Sign-Based Gradient Descent With Heterogeneous Data: Convergence and Byzantine Resilience

Richeng Jin, *Member, IEEE*, Yuding Liu, *Student Member, IEEE*, Yufan Huang, Xiaofan He, *Senior Member, IEEE*, Tianfu Wu, *Member, IEEE*, and Huaiyu Dai, *Fellow, IEEE*

***Abstract*— Communication overhead has become one of the major bottlenecks in the distributed training of modern deep neural networks. With such consideration, various quantization-based stochastic gradient descent (SGD) solvers have been proposed and widely adopted, among which SIGNSGD with majority vote shows a promising direction because of its communication efficiency and robustness against Byzantine attackers. However, SIGNSGD fails to converge in the presence of data heterogeneity, which is commonly observed in the emerging federated learning (FL) paradigm. In this article, a sufficient condition for the convergence of the sign-based gradient descent method is derived, based on which a novel magnitude-driven stochastic-sign-based gradient compressor is proposed to address the non-convergence issue of SIGNSGD. The convergence of the proposed method is established in the presence of *arbitrary* data heterogeneity. The Byzantine resilience of sign-based gradient descent methods is quantified, and the error-feedback mechanism is further incorporated to boost the learning performance. Experimental results on the MNIST dataset, the CIFAR-10 dataset, and the Tiny-ImageNet dataset corroborate the effectiveness of the proposed methods.**

***Index Terms*— Byzantine resilience, communication efficiency, data heterogeneity, federated learning (FL), sign-based gradient descent.**

Manuscript received 3 September 2023; accepted 10 December 2023. The work of Richeng Jin was supported in part by the National Natural Science Foundation of China under Grant 62301487, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LQ23F010021, and in part by the Ng Teng Fong Charitable Foundation in the Form of ZJU-SUTD IDEA Grant under Grant 188170-11102. The work of Tianfu Wu was supported in part by the Army Research Office (ARO) under Grant W911NF1810295 and Grant W911NF2210010; and in part by NSF under Grant IIS-1909644, Grant IIS-1822477, Grant CMMI-2024688, and Grant IUSE-201345. The work of Huaiyu Dai was supported in part by the U.S. National Science Foundation under Grant ECCS-2203214. *(Corresponding author: Richeng Jin.)*

Richeng Jin and Yuding Liu are with the Department of Information and Communication Engineering, Zhejiang University, Hangzhou 310007, China, also with the Zhejiang–Singapore Innovation and AI Joint Research Laboratory, Hangzhou 310007, China, and also with the Zhejiang Provincial Key Laboratory of Information Processing, Communication, and Networking (IPCAN), Hangzhou 310007, China (e-mail: richengjin@zju.edu.cn; yudingliu@zju.edu.cn).

Yufan Huang resides in Santa Clara, CA 95051 USA (e-mail: yufan.darren.huang@gmail.com).

Xiaofan He is with the School of Electronic Information, Wuhan University, Wuhan 430000, China (e-mail: xiaofanhe@whu.edu.cn).

Tianfu Wu and Huaiyu Dai are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: twu19@ncsu.edu; hdai@ncsu.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2023.3345367, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2023.3345367

I. Introduction

THE past decade has witnessed the great success that deep neural networks have achieved in modern society. However, training deep neural networks may take weeks due to the large size of the training dataset and the neural network model. One effective way to reduce the training time is to use distributed learning [1], in which the training data and computation are offloaded to the computing machines across the network. Nonetheless, it requires frequent exchange of gradients or model parameters between the workers (i.e., computing machines) and the parameter server, which renders the communication overhead prohibitive.

To alleviate the communication burden of the workers, there have been various gradient quantization methods [2], [3], [4], [5], [6] in the literature, among which the recently proposed SIGNSGD with majority vote [4] is of particular interest due to its robustness and has inspired follow-up works on sign-based gradient descent [7], [8].[1,2] On one hand, in SIGNSGD, during each communication round, only the signs of the gradients and the aggregation results are exchanged between the workers and the parameter server, which leads to around $32\times$ less communication overhead compared to the full-precision distributed SGD. On the other hand, the impact of potential attacks is mitigated since the attackers cannot manipulate the magnitude of the gradients, and therefore SIGNSGD enjoys Byzantine resilience [12]. Nonetheless, it has been shown that SIGNSGD may fail to converge when the data on different workers are heterogeneous [8], which hinders its deployment in practice. In particular, data heterogeneity is commonly observed in the emerging distributed learning paradigm of federated learning (FL).

In this work, we first derive a sufficient condition for the convergence of general sign-based gradient descent methods in which the workers adopt a 1-bit compressor $\mathcal{C}_1(\cdot)$ (which captures SIGNSGD as a special case with $\mathcal{C}_1(\cdot) = \text{sign}(\cdot)$) for gradient quantization, based on which a stochastic-sign-based SGD algorithm is proposed to address the non-convergence of SIGNSGD in the presence of data heterogeneity. More specifically, instead of directly transmitting the signs

[1]Note that we ignore the term "with majority vote" in the following discussions for the ease of presentation.

[2]Another orthogonal approach to reducing communication overhead is compressing the model weights directly (see [9], [10], [11]). In this work, we focus on gradient compression.

of gradients, the workers adopt a two-level stochastic quantization and transmit the signs of the quantized results. We note that different from the existing 1-bit stochastic quantization schemes (e.g., QSGD [2]), the proposed algorithm, termed Sto-SIGNSGD, also uses the majority vote rule in gradient aggregation, which allows the server-to-worker communication to be 1-bit compressed and ensures robustness as well. Then, we prove that Sto-SIGNSGD converges to the neighborhood of the (local) optimum under heterogeneous data distributions. The gap between the converged solution and the (local) optimum diminishes as the number of workers increases. More specifically, the convergence rate of Sto-SIGNSGD in the presence of data heterogeneity approaches that of SIGNSGD with homogeneous data as the number of workers increases.

The Byzantine resilience of the sign-based gradient descent methods is also quantified. More specifically, assuming that there are M normal (benign) workers, it is shown that the Byzantine resilience of the proposed algorithms is upper bounded by $|\sum_{m=1}^{M}(\boldsymbol{g}_m^{(t)})_i|/b_i, \forall i$, where $(\boldsymbol{g}_m^{(t)})_i$ is the ith entry of worker m's gradient at iteration t and $b_i \geq \max_m(\boldsymbol{g}_m^{(t)})_i$ is some design parameter. Particularly, b_i depends on the data heterogeneity (through $\max_m(\boldsymbol{g}_m^{(t)})_i$). As a special case, the proposed algorithms can tolerate $M-1$ Byzantine workers when the normal workers can access the same dataset (i.e., $(\boldsymbol{g}_m^{(t)})_i = (\boldsymbol{g}_j^{(t)})_i, \forall 1 \leq j, m \leq M$), which recovers the result of SIGNSGD.

We further extend the proposed algorithm to its error-feedback variant, termed EF-Sto-SIGNSGD. The server keeps track of the error induced by the majority vote operation and compensates for the error in the next communication round. Both the convergence and the Byzantine resilience are established. Extensive simulations are performed to demonstrate the effectiveness of all the proposed algorithms. To this end, this article makes three main contributions to the field of sign-based methods as follows.

1) We derive a sufficient condition for the convergence of sign-based gradient descent methods in the presence of data heterogeneity, based on which we propose Sto-SIGNSGD, which utilizes a stochastic-sign-based gradient compressor to overcome the convergence issue of SIGNSGD. We further improve the learning performance of the proposed algorithm by incorporating the error-feedback method.
2) We prove that Sto-SIGNSGD converges to the neighborhood of the (local) optimum in the heterogeneous data distribution scenario. As the number of workers increases, the convergence rate approaches that of SIGNSGD with homogeneous data distribution (and therefore distributed SGD with full precision).
3) We theoretically quantify the Byzantine resilience of the sign-based gradient descent methods, which depends on the heterogeneity of the local datasets of the workers as well as the capability of the attackers.

II. Related Works

A. Gradient Quantization

To accommodate the need of communication efficiency in distributed learning, various gradient compression methods have been proposed. Most of the existing works focus on unbiased compression that keeps the expectation of the shared parameters unchanged [13], [14]. QSGD [2], TernGrad [3], ATOMO [15], FedPAQ [6], and FedCOM [16] propose to use unbiased stochastic quantization schemes. Due to the unbiased nature of the adopted quantization methods, the convergence of the corresponding algorithms can be established. Sattler et al. [17] and Basu et al. [18] combine sparsification and quantization to further improve the compression rate. However, all these methods have not shown to be robust against Byzantine attacks.

