Distributed Adaptive Optimization with Gradient Sparsification

Anonymous Authors

ABSTRACT

A clear and well-documented LaTeX document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the "acmart" document class, this article presents and explains many of the common variations, as well as many of the formatting elements an author may use in the preparation of the documentation of their work.

ACM Reference Format:
Anonymous Authors. 2018. Distributed Adaptive Optimization with Gradient Sparsification. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, JAugust 22–27, 2020, San Diego, CA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Nowadays, more and more machine learning problems involve massive datasets as well as huge models. For instance, large neural networks (possibly with millions of parameters) often play an importance role in recent progresses on computer vision [9, 12, 16], natural language processing [10] and reinforcement learning [11, 24] applications, with possibly millions of training samples. In many cases, using sequential stochastic gradient descent (SGD) which computes noisy gradient estimates on sample mini-batches still cannot solve the scalability issue. Consequently, distributed computational architectures are highly welcomed and heavily used in practice to make the learning process scalable. For deep learning networks, one of the standard frameworks based on stochastic gradients consists of M local workers and one central/master server. Massive data could be distributed on the local workers, and all the workers can perform computing tasks (e.g. high performance computation on GPU's) simultaneously to substantially speed up the large-scale training process. In the distributed system, each worker recurrently receives a copy of up-to-date model parameter, takes mini-batches of the data, computes the stochastic gradient and sends it back to the master server for parameter update. Then the server broadcasts the updated parameter to workers which starts next iteration.

System architectures. Basically, there are two types of architecture differed in the timing of parameter updates upon receiving the gradients. In *synchronized* computing system, the central server aggregates all the gradients computed by local workers before making a parameter update. One of the drawbacks of this strategy is the increased communication time since we have to wait for the slowest worker to finish its job. This becomes the main bottleneck particularly with very large scale distributed systems with numerous local workers. Alternatively, *asynchronous*, or *non-synchronous* architecture is proposed to alleviate this computational overhead by allowing the master node to update the model parameter every time a mini-batch gradient is received. This accelerates the training process, but the learning performance may be deteriorated since the gradients used for an update might be computed in the past, with respect to previous parameter values. This is called the delayed gradients, which is common under non-synchronous setting because the time for gradient computation is different across the workers. Figure 1 shows an example of asynchronous cyclic distributed architecture with constant delays. There are many works considering the asynchronous distributed training with SGD, e.g. see [1, 19, 26, 30] for more references on related topics.

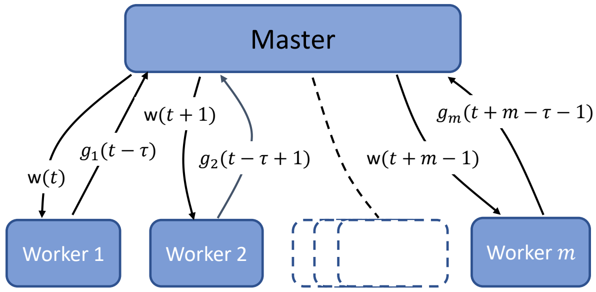

Figure 1: Illustration of cyclic distributed optimization system, with constant delay.

Gradient compression. Although distributed system runs very fast in principle, practical implementations often confront the problem of growing communication costs. In particular, since state-of-the-art deep neural networks usually contains millions or billions of parameters, frequent transmissions between the workers and the central server would typically be inefficient. To reduce the communication cost, several approaches are proposed. Gradient quantization methods save the communication overhead by transmitting the quantized low-precision gradients [3, 4, 13]. In this paper, we will mainly focus on another strategy called gradient sparsification [28, 29]. In either deterministic or probabilistic manner, we can transmit only a small portion of gradient coordinates to the master server. Empirical results [2] show that in some cases, using merely 1% coordinates can achieve similar performance as using the full gradients. Randomized sparsification schemes often produce unbiased estimate of the true stochastic gradient. It is also possible to use deterministic methods with biased stochastic gradient estimates to further simplify the appraoch. The idea of only

KDD '20, August 22–27, 2020, San Diego, CA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

Algorithm 1: AMSGrad [22]

```
Input: parameter $\beta_1$, $\beta_2$, and $\eta_t$
Initialize: $w_1 \in \mathcal{W}$, $\hat{v}_0 = v_0 = 0$, $m_0 = 0$
for $t = 1$ to $T$ do
    $g_t = \nabla f_t(w_t)$
    $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$
    $v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$
    $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$, $\hat{V}_t = diag(\hat{v}_t)$
    $w_{t+1} = \Pi_{\mathcal{W},\sqrt{\hat{V}_t}}(w_t - \eta_t \frac{m_t}{\sqrt{\hat{v}_t}})$ (element-wise division)
end for
```

transporting largest K gradient coordinates, which motivates the so-called Top$K$-SGD method, is developed and studied in [5, 20] and several following works discussing the convergence property, e.g. [23, 27].

Adaptive optimization. Given the efficiency of distributed computing structure, in many applications, deep neural networks are trained in a distributed manner. Yet, most of works aforementioned consider comparatively simple optimization methods such as SGD and proximal gradient methods. In recent years, adaptive optimization algorithms (e.g. AdaGrad [8], Adam [14], AMSGrad [6, 22] and etc.) become popular because of their superior performance on many important learning tasks. In many learning problems, adaptive methods have been shown to converge faster and generalize better. Unlike SGD, these methods use different implicit learning rates for different coordinates that keep changing adaptively throughout the training process, based on second order moment estimation. Algorithm 1 provides a constant hyper-parameter version of AMSGrad, while in practice decaying learning rate η_t is often adopted to boost the performance. Although adaptive methods are getting more and more popular, the corresponding distributed version has not been considered yet.

Our contributions. As adaptive methods have shown their advantages in training deep neural nets, scaling up the training process is in great demand. Two questions arise: 1) Can we implement adaptive algorithms in a distributed environment? 2) Can we combine compressed gradients to further speed up the training? The goal of this paper is to build connection among distributed computing, gradient compression and adaptive optimization. More precisely, the main contributions include:

- We seek to accelerate large scale learning and improve its performance by combining adaptive optimization method with distributed computing and sparsified gradient transmission. More specifically, we analyze the expected regret of AMSGrad algorithm with sparsified gradients in classical setting and also asynchronous distributed setting. To the best of our knowledge, this is the first work combining gradient compression and adaptive optimization method under distributed setting.
- Based on our theoretical analysis, we propose an adaptive sparsification scheme which aims at reducing the regret, at any target level of sparsity. We also provide and analyze an efficient algorithm to solve for the optimal coordinate selection strategy.
- We conduct experiments on various real world datasets with different network architectures to demonstrate the effectiveness of our proposed method.

Roadmap. In Section 2, we first introduce some preliminaries and backgrounds. In Section 3, we analyze the traditional sequential AMSGrad algorithm with gradient sparsification. Based on the results, in Section 4 we propose a sparification scheme specifically for the adaptive method. In Section 5, we consider the sparsified adaptive optimization in asynchronous distributed system. We show experiments on our proposed methods in Section 6, and conclude the paper in Section 7.

2 BACKGROUND AND NOTATIONS

Notations. Throughout the paper, the model parameters at time t is w_t, with parameter space $w_t \in \mathcal{W}, \forall t$. We use g_i to denote the i-th element of vector g, and $g_{t,i}$ the i-th element of g_t when there are many vectors indexed as $(g_1, g_2, ...)$. $\langle \cdot \rangle$ is the inner product, and $\| \cdot \|$ is the l_2 norm without further clarification. $\nabla f(w)$ denotes the gradient of function $f(w)$. In addition, $\oslash$ represents Hadamard (element-wise) division.

2.1 Gradient Sparsification for SGD

As mentioned above, we can divide gradient spasification strategies into two categories: deterministic and probabilistic. In general, deterministic methods are faster, but theoretical results are hard to establish. Thus, in this paper we will focus on probabillistic (or randomized) sparsification.

The paper [29] propose to use Bernoulli random variables to sample gradient coordinates in distributed SGD. Suppose a worker computes the mini-batch gradient g_t at time t. The sparsified gradient is defined in the form

$$\tilde{g}_i = \begin{cases} \frac{g_i}{p_i}, & \text{with probability } p_i, \\ 0, & \text{with probability } (1 - p_i), \end{cases} \tag{1}$$

This set of probability p_i, $i = 1, ..., d$, is solved via a constrained linear program which tries to reduce the bound on expected loss $\mathbb{E}[f(w_{t+1})]$ through minimizing $\mathbb{E}[\|\tilde{g}_t(w_t)\|^2]$. The solution gives optimal sparsity given a fixed variance budget on sparsified gradient $\tilde{g}_t$. To be specific, the closed-form solution is

$$p_i = \min(\lambda |g_i|, 1), \forall i \in [d], \tag{2}$$

where λ is some constant depending on g and the variance budget. As we can see, the probability of a coordinate being selected at time t is in general proportional to the magnitude of g_t. In [28], the authors propose the idea to sparsify the atomic decomposition (e.g. the SVD) of the gradient, by sending only several atoms. This is different from the classical "sparsity" concept in terms of the non-zeros elements. In this paper, we also consider the randomized sparsification scheme in the form of (1). However, as we will see later, the approach becomes more sophisticated and adaptive to the learning trajectory.

2.2 Stochastic convex optimization

In this present work, we consider optimization program under standard stochastic optimization setting [1, 6, 26], among others. In this context, our goal is to learn the model w which minimizes

$$\min_{w \in \mathcal{W}} f(w) = \min_{w \in \mathcal{W}} \mathbb{E}[F(w, x)],$$

with F, f convex and $x \sim P$ follows some data distribution. The regret is defined as

$$Regret = \sum_{t=1}^{T} f(w_t) - f(w^*),$$

with $w^* = \underset{w \in \mathcal{W}}{\operatorname{argmin}} f(w)$. Since we exploits stochastic gradients, expected regret should be analyzed herein. Also, following [1], we consider convergence by a slightly modified term,

$$\mathbb{E}[f(\hat{w}_T) - f(w^*)],$$

where $\hat{w}^T = \frac{1}{T}\sum_{t=1}^{T} w_t$, $w^* = \underset{w \in \mathcal{W}}{\operatorname{argmin}} f(w)$, and the expectation is taken with respect to all sources of randomness from the stochastic gradients, the sparsification process and possibly the delays. In [21, 31], the authors analyze delayed SGD in non-synchronous distributed systems. If all the delays are upper bounded by B, with bounded gradients the convergence rate of delayed SGD is $O(\frac{\sqrt{B}}{\sqrt{T}})$ when learning rate η_t follows $O(\frac{1}{\sqrt{t}})$. [1, 26] study similar problem for proximal and mirror descent. Both results find that the delays are asymptotically small under some smoothness conditions. Moreover, [26] studies extensions of SGD with carefully designed adaptive learning rate, which benefits the asynchronous distributed optimization.

