - \xi_t|| \geq c$. In this case, we have

$$T_2(\xi) = c \cdot (\cos(\bar{g}_t, \bar{g}_t + \xi_t) + \cos(\bar{g}_t, \bar{g}_t - \xi_t)) \geq 0, \tag{4.18}$$

where the inequality is due to Lemma 17.

Case II. One of $\|\bar{g}_t + \xi_t\|$ and $\|\bar{g}_t - \xi_t\|$ is less than c.

Case II (a). First we assume $\cos(\bar{g}_t, \bar{g}_t - \xi_t) < 0$. Then we must have $\cos(\bar{g}_t, -\xi_t) < 0$ so that $\langle \bar{g}_t, -\xi_t\rangle < 0$ and $\cos(\bar{g}_t, \bar{g}_t + \xi_t) = \frac{\langle \bar{g}_t, \bar{g}_t + \xi_t\rangle}{\|\bar{g}_t\|\|\bar{g}_t + \xi_t\|} > 0$. Then, from Lemma 18, we have

$$\|\bar{g}_t + \xi_t\| \geq \|\bar{g}_t - \xi_t\|, \tag{4.19}$$

and it follows that $\|\bar{g}_t - \xi_t\| \leq c$. So in this case, we have

$$\begin{aligned}
T_2(\xi_t) =& \|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t + \xi_t) + \|\text{clip}(\bar{g}_t - \xi, c)\| \cos(\bar{g}_t, \bar{g}_t - \xi) \\
=& c \cdot \cos(\bar{g}_t, \bar{g}_t + \xi_t) + \|\text{clip}(\bar{g}_t - \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t - \xi_t) \\
\geq& c \cdot \cos(\bar{g}_t, \bar{g}_t + \xi_t) + c \cdot \cos(\bar{g}_t, \bar{g}_t - \xi_t) \\
\geq& 0,
\end{aligned} \tag{4.20}$$

where the last inequality is due to Lemma 17.

Case II (b). Similar argument applies to the case with $\cos(\bar{g}_t, \bar{g}_t + \xi_t) < 0$.

Case II (c). Suppose $\cos(\bar{g}_t, \bar{g}_t + \xi_t) \geq 0, \cos(\bar{g}_t, \bar{g}_t - \xi_t) \geq 0$. Then $T_2(\xi_t) \geq 0$ holds trivially.

In summary, we have shown that the following holds:

$$\mathbb{E}[\langle \bar{g}_t, g_t \rangle] \geq \|\bar{g}_t\|^2 \mathbb{P}\left(\|\xi_t\| \leq \frac{c}{4}\right). \tag{4.21}$$

This completes the proof. **Q.E.D.**

When gradient is large

Now let us look at the case where gradient is large, i.e. $\|\bar{g}_t\| \geq \frac{3}{4}c$.

By definition, we have

$$\begin{aligned}
\mathbb{E}[\langle \bar{g}_t, g_t \rangle] =& \left\langle \bar{g}_t, \int_{\xi_t} \text{clip}(\bar{g}_t + \xi_t, c) \cdot p(\xi_t) d\xi \right\rangle \\
=& \int_{\xi_t} \langle \bar{g}_t, \text{clip}(\bar{g}_t + \xi_t, c) \rangle \cdot p(\xi_t) d\xi_t \\
=& \|\bar{g}_t\| \int_{\xi_t} \|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, \text{clip}(\bar{g}_t + \xi_t), c) \cdot p(\xi_t) d\xi_t \\
\overset{(i)}{=}& \|\bar{g}_t\| \int_{\xi_t} \|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, (\bar{g}_t + \xi_t)) \cdot p(\xi_t) d\xi_t \\
\overset{(ii)}{=}& \|\bar{g}_t\| \underbrace{\int_{\xi_t} \|\text{clip}(G(\bar{g}_t + \xi_t), c)\| \cos(G\bar{g}_t, G(\bar{g}_t + \xi_t)) \cdot p(\xi_t) d\xi_t}_{T_3},
\end{aligned} \tag{4.22}$$

where in (i) we used the fact that clipping operation keeps the direction, that is:

$$\frac{\text{clip}(\bar{g}_t + \xi, c)}{\|\text{clip}(\bar{g}_t + \xi, c)\|} = \frac{\bar{g}_t + \xi}{\|\bar{g}_t + \xi\|}, \ \forall\ \xi,\ \forall\ c > 0. \tag{4.23}$$

In (ii) we have introduced an arbitrary rotation matrix G, and we have used the fact that the angle between two vectors remains the same after the same rotation, and that the norm of the clip operation is rotation invariant:

$$\|\text{clip}(Gz, c)\| = \|\text{clip}(z, c)\|,\ \forall\ z \in \mathbb{R}^d,\ \forall\ c > 0. \tag{4.24}$$

In the following, we will show T_3 is a non-decreasing function of $\bar{g}_t$.

Since the rotation matrix G in T_3 is arbitrary, without loss of generality (wlog), we can assume that the first element of $\bar{g}_t$ equals $\|\bar{g}_t\|$ and the rest are all zeros, that is:

$$\bar{g}_t[1] = \|\bar{g}_t\| > 0,\ \bar{g}_t[i] = 0,\ 2 \leq i \leq d.$$

For notation simplicity, let us define $y := \|\bar{g}_t\|$. Then to show T_3 is a non-decreasing function of $\|\bar{g}_t\|$, it is sufficient to show that each term in the integration is a non-decreasing function of y. That is, for all ξ_t, the following quantity is a non-decreasing function of y for $y > 0$ when $\bar{g}_t = [y, 0, ..., 0]$:

$$\|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t + \xi_t). \tag{4.25}$$

To this end, we divide our analysis into two cases.

Case I: Suppose $\|\bar{g}_t + \xi_t\| \leq c$. In this case, (4.25) reduces to

$$\begin{aligned}\|\bar{g}_t + \xi_t\| \cos(\bar{g}_t, \bar{g}_t + \xi_t) &= \|\bar{g}_t + \xi_t\| \frac{\langle \bar{g}_t, \bar{g}_t + \xi_t \rangle}{\|\bar{g}_t\| \|\bar{g}_t + \xi_t\|} \\ &= \frac{\langle \bar{g}_t, \bar{g}_t + \xi_t \rangle}{\|\bar{g}_t\|} = \frac{y(y + \xi_{t,1})}{y} = y + \xi_{t,1}.\end{aligned} \tag{4.26}$$

Clearly, the above quantity is a monotonically increasing function of y.

Case II: Suppose $\|\bar{g}_t + \xi_t\| \geq c$. Then we have

$$\begin{aligned}\|\text{clip}(\bar{g}_t + \xi_t, c)\| \cos(\bar{g}_t, \bar{g}_t + \xi_t) =& c \cdot \cos(\bar{g}_t, \bar{g}_t + \xi_t) = c \frac{\langle \bar{g}_t, \bar{g}_t + \xi_t \rangle}{\|\bar{g}_t\| \|\bar{g}_t + \xi_t\|} \\ =& c \frac{y(y + \xi_{t,1})}{y\sqrt{(y + \xi_{t,1})^2 + \sum_{i=2}^d \xi_{t,i}^2}} \\ =& c \frac{(y + \xi_{t,1})}{\sqrt{(y + \xi_{t,1})^2 + \sum_{i=2}^d \xi_{t,i}^2}}. \end{aligned} \tag{4.27}$$

It is also easy to verify that the above function is a non-decreasing function of y.