The idea of sharing the signs of gradients in SGD can be traced back to 1-bit SGD [19]. Despite that sign-based quantization is biased in nature, [4], [12], [20] show theoretical and empirical evidence that sign-based gradient schemes can converge well in the homogeneous data distribution scenario. Safaryan and Richtárik [7] show the convergence of SIGNSGD given the assumption that the probability of wrong aggregation is less than 1/2. In the heterogeneous data distribution case, [8] shows that the convergence of SIGNSGD is not guaranteed and proposes to add carefully designed noise to ensure a convergence rate of $O(d^{(3/4)}/T^{(1/4)})$. However, their analysis assumes second-order differentiability of the noise probability density function and cannot be applied to some commonly used noise distributions (e.g., uniform and Laplace distributions). Safaryan and Richtárik [7] propose stochastic sign descent with momentum (SSDM) to accommodate the data heterogeneity, and another independent work proposes FedCOMGATE [16]. Compared to SSDM and FedCOMGATE, our proposed Stochastic-Sign SGD is stateless and therefore more suitable for cross-device FL as discussed in [21]. Moreover, the Byzantine resilience of sign-based method is further quantified, which is not considered in [7] and [16].

B. Error-Compensated SGD

Instead of directly using the biased approximation of the gradients, [19] corrects the quantization error by adding error feedback in subsequent updates and observes almost no accuracy loss empirically. Wu et al. [5] propose the error-compensated quantized SGD in quadratic optimization and prove its convergence for unbiased stochastic quantization. Stich et al. [22] show the convergence of the proposed error-compensated algorithm for strongly convex loss functions and [23] proves the convergence of sparsified gradient methods with error compensation for both convex and non-convex loss functions. Karimireddy et al. [24] propose EF-SIGNSGD, which combines the error compensation methods and SIGNSGD; however, only the single worker scenario is considered. Zheng et al. [25] further extend it to the multi-worker scenario and the convergence is established. However, it is required in these two works that the compression error cannot be larger than the original vector, i.e., there exists some constant $\delta > 0$ such that $||\mathcal{C}(\boldsymbol{x}) - \boldsymbol{x}||_2^2 \leq (1-\delta)||\boldsymbol{x}||_2^2$, which is not true for $\mathcal{C}(\cdot) = \text{sign}(\cdot)$. As a result, [24] and [25] require the workers to share a scaled version of the signs, which ruins the Byzantine resilience of SIGNSGD. Tang et al. [26]

consider more general compressors and prove the convergence under the assumption that the compressors have a bounded magnitude of the error. Nonetheless, none of these existing works take Byzantine resilience into consideration.

C. Byzantine Tolerant SGD in Heterogeneous Environment

There have been significant research interests in developing SGD-based Byzantine tolerant algorithms, most of which consider homogeneous data distribution, e.g., Krum [27], ByzantineSGD [28], the median-based algorithms [29], and coding-based methods [30], [31]. Bernstein et al. [12] show that SIGNSGD can tolerate up to half "blind" Byzantine workers who determine how to manipulate their gradients before observing the gradients. These robust aggregators utilize the statistics among normal workers to detect the outliers. However, they cannot deal with heterogeneous data since the server may fail to identify whether the outliers are due to data heterogeneity or attacks.

To accommodate the need for robust FL, some Byzantine-tolerant algorithms that can deal with heterogeneous data distributions have been developed. Li et al. [32] propose to incorporate a regularized term with the objective function. However, it requires strong convexity and can only converge to the neighborhood of the optimal solution. Xie et al. [33] use trimmed mean to aggregate the shared parameters. Data and Diggavi [34] adopt the RAGE algorithm in [35] for robust aggregation. Karimireddy et al. [36] adopt the Bucketing method to alleviate the data heterogeneity issue. Karimireddy et al. [37] propose a centered clipping-based method. Liu et al. [38] explore the redundancy in local loss functions for Byzantine resilience. Li et al. [39] propose a multitask learning framework that learns personalized models to achieve robustness and fairness. Xu et al. [40] combine norm-based threshold filtering and sign-based clustering. Wan et al. [41] propose a multiarmed bandit-based approach to distinguish attackers from normal workers. Despite that these methods provide certain degrees of Byzantine resilience, none of them take communication efficiency into consideration.

Ghosh et al. [42] combine gradient norm threshold filtering and gradient compression. Nonetheless, it requires knowledge about the fraction of Byzantine attackers. Zhu and Ling [43] combine the geometric median-based methods with gradient difference compression and the stochastic average gradient algorithm (SAGA). However, it requires strong convexity in the convergence analysis.

Another line of work, e.g., [44], [45], [46], relies on an auxiliary dataset on the server side to identify the Byzantine attackers, which, however, may not be feasible in practice.

III. Problem Formulation

In this article, we consider a typical distributed optimization problem with M normal workers as in [47]. Formally, the goal is to minimize the finite-sum objective of the form

$$\min_{w\in\mathbb{R}^d} F(w), \quad \text{where } F(w) \stackrel{\text{def}}{=} \frac{1}{M}\sum_{m=1}^{M} f_m(w). \tag{1}$$

For a machine learning problem, we have a sample space $I = X \times Y$, where X is a space of feature vectors and Y is a label space. Given the hypothesis space $\mathcal{W} \subseteq \mathbb{R}^d$, we define a loss function $l : \mathcal{W} \times I \to \mathbb{R}$ which measures the loss of prediction on the data point $(x, y) \in I$ made with the hypothesis vector $w \in \mathcal{W}$. In such a case, $f_m(w)$ is a local function defined by the local dataset of worker m and the hypothesis w. More specifically,

$$f_m(w) = \frac{1}{|D_m|}\sum_{(x_n,y_n)\in D_m} l(w;(x_n,y_n)) \tag{2}$$

where $|D_m|$ is the size of worker m's local dataset D_m. If the training data are distributed over the workers uniformly at random, then we would have $\mathbb{E}[f_m(w)] = F(w)$, where the expectation is over the training data distribution. This is the *homogeneous data distribution assumption* typically made in distributed optimization [47]. In many FL applications, however, the local datasets of the workers are heterogeneously distributed.

We consider a parameter server paradigm. At each communication round t, each worker m forms a batch of training samples, based on which it computes and transmits the stochastic gradient $\boldsymbol{g}_m^{(t)}$ as an estimate to the true gradient $\nabla f_m(w_m^{(t)})$. When worker m evaluates the gradient over its whole local dataset, we have $\boldsymbol{g}_m^{(t)} = \nabla f_m(w_m^{(t)})$. After receiving the gradients from the workers, the server performs aggregation and sends the aggregated gradient back to the workers. Finally, the workers update their local model weights using the aggregated gradient. In this sense, the classic stochastic gradient descent (SGD) algorithm [48] performs iterations of the form

$$w_m^{(t+1)} = w_m^{(t)} - \frac{\eta}{M}\sum_{m=1}^{M} \boldsymbol{g}_m^{(t)}. \tag{3}$$

In this case, since all the workers adopt the same update rule using the aggregated gradient, $w_m^{(t)}$'s are the same for all the workers. Therefore, in the following discussions, we omit the worker index m for ease of presentation. To accommodate the requirement of communication efficiency in FL, we adopt the popular idea of gradient quantization and assume that each worker m quantizes the gradient with a stochastic 1-bit compressor $\mathcal{C}_1(\cdot)$ and sends $\mathcal{C}_1(\boldsymbol{g}_m^{(t)})$ instead of its actual local gradient $\boldsymbol{g}_m^{(t)}$. Combining with the idea of majority vote in [4], the corresponding algorithm is presented in Algorithm 1.

Algorithm 1 Sign-Based Gradient Descent With Majority Vote

Input: learning rate η, current hypothesis vector $w^{(t)}$, M workers each with an independent gradient $\boldsymbol{g}_m^{(t)}$, the 1-bit compressor $\mathcal{C}_1(\cdot)$.

on server:

pull $\mathcal{C}_1(\boldsymbol{g}_m^{(t)})$ **from** worker m.

push $\tilde{\boldsymbol{g}}^{(t)} = sign\big(\frac{1}{M}\sum_{m=1}^{M}\mathcal{C}_1(\boldsymbol{g}_m^{(t)})\big)$ **to** all the workers.

on each worker:

update $w^{(t+1)} = w^{(t)} - \eta\tilde{\boldsymbol{g}}^{(t)}$.