3 ADAPTIVE OPTIMIZATION WITH SPARSIFIED GRADIENTS

In this section, we first consider the effect of using sparsified gradients in the generic single-machine sequential AMSGrad updates. Denote $g_t = \nabla f(w_t) \in \mathbb{R}^d$. The sparsified algorithm simply update line 3 and 4 in Algorithm 1 by $\tilde{g}_t$ instead of g_t, where $\tilde{g}_t$ is assumed to be like (1). Denote the probability of selecting i-th gradient coordinate at time t as $p_{t,i}$, and p_t as the corresponding probabilty vector. For now, we only assume general $0 \le p_{t,i} \le 1$. The detailed analysis will be provided in the next section.

- **A1.** (Convexity) f is a convex function such that $f(w) - f(z) \le \langle \nabla f(w), w - z \rangle$ holds for $\forall w, z \in \mathcal{W}$.
- **A2.** (Bounded diameter) Parameter space $\mathcal{W}$ has bounded diameter, $\|x - y\|_2 \le D$ for $\forall x, y \in \mathcal{W}$.
- **A3.** (Bounded gradient) All the gradients are bounded, such that $\|\nabla f_t(w)\|_\infty \le G_\infty$ for $\forall t \in [T]$, $w \in \mathcal{W}$.
- **A4.** (Bounded variance) There exists σ such that the stochastic gradients are unbiased, $\mathbb{E}[g_t] = \nabla f(w_t)$, and bounded variance, i.e.

$$\mathbb{E}[\|g_t - \nabla f(w_t)\|^2] \le \sigma^2, \forall t \in [T].$$

Assumptions **A1-A3** are commonly seen in adaptive optimization literature, e.g. [8, 14, 22]. **A4** is a standard assumption in studying stochastic gradient methods. Besides, since we consider randomized sparsification, here we also assume that the sparsified gradient $\tilde{g}_t$ is unbiased at $\forall t$.

The next theorem gives the convergence behavior when using sparsified stochastic gradients in AMSGrad updates. Following the setting of [22], we assume decaying learning rate $\eta_t = \eta/\sqrt{t}$ and a moderate decay of $\beta_{1,t} = \beta_1/t$.

Theorem 1. *Under Assumptions **A1-A3**, suppose decaying learning rate $\eta_t = \eta/\sqrt{t}$, and $\beta_{1,t} = \beta_1/t$. Let $\gamma = \beta_1/\sqrt{\beta_2} \le 1$. Consider AMSGrad updates (Algorithm 1) using sparsified stochastic gradients $\tilde{g}_t$ at each iteration. We have*

$$\mathbb{E}\left[\sum_{t=1}^{T}(f(w_t) - f(w^*))\right]$$

$$\le \frac{3dD^2 G_\infty \sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}} + \frac{\eta\sqrt{\log T + 1}}{(1-\beta_1)^2\sqrt{1-\beta_2}(1-\gamma)}\mathbb{E}\Big[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|\Big],$$

where $w^* = \underset{w \in \mathcal{W}}{\operatorname{argmin}} f(w)$, and $p_{min} = \min_{\substack{i \in [d], t \in [T] \\ p_{t,i} \neq 0}} p_{t,i}$.

Remark 1. *We provide an "intermediate" result here—Instead of continuing bounding the last term, we will more directly investigate this term to choose selection probability p_t in Section 4.*

Lemma 1. $\mathbb{E}[\|\nabla f(w_t) - \tilde{g}_t\|^2] \le \frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}}, \forall t \in [T]$.

Proof. By conditional on g_t, we have

$$\begin{aligned}
\mathbb{E}[\|\nabla f(w_t) - \tilde{g}_t\|^2] &= \mathbb{E}[\|\tilde{g}_t\|^2] - \|\nabla f(w_t)\|^2 \\
&= \mathbb{E}\big[\mathbb{E}[\|\tilde{g}_t\|^2 | g_t]\big] - \|\nabla f(w_t)\|^2 \\
&\le \mathbb{E}[\frac{\|g_t\|^2}{p_{min}}] - \|\nabla f(w_t)\|^2 \\
&\le \frac{1}{p_{min}}(\|\nabla f(w_t)\|^2 + \sigma^2) - \|\nabla f(w_t)\|^2 \\
&\le \frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}}.
\end{aligned}$$

To get the bound, we use law of total expectation for the second line, and Assumption **A3** for the last line. □

Lemma 2. $\mathbb{E}[\sqrt{\hat{v}_{i,t}}] \le G_\infty/\sqrt{p_{min}}$ *for all $i \in [d]$ and $t \in [T]$.*

Proof. Note that $\hat{v}_t = \max\{v_1, v_2, ..., v_t\}$. Hence, for the i-th coordinate, $\exists s \in [t]$ such that

$$\begin{aligned}
\mathbb{E}[\sqrt{\hat{v}_{t,i}}] = \mathbb{E}[v_{s,i}] &= \sqrt{1-\beta_2}\,\mathbb{E}\left[\sqrt{\sum_{j=1}^{s}\beta_2^{s-j}\tilde{g}_{j,i}^2}\right] \\
&\le \sqrt{1-\beta_2}\sqrt{\mathbb{E}[\sum_{j=1}^{s}\beta_2^{s-j}\tilde{g}_{j,i}^2]} \\
&= \sqrt{1-\beta_2}\sqrt{\sum_{j=1}^{s}\beta_2^{s-j}g_{j,i}^2/p_{j,i}} \\
&\le \sqrt{1-\beta_2}\frac{1}{\sqrt{1-\beta_2}}\sqrt{G_\infty^2/\min_j p_{j,i}} \\
&\le G_\infty/\sqrt{p_{min}},
\end{aligned}$$

where the first inequality is due to Jensen's inequality, and the second inequality is by Assumption **A3**. □

Lemma 3. *(McMahan and Streeter, 2010) For any $Q \in S_s^+$ and convex feasible set $\mathcal{W} \subset \mathbb{R}^d$, suppose $w_1 = \min_{w\in\mathcal{W}} \|Q^{1/2}(w - z_1)\|$, $w_2 = \min_{w\in\mathcal{W}} \|Q^{1/2}(w - z_2)\|$, then the following holds:*

$$\|Q^{1/2}(w_1 - w_2)\| \le \|Q^{1/2}(z_1 - z_2)\|.$$

Lemma 4. *(Reddi et.al, 2018) Under the setting of Theorem 1, we have*

$$\sum_{t=1}^{T} \eta_t \|\hat{V}_t^{-1/4} m_t\|^2 \le \frac{\eta\sqrt{\log T + 1}}{(1-\beta_1)(1-\gamma)\sqrt{(1-\beta_2)}} \sum_{i=1}^{d} \|\tilde{g}_{1:T,i}\|.$$

Lemma 5. $\sum_{n=1}^{N} \frac{1}{\sqrt{n}} \le 2\sqrt{N}$.

Proof. (of Theorem 1) The proof follows from [22]. The update rule gives

$$w_{t+1} = \Pi_{\mathcal{W},\sqrt{\hat{V}_t}}(w_t - \eta_t \hat{V}_t^{-1/2} m_t) = \min_{w\in\mathcal{W}} \|\hat{V}_t^{1/4}(w - (w_t - \eta_t \hat{V}_t^{-1/2} m_t))\|.$$

By Lemma 3 on w_{t+1} and w^*, we have

$$\begin{aligned} \|\hat{V}_t^{1/4}(w_{t+1} - w^*)\|^2 &\le \|\hat{V}_t^{1/4}(w_t - (\eta_t \hat{V}_t^{-1/2} m_t - w^*))\| \\ &= \|\hat{V}_t^{1/4}(w_t - w^*)\|^2 + \eta_t^2 \|\hat{V}_t^{-1/4} m_t\|^2 \\ &\quad + 2\eta_t \langle \beta_{1,t} m_{t-1} + (1-\beta_{1,t})\tilde{g}_t, w_t - w^* \rangle. \end{aligned}$$

Rearranging terms, we have

$$\begin{aligned} &\langle \tilde{g}_t, w_t - w^* \rangle \\ \le &\frac{1}{2\eta_t(1-\beta_{1,t})}\left[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1} - w^*)\|^2\right] \\ &+ \frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2 + \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t \|\hat{V}_t^{-1/4} m_{t-1}\|^2 \\ &+ \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2. \end{aligned}$$

On the other hand, taking expectation and by the convexity of function f, we obtain

$$\begin{aligned} &\mathbb{E}\Big[\sum_{t=1}^{T} f(w_t) - f(w^*)\Big] \qquad (3) \\ \le &\underbrace{\mathbb{E}\Big[\sum_{t=1}^{T}\langle \tilde{g}_t, w_t - w^*\rangle\Big]}_{A} + \underbrace{\mathbb{E}\Big[\sum_{t=1}^{T}\langle \nabla f(w_t) - \tilde{g}_t, w_t - w^*\rangle\Big]}_{B}. \end{aligned}$$

Term B is zero since $\tilde{g}_t$ is unbiased of $\nabla f(w_t)$, and w_t is independent of the stochastic gradient $\tilde{g}_t$ given the sigma-field $\mathcal{G}_{t-1} = \{\tilde{g}_1, ..., \tilde{g}_{t-1}\}$. For term A, we have

$$\begin{aligned} A \le\ &\mathbb{E}\Big[\frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2\Big] \\ &+ \mathbb{E}\Bigg[\sum_{t=1}^{T}\Big\{\frac{1}{2\eta_t(1-\beta_{1,t})}\left[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1} - w^*)\|^2\right] \\ &+ \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t\|\hat{V}_t^{-1/4} m_{t-1}\|^2 + \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg]. \end{aligned}$$

Using Lemma 4, for the first term we have

$$\begin{aligned} &\mathbb{E}\Big[\frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2\Big] \\ \le &\frac{\eta\sqrt{\log T + 1}}{(1-\beta_1)^2\sqrt{1-\beta_2}(1-\gamma)}\mathbb{E}\Big[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|\Big]. \end{aligned}$$

The remaining terms can be bounded by

$$\begin{aligned} \le &\mathbb{E}\Bigg[\sum_{t=1}^{T}\Big\{\frac{1}{2\eta_t(1-\beta_{1,t})}\left[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1} - w^*)\|^2\right] \\ &+ \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t\|\hat{V}_t^{-1/4} m_{t-1}\|^2 + \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg] \\ \le &\mathbb{E}\Bigg[\frac{\|\hat{V}_t^{1/4}(w_1 - w^*)\|^2}{2\eta(1-\beta_1)} + \frac{1}{2(1-\beta_1)}\sum_{t=2}^{T}\Big\{\frac{\|\hat{V}_t^{1/4}(w_t - w^*)\|^2}{\eta_t} \\ &- \frac{\|\hat{V}_{t-1}^{1/4}(w_t - w^*)\|^2}{\eta_{t-1}}\Big\} + \sum_{t=1}^{T}\Big\{\frac{\beta_{1,t}}{2\eta_t(1-\beta_1)}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg] \\ = &\mathbb{E}\Bigg[\underbrace{\sum_{i=1}^{d}\frac{\hat{v}_{1,i}^{1/2}(w_{1,i} - w_i^*)^2}{2\eta(1-\beta_1)} + \sum_{t=2}^{T}\sum_{i=1}^{d}\Big[\frac{\hat{v}_{t,i}^{1/2}}{\eta_t} - \frac{\hat{v}_{t-1,i}^{1/2}}{\eta_{t-1}}\Big]\frac{(w_{t,i} - w_i^*)^2}{2(1-\beta_1)}}_{I} \\ &+ \underbrace{\frac{1}{2(1-\beta_1)}\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{\beta_{1,t}\hat{v}_{t,i}^{1/2}(w_{t,i} - w_i^*)^2}{\eta_t}}_{II}\Bigg]. \end{aligned}$$

