Layerwise and Dimensionwise Adaptive Local Adaptive Gradient Method for Federated Learning

Anonymous Authors[1]

Abstract

In the emerging paradigm of Federated Learning, large amount of clients, such as mobile devices, are used to train possibly high-dimensional models on their respective data. Under the orchestration of a central server, the data needs to remain decentralized, as it can not be shared among clients or with the central server. Then, due to the low bandwidth of mobile devices, decentralized optimization methods need to shift the computation burden from those clients to the computation server while preserving *privacy* and reasonable *communication cost*. In the particular case of training Deep multilayer Neural Networks, under such settings, we propose in this paper, FED-LAMB, a novel Federated Learning method based on a Layerwise and Dimensionwise updates of the local models. A periodic averaging is added to obtain estimates of the desired global model parameters. We provide a thorough finite time convergence analysis for our algorithm, substantiated by numerical runs on benchmark datasets.

1. Introduction

A growing and important task while learning models on observed data, is the ability to train the latter over a large number of clients which could either be devices or distinct entities. In the paradigm of Federated Learning (FL) (**??**), the focus of our paper, a central server orchestrates the optimization over those clients under the constraint that the data can neither be centralized nor shared among the clients. 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 $\mathbb{R}^d$. While this formulation recalls that of distributed optimization, the core principle of FL is different that standard distributed paradigm.

FL currently suffers from two bottlenecks: communication efficiency and privacy. We focus on the former in this paper. While local updates, updates during which each client learn their local models, can reduce drastically the number of communication rounds between the central server and devices, new techniques must be employed to tackle this challenge. Some quantization (**??**) or compression (**?**) methods allow to decrease the number of bits communicated at each round and are efficient method in a distributed setting. The other approach one can take is to accelerate the local training on each device and thus sending a better local model to the server at each round.

Under the important setting of heterogenous data, i.e. the data among each device can be distributed according to different distributions, current local optimization algorithms are perfectible. The most popular method for FL is using multiple local Stochastic Gradient Descent (SGD) steps in each device, sending those local models to the server that computes the average over those received local vector of parameters and broadcasts it back to the devices. This is called FEDAVG and has been introduced in (**?**).

In (**?**), the authors motivate the usage of adaptive gradient optimization methods as a better alternative to the standard SGD inner loop in FEDAVG. They propose an adaptive gradient method, namely LOCAL AMSGRAD, with communication cost sublinear in T that is guaranteed to converge to stationary points in $\mathcal{O}(\sqrt{d/Tn})$, where T is the number of iterations.

Based on recent progress in adaptive methods for accelerating the training procedure, see (**?**), we propose a variant of LOCAL AMSGRAD integrating dimensionwise and layerwise adaptive learning rate in each device's local update.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Our contributions are as follows:

- We develop a novel optimization algorithm for federated learning, namely FED-LAMB, following a principled layerwise adaptation strategy to accelerate training of deep neural networks.
- We provide a rigorous theoretical understanding of the non asymptotic convergence rate of FED-LAMB. Based on the recent progress on nonconvex stochastic optimization, we derive for a any finite number of rounds performed by our method, a characterization of the rate at which the classical suboptimilality condition, *i.e.,* , the second order moment of the gradient of the objective function, decreases. Our bound in $\mathcal{O}(\sqrt{\frac{p\mathsf{L}}{nR}})$ matches state of the art methods in Federated Learning reaching a sublinear convergence in R, the total number of rounds.
- We exhibit the advantages of our method on several benchmarks supervised learning methods on both homogeneous and heterogeneous settings.

The plan of our paper is as follows. After having established a literature review of both realms of federated and adaptive learning in subsection 1.1, we develop in Section 2, our method, namely FED-LAMB, based on the computation per layer and per dimension, of a scaling factor in the traditional stepsize of AMSGrad. Theoretical understanding of our method's behaviour with respect to convergence towards a stationary point is developed in Section 3. We present numerical illustrations showing the advantages of our method in Section 4.

1.1. Related Work

Adaptive gradient methods. In classical stochastic nonconvex optimization, adaptive methods have proven to be the spearhead of any practicioner. Those gradient based optimization algorithms alleviate the possibly high nonconvexity of the objective function by adaptively updating each coordinate of their learning rate using past gradients. Most used examples AMSGRAD (**?**), ADAM (**?**), RMSPROP (**?**), ADADELTA (**?**), and NADAM (**?**).

Their popularity and efficiency are due to their great performance at training deep neural networks. They generally combine the idea of adaptivity from ADAGRAD (**??**), as explained above, and the idea of momentum from NESTEROV'S METHOD (**?**) or HEAVY BALL method (**?**) using past gradients. ADAGRAD displays a great edge when the gradient is sparse compared to other classical methods. Its update has a notable feature: it leverages an anisotropic learning rate depending on the magnitude of the gradient for each dimension which helps in exploiting the geometry of the data.

The anisotropic nature of this update represented a real breakthrough in the training of high dimensional and nonconvex loss functions. This adaptive learning rate helps accelerating the convergence when the gradient vector is sparse (**?**), yet, when applying ADAGRAD to train deep neural networks, it is observed that the learning rate might decay too fast, see (**?**) for more details. Consequently, (**?**) develops ADAM leveraging a moving average of the gradients divided by the square root of the second moment of this moving average (element-wise multiplication). A variant, called AMSGRAD described in (**?**) ought to fix ADAM failures and is presented in Algorithm 1.

Algorithm 1 AMSGRAD (**?**)

1: **Required**: parameter β_1, β_2, and η_t.
2: Init: $w_1 \in \Theta \subseteq \mathbb{R}^d$ and $v_0 = \epsilon 1 \in \mathbb{R}^d$.
3: **for** $t = 1$ to T **do**
4: Get mini-batch stochastic gradient g_t at w_t.
5: $\theta_t = \beta_1 \theta_{t-1} + (1-\beta_1) g_t$.
6: $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$.
7: $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$.
8: $w_{t+1} = w_t - \eta_t \frac{\theta_t}{\sqrt{\hat{v}_t}}$. (element-wise division)
9: **end for**

The difference between ADAM and AMSGRAD lies in Line 1 of Algorithm 1.