To see it is non-decreasing, define

$$r(z) = c \frac{z}{\sqrt{z^2 + q^2}}, \text{ with } c > 0. \tag{4.28}$$

Then it is easy to check that $r'(z) = c(1 - \frac{z^2}{z^2+q^2}) \geq 0$. By letting $z = y + \xi_{t,1}$ and $q^2 = \sum_{i=2}^d \xi_{t,i}^2$, we conclude that the r.h.s. of (4.27) is non-decreasing.

Since the clipping function is continuous, combined with the above results, we know (4.25) is a non-decreasing function of $\|\bar{g}_t\|$.

Then by utilizing the above non-decreasing property, and the assumption that $\|\bar{g}_t\| \geq 3c/4$, we have the following

$$\begin{aligned}\mathbb{E}[\langle \bar{g}_t, g_t \rangle] &= \mathbb{E}[\langle \bar{g}_t, \bar{g}_t + \xi_t \rangle] \\ &= \|\bar{g}_t\| T_3 \geq \|\bar{\bar{g}}_t\| T_3 \\ &\geq \|\bar{\bar{g}}_t\| \int_{\xi_t} \left\| \text{clip}\left(\frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|} + \xi_t, c \right) \right\| \cos\left(\frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|}, \frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|} + \xi_t \right) \cdot p(\xi_t) d\xi_t \\ &= \mathbb{E}\left[\langle \frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|}, \frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|} + \xi_t \rangle \right] \\ &= \mathbb{E}\left[\langle \bar{\bar{g}}_t, \bar{\bar{g}}_t + \xi_t \rangle \right], \end{aligned} \tag{4.29}$$

where we have defined $\bar{\bar{g}}_t := \frac{3 \cdot c}{4} \frac{\bar{g}_t}{\|\bar{g}_t\|}$, with $\|\bar{\bar{g}}_t\| = \frac{3}{4} c$.

From the first part of the theorem, we know for any vector $\|z\| = \frac{3}{4} c$, the following

holds

$$\mathbb{E}[\langle z, z+\xi_t\rangle] \geq \|z\|^2 \mathbb{P}\left(\|\xi_t\| < \frac{c}{4}\right) = \|z\| \left(\frac{3}{4} c \cdot \mathbb{P}(\|\xi_t\| < \frac{c}{4})\right). \tag{4.30}$$

Combining the above result with (4.29), we obtain the following:

$$\|\bar{\bar{g}}_t\| T_3 \geq E[\langle \bar{\bar{g}}_t, g_t\rangle] \geq \|\bar{\bar{g}}_t\| \left(\frac{3}{4} c \cdot \mathbb{P}(\|\xi_t\| < \frac{c}{4})\right). \tag{4.31}$$

This implies $T_3 \geq \frac{3}{4} c \cdot \mathbb{P}(\|\xi\| < \frac{c}{4})$. So we obtain

$$\mathbb{E}[\langle \bar{g}_t, g_t\rangle] = \mathbb{E}[\langle \bar{g}_t, \bar{g}_t + \xi_t\rangle] = \|\bar{g}_t\| T_3 \geq \|\bar{g}_t\| \frac{3 \cdot c}{4} \cdot \mathbb{P}\left(\|\xi\| < \frac{c}{4}\right). \tag{4.32}$$

The proof is completed. **Q.E.D.**

Technical lemmas

Lemma 17. *For any g and ξ, we have*

$$\cos(g, g+\xi) + \cos(g, g-\xi) \geq 0.$$

Proof: By definition of the cosine function, we have

$$\begin{aligned}
&\cos(g, g+\xi) + \cos(g, g-\xi) \\
=&\frac{\langle g, g+\xi\rangle}{\|g\|\|g+\xi\|} + \frac{\langle g, g-\xi\rangle}{\|g\|\|g-\xi\|} \\
=&\frac{\|g\|}{\|g+\xi\|} + \frac{\|g\|}{\|g-\xi\|} + \frac{\langle g, \xi\rangle}{\|g\|\|g+\xi\|} - \frac{\langle g, \xi\rangle}{\|g\|\|g-\xi\|} \\
=&\frac{\|g\|}{\|g+\xi\|} + \frac{\|g\|}{\|g-\xi\|} + \frac{\|\xi\| \cos(g, \xi)}{\|g+\xi\|} - \frac{\|\xi\| \cos(g, \xi)}{\|g-\xi\|} \\
=&\frac{\|g+\xi\|(\|g\| - \|\xi\| e) + \|g-\xi\|(\|g\| + \|\xi\| e)}{\|g+\xi\|\|g-\xi\|},
\end{aligned} \tag{4.33}$$

where in the last equality we have defined $e = \cos(g, \xi)$.

To prove the desired result, we only need the numerator of r.h.s. of (4.33) to be non-negative.

Denote $h(\xi) = \cos(g, g+\xi) + \cos(g, g-\xi)$, since h is rotation invariant, we can assume

wlog that $\xi_1 = \|\xi_1\| > 0$ and $\xi_{t,i} = 0$ for $2 \le i \le d$. Also, because $h(\xi) = h(-\xi)$, we can assume wlog that $g_1 \ge 0$. Now suppose $g_1 = a > 0$, $\sum_{i=2}^d g_i^2 = b^2$, Denote the numerator of r.h.s. of (4.33) as T_4, it can be written as

$$\begin{aligned} T_4 =& \|g+\xi\|(\|g\| - \|\xi\|e) + \|g-\xi\|(\|g\| + \|\xi\|e) \\ =& \underbrace{\sqrt{b^2+(a+\xi_1)^2}(\sqrt{a^2+b^2} - \xi_1 e)}_{T_5} + \underbrace{\sqrt{b^2+(a-\xi_1)^2}(\sqrt{a^2+b^2} + \xi_1 e)}_{T_6}, \end{aligned}$$

where we have defined

$$e := \cos(g,\xi) = \frac{\langle g, \xi\rangle}{\|g\|\|\xi\|} = \frac{a}{\sqrt{a^2+b^2}} \ge 0. \tag{4.34}$$

Now let us analyze the sign of T_4. Recall that by assumption, $\xi_1 > 0$ and $e \ge 0$. Then we know $T_6 \ge 0$. We have $T_4 \ge 0$ trivially when $T_5 \ge 0$, i.e. when $\xi_1 e \le \sqrt{a^2+b^2}$.