Intuitively, the performance of Algorithm 1 is limited by the probability of wrong aggregation, i.e.,

$$\text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}\mathcal{C}_1(\boldsymbol{g}_m^{(t)})\right) \neq \text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}\nabla f_m(w^{(t)})\right). \quad (4)$$

In SIGNSGD, $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{sign}(\boldsymbol{g}_m^{(t)})$ and (4) holds with a high probability when the local gradients $\nabla f_m(w^{(t)})$'s are highly different across workers (i.e., heterogeneous data distributions), which prevents its convergence. In this work, we propose a compressor sto-sign, which guarantees that (4) occurs with a probability that is strictly smaller than 0.5 and therefore the convergence of Algorithm 1 follows.

A. Threat Model

In addition to the M normal workers, it is assumed that there exist B Byzantine attackers, and its set is denoted as $\mathcal{B}$. Instead of using sto-sign, the Byzantine attackers can use an arbitrary compressor denoted by byzantine-sign. In this work, we consider the scenario in which each Byzantine attacker j follows the same procedure as the normal workers and obtains $\boldsymbol{g}_j^{(t)}, \forall j \in \mathcal{B}$. In addition, we consider general attackers that share the opposite signs of the true gradients [i.e., byzantine-sign$(\boldsymbol{g}_j^{(t)}) \neq \text{sign}((1/M)\sum_{m=1}^{M}\nabla f_m(w^{(t)}))$] with certain probabilities. More specifically, we denote the probability that Byzantine attacker j shares the wrong sign on the ith coordinate during the tth communication round as $q_{j,i}^{(t)} \in [0, 1]$.

IV. Algorithms and Convergence Analysis

In this section, we derive a sufficient condition for the convergence of sign-based gradient descent methods in the presence of data heterogeneity. For ease of presentation, we first consider a scalar case, which can be readily generalized to the vector case by applying the results independently on each coordinate. Moreover, we consider the scenario in which all the workers are benign. The Byzantine resilience of sto-sign will be discussed in Section V. The proofs of all the theoretical results are provided in Appendix A in the supplementary material.

Theorem 1 (Probability of Wrong Aggregation for Generic Sign-Based Compressor): Let $u_1, u_2, \ldots, u_M$ be M known and fixed real numbers and consider binary random variables $\hat{u}_m$, $1 \leq m \leq M$. Suppose $\bar{p} = (1/M)\sum_{m=1}^{M} P(\text{sign}((1/M)\sum_{m=1}^{M} u_m) \neq \hat{u}_m) < (1/2)$, then

$$P\left(\text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}\hat{u}_m\right) \neq \text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}u_m\right)\right) \leq \left[4\bar{p}(1-\bar{p})\right]^{\frac{M}{2}}. \quad (5)$$

Remark 1: Let $u_m = \nabla f_m(w^{(t)})_i$ be the ith coordinate of worker m's true local gradient and $\hat{u}_m = \text{sign}(\boldsymbol{g}_m^{(t)})_i$ the ith coordinate of the 1-bit estimator, $[4\bar{p}(1-\bar{p})]^{(M/2)} < 1/2$ is a sufficient condition that the probability of wrong aggregation on the ith coordinate is less than $1/2$, where $\bar{p} = (1/M)\sum_{m=1}^{M} P(\text{sign}(\nabla F(w^{(t)})_i) \neq \text{sign}(\boldsymbol{g}_m^{(t)})_i)$ is the average probability of wrong signs that characterizes the impact of data heterogeneity. Essentially, as long as $\bar{p} < 1/2$, there exists some M such that the probability of wrong aggregation is less than $1/2$, based on which the convergence of the sign-based gradient descent method can be established (see Theorem 2). Different from [7, Th. 3] that assumes homogeneous data and the same probability of wrong signs across the workers, we only require the average probability of wrong signs $\bar{p} < 1/2$ and thus can deal with heterogeneous data. Based on Theorem 1, we further propose a stochastic-sign-based compressor to overcome the non-convergence issue of SIGNSGD when $\bar{p} \geq 1/2$. Such a result is crucial in the heterogeneous data distribution scenario since the probability of wrong signs can be very different across workers.

In the above discussion, we show that SIGNSGD works for a sufficiently large M given that the average probability of wrong signs $\bar{p} < 1/2$. In the scenarios with more severe data heterogeneity where $\bar{p} \geq 1/2$, however, its convergence is not guaranteed. The reason is that, in SIGNSGD, the magnitude information of $\boldsymbol{g}_m^{(t)}$ is lost. In the following, we propose a two-level stochastic compressor sto-sign, which utilizes the magnitude information and therefore can deal with an *arbtrarily* heterogeneous data distribution. Formally, sto-sign is defined as follows.

Definition 1 (The Proposed Two-Level Stochastic Gradient Quantization): For any given gradient $\boldsymbol{g}_m^{(t)}$, the compressor sto-sign outputs sto-sign$(\boldsymbol{g}_m^{(t)}, \boldsymbol{b})$, where $\boldsymbol{b}$ is a vector of design parameters that controls the level of stochasticity. The ith entry of sto-sign$(\boldsymbol{g}_m^{(t)}, \boldsymbol{b})$ is given by

$$\text{sto-sign}(\boldsymbol{g}_m^{(t)}, \boldsymbol{b})_i = \begin{cases} 1, & \text{with probability } \dfrac{b_i + (\boldsymbol{g}_m^{(t)})_i}{2b_i} \\ -1, & \text{with probability } \dfrac{b_i - (\boldsymbol{g}_m^{(t)})_i}{2b_i} \end{cases} \quad (6)$$

where $(\boldsymbol{g}_m^{(t)})_i$ and $b_i \geq \max_m |(\boldsymbol{g}_m^{(t)})_i|$ are the ith entry of $\boldsymbol{g}_m^{(t)}$ and $\boldsymbol{b}$, respectively.

In sto-sign, the magnitude information is encoded in the mapping probabilities in (6). By introducing the stochasticity, sto-sign essentially makes use of the magnitude information (without incurring additional communication overhead) such that the probability of wrong aggregation can be theoretically bounded below $1/2$ for an arbitrary realization of $\boldsymbol{g}_m^{(t)}$'s, as shown in Corollary 1.

Corollary 1 (Probability of Wrong Aggregation for sto-sign): Let $u_1, u_2, \ldots, u_M$ be M known and fixed real numbers and consider binary random variables $\hat{u}_m = \text{sto-sign}(u_m, b)$, $1 \leq m \leq M$. We have $\bar{p}_{\text{sto}} = (1/M)\sum_{m=1}^{M} P(\text{sign}((1/M)\sum_{m=1}^{M} u_m) \neq \hat{u}_m) = ((bM - |\sum_{m=1}^{M} u_m|)/2bM) \leq (1/2)$, and

$$P\left(\text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}\hat{u}_m\right) \neq \text{sign}\left(\frac{1}{M}\sum_{m=1}^{M}u_m\right)\right) \leq \left(1 - \left(\frac{|\sum_{m=1}^{M} u_m|}{bM}\right)^2\right)^{\frac{M}{2}}. \quad (7)$$

Remark 2 (Selection of $\boldsymbol{b}$): Some discussions on the choice of the vector $\boldsymbol{b}$ in (6) are in order. We take the ith entry of $\boldsymbol{b}$ as an example. The ith entry of the gradient $\boldsymbol{g}_m^{(t)}$ corresponds to u_m in Corollary 1, and the average probability of wrong signs $\bar{p}_{\text{sto}} < (1/2)$ when the sum of gradients $\sum_{m=1}^{M} \boldsymbol{g}_m^{(t)}$ is

non-zero, which therefore addresses the non-convergence issue of SIGNSGD. According to Definition 1, $b_i \geq \max_m |(\boldsymbol{g}_m^{(t)})_i|$ and $0 \leq |\sum_{m=1}^M (\boldsymbol{g}_m^{(t)})_i|/(b_i M) \leq 1$. Since $(1-x^2)$ is a decreasing function of x when $0 \leq x \leq 1$, the bound in (7) is minimized when $b_i = \max_m |(\boldsymbol{g}_m^{(t)})_i|$. In practice, since $\max_m |(\boldsymbol{g}_m^{(t)})_i|$ is unknown, the selection of an appropriate $\boldsymbol{b}$ is an interesting problem deserving further investigation. Considering that gradient clipping is often adopted in training large-scale deep models to prevent gradient exploding, the selection of $\boldsymbol{b}$ can be accommodated accordingly. In our experiments, it is observed that a fixed vector $\boldsymbol{b}$ achieves satisfactory performance.