A natural extension of Algorithm 1 has been developed in (**?**) specifically for multi layered neural network. A principled layerwise adaptation strategy to accelerate training of deep neural networks using large mini-batches is proposed using either a standard stochastic gradient update or a generalized adaptive method under the setting of a classical single server empirical risk minimization problem.

Federated learning. An extension of the well known parameter server framework, where a model is being trained on several servers in a distributed manner, is called Federated Learning, see (**?**). Here, the central server only plays the role of compute power for aggregation and global update of the model. Compared with the distributed learning paradigm, in Federated Learning, the data stored in each worker must not be seen by the central server – preserving privacy is key – and the nature of those workers, which can be mobile devices, combined with their usually large amount make communication between the devices and the central server less appealing – communication cost needs to be controlled.

Thus, while traditional distributed gradient methods (**???**) do not respect those constraints, it has been proposed in (**?**), an algorithm called Federated Averaging – FED-AVG – extending parallel SGD with local updates performed on each device. In FED-AVG, each worker updates their own model parameters locally using SGD, and the local mod-

els are synchronized by periodic averaging on the central parameter server.

2. Layerwise and Dimensionwise Adaptive Methods

Beforehand, it is important to provide useful and important notations used throughout our paper.

Notations: We denote by θ the vector of parameters taking values in $\mathbb{R}^d$. For each layer $\ell \in [\![\mathsf{L}]\!]$, where L is the total number of layers of the neural networks, and each coordinate $j \in [\![p_\ell]\!]$ where p_ℓ is the dimension per layer ℓ, we note $\theta^{\ell,j}$ its jth coordinate. The gradient of f with respect to θ^ℓ is denoted by $\nabla_\ell f(\theta)$. The index $i \in [\![n]\!]$ denotes the index of the worker i in our federated framework. r and t are used as the round and local iteration numbers respectively. The smoothness per layer is denoted by L_ℓ for each layer $\ell \in [\![\mathsf{L}]\!]$. We note for each communication $r > 0$, the set of randomly drawn devices D^r performing local updates.

2.1. AMSGrad, Local AMSGrad and Periodic Averaging

Under our Federated setting, we stress on the important of reducing the communication cost at each round between the central server, used mainly for aggregation purposes, and the many clients used for gradient computation and local updates. Using Periodic Averaging after few local epochs, updating local models on each device, as developed in (**?**) is the gold standard for achieving such communication cost reduction. Intuitively, one rather shift the computation burden from the many clients to the central server as much as possible. This allows for fewer local epochs and a better global model, from a loss minimization (or model fitting) perspective.

The premises of that new paradigm are SGD updates performed locally on each device then averaged periodically, see (**??**). The heuristic efficiency of local updates using SGD and periodic averaging has been studied in (**??**) and shown to reach a similar sublinear convergence rate as in the standard distributed optimization settings.

Then, with the growing need of training far more complex models, such as deep neural networks, several efficient methods, built upon adaptive gradient algorithms, such as Local AMSGrad in (**?**), extended both empirically and theoretically, the benefits of performing local updates coupled with periodic averaging.

2.2. Layerwise and Dimensionwise Learning with Periodic Averaging

Recall that our original problem is the following optimization task:

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

where $f_i(\theta)$ is the loss function associated to the client $i \in [\![n]\!]$ and is parameterized, in our paper, by a deep neural network. The multilayer and nonconvex nature of the loss function implies having recourse to particular optimization methods in order to efficiently train our model. Besides, the distributed and clients low bandwidth constraints are strong motivations for improving existing methods performing (1).

Based on the periodic averaging and local AMSGrad algorithms, presented prior, we propose a layerwise and dimensionwise local AMS algorithm described in the following:

Algorithm 2 FED-LAMB for Federated Learning

1: **Input**: parameter β_1, β_2, and learning rate α_t.
2: Init: $\theta_0 \in \Theta \subseteq \mathbb{R}^d$, as the global model and $\hat{v}_0 = v_0 = \epsilon 1 \in \mathbb{R}^d$ and $\bar{\theta}_0 = \frac{1}{n}\sum_{i=1}^n \theta_0$.
3: **for** $r = 1$ to R **do**
4: Set $\theta^0_{r,i} = \bar{\theta}_{r-1}$
5: **for** parallel for device $d \in D^r$ **do**
6: Compute stochastic gradient $g_{r,i}$ at θ_r.
7: **for** $t = 1$ to T **do**
8: $m^t_{r,i} = \beta_1 m^{t-1}_{r-1,i} + (1-\beta_1) g_{r,i}$.
9: $m^t_{r,i} = m^t_{r,i} / (1-\beta_1^r)$.
10: $v^{t,i}_r = \beta_2 v^t_{r-1,i} + (1-\beta_2) g^2_{r,i}$.
11: $v^t_{r,i} = v^t_{r,i} / (1-\beta_2^r)$.
12: Compute ratio $p_{r,i} = \frac{m^t_{r,i}}{\sqrt{\hat{v}_r+\epsilon}}$.
13: Update local model for each layer ℓ:

$$\theta^{\ell,t}_{r,i} = \theta^{\ell,t-1}_{r,i} - \alpha_r \phi(\|\theta^{\ell,t-1}_{r,i}\|) \frac{p^\ell_{r,i} + \lambda\theta^{\ell,t-1}_{r,i}}{\|p^\ell_{r,i} + \lambda\theta^{\ell,t-1}_{r,i}\|}$$