Now assume $\xi_1 e > \sqrt{a^2+b^2}$. Below we will show that $T_6^2 \ge T_5^2$, which implies that $T_4 \ge 0$. To this end, have we calculate the differences of T_6^2 and T_5^2 as:

$$\begin{aligned} T_6^2 - T_5^2 =& (b^2+(a-\xi_1)^2)(\sqrt{a^2+b^2}+\xi_1 e)^2 - (b^2+(a+\xi_1)^2)(\sqrt{a^2+b^2}-\xi_1 e)^2 \\ =& 4b^2\xi_1 e\sqrt{a^2+b^2} + \underbrace{4\xi_1 e\sqrt{a^2+b^2}(a^2+\xi_1^2) - 4a\xi_1(a^2+b^2+\xi_1^2 e^2)}_{T_7}. \end{aligned}$$

For T_7, we can further simplify it as

$$\begin{aligned} T_7 =& 4\xi_1 e\sqrt{a^2+b^2}(a^2+\xi_1^2) - 4a\xi_1(a^2+b^2+\xi_1^2e^2) \\ \overset{(4.34)}{=}& 4\xi_1 a(a^2+\xi_1^2) - 4a\xi_1(a^2+b^2+\xi_1^2e^2) \\ =& 4\xi_1 a(\xi_1^2(1-e^2) - b^2) \\ \overset{(4.34)}{=}& 4\xi_1 a\left(\xi_1^2(\frac{b^2}{a^2+b^2}) - b^2\right) \\ =& 4\xi_1 a\left(b^2\cdot\frac{\xi_1^2-(a^2+b^2)}{a^2+b^2}\right) \\ \ge& 0, \end{aligned}$$

where the last inequality is because $\xi^2 \ge \xi^2 e^2$, and our assumption that $\xi^2 e^2 \ge a^2+b^2$ and $\xi_1 a > 0$.

Combining all above, we have

$$T_7 \geq 0 \implies T_6^2 - T_5^2 \geq 0 \implies T_4 \geq 0 \implies \cos(g, g+\xi) + \cos(g, g-\xi) \geq 0.$$

This completes the proof. **Q.E.D.**

Lemma 18. *For any g and ξ, we have*

$$\|g+\xi\| \geq \|g-\xi\|, \quad \textit{if } \cos(g,\xi) > 0, \tag{4.35a}$$

$$\|g+\xi\| \leq \|g-\xi\|, \quad \textit{if } \cos(g,\xi) < 0. \tag{4.35b}$$

Proof: Let us express ξ using a coordinate system with one axis parallel to g. Define the basis of this coordinate system as $v_1, v_2, ...v_d$ with $v_1 = g/\|g\|$. Then we have $\xi = \sum_{i=1}^d b_i v_i$ and $\cos(g,\xi) > 0$ if and only if $b_1 > 0$. In addition, we have

$$\|g+\xi\| = \sqrt{(\|g\| + b_1)^2 + \sum_{i=2}^d b_i^2},$$

and

$$\|g-\xi\| = \sqrt{(\|g\| - b_1)^2 + \sum_{i=2}^d b_i^2}.$$

Then it is clear that $\|g+\xi\| \geq \|g-\xi\|$ when $b_1 > 0$. This completes the proof of of (4.35a).

Similar arguments applies to the case with $\cos(g,\xi) < 0$. **Q.E.D.**

4.7.3 Proof of Theorem 6

Theorem 6. *Given m distributions with the pdf of the ith distribution being $p_i(\xi_t) = \phi_i(\|\xi_t - u_i\|)$ for some function ϕ_i. If $\nabla f(x_t) = \sum_{i=1}^m w_i u_i$ for some $w_i \geq 0, \sum_{i=1}^m w_i = 1$. Let $p'(\xi_t) = \sum_{i=1}^m w_i p_i(\xi_t - \nabla f(x_t))$, be a mixture of these distributions with zero mean.*

If $\langle u_i, \nabla f(x_t)\rangle \geq 0, \forall i \in [m]$*, we have:*

$$\mathbb{E}_{\xi_t \sim p'}[\langle \nabla f(x_t), g_t\rangle] \geq \|\nabla f(x_t)\| \sum_{i=1}^{m} w_i \min\left(\|u_i\|, \frac{3}{4}c\right) \cos(\nabla f(x_t), u_i) \mathbb{P}_{\xi_t \sim p_i}\left(\|\xi_t\| < \frac{c}{4}\right) \geq 0.$$

Proof: First, we notice that Theorem 5 can be restated into a more general, which holds for any vector g instead of $\nabla f(x_t)$, as follows.

Theorem 5 (restated)**.** *Given a random variable* $\xi \sim \tilde{p}$ *with* $\tilde{p}(\xi)$ *being a symmetric distribution. Then for any vector* $g \in \mathbb{R}^d$*, we have*

1. *If* $\|g\| \leq \frac{3}{4}c$, *then* $\mathbb{E}[\langle g, \mathit{clip}(g+\xi, c)\rangle] \geq \|g\|^2 \mathbb{P}\left(\|\xi\| < \frac{c}{4}\right)$; (4.36)
2. *If* $\|g\| > \frac{3}{4}c$, *then* $\mathbb{E}[\langle g, \mathit{clip}(g+\xi, c)\rangle] \geq \frac{3 \cdot c}{4} \|g\| \mathbb{P}\left(\|\xi\| < \frac{c}{4}\right)$. (4.37)

In addition, if $\xi \sim \hat{p}$ with $\hat{p}$ being a *spherical* distribution $\hat{p}(\xi) = \phi(\|\xi\|)$ for some function ϕ, t

$$\mathbb{E}[\mathrm{clip}(g+\xi, c)] = r \cdot g, \ \forall\ g \in \mathbb{R}^d, \tag{4.38}$$

where r is some constant. To see this, consider two vectors ξ_1 and ξ_2 satisfying

$$\|\xi_1\| = \|\xi_2\|, \ \cos(\xi_1, g) = \cos(\xi_2, g), \ \sin(\xi_1, g) = -\sin(\xi_2, g). \tag{4.39}$$

Then it is easy to see that $\|\xi_1 + g\| = \|\xi_2 + g\|$, and $\xi_1 + \xi_2$ aligns with g. It follows that

$$\begin{aligned}\mathrm{clip}(g+\xi_1, c) + \mathrm{clip}(g+\xi_2, c) &= \frac{g+\xi_1}{\|g+\xi_1\|} \cdot \min\{c, \|g+\xi_1|\|\} + \frac{g+\xi_2}{\|g+\xi_2\|} \cdot \min\{c, \|g+\xi_2|\|\} \\ &= \frac{g+\xi_1+g+\xi_2}{\|g+\xi_1\|} \cdot \min\{c, \|g+\xi_1|\|\} = \nu \cdot g,\end{aligned}$$

for some constant ν. That is, $\mathrm{clip}(g+\xi_1, c) + \mathrm{clip}(g+\xi_2, c)$ aligns with g. Now let $\xi_1 = g + v$ and $\xi_2 = g - v$, we can see that (4.39) will be satisfied. Then we can integrate such pairs over the spherical distribution $\hat{p}(\xi)$ and obtain (4.38).

Thus, the expected clipped gradient is in the same direction as g. Combining the above relation with restated Theorem 5 above, it follows that when $\tilde{p}$ is a spherical

distribution with $\tilde{p}(\xi) = \phi(\|\xi\|)$,

$$\mathbb{E}[\text{clip}(g + \xi, c)] = r \cdot g. \tag{4.40}$$

with $r \geq 0$ and

$$r \cdot \|g\| \geq \min\left(\frac{3 \cdot c}{4}, \|g\|\right) \mathbb{P}\left(\|\xi\| < \frac{c}{4}\right).$$

Now we can use the above results to prove the theorem.