In order to facilitate the discussion, the following commonly adopted assumptions are made.

Assumption 1 (Lower Bound): For all $\boldsymbol{x}$ and some constant F^*, we have objective value $F(\boldsymbol{x}) \geq F^*$.

Assumption 2 (Smoothness): $\forall \boldsymbol{y}, \boldsymbol{x}$, we require for some non-negative constants $L_1, L_2, \ldots, L_d$

$$F(\boldsymbol{y}) \leq F(\boldsymbol{x}) + \langle \nabla F(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + \sum_{i=1}^d \frac{L_i}{2}(y_i - x_i)^2 \quad (8)$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product. Denote $L = \sum_{i=1}^d L_i$.

Assumption 3 (Variance Bound): For any worker m, the stochastic gradient oracle gives an independent unbiased estimate g_m that has coordinate bounded variance

$$\mathbb{E}[\boldsymbol{g}_m] = \nabla f_m(w), \mathbb{E}[((\boldsymbol{g}_m)_i - \nabla f_m(w)_i)^2] \leq \sigma_i^2 \quad (9)$$

for a vector of non-negative constants $\bar{\sigma} = [\sigma_1, \ldots, \sigma_d]$.

*Theorem 2 (*S *to-*SIGN *SGD):* Suppose Assumptions 1–3 are satisfied, and the learning rate is set as $\eta = (1/(\sqrt{TL}))$. Then by running Algorithm 1 with $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{sto-sign}(\boldsymbol{g}_m^{(t)}, \boldsymbol{b})$ (termed Sto-SIGNSGD) and mini-batch size T for T iterations, we have

$$\frac{1}{T}\sum_{t=1}^T \sum_{i=1}^d c_{t,i} |\nabla F(w^{(t)})_i| \leq \frac{(F(w^{(0)}) - F^*)\sqrt{L}}{\sqrt{T}} + \frac{\sqrt{L}}{2\sqrt{T}} + 2\frac{||\sigma||_1}{\sqrt{MT}} \quad (10)$$

in which $c_{t,i} = 1 - 2\mathbb{E}[\big(1 - ((|(1/M)\sum_{m=1}^M (\boldsymbol{g}_m^{(t)})_i|^2)/b_i^2)\big)^{(M/2)}]$, where the expectation is over the randomness in stochastic gradients $\boldsymbol{g}_m^{(t)}$'s.

Given the results in Corollary 1, the proof of Theorem 2 follows the well-known strategy of relating the norm of the gradient to the expected improvement of the global objective in a single iteration. Accumulating the improvement over the iterations yields the convergence rate of the algorithm.

Remark 3: Compared to the convergence rate of SIGNSGD, the major differences are: 1) no assumption on the sampling noise is needed and 2) the coefficient $c_{t,i} < 1$, in which $\mathbb{E}[(1 - |(1/M)\sum_{m=1}^M (\boldsymbol{g}_m^{(t)})_i|^2/b_i^2)^{(M/2)}]$ measures the probability of wrong aggregation. When $|(1/M)\sum_{m=1}^M (\boldsymbol{g}_m^{(t)})_i| \neq 0$, $\lim_{M\to\infty} c_{t,i} = 1$, which suggests that the convergence rate of Sto-SIGNSGD with heterogeneous data approaches that of SIGNSGD with homogeneous data distribution in the large M regime as in FL (which can be in the order of millions [49]). Moreover, it is worth mentioning that large mini-batch sizes are not necessary for convergence, please refer to the results in Appendix B in the supplementary material.

Theorem 2 shows that the detrimental impact of data heterogeneity on SIGNSGD is addressed by adopting the sto-sign compressor in the large M regime. For a small M, if $\boldsymbol{g}_m^{(t)} = \nabla F(w^{(t)})$, $c_{t,i}$ decreases as $|\nabla F(w^{(t)})|$ decreases, and Theorem 2 indicates that Algorithm 1 converges to the point where the probability of wrong aggregation is $(1/2)$ (i.e., $c_{t,i} = 0$). As a result, there is a gap between the converged solution and the (local) optimum, which vanishes as M increases. This is because the bound in (7) may not be tight enough for small $|\nabla F(w^{(t)})|$. In the following, we address this issue for small M by showing the convergence of Sto-SIGNSGD with large b_i's. For ease of presentation, we make the following assumption on the number of workers.

Assumption 4: The total number of workers is odd.

Theorem 3: Suppose Assumptions 1, 2, and 4 are satisfied, $|\nabla F(w^{(t)})_i| \leq Q, \forall 1 \leq i \leq d, 1 \leq t \leq T$, and the learning rate is set as $\eta = (1/\sqrt{TL})$. Then by running Algorithm 1 with $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{sto-sign}(\nabla f_m(w^{(t)}), \boldsymbol{b})$ and $b_i = T^{1/6}L^{1/6}, \forall i$, for T iterations, we have

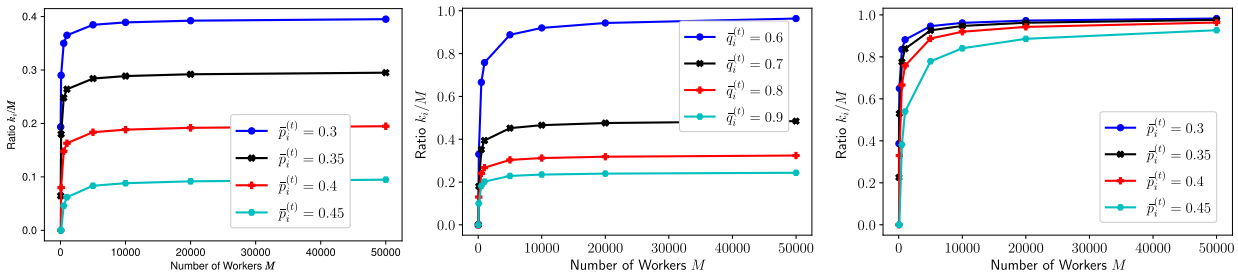
$$\frac{1}{T}\sum_{t=1}^T \sum_{i=1}^d |\nabla F(w^{(t)})_i|^2 \leq \frac{\sqrt{2\pi}(M-1)^{\frac{3}{2}}}{2(M^2-3M)}\left[\frac{(F(w^{(0)}) - F^*)L^{2/3}}{T^{1/3}} + \frac{L^{2/3}}{2T^{1/3}} + \frac{2}{T}\sum_{t=1}^T\sum_{i=1}^d |\nabla F(w^{(t)})_i| O\left(\frac{1}{T^{1/3}L^{1/3}}\right)\right]. \quad (11)$$

Remark 4: We note that Assumption 4 is introduced to ensure that there is always a winner in the majority vote as in [8]. This assumption can be readily relaxed if the parameter server breaks the tie uniformly at random if there is no winner when M is even. Moreover, Theorem 3 assumes that the workers evaluate the local gradients over the whole local dataset, and the corresponding results on the mini-batch SGD are presented in Appendix B in the supplementary material.

Remark 5: Given the same assumptions, [8] obtains a slower convergence rate of $O(1/T^{1/4})$. Theorem 3 shows a convergence rate of $O(1/(M^{1/2}T^{1/3}))$, which is still slower than that of the vanilla distributed SGD with full precision. This is because Theorem 3 adopts a large b_i, which leads to a probability of wrong aggregation that approaches $1/2$ as T increases. In our experiments, Algorithm 1 attains satisfactory performance for fixed b_i's. Moreover, this concern is also alleviated when the error-feedback mechanism is incorporated (see Theorem 5).

V. Byzantine Resilience

In this section, the Byzantine resilience of the sign-based gradient descent method is investigated. In addition to the M normal workers, it is assumed that there exist B Byzantine attackers, and its set is denoted as $\mathcal{B}$. Instead of using sto-sign, the Byzantine attackers can use an arbitrary compressor denoted by byzantine-sign. We consider general attackers



(a) (b) (c)

Fig. 1. (a) $\bar{q}_i^{(t)} = 1$. (b) $\bar{p}_i^{(t)} = 0.4$. (c) $\bar{q}_i^{(t)} = 1 - \bar{p}_i^{(t)}$.

that share the opposite signs of the true gradients [i.e., byzantine-sign$(\boldsymbol{g}_j^{(t)}) \neq \text{sign}\big((1/M)\sum_{m=1}^{M} \nabla f_m(w^{(t)})\big)$] with certain probabilities.