Regarding II, we have

$$\begin{aligned} \mathbb{E}[II] &= \frac{1}{2(1-\beta_1)}\mathbb{E}\Big[\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{\beta_1\hat{v}_{t,i}^{1/2}}{\eta\sqrt{t}}(w_{t,i} - w_i^*)^2\Big] \\ &\le \frac{D^2}{2\eta(1-\beta_1)}\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{1}{\sqrt{t}}\mathbb{E}[\hat{v}_{t,i}^{1/2}] \\ &\le \frac{dD^2 G_\infty\sqrt{T}}{\eta(1-\beta_1)\sqrt{p_{min}}}, \end{aligned}$$

where the first inequality is due to Assumption **A2** and the fact that $\beta_{1,t} = \beta_1/t \le 1$, and the last inequality is a consequence of Lemma 2 and Lemma 5. For I, we observe that it is a telescopic sum. After cancelling terms, we obtain

$$\begin{aligned} \mathbb{E}[I] &= \frac{1}{2\eta_T(1-\beta_1)}\sum_{i=1}^{d}\mathbb{E}[\hat{v}_{T,i}^{1/2}(w_{t,i} - w_i^*)^2] \\ &\le \frac{dD^2 G_\infty\sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}}, \end{aligned}$$

where the inequality is because $\eta_T = \eta/\sqrt{T}$, Assumption **A2** and Lemma 2. Combining parts together, the theorem is proved. □

As a smaller p_{min} typically implies higher compression sparsity, Theorem 1 indicates that high sparsity would increase the regret. When $p_{min} \equiv 1$, which means no sparsification is applied, Theorem 1 reduces to the classical online learning result [22] with full information feedback. The difference is that, in our result the last term becomes an expectation over the sparsified stochastic gradients, which would lead to extra variance term, while in the classical online learning analysis, the true gradients are assumed to be known.

Notice that p_{min} decreases with time T, and $\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|$ increases with T. This two quantities are primarily controlled by the sparsity level. Consequently, the convergence rate essentially depends on the sparsity of $\tilde{g}$. In [22], the authors claim that the regret of AMSGrad (in online learning setting) can reach $O(1/\sqrt{T})$ with proper choice of learning rate, and can be considerably smaller than that of SGD if $\sum_{i=1}^{d}\|g_{1:T,i}\| \ll \sqrt{dT}$. In our problem, if $p_{min} \gg 1/T$ and $\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|] \leq \sum_{i=1}^{d}\|g_{1:T,i}\|/\sqrt{p_{min,i}} \lesssim \sqrt{dT}$, we can also have roughly $O(1/\sqrt{T})$ convergence. However, the convergence may be slowed down when $\tilde{g}$ is very sparse. Thus, like other gradient sparsification methods for SGD [23], in practice one should find a good balance between learning performance and communication cost. Indeed, similar empirical observation was also reported for SGD with randomized sparsification (e.g. [29]), but theoretical results was not established therein.

4 STRATEGY FOR UPDATING p_t

In this section, we propose our gradient sparsification strategy. We will derive a new set of p_t at each iteration, with the effort to reduce the expected loss. Notice that in Theorem 1, the regret at time T contains the term $\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|]$. For gradient sparsification, our motivation is to choose p_t at each time t to minish this quantity, in order to lower down the expected regret bound. As we will see later, our method to determine p_t is also adaptive, in the sense that it relies on previous gradients (similar to m_t and v_t in AMSGrad). Our algorithm can achieve any desired sparsity level, denoted by $s = c/d$, where c is the expected number of selected coordinates. Note that $c \in \mathbb{R}$ is different from the exact number of selected coordinates, which must an integer.

4.1 Formulation

We start by deriving an upper bound on $\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|]$. At each time step t, we can compute

$$\begin{aligned}\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:t,i}\|] &= \mathbb{E}[\sum_{i=1}^{d}\sqrt{\tilde{g}^2_{1:t-1,i}+\tilde{g}^2_{t,i}}]\\ &= \sum_{i=1}^{d} p_{t,i}\sqrt{u_{t-1,i}+\frac{g^2_{t,i}}{p^2_{t,i}}}+(1-p_{t,i})\sqrt{u_{t-1,i}},\end{aligned}$$

where we define $u_{t,i} = \sum_{s=1}^{t}\tilde{g}^2_{1:s,i}$. Continuing the analysis, we have

$$\begin{aligned}\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:t,i}\|] &= \sum_{i=1}^{d} p_{t,i}\sqrt{u_{t-1,i}}\sqrt{(1+\frac{g^2_{t,i}}{p^2_{t,i}u_{t-1,i}})}+(1-p_{t,i})\sqrt{u_{t-1,i}}\\ &\leq \sum_{i=1}^{d} p_i\sqrt{u_{t-1,i}}(1+\frac{g^2_{t,i}}{2p^2_{t,i}u_{t-1,i}})+(1-p_{t,i})\sqrt{u_{t-1,i}}\\ &\leq \sum_{i=1}^{d}(\frac{g^2_{t,i}}{2p_{t,i}\sqrt{u_{t-1,i}}}+\sqrt{u_{t-1,i}}).\end{aligned}$$

Here the second line is a consequence of the inequality $\sqrt{1+x} \leq 1+\frac{x}{2}, \forall x \geq -1$. Note that the key idea is at time t, we can always treat past information u_{t-1} as given. Hence, a local worker computes p_t by minimizing

$$\begin{aligned}\min_{p}\sum_{i=1}^{d}\frac{g^2_{t,i}}{p_i\sqrt{u_{t-1,i}}},\ s.t. \sum_{i=1}^{d}p_i \leq c,\\ 0 < p_i \leq 1, \forall i \in [d].\end{aligned} \tag{4}$$

The solution is provided in the next theorem.

Theorem 2. *At each time t, the solution to problem (4) is the vector p_t such that*

$$p_{t,i} = \min\{\frac{|g_{t,i}|}{u^{1/4}_{t-1,i}\sqrt{\lambda}}, 1\},$$

where the constant λ depends on g, u and c.

Proof. We proceed with Lagrangian multipliers. Define

$$L(p_t,\lambda,\nu) = \sum_{i=1}^{d}\frac{g^2_{t,i}}{p_{t,i}\sqrt{u_{t-1,i}}}+\lambda(\sum_{i=1}^{d}p_{t,i}-c)+\sum_{i=1}^{d}\nu_i(p_{t,i}-1).$$

The KKT conditions read as $\forall i \in [d]$,

$$\begin{cases}-\dfrac{g^2_{t,i}}{p^2_{t,i}\sqrt{u_{t-1,i}}}+\lambda+\nu_i = 0,\\ \lambda \geq 0,\\ \nu_i \geq 0,\\ \nu_i(p_{t,i}-1) = 0,\\ \lambda(\sum_{i=1}^{d}p_{t,i}-c) = 0.\end{cases}$$

Solving the equations yields

$$p_{t,i} = \frac{|g_{t,i}|}{u^{1/4}_{t-1,i}\sqrt{\lambda+\nu_i}}.$$

By complementary slackness condition we know that if $p_{t,i} \neq 0$, then $\nu_i = 0$; and if $\nu_i \neq 0$, then $p_{t,i} = 1$. Thus, we further have

$$p_{t,i} = \begin{cases}1, & \nu_i \neq 0,\\ \frac{|g_{t,i}|}{u^{1/4}_{t-1,i}\sqrt{\lambda}}, & \nu_i = 0.\end{cases}$$

That is, for some coordinates we may have $p_{t,i} = 1$, and for others, the probability is proportional to the absolute scaled gradient $|g_{t,i}|/u^{1/4}_{t-1,i}$. It is easy to justify that those with probability equal to 1 should be the coordinates with largest magnitude in absolute scaled gradients. This proves the theorem. □

The solution differs from (2) for SGD in the u_{t-1} term which contains information of past gradients. Intuitively, the proposed scheme assigns more probability to coordinates with small $u_{t,i}$. As a result, we tend to explore new directions in the parameter space, by being more likely to select coordinates that have small magnitude, or have rarely been chosen in past iterations. In other words, our sparsification strategy is also adaptive to the learning trajectory.

Extensions. While the key idea of our proposal is to compute the selection probability depending on previous realizations of sparsified gradients, some ingredients of the algorithm could be set flexible to improve the empirical performance. For example, we can choose the power of u in the calculation, e.g. $\xi = \frac{1}{4}, \frac{1}{2}$ and etc. This parameter controls the impact of past gradients on the choice of $\tilde{g}_t$. Larger ξ implies higher dependency on the historical information.

4.2 Efficient implementation

In this subsection, we discuss how to solve for p_t, whose explicit formula is given by Theorem 2. First, let us sort the coordinates from high to low by $|g_{t,i}|/u_{t-1,i}^{1/4}$, and re-index from 1 to d. According to Theorem 3, we may assume that the first k dimensions have probability equal to 1. By the constraint $\sum_{i=1}^{d} p_{t,i} = c$, we have

$$k + \sum_{i>k}^{d} \frac{|g_{t,i}|}{u_{t-1,i}^{1/4}\sqrt{\lambda}} = c,$$

which implies that

$$\sqrt{\lambda} = \frac{1}{c-k} \sum_{i>k}^{d} \frac{|g_{t,i}|}{u_{t-1,i}^{1/4}}. \tag{5}$$

Since $p_{t,k+1} \leq 1$, we need to find the smallest k so that

$$p_{t,k+1} = \frac{|g_{t,k+1}|}{u_{t-1,k+1}^{1/4}\sqrt{\lambda}} \leq 1, \tag{6}$$

with λ given by (5). Using this procedure, we can get the exact closed-form solution of problem (4).

However, in practice, directly implementing the strategy described above requires partial sorting and search among the coordinates, which is possibly very expensive especially in high dimensional parameter space (e.g. large neural networks). To address this, we propose to use an iterative algorithm that only requires some basic operations to efficiently implement our scheme, which is summarized in Algorithm 2. Basically, first we compute the normalize probability, and set $p_{t,i} = \min\{p_{t,i}, 1\}$. Suppose we have k coordinates with $p_{t,i} \geq 1$. We keep all these k 1's and repeat the procedure on the remaining coordinates, with updated constraint parameter $c - k$, which is non-increasing during this process. The algorithm repeats this procedure and stops when all the normalized probabilities are no greater than 1.