The expectation can be splitted as

$$\mathbb{E}_{\xi_t \sim p'}[\langle \nabla f(x_t), g_t \rangle] = \sum_{i=1}^{m} w_i \mathbb{E}_{\xi_t \sim p_i}[\langle \nabla f(x_t), g_t \rangle].$$

Then, because (4.40) and $\mathbb{E}_{\xi_t \sim p_i}[g_t] = u_i$ and that p_i corresponds to a noise with spherical distribution added to u_i, we have

$$\mathbb{E}_{\xi_t \sim p_i}[\langle \nabla f(x_t), g_t \rangle] = \langle \nabla f(x_t), \mathbb{E}_{\xi_t \sim p_i}[g_t] \rangle = \langle \nabla f(x_t), r_i u_i \rangle,$$

with $r_i \|u_i\| \geq \min(\frac{3 \cdot c}{4}, \|u_i\|) \mathbb{P}_{\xi_t \sim p_i}\left(\|\xi_t\| < \frac{c}{4}\right)$. Since we assumed $\langle u_i, \nabla f(x_t) \rangle \geq 0$, we have

$$\mathbb{E}_{\xi_t \sim p'}[\langle \nabla f(x_t), g_t \rangle] \geq \|\nabla f(x_t)\| \sum_{i=1}^{m} w_i \min(\frac{3}{4}c, \|u_i\|) \cos(u_i, \nabla f(x_t)) \mathbb{P}_{\xi_t \sim p_i}\left(\|\xi_t\| < \frac{c}{4}\right) \geq 0.$$

This completes the proof. **Q.E.D.**

4.7.4 Proof of Theorem 8

Recall the algorithm has the following update rule

$$x_{t+1} = x_t - \alpha \left(\left(\frac{1}{|S_t|} \sum_{i \in S_t} \text{clip}(\nabla f(x_t) + \xi_{t,i}, c) \right) + Z_t \right), \tag{4.41}$$

where $g_{t,i} \triangleq \nabla f(x_t) + \xi_{t,i}$ is the stochastic gradient at iteration t evaluated on sample i, and S_t is a subset of whole dataset D; $Z_t \sim \mathcal{N}(0, \sigma^2 I)$ is the noise added for privacy.

We denote $g_t := \frac{1}{|S_t|}\sum_{i\in S_t}\text{clip}(\nabla f(x_t)+\xi_{t,i},c)$ in the remaining parts of the proof to simplify notation. It is clear that $\|g_t\|\le c$.

Following traditional convergence analysis of SGD using smoothness assumption, we first have:

$$\begin{aligned} f(x_{t+1}) &\le f(x_t)+\langle\nabla f(x_t),x_{t+1}-x_t\rangle+\frac{G}{2}\|x_{t+1}-x_t\|^2 \\ &\overset{(4.41)}{=} f(x_t)-\alpha\langle\nabla f(x_t),(g_t+Z_t)\rangle+\frac{G\alpha^2}{2}\|g_t+Z_t\|^2. \end{aligned} \tag{4.42}$$

Taking expectation conditioned on x_t, we have

$$\begin{aligned} \mathbb{E}[f(x_{t+1})|x_t] &\le f(x_t)-\alpha\mathbb{E}[\langle\nabla f(x_t),g_t\rangle|x_t]+\frac{1}{2}G\alpha^2(\mathbb{E}[\|g_t\|^2|x_t]+\sigma^2 d) \\ &\le f(x_t)-\alpha\mathbb{E}[\langle\nabla f(x_t),g_t\rangle|x_t]+\frac{1}{2}G\alpha^2(c^2+\sigma^2 d). \end{aligned} \tag{4.43}$$

Take overall expectation and sum over $t\in[T]$ and rearrange, we have

$$\sum_{t=1}^{T}\alpha\mathbb{E}[\langle\nabla f(x_t),g_t\rangle]\le f(x_1)-\mathbb{E}[f(x_{T+1})]+\frac{T}{2}G\alpha^2(c^2+\sigma^2 d). \tag{4.44}$$

Dividing both sides by $T\alpha$, we get

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\langle\nabla f(x_t),g_t\rangle]\le\frac{f(x_1)-\mathbb{E}[f(x_{T+1})]}{T\alpha}+\frac{1}{2}G\alpha(c^2+\sigma^2 d). \tag{4.45}$$

To achieve (ϵ,δ)-privacy, we need $\sigma^2=v\frac{Tc^2\ln(\frac{1}{\delta})}{n^2\epsilon^2}$ for some constant v by Theorem 1 in Abadi et al. [2016b]. Substituting the expression of σ^2 into the above inequality, we get

$$\frac{1}{T}\sum_{t=1}^{T}\langle\mathbb{E}[\nabla f(x_t),g_t\rangle]\le\frac{D_f}{T\alpha}+\frac{1}{2}G\alpha\left(c^2+v\frac{T\ln(\frac{1}{\delta})}{n^2\epsilon^2}c^2 d\right), \tag{4.46}$$

where we define $D_f := f(x_1)-\min_x f(x)$.

Setting $T\alpha = \frac{\sqrt{D_f}n\epsilon}{\sqrt{G}c\sqrt{d}\sqrt{\ln(\frac{1}{\delta})}}$, we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\langle\nabla f(x_t), g_t\rangle] \leq \left(\frac{1}{2}v+1\right)\frac{c\sqrt{D_f G d\ln(\frac{1}{\delta})}}{n\epsilon} + \frac{1}{2}G\alpha c^2. \tag{4.47}$$

Setting $\alpha = \frac{\sqrt{D_f d\ln(\frac{1}{\delta})}}{n\epsilon c\sqrt{G}}$, we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\langle\nabla f(x_t), g_t\rangle] \leq \left(\frac{1}{2}v+\frac{3}{2}\right)\frac{c\sqrt{D_f G d\ln(\frac{1}{\delta})}}{n\epsilon}. \tag{4.48}$$

The remaining step is to analyze the term on left hand side (l.h.s) of (4.48). We first notice that the gradient sampling scheme yields

$$\mathbb{E}[\langle\nabla f(x_t), g_t\rangle] = \mathbb{E}_{\xi_t\sim p}[\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle], \tag{4.49}$$

with ξ_t being a discrete random variable that can takes values $\xi_{t,i}, i \in D$ with equal probability and D is the whole dataset.

