Distributed Adaptive Learning with Gradient Compression

Anonymous Author(s)
Affiliation
Address
email

Abstract

In this paper, we present a novel optimization algorithm for single-machine and distributed learning, based on sparsification and error feedback techniques to lighten the communications between a central server and distributed workers. The method we introduce builds on the adaptivity of the AMSGrad method for nonconvex optimization, and includes a TopK operation to alleviate any communication bottleneck between a large amount of devices and a central computing server, combined with a correction of the natural bias induced by the latter compression operator. Despite the sparsity induced by our algorithm, we show that SPARS-AMSreaches a stationary point in $\mathcal{O}(1/\sqrt{T})$ iterations, matching that of state-of-the-art single-machine methods. We illustrate on benchmark datasets the effectiveness of our method both under the single-machine and distributed settings.

1 Introduction

Deep neural network has achieved the state-of-the-art learning performance on numerous AI applications, e.g., computer vision [23, 26, 47], Natural Language Processing [25, 54, 58], Reinforcement Learning [37, 45] and recommendation systems [16, 49]. With the increasing size of both data and deep networks, standard single machine training confronts with at least two major challenges:

- Due to the limited computing power of a single machine, it would take a long time to process the massive number of data samples—training would be slow.
- In many practical scenarios, data are typically stored in multiple servers, possibly at different locations, due to the storage constraints (massive user behavior data, Internet images, etc.) or privacy reasons [11]. Transmitting data might be costly.

Distributed learning framework [18] has been a common training strategy to tackle the above two issues. For example, in centralized distributed stochastic gradient descent (SGD) protocol, data are located at N local nodes, at which the gradients of the model are computed in parallel. In each iteration, a central server aggregates the local gradients, updates the global model, and transmits back the updated model to the local nodes for subsequent gradient computation. As we can see, this setting naturally solves aforementioned issues: 1) We use N computing nodes to train the model, so the time per training epoch can be largely reduced; 2) There is no need to transmit the local data to central server. Besides, distributed training also provides stronger error tolerance since the training process could continue even one local machine breaks down. As a result of these advantages, there has been a surge of study and applications on distributed systems [10, 39, 20, 24, 27, 35, 33].

Among many optimization strategies, SGD is still the most popular prototype in distributed training for its simplicity and effectiveness [14, 1, 36]. Yet, when the deep learning model is very large, the communication between local nodes and central server could be expensive. Burdensome gradient transmission would slow down the whole training system, or even be impossible because of the limited bandwidth in some applications. Thus, reducing the communication cost in distributed SGD has become an active topic, and an important ingredient of large-scale distributed systems (e.g. [42]). Solutions based on quantization, sparsification and other compression techniques of the local gradients are proposed, e.g., [4, 50, 48, 46, 3, 7, 17, 52, 28]. As one would expect, in most approaches, there exists a trade-off between compression and learning performance. In general, larger bias and variance of the compressed gradients usually bring more significant performance downgrade in terms of convergence [46, 2]. Interestingly, studies (e.g., [31]) show that the technique of *error feedback* is able to remedy the issue of such biased compressors, achieving same convergence rate as full-gradient SGD.

On the other hand, in recent years, adaptive optimization algorithms (e.g. AdaGrad [21], Adam [32] and AMSGrad [41]) have become popular because of their superior empirical performance. These methods use different implicit learning rates for different coordinates that keep changing adaptively throughout the training process, based on the learning trajectory. In many learning problems, adaptive methods have been shown to converge faster than SGD, sometimes with better generalization as well. However, the body of literature that combines adaptive methods with distributed training is still very limited. In this papar, we propose a distributed optimization algorithm with AMSGrad as the backbone, along with Top-k sparsification to reduce the communication cost.

1.1 Our contributions

We develop a simple optimization leveraging the adaptivity of AMSGrad, and the computational virtue of TopK sparsification, for tackling a large finite-sum of nonconvex objective functions.

Our technique is shown to be both theoretically and empirically effective under *the classical centralized setting* and *the distributed setting*.

In this contribution,

- We derive a sparsified AMSGrad with error feedback, called SPARS-AMS, with a single machine and provide its decentralized counter part.
- We provide a non-asymptotic convergence rate under each setting,
- We highlight the effectiveness of both methods through several numerical experiments

2 Related Work

2.1 Distributed SGD with compressed gradients

Quantization. As we mentioned before, SGD is the most commonly adopted optimization method in distributed training of deep neural nets. To reduce the expensive communication in large-scale distributed systems, extensive works have considered various compression techniques applied to the gradient transaction procedure. The first strategy is quantization. [19] condenses 32-bit floating numbers into 8-bits when representing the gradients. [42, 7, 31, 8] use the extreme 1-bit information (sign) of the gradients, combined with tricks like momentum, majority vote and memory. Other quantization-based methods include QSGD [4, 51, 57] and LPC-SVRG [55], leveraging unbiased stochastic quantization. The saving in communication of quantization methods is moderate: for example, 8-bit quantization reduces the cost to 25% (compared with 32-bit full-precision). Even in the extreme 1-bit case, the largest compression ratio is around $1/32 \approx 3.1\%$.

Sparsification. Gradient sparsification is another popular solution which may provide higher compression rate. Instead of commuting the full gradient, each local worker only passes a few coordinates to the central server and zeros out the others. Thus, we can more freely choose higher compression ratio (e.g., 1%, 0.1%), still achieving impressive performance in many applications [34]. Stochastic sparsification methods, including uniform sampling and magnitude based sampling [48], select coordinates based on some sampling probability yielding unbiased gradient compressors. Deterministic methods are simpler, e.g., Random-k, Top-k [46, 44] (selecting k elements with largest magnitude), Deep Gradient Compression [34], but usually lead to biased gradient estimation. In [28], the central server identifies heavy-hitters from the count-sketch [12] of the local gradients, which can be regarded as a noisy variant of Top-k strategy. More applications and analysis of compressed distributed SGD can be found in [30, 43, 5, 6, 29], among others.

Error Feedback. Biased gradient estimation, which is a consequence of many aforementioned methods (e.g., signSGD, Top-k), undermines the model training, both theoretically and empirically, with slower convergence and worse generalization [2, 9]. The technique of *error feedback* is able to "correct for the bias" and fix the problems. In this procedure, the difference between the true stochastic gradient and the compressed one is accumulated locally, which is then added back to the local gradients in later iterations. [46, 31] prove the $\mathcal{O}(\frac{1}{T})$ and $\mathcal{O}(\frac{1}{\sqrt{T}})$ convergence rate of EF-SGD in strongly convex and non-convex setting respectively, matching the rates of vanilla SGD [40, 22].

2.2 Adaptive optimization

In each SGD update, all the gradient coordinates share a same learning rate, either constant or decreasing over iterations. Adaptive optimization methods cast different learning rate on each dimension. AdaGrad [21] divides the gradient element-wisely by $\sqrt{\sum_{t=1}^{T} g_t^2} \in \mathbb{R}^d$, where $g_t \in \mathbb{R}^d$ is the gradient vector at time t and d is the model dimensionality. Thus, it intrinsically assigns different learning rates to different coordinates throughout the training—elements with smaller previous gradient magnitude tend to move a larger step. AdaGrad has been shown to perform well especially under some sparsity structure. AdaDelta [56] and Adam [32] introduce momentum and moving average of second moment estimation into AdaGrad which lead to better performance. AMSGrad [41] fixes the potential convergence issue of Adam, which will serve as the prototype in this paper. We present the psudocode in Algorithm . In general, adaptive optimization methods are easier to tune in practice, and usually exhibit faster convergence than SGD. Thus, they have been widely used in training deep learning models in language and computer vision applications, e.g., [15, 53, 59]. In distributed setting, the work [38] proposes a decentralized system in online optimization. However, communication efficiency is not considered. The recent work [13] is the most relevant to our paper. Yet, their method is based on Adam, and requires every local node to store a local estimation of first and second moment, thus being less efficient. We will present more detailed comparison in Section 3.

3 Communication-Efficient Adaptive Optimization

3.1 Gradient Compressors

In this paper, we mainly consider deterministic q-deviate compressors defined as below.

Assumption 1. *We say a compressor* $\mathcal{C} : \mathbb{R}^d \mapsto \mathbb{R}^d$ *is* q*-deviate if for* $\forall x \in \mathbb{R}^d$, $\exists\, 0 \leq q < 1$ *such that* $\|\mathcal{C}(x) - x\| \leq q\, \|x\|$.

Note that, smaller q indicates better approximation of the true gradient, and $q = 0$ implies no compression, i.e. $\mathcal{C}(x) = x$. We give two popular and highly efficient q-deviate compressors that will be compared in this paper.

Definition 1 (Top-k)**.** *For* $x \in \mathbb{R}^d$, *denote* $\mathcal{S}$ *as the size-*k *set of* $i \in [d]$ *with largest* k *magnitude* $|x_i|$. *The **Top-**k *compressor is defined as* $\mathcal{C}(x) = x_i$, *if* $i \in \mathcal{S}$; $\mathcal{C}(x) = 0$ *otherwise.*

Definition 2 (SIGN)**.** *For* $x \in \mathbb{R}^d$, *define the **SIGN** compressor as* $\mathcal{C}(x) = sign(x) \times \frac{1}{d}\sum_{i=1}^{d} |x_i|$.