Algorithm 2: Compute p_t

```
Input: $g_t \in \mathbb{R}^d$, $u_{t-1} \in \mathbb{R}^d$, $c \in \mathbb{R}$,power $\xi$
Initialize: Active set $\mathcal{A} = [d]$
while true do
    Compute $p_{t,\mathcal{A}} = c \cdot |g_{t,\mathcal{A}}| \oslash u^{\xi}_{t-1,\mathcal{A}} / \sum_{i\in\mathcal{A}} |g_{t,i}|/u^{\xi}_{t-1,i}$
    Find $\mathcal{I} = \{i \in \mathcal{A} : p_{t,i} >= 1\}$
    if $\mathcal{I} = \emptyset$ then
        break;
    end
    Set $p_{t,\mathcal{I}} = 1$, update $\mathcal{A} = \mathcal{A} \setminus \mathcal{I}$, $c = c - |\mathcal{I}|$
end
Output: The selection probability vector $p_t$
```

Proposition 1. *Algorithm 2 finds smallest k such that (6) holds.*

Proof. We start by looking at the first iteration of Algorithm 2. For simplicity we omit the subscript of time and denote $r_i = |g_i|/u_i^{1/4}$, $R = \sum_{j=1}^{d} r_j$ and $s_i = c \cdot \frac{r_i}{R}, i \in [d]$. Also W.L.O.G., we re-index the coordinates such that $s_1 \geq s_2 \cdots \geq s_k \geq 1 > s_{k+1} > \ldots$, though the ordering will not be explicitly implemented. Recall that by (6), our goal is to find the smallest k (after sorting from high to low) such that $\frac{(c-k)r_k}{R-\sum_{j=1}^{d} r_j} < 1$, which will be denoted as k^*. We consider the following three cases.

- $k = 0$. In this case, $s_i < 1, \forall i$. The algorithm immediately stops at first iteration and returns $p_i = s_i, i \in [d]$.
- $k = 1$. In this case $\frac{cr_1}{R} \geq 1$ and $\frac{cr_2}{R} < 1$. Notice that
$$(c-1)r_2 - (R - r_1) = (cr_2 - R) + (r_1 - r_2).$$
The first term is negative and the second term is positive, so we cannot affirm whether $\frac{(c-1)r_2}{R-r_1}$ is less than 1 or not. Therefore, we set $p_1 = 1$, exclude this coordinate from the active set $\mathcal{A}$ and proceed to next iteration with $c = c - 1$, $R = R - r_1$.
- $k > 1$. By similar reasoning, we know that given $\frac{cr_1}{R} \geq 1$ and $\frac{cr_k}{R} \geq 1$ for any $k > 1$, we have
$$(c-k)r_k - (R - \sum_{j=1}^{k} r_j) = (cr_k - R) + (\sum_{j=1}^{k} r_j - kr_k) \geq 0,$$
which implies
$$\frac{(c-k)r_k}{R - \sum_{j=1}^{d} r_j} \geq 1.$$
Therefore, we can safely set $p_i = 1, i \in [k]$, exclude $[k]$ from the active set, and run next iteration with $c = c - k$, $R = R - \sum_{j=1}^{k} r_j$.

Above analysis leads to the following claim: in each iteration, all coordinates with $s_i \geq 1$ are guaranteed to be smaller than k^*. Therefore, when the algorithm stops (all s_i's in active set $\mathcal{A}$ is less than 1), likewise in the first case, the coordinate with maximum r_i is assured to be k^*.

□

Proposition 1 suggests that Algorithm 2 solves for the closed-form solution. The benefit is that, as discussed in [29], this algorithm can be significantly accelerated on hardware supporting *single instruction multiple data* (SIMD), such as modern Intel CPUs with SSE/AVX instructions and etc.. This way, the sparsification procedure can be efficient in practice.

5 SPARSIFIED NON-SYNCHRONOUS ADAPTIVE OPTIMIZATION

In distributed computational architecture, people usually allow asynchronous updates collected from different workers to speed up the training. Since local workers have different computing power, the actual gradients used for updates may experience inherent delays. In this section, we investigate the convergence property of using sparsified AMSGrad in non-synchronous distributed setting, with delay gradients. We will consider the cyclic protocol which is described in Figure 1. More precisely, the parameter $w \in \mathcal{W}$ is maintained in the master server, and at time t, the i-th worker transmits to the master server a stale stochastic gradient $g_{t-\tau}$ computed at $w_{t-\tau}$. The master server makes an update to w_{t+1}, and passes it back to the i-th worker. Here the delay τ could either be deterministic or random.

Analytically, we will additionally need the following smoothness assumption:

- **B1.**(Smoothness) The loss function f is L-smooth (has L-Lipschitz gradients), i.e.,

  $$\|\nabla f(w) - \nabla f(z)\| \le L\|w - z\|, \forall w, z \in \mathcal{W},$$

  or equivalently,

  $$f(w) \le f(z) + \langle \nabla f(w), z - w\rangle + \frac{L}{2}\|z - w\|^2.$$

Assumption **B1** is critical when theoretically analyzing delayed gradients ([1, 26] and etc.). It casts some constraints on the delayed gradient saying that at each time t, the stale gradient $g_{t-\tau}$ cannot differ very much from the true gradient g_t. This is intuitively the case when updating with stale gradients can still converge. A counter example is, if the out-dated gradient is in the opposite direction of the true gradient, the update will make no sense and the algorithm may diverge.

5.1 Delay models

We mainly consider two types of delay models: constant delay and random bounded delay.

- **C1.** Constant delay: at each iteration t, the algorithm receives $\tilde{g}_{t-\tau)}$ for some constant $\tau(t) \equiv \tau$.
- **C2.** Bounded delay: at each iteration t, the algorithm receives $\tilde{g}_{t-\tau(t)}$, where $\mathbb{E}[\tau(t)^2] \le B^2 \le \infty$ for all t.

In the following analysis, for simplicity we intrinsically assume that some boundary and initial conditions of the delayed gradients are well handled, as this will not affect the main results. Besides, we assume that the map $t \mapsto t - \tau(t)$ is one-to-one, which means that each update can only be taken once. This is a trivial assumption in distributed architecture that could be easily satisfied.

5.2 Theoretical results

Definition 1. *The Bregman divergence between w and z w.r.t. a convex differentiable function f is*

$$D_f(w, q) = f(w) - f(z) - \langle \nabla f(z), w - z\rangle,$$

which is non-negative.

Lemma 6. $\mathbb{E}[\|w_{t+1} - w_t\|^2] \le \frac{\eta^2 d}{t(1-\beta_1)(1-\beta_2)(1-\gamma')}$, *with* $\gamma' = \beta_1/\beta_2$.

Proof. By the update rule of AMSGrad, we have

$$\mathbb{E}\big[\|w_{t+1} - w_t\|^2] = \mathbb{E}[\eta_t^2 \sum_{i=1}^{d} \frac{m_{t,i}^2}{\hat{v}_{t,i}}]$$

$$\le \mathbb{E}[\frac{\eta^2}{t} \sum_{i=1}^{d} \frac{\sum_{j=1}^{t}(1-\beta_{1,t})\prod_{p=1}^{t-j}\beta_{1,t-p+1}\tilde{g}_{j-\tau,i}^2}{(1-\beta_2)\sum_{j=1}^{t}\beta_2^{t-j}\tilde{g}_{j-\tau,i}^2}]$$

$$\le \frac{\eta^2}{t(1-\beta_2)}\mathbb{E}\big[\sum_{i=1}^{d} \frac{(\sum_{j=1}^{T}\prod_{p=1}^{t-j}\beta_{1,t-p+1})(\sum_{j=1}^{T}\prod_{p=1}^{t-j}\beta_{1,t-p+1}\tilde{g}_{j-\tau,i}^2)}{\sum_{j=1}^{t}\beta_2^{t-j}\tilde{g}_{j-\tau,i}^2}]$$

$$\le \frac{\eta^2}{t(1-\beta_1)(1-\beta_2)}\mathbb{E}\big[\sum_{i=1}^{d} \frac{\sum_{j=1}^{t}\beta_1^{t-j}\tilde{g}_{j-\tau,i}^2}{\sum_{j=1}^{t}\beta_2^{t-j}\tilde{g}_{j-\tau,i}^2}]$$

$$\le \frac{\eta^2 d}{t(1-\beta_1)(1-\beta_2)(1-\gamma')}.$$

□

Theorem 3. *Under Assumptions **A1-A4**, **B2** and delay model **C1**, suppose decaying learning rate* $\eta_t = \eta/\sqrt{t}$, *and* $\beta_{1,t} = \beta_1/t$. *Let* $\gamma = \beta_1/\sqrt{\beta_2} \le 1$ *and* $\gamma' = \beta_1/\beta_2$. *Then we have*

$$\mathbb{E}[\sum_{t=1}^{T}(f(w_{t+1}) - f(w^*))] \le C + D + E,$$

with

$$C = \frac{3dD^2G_\infty\sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}} + \frac{\eta\sqrt{\log T + 1}}{\sqrt{1-\beta_2}(1-\gamma)}\mathbb{E}\big[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|\big],$$

$$D = \tau dDG_\infty + \frac{(\tau+1)^2\eta^2 L}{2(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T + 2)$$
$$+ (\frac{1-p_{min}}{p_{min}}dG_\infty^2 + \frac{\sigma^2}{p_{min}} + \frac{\eta^2 d}{(1-\beta_1)(1-\beta_2)(1-\gamma')})\sqrt{T},$$

$$E = \frac{Ld\eta^2(\log T + 1)}{2(1-\beta_1)(1-\beta_2)(1-\gamma')},$$

where $w^* = \operatorname*{argmin}_{w\in\mathcal{W}} f(w)$, *and* $p_{min} = \min_{\substack{i\in[d],t\in[T]\\ p_{t,i}\neq 0}} p_{t,i}$.

Proof. (of Theorem 3) By convexity and L-Lipschitz continuity of ∇f, we have

$$\mathbb{E}\big[\sum_{t=1}^{T}(f(w_{t+1}) - f(w^*))\big]$$

$$\le \sum_{t=1}^{T}\langle \nabla f(w_t), w_{t+1} - w^*\rangle + \frac{L}{2}\sum_{t=1}^{T}\|w_t - w_{t+1}\|^2$$

$$= \underbrace{\mathbb{E}[\sum_{t=1}^{T}\langle \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle]}_{C} + \underbrace{\mathbb{E}[\sum_{t=1}^{T}\langle \nabla f(w_t) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle]}_{D}$$

$$\underbrace{+ \frac{L}{2}\mathbb{E}[\sum_{t=1}^{T}\|w_t - w_{t+1}\|^2].}_{E}$$

Now we seek to bound these three terms, respectively.