Now it is time to split the bias as following.

$$\mathbb{E}_{\xi_t\sim p}[\langle\nabla f(x_t), g_t\rangle] = \mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle] + \int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle(p_t(\xi_t)-\tilde{p_t}(\xi_t))d\xi_t,$$

with $\tilde{p}$ being a symmetric distribution. Applying Theorem 5, we have

$$\begin{aligned}\mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t,\rangle] &\geq \|\nabla f(x_t)\|^2\cdot\mathbb{P}_{\xi_t\sim\tilde{p}}(\|\xi_t\|<\frac{c}{4}), \quad \text{if } \|\nabla f(x_t)\|\leq\frac{3}{4}c\\ \mathbb{E}_{\xi_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle] &\geq \frac{3\cdot c}{4}\|\nabla f(x_t)\|\cdot\mathbb{P}_{\xi_t\sim\tilde{p}}(\|\xi_t\|<\frac{c}{4}), \quad \text{if } \|\nabla f(x_t)\|\geq\frac{3}{4}c.\end{aligned} \tag{4.50}$$

Now we bound the bias term using the Wasserstein distance as follows.

$$
\begin{aligned}
&-\int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle (p_t(\xi_t)-\tilde{p}_t(\xi_t))d\xi_t \\
=&\int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle (\tilde{p}(\xi_t)-p(\xi_t))d\xi_t \\
=&\int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle \tilde{p}(\xi_t)d\xi_t - \int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t', c)\rangle p(\xi_t')d\xi_t' \\
=&\int\int (\langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle - \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t', c)\rangle)\gamma(\xi_t,\xi_t')d\xi_t d\xi_t' \\
\le&\int\int |\langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle - \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t', c)\rangle|\gamma(\xi_t,\xi_t')d\xi_t d\xi_t', \qquad (4.51)
\end{aligned}
$$

where γ is any joint distribution with marginal $\tilde{p}$ and p. Thus, we have

$$
\begin{aligned}
&-\int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle (p_t(\xi_t)-\tilde{p}_t(\xi_t))d\xi_t \\
\le&\inf_{\gamma\in\Gamma(\tilde{p},p)}\int\int |\langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle - \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t', c)\rangle|\gamma(\xi_t,\xi_t')d\xi_t d\xi_t'
\end{aligned}
$$

where $\Gamma(\tilde{p},p)$ is the set of all coupling with marginals $\tilde{p}$ and p on the two factors, respectively. Define the distance function $d_{y,c}(a,b) = |\langle y, \mathrm{clip}(y+a,c)\rangle - \langle y, \mathrm{clip}(y+b,c)\rangle|$, we have

$$
\begin{aligned}
&-\int \langle \nabla f(x_t), \mathrm{clip}(\nabla f(x_t)+\xi_t, c)\rangle (p_t(\xi_t)-\tilde{p}_t(\xi_t))d\xi_t \\
\le&\inf_{\gamma\in\Gamma(\tilde{p},p)}\int\int d_{\nabla f(x_t),c}(\xi_t,\xi_t')\gamma(\xi_t,\xi_t')d\xi_t d\xi_t' := W_{\nabla f(x_t),c}(\tilde{p}_t,p_t), \qquad (4.52)
\end{aligned}
$$

where $W_{v,c}(p,p')$ is the Wasserstein distance between p and p' using the metric $d_{v,c}$.

In summary, define

$$
h(y) = \begin{cases} y^2, & \text{for } y \le 3c/4 \\ \frac{3\cdot c}{4}y, & \text{for } y > 3c/4 \end{cases},
$$

we have

$$
\frac{1}{T}\sum_{t=1}^{T}\mathbb{P}_{\xi_t\sim\tilde{p}_t}\left(\|\xi_t\| < \frac{c}{4}\right)h(\|\nabla f(x_t)\|) \le \left(\frac{1}{2}v+\frac{3}{2}\right)\frac{c\sqrt{D_f G d\ln(\frac{1}{\delta})}}{n\epsilon} + \frac{1}{T}\sum_{t=1}^{T}W_{\nabla f(x_t),c}(\tilde{p}_t,p_t). \qquad (4.53)
$$

This completes the proof. **Q.E.D.**

4.7.5 Proof of Theorem 9

Theorem 9. *Let $g_t := clip(\nabla f(x_t) + \xi_t + k\zeta_t, c)$ and $\zeta_t \sim \mathcal{N}(0, I)$. Then gradient clipping algorithm has following properties:*

$$\mathbb{E}_{\xi_t \sim p, \zeta_t}[\langle \nabla f(x_t), g_t \rangle] \geq \|\nabla f(x_t)\| \min\left\{\|\nabla f(x_t)\|, \frac{3}{4}c\right\} \mathbb{P}(\|k\zeta_t\| < \frac{c}{4}) - \|\nabla f(x_t)\| O\left(\frac{\sigma_{\xi_t}^2}{k^2}\right), \tag{4.54}$$

where $\sigma_{\xi_t}^2$ is the variance of the gradient noise ξ_t.

Proof: Define $W_t = \xi_t + k\zeta_t$ as the total noise added to the gradient $\nabla f(x_t)$ before clipping; Also let us denote $W_t \sim \bar{p}$. We know the following about W_t:

$$\mathbb{E}[W_t] = 0, \quad \bar{p}(W_t) = \int_{\xi_t} p(\xi_t) \frac{1}{k} \psi\left(\frac{W_t - \xi_t}{k}\right) d\xi_t, \text{ with } \psi \text{ being the pdf of } \mathcal{N}(0, I). \tag{4.55}$$

The proof idea is to bound the total variation distance between $\bar{p}(W_t)$ and $\frac{1}{k}\psi(\cdot)$ as $O(\frac{\sigma_{\xi_t}^2}{k^2})$, then use this distance to bound the clipping bias b_t. This implies $\bar{p}(W_t)$ will become more and more symmetric as k increases.

Towards this end, we have

$$\begin{aligned}
&\int_{W_t} \left| \bar{p}(W_t) - \frac{1}{k}\psi\left(\frac{W_t}{k}\right) \right| dW_t \\
=&\int_{W_t} \left| \int_{\xi_t} p(\xi_t) \frac{1}{k}\psi\left(\frac{W_t - \xi_t}{k}\right) d\xi_t - \frac{1}{k}\psi\left(\frac{W_t}{k}\right) \right| dW_t \\
=&k \int_{W_t'} \left| \int_{\xi_t} p(\xi_t) \frac{1}{k}\psi\left(W_t' - \frac{\xi_t}{k}\right) d\xi_t - \frac{1}{k}\psi(W_t') \right| dW_t'.
\end{aligned} \tag{4.56}$$

By Taylor's series, we have

$$\psi\left(W_t' - \frac{\xi_t}{k}\right) = \psi(W_t') + \left\langle \nabla\psi(W_t'), \frac{-\xi_t}{k} \right\rangle + \int_0^1 \left\langle \frac{\xi_t}{k}, \nabla^2\psi\left(W_t' - \tau\frac{\xi_t}{k}\right) \frac{\xi_t}{k} \right\rangle (1-\tau) d\tau. \tag{4.57}$$

Plugging the above into (4.56), we obtain:

$$
\begin{aligned}
&\int \left|\bar{p}(W_t) - \frac{1}{k}\psi(\frac{W_t}{k})\right| dW_t \\
=&\int_{W_t'} \left|\int_{\xi_t} p(\xi_t)\psi\left(W_t' - \frac{\xi_t}{k}\right) d\xi_t - \psi(W_t')\right| dW_t' \\
=&\int_{W_t'} \left|\int_{\xi_t} p(\xi_t)\int_0^1 \langle\frac{\xi_t}{k}, \nabla^2\psi(W_t' - t\frac{\xi_t}{k})\frac{\xi_t}{k}\rangle(1-\tau)d\tau d\xi_t\right| dW_t' \\
\leq&\int_0^1\int_{\xi_t}\int_{W_t'} \left|p(\xi_t)\langle\frac{\xi_t}{k}, \nabla^2\psi(W_t' - t\frac{\xi_t}{k})\frac{\xi_t}{k}\rangle(1-\tau)\right| dW_t' d\xi_t d\tau, \qquad (4.58)
\end{aligned}
$$

where the second equality is obtained by applying (4.57) and using the fact that ξ_t is zero mean.