Remark 1. *Here the scalar, mean magnitude, multiplied to* $sign(x)$ *ensures* $0 \leq q < 1$ *as required by Assumption 1, which can be shown by Cauchy-Schwartz inequality. In implementation, this scalar can be arbitrary since we can offset its influence by tuning the learning rate.*

Most modern machine learning tasks can be casted as a large finite-sum optimization problem written as:

$$\min_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} f_i(\theta) \tag{1}$$

where n denotes the number of workers, f_i represents the average loss for worker i and θ the global model parameter taking value in Θ, a subset of $\mathbb{R}^d$.

Some related work:

[31] develops variant of signSGD (as a biased compression schemes) for distributed optimization. Contributions are mainly on this error feedback variant. In [44], the authors provide theoretical results on the convergence of sparse Gradient SGD for distributed optimization (we want that for AMS here). [46] develops a variant of distributed SGD with sparse gradients too. Contributions include a memory term used while compressing the gradient (using top k for instance). Speeding up the convergence in $\frac{1}{T^3}$.

Consider standard synchronous distributed optimization setting. AMSGrad is used as the prototype, and the local workers is only in charge of gradient computation.

3.2 SPARS-AMS with Error Feedback

The key difference (and interesting part) of our TopK AMSGrad compared with the following arxiv paper "Quantized Adam"https://arxiv.org/pdf/2004.14180.pdf is that, in our model only gradients are transmitted. In "QAdam", each local worker keeps a local copy of moment estimator m and v, and compresses and transmits m/v as a whole. Thus, that method is very much like the sparsified distributed SGD, except that g is changed into m/v. In our model, the moment estimates m and v are computed only at the central server, with the compressed gradients instead of the full gradient. This would be the key (and difficulty) in convergence analysis.

Algorithm 1 SPARS-AMS for Distributed Learning

Input: parameter β_1, β_2, learning rate η_t.
Initialize: central server parameter $\theta_0 \in \Theta \subseteq \mathbb{R}^d$; $e_{1,i} = 0$ the error accumulator for each worker; sparsity parameter k; n local workers; $m_0 = 0$, $v_0 = 0$, $\hat{v}_0 = 0$
for $t = 1$ to T **do**
  parallel for worker $i \in [n]$ **do**:
    Receive model parameter θ_t from central server
    Compute stochastic gradient $g_{t,i}$ at θ_t
    Compute $\tilde{g}_{t,i} = TopK(g_{t,i} + e_{t,i}, k)$
    Update the error $e_{t+1,i} = e_{t,i} + g_{t,i} - \tilde{g}_{t,i}$
    Send $\tilde{g}_{t,i}$ back to central server
  end parallel
  Central server do:
  $\bar{g}_t = \frac{1}{n}\sum_{i=1}^{N} \tilde{g}_{t,i}$
  $m_t = \beta_1 m_{t-1} + (1-\beta_1)\bar{g}_t$
  $v_t = \beta_2 v_{t-1} + (1-\beta_2)\bar{g}_t^2$
  $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$
  Update the global model $\theta_{t+1} = \theta_t - \eta_t \frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}$
end for

4 Non-Asymptotic Convergence Analysis for the Single Machine and Decentralized settings

Several mild assumptions to make: Nonconvex and smooth loss function, unbiased stochastic gradient, bounded variance of the gradient, bounded norm of the gradient, control of the distance between the true gradient and its sparse variant.

Check [13] starting with single machine and extending to distributed settings (several machines).

Under the distributed setting, the goal is to derive an upper bound to the second order moment of the gradient of the objective function at some iteration $T_f \in [1, T]$.

We begin by making the following assumptions.

Assumption 2. *(Smoothness) For $i \in [\![n]\!]$, f_i is L-smooth:* $\|\nabla f_i(\theta) - \nabla f_i(\vartheta)\| \leq L \|\theta - \vartheta\|$.

Assumption 3. *(Unbiased and Bounded gradient **per worker**) For any iteration index $t > 0$ and worker index $i \in [\![n]\!]$, the stochastic gradient is unbiased and bounded from above:* $\mathbb{E}[g_{t,i}] = \nabla f_i(\theta_t)$ *and* $\|g_{t,i}\| \leq G_i$.

Assumption 4. *(Bounded variance **per worker**) For any iteration index $t > 0$ and worker index $i \in [\![n]\!]$, the variance of the noisy gradient is bounded: $\mathbb{E}[|g_{t,i} - \nabla f_i(\theta_t)|^2] < \sigma_i^2$.*

Denote by $Q(\cdot)$ the quantization operator Line 7 of Algorithm 1, which takes as input a gradient vector and returns a quantized version of it, and note $\tilde{g} := Q(g)$. Assume that

Denote for all $\theta \in \Theta$:

$$f(\theta) := \frac{1}{n} \sum_{i=1}^{n} f_i(\theta)\,, \tag{2}$$

where n denotes the number of workers.

Decentralized Workers Setting: The main theorem in the decentralized setting reads:

Theorem 1. *Under Assumption 2 to Assumption 4, the sequence of iterates $\{\theta_t\}_{t>0}$ output from Algorithm 1 satisfies:*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \frac{\mathbb{E}[f(\theta_0) - f(\theta_T)]}{\Delta_1 \eta_t T} + d\frac{\Delta_3}{\Delta_1 \eta_t T} + \frac{\Delta_2}{\Delta_1 T} + \frac{1-\beta_1}{\Delta_1}\epsilon^{-\frac{1}{2}}\sqrt{(q^2+1)}G^2 \tag{3}$$

where $\{\eta_t\}_{t>0}$ is the sequence of stepsizes and:

$$\begin{aligned} &\Delta_1 := \frac{(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \quad , \quad \Delta_2 := q^2 + \frac{G^2}{\epsilon 2n^2}\overline{\beta}_1 \\ &\Delta_3 := \left(\frac{L}{2} + 1 + \frac{\beta_1 L}{1-\beta_1}\right)(1-\beta_2)^{-1}(1-\frac{\beta_1^2}{\beta_2})^{-1} \end{aligned} \tag{4}$$

We remark from this bound in Theorem 1, that the more quantization we apply to our gradient vectors ($q \uparrow$), the larger the upper bound of the stationary condition is, *i.e.,* the slower the algorithm is. This is intuitive as using compressed quantities will definitely impact the algorithm speed. We will observe in the numerical section below that a trade-off on the level of quantization q can be found to achieve similar speed of convergence with less computation resources used throughout the training.

Corollary 1. *Under Assumption 2 to Assumption 4,setting the stepsize as $\eta_t = L\sqrt{\frac{n}{T}}$, the sequence of iterates $\{\theta_t\}_{t>0}$ output from Algorithm 1 satisfies:*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \mathcal{O}(\frac{1}{L\sqrt{n}T} + d\frac{L}{\sqrt{n}T} + \frac{1}{T}),$$

Single Machine Setting: We first provide the formulation of our method in the single machine settings in Algorithm 2. Here, the data and the computation are all performed on a single machine.

Algorithm 2 SPARS-AMS : Single machine setting

Input: parameter β_1, β_2, learning rate η_t.
Initialize: central server parameter $\theta_1 \in \Theta \subseteq \mathbb{R}^d$; $e_1 = 0$ the error accumulator; sparsity parameter k; $m_0 = 0$, $v_0 = 0$, $\hat{v}_0 = 0$
for $t = 1$ to T **do**
    Compute stochastic gradient $g_t = g_{t,i_t}$ at θ_t for randomly sampled index i_t
    Compute $\tilde{g}_t = TopK(g_t + e_t, k)$
    Update the error $e_{t+1} = e_t + g_t - \tilde{g}_t$
    $m_t = \beta_1 m_{t-1} + (1-\beta_1)\tilde{g}_t$
    $v_t = \beta_2 v_{t-1} + (1-\beta_2)\tilde{g}_t^2$
    $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$
    Update the global model $\theta_{t+1} = \theta_t - \eta_t \frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}$
end for

The convergence rate of the vector of parameters estimated via Algorithm 2 is given below:

Theorem 2. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize $\{\eta_t\}_{t>0} = \frac{1}{\sqrt{T}}$, the sequence of iterates $\{\theta_t\}_{t>0}$ output from Algorithm 2 satisfies:*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \mathcal{O}(\frac{1}{\sqrt{T}} + \frac{1}{T}),$$

5 Experiments

Our proposed TopK-EF with AMSGrad matches that of full AMSGrad, in distributed learning. Number of local workers is 20. Error feedback fixes the convergence issue of using solely the TopK gradient.