Note that the convergence of Algorithm 1 is determined by the probability of wrong aggregation (i.e., more than half of the workers share the wrong signs). Recall that $q_{j,i}^{(t)}$ denotes the probability that Byzantine attacker j shares the wrong sign on the ith coordinate during the tth communication round. In the following analysis, let $Z_i^{(t)}$ denote the number of workers (including the attackers) that share (quantized) gradients with different signs from the true gradient $\nabla F(w^{(t)})$ on the ith coordinate. Then, $Z_i^{(t)}$ is a Poisson binomial variable, and we need the probability of wrong aggregation less than $(1/2)$ [i.e., $P(Z_i^{(t)} > (M/2)) < (1/2)$] to ensure convergence. Therefore, we can prove the following theorem.

Theorem 4 (Byzantine Resilience of Generic Sign-based Gradient Descent Methods): During tth communication round, let $(1/M)\sum_{m=1}^{M} P(\text{sign}(\nabla F(w^{(t)}))_i \neq \mathcal{C}_1(\boldsymbol{g}_m^{(t)})_i) = \bar{p}_i^{(t)} < (1/2)$, then Algorithm 1 can at least tolerate k_i Byzantine attackers (i.e., the probability of wrong aggregation is smaller than $(1/2)$, which is a sufficient condition for the convergence of Algorithm 1) on the ith coordinate of the gradient with $(1/k_i)\sum_{j\in\mathcal{B}} q_{j,i}^{(t)} = \bar{q}_i^{(t)} > (1/2)$ if there exist positive constants a and c such that

$$\left[e^{a}\bar{q}_i^{(t)} + e^{-a}\left(1-\bar{q}_i^{(t)}\right)\right]^{k_i}\left[e^{a}\bar{p}_i^{(t)} + e^{-a}\left(1-\bar{p}_i^{(t)}\right)\right]^{M} \leq \frac{1-c}{2}. \quad (12)$$

Overall, the number of Byzantine workers that the algorithms can tolerate is given by $\min_{1\leq i\leq d} k_i$.

Corollary 2 (Robustness Against the Most Powerful Attackers): When $\bar{q}_i^{(t)} = 1$, i.e., the attackers always share the wrong signs (which is considered the worst case), (12) is equivalent to

1) $k_i < M - 2M\bar{p}_i^{(t)}$;
2) there exists some positive constant c such that

$$\left[\frac{(M-k_i)(1-\bar{p}_i^{(t)})}{(M+k_i)\bar{p}_i^{(t)}}\right]^{\frac{k_i}{2}}\left(\sqrt{\frac{M-k_i}{M+k_i}} + \sqrt{\frac{M+k_i}{M-k_i}}\right)^{M} \times\left[\bar{p}_i^{(t)}\left(1-\bar{p}_i^{(t)}\right)\right]^{\frac{M}{2}} \leq \frac{1-c}{2}. \quad (13)$$

Remark 6: Theorem 4 and Corollary 2 hold for any sign-based compressor $\mathcal{C}_1(\cdot)$. When $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{sto-sign}(\nabla f_m(w^{(t)}), \boldsymbol{b})$, we have $\bar{p}_i^{(t)} = ((b_i M - |\sum_{m=1}^{M} \nabla f_m(w^{(t)})_i|)/2b_i M)$ and the first condition above is reduced to $k_i \leq ((|\sum_{m=1}^{M} \nabla f_m(w^{(t)})_i|)/b_i)$. If we set $b_i = \max_m |\nabla f_m(w^{(t)})_i|$ as in Section IV, the first condition in Corollary 2 gives $k_i < (|\sum_{m=1}^{M} \nabla f_m(w^{(t)})_i|)/(\max_m |\nabla f_m(w^{(t)})_i|)$, which means that the Byzantine resilience depends on the heterogeneity of the local datasets. In an ideal scenario where the workers have the same local datasets, i.e., $\nabla f_m(w^{(t)})_i = \nabla f_n(w^{(t)})_i, \forall m, n$, Corollary 2 gives $\bar{p}_i^{(t)} = 0$ and $k_i < M$. Therefore, it can tolerate $M-1$ Byzantine workers.

For the second condition in Corollary 2, it can be observed that for any finite and fixed k_i, the left-hand side of (13) approaches $((1-\bar{p}_i^{(t)})/(\bar{p}_i^{(t)}))^{(k_i/2)}[4\bar{p}_i^{(t)}(1-\bar{p}_i^{(t)})]^{(M/2)}$ as M goes to infinity. Fig. 1 numerically examines the ratio (k_i/M) that satisfies (12). It is shown from Fig. 1(a) that, in the worst case scenario in which $q_i^{(t)} = 1$, the number of Byzantine attackers that Algorithm 1 can tolerate increases as the number of normal workers M increases and approaches $k_i = (1-2\bar{p}_i^{(t)})M$, which is given by the first condition in Corollary 2.

Fig. 1(a) and (b) shows that for fixed $\bar{p}_i^{(t)}$ and $\bar{q}_i^{(t)}$, (k_i/M) approaches $(1-2\bar{p}_i^{(t)})/(2\bar{q}_i^{(t)}-1)$ as the number of benign workers M increases. That is, stronger normal workers and weaker attackers suggest better Byzantine resilience. With such consideration, we further examine the attackers with average capability defined as follows.

Definition 2: An attacker j is of average capability if $\Pr(\text{sign}(\nabla F(w^{(t)}))_i \neq \text{byzantine-sign}(\boldsymbol{g}_j^{(t)})_i) = 1-\bar{p}_i^{(t)}$, i.e., attacker j shares the wrong sign with a probability of $1-\bar{p}_i^{(t)}$, in which $\bar{p}_i^{(t)}$ is the average probability of wrong signs for the benign workers.

Remark 7: We provide one possible attacker of average capability as follows: 1) the attacker's local gradients $\boldsymbol{g}_j^{(t)}$ satisfy $\Pr(\text{sign}(\nabla F(w^{(t)}))_i \neq \text{sign}(\boldsymbol{g}_j^{(t)})_i) = \bar{p}_i^{(t)}$, i.e., it estimates the signs of the true gradients $\nabla F(w^{(t)})$ incorrectly with a probability of $\bar{p}_i^{(t)}$, in which $\bar{p}_i^{(t)}$ is the average probability of wrong signs over the benign workers and 2) it shares the opposite of its local gradients, i.e., $-\text{sign}(\boldsymbol{g}_j^{(t)})_i$, with the parameter server. It is worth mentioning that the attackers considered in [12] are of average capability.

For attackers with average capability, we have $\bar{q}_i^{(t)} = 1-\bar{p}_i^{(t)}$. It can be observed from Fig. 1(c) that as the number of workers M increases, the ratio (k_i/M) increases and approaches 1, i.e., half of the total workers can be Byzantine attackers with average capability.

Remark 8: When the normal workers adopt the 1-bit compressor $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{sto-sign}(\nabla f_m(w^{(t)}), \boldsymbol{b})$, the average probability of wrong signs $\bar{p}_i^{(t)} = ((bM - |\sum_{m=1}^{M} \nabla f_m(w^{(t)})_i|)/2bM)$. In this case, $\bar{q}_i^{(t)} = 1-\bar{p}_i^{(t)}$ corresponds

to the scenario in which the attackers also adopt the sto-sign compressor (and flip the signs before sharing them with the parameter server), and their local gradients satisfy $\sum_{j=M+1}^{M+k_i} \boldsymbol{g}_j^{(t)} = \sum_{m=1}^{M} \nabla f_m(w^{(t)})$. That being said, the union of the local datasets of the attackers is the same as that of the normal workers.

VI. Error-Feedback Variant

To further improve the performance of Algorithm 1, we incorporate the error-feedback technique and propose its error-feedback variant (i.e., Algorithm 2), where the server utilizes an α-approximate compressor $\mathcal{C}(\cdot)$ (i.e., $||\mathcal{C}(\boldsymbol{x})-\boldsymbol{x}||_2^2 \leq (1-\alpha)||\boldsymbol{x}||_2^2, \forall \boldsymbol{x}$ [24]) and keeps track of the corresponding compression error. Algorithm 2 with $\mathcal{C}_1(\cdot) = \text{sto-sign}(\cdot)$ is termed EF-Sto-SIGNSGD. To facilitate the analysis, we further introduce the commonly used smoothness assumption as follows.

Assumption 5 (Smoothness): $\forall \boldsymbol{x}, \boldsymbol{y}$, we require for some non-negative constant $\tilde{L}$

$$F(\boldsymbol{x}) \leq F(\boldsymbol{y}) + \langle \nabla F(\boldsymbol{y}), \boldsymbol{x}-\boldsymbol{y} \rangle + \frac{\tilde{L}}{2}||\boldsymbol{x}-\boldsymbol{y}||_2^2 \tag{14}$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product.