14: **end for**
15: Devices send $\theta^T_{r,i} = [\theta^{\ell,T}_{r,i}]_{\ell=1}^{\mathsf{L}}$ and $v^T_{r,i}$ to server.
16: **end for**
17: Server computes the averages of the local models $\bar{\theta}^\ell_r = \frac{1}{n}\sum_{i=1}^n \theta^{\ell,T}_{r,i}$ and $\hat{v}_{r+1} = \max(\hat{v}_r, \frac{1}{n}\sum_{i=1}^n v^T_{r,i})$ and send them back to the devices.
18: **end for**

Algorithm 2 is a natural adaptation of the vanilla AMSGrad method, for *multilayer* neural networks under the *distributed* settings. In particular, while Line 2 and Line 2 corresponds to the standard approximation of the first and second moments, via the smooth updates allowed by the tuning parameters β_1 and β_2 respectively and that both Lines 2-2 are

correct the biases of those estimates, the final local update in Line 2 is novel and corresponds to the specialization per layer of our federated method. Note that a scaling factor is applied to the learning rate α_r at each round $r > 0$ via the quantity $\phi(\|\theta_{r,i}^{\ell,t-1}\|)$ depending on the dimensionwise and layerwise quantity computed in Line 2. This function is user designed and can be set to the identity function.

The adaptivity of our federated method is thus manifold. There occurs a per dimension normalization with respect to the square root of the second moment used in adaptive gradient methods and a layerwise normalization obtained via the final local update (Line 2).

3. On The Convergence of FED-LAMB

We develop in this section, the theoretical analysis of Algorithm2. based on classical result for stochastic nonconvex optimization, we characterize the convergence of our algorithm by upper bounding the second order moment of the objective function in (1). This suboptimality condition becomes smaller as the algorithm progresses and being able to upper bound it describes how fast our method reaches an ϵ-stationary point. Particularly, in our case, we will focus on the evolution of the following quantity: $\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}[\|\nabla f(\overline{\theta_r})\|^2$ which is the average over a finite time of R rounds of communication, evaluated at the periodically average vector of parameters noted $\overline{\theta_r} = [\bar{\theta}_r^{\ell}]_{\ell=1}^{\mathsf{L}}$ where $\bar{\theta}_r^{\ell}$ is defined in Line 2 of Algorithm2.

In the context of nonconvex stochastic optimization for distributed devices, assume the following:

H 1. *For $i \in [\![n]\!]$ and $\ell \in [\![L]\!]$, f_i is L-smooth: $\|\nabla f_i(\theta) - \nabla f_i(\vartheta)\| \leq L_\ell \left\|\theta^\ell - \vartheta^\ell\right\|$.*

We add some classical assumption in the unbiased stochastic optimization realm, on the gradient of the objective function:

H2. *The stochastic gradient is unbiased for any iteration $r > 0$: $\mathbb{E}[g_r] = \nabla f(\theta_r)$ and is bounded from above, i.e., $\|g_t\| \leq M$.*

H3. *The variance of the stochastic gradient is bounded for any iteration $r > 0$ and any dimension $j \in [\![d]\!]$: $\mathbb{E}[|g_r^j - \nabla f(\theta_r)^j|^2] < \sigma^2$.*

H4. *For any value $a \in \mathbb{R}_+^*$, there exists strictly positive constants such that $\phi_m \leq \phi(a) \leq \phi_M$.*

We now state our main result regarding the non asymptotic convergence analysis of our Algorithm 2:

Theorem 1. *Consider $\{\overline{\theta_r}\}_{r>0}$, the sequence of parameters obtained running Algorithm 2. Then, if the number of local epochs is set to $T = 1$ and $\epsilon = \lambda = 0$, we have:*

$$\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}[\|\nabla f(\overline{\theta_r})\|^2 \leq dd \tag{2}$$

4. Numerical experiments

In this section, we conduct numerical experiments on various datasets and network architectures to testify the effectiveness of our proposed method in practice.

Settings. In our experiment, we will evaluate three federated learning algorithms: 1) Fed-SGD, 2) Fed-AMS and 3) our proposed Fed-LAMB (Algorithm 2), where the first two serve as the baseline methods. For adaptive methods 2) and 3), we set $\beta_1 = 0.9$, $\beta_2 = 0.999$ as default and recommended (**?**). Regarding federated learning environment, we use 50 local workers with 0.5 participation rate. That means, we randomly pick half of the workers to be active for training in each round. To best accord with real scenarios where the local training batch size is usually limited, we set a relatively small local update batch size as 32. In each round, the training samples are allocated to the active devices, and one local epoch is finished after all the local devices run one epoch over their received samples by batch training. We test different number of local epochs in our experiments. For each dataset and number of local epochs, we tune the constant learning rate α for each algorithm in logarithm scale. For LocalLAMB, the parameter λ in Algorithm 2 controlling the overall scale of the layerwise gradients is tuned from $\{0, 0.01, 0.1\}$. For each run, we take the model performance with the best α and λ. The reported results are averaged over three independent runs each with same initialization.

Models. We test the performance of different federated learning algorithms on MNIST and CIFAR10 image classification datasets. For MNIST, we apply 1) a simple multilayer perceptron (MLP), which has one hidden layer containing 200 cells with dropout; 2) Convolutional Neural Network (CNN), which has two max-pooled convolutional layers followed by a dropout layer and two fully-connected layers with 320 and 50 cells respectively. For CIFAR10, we implement: 1) a CNN with three convolutional layers followed by two fully-connected layers, and 2) a ResNet-9 model proposed by (**?**). All the networks use ReLU as the activation function.

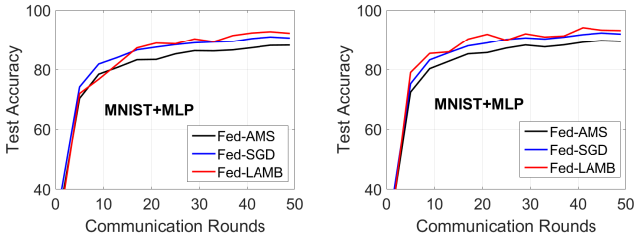

Figure 1. Test accuracy on CNN + MNIST. Non-iid data distribution.