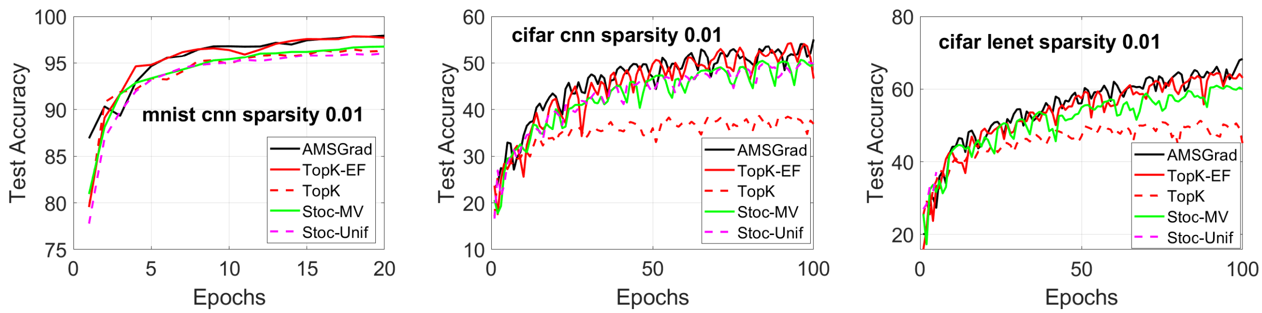

Figure 1: Test accuracy.

6 Conclusion

References

[1] Naman Agarwal, Ananda Theertha Suresh, Felix X. Yu, Sanjiv Kumar, and Brendan McMahan. cpsgd: Communication-efficient and differentially-private distributed SGD. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7575–7586, 2018.

[2] Ahmad Ajalloeian and Sebastian U Stich. Analysis of sgd with biased gradient estimators. *arXiv preprint arXiv:2008.00051*, 2020.

[3] Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. *arXiv preprint arXiv:1704.05021*, 2017.

[4] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pages 1709–1720, 2017.

[5] Dan Alistarh, Torsten Hoefler, Mikael Johansson, Sarit Khirirat, Nikola Konstantinov, and Cédric Renggli. The convergence of sparsified gradient methods. *arXiv preprint arXiv:1809.10505*, 2018.

[6] Debraj Basu, Deepesh Data, Can Karakus, and Suhas N. Diggavi. Qsparse-local-sgd: Distributed SGD with quantization, sparsification and local computations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14668–14679, 2019.

[7] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pages 560–569. PMLR, 2018.

[8] Jeremy Bernstein, Jiawei Zhao, Kamyar Azizzadenesheli, and Anima Anandkumar. signsgd with majority vote is communication efficient and fault tolerant. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[9] Aleksandr Beznosikov, Samuel Horváth, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *CoRR*, abs/2002.12410, 2020.

[10] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[11] Ken Chang, Niranjan Balachandar, Carson K. Lam, Darvin Yi, James M. Brown, Andrew Beers, Bruce R. Rosen, Daniel L. Rubin, and Jayashree Kalpathy-Cramer. Distributed deep learning networks among institutions for medical imaging. *J. Am. Medical Informatics Assoc.*, 25(8):945–954, 2018.

[12] Moses Charikar, Kevin C. Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *Automata, Languages and Programming, 29th International Colloquium, ICALP 2002, Malaga, Spain, July 8-13, 2002, Proceedings*, volume 2380 of *Lecture Notes in Computer Science*, pages 693–703. Springer, 2002.

[13] Congliang Chen, Li Shen, Haozhi Huang, Qi Wu, and Wei Liu. Quantized adam with error feedback. *arXiv preprint arXiv:2004.14180*, 2020.

[14] Trishul Chilimbi, Yutaka Suzue, Johnson Apacible, and Karthik Kalyanaraman. Project adam: Building an efficient and scalable deep learning training system. In *Symposium on Operating Systems Design and Implementation*, pages 571–582, 2014.

[15] Dami Choi, Christopher J. Shallue, Zachary Nado, Jaehoon Lee, Chris J. Maddison, and George E. Dahl. On empirical comparisons of optimizers for deep learning. *CoRR*, abs/1910.05446, 2019.

[16] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15-19, 2016*, pages 191–198. ACM, 2016.

[17] Christopher De Sa, Matthew Feldman, Christopher Ré, and Kunle Olukotun. Understanding and optimizing asynchronous low-precision stochastic gradient descent. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, pages 561–574, 2017.

[18] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew W. Senior, Paul A. Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1232–1240, 2012.

[19] Tim Dettmers. 8-bit approximations for parallelism in deep learning. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[20] John C Duchi, Alekh Agarwal, and Martin J Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

[21] John C. Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *COLT 2010 - The 23rd Conference on Learning Theory, Haifa, Israel, June 27-29, 2010*, pages 257–269, 2010.

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

[23] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[24] Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *CoRR*, abs/1706.02677, 2017.

[25] Alex Graves, Abdel-rahman Mohamed, and Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*, pages 6645–6649. IEEE, 2013.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[27] Mingyi Hong, Davood Hajinezhad, and Ming-Min Zhao. Prox-pda: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks. In *International Conference on Machine Learning*, pages 1529–1538, 2017.

[28] Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Vladimir Braverman, Ion Stoica, and Raman Arora. Communication-efficient distributed SGD with sketching. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13144–13154, 2019.

[29] Jiawei Jiang, Fangcheng Fu, Tong Yang, and Bin Cui. Sketchml: Accelerating distributed machine learning with data sketches. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, pages 1269–1284. ACM, 2018.

[30] Peng Jiang and Gagan Agrawal. A linear speedup analysis of distributed deep learning with sparse and quantized communication. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 2530–2541, 2018.

[31] Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian U Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. *arXiv preprint arXiv:1901.09847*, 2019.

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

[33] Anastasia Koloskova, Sebastian U Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning*, pages 3478–3487, 2019.

[34] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[35] Songtao Lu, Xinwei Zhang, Haoran Sun, and Mingyi Hong. Gnsd: A gradient-tracking based nonconvex stochastic algorithm for decentralized optimization. In *2019 IEEE Data Science Workshop (DSW)*, pages 315–321, 2019.

[36] Hiroaki Mikami, Hisahiro Suganuma, Yoshiki Tanaka, Yuichi Kageyama, et al. Massively distributed sgd: Imagenet/resnet-50 training in a flash. *arXiv preprint arXiv:1811.05233*, 2018.

[37] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013.

[38] Parvin Nazari, Davoud Ataee Tarzanagh, and George Michailidis. Dadam: A consensus-based distributed adaptive gradient method for online optimization. *arXiv preprint arXiv:1901.09109*, 2019.

[39] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48, 2009.

[40] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[41] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

[42] Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, pages 1058–1062. ISCA, 2014.

[43] Zebang Shen, Aryan Mokhtari, Tengfei Zhou, Peilin Zhao, and Hui Qian. Towards more efficient stochastic decentralized learning: Faster convergence and sparse communication. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4631–4640. PMLR, 2018.

[44] Shaohuai Shi, Kaiyong Zhao, Qiang Wang, Zhenheng Tang, and Xiaowen Chu. A convergence analysis of distributed sgd with communication-efficient gradient sparsification. In *IJCAI*, pages 3411–3417, 2019.

[45] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nat.*, 550(7676):354–359, 2017.

[46] Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified sgd with memory. In *Advances in Neural Information Processing Systems*, pages 4447–4458, 2018.

[47] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios D. Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.*, 2018:7068349:1–7068349:13, 2018.

[48] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. Gradient sparsification for communication-efficient distributed optimization. In *Advances in Neural Information Processing Systems*, pages 1299–1309, 2018.

[49] Jian Wei, Jianhua He, Kai Chen, Yi Zhou, and Zuoyin Tang. Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, 69:29–39, 2017.

[50] Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. *arXiv preprint arXiv:1705.07878*, 2017.

[51] Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized SGD and its applications to large-scale distributed optimization. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5321–5329. PMLR, 2018.

[52] Guandao Yang, Tianyi Zhang, Polina Kirichenko, Junwen Bai, Andrew Gordon Wilson, and Chris De Sa. Swalp: Stochastic weight averaging in low precision training. In *International Conference on Machine Learning*, pages 7015–7024. PMLR, 2019.

[53] Yang You, Jing Li, Sashank J. Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training BERT in 76 minutes. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[54] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing [review article]. *IEEE Comput. Intell. Mag.*, 13(3):55–75, 2018.

[55] Yue Yu, Jiaxiang Wu, and Junzhou Huang. Exploring fast and communication-efficient algorithms in large-scale distributed networks. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 674–683. PMLR, 2019.

[56] Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[57] Hantian Zhang, Jerry Li, Kaan Kara, Dan Alistarh, Ji Liu, and Ce Zhang. Zipml: Training linear models with end-to-end low precision, and a little bit of deep learning. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 4035–4043. PMLR, 2017.

[58] Lei Zhang, Shuai Wang, and Bing Liu. Deep learning for sentiment analysis: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 8(4), 2018.

[59] Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav Artzi. Revisiting few-sample BERT fine-tuning. *CoRR*, abs/2006.05987, 2020.