The convergence and Byzantine resilience of Algorithm 2 are shown below.

Algorithm 2 Error-Feedback Stochastic-Sign SGD With Majority Vote

Input: learning rate η, current hypothesis vector $w^{(t)}$, current residual error vector $\tilde{\boldsymbol{e}}^{(t)}$, M workers each with an independent gradient $\boldsymbol{g}_m^{(t)} = \nabla f_m(w^{(t)})$, the 1-bit compressor $\mathcal{C}_1(\cdot)$.

on server:

pull $\mathcal{C}_1(\boldsymbol{g}_m^{(t)})$ **from** worker m.

push $\tilde{\boldsymbol{g}}^{(t)} = \mathcal{C}\big(\frac{1}{M}\sum_{m=1}^{M}\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) + \tilde{\boldsymbol{e}}^{(t)}\big)$ **to** all the workers,

update residual error

$$\tilde{\boldsymbol{e}}^{(t+1)} = \frac{1}{M}\sum_{m=1}^{M}\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) + \tilde{\boldsymbol{e}}^{(t)} - \tilde{\boldsymbol{g}}^{(t)}. \tag{15}$$

on each worker:

update $w^{(t+1)} = w^{(t)} - \eta\tilde{\boldsymbol{g}}^{(t)}$.

Theorem 5 (Convergence of EF-Sto-SIGNSGD): When Assumptions 1, 3, and 5 are satisfied, by running Algorithm 2 with $\eta = (1/\sqrt{Td})$, mini-batch size of T, $\mathcal{C}_1(\boldsymbol{g}_m^{(t)}) = \text{Sto-SIGN}(\boldsymbol{g}_m^{(t)}, \boldsymbol{b})$ and $\boldsymbol{b} = b \cdot \mathbf{1}$ for T iterations, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\frac{||\nabla F(w^{(t)})||_2^2}{b} \leq \frac{(F(w_0)-F^*)\sqrt{d}}{\sqrt{T}} + \frac{(1+\tilde{L}+\tilde{L}^2\beta)\sqrt{d}}{\sqrt{T}} + 2\frac{||\sigma||_1}{\sqrt{MT}} \tag{16}$$

where β is some positive constant.

Remark 9: In our experiments, the server adopts the compressor $\mathcal{C}(\boldsymbol{x}) = (||\boldsymbol{x}||_1/d)\text{sign}(\boldsymbol{x})$, which is an α-approximate compressor [24]. In this case, the server needs to share $(||\boldsymbol{x}||_1/d)$ with the workers. This incurs an additional communication overhead of 32 bits, which is negligible.

Remark 10: The proposed method is different from scaled SIGNSGD [24], [25] in three aspects.

1) Scaled SIGNSGD is not robust against re-scaling attacks. More specifically, since all the workers are supposed to share the scaled norm of the gradients, the attackers can ruin the training process by manipulating the magnitudes of the gradients (e.g., sharing arbitrarily large magnitudes). In our scheme, all the workers share the stochastic signs, and the parameter server takes the majority vote during the aggregation stage. The impact of the attackers is therefore limited, which provides Byzantine resilience.
2) The error-feedback mechanism in [24] and [25] requires additional memory on the worker side, while the convergence of scaled SIGNSGD without error feedback is unknown. In our scheme, error feedback is only used on the server's side.
3) The convergence of scaled SIGNSGD with error feedback is only established in the homogeneous data distribution scenario.

We obtain the Byzantine resilience of Algorithm 2 as follows.

Theorem 6 (Byzantine Resilience of Algorithm 2): At each iteration t, Algorithm 2 can at least tolerate $k_i < M(1-2\bar{p}_i^{(t)})/(2\bar{q}_i^{(t)} - 1)$ Byzantine attackers (i.e., the convergence of Algorithm 2 is guaranteed) with $(1/B)\sum_{j\in\mathcal{B}} q_{j,i}^{(t)} = \bar{q}_i^{(t)}$ on the ith coordinate of the gradient. Overall, the number of Byzantine workers that Algorithm 2 can tolerate is given by $\min_{1\leq i\leq d} k_i$.

Remark 11: It can be noticed that when $\bar{q}_i^{(t)} = 1$, the number of attackers that Algorithm 2 can tolerate is the same as that specified by the first condition in Corollary 2, and the second condition is not needed.

VII. Experiments

In the experiments, we first validate our theoretical results in the minimization of the well-known Rosenbrock function. Then, we examine the performance of the proposed method on the MNIST dataset, the CIFAR-10 dataset, and the Tiny-ImageNet dataset.[3] Finally, we present the performance of the proposed method in the partial worker participation scenario as in FL, in which only a portion of workers are sampled for training during each communication round.

A. Minimization of the Rosenbrock Function

We consider the minimization of the well-known Rosenbrock function with ten variables [7]

$$F(x) = \sum_{i=1}^{d} F_i(x), \text{ where } F_i(x) = 100(x_{i+1}-x_i^2)+(1-x_i)^2. \tag{17}$$

[3]Note that we focus on the image classification tasks in the experiments since data heterogeneity can be readily captured, for example, by the number of classes of data that each user stores. For tasks like natural language processing, measuring and controlling the data heterogeneity is difficult, if not impossible.

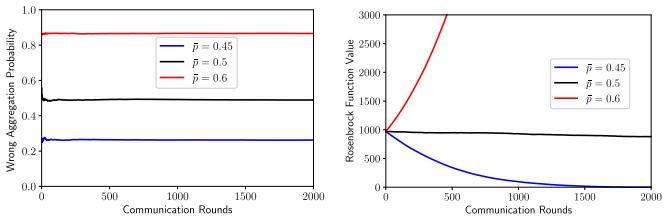


Fig. 2. Probability of wrong aggregation (left). Rosenbrock function value that is minimized (right).

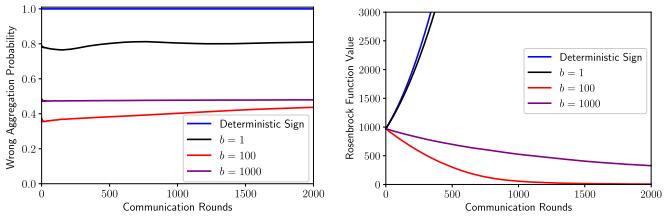


Fig. 3. Comparison between deterministic sign and sto-sign.

We first examine the impact of the average probability of wrong signs $\bar{p}$ to validate our analysis in Theorem 1. More specifically, we randomly generate a vector $\boldsymbol{v}$ of length M with $\boldsymbol{v}_i \in (0, 1), \forall 1 \leq i \leq M$. Then, $\boldsymbol{v}$ is normalized such that $(1/M)\sum_{i=1}^{M} \boldsymbol{v}_i = \bar{p}$. During each communication round, we first compute the global gradient $\nabla F(x)$, after which the signs of the workers are generated following the probability of wrong signs $\boldsymbol{v}$. Finally, the aggregated results are obtained by majority vote. Fig. 2 shows the probability of wrong aggregation and the objective function value for $M = 30$. It can be observed that when the average probability of wrong signs $\bar{p} < (1/2)$, the overall probability of wrong aggregation is smaller than $(1/2)$ when $M = 30$, and the sign-based algorithm converges. When the average probability of wrong signs $\bar{p} > (1/2)$, it fails to converge.

Then, we examine the effectiveness of sto-sign. To simulate the heterogeneity across the workers, we first generate a vector $\boldsymbol{v}$ of length M, with $\boldsymbol{v}_i \in (0, 1), \forall 1 \leq i \leq M$. The first $0.7M$ entries are scaled by a negative factor such that the average of all the entries is 1. The local objective function of worker m is given by $\boldsymbol{v}_m F(x)$. Fig. 3 shows the performance of deterministic sign and sto-sign with different selection of $\boldsymbol{b} = b \cdot \mathbf{1}$. Since 70% of the workers always have the wrong signs, the aggregation results of deterministic sign (as in SIGNSGD) are wrong with probability 1. As a result, SIGNSGD fails to converge in this case. For sto-sign, it can be observed that as b increases, the probability of wrong aggregation first decreases, and then increases. Moreover, thanks to the stochasticity introduced in sto-sign, the probability of wrong aggregation with a suitable b (e.g., $b = 100$) is smaller than $(1/2)$, and the corresponding algorithm (i.e., Sto-SIGNSGD) converges.

