FedBAT: Communication-Efficient Federated Learning via Learnable Binarization

Shiwei Li [1 *] Wenchao Xu [2] Haozhao Wang [1] Xing Tang [3] Yining Qi [1] Shijie Xu [3] Weihong Luo [3] Yuhua Li [1]
Xiuqiang He [3] Ruixuan Li [1]

Abstract

Federated learning is a promising distributed machine learning paradigm that can effectively exploit large-scale data without exposing users' privacy. However, it may incur significant communication overhead, thereby potentially impairing the training efficiency. To address this challenge, numerous studies suggest binarizing the model updates. Nonetheless, traditional methods usually binarize model updates in a post-training manner, resulting in significant approximation errors and consequent degradation in model accuracy. To this end, we propose **Federated Binarization-Aware Training (FedBAT)**, a novel framework that directly learns binary model updates during the local training process, thus inherently reducing the approximation errors. FedBAT incorporates an innovative binarization operator, along with meticulously designed derivatives to facilitate efficient learning. In addition, we establish theoretical guarantees regarding the convergence of FedBAT. Extensive experiments are conducted on four popular datasets. The results show that FedBAT significantly accelerates the convergence and exceeds the accuracy of baselines by up to 9%, even surpassing that of FedAvg in some cases.

1. Introduction

Federated learning (FL) (McMahan et al., 2017) stands out as a promising distributed machine learning paradigm designed to safeguard data privacy. It enables distributed clients to collaboratively train a global model without sharing their local data, thus can protect user data privacy. In the classic FL system, a central server initially broadcasts the global model to several clients, who then train it using their respective datasets. After local training, the clients send their model parameters or model updates back to the server, who aggregates them to create a new global model for the next round of training. This process is repeated for several rounds until the global model converges.

Despite FL's success in preserving local data privacy, the iterative transmission of model parameters introduces considerable communication overhead, adversely affecting training efficiency. Specifically, the communication occurs in two stages: the uplink, where clients send model updates to the server, and the downlink, where the server broadcasts the global model to clients. As suggested by Hönig et al. (2022), the uplink typically imposes a tighter bottleneck than the downlink, particularly due to the global mobile upload bandwidth being less than one fourth of the download bandwidth. Therefore, in this paper, we seek to compress the uplink communication, as much research (Reisizadeh et al., 2020; Isik et al., 2023; Tang et al., 2023) has done.

SignSGD (Bernstein et al., 2018) is an effective binarization technique for reducing communication volume. It was originally proposed to communicate only the signs of gradients in distributed training systems. Recently, it has also been naturally applied to binarize model updates in FL (Ferreira et al., 2021). Specifically, it binarizes model updates $\boldsymbol{m} \in \mathbb{R}^d$ as $\hat{\boldsymbol{m}} = \alpha \cdot \bar{\boldsymbol{m}} = \alpha \cdot \text{sign}(\boldsymbol{m})$, where α denotes the magnitude of binary values, termed the step size and typically tuned as a hyperparameter. SignSGD can efficiently compress the uplink communication by a factor of 32. However, the binarized model updates inevitably contains approximation errors in comparison to the original updates, which affects both the convergence speed and the model accuracy.

To improve the performance of SignSGD, subsequent research has mainly explored from two aspects, including error feedback methods (Karimireddy et al., 2019) and stochastic sign methods (Chen et al., 2020; Safaryan & Richtárik, 2021). EF-SignSGD (Karimireddy et al., 2019) aims to compensate for the errors introduced by binariza-

[*]This work was done when Shiwei Li worked as an intern at FiT, Tencent. [1]Huazhong University of Science and Technology, Wuhan, China [2]The Hong Kong Polytechnic University, Hong Kong, China [3]FiT, Tencent, Shenzhen, China. Correspondence to: Haozhao Wang <hz_wang@hust.edu.cn>, Xing Tang <xing.tang@hotmail.com>, Ruixuan Li <rxli@hust.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

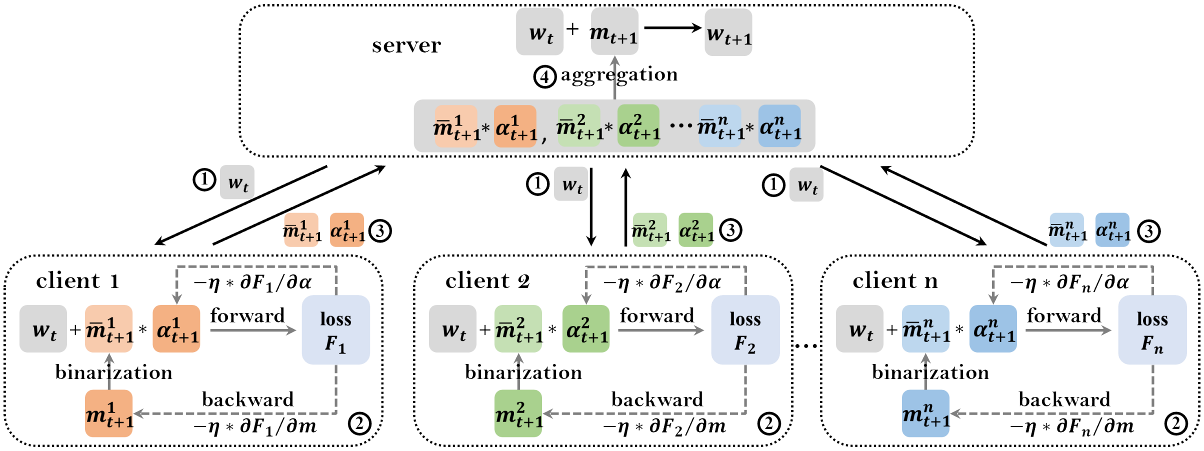

server
$w_t + m_{t+1} \rightarrow w_{t+1}$
④ aggregation
$\bar{m}^1_{t+1} * \alpha^1_{t+1}, \bar{m}^2_{t+1} * \alpha^2_{t+1} \cdots \bar{m}^n_{t+1} * \alpha^n_{t+1}$
① w_t
① w_t
① w_t
$\bar{m}^1_{t+1}$ α^1_{t+1} ③
$\bar{m}^2_{t+1}$ α^2_{t+1} ③
$\bar{m}^n_{t+1}$ α^n_{t+1} ③
client 1
$-\eta * \partial F_1/\partial \alpha$
$w_t + \bar{m}^1_{t+1} * \alpha^1_{t+1}$
forward
loss F_1
binarization
backward
m^1_{t+1}
$-\eta * \partial F_1/\partial m$
②
client 2
$-\eta * \partial F_2/\partial \alpha$
$w_t + \bar{m}^2_{t+1} * \alpha^2_{t+1}$
forward
loss F_2
binarization
backward
m^2_{t+1}
$-\eta * \partial F_2/\partial m$
②
client n
$-\eta * \partial F_n/\partial \alpha$
$w_t + \bar{m}^n_{t+1} * \alpha^n_{t+1}$
forward
loss F_n
binarization
backward
m^n_{t+1}
$-\eta * \partial F_n/\partial m$
②

Figure 1. An illustration of the t-th round within the FedBAT framework. ① downlink: the server sends model parameters $\mathbf{w}_t$ to clients; ② local training: clients train the model updates ($\boldsymbol{m}_{t+1}$ and α_{t+1}) via learnable binarization; ③ uplink: clients upload their binary model updates ($\bar{\boldsymbol{m}}_{t+1}$ and α_{t+1}) to the server. ④ model aggregation: the server aggregates binary model updates to generate $\mathbf{w}_{t+1}$.

tion. Notably, EF-SignSGD adds the binarized errors from the previous round onto the current model updates before conducting another binarization. On the other hand, stochastic sign methods aim to alleviate the bias estimation of SignSGD, namely $\mathbb{E}[\alpha \cdot \text{sign}(\boldsymbol{m})] \neq \boldsymbol{m}$. The key idea here is to perturb local model updates with random noise. Noisy-SignSGD (Chen et al., 2020) adds Gaussian noise $\zeta \sim N(0, \sigma^2)$ to the model update and then performs binarization, where σ is an adjustable standard deviation. Similarly, for a model update $\boldsymbol{m}$, Stoc-SignSGD (Safaryan & Richtárik, 2021) adds uniform noise $\zeta \sim U(-\|\boldsymbol{m}\|, \|\boldsymbol{m}\|)$ to $\boldsymbol{m}$ before binarization. In this way, each element $\boldsymbol{m}_i$ will be binarized into +1 with a probability of $({}^1\!/_2 + {}^{\boldsymbol{m}_i}\!/_{2\|\boldsymbol{m}\|})$ and into -1 with a probability of $({}^1\!/_2 - {}^{\boldsymbol{m}_i}\!/_{2\|\boldsymbol{m}\|})$.

The above research may differ in their motivations and specific solutions, nevertheless, they share a fundamental similarity. Before binarization, a compensation term or disturbance term will be superimposed on the model updates to be binarized, which is the binarized errors from the previous round in EF-SignSGD, Gaussian noise in Noisy-SignSGD, and uniform noise in Stoc-SignSGD. Although applying these methods in FL can indeed enhance the performance of SignSGD (see Section 5.2), it is crucial to acknowledge that they still suffer from two main drawbacks. On one hand, the above methods still binarize model updates in a post-training manner as SignSGD does. Binarized errors are introduced solely after the training phase, depriving them of the opportunity to be optimized during local training. On the other hand, the step size is usually kept as a hyperparameter, necessitating significant human effort to tune it for various applications. However, even with careful tuning, the resulting performance may still fall short of the optimum.

In this paper, we aim to solve these two issues by exploiting the local training process of FL. Specifically, our primary goal is to equip clients with the capability of directly learning binary model updates and the corresponding step size. To this end, we propose a novel training paradigm, termed **Federated Binarization-Aware Training (FedBAT)**. The illustration of each round within FedBAT is presented in Figure 1. The key idea of FedBAT is to binarize the model update with the step size during the forward propagation. This entails calculating the output loss based on the binarized model updates. It provides an opportunity to compute gradients for both the binarized model updates and the step size, facilitating their subsequent optimization. However, the derivative of the vanilla binarization operator is always zero, making the gradient descent algorithm infeasible. To solve this issue, we further introduce a learnable binarization operator with well-designed derivatives.

The main contributions can be summarized as follows:

- We analyze that the drawbacks of existing federated binarization methods lie in their post-training manner. This opens new avenues for binarization in FL.
- Based on our analysis, we propose FedBAT to learn binary model updates during the local training process of FL. For this, a novel binarization operator is employed with well-designed derivatives.
- Theoretically, we establish convergence guarantees for FedBAT, demonstrating a comparable convergence rate to its uncompressed counterpart, the FedAvg.
- Experimentally, we validate FedBAT on four popular datasets. The experimental results show that FedBAT can significantly improve the convergence speed and test accuracy of existing binarization methods, even surpassing the accuracy of FedAvg in some cases.

2. Preliminaries

FL involves N clients connecting to a server. The general goal of FL is to train a global model by multiple rounds of local training on each client's local dataset. Denoting the objective function of the k-th client as F_k, FL can be formulated as

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{k=1}^{N} p_k F_k(\mathbf{w}), \tag{1}$$

where p_k is the proportion of the k-th client's data to all the data of the N clients. FedAvg (McMahan et al., 2017) is a widely used FL algorithm. In the t-th round, the server sends the model parameters $\mathbf{w}_t$ to several randomly selected K clients, denoted as $\mathbb{S}_t$. Each selected client performs certain steps of local training and sends the model update $\mathbf{w}_{t+1}^k - \mathbf{w}_t$ back to the server. The server aggregates these updates to generate a new global model as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \sum_{k \in \mathbb{S}_t} p'_k (\mathbf{w}_{t+1}^k - \mathbf{w}_t), \tag{2}$$

where $p'_k = {}^{p_k}/_{\sum_{\mathbb{S}_t} p_k}$ denotes the proportion of the k-th client's data to all the data used in the t-th round.

In this paper, we employ SignSGD and its aforementioned variants to compress the uplink communication in FL. Each client only needs to send the signs of its model updates to the server. Note that the signs undergo a encoding $\{-1, 1\} \rightarrow \{0, 1\}$ before transmission, and are decoded back after transmission. For simplicity, we omit this process in the following pages. Therefore, Eq.2 can be rewritten as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \sum_{k \in \mathbb{S}_t} p'_k \alpha \cdot \text{sign}(\mathbf{w}_{t+1}^k - \mathbf{w}_t). \tag{3}$$

3. Methodology

3.1. Motivations and Objectives

In the t-th round, the objective of local training for the k-th client can be formulated as

$$\min_{\boldsymbol{m}_{t+1}^k} F_k(\mathbf{w}_t + \boldsymbol{m}_{t+1}^k), \tag{4}$$

where F_k is the loss function of the k-th client. $\mathbf{w}_t$ is the global model parameter in the t-th round and $\boldsymbol{m}_{t+1}^k$ represents the model updates to be learned. Binarization is not considered as a constraint during local training but is only performed after obtaining the final updates $\boldsymbol{m}_{t+1}^k$.

The binarized updates $\hat{\boldsymbol{m}}_{t+1}^k$ inevitably contain errors compared to the original updates $\boldsymbol{m}_{t+1}^k$, thus reducing the model accuracy and slowing down the convergence speed. A natural motivation is to correct or compensate for the binarized errors during the local training process. In light of this, we try to introduce binarization of model updates into local training, thereby directly learning binarized model updates and their corresponding step size. Specifically, the objective of local training can be formulated as

$$\min_{\boldsymbol{m}_{t+1}^k, \alpha_{t+1}^k} F_k(\mathbf{w}_t + \mathcal{S}(\boldsymbol{m}_{t+1}^k, \alpha_{t+1}^k)), \tag{5}$$

where $\mathcal{S}(\boldsymbol{m}_{t+1}^k, \alpha_{t+1}^k)$ are the binary model updates to be learned. $\mathcal{S}$ is a binarization operator. The model updates $\boldsymbol{m}_{t+1}^k$ and the step size α_{t+1}^k are learnable parameters.