A Intermediary Lemmas

Lemma 1. *Under Assumption 3 and Assumption 4 we have for any iteration $t > 0$:*

$$\|m_t\|^2 \leq (q^2+1)G^2 \quad \textit{and} \quad \hat{v}_t \leq (q^2+1)G^2 \tag{5}$$

where m_t and $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$ are defined Line 15 of Algorithm 1 and $G^2 = \frac{1}{n}\sum_{i=1}^N G_i^2$.

Lemma 2. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize $\{\eta_t\}_{t>0}$, we have:*

$$-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \mid (\hat{V}_{t+1} + \epsilon \mathsf{I}_\mathsf{d})^{-1/2}\bar{g}_t \right\rangle] \leq -\frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2} \tag{6}$$

where I_d is the identity matrix, $\hat{V}_t$ the diagonal matrix which diagonal entries are $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$ defined Line 15 of Algorithm 1 and $\bar{g}_t$ is the aggregation of all ***quantized*** *gradients from the workers.*

Lemma 3. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize $\{\eta_t\}_{t>0}$, we have:*

$$\begin{aligned}
\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \leq &- \frac{\eta_{t+1}(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2} \\
&- \eta_{t+1}\beta_1\mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \mid (\hat{V}_t + \epsilon \mathsf{I}_\mathsf{d})^{-1/2} m_t \right\rangle] \\
&+ \left(\frac{L}{2} + \beta_1 L\right)\|\theta_t - \theta_{t-1}\|^2 \\
&+ \eta_{t+1}G^2\mathbb{E}[\sum_{j=1}^d \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]]
\end{aligned} \tag{7}$$

where d denotes the dimension of the parameter vector

B Proofs

B.1 Proof of Lemmas

Lemma. *Under Assumption 3 and Assumption 4 we have for any iteration $t > 0$:*

$$\|m_t\|^2 \leq (q^2+1)G^2 \quad \textit{and} \quad \hat{v}_t \leq (q^2+1)G^2 \tag{8}$$

where m_t and $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$ are defined Line 15 of Algorithm 1 and $G^2 = \frac{1}{n}\sum_{i=1}^N G_i^2$.

Proof. We start by writing

$$\|\bar{g}_t\|^2 = \left\|\frac{1}{n}\sum_{i=1}^N \tilde{g}_{t,i}\right\|^2 \leq \frac{1}{n}\sum_{i=1}^N \|\tilde{g}_{t,i}\|^2 \tag{9}$$

Though, using Assumption 3 and Assumption 4 we have:

$$\|\tilde{g}_{t,i}\|^2 = \|g_{t,i} + \tilde{g}_{t,i} - g_{t,i}\|^2 \leq \|g_{t,i}\|^2 + \|\tilde{g}_{t,i} - g_{t,i}\|^2 \leq (q^2+1)G_i^2 \tag{10}$$

Hence

$$\|\bar{g}_t\|^2 \leq (q^2+1)G^2 \tag{11}$$

where $G^2 = \frac{1}{n}\sum_{i=1}^N G_i^2$. Then, by construction in Algorithm 1:

$$\|m_t\|^2 \leq \beta_1^2\|m_{t-1}\|^2 + (1-\beta_1)^2\|\bar{g}_t\|^2 \leq \beta_1^2\|m_{t-1}\|^2 + (1-\beta_1)^2(q^2+1)G^2 \tag{12}$$

Since we have by initialization that $\|m_0\|^2 \leq G^2$, then we prove by induction that $\|m_t\|^2 \leq (q^2+1)G^2$.

Similarly

$$\hat{v}_t = \max(v_t, \hat{v}_{t-1}) = \max(\hat{v}_{t-1}, \beta_2 v_{t-1} + (1-\beta_2)\bar{g}_t^2) \leq max(\hat{v}_{t-1}, \beta_2 v_{t-1} + (1-\beta_2)(q^2+1)G^2) \tag{13}$$

□

Lemma. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize $\{\eta_t\}_{t>0}$, we have:*

$$-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_t \right\rangle] \leq -\frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2 \frac{G^2 \eta_{t+1}}{\epsilon 2 n^2} \tag{14}$$

where $\mathsf{I_d}$ is the identity matrix, $\hat{V}_t$ the diagonal matrix which diagonal entries are $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$ defined Line 15 of Algorithm 1 and $\bar{g}_t$ is the aggregation of all ***quantized*** *gradients from the workers.*

Proof. We first decompose $\bar{g}_t$ as the sum of the unbiased stochastic gradients and its quantized versions as computed Line 7 of Algorithm 1:

$$\bar{g}_t = \frac{1}{n}\sum_{i=1}^{N} \tilde{g}_{t,i} = \frac{1}{n}\sum_{i=1}^{N} [g_{t,i} + \tilde{g}_{t,i} - g_{t,i}] \tag{15}$$

Hence,

$$\begin{aligned} T_1 :=& -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_t \right\rangle] \\ =& \underbrace{-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \frac{1}{n}\sum_{i=1}^{N} g_{t,i} \right\rangle]}_{t1} \underbrace{-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \frac{1}{n}\sum_{i=1}^{N} \tilde{g}_{t,i} - g_{t,i} \right\rangle]}_{t_2} \end{aligned} \tag{16}$$

Bounding t_1: Using the Tower rule, we have:

$$\begin{aligned} t_1 :=& -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \frac{1}{n}\sum_{i=1}^{N} g_{t,i} \right\rangle] \\ =& -\eta_{t+1}\mathbb{E}[\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \frac{1}{n}\sum_{i=1}^{N} g_{t,i} \right\rangle |\mathcal{F}_t]] \\ =& -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \mathbb{E}[\frac{1}{n}\sum_{i=1}^{N} g_{t,i} |\mathcal{F}_t] \right\rangle] \end{aligned} \tag{17}$$

Using Assumption 3 and Lemma 1, we have that

$$\begin{aligned} t_1 :=& -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \frac{1}{n}\sum_{i=1}^{N} g_{t,i} \right\rangle] \\ \leq& -\eta_{t+1}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \end{aligned} \tag{18}$$

Bounding t_2:

We first recall Young's inequality with a constant $\delta \in (0,1)$ as follows:

$$\langle X \,|\, Y \rangle \leq \frac{1}{\delta}\|X\|^2 + \delta \|Y\|^2 \, . \tag{19}$$

Using Young's inequality (19) with parameter equal to 1:

$$
\begin{aligned}
t_2 \leq & \frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + \frac{\eta_{t+1}}{2n^2}\mathbb{E}[\|(\hat{V}_{t+1}+\epsilon \mathsf{I_d})^{-1/2}\sum_{i=1}^{N}\{\tilde{g}_{t,i} - g_{t,i}\}\|^2] \\
\overset{(a)}{\leq} & \frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + \frac{\eta_{t+1}}{2n^2}\mathbb{E}[\|(\hat{V}_{t+1}+\epsilon \mathsf{I_d})^{-1/2}\|^2\sum_{i=1}^{N}\{\tilde{g}_{t,i} - g_{t,i}\}\|^2] \\
\overset{(b)}{\leq} & \frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + \frac{\eta_{t+1}}{2n^2}\mathbb{E}[\|(\hat{V}_{t+1}+\epsilon \mathsf{I_d})^{-1/2}\|^2]\mathbb{E}[\|\sum_{i=1}^{N}\{\tilde{g}_{t,i} - g_{t,i}\}\|^2] \\
\overset{(c)}{\leq} & \frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + \frac{\eta_{t+1}}{\epsilon 2n^2}\mathbb{E}[\|\sum_{i=1}^{N}\tilde{g}_{t,i} - g_{t,i}\|^2] \\
\overset{(d)}{\leq} & \frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2}
\end{aligned}
\tag{20}
$$

where (a) uses the Cauchy-Schwartz inequality, (b) is due to the non-negativeness of both $\hat{V}_{t+1}$ and $\|\sum_{i=1}^{N}\{g_{t,i} + \tilde{g}_{t,i} - g_{t,i}\}\|^2$ and (c) uses the Triangle inequality. We use Assumption 1 and Assumption 4 in (d).

Finally, combining (18) and (20) yields

$$
-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t)\,|\,(\hat{V}_{t+1}+\epsilon \mathsf{I_d})^{-1/2}\bar{g}_t\right\rangle] \leq -\frac{\eta_{t+1}}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2}
\tag{21}
$$

□

Lemma. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize* $\{\eta_t\}_{t>0}$, *we have:*

$$
\begin{aligned}
\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \leq & -\frac{\eta_{t+1}(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2} \\
& - \eta_{t+1}\beta_1\mathbb{E}[\left\langle \nabla f(\theta_{t-1})\,|\,(\hat{V}_t+\epsilon \mathsf{I_d})^{-1/2}m_t\right\rangle] \\
& + \left(\frac{L}{2}+\beta_1 L\right)\|\theta_t - \theta_{t-1}\|^2 \\
& + \eta_{t+1}G^2\mathbb{E}[\sum_{j=1}^{d}\left[(\hat{v}_{t+1}^j+\epsilon)^{-1/2} - (\hat{v}_t^j+\epsilon)^{-1/2}\right]]
\end{aligned}
\tag{22}
$$

where d *denotes the dimension of the parameter vector*

Proof. By assumption Assumption 2, we can write the smoothness condition on the overall objective (2), between iteration t and $t+1$:

$$
f(\theta_{t+1}) \leq f(\theta_t) + \langle \nabla f(\theta_t)\,|\,\theta_{t+1}-\theta_t\rangle + \frac{L}{2}\|\theta_{t+1}-\theta_t\|^2
\tag{23}
$$

Denote by $\hat{V}_t$ the diagonal matrix which diagonal entries are $\hat{v}_t = \max(v_t, \hat{v}_{t-1})$ defined Line 15 of Algorithm 1. Hence, we obtain,

$$
f(\theta_{t+1}) \leq f(\theta_t) - \eta_{t+1}\left\langle \nabla f(\theta_t)\,|\,(\hat{V}_{t+1}+\epsilon \mathsf{I_d})^{-1/2}m_{t+1}\right\rangle + \frac{L}{2}\|\theta_{t+1}-\theta_t\|^2
\tag{24}
$$

where $\mathsf{I_d}$ denotes the identity matrix.