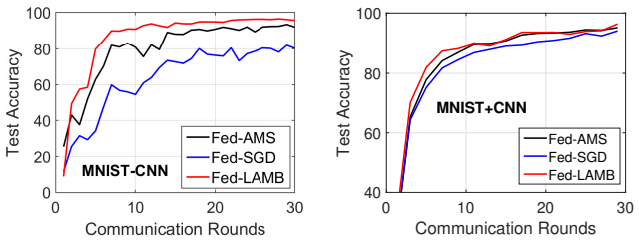

Figure 2. Test accuracy on CNN + MNIST. Non-iid data distribution.

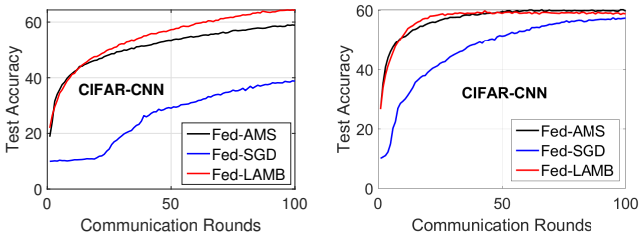

Figure 3. Test accuracy on CNN + CIFAR10. iid data distribution.

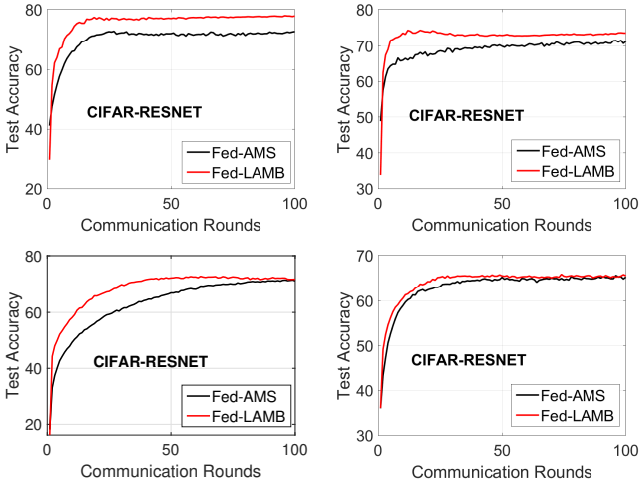

Figure 4. Test accuracy on ResNet + CIFAR10. iid data distribution.

5. Conclusion

A. Appendix

B. Theoretical Analysis

B.1. Intermediary Lemmas

Lemma 1. *Consider $\{\overline{\theta_r}\}_{r>0}$, the sequence of parameters obtained running Algorithm 2. Then for $i \in [\![n]\!]$:*

$$\|\overline{\theta_r} - \theta_{r,i}\| \leq \alpha^2 M^2 \phi_M^2 \frac{(1-\beta_2)p}{v_0} \tag{3}$$

Proof. Assuming the simplest case when $T = 1$, i.e. one local iteration, then by construction of Algorithm 2, we have for all $\ell \in [\![\mathsf{L}]\!]$, $i \in [\![n]\!]$ and $r > 0$:

$$\theta_{r,i}^{\ell} = \overline{\theta_r}^{\ell} - \alpha\phi(\|\theta_{r,i}^{\ell,t-1}\|)p_{r,i}^{j}/\|p_{r,i}^{\ell}\| = \overline{\theta_r}^{\ell} - \alpha\phi(\|\theta_{r,i}^{\ell,t-1}\|)\frac{m_{r,i}^{t}}{\sqrt{v_r^t}}\frac{1}{\|p_{r,i}^{\ell}\|} \tag{4}$$

leading to

$$\begin{aligned}\|\overline{\theta_r} - \theta_{r,i}\|^2 &= \left\langle \overline{\theta_r}^{\ell} - \theta_{r,i}^{\ell} \,|\, \overline{\theta_r}^{\ell} - \theta_{r,i}^{\ell} \right\rangle \\ &\leq \alpha^2 M^2 \phi_M^2 \frac{(1-\beta_2)p}{v_0}\end{aligned} \tag{5}$$

which concludes the proof. □

B.2. Proof of Theorem 1

Theorem. *Consider $\{\overline{\theta_r}\}_{r>0}$, the sequence of parameters obtained running Algorithm 2. Then, if the number of local epochs is set to $T = 1$ and $\epsilon = \lambda = 0$, we have:*

$$\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}[\|\nabla f(\overline{\theta_r})\|^2 \leq dd \tag{6}$$

Case with $T = 1$, $\epsilon = 0$ and $\lambda = 0$: Using H1, we have:

$$\begin{aligned}f(\bar{\vartheta}_{r+1}) &\leq f(\bar{\vartheta}_r) + \left\langle \nabla f(\bar{\vartheta}_r) \,|\, \bar{\vartheta}_{r+1} - \bar{\vartheta}_r \right\rangle + \sum_{\ell=1}^{L}\frac{L_\ell}{2}\|\bar{\vartheta}_{r+1}^{\ell} - \bar{\vartheta}_r^{\ell}\|^2 \\ &\leq f(\bar{\vartheta}_r) + \sum_{\ell=1}^{\mathsf{L}}\sum_{j=1}^{p_\ell}\nabla_\ell f(\bar{\vartheta}_r)^j(\bar{\vartheta}_{r+1}^{\ell,j} - \bar{\vartheta}_r^{\ell,j}) + \sum_{\ell=1}^{L}\frac{L_\ell}{2}\|\bar{\vartheta}_{r+1}^{\ell} - \bar{\vartheta}_r^{\ell}\|^2\end{aligned} \tag{7}$$

Taking expectations on both sides leads to:

$$-\mathbb{E}[\left\langle \nabla f(\bar{\vartheta}_r) \,|\, \bar{\vartheta}_{r+1} - \bar{\vartheta}_r \right\rangle] \leq \mathbb{E}[f(\bar{\vartheta}_r) - f(\bar{\vartheta}_{r+1})] + \sum_{\ell=1}^{L}\frac{L_\ell}{2}\mathbb{E}[\|\bar{\vartheta}_{r+1}^{\ell} - \bar{\vartheta}_r^{\ell}\|^2] \tag{8}$$