Finally, we examine the impact of the number of workers M. More specifically, in addition to the normal workers, there are $0.9 \times M$ Byzantine attackers with $\bar{q} = 1 - \bar{p}$. The other settings are the same as Fig. 2. It can be observed from Fig. 4 that both the probability of wrong aggregation and the objective decrease as M increases, which corroborates our discussion in Remark 6.

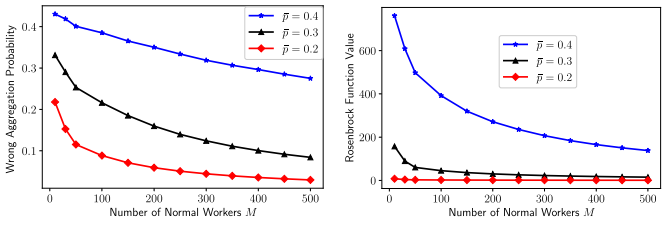


Fig. 4. Impact of the number of workers M.

B. Results on MNIST, CIFAR-10, and Tiny-ImageNet

In this section, we implement our proposed method with a two-layer fully connected neural network on the standard MNIST dataset, VGG-9 [50] on the CIFAR-10 dataset, and compact convolutional transformer (CCT) [51] on the Tiny-ImageNet dataset. For MNIST and CIFAR-10, we consider a scenario of $M = 31$ normal workers. To simulate the heterogeneous data distribution scenario, each worker only stores exclusive data for one out of the ten categories, unless otherwise noted. For Tiny-ImageNet, we consider a scenario of $M = 10$ normal workers and follow [52] to simulate heterogeneous data distribution, in which the training data on each worker are drawn independently with class labels following a Dirichlet distribution $\text{Dir}(\alpha)$ with $\alpha = 0.1$. Besides, for MNIST, the workers evaluate their gradients over the whole local datasets, while for CIFAR-10 and Tiny-ImageNet, the workers train their local models with mini-batch sizes of 32 and 256, respectively. More details can be found in Appendix C in the supplementary material.

We validate our analysis and compare the proposed method with SIGNSGD [4] and the commonly adopted FL baseline FedAvg [47]. In the presence of data heterogeneity, the proposed method converges faster than FedAvg with respect to communication cost and SIGNSGD fails to converge. Moreover, the proposed method demonstrates better Byzantine resilience than SIGNSGD, while FedAvg is not Byzantine resilient in design.

1) Selection of $\boldsymbol{b}$: In the fixed $\boldsymbol{b}$ scenarios, we set $\boldsymbol{b} = b \cdot \mathbf{1}$ for some positive constant b. In this case, $b_i < \max_m |(\boldsymbol{g}_m^{(t)})_i|$ may not hold for some i, and the probabilities defined in (6) fall out of the range $[0, 1]$. We round them to 1 if they are positive and 0 otherwise. For "Optimal $\boldsymbol{b}$," we set $b_i = \max_m |(\boldsymbol{g}_m^{(t)})_i|, \forall i$. It can be observed from Fig. 5 that for fixed $\boldsymbol{b}$, b should be large enough to optimize the performance. As b keeps increasing, both the training and the testing accuracy decrease, which corroborates our analysis. Furthermore, with the same communication overhead, Sto-SIGNSGD with a fixed $\boldsymbol{b}$ achieves a higher testing accuracy than FedAvg (especially when the allowed communication overhead is small) and SIGNSGD, which demonstrates its effectiveness. Moreover, the testing accuracy of Sto-SIGNSGD with a fixed $\boldsymbol{b}$ approaches that with optimal $\boldsymbol{b}$.

Considering that the gradients may change slowly during the training process, we propose to update $\boldsymbol{b}$ periodically

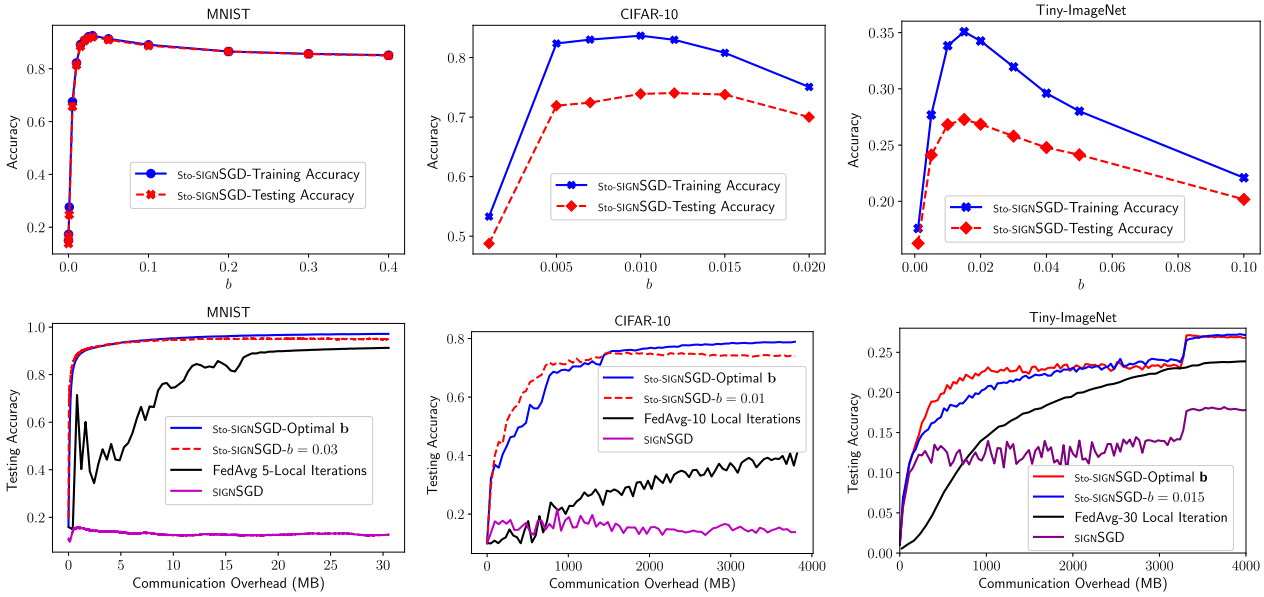


Fig. 5. Training and testing accuracy of Sto-SIGNSGD for different $\boldsymbol{b} = b \cdot \mathbf{1}$. We run 200, 8000, and 6000 communication rounds for MNIST, CIFAR-10, and Tiny-ImageNet, respectively (top). The testing accuracy of Sto-SIGNSGD, SIGNSGD, and FedAvg [47] with respect to the total communication overhead. We tune the number of local iterations for FedAvg from the set {1, 5, 10, 20, 30, 40} (bottom). Both for the top and bottom figures.

and examine two heuristic schemes for the selection of $\boldsymbol{b}$ as follows.

1) In scheme I, after every K communication rounds, each worker m shares its gradient $\boldsymbol{g}_m^{(t)}$ in full precision. Then, the server finds the "optimal $\boldsymbol{b}$" as in Remark 2 (i.e., $b_i = \max_m |(\boldsymbol{g}_m^{(t)})_i|$) and sends it back to the workers, which remains fixed in the following K communication rounds. In this case, for K communication rounds, the communication overhead of the proposed scheme is $(K+31)d$. The corresponding communication overhead for full precision distribution SGD and SIGNSGD are $32Kd$ and Kd, respectively.[4]
2) In scheme II, after every K communication rounds, each worker m computes the layer-wise median of the absolute values of the gradients $\{\text{median}(|\boldsymbol{g}_{m,1}^{(t)}|), \text{median}(|\boldsymbol{g}_{m,2}^{(t)}|), \ldots, \text{median}(|\boldsymbol{g}_{m,J}^{(t)}|)\}$ in which $\boldsymbol{g}_{m,j}^{(t)}$ is the jth block (which corresponds to the jth layer of the neural network) of worker m's gradient at communication round t. Then, each worker m further takes the median and shares $\tilde{b}_m = \text{median}(\{\text{median}(|\boldsymbol{g}_{m,1}^{(t)}|), \text{median}(|\boldsymbol{g}_{m,2}^{(t)}|), \ldots, \text{median}(|\boldsymbol{g}_{m,J}^{(t)}|)\})$ with the server. The server sets $\boldsymbol{b} = \text{median}(\tilde{b}_m) \cdot \mathbf{1}$ and sends it back to the workers, which will be used in the following K communication rounds. In this case, for K communication rounds, the communication overhead is $Kd + 32$.