Next, we will introduce the learnable binarization operator $\mathcal{S}$ in Section 3.2, and then present the detailed pipeline of FedBAT in Section 3.3. Theoretical guarantees on the convergence of FedBAT will be provided in the next section.

3.2. Learnable Binarization

We first define a binarization operator as follows:

$$\mathcal{S}(x, \alpha) = \begin{cases} \alpha & x > \alpha, \\ \mathcal{S}'(x, \alpha) & -\alpha \leq x \leq \alpha, \\ -\alpha & x < -\alpha, \end{cases} \tag{6}$$

where $\mathcal{S}'(x, \alpha)$ is a stochastic binarization with uniform noise $\zeta \sim U(0, 1)$ added as follows:

$$\begin{aligned} \mathcal{S}'(x, \alpha) &= \alpha(2\lfloor {}^{\alpha+x}/_{2\alpha} + \zeta \rfloor - 1) \\ &= \begin{cases} \alpha & \text{w.p. } {}^{\alpha+x}/_{2\alpha}, \\ -\alpha & \text{w.p. } {}^{\alpha-x}/_{2\alpha}. \end{cases} \end{aligned} \tag{7}$$

However, the floor function $\lfloor x \rfloor$ returns the largest integer not exceeding x, and its derivative is always zero, which makes gradient descent infeasible. To solve this issue, we adopt Straight-through Estimator (STE) (Hinton, 2012) to treat the floor function as an identity map during backpropagation, that is to say its derivative is equal to 1. Recent work has demonstrated that STE works as a first-order approximation of the gradient and affirmed its efficacy (Liu et al., 2023). Therefore, the gradient of x can be estimated by:

$${}^{\partial \mathcal{S}}/_{\partial x} = \begin{cases} 0 & x > \alpha, \\ 1 & -\alpha \leq x \leq \alpha, \\ 0 & x < -\alpha, \end{cases} \tag{8}$$

and the gradient of α can be estimated by:

$${}^{\partial \mathcal{S}}/_{\partial \alpha} = \begin{cases} 1 & x > \alpha, \\ 2\lfloor {}^{\alpha+x}/_{2\alpha} + \zeta \rfloor - {}^{x+\alpha}/_{\alpha} & -\alpha \leq x \leq \alpha, \\ -1 & x < -\alpha, \end{cases} \tag{9}$$

It is worth noting that α represents the step size and is supposed to be a positive number, otherwise the meaning of Eq.6 will be wrong. To restrict the value of α to be positive, we calculate α as follows:

$$\alpha = \alpha' e^{\rho \alpha_e}, \tag{10}$$

where α_e becomes the learnable step size, initialized to zero. α' is the initial value of α and ρ is a hyperparameter that regulates the pace of the optimization process for α_e.

So far, we have developed a derivable binarization operator, which can be seamlessly integrated into a neural network. Let $\mathbf{x}$ denotes a layer of parameters within the network, and α denotes the corresponding step size calculated by Eq.10. During forward propagation, $\mathbf{x}$ will undergo element-wise binarization using Eq.6, where α is shared by all elements in $\mathbf{x}$. During backpropagation, $\mathbf{x}$ and α will be optimized by the gradients calculated from Eq.8 and Eq.9.

3.3. Federated Binarization-Aware Training

Here, we integrate the learnable binarization operator into local training of FL. In contrast to SignSGD, this allows clients to consider binarized errors and make corrections when optimizing locally. We refer to the proposed training framework as Federated Binarization-Aware Training (FedBAT), and the pipeline is shown in Algorithm 1.

In FedBAT, the operating procedures of the server is the same as that in FedAvg. As described in lines 4-8, at the beginning of each round, the server sends global model parameters to several randomly selected clients. It subsequently collects the binarized model updates returned by the clients to calculate new global model parameters as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \sum_{k \in \mathbb{S}_t} p'_k \hat{\boldsymbol{m}}^k_{t+1} = \mathbf{w}_t + \sum_{k \in \mathbb{S}_t} p'_k \alpha^k_{t+1} \bar{\boldsymbol{m}}^k_{t+1}, \tag{11}$$

where $\hat{\boldsymbol{m}}^k_{t+1}$ and α^k_{t+1} are the binarized model updates and the step size of the k-th client. Note that the model update of each layer within the model has a unique step size, thus, Eq.11 is actually performed layer-wise. For simplicity, we omit the representations of the different layers.

Before introducing the local training procedures in FedBAT, we highlight the variances in the local model architectures. We reiterate that FedBAT is achieved by performing the binarization operator defined in Eq.6 on the model updates during local training. However, there is no explicit representation of model updates within the vanilla model. Therefore, we allow local models to maintain an extra copy of model parameters to represent model updates, denoted as $\boldsymbol{m}$. In addition, the step size α of the update within each layer is also kept as a learnable parameter.

The local training process comprises two distinct stages: full-precision training and binarization-aware training. In the t-th round, as depicted in lines 11-12, each client loads the global model parameters $\mathbf{w}_t$ and initialize its model updates $\boldsymbol{m}_t$ with zeros. Given that the model updates commence as zeros, learning to binarize them becomes more challenging. Therefore, to secure more precise initialization values for the model updates, we initially conduct training without binarization, that is the full-precision training where model updates will be optimized as follows:

Algorithm 1 Federated Binarization-Aware Training

```
1: Input: the iteration rounds $R$; the local steps $\tau$; the local warm-up ratio $\phi$, the coefficient of the step size $\rho$.
2: function server($R$)
3:   Initialize global model parameters with $\mathbf{w}_0$.
4:   for $t = 0$ to $R - 1$ do
5:     Send global model parameters $\mathbf{w}_t$ to clients.
6:     Get model updates ($\bar{\boldsymbol{m}}_{t+1}$, $\alpha_{t+1}$) from clients.
7:     Aggregate binary model updates by Eq.11.
8:   end for
9: end function
10: function client($\mathbf{w}_t$, $\tau$, $\phi$, $\rho$)
11:   Initialize local model parameters with $\mathbf{w}_t$.
12:   Initialize local model updates $\boldsymbol{m}_t$ with zeros.
13:   for $s = 0$ to $\tau - 1$ do
14:     if $s < \lfloor \phi\tau \rfloor$ then
15:       Run full-precision training by Eq.12.
16:     else if $s = \lfloor \phi\tau \rfloor$ then
17:       Initialize the step size layer-wise by Eq.13.
18:       Run binarization-aware training by Eq.14.
19:     else
20:       Run binarization-aware training by Eq.14.
21:     end if
22:   end for
23:   $\bar{\boldsymbol{m}}^k_{t+1}$, $\alpha^k_{t+1} = \bar{\boldsymbol{m}}^k_{t,\tau-1}$, $\alpha^k_{t,\tau-1}$.
24:   Send binary model updates ($\bar{\boldsymbol{m}}^k_{t+1}$, $\alpha^k_{t+1}$) to server.
25: end function
```

$$\boldsymbol{m}^k_{t,s+1} = \boldsymbol{m}^k_{t,s} - \eta_t {\partial F_k(\mathbf{w}^k_t + \boldsymbol{m}^k_{t,s})}/{\partial \boldsymbol{m}^k_{t,s}}. \tag{12}$$

A warm-up ratio ϕ is defined to denote the proportion of full-precision training to the entire local training. After the full-precision training, α' and α_e defined in Eq.10 will be initialized for each layer of the k-th client as follows:

$$(\alpha')^k_l = {\|(\boldsymbol{m})^k_l\|_1}/{d^k_l}, \quad (\alpha_e)^k_l = 0, \tag{13}$$

where $(\boldsymbol{m})^k_l \in \mathbb{R}^{d^k_l}$ represents the model update of the l-th layer in the k-th client. Next, FedBAT performs binarization-aware training, optimizing the step size and the binarized model update using Eq.8 and Eq.9 as follows:

$$\begin{aligned} \boldsymbol{m}^k_{t,s+1} &= \boldsymbol{m}^k_{t,s} - \eta_t {\partial F_k(\mathbf{w}^k_t + \hat{\boldsymbol{m}}^k_{t,s})}/{\partial \boldsymbol{m}^k_{t,s}}, \\ \alpha^k_{t,s+1} &= \alpha^k_{t,s} - \eta_t {\partial F_k(\mathbf{w}^k_t + \hat{\boldsymbol{m}}^k_{t,s})}/{\partial \alpha^k_{t,s}}, \\ \hat{\boldsymbol{m}}^k_{t,s+1} &= \mathcal{S}(\boldsymbol{m}^k_{t,s+1}, \alpha^k_{t,s+1}). \end{aligned} \tag{14}$$

After local training, the k-th client will send its binarized model update $\bar{\boldsymbol{m}}^k_{t+1}$ and the step size α^k_{t+1} to the server.

4. Convergence Analysis

In this section, we provide theoretical guarantees for the convergence of FedBAT, while considering the data heterogeneity within FL. For simplicity, we focus on the case where the warm-up ratio ϕ is zero, meaning that the warm-up training, as defined by Eq.12, is not performed. Furthermore, to ensure the unbiased property of the binarization operator $\mathcal{S}$ in Eq.6, we set the step size as $\alpha_{t,s}^k = \|\boldsymbol{m}_{t,s}^k\|_\infty$ in each step instead of optimizing it during local training. Then we give the following notations and assumptions.

Notations. Let F^* and F_k^* be the minimum values of F and F_k, respectively, then $\Gamma = F^* - \sum_{k=1}^N p_k F_k^*$ can be used to quantify the degree of data heterogeneity. τ is the number of local steps. In Section 3.3, the subscripts $t \in [R]$ and $s \in [\tau]$ are used to represent the serial number of global rounds and local iterations, respectively. In the following analysis, we will only use the subscript t to represent the cumulative number of iteration steps in the sense that $t \in [T], T = R\tau$.

Assumption 1. *$F_1, ..., F_N$ are all L-smooth: for $\mathbf{w}$ and $\mathbf{v}$, $F_k(\mathbf{v}) \le F_k(\mathbf{w}) + (\mathbf{v}-\mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{L}{2}\|\mathbf{v}-\mathbf{w}\|^2$.*

Assumption 2. *$F_1, ..., F_N$ are u-strongly convex: for all $\mathbf{w}$ and $\mathbf{v}$, $F_k(\mathbf{v}) \ge F_k(\mathbf{w}) + (\mathbf{v}-\mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{v}-\mathbf{w}\|^2$.*

Assumption 3. *Let ξ_t^k be sampled from the k-th client's local data uniformly at random. The variance of stochastic gradients in each device is bounded: $\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k) - \nabla F_k(\mathbf{w}_t^k)\|^2 \le \sigma^2$ for all $k = 1, ..., N$.*

Assumption 4. *The expected squared norm of stochastic gradients is uniformly bounded, i.e., $\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k)\|^2 \le G^2$ for all $k = 1, ..., N$ and $t = 1, ..., T$.*

Assumption 5. *The variance of binarization $\mathcal{S}$ grows with the l_2-norm of its argument, i.e., $\mathbb{E}\|\mathcal{S}(\mathbf{x}) - \mathbf{x}\| \le q\|\mathbf{x}\|$.*

Assumptions 1-4 are commonplace in standard optimization analyses (Stich et al., 2018; Yu et al., 2019; Li et al., 2020). The condition in Assumption 5 is satisfied with many compression schemes including the binarization operator $\mathcal{S}$ as defined in Eq.6. Assumption 5 is also used in (Karimireddy et al., 2019; Reisizadeh et al., 2020) to analyze the convergence of federated algorithms. Theorems 1 and 2 show the convergence of FedBAT under the strongly convex assumption with full and partial device participation, respectively. The convergence of FedBAT under the non-convex assumption with partial device participation is shown in Theorems 3. All proofs are provided in Appendix.

Theorem 1. *Let Assumptions 1-5 hold and L, μ, σ, G, q be defined therein. Choose $\kappa = \frac{L}{\mu}, \gamma = \max\{8\kappa, \tau\} - 1$ and the learning rate $\eta_t = \frac{2}{\mu(\gamma+t)}$. Then FedBAT with full device participation satisfies*

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \le \frac{\kappa}{\gamma + T}\left(\frac{2B}{\mu} + \frac{\mu(\gamma+1)}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right), \tag{15}$$

where $B = \sum_{k=1}^N p_k^2 \sigma^2 + 6L\Gamma + 8(1+q^2)(\tau-1)^2 G^2 + 4\sum_{k=1}^N p_k^2 q^2 \tau^2 G^2$.

Theorem 2. *Let Assumptions 1-5 hold and L, μ, σ, G, q be defined therein. Let κ, γ, η_t be defined in Theorem 1. Assuming that K devices are randomly selected to participate in each round of training and their data is balanced in the sense that $p_1 = ... = p_N = \frac{1}{N}$. Then the same bound in Theorem 1 holds if we redefine the value of B to $B = \frac{\sigma^2}{N} + 6L\Gamma + 8(1+q^2)(\tau-1)^2 G^2 + 4\frac{q^2(N-1)+N-K}{K(N-1)}\tau^2 G^2$.*

Remark 1. *By setting $K = N$, Theorem 2 transforms into Theorem 1. By setting $q = 0$, Theorem 1 and 2 are equivalent to the analysis of FedAvg in (Li et al., 2020). By setting $K = N$ and $\tau = 1$, Theorem 1 and 2 recovers the convergence rate of Stoc-SignSGD (Safaryan & Richtárik, 2021) when used in distributed training. By setting $K = N, \tau = 1$ and $q = 0$, Theorem 1 and 2 recovers the convergence rate of vanilla SGD, i.e., $\mathcal{O}(\frac{1}{T})$ for strongly-convex losses.*

Theorem 3. *Let Assumptions 1 and 3-5 hold, i.e., without the convex assumption, and L, σ, G, q be defined therein. Assume the learning rate is set to $\eta = \frac{1}{L\sqrt{T}}$ and the local dataset is balanced, then the following first-order stationary condition holds for FedBAT with partial device participation*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\mathbf{w}_t)\|^2 \le \frac{2L(F(\mathbf{w}_0) - F^* + \Gamma)}{\sqrt{T}} + \frac{P}{\sqrt{T}} + \frac{Q}{T}, \tag{16}$$

where $P = \frac{\sigma^2}{N} + 4\frac{q^2(N-1)+N-K}{K(N-1)}\tau^2 G^2$ and $Q = 4(1+q^2)(\tau-1)^2 G^2$.