Yet, we observe that, using the classical intermediate quantity, used for proving convergence results of adaptive optimization methods, see (), we have:

$$\bar{\vartheta}_r = \bar{\theta}_r + \frac{\beta_1}{1-\beta_1}(\bar{\theta}_r - \bar{\theta}_{r-1}) \tag{9}$$

where $\bar{\theta}_r$ denotes the average of the local models at round r. Then for each layer ℓ,

$$\bar{\vartheta}^\ell_{r+1} - \bar{\vartheta}^\ell_r = \frac{1}{1-\beta_1}(\bar{\theta}^\ell_{r+1} - \bar{\theta}^\ell_r) - \frac{\beta_1}{1-\beta_1}(\bar{\theta}^\ell_r - \bar{\theta}^\ell_{r-1}) \tag{10}$$

$$= \frac{\alpha_r}{1-\beta_1}\frac{1}{n}\sum_{i=1}^n \frac{\phi(\|\theta^\ell_{r,i}\|)}{\|p^\ell_{r,i}\|}p^\ell_{r,i} - \frac{\alpha_{r-1}}{1-\beta_1}\frac{1}{n}\sum_{i=1}^n \frac{\phi(\|\theta^\ell_{r-1,i}\|)}{\|p^\ell_{r-1,i}\|}p^\ell_{r-1,i} \tag{11}$$

$$= \frac{\alpha\beta_1}{1-\beta_1}\frac{1}{n}\sum_{i=1}^n \left(\frac{\phi(\|\theta^\ell_{r,i}\|)}{\sqrt{v^t_r}\|p^\ell_{r,i}\|} - \frac{\phi(\|\theta^\ell_{r-1,i}\|)}{\sqrt{v^t_{r-1}}\|p^\ell_{r-1,i}\|}\right) m^t_{r-1} + \frac{\alpha}{n}\sum_{i=1}^n \frac{\phi(\|\theta^\ell_{r,i}\|)}{\sqrt{v^t_r}\|p^\ell_{r,i}\|}g_{r,i} \tag{12}$$

where we have assumed a constant learning rate α.

We note for all $\theta \in \Theta$, the majorant $G > 0$ such that $\phi(\|\theta\|) \leq G$. Then, following (8), we obtain:

$$-\mathbb{E}[\langle \nabla f(\bar{\vartheta}_r) \,|\, \bar{\vartheta}_{r+1} - \bar{\vartheta}_r\rangle] \leq \mathbb{E}[f(\bar{\vartheta}_r) - f(\bar{\vartheta}_{r+1})] + \sum_{\ell=1}^L \frac{L_\ell}{2}\mathbb{E}[\|\bar{\vartheta}_{r+1} - \bar{\vartheta}_r\|^2] \tag{13}$$

Developing the LHS of (13) using (10) leads to

$$\langle \nabla f(\bar{\vartheta}_r) \,|\, \bar{\vartheta}_{r+1} - \bar{\vartheta}_r\rangle = \sum_{\ell=1}^{\mathsf{L}}\sum_{j=1}^{p_\ell} \nabla_\ell f(\bar{\vartheta}_r)^j(\bar{\vartheta}^{\ell,j}_{r+1} - \bar{\vartheta}^{\ell,j}_r) \tag{14}$$

$$= \frac{\alpha\beta_1}{1-\beta_1}\frac{1}{n}\sum_{\ell=1}^{\mathsf{L}}\sum_{j=1}^{p_\ell}\nabla_\ell f(\bar{\vartheta}_r)^j\left[\sum_{i=1}^n\left(\frac{\phi(\|\theta^\ell_{r,i}\|)}{\sqrt{v^t_r}\|p^\ell_{r,i}\|} - \frac{\phi(\|\theta^\ell_{r-1,i}\|)}{\sqrt{v^t_{r-1}}\|p^\ell_{r-1,i}\|}\right)m^t_{r-1}\right] \tag{15}$$

$$\underbrace{-\frac{\alpha}{n}\sum_{\ell=1}^{\mathsf{L}}\sum_{j=1}^{p_\ell}\nabla_\ell f(\bar{\vartheta}_r)^j\sum_{i=1}^n\frac{\phi(\|\theta^\ell_{r,i}\|)}{\sqrt{v^t_r}\|p^\ell_{r,i}\|}g_{r,i}}_{=A_1} \tag{16}$$

Term A_1: Since we have that $\|p^\ell_{r,i}\| \leq \sqrt{\frac{p_\ell}{1-\beta_2}}$ and $1/\sqrt{v^t_r} \leq 1/\sqrt{v_0}$, using H2, we develop the term A_1 as follows:

$$A_1 \leq -\frac{\alpha}{n}\sum_{\ell=1}^{\mathsf{L}}\sum_{j=1}^{p_\ell}\nabla_\ell f(\bar{\vartheta}_r)^j\sum_{i=1}^n\frac{\phi(\|\theta^\ell_{r,i}\|)}{\sqrt{v^t_r}\|p^\ell_{r,i}\|}g_{r,i} \tag{17}$$

$$\leq -\frac{\alpha}{n}\sum_{\ell=1}^{\mathsf{L}}\sqrt{\frac{1-\beta_2}{M^2 p_\ell}}\sum_{i=1}^n\sum_{j=1}^{p_\ell}\phi(\|\theta^\ell_{r,i}\|)\nabla_\ell f(\bar{\vartheta}_r)^j g^{\ell,j}_{r,i} \tag{18}$$

$$-\frac{\alpha}{n}\sum_{\ell=1}^{\mathsf{L}}\sum_{i=1}^n\sum_{j=1}^{p_\ell}\left(\phi(\|\theta^\ell_{r,i}\|)\nabla_\ell f(\bar{\vartheta}_r)^j\frac{p^\ell_{r,i}}{\|p^\ell_{r,i}\|}\right)1\left(\text{sign}(\nabla_\ell f(\bar{\vartheta}_r)^j) \neq \text{sign}(p^\ell_{r,i})\right) \tag{19}$$