We now take the expectation of those various terms conditioned on the filtration $\mathcal{F}_t$ of the total randomness up to iteration t.

$$\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \leq -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] + \frac{L}{2}\mathbb{E}[\|\theta_{t+1} - \theta_t\|^2] \tag{25}$$

We now focus on the computation of the inner product obtained in the equation above. We have

$$\begin{aligned}
&\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] \qquad (26)\\
=&\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} + (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} - (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle]\\
=&\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] + \eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, \left[(\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} - (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2}\right] m_{t+1} \right\rangle]\\
=&\eta_{t+1}\beta_1\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle] + \eta_{t+1}(1-\beta_1)\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_{t} \right\rangle]\\
&+ \eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, \left[(\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} - (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2}\right] m_{t+1} \right\rangle] \qquad (27)
\end{aligned}$$

where $\bar{g}_t$ is the aggregated gradients from all workers.

Plugging the above in (25) yields:

$$\begin{aligned}
&\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)]\\
\leq &\underbrace{-\beta_1\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle]}_{A_t} \eta_{t+1}\\
&\underbrace{-\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, \left[(\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} - (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2}\right] m_{t+1} \right\rangle]}_{B_t} \eta_{t+1} \qquad (28)\\
&\underbrace{-(1-\beta_1)\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_{t} \right\rangle]}_{C_t} \eta_{t+1} + \frac{L}{2}\mathbb{E}[\|\theta_{t+1} - \theta_t\|^2]
\end{aligned}$$

To begin with, by the tower rule, we have that

$$\begin{aligned}
A_t &= -\beta_1\mathbb{E}[\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle |\mathcal{F}_t]] \qquad (29)\\
&= -\beta_1 \left\langle \nabla f(\theta_{t-1}) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle] - \beta_1 \left\langle \nabla f(\theta_t) - \nabla f(\theta_{t-1}) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle] \qquad (30)\\
& \qquad (31)
\end{aligned}$$

where we recognize the first term as the term in (26), at iteration $t-1$ and hence apply the same decomposition as in (27). Coupling with the smoothness of f, which gives that

$$-\beta_1 \left\langle \nabla f(\theta_t) - \nabla f(\theta_{t-1}) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle] \leq \frac{\beta_1 L}{\eta_{t-1}} \|\theta_t - \theta_{t-1}\|^2$$

we obtain,

$$\begin{aligned}
A_t &= -\beta_1\mathbb{E}[\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t} + \epsilon \mathsf{I_d})^{-1/2} m_{t} \right\rangle |\mathcal{F}_t]]\\
&\leq \eta_{t+1}\beta_1(A_{t-1} + B_{t-1} + C_{t-1}) + \eta_{t+1}\frac{\beta_1 L}{\eta_{t-1}} \|\theta_t - \theta_{t-1}\|^2
\end{aligned} \tag{32}$$

Then,

$$
\begin{aligned}
B_t &= -\mathbb{E}[\left\langle \nabla f(\theta_t) \mid \left[(\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} - (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2}\right] m_{t+1} \right\rangle] \\
&= \mathbb{E}[\sum_{j=1}^{d} \nabla^j f(\theta_t) m_{t=1}^j \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]] \\
&\overset{(a)}{\leq} \mathbb{E}[\|\nabla f(\theta_t)\| \, \|m_{t=1}\| \sum_{j=1}^{d} \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]] \\
&\overset{(b)}{\leq} G^2 \mathbb{E}[\sum_{j=1}^{d} \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]]
\end{aligned}
\tag{33}
$$

where $\nabla^j f(\theta_t)$ denotes the j-th component of the gradient vector $\nabla f(\theta_t)$, (a) uses of the Cauchy-Schwartz inequality and (b) boils down from the norm of the gradient vector boundedness assumption 3, denoting $G := \frac{1}{n}\sum_{i=1}^{n} G_i$.

Plugging the above into (28) yields

$$
\begin{aligned}
\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \leq & \eta_{t+1}(A_t + B_t + C_t) + \frac{L}{2}\mathbb{E}[\|\theta_{t+1} - \theta_t\|^2] \\
\leq & - \eta_{t+1}\beta_1 \mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \mid (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle] \\
& + \eta_{t+1} G^2 \mathbb{E}[\sum_{j=1}^{d} \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]] \\
& + \left(\frac{L}{2} + \eta_{t+1}\frac{\beta_1 L}{\eta_{t-1}}\right) \|\theta_t - \theta_{t-1}\|^2 \\
& - \eta_{t+1}(1-\beta_1)\mathbb{E}[\left\langle \nabla f(\theta_t) \mid (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_t \right\rangle]
\end{aligned}
\tag{34}
$$

We bound the last term on the RHS, $-\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \mid (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} \bar{g}_t \right\rangle]$ with Lemma 2

Under the assumption that we use a decreasing stepsize such that $\eta_{t+1} \leq \eta_t$, and given that according to Line 15 we have that $\hat{v}_{t+1} \geq \hat{v}_t$ by construction, we obtain

$$
\begin{aligned}
\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \leq & - \frac{\eta_{t+1}(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2 \frac{G^2 \eta_{t+1}}{\epsilon 2 n^2} \\
& - \eta_{t+1}\beta_1 \mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \mid (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle] \\
& + \left(\frac{L}{2} + \beta_1 L\right) \|\theta_t - \theta_{t-1}\|^2 \\
& + \eta_{t+1} G^2 \mathbb{E}[\sum_{j=1}^{d} \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]]
\end{aligned}
\tag{35}
$$

Finally, using Lemma 2, we obtain the desired result. □

B.2 Proof of Theorem 1

Theorem. *Under Assumption 2 to Assumption 4, with a constant stepsize $\eta_t = \eta = \frac{L}{\sqrt{T}}$, we have:*

$$
\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \frac{\mathbb{E}[f(\theta_0) - f(\theta_T)]}{L\Delta_1 \sqrt{T}} + d\frac{L\Delta_3}{\Delta_1 \sqrt{T}} + \frac{\Delta_2}{\eta \Delta_1 T} + \frac{1-\beta_1}{\Delta_1}\epsilon^{-\frac{1}{2}}\sqrt{(q^2+1)}G^2
\tag{36}
$$

where

$$\Delta_1 := \frac{(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \quad , \quad \Delta_2 := q^2 + \sum_{k=t+1}^{\infty} \beta_1^{k-t+2} \frac{G^2}{\epsilon 2n^2}$$
$$\Delta_3 := \left(\frac{L}{2} + 1 + \frac{\beta_1 L}{1-\beta_1}\right)(1-\beta_2)^{-1}(1-\frac{\beta_1^2}{\beta_2})^{-1} \tag{37}$$

Proof. By Lemma 3 we have

$$\begin{aligned}\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \le & -\frac{\eta_{t+1}(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2\frac{G^2\eta_{t+1}}{\epsilon 2n^2} \\ & - \eta_{t+1}\beta_1\mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \,|\, (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle] \\ & + \left(\frac{L}{2} + \beta_1 L\right)\|\theta_t - \theta_{t-1}\|^2 \\ & + \eta_{t+1}G^2\mathbb{E}[\sum_{j=1}^{d}\left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2}\right]]\end{aligned} \tag{38}$$

Let us consider the following sequence, defined for all $t > 0$:

$$R_t := f(\theta_t) - \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1}\mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \,|\, (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle] \tag{39}$$

We compute the following expectation:

$$\begin{aligned}\mathbb{E}[R_{t+1}] - \mathbb{E}[R_t] = & \mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] - \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] \\ & + \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1}\mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \,|\, (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle]\end{aligned} \tag{40}$$

Using the Assumption 2, we note that:

$$\mathbb{E}[f(\theta_{t+1}) - f(\theta_t)] \le -\eta_{t+1}\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] + \frac{L}{2}\|\theta_{t+1} - \theta_t\|^2 \tag{41}$$

which yields

$$\begin{aligned}\mathbb{E}[R_{t+1}] - \mathbb{E}[R_t] = & -(\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2})\mathbb{E}[\left\langle \nabla f(\theta_t) \,|\, (\hat{V}_{t+1} + \epsilon \mathsf{I_d})^{-1/2} m_{t+1} \right\rangle] \\ & + \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1}\mathbb{E}[\left\langle \nabla f(\theta_{t-1}) \,|\, (\hat{V}_t + \epsilon \mathsf{I_d})^{-1/2} m_t \right\rangle] \\ & + \frac{L}{2}\|\theta_{t+1} - \theta_t\|^2 \\ \le & (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2})\mathbb{E}[A_t + B_t + C_t] \\ & - \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1}\mathbb{E}[A_{t-1} + B_{t-1} + C_{t-1}] \\ & + \frac{L}{2}\|\theta_{t+1} - \theta_t\|^2\end{aligned} \tag{42}$$

where A_t, B_t, C_t are defined in (28).