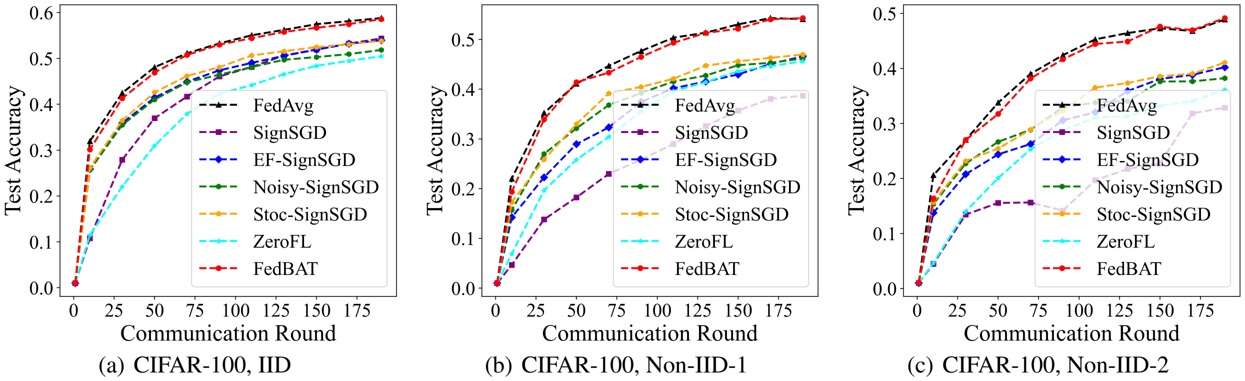
Remark 2. *Under the conditions of Theorems 1-3, the convergence rate of both FedBAT and FedAvg ($q = 0$) is $\mathcal{O}(\frac{1}{T})$ in the strongly convex setting, and $\mathcal{O}(\frac{1}{T}) + \mathcal{O}(\frac{1}{\sqrt{T}})$ in the non-convex setting.*

Remark 3. *For ease of analysis, FedBAT is discussed in the case where the step size is set as $\alpha_i^{t,s} = \|\boldsymbol{m}_i^{t,s}\|_\infty$ without optimization. However, learning the step size shall be able to achieve smaller value of q and enhance the performance of FedBAT. Empirically, we show in Section 5 that optimizing the step size as defined in Eq.13-14 achieves better accuracy.*

5. Experiments

5.1. Experimental Setup

Datasets and Models. In this section, we evaluate FedBAT using four widely recognized datasets: FMNIST (Xiao et al., 2017), SVHN (Netzer et al., 2011), CIFAR-10 and CIFAR-100 (Krizhevsky & Hinton, 2009). To showcase the versatility of FedBAT, we also assess its performance across different model architectures. Specifically, we employ a CNN with four convolution layers and one fully connected layer for FMNIST and SVHN, and ResNet-10 (He et al., 2016) for CIFAR-10 and CIFAR-100. Detailed model architectures are available in Appendix A.1.



(a) CIFAR-100, IID (b) CIFAR-100, Non-IID-1 (c) CIFAR-100, Non-IID-2

Figure 2. Convergence curves of FedBAT and baselines on CIFAR-100 with 100 clients.

Data Partitioning. We consider both cases of IID and Non-IID data distribution, referring to the data partitioning benchmark of FL (Li et al., 2022). Under IID partitioning, an equal quantity of data is randomly sampled for each client. The Non-IID scenario further encompasses two distinct label distributions, termed Non-IID-1 and Non-IID-2. In Non-IID-1, the proportion of the same label among clients follows the Dirichlet distribution (Yurochkin et al., 2019), while in Non-IID-2, each client only contains data of partial labels. For CIFAR-100, we set the Dirichlet parameter to 0.1 in Non-IID-1 and assign 10 random labels to each client in Non-IID-2. For the other datasets, we set the Dirichlet parameter to 0.3 in Non-IID-1 and assign 3 random labels to each client in Non-IID-2.

Baseline Methods. All experiments are conducted on Flower (Beutel et al., 2020), an open-source training platform for FL. FedAvg (McMahan et al., 2017) is adopted as the backbone training algorithm. We compare FedBAT with the binarization methods discussed in Section 1, including SignSGD (Bernstein et al., 2018), EF-SignSGD (Karimireddy et al., 2019), Noisy-SignSGD (Chen et al., 2020) and Stoc-SignSGD (Safaryan & Richtárik, 2021). We also compare FedBAT with ZeroFL (Qiu et al., 2022), a method that exploits local sparsity to compress communications. To ensure similar traffic volume to the binarization methods, we set the sparsity ratio of ZeroFL to 97%. Details about the baselines are provided in Appendix A.2.

Hyperparameters. The number of clients is set to 30 and 100, respectively. 10 clients will participate in every round. The local epoch is set to 10 and the batch size is set to 64. SGD (Bottou, 2010) is used as the local optimizer. The learning rate is tuned from (1.0, 0.1, 0.01) and set to 0.1. The number of rounds are set to 100 for CNN and 200 for ResNet-10. For the baselines, each hyperparameter is carefully tuned among (1.0, 0.1, 0.01, 0.001), including the step size and the coefficient of noise. Detailed tuning process and results are provided in Appendix A.2. In FedBAT, the coefficient ρ is set to 6 and the warm-up ratio ϕ is set to 0.5 by default. Each experiment is run five times on Nvidia 3090 GPUs with Intel Xeon E5-2673 CPUs. Average results and the standard deviation are reported.

5.2. Overall Performance

In this subsection, we compare the performance of FedBAT and the baselines by the test accuracy and the convergence speed. All numerical results are reported in Table 1. The convergence curves on CIFAR-100 with 100 clients are shown in Figure 2. The convergence curves on other datasets are provided in Appendix B.3. In addition, experimental results of more clients and more related baselines are also available in Appendix B.

As shown in Table 1, compared with SignSGD, other binarization baselines can indeed improve the test accuracy in most cases. However, in a few cases, such as CIFAR-10 under the Non-IID-1 data distribution, their accuracy can be even worse than SignSGD, which reveals the limitations of existing binarization methods. For ZeroFL, we observe a generally lower accuracy compared to the binarization methods. This discrepancy can be attributed to the higher sparsity ratio, which hinders the update of most parameters. In addition, all baselines suffer significant accuracy loss compared to FedAvg, particularly in scenarios involving high degrees of Non-IID data distribution. On the contrary, FedBAT can generally achieve comparable or even higher accuracy than FedAvg, irrespective of data distribution or the number of clients. In terms of convergence speed, as shown in the Figure 2, FedBAT can consistently outperform all binarization baselines and approach that of FedAvg.

5.3. Ablation Studies

In this subsection, we conduct ablation studies to assess the feasibility and superiority of FedBAT's design. Specifically, we adjust the coefficient of the step size ρ and the

Table 1. The test accuracy of all methods on four datasets. The best accuracy is bolded and the next best accuracy is underlined.

	$N = 30$			$N = 100$		
	IID	Non-IID-1	Non-IID-2	IID	Non-IID-1	Non-IID-2
FMNIST with CNN						
FedAvg (McMahan et al., 2017)	**92.5** (± 0.1)	90.7 (± 0.1)	88.9 (± 0.2)	**92.0** (± 0.1)	90.5 (± 0.2)	88.7 (± 0.2)
SignSGD (Bernstein et al., 2018)	91.3 (± 0.1)	86.5 (± 1.0)	78.9 (± 1.3)	90.3 (± 0.1)	88.2 (± 0.2)	80.5 (± 1.0)
EF-SignSGD (Karimireddy et al., 2019)	92.3 (± 0.1)	90.5 (± 0.1)	88.6 (± 0.2)	91.3 (± 0.1)	89.7 (± 0.1)	87.4 (± 0.1)
Noisy-SignSGD (Chen et al., 2020)	92.1 (± 0.1)	90.3 (± 0.2)	87.6 (± 0.3)	91.0 (± 0.1)	89.4 (± 0.1)	86.9 (± 0.1)
Stoc-SignSGD (Safaryan & Richtárik, 2021)	91.7 (± 0.1)	89.5 (± 0.2)	82.6 (± 0.8)	90.6 (± 0.1)	88.5 (± 0.2)	84.8 (± 0.8)
ZeroFL (Qiu et al., 2022)	91.0 (± 0.1)	89.3 (± 0.3)	87.4 (± 0.2)	90.2 (± 0.1)	88.8 (± 0.2)	86.6 (± 0.1)
FedBAT	**92.5** (± 0.1)	**90.8** (± 0.2)	**89.1** (± 0.3)	91.8 (± 0.1)	**90.6** (± 0.1)	**89.0** (± 0.4)
SVHN with CNN						
FedAvg (McMahan et al., 2017)	92.7 (± 0.1)	90.9 (± 0.1)	89.2 (± 0.2)	92.1 (± 0.1)	89.7 (± 0.3)	88.9 (± 0.2)
SignSGD (Bernstein et al., 2018)	92.3 (± 0.1)	80.1 (± 1.5)	66.1 (± 1.3)	90.7 (± 0.1)	80.4 (± 1.4)	64.7 (± 1.9)
EF-SignSGD (Karimireddy et al., 2019)	92.6 (± 0.2)	90.7 (± 0.1)	88.3 (± 0.1)	91.8 (± 0.1)	89.2 (± 0.2)	86.8 (± 0.4)
Noisy-SignSGD (Chen et al., 2020)	92.2 (± 0.1)	90.3 (± 0.2)	88.2 (± 0.1)	90.9 (± 0.2)	88.7 (± 0.1)	86.9 (± 0.2)
Stoc-SignSGD (Safaryan & Richtárik, 2021)	92.3 (± 0.1)	88.0 (± 0.6)	86.7 (± 0.8)	90.7 (± 0.2)	87.7 (± 0.3)	84.8 (± 0.2)
ZeroFL (Qiu et al., 2022)	91.7 (± 0.1)	90.2 (± 0.1)	87.6 (± 0.3)	90.3 (± 0.1)	88.4 (± 0.3)	87.0 (± 0.2)
FedBAT	**92.9** (± 0.1)	**91.1** (± 0.1)	**89.3** (± 0.2)	**92.5** (± 0.1)	**90.7** (± 0.1)	**89.2** (± 0.2)
CIFAR-10 with ResNet-10						
FedAvg (McMahan et al., 2017)	**91.5** (± 0.1)	**89.0** (± 0.2)	**83.7** (± 0.4)	**89.3** (± 0.1)	84.6 (± 0.2)	80.9 (± 0.5)
SignSGD (Bernstein et al., 2018)	88.9 (± 0.1)	87.8 (± 0.2)	76.1 (± 1.6)	87.3 (± 0.2)	82.2 (± 0.4)	76.6 (± 0.9)
EF-SignSGD (Karimireddy et al., 2019)	90.8 (± 0.1)	87.4 (± 0.2)	78.3 (± 0.9)	87.4 (± 0.2)	81.8 (± 0.5)	76.6 (± 0.9)
Noisy-SignSGD (Chen et al., 2020)	90.1 (± 0.2)	86.2 (± 0.1)	78.3 (± 0.7)	85.5 (± 0.2)	80.2 (± 0.3)	72.7 (± 0.4)
Stoc-SignSGD (Safaryan & Richtárik, 2021)	88.7 (± 0.2)	86.2 (± 0.2)	77.8 (± 0.4)	85.9 (± 0.2)	80.5 (± 0.3)	74.1 (± 1.0)
ZeroFL (Qiu et al., 2022)	89.0 (± 0.1)	86.3 (± 0.1)	79.0 (± 0.6)	85.2 (± 0.2)	78.4 (± 0.2)	73.8 (± 0.6)
FedBAT	91.2 (± 0.1)	88.6 (± 0.1)	82.8 (± 0.1)	89.2 (± 0.2)	**84.9** (± 0.3)	**81.0** (± 0.6)
CIFAR-100 with ResNet-10						
FedAvg (McMahan et al., 2017)	**67.7** (± 0.2)	**64.1** (± 0.3)	**54.5** (± 0.4)	**59.2** (± 0.3)	**55.4** (± 0.8)	49.0 (± 0.6)
SignSGD (Bernstein et al., 2018)	58.9 (± 0.6)	53.9 (± 0.2)	34.3 (± 1.5)	54.2 (± 0.3)	39.3 (± 0.4)	32.5 (± 1.7)
EF-SignSGD (Karimireddy et al., 2019)	65.6 (± 0.2)	59.7 (± 0.4)	50.7 (± 0.4)	53.8 (± 0.5)	46.8 (± 0.5)	40.2 (± 0.6)
Noisy-SignSGD (Chen et al., 2020)	65.3 (± 0.2)	58.3 (± 0.2)	46.6 (± 0.2)	52.6 (± 0.6)	46.2 (± 0.5)	38.3 (± 0.3)
Stoc-SignSGD (Safaryan & Richtárik, 2021)	61.1 (± 0.4)	57.8 (± 0.4)	46.2 (± 0.7)	54.2 (± 0.2)	47.2 (± 0.3)	40.1 (± 0.6)
ZeroFL (Qiu et al., 2022)	63.7 (± 0.2)	59.9 (± 0.5)	47.7 (± 0.8)	50.5 (± 0.5)	45.6 (± 0.4)	36.1 (± 0.5)
FedBAT	66.3 (± 0.1)	63.9 (± 0.4)	53.9 (± 0.3)	58.6 (± 0.3)	54.3 (± 0.4)	**49.2** (± 0.6)

Table 2. The test accuracy of FedBAT with varying ρ.