Taking the expectations on both sides yields:

$$\mathbb{E}[A_1] \leq -\alpha \sum_{\ell=1}^{\mathsf{L}} \sqrt{\frac{1-\beta_2}{M^2 p_\ell}} \sum_{i=1}^{n} \sum_{j=1}^{p_\ell} \mathbb{E}\left[\phi(\|\theta_{r,i}^\ell\|) \nabla_\ell f(\bar{\vartheta}_r)^j g_{r,i}^{\ell,j}\right] \tag{20}$$

$$- \frac{\alpha}{n} \sum_{\ell=1}^{\mathsf{L}} \sum_{i=1}^{n} \sum_{j=1}^{p_\ell} \mathbb{E}\left[\phi(\|\theta_{r,i}^\ell\|) \nabla_\ell f(\bar{\vartheta}_r)^j \frac{p_{r,i}^\ell}{\|p_{r,i}^\ell\|} 1\left(\text{sign}(\nabla_\ell f(\bar{\vartheta}_r)^j) \neq \text{sign}(p_{r,i}^\ell)\right)\right] \tag{21}$$

$$\leq -\frac{\alpha}{n} \sum_{\ell=1}^{\mathsf{L}} \phi_m \sqrt{\frac{1-\beta_2}{M^2 p_\ell}} \sum_{i=1}^{n} \sum_{j=1}^{p_\ell} (\nabla_\ell f(\bar{\vartheta}_r)^j)^2 \tag{22}$$

$$- \frac{\alpha}{n} \sum_{\ell=1}^{\mathsf{L}} \sum_{i=1}^{n} \sum_{j=1}^{p_\ell} \phi_M \mathbb{E}\left[\left|\nabla_\ell f(\bar{\vartheta}_r)^j \frac{p_{r,i}^\ell}{\|p_{r,i}^\ell\|}\right| 1\left(\text{sign}(\nabla_\ell f(\bar{\vartheta}_r)^j) \neq \text{sign}(p_{r,i}^\ell)\right)\right] \tag{23}$$

$$\tag{24}$$

where we have used assumption H4.

Since for any ℓ, i, j, we have

$$\mathbb{E}\left[\left|\nabla_\ell f(\bar{\vartheta}_r)^j \frac{p_{r,i}^\ell}{\|p_{r,i}^\ell\|}\right| 1\left(\text{sign}(\nabla_\ell f(\bar{\vartheta}_r)^j) \neq \text{sign}(p_{r,i}^\ell)\right)\right] \leq \left|\nabla_\ell f(\bar{\vartheta}_r)^j\right| \mathbb{P}\left(\text{sign}(\nabla_\ell f(\bar{\vartheta}_r)^j) \neq \text{sign}(p_{r,i}^\ell)\right) \tag{25}$$

Then, we obtain

$$\mathbb{E}[A_1] \leq -\alpha\phi_m \sqrt{\frac{\mathsf{L}(1-\beta_2)}{M^2 p}} \mathbb{E}[\|\overline{\nabla f}(\bar{\vartheta}_r)\|^2] - \alpha\phi_M \sum_{\ell=1}^{\mathsf{L}} \sum_{i=1}^{n} \sum_{j=1}^{p_\ell} \frac{\sigma_i^{\ell,j}}{\sqrt{n}} \tag{26}$$

where $\overline{\nabla f}(\cdot) = \sum_{i=1}^n \nabla f_i(\cdot)$

We now need to bound the following terms:

$$A_r^2 := \mathbb{E}[\|\bar{\vartheta}_{r+1} - \bar{\vartheta}_r\|^2] \tag{27}$$

$$A_r^3 := \frac{\alpha\beta_1}{1-\beta_1} \frac{1}{n} \sum_{\ell=1}^{\mathsf{L}} \sum_{j=1}^{p_\ell} \nabla_\ell f(\bar{\vartheta}_r)^j \left[\sum_{i=1}^{n} \left(\frac{\phi(\|\theta_{r,i}^\ell\|)}{\sqrt{v_r^t}\|p_{r,i}^\ell\|} - \frac{\phi(\|\theta_{r-1,i}^\ell\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^\ell\|}\right) m_{r-1}^t\right] \tag{28}$$

Term A_r^2: According to definition (9), for each layer $\ell \in [\![\mathsf{L}]\!]$, we have, using the Cauchy-Schwartz inequality, that:

$$\|\bar{\vartheta}_{r+1}^\ell - \bar{\vartheta}_r^\ell\|^2 = \left\|\frac{\alpha\beta_1}{1-\beta_1} \frac{1}{n} \sum_{i=1}^{n} \left(\frac{\phi(\|\theta_{r,i}^\ell\|)}{\sqrt{v_r^t}\|p_{r,i}^\ell\|} - \frac{\phi(\|\theta_{r-1,i}^\ell\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^\ell\|}\right) m_{r-1}^t + \frac{\alpha}{n} \sum_{i=1}^{n} \frac{\phi(\|\theta_{r,i}^\ell\|)}{\sqrt{v_r^t}\|p_{r,i}^\ell\|} g_{r,i}\right\|^2 \tag{29}$$

$$\leq 2\frac{\alpha^2}{n^2} \left\|\frac{\beta_1}{1-\beta_1} \sum_{i=1}^{n} \left(\frac{\phi(\|\theta_{r,i}^\ell\|)}{\sqrt{v_r^t}\|p_{r,i}^\ell\|} - \frac{\phi(\|\theta_{r-1,i}^\ell\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^\ell\|}\right) m_{r-1}^t\right\|^2 + \frac{1}{n^2} \left\|\sum_{i=1}^{n} \frac{\phi(\|\theta_{r,i}^\ell\|)}{\sqrt{v_r^t}\|p_{r,i}^\ell\|} g_{r,i}\right\|^2 \tag{30}$$