Noticing that $\tau \leq 1$ and define $\hat{W}_t = W_t' - \tau\frac{\xi_t}{k}$, we have

$$
\begin{aligned}
&\int_{W_t'} \left|p(\xi_t)\left\langle\frac{\xi_t}{k}, \nabla^2\psi(W_t' - \tau\frac{\xi_t}{k})\frac{\xi_t}{k}\right\rangle(1-\tau)\right| dW_t' \\
=&p(\xi_t)(1-\tau)\int_{\hat{W}_t}\left|\left\langle\frac{\xi_t}{k}, \nabla^2\psi(\hat{W}_t')\frac{\xi_t}{k}\right\rangle\right| \frac{dW_t'}{d\hat{W}_t} d\hat{W}_t \\
=&p(\xi_t)(1-\tau)\int_{\hat{W}_t}\left|\langle\frac{\xi_t}{k}, \nabla^2\psi(\hat{W}_t)\frac{\xi_t}{k}\rangle\right| d\hat{W}_t, \\
=&p(\xi_t)(1-\tau)\int_{R\hat{W}_t}\left|\langle\frac{R\xi_t}{k}, \nabla^2\psi(R\hat{W}_t)\frac{R\xi_t}{k}\rangle\right| dR\hat{W}_t \\
=&p(\xi_t)(1-\tau)\int_{\hat{W}_t}\left|\langle\frac{R\xi_t}{k}, \nabla^2\psi(\hat{W}_t)\frac{R\xi_t}{k}\rangle\right| d\hat{W}_t, \qquad (4.59)
\end{aligned}
$$

where R is an arbitrary rotation matrix which means the integration term only depends on $\|\frac{\xi_t}{k}\|$. Thus we can assume $\xi_{t,1} = \|\xi_t\|$ and $\xi_{t,i} = 0$ for $i \geq 2$, wlog. Then, we have

$$
\begin{aligned}
&\int_{W_t'} \left|p(\xi_t)\langle\frac{\xi_t}{k}, \nabla^2\psi(W_t' - \tau\frac{\xi_t}{k})\frac{\xi_t}{k}\rangle(1-\tau)\right| dW_t' \\
\leq&p(\xi_t)(1-\tau)\int_{W_t'}\frac{\|\xi_t\|^2}{k^2}\left|\nabla^2_{1,1}\psi(W_t' - \tau\frac{\xi_t}{k})\right| dW_t' \\
\leq&p(\xi_t)(1-\tau)\int_{\hat{W}_t}\frac{\|\xi_t\|^2}{k^2}\left|\nabla^2_{1,1}\psi(\hat{W}_t)\right| |Det(\frac{dW_t'}{d\hat{W}_t})| d\hat{W}_t \\
\leq&p(\xi_t)(1-\tau)\frac{\|\xi_t\|^2}{k^2}q, \qquad (4.60)
\end{aligned}
$$

where we have define $\hat{W}_t = W'_t - \tau\frac{\xi_t}{k}$ and $q = \int_{-\infty}^{\infty}|h''(x)|dx$ with $h(x)$ being the pdf of 1-dimensional standard normal distribution. Thus, q is a dimension independent constant.

Substituting (4.60) into (4.58), we get

$$\begin{aligned}
&\int\left|\bar{p}(W_t) - \frac{1}{k}\psi\left(\frac{W_t}{k}\right)\right| d_{W_t}\\
\leq&\int_0^1\int_{\xi_t}\int_{W'_t}\left|p(\xi_t)\left\langle\frac{\xi_t}{k}, \nabla^2\psi(W'_t - \tau\frac{\xi_t}{k})\frac{\xi_t}{k}\right\rangle(1-\tau)\right| dW'_t d\xi_t d\tau\\
\leq&\int_0^1\int_{\xi_t} p(\xi_t)(1-\tau)\frac{\|\xi_t\|^2}{k^2}q d\xi_t d\tau\\
=&\int_0^1(1-\tau)\frac{\sigma^2_{\xi_t}}{k^2}q d\tau\\
=&\frac{1}{2}\frac{\sigma^2_{\xi_t}}{k^2}q,
\end{aligned} \tag{4.61}$$

where we used the fact that $\mathbb{E}[\xi_t] = 0$ and defined $\sigma^2_{\xi_t}$ being the variance of ξ_t.

By (4.5), we know that the following holds:

$$\begin{aligned}
\mathbb{E}_{\xi_t\sim p,\zeta_t}[\langle\nabla f(x_t), g_t\rangle] =& \mathbb{E}_{W_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle]\\
&+\underbrace{\int\langle\nabla f(x_t), \mathrm{clip}(\nabla f(x_t) + W_t, c)\rangle(p_t(W_t) - \tilde{p}_t(W_t))dW_t}_{b_t}.
\end{aligned} \tag{4.62}$$

Let $\tilde{p}$ be the pdf of $k\zeta_t$, from Theorem 5, we have

$$\mathbb{E}_{W_t\sim\tilde{p}}[\langle\nabla f(x_t), g_t\rangle] \geq \|\nabla f(x_t)\|\min\left\{\frac{3}{4}c, \|\nabla f(x_t)\|\right\}\mathbb{P}(\|k\zeta_t\| \leq \frac{c}{4}). \tag{4.63}$$

In addition, we can bound b_t as

$$|b_t| \leq \|\nabla f(x_t)\|\cdot c\cdot\int|p_t(\xi_t) - \tilde{p}_t(\xi_t)|d\xi_t \overset{(4.61)}{\leq} \|\nabla f(x_t)\|\cdot\frac{c}{2}\cdot\frac{\sigma^2_{\xi_t}}{k^2}q = \|\nabla f(x_t)\|O\left(\frac{\sigma^2_{\xi_t}}{k^2}\right). \tag{4.64}$$

Combining (4.62), (4.64), and (4.63) finishes the proof. **Q.E.D.**

4.7.6 Additional results and discussions on the probability term gradient correction in Section 4.5

Table 4.1: Scalability of $E_{\xi_t=0,\zeta_t}[\langle \nabla f(x_t), g_t\rangle]$ w.r.t. d and k

	$d=1$	$d=10$	$d=100$	$d=1,000$	$d=10,000$
$k=1$	10	9.572	7.077	3.015	0.995
$k=10$	6.788	2.961	0.992	0.316	0.1
$k=100$	0.758	0.316	0.098	0.032	0.01
$k=1000$	0.084	0.019	0.011	0.003	0.001