	$\rho = 0$	$\rho = 2$	$\rho = 4$	$\rho = 6$	$\rho = 8$	$\rho = 10$
FMNIST	87.9	88.2	88.9	**89.0**	88.7	88.9
SVHN	88.5	89.0	89.2	89.2	**89.5**	88.7
CIFAR-10	78.3	79.0	80.4	**81.0**	80.6	80.1
CIFAR-100	41.2	48.0	48.7	**49.2**	48.8	48.7

Table 3. The test accuracy of FedBAT with varying ϕ.

	$\phi = 0.1$	$\phi = 0.3$	$\phi = 0.5$	$\phi = 0.7$	$\phi = 0.9$
FMNIST	88.6	88.7	**89.0**	**89.0**	88.5
SVHN	88.4	88.8	**89.2**	88.5	88.6
CIFAR-10	75.9	79.9	**81.0**	80.7	80.3
CIFAR-100	42.9	48.6	**49.2**	48.9	48.3

local warm-up ratio ϕ in the context of the Non-IID-2 data distribution with 100 clients. We show in the subsequent experiments that FedBAT also outperforms the baseline methods in terms of hyperparameter tuning.

We first explore the FedBAT variant with $\rho = 0$, where the step size α remains fixed at its initial value α' without undergoing any optimization. Comparing Table 1 and Table 2, it can be found that FedBAT ($\rho = 0$) still achieves better accuracy than all binarization baselines, which proves the superiority of binarizing model updates during local training. Nevertheless, the accuracy of FedBAT ($\rho = 0$) remains inferior to FedAvg, which underscores the need for an adaptive step size. Therefore, we further tune ρ among $\{2, 4, 6, 8, 10\}$ to verify the effect and robustness of learning the step size α. As illustrated in Table 2, setting the parameter ρ to 2 enhances the accuracy of FedBAT, aligning it more closely with FedAvg. The gradual increase in the value of ρ leads to a slight continuous improvement in the accuracy of FedBAT until ρ reaches 10. It is noteworthy that for ρ values of 4, 6, and 8, the accuracy difference in FedBAT is comparatively minimal. This indicates a broad optimal range for the hyperparameter ρ, highlighting the robustness of FedBAT to variations in ρ.

Another hyperparameter of FedBAT is the local warm-up ratio ϕ, which balances the trade-off between the binarization of model updates and their initialization. Here, we tune ϕ among $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. As shown in the Table 3, the optimal accuracy for FedBAT is always achieved when ϕ is set to 0.5, indicating an equilibrium between the significance of initializing and binarizing model updates. For ϕ values of 0.3 or 0.7, there is a marginal reduction in accuracy, although the variance in accuracy is minimal. Notably, further deviation by increasing ϕ to 0.9 or decreasing it to 0.1 leads to more decline in the accuracy.

Following the above ablation studies, we summarize the essential findings regarding hyperparameter tuning about FedBAT. It is advisable to set the local warm-up ratio ϕ to 0.5 as a default configuration without necessitating hyperparameter tuning. The default setting for ρ in FedBAT is recommended to be 6, with an option to tune it within the interval of [4,8] for marginal accuracy enhancements.

6. Related Work

6.1. Communication-Efficient Federated Learning

Existing methods reduce communication costs in FL from two aspects: model compression and gradient compression. The proposed FedBAT belongs to the latter paradigm.

Frequently transferring large models between the server and clients is a significant burden, especially for clients with limited communication bandwidth. Therefore, researchers have turned to model compression techniques. Yang et al. (2021) train and communicate a binary neural network in FL. Different from their motivation, our focus lies in learning binary model updates rather than binary weights. (Caldas et al., 2018; Bouacida et al., 2021) enable clients to train randomly selected sub-models of a larger server model. Hyeon-Woo et al. (2022) use matrix factorization to reduce the size of the model that needs to be transferred. (Li et al., 2021; Isik et al., 2023) transfer a pruned model for efficient communication. Specifically, FedPM (Isik et al., 2023) trains and communicates only a binary mask for each model parameter, while keeping the model parameters at their randomly initialized values. It is worth noting that FedPM has certain similarities with FedBAT. They both keep the base model parameters fixed during local training. FedPM trains binary masks to prune the base model, while FedBAT trains binary model updates with respect to the base model. The model trained by FedPM enjoys the advantage of sparsity, however, the randomly initialized model parameters do not undergo any updates. Due to the lack of effective training on randomly initialized parameters, FedPM may not be able to achieve a satisfactory accuracy as FedAvg (Vallapuram et al., 2022).

Apart from the model compression, another way to reduce communication costs is gradient compression. It is generally achieved by pruning (Qiu et al., 2022) or quantization (including binarization) (Reisizadeh et al., 2020; Bernstein et al., 2018) on the model updates. To enhance efficiency, recent quantization methods (Jhunjhunwala et al., 2021; Qu et al., 2022; Hönig et al., 2022) try to adjust the quantization bitwidth used in different rounds adaptively. However, they suffer from the same post-training manner as existing binarization methods. We posit that the concept of FedBAT can be seamlessly applied to federated quantization.

6.2. Binarization

BinaryConnect (Courbariaux et al., 2015) is a pioneering work on learning binary weights. It binarizes weights into $\{-1, +1\}$ to calculate the output loss. During backpropagation, the full precision weights are optimized by STE (Hinton, 2012). However, BinaryConnect does not involve learning the step size. Further, BWN and XNOR-Net are proposed to calculate a step size by minimizing the binarized errors (Rastegari et al., 2016). To be specific, the step size is set to the average of absolute weight values. Later, Hou et al. (2017) propose a proximal Newton algorithm with diagonal Hessian approximation that directly minimizes the loss with respect to the binarized weights and the step size. However, it requires the second-order gradient information, usually not available in SGD. In addition, considering that binarization is essentially a form of 1-bit quantization, many quantization-aware training methods are also suitable for binarization. For example, PACT (Choi et al., 2018) and LSQ (Esser et al., 2020) achieve end-to-end optimization by designing reasonable gradient for the step size.

In this paper, we discovered the post-training manner of model updates compression in FL. We seek to solve this problem by directly learning binary model updates during local training. However, the above methods are designed to learn binarized or quantized weights by a long period of centralized training. It is difficult for them to learn accurate binary model updates during a short local training period. Therefore, after designing the gradient of the step size, we also introduced a temperature ρ to regulate the pace of its optimization, along with the implementation of warm-up training for improved initialization.

7. Conclusion

We analyzed the challenges faced by existing binarization methods when applied in the context of FL. The analysis encourages us to leverage the local training process to learn binary model updates, instead of binarizing them after training. Therefore, we propose FedBAT, a federated training framework designed with a focus on binarization awareness. FedBAT has undergone comprehensive theoretical examination and experimental validation. It is able to exceed the binarization baselines in terms of the test accuracy and convergence speed, and is comparable to FedAvg.

An inherent limitation of FedBAT resides in the requirement for the client to retain two distinct copies of model parameters: one for the initialization model and another for model updates. Although only one copy of parameters shall be trained, this slightly increases the memory overhead of the local training process in FedBAT. However, by compressing the initialization model transmitted from the server before communication, the memory overhead in FedBAT can be alleviated, as well as the downlink communication can be further compressed. We leave this as future work.

Acknowledgements

This work is supported in part by National Natural Science Foundation of China under grants 62376103, 62302184, 62206102, Science and Technology Support Program of Hubei Province under grant 2022BAA046, and Ant Group through CCF-Ant Research Fund.

Impact Statement

In this paper, we propose FedBAT to improve the communication efficiency of federated learning. It is our contention that FedBAT does not result in any adverse social impact. Instead, FedBAT enhances user privacy protection to a certain extent as the model updates uploaded by the clients are binarized and noise is introduced in the process.

References

Bernstein, J., Wang, Y., Azizzadenesheli, K., and Anandkumar, A. SIGNSGD: compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, pp. 559–568. PMLR, 2018.

Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Parcollet, T., and Lane, N. D. Flower: A friendly federated learning research framework. *CoRR*, abs/2007.14390, 2020.

Bottou, L. Large-scale machine learning with stochastic gradient descent. In *19th International Conference on Computational Statistics, COMPSTAT*, pp. 177–186. Physica-Verlag, 2010.

Bouacida, N., Hou, J., Zang, H., and Liu, X. Adaptive federated dropout: Improving communication efficiency and generalization for federated learning. In *2021 IEEE Conference on Computer Communications Workshops, INFOCOM Workshops*, pp. 1–6. IEEE, 2021.

Caldas, S., Konečný, J., McMahan, H. B., and Talwalkar, A. Expanding the reach of federated learning by reducing client resource requirements. *CoRR*, abs/1812.07210, 2018.

Chen, X., Chen, T., Sun, H., Wu, Z. S., and Hong, M. Distributed training with heterogeneous data: Bridging median- and mean-based algorithms. In *Advances in Neural Information Processing Systems, NeurIPS*, volume 33, pp. 21616–21626, 2020.

Choi, J., Wang, Z., Venkataramani, S., Chuang, P. I., Srinivasan, V., and Gopalakrishnan, K. PACT: parameterized clipping activation for quantized neural networks. *CoRR*, abs/1805.06085, 2018.

Courbariaux, M., Bengio, Y., and David, J.-P. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Esser, S. K., McKinstry, J. L., Bablani, D., Appuswamy, R., and Modha, D. S. Learned step size quantization. In *8th International Conference on Learning Representations, ICLR*. OpenReview.net, 2020.

Ferreira, P. A., da Silva, P. N., Gottin, V., Stelling, R., and Calmon, T. Bayesian signsgd optimizer for federated learning. In *Advances in Neural Information Processing Systems, NeurIPS*, 2021.

Glorot, X., Bordes, A., and Bengio, Y. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS*, pp. 315–323, 2011.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 770–778. IEEE Computer Society, 2016.

Hinton, G. Neural networks for machine learning. *Coursera video lectures*, 2012.

Hönig, R., Zhao, Y., and Mullins, R. D. Dadaquant: Doubly-adaptive quantization for communication-efficient federated learning. In *International Conference on Machine Learning, ICML*, volume 162, pp. 8852–8866. PMLR, 2022.

Hou, L., Yao, Q., and Kwok, J. T. Loss-aware binarization of deep networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Hyeon-Woo, N., Ye-Bin, M., and Oh, T. Fedpara: Low-rank hadamard product for communication-efficient federated learning. In *The Tenth International Conference on Learning Representations, ICLR*. OpenReview.net, 2022.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML*, pp. 448–456, 2015.

Isik, B., Pase, F., Gündüz, D., Weissman, T., and Zorzi, M. Sparse random networks for communication-efficient federated learning. In *The Eleventh International Conference on Learning Representations, ICLR*. OpenReview.net, 2023.

Jhunjhunwala, D., Gadhikar, A., Joshi, G., and Eldar, Y. C. Adaptive quantization of model updates for communication-efficient federated learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pp. 3110–3114. IEEE, 2021.

Karimireddy, S. P., Rebjock, Q., Stich, S. U., and Jaggi, M. Error feedback fixes signsgd and other gradient compression schemes. In *Proceedings of the 36th International Conference on Machine Learning,ICML*, volume 97, pp. 3252–3261. PMLR, 2019.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. *Technical report, University of Toronto*, 2009.

Li, A., Sun, J., Zeng, X., Zhang, M., Li, H., and Chen, Y. Fedmask: Joint computation and communication-efficient personalized federated learning via heterogeneous masking. In *SenSys '21: The 19th ACM Conference on Embedded Networked Sensor Systems*, pp. 42–55. ACM, 2021.

Li, Q., Diao, Y., Chen, Q., and He, B. Federated learning on non-iid data silos: An experimental study. In *38th IEEE International Conference on Data Engineering, ICDE*, pp. 965–978. IEEE, 2022.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of fedavg on non-iid data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Liu, L., Dong, C., Liu, X., Yu, B., and Gao, J. Bridging discrete and backpropagation: Straight-through and beyond. *CoRR*, abs/2304.08612, 2023.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 54, pp. 1273–1282. PMLR, 2017.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Qiu, X., Fernández-Marqués, J., de Gusmao, P. P. B., Gao, Y., Parcollet, T., and Lane, N. D. Zerofl: Efficient on-device training for federated learning with local sparsity. In *The Tenth International Conference on Learning Representations, ICLR*. OpenReview.net, 2022.

Qu, L., Song, S., and Tsui, C. Feddq: Communication-efficient federated learning with descending quantization. In *IEEE Global Communications Conference, GLOBECOM*, pp. 281–286. IEEE, 2022.

Raihan, M. A. and Aamodt, T. M. Sparse weight activation training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 2020.

Rastegari, M., Ordonez, V., Redmon, J., and Farhadi, A. Xnor-net: Imagenet classification using binary convolutional neural networks. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, pp. 525–542, 2016.

Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., and Pedarsani, R. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 108, pp. 2021–2031. PMLR, 2020.

Safaryan, M. and Richtárik, P. Stochastic sign descent methods: New algorithms and better theory. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139, pp. 9224–9234. PMLR, 2021.

Stich, S. U., Cordonnier, J., and Jaggi, M. Sparsified SGD with memory. In *Advances in Neural Information Processing Systems 31: Annual Conferenceon Neural Information Processing Systems, NeurIPS*, pp. 4452–4463, 2018.

Tang, Z., Wang, Y., and Chang, T. z-signfedavg: A unified stochastic sign-based compression for federated learning. *CoRR*, abs/2302.02589, 2023.

Vallapuram, A. K., Zhou, P., Kwon, Y. D., Lee, L. H., Xu, H., and Hui, P. Hidenseek: Federated lottery ticket via server-side pruning and sign supermask. *CoRR*, abs/2206.04385, 2022.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

Yang, Y., Zhang, Z., and Yang, Q. Communication-efficient federated learning with binary neural networks. *IEEE J. Sel. Areas Commun.*, 39(12):3836–3850, 2021.

Yu, H., Yang, S., and Zhu, S. Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, pp. 5693–5700. AAAI Press, 2019.

Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K. H., Hoang, T. N., and Khazaeni, Y. Bayesian nonparametric federated learning of neural networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, volume 97, pp. 7252–7261. PMLR, 2019.

A. Detailed Experimental Settings

A.1. Model Architectures

In this paper, we employ a CNN for FMNIST and SVHN, ResNet-10 for CIFAR-10 and CIFAR-100. The detailed model architectures are shown in Table 4. The BasicBlock used by ResNet is the same as defined in (He et al., 2016). Batch normalization (BN) (Ioffe & Szegedy, 2015) is used to ensure stable training and ReLU (Glorot et al., 2011) is employed as the activation function.

Table 4. Model architectures of the CNN, ResNet-10.

CNN (FMNIST)	CNN (SVHN)	ResNet-10 (CIFAR-10)	ResNet-10 (CIFAR-100)
Convd2d(1,32,3)	Convd2d(3,32,3)	Convd2d(3,32,3)	Convd2d(3,32,3)
Convd2d(32,64,3)	Convd2d(32,64,3)	BasicBlock(32)	BasicBlock(32)
Convd2d(64,128,3)	Convd2d(64,128,3)	BasicBlock(64)	BasicBlock(64)
Convd2d(128,256,3)	Convd2d(128,256,3)	BasicBlock(128)	BasicBlock(128)
Linear(256,10)	Linear(1024,10)	BasicBlock(256)	BasicBlock(256)
		Linear(256,10)	Linear(256,100)

A.2. Baseline Methods

In our experiments, we compare FedBAT with five baselines, including SignSGD (Bernstein et al., 2018), EF-SignSGD (Karimireddy et al., 2019), Noisy-SignSGD (Chen et al., 2020), Stoc-SignSGD (Safaryan & Richtárik, 2021) and ZeroFL (Qiu et al., 2022). Here, we introduce each baseline and the hyperparameters involved in detail. We declare that all hyperparameters are tuned in $\{1.0, 0.1, 0.01, 0.001\}$ for each dataset. The hyperparameter of SignSGD is the step size α, which is tuned and set to 0.001 for all datasets. In EF-SignSGD, there is no hyperparameter to tune. Notably, EF-SignSGD adds the binarization errors from the previous round onto the current model updates before conducting another binarization. Furthermore, it sets the step size α to ${}^{\|\boldsymbol{m}\|_1}/_{d}$ for a more precise binarization of $\boldsymbol{m} \in \mathbb{R}^d$. Noisy-SignSGD adds Gaussian noise $\xi \sim N(0, \sigma^2)$ to the model update and then performs binarization, where σ is an adjustable standard deviation. We set the step size α and the standard deviation σ to 0.01 and 0.01 for Noisy-SignSGD. Stoc-SignSGD adds uniform noise $\xi \sim U(-\|\boldsymbol{m}\|, \|\boldsymbol{m}\|)$ to a model update $\boldsymbol{m}$ before binarization. In this way, each element $\boldsymbol{m}_i$ will be binarized into +1 with probability $({}^{1}/_{2} + {}^{\boldsymbol{m}_i}/_{2\|\boldsymbol{m}\|})$ and into -1 with probability $({}^{1}/_{2} - {}^{\boldsymbol{m}_i}/_{2\|\boldsymbol{m}\|})$. However, we observe that the value of $\|\boldsymbol{m}\|$ is so large that the above two probabilities are both close to 0.5. Therefore, in our experiments, we replaced $\|\boldsymbol{m}\|$ with $\|\boldsymbol{m}\|_\infty$ for a better binarization. Besides, the step size of Stoc-SignSGD is tuned and set to 0.01. ZeroFL performs SWAT (Raihan & Aamodt, 2020) in local training and prune the model updates after training. The sparsity ratio is set to 97% to ensure a communication compression ratio similar to the binarization methods.

B. Additional Experiment Results

B.1. Additional Baselines

In this subsection, we test additional baselines on CIFAR-10, including FedPAQ (Reisizadeh et al., 2020) and BiFL (Yang et al., 2021). FedPAQ quantizes model updates after local training. Table 5 shows that FedPAQ achieve comparable accuracy to FedBAT with a communication costs ranging from 3 to 4 bits per parameter (bpp). BiFL trains a binary neural network (BNN) during local training. There are two ways to upload local weights in BiFL, one is to communicate full-precision weights (BiFL-Full), and the other is to communicate binarized weights (BiFL). BiFL-Full achieves higher accuracy as communication is uncompressed. However, BiFL directly binarizes model weights rather than binarizing model updates, resulting in much lower accuracy compared to SignSGD.

Table 5. The test accuracy of additional baselines (including FedPAQ and BiFL) on CIFAR-10 with 100 clients.

	FedAvg	SignSGD	FedPAQ [3-bit]	FedPAQ [4-bit]	BiFL	BiFL-Full	FedBAT
IID	89.3	87.3	88.8	89.1	61.8	86.8	89.2
Non-IID-1	84.6	82.2	83.9	84.5	37.4	84.3	84.9
Non-IID-2	80.9	76.6	80.5	80.8	25.1	79.6	81.0

B.2. More Clients

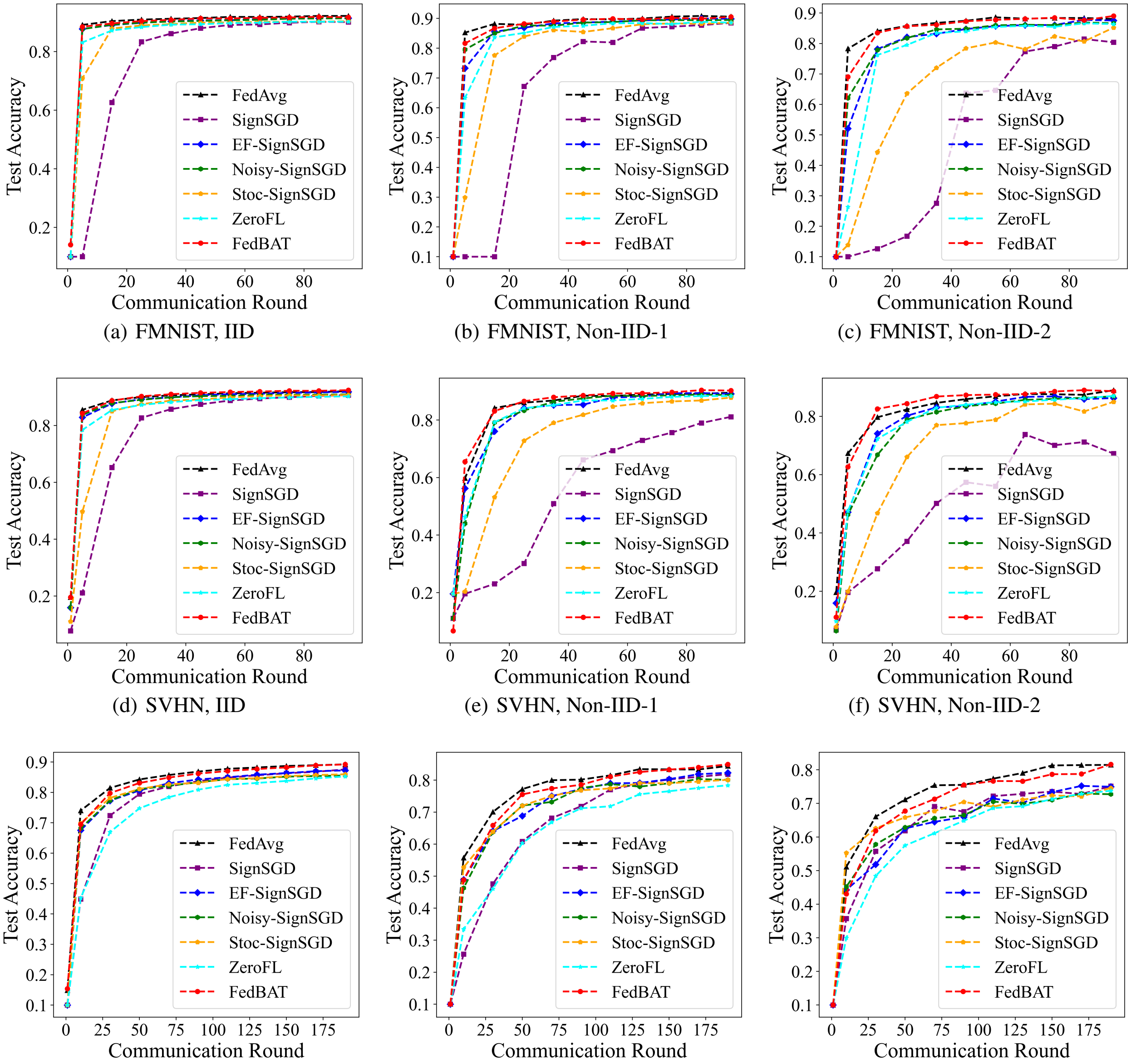
In Section 5, the number of clients is set to 30 and 100. Now, we expand the number of clients to 200. As shown in Table 6, despite this increase, FedBAT consistently achieves accuracy comparable to FedAvg and notably surpasses SignSGD.

Table 6. The test accuracy of FedAvg, SignSGD and FedBAT with 200 clients.

	FMNIST with CNN			SVHN with CNN			CIFAR-10 with ResNet-10			CIFAR-100 with ResNet-10		
	IID	Non-IID-1	Non-IID-2	IID	Non-IID-1	Non-IID-2	IID	Non-IID-1	Non-IID-2	IID	Non-IID-1	Non-IID-2
FedAvg	91.6	90.2	88.4	91.4	88.6	87.8	84.9	81.0	78.2	49.7	48.8	42.9
SignSGD	89.5	87.5	80.2	89.6	79.6	67.6	84.2	77.8	71.0	46.1	34.2	30.8
FedBAT	91.4	89.9	88.1	91.4	89.2	87.8	85.0	80.8	77.8	49.1	48.4	43.2

B.3. Additional Convergence Curves

Due to limited space, we illustrate the convergence curves of various methods for the FMNIST, SVHN and CIFAR-10 datasets in Figure 3. The convergence patterns of the methods closely resemble those observed in Figure 2.



(a) FMNIST, IID

(b) FMNIST, Non-IID-1

(c) FMNIST, Non-IID-2

(d) SVHN, IID

(e) SVHN, Non-IID-1

(f) SVHN, Non-IID-2

(g) CIFAR-10, IID

(h) CIFAR-10, Non-IID-1

(i) CIFAR-10, Non-IID-2

Figure 3. Convergence curves of FedBAT and baselines on FMNIST, SVHN and CIFAR-10 with 100 clients.

C. Proof of Theorem 1

In this section, we analyze FedBAT in the setting of full device participation. The theoretical analysis in this paper is rooted in the findings about FedAvg presented in (Li et al., 2020).

C.1. Additional Notation

Let $\mathbf{w}_t^k$ be the model parameters maintained in the k-th device at the t-th step. Let $\mathcal{I}_\tau$ be the set of global synchronization steps, i.e., $\mathcal{I}_\tau = \{n\tau | n = 1, 2, ...\}$. If $t+1 \in \mathcal{I}_\tau$, i.e., the time step to communication, FedBAT activates all devices. Then the optimization of FedBAT can be described as

$$\mathbf{v}_{t+1}^k = \mathbf{w}_t^k - \eta_t \nabla F_k(\mathbf{x}_t^k, \xi_t^k) \tag{17}$$

$$\mathbf{x}_t^k = \mathcal{S}_m(\mathbf{w}_t^k) \tag{18}$$

$$\mathbf{w}_{t+1}^k = \begin{cases} \mathbf{v}_{t+1}^k & \text{if } t+1 \notin \mathcal{I}_\tau, \\ \sum_{k=1}^N p_k \mathcal{S}_m(\mathbf{v}_{t+1}^k) & \text{if } t+1 \in \mathcal{I}_\tau. \end{cases} \tag{19}$$

Here, the variable $\mathbf{v}_{t+1}^k$ is introduced to represent the immediate result of one step SGD update from $\mathbf{w}_t^k$. We interpret $\mathbf{w}_{t+1}^k$ as the parameters obtained after communication steps (if possible). Also, an additional variable $\mathbf{x}_t^k$ is introduced to represent the result of performing binarization on model updates.

In our analysis, we define two virtual sequences $\bar{\mathbf{v}}_t = \sum_{k=1}^N p_k \mathbf{v}_t^k$ and $\bar{\mathbf{w}}_t = \sum_{k=1}^N p_k \mathbf{w}_t^k$. $\bar{\mathbf{v}}_{t+1}$ results from an single step of SGD from $\bar{\mathbf{w}}_t$. When $t+1 \notin \mathcal{I}_\tau$, both are inaccessible. When $t+1 \in \mathcal{I}_\tau$, we can only fetch $\bar{\mathbf{w}}_{t+1}$. For convenience, we define $\bar{\mathbf{g}}_t = \sum_{k=1}^N p_k \nabla F_k(\mathbf{x}_t^k)$ and $\mathbf{g}_t = \sum_{k=1}^N p_k \nabla F_k(\mathbf{x}_t^k, \xi_t^k)$. Therefore, $\bar{\mathbf{v}}_{t+1} = \bar{\mathbf{w}}_t - \eta_t \mathbf{g}_t$ and $\mathbb{E}\mathbf{g}_t = \bar{\mathbf{g}}_t$. Notably, for any $t \geq 0$, there exists a $t_0 \leq t$, such that $t - t_0 \leq \tau - 1$ and $\mathbf{w}_{t_0}^k = \bar{\mathbf{w}}_{t_0}$ for all $k = 1, 2, ..., N$. In this case, $\mathbf{x}_t^k = \mathcal{S}_m(\mathbf{w}_t^k) = \mathcal{S}(\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}) + \bar{\mathbf{w}}_{t_0}$. Therefore, we have $\mathbb{E}\mathbf{x}_t^k = \mathbf{w}_t^k$ and $\mathbb{E}\|\mathbf{x}_t^k - \mathbf{w}_t^k\|^2 \leq q^2 \|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2$.