Taking the expectation on both sides leads to:

$$\begin{aligned}
\mathbb{E}[\|\bar{\vartheta}_{r+1}^{\ell} - \bar{\vartheta}_{r}^{\ell}\|^2] &\leq 2\alpha^2 \mathbb{E}\left[\left\|\frac{\beta_1}{1-\beta_1}\sum_{i=1}^{n}\left(\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} - \frac{\phi(\|\theta_{r-1,i}^{\ell}\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^{\ell}\|}\right) m_{r-1}^t\right\|^2\right] + \frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} g_{r,i}\right\|^2\right] \\
&\leq 2\frac{\alpha^2}{n^2}\mathbb{E}\left[\left\|\frac{\beta_1}{1-\beta_1}\sum_{i=1}^{n}\left(\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} - \frac{\phi(\|\theta_{r-1,i}^{\ell}\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^{\ell}\|}\right) m_{r-1}^t\right\|^2\right] \\
&\quad + \frac{1}{n^2}\mathbb{E}\left[\left|\sum_{i=1}^{n}\sum_{j=1}^{p}\left\langle \Gamma_{r,i}^j(\nabla f_i(\theta_r)^j + g_{r,i}^j - \nabla f_i(\theta_r)^j) \,|\, \Gamma_{r,i}^j(\nabla f_i(\theta_r)^j + g_{r,i}^j - \nabla f_i(\theta_r)^j)\right\rangle\right|\right] \\
&\leq 2\frac{\alpha^2}{n^2}\mathbb{E}\left[\left\|\frac{\beta_1}{1-\beta_1}\sum_{i=1}^{n}\left(\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} - \frac{\phi(\|\theta_{r-1,i}^{\ell}\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^{\ell}\|}\right) m_{r-1}^t\right\|^2\right] \\
&\quad + \frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r)\right\|^2\right] + \frac{1}{n}\left\|\sum_{i=1}^{n}\sigma_i^2\mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\right]\right\|^2
\end{aligned} \tag{31}$$

where the last line uses assumptions H2 and H3 (unbiased gradient and bounded variance of the stochastic gradient) and $\Gamma := \frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}$.

On the other hand, using the bound on the gradient H2,

$$\begin{aligned}
&\sum_{r=1}^{R}\mathbb{E}\left[\left\|\frac{\beta_1}{1-\beta_1}\sum_{i=1}^{n}\left(\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} - \frac{\phi(\|\theta_{r-1,i}^{\ell}\|)}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^{\ell}\|}\right) m_{r-1}^t\right\|^2\right] \\
&\leq \frac{\beta_1^2}{(1-\beta_1)^2}M^2\phi_M^2\sum_{r=1}^{R}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\left(\frac{1}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} - \frac{1}{\sqrt{v_{r-1}^t}\|p_{r-1,i}^{\ell}\|}\right)\right\|^2\right] \\
&\leq \frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\sum_{r=1}^{R}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\left(\frac{1}{\sqrt{v_r^t}} - \frac{1}{\sqrt{v_{r-1}^t}}\right)\right\|^2\right] \\
&\leq \frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\sum_{r=1}^{R}\mathbb{E}\left[\left|\sum_{i=1}^{n}\sum_{j=1}^{p}\left(\frac{1}{\sqrt{v_r^{t,j}}} - \frac{1}{\sqrt{v_{r-1}^{t,j}}}\right)\right|\right] \\
&\leq \frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\frac{np}{v_0}
\end{aligned} \tag{32}$$

where, in the telescopic sum, we have used the initial value v_0 of the non decreasing sequence $\{v_r^t\}_{r>0}$ by construction (max operator).

Combining (32) into (31) and summing over the total number of rounds R yields

$$\begin{aligned}
\sum_{r=1}^{R}A_r^2 := \sum_{r=1}^{R}\mathbb{E}[\|\bar{\vartheta}_{r+1}^{\ell} - \bar{\vartheta}_{r}^{\ell}\|^2] &\leq \frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\frac{np}{v_0} \\
&\quad + \sum_{r=1}^{R}\left[\frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r)\right\|^2\right] + \frac{1}{n}\left\|\sum_{i=1}^{n}\sigma_i^2\mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\right]\right\|^2\right]
\end{aligned} \tag{33}$$

Term A_r^3: According to similar arguments on the non decreasing sequence involved in the algorithm as in the previous series of calculations, observe that

$$\sum_{r=1}^{R} A_r^3 \leq \frac{\alpha\beta_1}{1-\beta_1}\sqrt{(1-\beta_2)p}\frac{\mathsf{L}M^2}{\sqrt{v_0}} \tag{34}$$

Plugging (26) into (13) combined with (33) and (34) injected into the original smoothness definition (8) summed over the total number of rounds:

$$-\sum_{r=1}^{R}\mathbb{E}[\langle\nabla f(\bar{\vartheta}_r)\,|\,\bar{\vartheta}_{r+1}-\bar{\vartheta}_r\rangle] \leq \sum_{r=1}^{R}\mathbb{E}[f(\bar{\vartheta}_r)-f(\bar{\vartheta}_{r+1})] + \sum_{r=1}^{R}\sum_{\ell=1}^{L}\frac{L_\ell}{2}\mathbb{E}[\|\bar{\vartheta}_{r+1}^{\ell}-\bar{\vartheta}_r^{\ell}\|^2] \tag{35}$$

gives:

$$\begin{aligned}
&\sum_{r=1}^{R}\alpha\phi_m\sqrt{\frac{\mathsf{L}(1-\beta_2)}{M^2p}}\mathbb{E}[\|\overline{\nabla f}(\bar{\vartheta}_r)\|^2] - \alpha\phi_M\sum_{\ell=1}^{\mathsf{L}}\sum_{i=1}^{n}\sum_{j=1}^{p_\ell}\frac{\sigma_i^{\ell,j}}{\sqrt{n}} + \frac{\alpha\beta_1}{1-\beta_1}\sqrt{(1-\beta_2)p}\frac{\mathsf{L}M^2}{\sqrt{v_0}}\\
&\leq \sum_{r=1}^{R}\mathbb{E}[f(\bar{\vartheta}_r)-f(\bar{\vartheta}_{r+1})] + \sum_{\ell=1}^{L}\frac{L_\ell}{2}\frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\frac{np}{v_0}\\
&\quad -\sum_{r=1}^{R}\left[\frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r)\right\|^2\right] + \frac{1}{n}\left\|\sum_{i=1}^{n}\sigma_i^2\mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\right\|^2\right]\right]
\end{aligned} \tag{36}$$

Noting that $\sum_{r=1}^{R}\mathbb{E}[f(\bar{\vartheta}_r)-f(\bar{\vartheta}_{r+1})] = f(\bar{\vartheta}_1)-\mathbb{E}[f(\bar{\vartheta}_{R+1})]$, we obtain

$$\begin{aligned}
&\sum_{r=1}^{R}\alpha\phi_m\sqrt{\frac{\mathsf{L}(1-\beta_2)}{M^2p}}\mathbb{E}[\|\overline{\nabla f}(\bar{\vartheta}_r)\|^2] + \frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r)\right\|^2\right]\\
&\leq f(\bar{\vartheta}_1)-\mathbb{E}[f(\bar{\vartheta}_{R+1})] + \frac{1}{n}\left\|\sum_{i=1}^{n}\sigma_i^2\mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\right\|^2\right] + \alpha\phi_M\sum_{\ell=1}^{\mathsf{L}}\sum_{i=1}^{n}\sum_{j=1}^{p_\ell}\frac{\sigma_i^{\ell,j}}{\sqrt{n}} + \frac{\alpha\beta_1}{1-\beta_1}\sqrt{(1-\beta_2)p}\frac{\mathsf{L}M^2}{\sqrt{v_0}}\\
&+\sum_{\ell=1}^{L}\frac{L_\ell}{2}\frac{\beta_1^2}{(1-\beta_1)^2}\frac{\mathsf{L}(1-\beta_2)}{p}M^2\phi_M^2\frac{np}{v_0}
\end{aligned} \tag{37}$$

leading to

$$\begin{aligned}
\sum_{r=1}^{R}\frac{1}{n^2}\mathbb{E}\left[\left\|\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r)\right\|^2\right] &\leq f(\bar{\vartheta}_1)-\mathbb{E}[f(\bar{\vartheta}_{R+1})] + \frac{1}{n}\left\|\sum_{i=1}^{n}\sigma_i^2\mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\right\|^2\right]\\
&+\alpha\phi_M\sigma\mathsf{L}p\sqrt{n} + \frac{\overline{L}_\ell\beta_1^2\mathsf{L}(1-\beta_2)M^2\phi_M^2n}{2(1-\beta_1)^2v_0} + \frac{\alpha\beta_1}{1-\beta_1}\sqrt{(1-\beta_2)p}\frac{\mathsf{L}M^2}{\sqrt{v_0}}
\end{aligned} \tag{38}$$

where $\overline{L}_\ell = \sum_{\ell=1}^{\mathsf{L}} L_\ell$ is the sum of all smoothness constants.

Consider the following inequality:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|}\nabla f_i(\theta_r) \leq \phi_M(1-\beta_2)\frac{\overline{\nabla} f(\theta_r)}{\sqrt{v_r^t}} \tag{39}$$

where $\overline{\nabla} f(\theta_r) := \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\theta_r)$. And using the Cauchy-Schwartz inequality we have

$$\left\| \frac{\overline{\nabla} f(\theta_r)}{\sqrt{v_r^t}} \right\| \geq \frac{1}{2} \left\| \frac{\nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\| - \left\| \frac{\overline{\nabla} f(\theta_r) - \nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\| \tag{40}$$

Using Lemma 1 and the smoothness assumption H1, we have

$$\begin{aligned} \left\| \frac{\overline{\nabla} f(\theta_r)}{\sqrt{v_r^t}} \right\| &\geq \frac{1}{2} \left\| \frac{\nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\| - \left\| \frac{\overline{\nabla} f(\theta_r) - \nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\| \\ &\geq \frac{1}{2} \left\| \frac{\nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\| - \overline{L}\alpha^2 M^2 \phi_M^2 \frac{(1-\beta_2)p}{v_0} \end{aligned} \tag{41}$$

Plugging the above inequality into (38) returns:

$$\begin{aligned} \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}\left[\left\| \frac{\nabla f(\overline{\theta_r})}{\sqrt{v_r^t}} \right\|^2 \right] &\leq \left(f(\bar{\vartheta}_1) - \mathbb{E}[f(\bar{\vartheta}_{R+1})] \right) + \frac{1}{n} \left\| \sum_{i=1}^{n} \sigma_i^2 \mathbb{E}\left[\frac{\phi(\|\theta_{r,i}^{\ell}\|)}{\sqrt{v_r^t}\|p_{r,i}^{\ell}\|} \right] \right\|^2 \\ &\quad + \alpha\phi_M \sigma \mathsf{L} p\sqrt{n} + \frac{\overline{L}_{\ell}\beta_1^2 \mathsf{L}(1-\beta_2) M^2 \phi_M^2 n}{2(1-\beta_1)^2 v_0} + \frac{\alpha\beta_1}{1-\beta_1}\sqrt{(1-\beta_2)p}\frac{\mathsf{L}M^2}{\sqrt{v_0}} + \overline{L}\alpha^2 M^2 \phi_M^2 \frac{(1-\beta_2)p}{R v_0} \end{aligned} \tag{42}$$

concluding the proof of our main convergence result.