Theorem 9 says that after adding the Gaussian noise $k\zeta_t$ before clipping, the clipping bias can decrease. Meanwhile, the expected descent also decreases because $\mathbb{P}(\|k\zeta_t\| < \frac{c}{4})$ decreases with k. To get a more clear understanding of the theorem, consider $d = 1$, then $\mathbb{P}(\|k\zeta_t\| < \frac{c}{4}) = \mathrm{erf}(\frac{c}{4k})$ which decreases with an order of $O(\frac{1}{k})$. This rate is slower than the $O(\frac{1}{k^2})$ diminishing rate of the clipping bias. Thus, as k becomes large, the clipping bias will be negligible compared with the expected descent. This will translate to a **slower** convergence rate with a **better** final gradient bound in convergence analysis. The key idea of adding $k\zeta_t$ before clipping is to "symmetrify" the overall gradient noise distribution. By adding the isotropic symmetric noise $k\zeta_t$, the distribution of the resulting gradient noise $W_t \triangleq \xi_t + k\zeta_t$ will become increasingly more symmetric as k increases. In particular, the total variation distance between the distribution of W_t and $k\zeta_t$ decreases at a rate of $O(\frac{1}{k^2})$ which can be further used to bound the clipping bias. Then, one can apply Theorem 5 to lower bound $E_{\xi_t=0,\zeta_t}[\langle \nabla f(x_t), g_t\rangle]$ by letting $\tilde{p}$ be the distribution of $k\zeta_t$. We believe the lower bounds in Theorem 9 can be further improved when $d > 1$, notice that $\mathbb{P}(\|k\zeta_t\| < \frac{c}{4})$ tends to decrease fast with k when d being large.

However, we observe $\mathbb{E}_{\xi_t\sim p,\zeta_t}[\langle \nabla f(x_t), g_t\rangle]$ decreases with a rate of $O(1/d)$ and $O(1/k)$ in practice for fixed $\|\nabla f(x_t)\|$ and $\xi_t = 0$ (see Table 4.1 for $\|\nabla f(x_t)\| = 10$, the expectation $\mathbb{E}_{\xi_t=0,\zeta_t}[\langle \nabla f(x_t), g_t\rangle]$ is evaluated over 10^5 samples of $\zeta_t \sim \mathcal{N}(0, I)$). In addition, one can prove that the lower bounds in Theorem 5 are tight up to a constant when $d = 1$ or $\tilde{p}(\xi_t)$ is a distribution on a one dimensional subspace. This implies the lower bound can only be improved by using more properties of isotropic distributions like $\mathcal{N}(0, I)$ or resorting to a more general form of the lower bounds. We found this to be non-trivial and decide to leave it for future research.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016a. URL `http://arxiv.org/abs/1603.04467`.

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016b.

Charles Audet and John E Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on optimization*, 17(1):188–217, 2006.

Charles Audet and Warren Hare. *Derivative-free and blackbox optimization*. Springer, 2017.

Krishnakumar Balasubramanian and Saeed Ghadimi. Zeroth-order (non)-convex stochastic optimization via conditional gradient and gradient updates. In *Advances in Neural Information Processing Systems*, pages 3455–3464, 2018.

L. Balles and P. Hennig. Dissecting adam: The sign, magnitude and variance of stochastic gradients. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 404–413, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018.

Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. pages 464–473, 2014. doi: 10.1109/FOCS.2014.56. URL `https://doi.org/10.1109/FOCS.2014.56`.

A. Basu, S. De, A. Mukherjee, and E. Ullah. Convergence guarantees for rmsprop and adam in non-convex optimization and their comparison to nesterov acceleration on autoencoders. *arXiv preprint arXiv:1807.06766*, 2018.

Brett K. Beaulieu-Jones, Zhiwei Steven Wu, Chris Williams, Ran Lee, Sanjeev P. Bhavnani, James Brian Byrd, and Casey S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes*, 12(7):e005122, 2019. doi: 10.1161/CIRCOUTCOMES.118.005122. URL `https://www.ahajournals.org/doi/abs/10.1161/CIRCOUTCOMES.118.005122`.

S. Becker, Y. Le Cun, et al. Improving the convergence of back-propagation learning with second order methods. In *Proceedings of the 1988 connectionist models summer school*, pages 29–37. San Matteo, CA: Morgan Kaufmann, 1988.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. Signsgd: Compressed optimisation for non-convex problems. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 559–568, 2018.

L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J Su. Deep learning with gaussian differential privacy. *arXiv preprint arXiv:1911.11607*, 2019.

N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, pages 39–57, 2017.

C. Cartis, N. I. Gould, and P. L. Toint. On the complexity of steepest descent, newton's and regularized newton's methods for nonconvex unconstrained optimization problems. *Siam journal on optimization*, 20(6):2833–2852, 2010.

J. Chen and Q. Gu. Closing the generalization gap of adaptive gradient methods in training deep neural networks. *arXiv preprint arXiv:1806.06763*, 2018.

J. Chen, J. Yi, and Q. Gu. A Frank-Wolfe framework for efficient and effective adversarial attacks. *arXiv preprint arXiv:1811.10828*, 2018.

P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017.

Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. In *International Conference on Learning Representations*, 2019a.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in Neural Information Processing Systems*, 32, 2019b.

Xiangyi Chen, Steven Z Wu, and Mingyi Hong. Understanding gradient clipping in private sgd: A geometric perspective. *Advances in Neural Information Processing Systems*, 33:13773–13782, 2020.

M. Cheng, T. Le, P.-Y. Chen, J. Yi, H. Zhang, and C.-J. Hsieh. Query-efficient hard-label black-box attack: An optimization-based approach. *arXiv preprint arXiv:1807.04457*, 2018.

A. R. Conn, K. Scheinberg, and L. Vicente. Global convergence of general derivative-free trust-region algorithms to first-and second-order critical points. *SIAM Journal on Optimization*, 20(1):387–415, 2009a.

A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to derivative-free optimization*, volume 8. Siam, 2009b.

S. Dasgupta and A. Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Struct. Algorithms*, 22(1):60–65, January 2003. ISSN 1042-9832. doi: 10.1002/rsa.10073. URL `http://dx.doi.org/10.1002/rsa.10073`.

Y. Dauphin, H. de Vries, and Y. Bengio. Equilibrated adaptive learning rates for non-convex optimization. In *Advances in neural information processing systems*, pages 1504–1512, 2015.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

T. Dozat. Incorporating nesterov momentum into adam. 2016.

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

J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems*, pages 2962–2970, 2015.

X. Gao, B. Jiang, and S. Zhang. On the information-adaptive variants of the ADMM: an iteration complexity perspective. *Optimization Online*, 12, 2014.

E. Ghadimi, H. R. Feyzmahdavian, and M. Johansson. Global convergence of the heavy-ball method for convex optimization. In *2015 European Control Conference (ECC)*, pages 310–315. IEEE, 2015.

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

S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016.

S. Ghadimi, G. Lan, and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1-2): 267–305, 2016.

B. Gu, Z. Huo, and H. Huang. Zeroth-order asynchronous doubly stochastic algorithm with variance reduction. *arXiv preprint arXiv:1612.01425*, 2016.