C.2. Key Lemmas

To convey our proof clearly, it would be necessary to prove certain useful lemmas. We defer the proof of these lemmas to latter section and focus on proving the main theorem.

Lemma 1. *(Results of one step SGD). Assume Assumption 1 and 2. If $\eta_t \leq \frac{1}{4L}$, we have*

$$\mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \leq (1 - \eta_t \mu)\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta_t^2 \mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 + 6L\eta_t^2 \Gamma + 2\sum_{k=1}^N p_k \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2. \tag{20}$$

Lemma 2. *(Bounding the variance). Assume Assumption 3 holds. It follows that*

$$\mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 \leq \sum_{k=1}^N p_k^2 \sigma^2. \tag{21}$$

Lemma 3. *(Bounding the divergence of $\mathbf{x}_t^k$). Assume Assumption 4, that η_t is non-increasing and $\eta_t \leq 2\eta_{t+\tau}$ for all $t \geq 0$. It follows that*

$$\sum_{k=1}^N p_k \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 \leq 4(1+q^2)\eta_t^2(\tau - 1)^2 G^2. \tag{22}$$

Lemma 4. *(Bounding the divergence of $\bar{\mathbf{w}}_t$). Assume Assumption 4 and 5, that η_t is non-increasing and $\eta_t \leq 2\eta_{t+\tau}$ for all $t \geq 0$. It follows that*

$$\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \leq 4\sum_{k=1}^N p_k^2 q^2 \eta_t^2 \tau^2 G^2. \tag{23}$$

C.3. Completing the Proof of Theorem 1

Note that $\mathbb{E}\bar{\mathbf{w}}_t = \mathbb{E}\bar{\mathbf{v}}_t$ when we take expectation to erase the randomness of stochastic binarization, therefore

$$\begin{aligned}\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \mathbf{w}^*\|^2 &= \mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1} + \bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \\ &= \mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 + \mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2.\end{aligned} \tag{24}$$

Let $\Delta_t = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2$. From Lemma 1-4, it follows that

$$\Delta_{t+1} \leq (1 - \eta_t \mu)\Delta_t + \eta_t^2 B, \tag{25}$$

where

$$B = \sum_{k=1}^{N} p_k^2 \sigma^2 + 6L\Gamma + 8(1+q^2)(\tau-1)^2 G^2 + 4\sum_{k=1}^{N} p_k^2 q^2 \tau^2 G^2. \tag{26}$$

For a diminishing stepsize, $\eta_t = \frac{\beta}{t+\gamma}$ for some $\beta > \frac{1}{\mu}$ and $\gamma > 0$ such that $\eta_1 \leq \min\{\frac{1}{\mu}, \frac{1}{4L}\} = \frac{1}{4L}$ and $\eta_t \leq 2\eta_{t+\tau}$. We will prove $\Delta_t \leq \frac{v}{t+\gamma}$ where $v = \max\{\frac{\beta^2 B}{\beta\mu - 1}, (\gamma+1)\Delta_1\}$. We prove it by induction. Firstly, the definition of v ensures that it holds for $t = 1$. Assume the conclusion holds for some t, it follows that

$$\begin{aligned}\Delta_{t+1} &\leq (1 - \eta_t\mu)\Delta_t + \eta_t^2 B \\ &\leq (1 - \frac{\beta\mu}{t+\gamma})\frac{v}{t+\gamma} + \frac{\beta^2 B}{(t+\gamma)^2} \\ &= \frac{t+\gamma-1}{(t+\gamma)^2}v + \frac{\beta^2 B}{(t+\gamma)^2} - \frac{\beta\mu - 1}{(t+\gamma)^2}v \\ &\leq \frac{v}{t+\gamma+1}.\end{aligned} \tag{27}$$

Then by the L-smoothness of F,

$$\mathbb{E}[F(\bar{\mathbf{w}}_t)] - F^* \leq \frac{L}{2}\Delta_t \leq \frac{L}{2}\frac{v}{\gamma+t}. \tag{28}$$

Specifically, if we choose $\beta = \frac{2}{\mu}, \gamma = \max\{8\frac{L}{\mu}, \tau\} - 1$ and denote $\kappa = \frac{L}{\mu}$, then $\eta_t = \frac{2}{\mu}\frac{1}{\gamma+t}$. One can verify that the choice of η_t satisfies $\eta_t \leq 2\eta_{t+\tau}$ for $t \geq 1$. Then, we have

$$v = \max\{\frac{\beta^2 B}{\beta\mu - 1}, (\gamma+1)\Delta_1\} \leq \frac{\beta^2 B}{\beta\mu - 1} + (\gamma+1)\Delta_1 \leq \frac{4B}{\mu^2} + (\gamma+1)\Delta_1, \tag{29}$$

and

$$\mathbb{E}[F(\bar{\mathbf{w}}_t)] - F^* \leq \frac{L}{2}\frac{v}{\gamma+t} \leq \frac{\kappa}{\gamma+t}(\frac{2B}{\mu} + \frac{\mu(\gamma+1)}{2}\Delta_1). \tag{30}$$

C.4. Deferred Proofs of Key Lemmas

Proof of Lemma 1. Notice that $\bar{\mathbf{v}}_{t+1} = \bar{\mathbf{w}}_t - \eta_t \mathbf{g}_t$ and $\mathbb{E}\mathbf{g}_t = \bar{\mathbf{g}}_t$, then

$$\begin{aligned}\mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 &= \mathbb{E}\|\bar{\mathbf{w}}_t - \eta_t\mathbf{g}_t - \mathbf{w}^* - \eta_t\bar{\mathbf{g}}_t + \eta_t\bar{\mathbf{g}}_t\|^2 \\ &= \underbrace{\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^* - \eta_t\bar{\mathbf{g}}_t\|^2}_{A_1} + \eta_t^2\mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2.\end{aligned} \tag{31}$$

We next focus on bounding A_1. Again we split A_1 into three terms:

$$\|\bar{\mathbf{w}}_t - \mathbf{w}^* - \eta_t\bar{\mathbf{g}}_t\|^2 = \|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 \underbrace{-2\eta_t\langle\bar{\mathbf{w}}_t - \mathbf{w}^*, \bar{\mathbf{g}}_t\rangle}_{B_1} + \underbrace{\eta_t^2\|\bar{\mathbf{g}}_t\|^2}_{B_2}. \tag{32}$$

From the the L-smoothness of F_k, it follows that

$$\|\nabla F_k(\mathbf{x}_t^k)\|^2 \leq 2L(F_k(\mathbf{x}_t^k) - F_k^*). \tag{33}$$

By the convexity of $\| \cdot \|^2$ and Eq. 33, we have

$$B_2 = \eta_t^2 \|\bar{\mathbf{g}}_t\|^2 \leq \eta_t^2 \sum_{k=1}^{N} p_k \|\nabla F_k(\mathbf{x}_t^k)\|^2 \leq 2L\eta_t^2 \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k^*). \tag{34}$$

Note that

$$\begin{aligned} B_1 &= -2\eta_t \left\langle \bar{\mathbf{w}}_t - \mathbf{w}^*, \bar{\mathbf{g}}_t \right\rangle = -2\eta_t \sum_{k=1}^{N} p_k \left\langle \bar{\mathbf{w}}_t - \mathbf{w}^*, \nabla F_k(\mathbf{x}_t^k) \right\rangle \\ &= -2\eta_t \sum_{k=1}^{N} p_k \left\langle \bar{\mathbf{w}}_t - \mathbf{x}_t^k, \nabla F_k(\mathbf{x}_t^k) \right\rangle - 2\eta_t \sum_{k=1}^{N} p_k \left\langle \mathbf{x}_t^k - \mathbf{w}^*, \nabla F_k(\mathbf{x}_t^k) \right\rangle . \end{aligned} \tag{35}$$

By Cauchy-Schwarz inequality and AM-GM inequality, we have

$$-2 \left\langle \bar{\mathbf{w}}_t - \mathbf{x}_t^k, \nabla F_k(\mathbf{x}_t^k) \right\rangle \leq \frac{1}{\eta_t} \|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \eta_t \|\nabla F_k(\mathbf{x}_t^k)\|^2. \tag{36}$$

By the μ-strong convexity of F_k, we have

$$-\left\langle \mathbf{x}_t^k - \mathbf{w}^*, \nabla F_k(\mathbf{x}_t^k) \right\rangle \leq -(F_k(\mathbf{x}_t^k) - F_k(\mathbf{w}^*)) - \frac{\mu}{2} \|\mathbf{x}_t^k - \mathbf{w}^*\|^2. \tag{37}$$

Therefore, we have

$$\begin{aligned} A_1 = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^* - \eta_t \bar{\mathbf{g}}_t\|^2 \leq \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 &+ 2L\eta_t^2 \mathbb{E} \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k^*) \\ &+ \eta_t \mathbb{E} \sum_{k=1}^{N} p_k (\frac{1}{\eta_t} \|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \eta_t \|\nabla F_k(\mathbf{x}_t^k)\|^2) \\ &- 2\eta_t \mathbb{E} \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k(\mathbf{w}^*) + \frac{\mu}{2} \|\mathbf{x}_t^k - \mathbf{w}^*\|^2) \\ &\leq (1 - \mu\eta_t) \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \sum_{k=1}^{N} p_k \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 \\ &+ \underbrace{\tau [4L\eta_t^2 \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k^*) - 2\eta_t \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k(\mathbf{w}^*))]}_{C} \end{aligned} \tag{38}$$

where we use Eq.33 again and the inequality $-\mathbb{E}\|\mathbf{x}_t^k - \mathbf{w}^*\|^2 = -\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 - \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 \leq -\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2$. We next aim to bound C. We define $\gamma_t = 2\eta_t(1 - 2L\eta_t)$. Since $\eta_t \leq \frac{1}{4L}, \eta_t \leq \gamma_t \leq 2\eta_t$. Then we split C into two terms:

$$\begin{aligned} C &= -2\eta_t(1 - 2L\eta_t) \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F_k^*) + 2\eta_t \sum_{k=1}^{N} p_k (F_k(\mathbf{w}^*) - F_k^*) \\ &= -\gamma_t \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F^*) + (2\eta_t - \gamma_t) \sum_{k=1}^{N} p_k (F^* - F_k^*) \\ &= \underbrace{-\gamma_t \sum_{k=1}^{N} p_k (F_k(\mathbf{x}_t^k) - F^*)}_{D} + 4L\eta_t^2 \Gamma \end{aligned} \tag{39}$$

where in the last equation, we use the notation $\Gamma = \sum_{k=1}^{N} p_k (F^* - F_k^*) = F^* - \sum_{k=1}^{N} p_k F_k^*$.

To bound D, we have

$$\begin{aligned}\sum_{k=1}^{N} p_k(F_k(\mathbf{x}_t^k) - F^*) &= \sum_{k=1}^{N} p_k(F_k(\mathbf{x}_t^k) - F_k(\bar{\mathbf{w}}_t)) + \sum_{k=1}^{N} p_k(F_k(\bar{\mathbf{w}}_t) - F^*) \\ &\geq \sum_{k=1}^{N} p_k \left\langle \nabla F_k(\bar{\mathbf{w}}_t), \mathbf{x}_t^k - \bar{\mathbf{w}}_t \right\rangle + F(\bar{\mathbf{w}}_t) - F^* \\ &\geq -\frac{1}{2}\sum_{k=1}^{N} p_k[\eta_t \|\nabla F_k(\bar{\mathbf{w}}_t)\|^2 + \frac{1}{\eta_t}\|\mathbf{x}_t^k - \bar{\mathbf{w}}_t\|^2] + F(\bar{\mathbf{w}}_t) - F^* \\ &\geq -\sum_{k=1}^{N} p_k[\eta_t L(F_k(\bar{\mathbf{w}}_t) - F_k^*) + \frac{1}{2\eta_t}\|\mathbf{x}_t^k - \bar{\mathbf{w}}_t\|^2] + F(\bar{\mathbf{w}}_t) - F^*\end{aligned} \tag{40}$$

where the first inequality results from the convexity of F_k, the second inequality from AM-GM inequality and the third inequality from Eq. 33. Therefore

$$\begin{aligned} C &= \gamma_t \sum_{k=1}^{N} p_k[\eta_t L(F_k(\bar{\mathbf{w}}_t) - F_k^*) + \frac{1}{2\eta_t}\|\mathbf{x}_t^k - \bar{\mathbf{w}}_t\|^2] - \gamma_t(F(\bar{\mathbf{w}}_t) - F^*) + 4L\eta_t^2\Gamma \\ &= \gamma_t(\eta_t L - 1)\sum_{k=1}^{N} p_k(F_k(\bar{\mathbf{w}}_t) - F_k^*) + (4L\eta_t^2 + \gamma_t\eta_t L)\Gamma + \frac{\gamma_t}{2\eta_t}\sum_{k=1}^{N} p_k\|\mathbf{x}_t^k - \bar{\mathbf{w}}_t\|^2 \\ &\leq 6L\eta_t^2\Gamma + \sum_{k=1}^{N} p_k\|\mathbf{x}_t^k - \bar{\mathbf{w}}_t\|^2 \end{aligned} \tag{41}$$

where in the last inequality, we use the following facts: (1)$\eta_t L - 1 \leq -\frac{3}{4} \leq 0$ and $\sum_{k=1}^{N} p_k(F_k(\bar{\mathbf{w}}_t) - F^*) = F(\bar{\mathbf{w}}_t) - F^* \geq 0$ (2) $\Gamma \geq 0$ and $4L\eta_t^2 + \gamma_t\eta_t L \leq 6\eta_t^2 L$ and (3) $\frac{\gamma_t}{2\eta_t} \leq 1$.