Step 1. For the first term, we have

$$C = \mathbb{E}[\langle \tilde{g}_{t-\tau}, w_t - w^*\rangle] + \mathbb{E}[\langle \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle].$$

For the first term, we can use the same procedures as Theorem 1, with $\tilde{g}_t$ replaced by $\tilde{g}_{t-\tau}$. It is easy to show that the bound for part A in (9) still holds in this case (when updated by $\tilde{g}_{t-\tau}$). On the other hand,

$$\mathbb{E}[\sum_{t=1}^{T}\langle \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle]$$

$$\leq \mathbb{E}[\sum_{t=1}^{T} \eta_t \sum_{i=1}^{d} \frac{\sum_{j=1}^{t} \beta_1^{t-j} \tilde{g}_{j-\tau,i} \tilde{g}_{t-\tau,i}}{\sqrt{(1-\beta_2) \sum_{j=1}^{t} \beta_2^{t-j} \tilde{g}_{j-\tau,i}^2}}]$$

$$\leq \mathbb{E}[\sum_{t=1}^{T} \frac{\eta}{\sqrt{t}\sqrt{1-\beta_2}} \sum_{i=1}^{d} \sum_{j=1}^{t} \gamma^{t-j} \tilde{g}_{t-\tau,i}]$$

$$\leq \frac{\eta}{\sqrt{1-\beta_2}(1-\gamma)} \sum_{i=1}^{d} \mathbb{E}[\sum_{t=1}^{T} \tilde{g}_{t-\tau,i} \frac{1}{\sqrt{t}}]$$

$$\leq \frac{\eta}{\sqrt{1-\beta_2}(1-\gamma)} \sum_{i=1}^{d} \mathbb{E}[\|\tilde{g}_{1:T,i}\|] \sqrt{\sum_{t=1}^{T} \frac{1}{t}}$$

$$\leq \frac{\eta\sqrt{1+\log T}}{\sqrt{1-\beta_2}(1-\gamma)} \sum_{i=1}^{d} \mathbb{E}[\|\tilde{g}_{1:T,i}\|].$$

To get the result, we appeal to Lemma 10, Cauchy-Schwartz inequality, and use the facts that $\beta_1 \leq 1$ and $\tau \geq 0$. As a result,

$$C \leq \frac{3dD^2 G_\infty \sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}} + \frac{\eta\sqrt{\log T + 1}}{\sqrt{1-\beta_2}(1-\gamma)} \mathbb{E}[\sum_{i=1}^{d} \|\tilde{g}_{1:T,i}\|].$$

Step 2. Regarding D, we have the decomposition

$$D = \sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^* \rangle] \quad (7)$$
$$+ \sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^* \rangle].$$

Lemma 7. $\log T + \frac{1}{T} \leq \sum_{t=1}^{T} \frac{1}{t} \leq \log T + 1$.

The first term admits the following lemma.

Lemma 8. *Under assumptions of Theorem 3, we have*

$$\sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^* \rangle]$$
$$\leq \tau dDG_\infty + \frac{(\tau+1)^2 \eta^2 L}{2(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log T + 2),$$

where $\gamma' = \beta_1/\beta_2$.

Proof. First, we will use the following well-known equality of Bregman divergence. $\forall z, b, c, d$, we have

$$\langle \nabla f(a) - \nabla f(b), c - d \rangle = D_f(d, a) - D_f(d, b) - D_f(c, a) + D_f(c, b).$$

Hence we have for any $t \in [T]$,

$$\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^* \rangle$$
$$\leq D_f(w^*, w_t) - D_f(w^*, w_{t-\tau}) - D_f(w_{t+1}, w_t) + D_f(w_{t+1}, w_{t-\tau}).$$

By assumption **B1**, we have

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

which implies

$$D_f(w_{t+1}, w_{t-\tau}) \leq \frac{L}{2} \|w_{t+1} - w_{t-\tau}\|^2.$$

Since Bregman divergence is non-negative, we obtain

$$\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^* \rangle$$
$$\leq D_f(w^*, w_t) - D_f(w^*, w_{t-\tau}) + \frac{L}{2} \|w_{t+1} - w_{t-\tau}\|^2, \quad (8)$$

which further gives

$$\sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^* \rangle]$$
$$\leq \mathbb{E}[\sum_{t=T-\tau+1}^{T} D_f(w^*, w_t) + \frac{L}{2} \sum_{t=1}^{T} \|w_{t+1} - w_{t-\tau}\|^2].$$

For the first term, notice that

$$\begin{aligned} \mathbb{E}[D_f(w^*, w_t)] &= \mathbb{E}[f(w^*) - f(w_t) - \langle \nabla f(w_t), w^* - w_t \rangle] \\ &\leq \mathbb{E}[\|\nabla f(w_t)\| \cdot \|w^* - w_t\|] \\ &\leq dDG_\infty, \end{aligned}$$

where the first inequality holds because the optimality of w^*. For the second part, by triangle inequality we have

$$\|w_{t+1} - w_{t-\tau}\|^2$$
$$\leq (\|w_{t+1} - w_t\| + \|w_t - w_{t-1}\| + \ldots + \|w_{t-\tau+1} - w_{t-\tau}\|)^2$$
$$= \sum_{k=0}^{\tau} \|w_{t-k+1} - w_{t-k}\|^2 + 2 \sum_{k=0}^{\tau-1} \sum_{p>k}^{\tau} \|w_{t-k+1} - w_{t-k}\| \cdot \|w_{t-p+1} - w_{t-p}\|$$
$$\leq (\tau+1) \sum_{k=0}^{\tau} \|w_{t-k+1} - w_{t-k}\|^2,$$

where the last line is due to Young's inequality. Therefore, using Lemma 6,

$$\mathbb{E}[\|w_{t+1} - w_{t-\tau}\|^2]$$
$$\leq \frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} \sum_{k=0}^{\tau} \frac{1}{t-k}$$
$$= \frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\sum_{s=1}^{t} \frac{1}{s} - \sum_{s=1}^{t-\tau-1} \frac{1}{s})$$
$$\leq \frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log \frac{t}{t-\tau-1} + 1)$$
$$\leq \frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log(1 + \frac{\tau+1}{t-\tau-1}) + 1)$$
$$\leq \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\frac{1}{t-\tau-1} + 2).$$

The last line holds because of the inequality $\log 1 + x \leq x$ for $x \geq 0$. Consequently, applying Lemma 10 gives

$$\mathbb{E}[\sum_{t=1}^{T} \|w_{t+1} - w_{t-\tau}\|^2]$$
$$\leq \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\sum_{t=1}^{T} \frac{1}{t-\tau-1} + 2)$$
$$\leq \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log T + 2).$$

Combining parts together, the proof is complete. □

For the second term in (10), we notice that

$$\mathbb{E}[\langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^* \rangle]$$

$$=\mathbb{E}[\langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_t - w^*\rangle + \langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle]$$
$$\le \frac{1}{2\sqrt{t}}\mathbb{E}[\|\nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}\|^2] + \frac{\sqrt{t}}{2}\mathbb{E}[\|w_{t+1} - w_t\|^2].$$

The expected value of the first term in second line equals to 0 since w_t, given the sigma-filed containing $\tilde{g}_1, ., , , \tilde{g}_{t-\tau-1}$, is independent of $\tilde{g}_{t-\tau}$, according to the updates of delayed AMSGrad algorithm, and $\tilde{g}_{t-\tau}$ is unbiased of $\nabla f(w_{t-\tau})$. Furthermore, we have the variance of compressed gradient $\tilde{g}_{t-\tau,i}$, $\forall i \in [d]$,

$$\mathbb{E}[\|\nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}\|^2] \le \frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}}.$$

Combining Lemma 5, Lemma 6 and Lemma 10, and sum over 1 to T, we derive

$$\sum_{t=1}^{T} \mathbb{E}[\langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle]$$
$$\le (\frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}} + \frac{\eta^2 d}{(1-\beta_1)(1-\beta_2)(1-\gamma')})\sqrt{T}.$$

Together with Lemma 11, we finally obtain

$$D \le \tau d D G_\infty + \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T + 2)$$
$$+ (\frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}} + \frac{\eta^2 d}{(1-\beta_1)(1-\beta_2)(1-\gamma')})\sqrt{T}.$$

Step 3. The term E follows from the proof in step 2. More specifically, by Lemma 6 we have

$$E \le \frac{L}{2}\sum_{t=1}^{T} \frac{d\eta^2}{t(1-\beta_1)(1-\beta_2)(1-\gamma')}$$
$$\le \frac{Ld\eta^2(\log T + 1)}{2(1-\beta_1)(1-\beta_2)(1-\gamma')}.$$

Combining parts together, the proof of theorem is complete. □

The above analysis implies the following corollaries on the convergence rate.

Corollary 1. *(Constant Delay) Under **C1** and the assumptions in Theorem 3, the convergence rate of delayed compressed AMSGrad is*

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) = O\left(\frac{\log T}{\sqrt{p_{min}}\sqrt{T}} + \frac{\tau}{T} + \frac{\tau^2 \log T}{T}\right),$$

Proof. By convexity, we know that

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) \le \frac{1}{T}\mathbb{E}[\sum_{t=1}^{T}(f(w_t) - f(w^*))],$$

Regarding all terms without delay parameter τ as constants, by Theorem 3 we obtain

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) \le C^* + D^* + E^*,$$

where

$$C^* = \frac{3dD^2G_\infty}{2\eta(1-\beta_1)\sqrt{p_{min}}\sqrt{T}} + \frac{\eta\sqrt{\log T + 1}}{\sqrt{1-\beta_2}(1-\gamma)T}\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|]$$
$$= \frac{c_0}{\sqrt{p_{min}}\sqrt{T}} + \frac{c_1\sqrt{\log T + 1}}{\sqrt{p_{min}}\sqrt{T}},$$

$$D^* \le \frac{c_2\tau}{T} + \frac{c_3(\tau+1)^2(\log T + 2)}{T} + \frac{c_4}{\sqrt{T}}$$
$$\le \frac{c_2\tau}{T} + \frac{c_3\tau^2\log T}{T} + \frac{c_4}{\sqrt{T}},$$

and $E^* = \frac{c_5 \log T}{T}$. Combining together and assume T is large enough such that $\log T \ge \sqrt{\log T + 1}$, we have

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) = O\left(\frac{\log T}{\sqrt{p_{min}}\sqrt{T}} + \frac{\tau}{T} + \frac{\tau^2 \log T}{T}\right),$$

This completes the proof. □

Alternatively, we may consider random $\tau(t)$, as described in **C2** delay model.

Corollary 2. *(Random Delay) Under **C2** and the assumptions in Theorem 3, we have*

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) = O\left(\frac{\log T}{\sqrt{p_{min}}\sqrt{T}} + \frac{B^2}{T} + \frac{B^2 \log T}{T}\right).$$

Proof. The proof follows from Corollary 1 but we take expectation over $\tau(t)$. Note that $\mathbb{E}[\tau(t)] \le \sqrt{\mathbb{E}[\tau(t)^2]} \le B^2$. We can also write

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) \le C^* + D^* + E^*,$$

with C^*, E^* same as those in the proof Corollary 1. We only need to consider D^* since it contains the delay. Regarding D^*, notice that we can get a modified version of Lemma 11 as follows.