We use (32) and (33) to bound A_t and B_t, and Lemma 2 to bound C_t where we precise that the learning rate η_{t+1} becomes $\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}$. Hence

$$
\begin{aligned}
\mathbb{E}[R_{t+1}] - \mathbb{E}[R_t] \leq & \left((\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2})\beta_1 - \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1} \right) \mathbb{E}[A_{t-1} + B_{t-1} + C_{t-1}] \\
& + (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}) G^2 \mathbb{E}[\sum_{j=1}^{d} \left[(\hat{v}_{t+1}^j + \epsilon)^{-1/2} - (\hat{v}_t^j + \epsilon)^{-1/2} \right]] \\
& + \left(\frac{L}{2} + (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}) \frac{\beta_1 L}{\eta_{t-1}} \right) \|\theta_{t+1} - \theta_t\|^2 \\
& - (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}) \frac{(1-\beta_1)}{2} (\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
& + q^2 \eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2} \frac{G^2}{\epsilon 2 n^2}
\end{aligned} \tag{43}
$$

where the last term in the LHS is due to Lemma 3.

By assumption, we have that for all $t > 0$, $\eta_{t=1} \leq \eta_t$. Also, set the tuning parameters such that

$$
\eta_t + \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1} \leq \frac{\eta_t}{1-\beta_1} \tag{44}
$$

so that

$$
\begin{aligned}
& (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2})\beta_1 - \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1} = 0 \\
\iff & (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2})\beta_1 = \sum_{k=t}^{\infty} \eta_k \beta_1^{k-t+1}
\end{aligned} \tag{45}
$$

Note that $-(\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}) \frac{(1-\beta_1)}{2} (\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \leq -\eta_{t+1} \frac{(1-\beta_1)}{2} (\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}$ since $\sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2} \geq 0$.

The above coupled with (43) yields

$$
\begin{aligned}
\mathbb{E}[R_{t+1}] - \mathbb{E}[R_t] \leq & - \eta_{t+1} \frac{(1-\beta_1)}{2} (\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] + q^2 \eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2} \frac{G^2}{\epsilon 2 n^2} \\
& - (\eta_{t+1} + \sum_{k=t+1}^{\infty} \eta_k \beta_1^{k-t+2}) G^2 \mathbb{E}[\sum_{j=1}^{d} \left[(\hat{v}_t^j + \epsilon)^{-1/2} - (\hat{v}_{t+1}^j + \epsilon)^{-1/2} \right]] \\
& + \left(\frac{L}{2} + 1 + \frac{\beta_1 L}{1-\beta_1} \right) \mathbb{E}[\|\theta_{t+1} - \theta_t\|^2]
\end{aligned} \tag{46}
$$

We now sum from $t = 0$ to $t = T - 1$ the inequality in (46), and divide it by T:

$$
\begin{aligned}
&\eta\frac{(1-\beta_1)}{2}(\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
\leq &\frac{\mathbb{E}[R_0]-\mathbb{E}[R_T]}{T} + \frac{q^2\eta + \sum_{k=t+1}^{\infty}\eta\beta_1^{k-t+2}\frac{G^2}{\epsilon 2n^2}}{T} \\
&+\left(\frac{L}{2}+1+\frac{\beta_1 L}{1-\beta_1}\right)\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\theta_{t+1}-\theta_t\|^2]
\end{aligned}
\tag{47}
$$

where we have used the fact that $(\hat{v}_t^j + \epsilon)^{-1/2} - (\hat{v}_{t+1}^j + \epsilon)^{-1/2} \geq 0$ for all dimension $j \in [d]$ by construction of $\hat{v}_{t+1}^j$.

We now bound the two remaining terms:

Bounding $-\mathbb{E}[R_T]$:

By definition (39) of R_t we have, using Lemma 1:

$$
\begin{aligned}
-\mathbb{E}[R_T] &\leq \sum_{k=t}^{\infty}\eta_k\beta_1^{k-t+1}\mathbb{E}[\left\langle\nabla f(\theta_{t-1})\,|\,(\hat{V}_t+\epsilon\mathsf{I}_\mathsf{d})^{-1/2}m_t\right\rangle] - f(\theta_T) \\
&\leq \|\sum_{k=t}^{\infty}\eta_k\beta_1^{k-t+1}\|\|\nabla f(\theta_{t-1})\|\|(\hat{V}_t+\epsilon\mathsf{I}_\mathsf{d})^{-1/2}m_t\| \\
&\leq \eta_{t+1}(1-\beta_1)\epsilon^{-\frac{1}{2}}\sqrt{(q^2+1)}G^2 - f(\theta_T)
\end{aligned}
\tag{48}
$$

Bounding $\sum_{t=0}^{T-1}\mathbb{E}[\|\theta_{t+1}-\theta_t\|^2]$:

By definition in Algorithm 1:

$$
\|\theta_{t+1}-\theta_t\|^2 = \eta_{t+1}^2\left[(\hat{V}_{t+1}+\epsilon\mathsf{I}_\mathsf{d})^{-\frac{1}{2}}m_{t+1}\right]^2 = \eta_{t+1}^2\sum_{j=1}^{d}\frac{|m_{t+1}^j|^2}{\hat{v}_{t+1}^j+\epsilon}
\tag{49}
$$

For any dimension $j \in [d]$,

$$
\begin{aligned}
|m_{t+1}^j|^2 &= |\beta_1 m_t^j + (1-\beta_1)\bar{g}_t^j|^2 \\
&\leq \beta_1(\beta_1^2|m_{t-1}^j|^2 + (1-\beta_1)^2|\bar{g}_{t-1}^j|^2) + |\bar{g}_t^j|^2 \\
&\leq \sum_{k=0}^{t}\beta_1^{2(t-k)}|\bar{g}_k^j|^2 \\
&\leq \sum_{k=0}^{t}\frac{\beta_1^{2(t-k)}}{\beta_2^{t-k}}\beta_2^{t-k}|\bar{g}_k^j|^2
\end{aligned}
\tag{50}
$$

Using Cauchy-Schwartz inequality we obtain

$$
\begin{aligned}
|m_{t+1}^j|^2 \leq \sum_{k=0}^{t}\frac{\beta_1^{2(t-k)}}{\beta_2^{t-k}}\beta_2^{t-k}|\bar{g}_k^j|^2 &\leq \sum_{k=0}^{t}\left(\frac{\beta_1^2}{\beta_2}\right)^{t-k}\sum_{k=0}^{t}\beta_2^{t-k}|\bar{g}_k^j|^2 \\
&\leq \frac{1}{1-\frac{\beta_1^2}{\beta_2}}\sum_{k=0}^{t}\beta_2^{t-k}|\bar{g}_k^j|^2
\end{aligned}
\tag{51}
$$

On the other hand we have

$$
\hat{v}_{t+1}^j \geq \beta_2\hat{v}_t^j + (1-\beta_2)(\bar{g}_t^j)^2
\tag{52}
$$

and since it is also true for iteration $t = 1$, we have by induction replacing v_t^j in the above that

$$\hat{v}_{t+1}^j \geq (1-\beta_2) \sum_{k=0}^{t} \beta_2^{t-k} |\bar{g}_k^j|^2 \iff \frac{\sum_{k=0}^{t} \beta_2^{t-k} |\bar{g}_k^j|^2}{\hat{v}_{t+1}^j} \leq (1-\beta_2)^{-1} \tag{53}$$

Hence, we can derive from (49) that

$$\begin{aligned}
\|\theta_{t+1} - \theta_t\|^2 = \eta_{t+1}^2 \sum_{j=1}^{d} \frac{|m_{t+1}^j|^2}{\hat{v}_{t+1}^j + \epsilon} &\leq \eta_{t+1}^2 \sum_{j=1}^{d} \frac{|m_{t+1}^j|^2}{\hat{v}_{t+1}^j} \\
&\overset{(a)}{\leq} \eta_{t+1}^2 \sum_{j=1}^{d} \frac{1}{1 - \frac{\beta_1^2}{\beta_2}} \frac{\sum_{k=0}^{t} \beta_2^{t-k} |\bar{g}_k^j|^2}{\hat{v}_{t+1}^j} \\
&\overset{(b)}{\leq} \eta_{t+1}^2 d (1-\beta_2)^{-1} (1 - \frac{\beta_1^2}{\beta_2})^{-1}
\end{aligned} \tag{54}$$

where (a) uses (51) and (b) uses (53).