Recalling the expression of A_1 and plugging C into it, we have

$$A_1 = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^* - \eta_t\bar{\mathbf{g}}_t\|^2 \leq (1 - \mu\eta_t)\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 + 2\sum_{k=1}^{N} p_k\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + 6L\eta_t^2\Gamma. \tag{42}$$

Plugging A_1 into Eq. 31, we have the result in Lemma 1

$$\mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \leq (1 - \eta_t\mu)\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta_t^2\mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 + 6L\eta_t^2\Gamma + 2\sum_{k=1}^{N} p_k\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2. \tag{43}$$

Proof of Lemma 2. From Assumption 3, the variance of the stochastic gradients in device k is bounded by σ^2, then

$$\begin{aligned}\mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 &= \mathbb{E}\|\sum_{k=1}^{N} p_k(\nabla F_k(\mathbf{w}_t^k, \xi_t^k) - \nabla F_k(\mathbf{w}_t^k))\|^2 \\ &= \sum_{k=1}^{N} p_k^2\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k) - \nabla F_k(\mathbf{w}_t^k)\|^2 \\ &\leq \sum_{k=1}^{N} p_k^2\sigma^2\end{aligned} \tag{44}$$

Proof of Lemma 3. Considering that $\mathbb{E}\mathbf{x}_t^k = \mathbf{w}_t^k$, we have

$$\begin{aligned}\sum_{k=1}^{N} p_k\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 &= \sum_{k=1}^{N} p_k\mathbb{E}\|(\bar{\mathbf{w}}_t - \mathbf{w}_t^k) - (\mathbf{w}_t^k - \mathbf{x}_t^k)\|^2 \\ &= \sum_{k=1}^{N} p_k\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}_t^k\|^2 + \sum_{k=1}^{N} p_k\mathbb{E}\|\mathbf{w}_t^k - \mathbf{x}_t^k\|^2\end{aligned} \tag{45}$$

Since FedBAT requires a communication each τ steps. Therefore, for any $t \geq 0$, there exists a $t_0 \leq t$, such that $t - t_0 \leq \tau - 1$ and $\mathbf{w}_{t_0}^k = \bar{\mathbf{w}}_{t_0}$ for all $k = 1, 2, ..., N$. Also, we use the fact that η_t is non-increasing and $\eta_{t_0} \leq 2\eta_t$ for all $t - t_0 \leq \tau - 1$, then

$$\begin{aligned}\sum_{k=1}^{N} p_k \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}_t^k\|^2 &= \sum_{k=1}^{N} p_k \mathbb{E}\|(\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}) - (\bar{\mathbf{w}}_t - \bar{\mathbf{w}}_{t_0})\|^2 \\ &\leq \sum_{k=1}^{N} p_k \mathbb{E}\|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2 \\ &\leq \sum_{k=1}^{N} p_k \mathbb{E} \sum_{t=t_0}^{t-1} (\tau - 1)\eta_t^2 \|\nabla F_k(\mathbf{x}_t^k, \xi_t^k)\|^2 \\ &\leq \sum_{k=1}^{N} p_k \sum_{t=t_0}^{t-1} (\tau - 1)\eta_t^2 G^2 \\ &\leq \sum_{k=1}^{N} p_k \eta_t^2 (\tau - 1)^2 G^2 \\ &\leq 4\eta_t^2 (\tau - 1)^2 G^2\end{aligned} \tag{46}$$

Here in the first inequality, we use $\mathbb{E}\|X - \mathbb{E}X\|^2 \leq \mathbb{E}\|X\|^2$ where $X = \mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}$ with probability p_k. In the second inequality, we use Jensen inequality:

$$\|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2 = \|\sum_{t=t_0}^{t-1} \eta_t \nabla F_k(\mathbf{w}_t^k, \xi_t^k)\|^2 \leq (t - t_0) \sum_{t=t_0}^{t-1} \eta_t^2 \|\nabla F_k(\mathbf{w}_t^k, \xi_t^k)\|^2. \tag{47}$$

In the third inequality, we use $\eta_t \leq \eta_{t_0}$ for $t \geq t_0$ and $\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k)\|^2 \leq G^2$ for $k = 1, 2, ..., N$ and $t \geq 1$. In the last inequality, we use $\eta_{t_0} \leq 2\eta_{t_0+\tau} \leq 2\eta_t$ for $t_0 \leq t \leq t_0 + \tau$.

According to Assumption 5, we have $\mathbb{E}\|\mathbf{w}_t^k - \mathbf{x}_t^k\|^2 \leq q^2 \mathbb{E}\|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2$ as discussed in Section C.1. Then the second term in Eq. 45 can be bounded by reusing the result in Eq. 46 as

$$\sum_{k=1}^{N} p_k \mathbb{E}\|\mathbf{w}_t^k - \mathbf{x}_t^k\|^2 \leq q^2 \sum_{k=1}^{N} p_k \mathbb{E}\|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2 \leq 4q^2\eta_t^2(\tau - 1)^2 G^2. \tag{48}$$

Plugging Eq. 46 and Eq. 48 into Eq. 45, we have the result in Lemma 3

$$\sum_{k=1}^{N} p_k \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 \leq 4(1 + q^2)\eta_t^2(\tau - 1)^2 G^2. \tag{49}$$

Proof of Lemma 4. Notice that $\bar{\mathbf{w}}_{t+1} = \bar{\mathbf{v}}_{t+1}$ when $t + 1 \notin \mathcal{I}_\tau$ and $\bar{\mathbf{w}}_{t+1} = \sum_{k=1}^{N} p_k \mathcal{S}_m(\mathbf{v}_{t+1}), \bar{\mathbf{v}}_{t+1} = \sum_{k=1}^{N} p_k \mathbf{v}_{t+1}$ when $t + 1 \in \mathcal{I}_\tau$. Hence, if $t + 1 \in \mathcal{I}_\tau$, we have

$$\begin{aligned}\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 &= \mathbb{E}\|\sum_{k=1}^{N} p_k(\mathcal{S}_m(\mathbf{v}_{t+1}^k) - \mathbf{v}_{t+1}^k)\|^2 \\ &= \sum_{k=1}^{N} p_k^2 \mathbb{E}\|(\mathcal{S}_m(\mathbf{v}_{t+1}^k) - \mathbf{v}_{t+1}^k)\|^2 \\ &\leq \sum_{k=1}^{N} p_k^2 q^2 \mathbb{E}\|\mathbf{v}_{t+1}^k - \bar{\mathbf{w}}_{t_0}\|^2 \\ &\leq 4\sum_{k=1}^{N} p_k^2 q^2 \eta_t^2 \tau^2 G^2,\end{aligned} \tag{50}$$

where the last inequality follows the result in Eq.47.

D. Proof of Theorem 2

In this section, we analyze FedBAT in the setting of partial device participation.

D.1. Additional Notation

Recall that $\mathbf{w}_t^k$ is the model parameter maintained in the k-th device at the t-th step. $\mathcal{I}_\tau = \{n\tau | n = 1, 2, ..., N\}$ is the set of global synchronization steps. Again, $\bar{\mathbf{g}}_t = \sum_{k=1}^N p_k \nabla F_k(\mathbf{x}_t^k)$ and $\mathbf{g}_t = \sum_{k=1}^N p_k \nabla F_k(\mathbf{x}_t^k, \xi_t^k)$. Therefore, $\bar{\mathbf{v}}_{t+1} = \bar{\mathbf{w}}_t - \eta_t \mathbf{g}_t$ and $\mathbb{E}\mathbf{g}_t = \bar{\mathbf{g}}_t$.

Now we consider the case where FedBAT samples a random set $\mathbb{S}_t$ of devices to participate in each round of training. This make the analysis a little bit intricate, since $\mathbb{S}_t$ varies each τ steps. Following (Li et al., 2020), we assume that FedBAT always activates all devices at the beginning of each round and then uses the parameters maintained in only a few sampled devices to produce the next-round parameter. It is clear that this updating scheme is equivalent to the original. As assumed in Theorem 2 that $p_1 = ... = p_N = \frac{1}{N}$, the update of FedBAT with partial devices active can be described as: for all $k \in [N]$,

$$\mathbf{v}_{t+1}^k = \mathbf{w}_t^k - \eta_t \nabla F_k(\mathbf{x}_t^k, \xi_t^k) \tag{51}$$

$$\mathbf{x}_t^k = \mathcal{S}_m(\mathbf{w}_t^k) \tag{52}$$

$$\mathbf{w}_{t+1}^k = \begin{cases} \mathbf{v}_{t+1}^k & \text{if } t+1 \notin \mathcal{I}_\tau, \\ \frac{\sum_{k \in \mathbb{S}_{t+1}} p_k \mathcal{S}_m(\mathbf{v}_t^k)}{\sum_{k \in \mathbb{S}_{t+1}} p_k} = \frac{1}{K} \sum_{k \in \mathbb{S}_{t+1}} \mathcal{S}_m(\mathbf{v}_{t+1}^k) & \text{if } t+1 \in \mathcal{I}_\tau. \end{cases} \tag{53}$$

D.2. Key Lemmas

Lemma 5. *(Unbiased sampling scheme). In the case of partial device participation in Theorem 2, we have*

$$\mathbb{E}\bar{\mathbf{w}}_{t+1} = \mathbb{E}\bar{\mathbf{v}}_{t+1}. \tag{54}$$

Lemma 6. *(Bounding the variance of $\bar{\mathbf{w}}_t$). In the case of partial device participation in Theorem 2, with Assumption 4 and 5, assume that η_t is non-increasing and $\eta_t \leq \eta_{t+\tau}$ for all $t \geq 0$. It follows that*

$$\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \leq 4\frac{q^2(N-1) + N - K}{K(N-1)} \eta_t^2 \tau^2 G^2. \tag{55}$$

D.3. Completing the Proof of Theorem 2

Using Lemma 5, Eq.24 still holds, that is

$$\begin{aligned} \mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \mathbf{w}^*\|^2 &= \mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1} + \bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \\ &= \mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 + \mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \end{aligned} \tag{56}$$

Let $\Delta_t = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2$. From Lemma 1, 2, 3 and 6, it follows that

$$\Delta_{t+1} \leq (1 - \eta_t \mu)\Delta_t + \eta_t^2 B \tag{57}$$

where

$$B = \frac{\sigma^2}{N} + 6L\Gamma + 8(1+q^2)(\tau-1)^2 G^2 + 4\frac{q^2(N-1) + N - K}{K(N-1)} \tau^2 G^2 \tag{58}$$

The only difference between Eq.57 and Eq.25 is the value of constant B. Following the same process, we can get the result of Theorem 2

$$\mathbb{E}[F(\bar{\mathbf{w}}_t)] - F^* \leq \frac{L}{2}\frac{v}{\gamma + t} \leq \frac{\kappa}{\gamma + t}(\frac{2B}{\mu} + \frac{\mu(\gamma+1)}{2}\|\mathbf{w}_1 - \mathbf{w}^*\|^2), \tag{59}$$

where $\gamma = \max\{8\frac{L}{\mu}, \tau\} - 1$ and $\kappa = \frac{L}{\mu}$.

D.4. Deferred Proofs of Key Lemmas

Proof of Lemma 5. Considering the partial device participation in Theorem 2, $\bar{\mathbf{w}}_{t+1} = \bar{\mathbf{v}}_{t+1}$ when $t+1 \notin \mathcal{I}_\tau$ and $\bar{\mathbf{w}}_{t+1} = \frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathcal{S}_m(\mathbf{v}^k_{t+1}), \bar{\mathbf{v}}_{t+1} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{v}^k_{t+1}$ when $t+1 \in \mathcal{I}_\tau$. In the latter case, there are two kinds of randomness between $\bar{\mathbf{w}}_{t+1}$ and $\bar{\mathbf{v}}_{t+1}$, respectively from the client's random selection and stochastic binarization. To distinguish them, we use the notation $\mathbb{E}_{\mathbb{S}_t}$ when we take expectation to erase the randomness of device selection, and use the notation $\mathbb{E}_{\mathcal{S}}$ when we take expectation to erase the randomness of binarization. Therefore, when $t+1 \in \mathcal{I}_\tau$, we have

$$\mathbb{E}\bar{\mathbf{w}}_{t+1} = \mathbb{E}_{\mathbb{S}_t}[\mathbb{E}_{\mathcal{S}}\bar{\mathbf{w}}_{t+1}] = \mathbb{E}_{\mathbb{S}_t}[\mathbb{E}_{\mathcal{S}}\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathcal{S}_m(\mathbf{v}^k_{t+1})] = \mathbb{E}_{\mathbb{S}_t}[\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathbf{v}^k_{t+1}] = \bar{\mathbf{v}}_{t+1} \tag{60}$$

Proof of Lemma 6. Notice that $\bar{\mathbf{w}}_{t+1} = \bar{\mathbf{v}}_{t+1}$ when $t+1 \notin \mathcal{I}_\tau$ and $\bar{\mathbf{w}}_{t+1} = \frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathcal{S}_m(\mathbf{v}^k_{t+1}), \bar{\mathbf{v}}_{t+1} = \sum_{k=1}^{N} p_k \mathbf{v}_{t+1}$ when $t+1 \in \mathcal{I}_\tau$. Hence, we have

$$\begin{aligned}
\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 &= \mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathcal{S}_m(\mathbf{v}^k_{t+1}) - \bar{\mathbf{v}}_{t+1}\|^2 \\
&= \mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} (\mathcal{S}_m(\mathbf{v}^k_{t+1}) - \mathbf{v}^k_{t+1}) + \frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathbf{v}^k_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \\
&= \underbrace{\mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} (\mathcal{S}_m(\mathbf{v}^k_{t+1}) - \mathbf{v}^k_{t+1})\|^2}_{A_1} + \underbrace{\mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathbf{v}^k_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2}_{A_2}
\end{aligned} \tag{61}$$