Lemma 9. *Under assumptions of Theorem 3, but with **C2** delay, we have*

$$\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau(t)}), w_{t+1} - w^*\rangle]$$
$$\le (1 + 2B^2)dDG_\infty + \frac{(B+1)^2\eta^2 L}{2(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T + 2),$$

where $\gamma' = \beta_1/\beta_2$.

Proof. For $\forall t \in [T]$, we have

$$\langle \nabla f(w_t) - \nabla f(w_{t-\tau(t)}), w_{t+1} - w^*\rangle$$
$$\le D_f(w^*, w_t) - D_f(w^*, w_{t-\tau(t)}) + \frac{L}{2}\|w_{t+1} - w_{t-\tau(t)}\|^2.$$

Perform telescope summation, we get

$$\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau(t)}), w_{t+1} - w^*\rangle]$$
$$\le \mathbb{E}[\sum_{t:t+\tau(t)>T} D_f(w^*, w_t) + \frac{L}{2}\sum_{t=1}^{T}\|w_{t+1} - w_{t-\tau(t)}\|^2].$$

We know from Lemma 11 that For the first term, notice that

$$\mathbb{E}[D_f(w^*, w_t)] \le dDG_\infty,$$

so it remains to bound the expected cardinality of $s = \{t \in [T] : t + \tau(t) > T\}$. Using Markov inequality, we have

$$\mathbb{E}[card(s)] = \sum_{t=1}^{T} Pr[\tau(t)^2 > (T-t)^2]$$

$$\leq 1 + \mathbb{E}[\tau(t)^2] \sum_{t=1}^{T-1} \frac{1}{(T-t)^2} \leq 1 + 2B^2,$$

by our assumption and the well-known bound $\sum_{k=1}^{n} \frac{1}{k^2} < 2$, for $\forall n > 0$. For the second part, similarly we have

$$\|w_{t+1} - w_{t-\tau(t)}\|^2 \leq (\tau(t)+1) \sum_{k=0}^{\tau(t)} \|w_{t-k+1} - w_{t-k}\|^2,$$

Taking expectation yields,

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_{t-\tau}\|^2] &\leq \mathbb{E}[\frac{(\tau(t)+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} \sum_{k=0}^{\tau(t)} \frac{1}{t-k}] \\
&\leq \mathbb{E}[\frac{(\tau(t)+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log \frac{t}{t-\tau(t)-1} + 1)] \\
&\leq \mathbb{E}[\frac{(\tau(t)+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\frac{1}{t-\tau(t)-1} + 2)].
\end{aligned}$$

Summing over T, we obtain

$$\begin{aligned}
&\frac{L}{2} \sum_{t=1}^{T} \mathbb{E}[\|w_{t+1} - w_{t-\tau}\|^2] \\
\leq &\mathbb{E}[\frac{(\tau(t)+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\sum_{t=1}^{T} \frac{1}{t-\tau(t)-1} + 2)] \\
\leq &\mathbb{E}[\frac{(\tau(t)+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log T + 2)] \\
\leq &\frac{(B+1)^2\eta^2 L}{2(1-\beta_1)(1-\beta_2)(1-\gamma')} (\log T + 2),
\end{aligned}$$

where the second inequality is due to the assumption that $t \mapsto t - \tau(t)$ is one-to-one, so $\sum_{t=1}^{T} \frac{1}{t-\tau(t)-1} \leq \sum_{t=1}^{T} \frac{1}{t} \leq \log T + 1$ holds. The proof is complete by putting parts together. □

Following the proof of Corollary 1, after some simplification we get

$$\mathbb{E}[f(\hat{w}(T))] - f(w^*) = O\left(\frac{\log T}{\sqrt{p_{min}}\sqrt{T}} + \frac{B^2}{T} + \frac{B^2 \log T}{T} \right),$$

which completes the proof. □

Corollary 1 and Corollary 2 implies that similar to delayed SGD [1], the influence of delayed gradients is small when T is sufficiently large. For the convergence rate of both **C1** and **C2** delay models, we can decompose the first term as the model convergence rate incurred by the sparsified AMSGrad algorithm, and the remaining terms containing $\tau(t)$ as the extra cost brought by delayed gradients. Note that the $1/\sqrt{p_{min}}$ term only appears in the model convergence rate, but not in the delay effect. This suggests that, when $\sqrt{p_{min}}$ is small (i.e. high sparsity), the influence of delayed gradients would be comparatively small.

6 EXPERIMENTS

In this section, we demonstrate the effectiveness of our proposed sparsified AMSGrad algorithm on various datasets and models, and compare the adaptive sparsification strategy with other non-adaptive schemes. In addition, although this paper mainly focuses on the theoretical analysis of asynchronous distributed adaptive optimization, we also provide some empirical results that is helpful for practical applications.

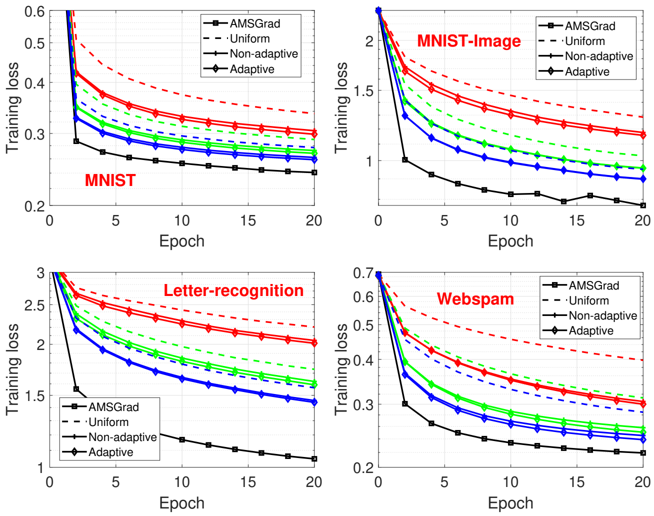

Figure 2: Training lose vs. epochs on logistic regression: training loss of different compressing strategies on AMSGrad. Red, green and blue curves represent sparsity=0.01,0.05,0.1, respectively.

6.1 Compressed AMSGrad

First, we test the sequential compressed adaptive optimization method with gradient sparsification on various learning models.

Datasets and models. We use three models in this experiment: logistic regression, fully-connected neural networks and deep neural nets. For the fully-connect architecture, we use a 2-layer ReLU network with 128 and 64 neurons per layer. The shallow neural network, along with the logistic regression, is used for classification on four datasets: *MNIST* [18], *MNIST-Rand* [17], *Letter-recognition* and *Webspam* [7]. *MNIST-Image* is a variant of *MNIST* with extra random noise. For deep learning experiment, we perform classification task on CIFAR-10 dataset [15] with All-CNN network [25]. The network consists of multiple convolutional layers and pooling layers, connected by rectified activation functions. Softmax function is used as the last layer before output for all the models.

Methods. The adaptive optimization framework adopted is AMSGrad, which also serves as the baseline method. We compare the following optimization algorithms.

- The vanilla AMSGrad method, with no gradient sparsification.
- AMSGrad+Uniform sparsification. Each gradient coordinate has equal chance of being chosen. In (1), we simply set $p_i = 1/d, \forall i$.
- AMSGrad+Non-adaptive sparsification (proposed for sparsified SGD [29]). The selection probability is given by (2). In Algorithm 2, we remove u_{t-1} terms from line 2, and set $p_{t,\mathcal{A}} = c \cdot |g_{t,\mathcal{A}}| / \sum_{i \in \mathcal{A}} |g_{t,i}|$. This gives solution identical to (2).

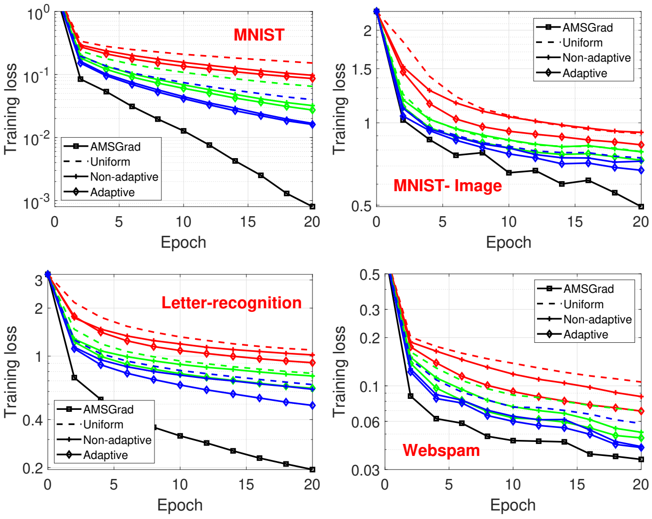

Figure 3: Training loss vs. epochs on 2-layer 128fc-64fc neural network. Red, green and blue curves represent sparsity=0.01,0.05,0.1, respectively.

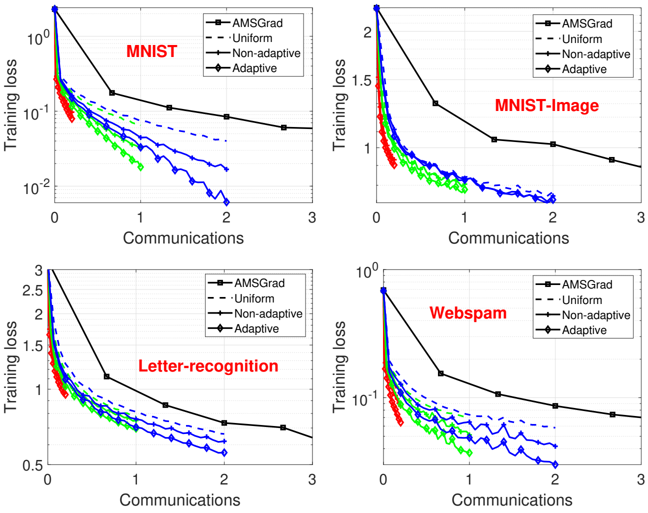

Figure 4: Test accuracy vs. epochs on 2-layer 128fc-64fc neural network. Red, green and blue curves represent sparsity=0.01,0.05,0.1, respectively.

- AMSGrad+Our proposed adaptive sparsification. We apply Algorithm 2 for gradient sparsification, where we set $\xi = 1/2, 1/4$ and report the best result.

To make a fair comparison on the effectiveness of gradient sparsification, we fix the learning rate $\eta = 0.001$, and use default hyper-parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$ for AMSGrad. The mini-batch size is 128. In neural networks, since the gradient magnitude in each layer is different, we apply sparsification individually for each layer. For all sparsification schemes, we test sparsity level $s = \{0.01, 0.05, 0.1\}$ by letting $c = s \cdot d$ in Algorithm 2.

Comparison of sparsification schemes. In Figure 2 and Figure 3, we report the training loss against number of epochs for 3 sparsity levels on logistic regression and 2-layer neural network, respectively. In the plots, red, green and blue curves represent $s = 0.01$, $s = 0.05$ and $s = 0.1$ correspondingly. From the plots, we clearly observe:

- In general, the the speed of convergence gets slower with higher target sparsity, in all the figures. This is consistent with our conclusions on Theorem 1.
- In all cases, the uniform sparsification strategy is significantly worse than adaptive sparsification, at all sparsity levels. For 2-layer neural network, similar performances can be observed for uniform strategy and non-adaptive method on *MNIST-Image* dataset.
- Our proposed adaptive sparsification strategy outperforms the non-adaptive approach, and the improvement is more substantial with high sparsity (e.g. $s = 0.01$).