Plugging the two bounds in (47), we obtain the following bound:

$$\begin{aligned}
\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq & \frac{\mathbb{E}[f(\theta_0) - f(\theta_T)]}{\eta \Delta_1 T} + \frac{q^2 \eta + \sum_{k=t+1}^{\infty} \eta \beta_1^{k-t+2} \frac{G^2}{\epsilon 2 n^2}}{\eta \Delta_1 T} \\
&+ \frac{1-\beta_1}{\Delta_1} \epsilon^{-\frac{1}{2}} \sqrt{(q^2+1)} G^2 \\
&+ \left(\frac{L}{2} + 1 + \frac{\beta_1 L}{1-\beta_1} \right) \frac{1}{\eta \Delta_1} \eta^2 d (1-\beta_2)^{-1} (1 - \frac{\beta_1^2}{\beta_2})^{-1}
\end{aligned} \tag{55}$$

where $\Delta_1 := \frac{(1-\beta_1)}{2} (\epsilon + \frac{(q^2+1)G^2}{1-\beta_2})^{-\frac{1}{2}}$

With a constant stepsize $\eta = \frac{L}{\sqrt{T}}$ we get the final convergence bound as follows:

$$\begin{aligned}
\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq & \frac{\mathbb{E}[f(\theta_0) - f(\theta_T)]}{L \Delta_1 \sqrt{T}} + d \frac{L \Delta_3}{\Delta_1 \sqrt{T}} \\
&+ \frac{\Delta_2}{\Delta_1 T} + \frac{1-\beta_1}{\Delta_1} \epsilon^{-\frac{1}{2}} \sqrt{(q^2+1)} G^2
\end{aligned} \tag{56}$$

where $\Delta_2 := q^2 + \sum_{k=t+1}^{\infty} \beta_1^{k-t+2} \frac{G^2}{\epsilon 2 n^2}$ and $\Delta_3 := \left(\frac{L}{2} + 1 + \frac{\beta_1 L}{1-\beta_1} \right) (1-\beta_2)^{-1} (1 - \frac{\beta_1^2}{\beta_2})^{-1}$.

□

B.3 Proof of Theorem 2

Theorem. *Under Assumption 2 to Assumption 4, with a decreasing sequence of stepsize* $\{\eta_t\}_{t>0} = \frac{1}{\sqrt{T}}$*, the sequence of iterates* $\{\theta_t\}_{t>0}$ *output from Algorithm 2 satisfies:*

$$\frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \mathcal{O}(\frac{1}{\sqrt{T}} + \frac{1}{T}),$$

Proof. Let m_t' be the first moment moving average of standard AMSGrad using full gradients, *i.e.,* the gradient with respect to the index data point i_t computed Line 4 of Algorithm 2 before applying any compression operator.

Denote

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\tilde{g}_t \quad \text{and} \quad m'_t = \beta_1 m'_{t-1} + (1-\beta_1) g_t$$
$$a_t = \frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}, \quad \text{and} \quad a'_t = \frac{m'_t}{\sqrt{\hat{v}'_t + \epsilon}}.$$

By construction we have $m'_t = (1-\beta_1)\sum_{i=1}^{k} \beta_1^{t-i} g_t$.

Denote the following quantities

$$\mathcal{E}_{t+1} := \frac{(1-\beta_1)\sum_{i=1}^{t+1} \beta_1^{t+1-i} e_i}{\sqrt{\hat{v}_t + \epsilon}}$$
$$\theta'_{t+1} := \theta_{t+1} - \eta \mathcal{E}_{t+1}.$$

Then,

$$\begin{aligned}
\theta'_{t+1} &= \theta_{t+1} - \eta \mathcal{E}_{t+1} \\
&= \theta_t - \eta \frac{(1-\beta_1)\sum_{i=1}^{t} \beta_1^{t-i} \tilde{g}_i + (1-\beta_1)\sum_{i=1}^{t+1} \beta_1^{t+1-i} e_i}{\sqrt{\hat{v}_t + \epsilon}} \\
&= \theta_t - \eta \frac{(1-\beta_1)\sum_{i=1}^{t} \beta_1^{t-i} (\tilde{g}_i + e_{i+1}) + (1-\beta)\beta_1^t e_1}{\sqrt{\hat{v}_t + \epsilon}} \\
&= \theta_t - \eta \frac{(1-\beta_1)\sum_{i=1}^{t} \beta_1^{t-i} e_i}{\sqrt{\hat{v}_t + \epsilon}} - \eta \frac{m'_t}{\sqrt{\hat{v}_t + \epsilon}} \\
&\overset{(a)}{=} \theta'_t - \eta \frac{m'_t}{\sqrt{\hat{v}_t + \epsilon}} := \theta'_t - \eta a'_t,
\end{aligned}$$

where (a) uses the fact that $\tilde{g}_t + e_{t+1} = g_t + e_t$, $e_1 = 0$ at initialization. By smoothness assumption Assumption 2 we have

$$f(\theta'_{t+1}) \leq f(\theta'_t) - \eta \langle \nabla f(\theta'_t), a'_t \rangle + \frac{L}{2} \|\theta'_{t+1} - \theta'_t\|^2.$$

Thus,

$$\begin{aligned}
\mathbb{E}[f(\theta'_{t+1}) - f(\theta'_t)] &\leq -\eta \mathbb{E}[\langle \nabla f(\theta'_t), a'_t \rangle] + \frac{\eta^2 L}{2} \mathbb{E}[\|a'_t\|^2] &(57) \\
&= -\eta \mathbb{E}[\langle \nabla f(\theta_t), a'_t \rangle] + \frac{\eta^2 L}{2} \mathbb{E}[\|a'_t\|^2] + \eta \mathbb{E}[\langle \nabla f(\theta_t) - \nabla f(\theta'_t), a'_t \rangle] &(58) \\
&\leq -\eta \mathbb{E}[\langle \nabla f(\theta_t), a'_t \rangle] + \frac{\eta^2 L}{2} \mathbb{E}[\|a'_t\|^2] + \eta^2 L \mathbb{E}[\|\mathcal{E}_t\| \|a'_t\|] &(59)
\end{aligned}$$

Bounding the first term (extracting ∇f). We have

$$\begin{aligned}
M_t := -\mathbb{E}[\langle \nabla f(\theta_t), a'_t \rangle] &= -\mathbb{E}[\langle \nabla f(\theta_t), \frac{m'_t}{\sqrt{\hat{v}_t + \epsilon}} \rangle] \\
&= \underbrace{-\mathbb{E}[\langle \nabla f(\theta_t), \frac{m'_t}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle]}_{I} + \underbrace{\mathbb{E}[\langle \nabla f(\theta_t), (\frac{1}{\sqrt{\hat{v}_{t-1} + \epsilon}} - \frac{1}{\sqrt{\hat{v}_t + \epsilon}}) m'_t \rangle]}_{II}.
\end{aligned}$$

To bound I, note that

$$\begin{aligned}
I &= -\mathbb{E}[\langle \nabla f(\theta_t), \frac{(1-\beta_1) g_t}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle] - \mathbb{E}[\langle \nabla f(\theta_t), \frac{\beta_1 m'_{t-1}}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle] \\
&= -\mathbb{E}\mathbb{E}[\langle \nabla f(\theta_t), \frac{(1-\beta_1) g_t}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle | \mathcal{F}_{t-1}] - \mathbb{E}[\langle \nabla f(\theta_t), \frac{\beta_1 m'_{t-1}}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle] \\
&= -(1-\beta_1) \mathbb{E}[\frac{\|\nabla f(\theta_t)\|^2}{\sqrt{\hat{v}_{t-1} + \epsilon}}] - \mathbb{E}[\langle \nabla f(\theta_t), \frac{\beta_1 m'_{t-1}}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle] \\
&\leq -\frac{1-\beta_1}{\sqrt{(q^2+1)G^2 + \epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] - \beta_1 \mathbb{E}[\langle \nabla f(\theta_t), \frac{m'_{t-1}}{\sqrt{\hat{v}_{t-1} + \epsilon}} \rangle].
\end{aligned}$$

Regarding the second term, we have

$$\begin{aligned}
-\mathbb{E}[\langle \nabla f(\theta_t), \frac{m'_{t-1}}{\sqrt{\hat{v}_{t-1}+\epsilon}} \rangle] &= -\mathbb{E}[\langle \nabla f(\theta_{t-1}), \frac{m'_{t-1}}{\sqrt{\hat{v}_{t-1}+\epsilon}} \rangle] - \mathbb{E}[\langle \nabla f(\theta_t) - \nabla f(\theta_{t-1}), \frac{m'_{t-1}}{\sqrt{\hat{v}_{t-1}+\epsilon}} \rangle] \\
&= M_{t-1} + \eta L \mathbb{E}[\| \frac{m_{t-1}}{\sqrt{\hat{v}_{t-1}+\epsilon}} \| \| \frac{m'_{t-1}}{\sqrt{\hat{v}_{t-1}+\epsilon}} \|] \\
&\le M_{t-1} + \frac{\eta L (q^2+1) G^4}{\epsilon}.
\end{aligned}$$