To bound A_1, we have

$$\begin{aligned}
A_1 &= \mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} (\mathcal{S}_m(\mathbf{v}^k_{t+1}) - \mathbf{v}^k_{t+1})\|^2 \\
&= \frac{1}{K^2}\sum_{k\in\mathbb{S}_{t+1}} \mathbb{E}\|(\mathcal{S}_m(\mathbf{v}^k_{t+1}) - \mathbf{v}^k_{t+1})\|^2 \\
&\le \frac{1}{K^2}\sum_{k\in\mathbb{S}_{t+1}} 4q^2\eta_t^2\tau^2G^2 \\
&= \frac{4}{K}q^2\eta_t^2\tau^2G^2,
\end{aligned} \tag{62}$$

where in the inequality we use the result of Eq.50. Then, to bound A_2, we have

$$\begin{aligned}
A_2 &= \mathbb{E}\|\frac{1}{K}\sum_{k\in\mathbb{S}_{t+1}} \mathbf{v}^k_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \\
&= \frac{1}{K^2}\mathbb{E}\|\sum_{i=1}^{N} \mathbb{I}\{i\in\mathbb{S}_{t+1}\}(\mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1})\|^2 \\
&= \frac{1}{K^2}\mathbb{E}[\sum_{i=1}^{N} \mathbb{P}(i\in\mathbb{S}_{t+1})\|\mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 + \sum_{i\neq j}\mathbb{P}(i,j\in\mathbb{S}_{t+1})\left\langle \mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1}, \mathbf{v}^j_{t+1} - \bar{\mathbf{v}}_{t+1}\right\rangle] \\
&= \frac{1}{KN}\mathbb{E}\sum_{i=1}^{N}\|\mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 + \frac{K-1}{KN(N-1)}\mathbb{E}\sum_{i\neq j}\left\langle \mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1}, \mathbf{v}^j_{t+1} - \bar{\mathbf{v}}_{t+1}\right\rangle \\
&= \frac{N-K}{KN(N-1)}\sum_{i=1}^{N}\mathbb{E}\|\mathbf{v}^i_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2
\end{aligned} \tag{63}$$

where we use the following equalities: (1) $\mathbb{P}(i\in\mathbb{S}_{t+1}) = \frac{K}{N}$ and $\mathbb{P}(i,j\in\mathbb{S}_{t+1}) = \frac{K(K-1)}{N(N-1)}$ for all $i \neq j$ and (2)

$\sum_{i=1}^{N} \|\mathbf{v}_{t+1}^i - \bar{\mathbf{v}}_{t+1}\|^2 + \sum_{i \neq j} \left\langle \mathbf{v}_{t+1}^i - \bar{\mathbf{v}}_{t+1}, \mathbf{v}_{t+1}^j - \bar{\mathbf{v}}_{t+1} \right\rangle = 0$. Also using the result of Eq.50, we have

$$\begin{aligned} A_2 &= \frac{N-K}{KN(N-1)} \sum_{i=1}^{N} \mathbb{E}\|\mathbf{v}_{t+1}^i - \bar{\mathbf{v}}_{t+1}\|^2 \\ &\leq \frac{N-K}{K(N-1)} 4\eta_t^2 \tau^2 G^2 \end{aligned} \tag{64}$$

Plugging A_1 and A_2, we have the result in Lemma 6

$$\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \leq 4 \frac{q^2(N-1) + N - K}{K(N-1)} \eta_t^2 \tau^2 G^2. \tag{65}$$

E. Proof of Theorem 3

In this section, we proof the convergence of FedBAT under non-convex settings. We only use Assumptions 1, 3, 4, 5. Note that Assumptions 2 is about the strongly convex, which we do not use in this case.

E.1. Additional Notation

Let us review the notations in Section D.1. $\mathbf{w}_t^k$ is the model parameters maintained in the k-th device at the t-th step. Let $\mathcal{I}_\tau$ be the set of global synchronization steps, i.e., $\mathcal{I}_\tau = \{n\tau | n = 1, 2, ...\}$. If $t+1 \in \mathcal{I}_\tau$, i.e., the time step to communication, FedBAT activates all devices. Then the update of FedBAT can be described as

$$\mathbf{v}_{t+1}^k = \mathbf{w}_t^k - \eta_t \nabla F_k(\mathbf{x}_t^k, \xi_t^k) \tag{66}$$

$$\mathbf{x}_t^k = \mathcal{S}_m(\mathbf{w}_t^k) \tag{67}$$

$$\mathbf{w}_{t+1}^k = \begin{cases} \mathbf{v}_{t+1}^k & \text{if } t+1 \notin \mathcal{I}_\tau, \\ \sum_{k=1}^{N} p_k \mathcal{S}_m(\mathbf{v}_{t+1}^k) & \text{if } t+1 \in \mathcal{I}_\tau. \end{cases} \tag{68}$$

Here, the variable $\mathbf{v}_{t+1}^k$ is introduced to represent the immediate result of one step SGD update from $\mathbf{w}_t^k$. We interpret $\mathbf{w}_{t+1}^k$ as the parameter obtained after communication steps (if possible). Also, an additional variable $\mathbf{x}_t^k$ is introduced to represent the result of binarization on model update.

In our analysis, we define two virtual sequences $\bar{\mathbf{v}}_t = \sum_{k=1}^{N} p_k \mathbf{v}_t^k$ and $\bar{\mathbf{w}}_t = \sum_{k=1}^{N} p_k \mathbf{w}_t^k$. It is obviously that $\mathbb{E}\bar{\mathbf{v}}_t = \mathbb{E}\bar{\mathbf{w}}_t$. $\bar{\mathbf{v}}_{t+1}$ results from an single step of SGD from $\bar{\mathbf{w}}_t$. When $t+1 \notin \mathcal{I}_\tau$, both are inaccessible. When $t+1 \in \mathcal{I}_\tau$, we can only fetch $\bar{\mathbf{w}}_{t+1}$. For convenience, we define $\bar{\mathbf{g}}_t = \nabla F(\mathbf{x}_t^k) = \sum_{k=1}^{N} p_k \nabla F_k(\mathbf{x}_t^k)$ and $\mathbf{g}_t = \sum_{k=1}^{N} p_k \nabla F_k(\mathbf{x}_t^k, \xi_t^k)$. Therefore, $\bar{\mathbf{v}}_{t+1} = \bar{\mathbf{w}}_t - \eta_t \mathbf{g}_t$ and $\mathbb{E}\mathbf{g}_t = \bar{\mathbf{g}}_t$. Notably, for any $t \geq 0$, there exists a $t_0 \leq t$, such that $t - t_0 \leq \tau - 1$ and $\mathbf{w}_{t_0}^k = \bar{\mathbf{w}}_{t_0}$ for all $k = 1, 2, ..., N$. In this case, $\mathbf{x}_t^k = \mathcal{S}_m(\mathbf{w}_t^k) = \mathcal{S}(\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}) + \bar{\mathbf{w}}_{t_0}$. Therefore, we have $\mathbb{E}\mathbf{x}_t^k = \mathbf{w}_t^k$ and $\mathbb{E}\|\mathbf{x}_t^k - \mathbf{w}_t^k\|^2 \leq q^2 \|\mathbf{w}_t^k - \bar{\mathbf{w}}_{t_0}\|^2$.

E.2. Key Lemmas

Lemma 7. *Assume Assumption 1 and 3, we have*

$$\begin{aligned} \mathbb{E}F(\bar{\mathbf{w}}_{t+1}) \leq \mathbb{E}F(\bar{\mathbf{w}}_t) &- \frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 + \frac{L\eta^2 - \eta}{2}\mathbb{E}\|\bar{\mathbf{g}}_t\|^2 \\ &+ \frac{\eta}{2} L^2 \frac{1}{N} \sum_{k=1}^{N} \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \frac{L\eta^2}{2}\mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 + \frac{L}{2}\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \end{aligned} \tag{69}$$

E.3. Completing the Proof of Theorem 3

Since $\eta = \frac{1}{L\sqrt{T}}$, we have $\frac{L\eta^2 - \eta}{2} \leq 0$, then Eq.(69) can be rewritten as

$$\mathbb{E}F(\bar{\mathbf{w}}_{t+1}) \leq \mathbb{E}F(\bar{\mathbf{w}}_t) - \frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 + \frac{\eta}{2} L^2 \frac{1}{N} \sum_{k=1}^{N} \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \frac{L\eta^2}{2}\mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 + \frac{L}{2}\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \tag{70}$$

The last three terms can be bounded by Lemmas 2, 3 and 6. Note that these lemmas require no convex assumption. Therefore,

$$\begin{aligned}\mathbb{E}F(\bar{\mathbf{w}}_{t+1}) \le{}& \mathbb{E}F(\bar{\mathbf{w}}_t) - \frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 \\ &+ \frac{\eta}{2}L^2 4(1+q^2)\eta^2(\tau-1)^2G^2 + \frac{L\eta^2}{2}\frac{\sigma^2}{N} + \frac{L}{2}4\frac{q^2(N-1)+N-K}{K(N-1)}\eta_t^2\tau^2G^2\end{aligned} \tag{71}$$

Now rearranging the terms and summing over $t = 0, ..., T-1$ yield that

$$\begin{aligned}\frac{1}{2}\eta\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 \le{}& \frac{F(\mathbf{w}_0) - F(\mathbf{w}_T)}{T} \\ &+ \frac{\eta}{2}L^2 4(1+q^2)\eta^2(\tau-1)^2G^2 + \frac{L\eta^2}{2}\frac{\sigma^2}{N} + \frac{L}{2}4\frac{q^2(N-1)+N-K}{K(N-1)}\eta_t^2\tau^2G^2\end{aligned} \tag{72}$$

Picking the learning rate $\eta = \frac{1}{L\sqrt{T}}$, and with $F(\mathbf{w}_T) \ge \frac{1}{N}\sum_{k=1}^{N} F_k^* = F^* - \Gamma$, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 \le \frac{2L(F(\bar{\mathbf{w}}_0) - F^* + \Gamma)}{\sqrt{T}} + \frac{P}{\sqrt{T}} + \frac{Q}{T}, \tag{73}$$

where $P = \frac{\sigma^2}{N} + 4\frac{q^2(N-1)+N-K}{K(N-1)}\tau^2G^2$ and $Q = 4(1+q^2)(\tau-1)^2G^2$.

E.4. Deferred Proofs of Key Lemmas

Proof of Lemma 7. For any L-smooth function F, we have

$$F(\bar{\mathbf{w}}_{t+1}) \le F(\bar{\mathbf{v}}_{t+1}) + \langle \nabla F(\bar{\mathbf{v}}_{t+1}), \bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\rangle + \frac{L}{2}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \tag{74}$$

As $\mathbb{E}\bar{\mathbf{w}}_{t+1} = \mathbb{E}\bar{\mathbf{v}}_{t+1}$, taking expectations for the randomness of stochastic binarization and client selection yields that

$$\mathbb{E}F(\bar{\mathbf{w}}_{t+1}) \le \mathbb{E}F(\bar{\mathbf{v}}_{t+1}) + \frac{L}{2}\mathbb{E}\|\bar{\mathbf{w}}_{t+1} - \bar{\mathbf{v}}_{t+1}\|^2 \tag{75}$$

Since $\bar{\mathbf{v}}_{t+1} = \bar{\mathbf{w}}_t - \eta\mathbf{g}_t$, with L-smoothness, we have

$$F(\bar{\mathbf{v}}_{t+1}) \le F(\bar{\mathbf{w}}_t) - \eta\,\langle \nabla F(\bar{\mathbf{w}}_t), \mathbf{g}_t\rangle + \frac{L\eta^2}{2}\|\mathbf{g}_t\|^2 \tag{76}$$

The inner product term above can be written in expectation as follows:

$$2\mathbb{E}\,\langle \nabla F(\bar{\mathbf{w}}_t), \mathbf{g}_t\rangle = \mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 + \mathbb{E}\|\mathbf{g}_t\|^2 - \mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t) - \mathbf{g}_t\|^2 \tag{77}$$

Now, we consider the last term in Eq.77 with the fact that $\mathbb{E}\mathbf{g}_t = \mathbb{E}\bar{\mathbf{g}}_t$

$$\begin{aligned}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t) - \mathbf{g}_t\|^2 &= \mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t) - \bar{\mathbf{g}}_t + \bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \\ &= \mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t) - \bar{\mathbf{g}}_t\|^2 + \mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \\ &= \mathbb{E}\|\frac{1}{N}\sum_{k=1}^{N}(\nabla F_k(\bar{\mathbf{w}}_t) - \nabla F_k(\mathbf{x}_t^k))\|^2 + \mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \\ &\le L^2\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2\end{aligned} \tag{78}$$

Further, we have

$$-\eta\mathbb{E}\,\langle \nabla F(\bar{\mathbf{w}}_t), \mathbf{g}_t\rangle \le -\frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 - \frac{\eta}{2}\mathbb{E}\|\mathbf{g}_t\|^2 + \frac{\eta}{2}L^2\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \frac{\eta}{2}\mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \tag{79}$$

Summing Eq.79 into Eq.77, we have

$$\mathbb{E}F(\bar{\mathbf{v}}_{t+1}) \leq \mathbb{E}F(\bar{\mathbf{w}}_t) - \frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 + (\frac{L\eta^2}{2} - \frac{\eta}{2})\mathbb{E}\|\mathbf{g}_t\|^2 + \frac{\eta}{2}L^2\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \frac{\eta}{2}\mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \tag{80}$$

$\mathbb{E}\|\mathbf{g}_t\|^2$ can be expanded as follows:

$$\mathbb{E}\|\mathbf{g}_t\|^2 = \mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t + \bar{\mathbf{g}}_t\|^2 = \mathbb{E}\|\bar{\mathbf{g}}_t\|^2 + \mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 \tag{81}$$

Therefore, we have

$$\mathbb{E}F(\bar{\mathbf{v}}_{t+1}) \leq \mathbb{E}F(\bar{\mathbf{w}}_t) - \frac{\eta}{2}\mathbb{E}\|\nabla F(\bar{\mathbf{w}}_t)\|^2 + (\frac{L\eta^2}{2} - \frac{\eta}{2})\mathbb{E}\|\bar{\mathbf{g}}_t\|^2 + \frac{\eta}{2}L^2\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{x}_t^k\|^2 + \frac{L\eta^2}{2}\mathbb{E}\|\bar{\mathbf{g}}_t - \mathbf{g}_t\|^2 \tag{82}$$

Finally, summing Eq.82 into Eq.75 yields the result in Lemma 7.