REFERENCES

[1] Alekh Agarwal and John C. Duchi. 2011. Distributed Delayed Stochastic Optimization. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.* 873–881.

[2] Alham Fikri Aji and Kenneth Heafield. 2017. Sparse Communication for Distributed Gradient Descent. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017.* 440–445.

[3] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. 2017. QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA.* 1709–1720.

[4] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. 2018. SIGNSGD: Compressed Optimisation for Non-Convex Problems. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018.* 559–568.

[5] Chia-Yu Chen, Jungwook Choi, Daniel Brand, Ankur Agrawal, Wei Zhang, and Kailash Gopalakrishnan. 2018. AdaComp : Adaptive Residual Gradient Compression for Data-Parallel Distributed Training. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18).* New Orleans, LA, 2827–2835.

[6] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. 2019. On the Convergence of A Class of Adam-Type Algorithms for Non-Convex Optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019.*

[7] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[8] John C. Duchi, Elad Hazan, and Yoram Singer. 2010. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. In *COLT 2010 - The 23rd Conference on Learning Theory, Haifa, Israel, June 27-29, 2010.* 257–269.

[9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems (NIPS).* Montreal, Canada, 2672–2680.

[10] Alex Graves, Abdel-rahman Mohamed, and Geoffrey E. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* Vancouver, Canada, 6645–6649.

[11] Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. 2016. Deep Reinforcement Learning with a Natural Language Action Space. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL).* Berlin, Germany.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, (CVPR).* Las Vegas, NV, 770–778.

[13] Peng Jiang and Gagan Agrawal. 2018. A Linear Speedup Analysis of Distributed Deep Learning with Sparse and Quantized Communication. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.* 2530–2541.

[14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[15] Alex Krizhevsky et al. 2009. Learning multiple layers of features from tiny images. (2009).

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*. Lake Tahoe, NV, 1106–1114.

[17] Hugo Larochelle, Dumitru Erhan, Aaron C. Courville, James Bergstra, and Yoshua Bengio. 2007. An empirical evaluation of deep architectures on problems with many factors of variation. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*. 473–480.

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[19] Xiangru Lian, Yijun Huang, Yuncheng Li, and Ji Liu. 2015. Asynchronous Parallel Stochastic Gradient for Nonconvex Optimization. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. 2737–2745.

[20] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. 2018. Deep Gradient Compression: Reducing the Communication Bandwidth for Distributed Training. In *6th International Conference on Learning Representations (ICLR)*. Vancouver, Canada.

[21] Angelia Nedich, Dimitri P Bertsekas, and Vivek S Borkar. 2001. Distributed asynchronous incremental subgradient methods. *Studies in Computational Mathematics* 8, C (2001), 381–407.

[22] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. 2018. On the Convergence of Adam and Beyond. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

[23] Shaohuai Shi, Kaiyong Zhao, Qiang Wang, Zhenheng Tang, and Xiaowen Chu. 2019. A Convergence Analysis of Distributed SGD with Communication-Efficient Gradient Sparsification. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*. Macao, China, 3411–3417.

[24] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 7587 (2016), 484–489.

[25] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. 2015. Striving for Simplicity: The All Convolutional Net. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*.

[26] Suvrit Sra, Adams Wei Yu, Mu Li, and Alexander J. Smola. 2015. AdaDelay: Delay Adaptive Distributed Stochastic Convex Optimization. *CoRR* abs/1508.05003 (2015).

[27] Sebastian U. Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. 2018. Sparsified SGD with Memory. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*. 4452–4463.

[28] Hongyi Wang, Scott Sievert, Shengchao Liu, Zachary B. Charles, Dimitris S. Papailiopoulos, and Stephen Wright. 2018. ATOMO: Communication-efficient Learning via Atomic Sparsification. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*. 9872–9883.

[29] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. 2018. Gradient Sparsification for Communication-Efficient Distributed Optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*. Montréal, Canada, 1306–1316.

[30] Shuxin Zheng, Qi Meng, Taifeng Wang, Wei Chen, Nenghai Yu, Zhiming Ma, and Tie-Yan Liu. 2017. Asynchronous Stochastic Gradient Descent with Delay Compensation. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. 4120–4129.

[31] Martin Zinkevich, Alexander J. Smola, and John Langford. 2009. Slow Learners are Fast. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada*. 2331–2339.

7

Proof. (of Theorem 1) The proof follows from [22]. The update rule gives

$$w_{t+1} = \Pi_{\mathcal{W},\sqrt{\hat{V}_t}}(w_t - \eta_t \hat{V}_t^{-1/2} m_t) = \min_{w\in\mathcal{W}} \|\hat{V}_t^{1/4}(w - (w_t - \eta_t \hat{V}_t^{-1/2} m_t))\|.$$

By Lemma 3 on w_{t+1} and w^*, we have

$$\begin{aligned}\|\hat{V}_t^{1/4}(w_{t+1}-w^*)\|^2 &\le \|\hat{V}_t^{1/4}(w_t - (\eta_t \hat{V}_t^{-1/2} m_t - w^*))\| \\ &= \|\hat{V}_t^{1/4}(w_t - w^*)\|^2 + \eta_t^2 \|\hat{V}_t^{-1/4} m_t\|^2 \\ &+ 2\eta_t \langle \beta_{1,t} m_{t-1} + (1-\beta_{1,t})\tilde{g}_t, w_t - w^*\rangle.\end{aligned}$$

Rearranging terms, we have

$$\begin{aligned}&\langle \tilde{g}_t, w_t - w^*\rangle \\ \le &\frac{1}{2\eta_t(1-\beta_{1,t})}\big[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1}-w^*)\|^2\big] \\ &+ \frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2 + \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t \|\hat{V}_t^{-1/4} m_{t-1}\|^2 \\ &+ \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2.\end{aligned}$$

On the other hand, taking expectation and by the convexity of function f, we obtain

$$\begin{aligned}&\mathbb{E}\big[\sum_{t=1}^{T} f(w_t) - f(w^*)\big] \qquad (9) \\ \le &\underbrace{\mathbb{E}\big[\sum_{t=1}^{T}\langle \tilde{g}_t, w_t - w^*\rangle\big]}_{A} + \underbrace{\mathbb{E}\big[\sum_{t=1}^{T}\langle \nabla f(w_t) - \tilde{g}_t, w_t - w^*\rangle\big]}_{B}.\end{aligned}$$

Term B is zero since $\tilde{g}_t$ is unbiased of $\nabla f(w_t)$, and w_t is independent of the stochastic gradient $\tilde{g}_t$ given the sigma-field $\mathcal{G}_{t-1} = \{\tilde{g}_1, ..., \tilde{g}_{t-1}\}$. For term A, we have

$$\begin{aligned}A \le &\mathbb{E}\big[\frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2\big] \\ + &\mathbb{E}\Bigg[\sum_{t=1}^{T}\Big\{\frac{1}{2\eta_t(1-\beta_{1,t})}\big[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1}-w^*)\|^2\big] \\ &+ \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t\|\hat{V}_t^{-1/4} m_{t-1}\|^2 + \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg].\end{aligned}$$

Using Lemma 4, for the first term we have

$$\begin{aligned}&\mathbb{E}\big[\frac{\eta_t}{2(1-\beta_{1,t})}\|\hat{V}_t^{-1/4} m_t\|^2\big] \\ \le &\frac{\eta\sqrt{\log T + 1}}{(1-\beta_1)^2\sqrt{1-\beta_2}(1-\gamma)}\mathbb{E}\big[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|\big].\end{aligned}$$

The remaining terms can be bounded by

$$\begin{aligned}\le &\mathbb{E}\Bigg[\sum_{t=1}^{T}\Big\{\frac{1}{2\eta_t(1-\beta_{1,t})}\big[\|\hat{V}_t^{1/4}(w_t - w^*)\|^2 - \|\hat{V}_t^{1/4}(w_{t+1}-w^*)\|^2\big] \\ &+ \frac{\beta_{1,t}}{2(1-\beta_{1,t})}\eta_t\|\hat{V}_t^{-1/4} m_{t-1}\|^2 + \frac{\beta_{1,t}}{2\eta_t(1-\beta_{1,t})}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg] \\ \le &\mathbb{E}\Bigg[\frac{\|\hat{V}_t^{1/4}(w_1 - w^*)\|^2}{2\eta(1-\beta_1)} + \frac{1}{2(1-\beta_1)}\sum_{t=2}^{T}\Big\{\frac{\|\hat{V}_t^{1/4}(w_t - w^*)\|^2}{\eta_t} \\ &- \frac{\|\hat{V}_{t-1}^{1/4}(w_t - w^*)\|^2}{\eta_{t-1}}\Big\} + \sum_{t=1}^{T}\Big\{\frac{\beta_{1,t}}{2\eta_t(1-\beta_1)}\|\hat{V}_t^{1/4}(w_t - w^*)\|^2\Big\}\Bigg] \\ = &\mathbb{E}\Bigg[\underbrace{\sum_{i=1}^{d}\frac{\hat{v}_{1,i}^{1/2}(w_{1,i}-w_i^*)^2}{2\eta(1-\beta_1)} + \sum_{t=2}^{T}\sum_{i=1}^{d}\big[\frac{\hat{v}_{t,i}^{1/2}}{\eta_t} - \frac{\hat{v}_{t-1,i}^{1/2}}{\eta_{t-1}}\big]\frac{(w_{t,i}-w_i^*)^2}{2(1-\beta_1)}}_{I} \\ &+ \underbrace{\frac{1}{2(1-\beta_1)}\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{\beta_{1,t}\hat{v}_{t,i}^{1/2}(w_{t,i}-w_i^*)^2}{\eta_t}}_{II}\Bigg].\end{aligned}$$

Regarding II, we have

$$\begin{aligned}\mathbb{E}[II] = &\frac{1}{2(1-\beta_1)}\mathbb{E}\big[\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{\beta_1\hat{v}_{t,i}^{1/2}}{\eta\sqrt{t}}(w_{t,i}-w_i^*)^2\big] \\ \le &\frac{D^2}{2\eta(1-\beta_1)}\sum_{t=1}^{T}\sum_{i=1}^{d}\frac{1}{\sqrt{t}}\mathbb{E}[\hat{v}_{t,i}^{1/2}] \\ \le &\frac{dD^2 G_\infty\sqrt{T}}{\eta(1-\beta_1)\sqrt{p_{min}}},\end{aligned}$$

where the first inequality is due to Assumption **A2** and the fact that $\beta_{1,t} = \beta_1/t \le 1$, and the last inequality is a consequence of Lemma 2 and Lemma 5. For I, we observe that it is a telescopic sum. After cancelling terms, we obtain

$$\begin{aligned}\mathbb{E}[I] = &\frac{1}{2\eta_T(1-\beta_1)}\sum_{i=1}^{d}\mathbb{E}[\hat{v}_{T,i}^{1/2}(w_{t,i}-w_i^*)^2] \\ \le &\frac{dD^2 G_\infty\sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}},\end{aligned}$$

where the inequality is because $\eta_T = \eta/\sqrt{T}$, Assumption **A2** and Lemma 2. Combining parts together, the theorem is proved. □

8

Proof. (of Theorem 3) By convexity and L-Lipschitz continuity of ∇f, we have

$$\begin{aligned}&\mathbb{E}\big[\sum_{t=1}^{T}(f(w_{t+1}) - f(w^*))\big] \\ \le &\sum_{t=1}^{T}\langle \nabla f(w_t), w_{t+1} - w^*\rangle + \frac{L}{2}\sum_{t=1}^{T}\|w_t - w_{t+1}\|^2 \\ = &\underbrace{\mathbb{E}\big[\sum_{t=1}^{T}\langle \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle\big]}_{C} + \underbrace{\mathbb{E}\big[\sum_{t=1}^{T}\langle \nabla f(w_t) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle\big]}_{D} \\ &+ \underbrace{\frac{L}{2}\mathbb{E}\big[\sum_{t=1}^{T}\|w_t - w_{t+1}\|^2\big]}_{E}.\end{aligned}$$

Now we seek to bound these three terms, respectively.