Putting parts together we obtain

$$I \le \beta_1 M_{t-1} + \frac{\eta \beta_1 L (q^2+1) G^4}{\epsilon} - \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2].$$

For II, it holds that

$$II \le G^2 \mathbb{E}[\sum_{i=1}^d | \frac{1}{\sqrt{\hat{v}_{t-1}+\epsilon}} - \frac{1}{\sqrt{\hat{v}_t+\epsilon}} |].$$

Thus, we arrive at

$$\begin{aligned}
M_t &\le \beta_1 M_{t-1} + \frac{\eta \beta_1 L (q^2+1) G^4}{\epsilon} + G^2 \mathbb{E}[\sum_{i=1}^d | \frac{1}{\sqrt{\hat{v}_{t-1}+\epsilon}} - \frac{1}{\sqrt{\hat{v}_t+\epsilon}} |] - \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
&:= \beta_1 M_{t-1} + \frac{\eta \beta_1 L (q^2+1) G^4}{\epsilon} + G^2 H_t - \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
&\le \beta_1 M_{t-1} + \frac{\eta \beta_1 L (q^2+1) G^4}{\epsilon} + G^2 H_t.
\end{aligned}$$

By induction, we have

$$M_t \le \beta_1^{t-1} M_1 + G^2 \sum_{i=0}^{t-2} \beta_1^i H_{t-i} + \frac{\eta \beta_1 L (q^2+1) G^4}{(1-\beta_1)\epsilon} - \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2],$$

since $\beta_1 < 1$. Summing over $t = 1, ..., T$, we obtain

$$\begin{aligned}
\sum_{t=1}^T M_t &\le \sum_{t=1}^T \beta_1^{t-1} M_1 + G^2 \sum_{t=2}^T \sum_{i=0}^{t-2} \beta_1^i H_{t-i} + \frac{T \eta \beta_1 L (q^2+1) G^4}{(1-\beta_1)\epsilon} - \sum_{t=1}^T \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
&\overset{(a)}{\le} \frac{dG^2}{(1-\beta_1)\sqrt{\epsilon}} + G^2 \sum_{t=2}^T (\sum_{i=0}^{T-t} \beta_1^{t-i}) H_t + \frac{T \eta \beta_1 L (q^2+1) G^4}{(1-\beta_1)\epsilon} - \sum_{t=1}^T \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
&\le \frac{dG^2}{(1-\beta_1)\sqrt{\epsilon}} + \frac{G^2}{1-\beta_1} \sum_{t=2}^T \mathbb{E}[\sum_{i=1}^d | \frac{1}{\sqrt{\hat{v}_{t-1}+\epsilon}} - \frac{1}{\sqrt{\hat{v}_t+\epsilon}} | \\
&\qquad + \frac{T \eta \beta_1 L (q^2+1) G^4}{(1-\beta_1)\epsilon} - \sum_{t=1}^T \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2] \\
&\overset{(b)}{\le} \frac{2dG^2}{(1-\beta_1)\sqrt{\epsilon}} + \frac{T \eta \beta_1 L (q^2+1) G^4}{(1-\beta_1)\epsilon} - \sum_{t=1}^T \frac{1-\beta_1}{\sqrt{(q^2+1)G^2+\epsilon}} \mathbb{E}[\|\nabla f(\theta_t)\|^2],
\end{aligned}$$

where (a) is because $M_1 = \mathbb{E}[\langle \nabla f(\theta_1), a'_0 \rangle] \le \beta_1 dG^2/\sqrt{\epsilon}$, and (b) is derived by cancelling terms due to the fact that $\{\hat{v}_t\}_{t>0}$ is a non-decreasing sequence, hence $\hat{v}_t \le \hat{v}_{t-1}$. It remains to bound the last two terms in (59).

Bounding the variance term. We have

$$\mathbb{E}[\|a'_t\|^2] = \mathbb{E}[\| \frac{m'_t}{\sqrt{\hat{v}_t+\epsilon}} \|^2] \le \frac{1}{\epsilon} \mathbb{E}[\|m'_t\|^2],$$

and by Young's inequality,

$$\begin{aligned}\mathbb{E}[\|m'_t\|^2] &= \mathbb{E}[\|\beta_1 m'_{t-1} + (1-\beta_1)g_t\|^2] \\ &\leq (1+\frac{\rho}{2})\beta_1^2\mathbb{E}[\|m'_{t-1}\|^2] + (1+\frac{1}{2\rho})(1-\beta_1)^2\mathbb{E}[\|g_t\|^2].\end{aligned}$$

Choosing $\rho = 2(1-\beta_1^2)$, we derive

$$\begin{aligned}\mathbb{E}[\|m'_t\|^2] &\leq \beta_1^2(2-\beta_1^2)\mathbb{E}[\|m'_{t-1}\|^2] + (1-\beta_1)^2(1+\frac{1}{4(1-\beta_1^2)})\mathbb{E}[\|g_t\|^2] \\ &\leq \frac{(1-\beta_1)^2}{1-\beta_1^2(2-\beta_1)^2}(1+\frac{1}{4(1-\beta_1^2)})\sigma^2 := C\sigma^2,\end{aligned}$$

due to $\beta_1 < 1$, $m'_0 = 0$ and the bounded variance assumption. Hence,

$$\mathbb{E}[\|a'_t\|^2] \leq \frac{C\sigma^2}{\epsilon}.$$

Bounding the compression error. For the last term in (59), again by induction,

$$\begin{aligned}\|e_t\| &= \|e_{t-1} + g_{t-1} - \tilde{g}_{t-1}\| \\ &= \|g_{t-1} + e_{t-1} - TopK(g_{t-1}+e_{t-1}, k)\| \\ &\leq q\,\|g_{t-1}+e_{t-1}\| \\ &\leq q\,\|e_{t-1}\| + q\,\|g_{t-1}\| \\ &\leq \frac{q}{1-q}G.\end{aligned} \tag{60}$$

Since $\|a'_t\|^2 \leq G/\epsilon$, we derive

$$\mathbb{E}[\|\mathcal{E}_t\|\|a'_t\|] \leq \frac{qG^2}{(1-q)\epsilon}.$$

Completing the proof. Summing (59) over $t = 1, ..., T$ and integrating things together, we have

$$\begin{aligned}\mathbb{E}[f(\theta'_{T+1}) - f(\theta'_1)] &\leq \eta\sum_{t=1}^{T} M_t + \frac{T\eta^2 CL\sigma^2}{2\epsilon} + \frac{T\eta^2 LqG^2}{(1-q)\epsilon} \\ &\leq -\sum_{t=1}^{T}\frac{\eta(1-\beta_1)}{\sqrt{(q^2+1)G^2+\epsilon}}\mathbb{E}[\|\nabla f(\theta_t)\|^2] + \frac{2\eta dG^2}{(1-\beta_1)\sqrt{\epsilon}} \\ &\quad + \frac{T\eta^2\beta_1 L(q^2+1)G^4}{(1-\beta_1)\epsilon} + \frac{T\eta^2 CL\sigma^2}{2\epsilon} + \frac{T\eta^2 LqG^2}{(1-q)\epsilon}.\end{aligned}$$

Thus,

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(\theta_t)\|^2] &\leq C'\Big(\frac{\mathbb{E}[f(\theta'_1) - f(\theta'_{T+1})]}{T\eta} + \frac{2dG^2}{T(1-\beta_1)\sqrt{\epsilon}} \\ &\quad + \frac{\eta\beta_1 L(q^2+1)G^4}{(1-\beta_1)\epsilon} + \frac{\eta CL\sigma^2}{2\epsilon} + \frac{\eta LqG^2}{(1-q)\epsilon}\Big) \\ &\leq C'\Big(\frac{\mathbb{E}[f(\theta_1) - f(\theta^*)]}{T\eta} + \frac{2dG^2}{T(1-\beta_1)\sqrt{\epsilon}} \\ &\quad + \frac{\eta\beta_1 L(q^2+1)G^4}{(1-\beta_1)\epsilon} + \frac{\eta CL\sigma^2}{2\epsilon} + \frac{\eta LqG^2}{(1-q)\epsilon}\Big).\end{aligned}$$

where $C' = \frac{\sqrt{(q^2+1)G^2+\epsilon}}{1-\beta_1}$, and $C = \frac{(1-\beta_1)^2}{1-\beta_1^2(2-\beta_1)^2}(1+\frac{1}{4(1-\beta_1^2)})$. The last inequality is because $\theta'_1 = \theta_1$, and $\theta^* = \arg\min_\theta f(\theta)$.

Taking decreasing learning rate $\eta = 1/\sqrt{T}$, we obtain

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(\theta_t)\|^2] \leq \mathcal{O}(\frac{1}{\sqrt{T}} + \frac{1}{T}),$$

matching the convergence rate of SGD with error feedback [31].

$\square$