Guy Gur-Ari, Daniel A. Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *CoRR*, abs/1812.04754, 2018. URL `http://arxiv.org/abs/1812.04754`.

Magnus R Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(6):409, 1952.

H. Huang, C. Wang, and B. Dong. Nostalgic adam: Weighing more of the past gradients when designing the adaptive learning rate. *arXiv preprint arXiv:1805.07557*, 2018.

A. Ilyas, K. Engstrom, A. Athalye, and J. Lin. Black-box adversarial attacks with limited queries and information. In *Proceedings of the 35th International Conference on Machine Learning*, July 2018a. URL `https://arxiv.org/abs/1804.08598`.

A. Ilyas, L. Engstrom, A. Athalye, and J. Lin. Black-box adversarial attacks with limited queries and information. *arXiv preprint arXiv:1804.08598*, 2018b.

A. Ilyas, L. Engstrom, and A. Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *arXiv preprint arXiv:1807.07978*, 2018c.

C. Jin, P. Netrapalli, and M. I. Jordan. Accelerated gradient descent escapes saddle points faster than gradient descent. *arXiv preprint arXiv:1711.10456*, 2017.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

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

L. Kotthoff, C. Thornton, H. H. Hoos, F. Hutter, and K. Leyton-Brown. Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka. *J. Mach. Learn. Res.*, 18(1):826–830, January 2017. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=3122009.3122034`.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

Sébastien Le Digabel. Algorithm 909: Nomad: Nonlinear optimization with the mads algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 37(4):44, 2011.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

L. Lei, C. Ju, J. Chen, and M. I. Jordan. Non-convex finite-sum optimization via scsg methods. In *Advances in Neural Information Processing Systems*, pages 2345–2355, 2017.

X. Li and F. Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. *arXiv preprint arXiv:1805.08114*, 2018.

Xinyan Li, Qilong Gu, Yingxue Zhou, Tiancong Chen, and Arindam Banerjee. Hessian based analysis of SGD for deep nets: Dynamics and generalization. In Carlotta Demeniconi and Nitesh V. Chawla, editors, *Proceedings of the 2020 SIAM International Conference on Data Mining, SDM 2020, Cincinnati, Ohio, USA, May 7-9, 2020 [the conference was canceled because of the coronavirus pandemic, the reviewed papers are*

published in this volume], pages 190–198. SIAM, 2020. doi: 10.1137/1.9781611976236.22. URL `https://doi.org/10.1137/1.9781611976236.22`.

X. Lian, H. Zhang, C.-J. Hsieh, Y. Huang, and J. Liu. A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order. In *Advances in Neural Information Processing Systems*, pages 3054–3062, 2016.

L. Liu, M. Cheng, C.-J. Hsieh, and D. Tao. Stochastic zeroth-order optimization via variance reduction method. *arXiv preprint arXiv:1805.11811*, 2018a.

S. Liu, J. Chen, P.-Y. Chen, and A. O. Hero. Zeroth-order online ADMM: Convergence analysis and applications. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, pages 288–297, April 2018b.

S. Liu, P.-Y. Chen, X. Chen, and M. Hong. signSGD via zeroth-order oracle. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=BJe-DsC5Fm`.

Sijia Liu, Bhavya Kailkhura, Pin-Yu Chen, Paishun Ting, Shiyu Chang, and Lisa Amini. Zeroth-order stochastic variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems*, 31, 2018c.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

A. Nemirovskii, D. B. Yudin, and E. R. Dawson. Problem complexity and method efficiency in optimization. 1983.

Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/kˆ 2). In *Doklady AN USSR*, volume 269, pages 543–547, 1983.

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

Y. Nesterov and V. Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 2(17):527–566, 2015.

Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

P. Ochs, T. Brox, and T. Pock. ipiasco: Inertial proximal algorithm for strongly convex optimization. *Journal of Mathematical Imaging and Vision*, 53(2):171–181, 2015.

T. T. Phuong and L. T. Phong. On the convergence proof of amsgrad and a new version. *arXiv preprint arXiv:1904.03590*, 2019.

Venkatadheeraj Pichapati, Ananda Theertha Suresh, Felix X Yu, Sashank J Reddi, and Sanjiv Kumar. Adaclip: Adaptive clipping for private sgd. *arXiv preprint arXiv:1908.07643*, 2019.

P. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Michael JD Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in optimization and numerical analysis*, pages 51–67. Springer, 1994.

Michael JD Powell. The bobyqa algorithm for bound constrained optimization without derivatives. *Cambridge NA Report NA2009/06, University of Cambridge, Cambridge*, pages 26–46, 2009.

S. J. Reddi, A. Hefny, S. Sra, B. Poczos, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323, 2016a.

S. J. Reddi, S. Sra, B. Poczos, and A. J. Smola. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. In *Advances in Neural Information Processing Systems*, pages 1145–1153, 2016b.

S. J. Reddi, S. Kale, and S. Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

L. M. Rios and N. V. Sahinidis. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3): 1247–1293, 2013.

A. K. Sahu, M. Zaheer, and S. Kar. Towards gradient free and projection free stochastic optimization. *arXiv preprint arXiv:1810.03233*, 2018.

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2016.

Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013.

F. Suya, Y. Tian, D. Evans, and P. Papotti. Query-limited black-box attacks to classifiers. *arXiv preprint arXiv:1712.08713*, 2017.

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.

Om Thakkar, Galen Andrew, and H. Brendan McMahan. Differentially private learning with adaptive clipping. *CoRR*, abs/1905.03871, 2019. URL `http://arxiv.org/abs/1905.03871`.

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

C.-C. Tu, P. Ting, P.-Y. Chen, S. Liu, H. Zhang, J. Yi, C.-J. Hsieh, and S.-M. Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. *arXiv preprint arXiv:1805.11770*, 2018.

A Ismael F Vaz and Luís N Vicente. Pswarm: a hybrid solver for linearly constrained global derivative-free optimization. *Optimization Methods & Software*, 24(4-5):669–685, 2009.

Di Wang and Jinhui Xu. Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications*

of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 1182–1189. AAAI Press, 2019. doi: 10.1609/aaai.v33i01.33011182. URL `https://doi.org/10.1609/aaai.v33i01.33011182`.

Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1226–1235, 2019.

R. Ward, X. Wu, and L. Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes, from any initialization. *arXiv preprint arXiv:1806.01811*, 2018.

D. Whitley. A genetic algorithm tutorial. *Statistics and computing*, 4(2):65–85, 1994.

Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=BkgzniCqY7`.

T. Yang, Q. Lin, and Z. Li. Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization. *arXiv preprint arXiv:1604.03257*, 2016.

M. D. Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations*, 2020.

D. Zhou, Y. Tang, Z. Yang, Y. Cao, and Q. Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.

Yuqing Zhu and Yu-Xiang Wang. Poission subsampled rényi differential privacy. In *International Conference on Machine Learning*, pages 7634–7642, 2019.

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.

F. Zou and L. Shen. On the convergence of adagrad with momentum for training deep neural networks. *arXiv preprint arXiv:1808.03408*, 2018.