Step 1. For the first term, we have

$$C = \mathbb{E}[\langle \tilde{g}_{t-\tau}, w_t - w^*\rangle] + \mathbb{E}[\langle \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle].$$

For the first term, we can use the same procedures as Theorem 1, with $\tilde{g}_t$ replaced by $\tilde{g}_{t-\tau}$. It is easy to show that the bound for part A in (9) still holds in this case (when updated by $\tilde{g}_{t-\tau}$). On the other hand,

$$\begin{aligned}
&\mathbb{E}[\sum_{t=1}^{T}\langle \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle]\\
\le&\mathbb{E}[\sum_{t=1}^{T}\eta_t\sum_{i=1}^{d}\frac{\sum_{j=1}^{t}\beta_1^{t-j}\tilde{g}_{j-\tau,i}\tilde{g}_{t-\tau,i}}{\sqrt{(1-\beta_2)\sum_{j=1}^{t}\beta_2^{t-j}\tilde{g}_{j-\tau,i}^2}}]\\
\le&\mathbb{E}[\sum_{t=1}^{T}\frac{\eta}{\sqrt{t}\sqrt{1-\beta_2}}\sum_{i=1}^{d}\sum_{j=1}^{t}\gamma^{t-j}\tilde{g}_{t-\tau,i}]\\
\le&\frac{\eta}{\sqrt{1-\beta_2}(1-\gamma)}\sum_{i=1}^{d}\mathbb{E}[\sum_{t=1}^{T}\tilde{g}_{t-\tau,i}\frac{1}{\sqrt{t}}]\\
\le&\frac{\eta}{\sqrt{1-\beta_2}(1-\gamma)}\sum_{i=1}^{d}\mathbb{E}[\|\tilde{g}_{1:T,i}\|]\sqrt{\sum_{t=1}^{T}\frac{1}{t}}\\
\le&\frac{\eta\sqrt{1+\log T}}{\sqrt{1-\beta_2}(1-\gamma)}\sum_{i=1}^{d}\mathbb{E}[\|\tilde{g}_{1:T,i}\|].
\end{aligned}$$

To get the result, we appeal to Lemma 10, Cauchy-Schwartz inequality, and use the facts that $\beta_1 \le 1$ and $\tau \ge 0$. As a result,

$$C \le \frac{3dD^2G_\infty\sqrt{T}}{2\eta(1-\beta_1)\sqrt{p_{min}}} + \frac{\eta\sqrt{\log T + 1}}{\sqrt{1-\beta_2}(1-\gamma)}\mathbb{E}[\sum_{i=1}^{d}\|\tilde{g}_{1:T,i}\|].$$

Step 2. Regarding D, we have the decomposition

$$\begin{aligned}
D = &\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^*\rangle] \qquad (10)\\
&+\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle].
\end{aligned}$$

Lemma 10. $\log T + \frac{1}{T} \le \sum_{t=1}^{T}\frac{1}{t} \le \log T + 1$.

The first term admits the following lemma.

Lemma 11. *Under assumptions of Theorem 3, we have*

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^*\rangle]\\
\le&\tau dDG_\infty + \frac{(\tau+1)^2\eta^2L}{2(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T + 2),
\end{aligned}$$

where $\gamma' = \beta_1/\beta_2$.

Proof. First, we will use the following well-known equality of Bregman divergence. $\forall z, b, c, d$, we have

$$\langle \nabla f(a) - \nabla f(b), c - d\rangle = D_f(d,a) - D_f(d,b) - D_f(c,a) + D_f(c,b).$$

Hence we have for any $t \in [T]$,

$$\begin{aligned}
&\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^*\rangle\\
\le&D_f(w^*, w_t) - D_f(w^*, w_{t-\tau}) - D_f(w_{t+1}, w_t) + D_f(w_{t+1}, w_{t-\tau}).
\end{aligned}$$

By assumption **B1**, we have

$$f(w_{t+1}) \le f(w_{t-\tau}) + \langle \nabla f(w_{t-\tau}), w_{t+1} - w_{t-\tau}\rangle + \frac{L}{2}\|w_{t+1} - w_{t-\tau}\|^2,$$

which implies

$$D_f(w_{t+1}, w_{t-\tau}) \le \frac{L}{2}\|w_{t+1} - w_{t-\tau}\|^2.$$

Since Bregman divergence is non-negative, we obtain

$$\begin{aligned}
&\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^*\rangle\\
\le&D_f(w^*, w_t) - D_f(w^*, w_{t-\tau}) + \frac{L}{2}\|w_{t+1} - w_{t-\tau}\|^2, \qquad (11)
\end{aligned}$$

which further gives

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[\langle \nabla f(w_t) - \nabla f(w_{t-\tau}), w_{t+1} - w^*\rangle]\\
\le&\mathbb{E}[\sum_{t=T-\tau+1}^{T}D_f(w^*, w_t) + \frac{L}{2}\sum_{t=1}^{T}\|w_{t+1} - w_{t-\tau}\|^2].
\end{aligned}$$

For the first term, notice that

$$\begin{aligned}
\mathbb{E}[D_f(w^*, w_t)] &= \mathbb{E}[f(w^*) - f(w_t) - \langle \nabla f(w_t), w^* - w_t\rangle]\\
&\le \mathbb{E}[\|\nabla f(w_t)\| \cdot \|w^* - w_t\|]\\
&\le dDG_\infty,
\end{aligned}$$

where the first inequality holds because the optimality of w^*. For the second part, by triangle inequality we have

$$\begin{aligned}
&\|w_{t+1} - w_{t-\tau}\|^2\\
\le&(\|w_{t+1} - w_t\| + \|w_t - w_{t-1}\| + ... + \|w_{t-\tau+1} - w_{t-\tau}\|)^2\\
=&\sum_{k=0}^{\tau}\|w_{t-k+1} - w_{t-k}\|^2 + 2\sum_{k=0}^{\tau-1}\sum_{p>k}^{\tau}\|w_{t-k+1} - w_{t-k}\| \cdot \|w_{t-p+1} - w_{t-p}\|\\
\le&(\tau+1)\sum_{k=0}^{\tau}\|w_{t-k+1} - w_{t-k}\|^2,
\end{aligned}$$

where the last line is due to Young's inequality. Therefore, using Lemma 6,

$$\begin{aligned}
&\mathbb{E}[\|w_{t+1} - w_{t-\tau}\|^2]\\
\le&\frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}\sum_{k=0}^{\tau}\frac{1}{t-k}\\
=&\frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\sum_{s=1}^{t}\frac{1}{s} - \sum_{s=1}^{t-\tau-1}\frac{1}{s})\\
\le&\frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log\frac{t}{t-\tau-1} + 1)\\
\le&\frac{(\tau+1)\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log(1 + \frac{\tau+1}{t-\tau-1}) + 1)\\
\le&\frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\frac{1}{t-\tau-1} + 2).
\end{aligned}$$

The last line holds because of the inequality $\log 1 + x \le x$ for $x \ge 0$. Consequently, applying Lemma 10 gives

$$\mathbb{E}[\sum_{t=1}^{T}\|w_{t+1} - w_{t-\tau}\|^2]$$

$$\le \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\sum_{t=1}^{T}\frac{1}{t-\tau-1}+2)$$

$$\le \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T+2).$$

Combining parts together, the proof is complete. □

For the second term in (10), we notice that

$$\mathbb{E}[\langle \triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle]$$

$$= \mathbb{E}[\langle \triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_t - w^*\rangle + \langle \triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w_t\rangle]$$

$$\le \frac{1}{2\sqrt{t}}\mathbb{E}[\|\triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}\|^2] + \frac{\sqrt{t}}{2}\mathbb{E}[\|w_{t+1} - w_t\|^2].$$

The expected value of the first term in second line equals to 0 since w_t, given the sigma-filed containing $\tilde{g}_1, ., , , \tilde{g}_{t-\tau-1}$, is independent of $\tilde{g}_{t-\tau}$, according to the updates of delayed AMSGrad algorithm, and $\tilde{g}_{t-\tau}$ is unbiased of $\triangledown f(w_{t-\tau})$. Furthermore, we have the variance of compressed gradient $\tilde{g}_{t-\tau,i}$, $\forall i \in [d]$,

$$\mathbb{E}[\|\triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}\|^2] \le \frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}}.$$

Combining Lemma 5, Lemma 6 and Lemma 10, and sum over 1 to T, we derive

$$\sum_{t=1}^{T}\mathbb{E}[\langle \triangledown f(w_{t-\tau}) - \tilde{g}_{t-\tau}, w_{t+1} - w^*\rangle]$$

$$\le (\frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}} + \frac{\eta^2 d}{(1-\beta_1)(1-\beta_2)(1-\gamma')})\sqrt{T}.$$

Together with Lemma 11, we finally obtain

$$D \le \tau d D G_\infty + \frac{(\tau+1)^2\eta^2}{(1-\beta_1)(1-\beta_2)(1-\gamma')}(\log T+2)$$

$$+ (\frac{1-p_{min}}{p_{min}} dG_\infty^2 + \frac{\sigma^2}{p_{min}} + \frac{\eta^2 d}{(1-\beta_1)(1-\beta_2)(1-\gamma')})\sqrt{T}.$$

Step 3. The term E follows from the proof in step 2. More specifically, by Lemma 6 we have

$$E \le \frac{L}{2}\sum_{t=1}^{T}\frac{d\eta^2}{t(1-\beta_1)(1-\beta_2)(1-\gamma')}$$

$$\le \frac{Ld\eta^2(\log T+1)}{2(1-\beta_1)(1-\beta_2)(1-\gamma')}.$$

Combining parts together, the proof of theorem is complete. □