We examine the performance of schemes I and II by running Sto-SIGNSGD for 200 and 8000 communication rounds for MNIST and CIFAR-10, respectively. The corresponding results are presented in Tables I and II. In particular, the scheme I with $K = 1$ essentially corresponds to "Optimal $\boldsymbol{b}$" discussed in Remark 2. It can be observed from Tables I and II

[4]Note that sharing the gradients in full precision may incur data privacy issues [53]. The implementation of the proposed method with privacy guarantees is left as future work.

TABLE I
TESTING ACCURACY OF STO-SIGNSGD ON MNIST

K	1	100	200
SCHEME I	93.07±0.29%	92.39±0.17%	90.99±0.35%
SCHEME II	90.14±0.27%	90.23±0.21%	90.16±0.20%

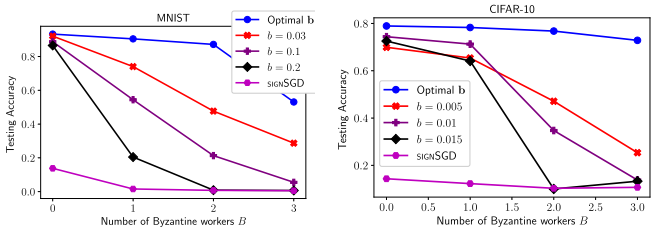


Fig. 6. Sto-SIGNSGD for different numbers of Byzantine workers and different $\boldsymbol{b}$'s.

that scheme I with $K = 100$ achieves comparable performance to "optimal $\boldsymbol{b}$," which demonstrates the effectiveness of the periodical updating scheme. Moreover, with a negligible communication cost of 32 bits for each round, scheme II with $K = 1$ also achieves satisfactory performance.

2) Byzantine Resilience: Fig. 6 shows the performance of Sto-SIGNSGD for different selection of $\boldsymbol{b} = b \cdot \mathbf{1}$ and different number of Byzantine workers B. For MNIST, the Byzantine attackers evaluate their gradients over the whole training dataset; for CIFAR-10, the mini-batch sizes of the Byzantine attackers are set to 32. All the attackers flip the signs of gradients and send them to the server (termed "flip sign" attack). As the number of Byzantine workers increases, both the training and the testing accuracy of Sto-SIGNSGD with a larger b drop much faster than that with a smaller b, which conforms to our analysis above that a larger b results in

TABLE II
TESTING ACCURACY OF STO-SIGNSGD ON CIFAR-10

K	1	100	200	300	400
SCHEME I	78.91±0.25%	77.90±0.19%	75.91±0.34%	74.92±0.26%	74.11±0.38%
SCHEME II	73.26±0.20%	70.87±0.33%	70.25±0.23%	69.22±0.53%	69.25±0.60%

TABLE III
TESTING ACCURACY OF STO-SIGNSGD ON MNIST UNDER ATTACKS

NUMBER OF ATTACKERS	1	2	3
FIXED $\boldsymbol{b} = 0.03 \cdot \mathbf{1}$	74.04±0.29%	47.68±1.57%	28.69±1.98%
SCHEME II, $K = 100$	74.64±0.23%	47.61±1.39%	18.37±2.33%

TABLE IV
TESTING ACCURACY OF STO-SIGNSGD ON CIFAR-10 UNDER ATTACKS

NUMBER OF ATTACKERS	1	2	3
FIXED $\boldsymbol{b} = 0.01 \cdot \mathbf{1}$	71.27±0.34%	34.73±2.82%	13.80±1.54%
SCHEME II, $K = 100$	70.24±0.38%	58.32±1.60%	25.74±3.70%

TABLE V
TESTING ACCURACY OF STO-SIGNSGD ON MNIST

B	SIGNSGD 2 LABELS	OPTIMAL $\boldsymbol{b}$ 2 LABELS	SIGNSGD 4 LABELS	OPTIMAL $\boldsymbol{b}$ 4 LABELS
0	70.03±0.71%	92.34±0.40%	90.53±0.16%	93.12±0.10%
1	66.31±0.62%	93.14±0.29%	88.21±0.40%	93.38±0.18%
2	60.19±1.35%	92.71±0.14%	87.34±0.57%	93.39±0.13%
3	56.23±1.57%	91.13±0.51%	82.49±1.30%	92.19±0.27%
4	47.44±0.85%	84.49±1.11%	81.51±1.39%	92.31±0.20%

TABLE VI
TESTING ACCURACY OF STO-SIGNSGD ON CIFAR-10

B	FLIP SIGN	LIE	GAUSSIAN COLLUDE	GAUSSIAN
1	71.27±0.24%	71.91±0.28%	74.08±0.22%	74.09±0.24%
3	13.80±2.43%	38.80±1.04%	71.82±0.28%	74.11±0.27%
5	-	-	68.74±0.33%	71.88±0.40%
10	-	-	34.32±1.77%	71.51±0.34%

worse Byzantine resilience. It is also observed that SIGNSGD essentially fails in this extremely heterogeneous data distribution setting (where each worker holds exclusive data), even without attackers.

We further examine the performance of scheme II in the presence of attacks (which sends large $\tilde{b}_m$'s), and the results are presented in Tables III and IV. In particular, since the server sets $\boldsymbol{b} = \text{median}(\tilde{b}_m) \cdot \mathbf{1}$, the impact of attackers is mitigated. It can be observed that, in this case, scheme II with $K = 100$ essentially achieves comparable performance to $\boldsymbol{b} = 0.03 \cdot \mathbf{1}$ and $\boldsymbol{b} = 0.01 \cdot \mathbf{1}$ for MNIST and CIFAR-10, respectively.

To examine the impact of data heterogeneity, we vary the number of labels of each worker's local training dataset in Table V. It can be observed that the testing accuracy of SIGNSGD improves when the training data becomes more homogeneously distributed across workers. Both SIGNSGD and STO-SIGNSGD obtain better Byzantine resilience as the number of labels increases. Finally, STO-SIGNSGD with optimal $\boldsymbol{b}$ still outperforms SIGNSGD, which indicates that introducing the stochasticity is still beneficial in the more homogeneous data distribution scenarios.

We also examine the performance of STO-SIGNSGD with $\boldsymbol{b} = 0.01 \cdot \mathbf{1}$ against other commonly adopted Byzantine attacks, including the Gaussian attack [27] and the little is enough (LIE) attack [54] in Table VI. More specifically, the attackers send the signs of the attacked gradients to the server. In "Gaussian Collude," we assume that all the attackers share the same signs (generated by one Gaussian attack), while in "Gaussian," each attacker generates its attacked gradient independently. It can be observed that compared to the "flip sign" attack in Fig. 6, STO-SIGNSGD performs better under "LIE," "Gaussian Collude," and "Gaussian" attacks.

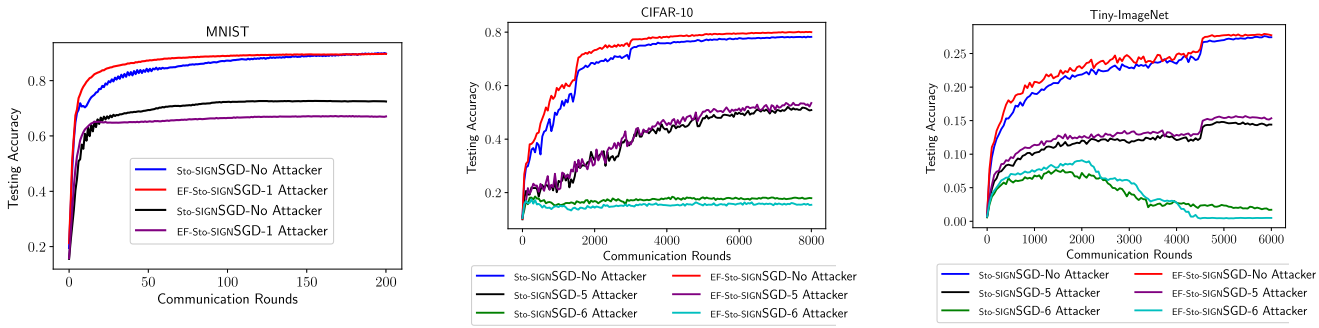


Fig. 7. MNIST with $\boldsymbol{b} = 0.02 \cdot \mathbf{1}$ (left). CIFAR-10 with optimal $\boldsymbol{b}$ (middle). Tiny-ImageNet with $\boldsymbol{b} = 0.015 \cdot \mathbf{1}$ (right).

TABLE VII
TESTING ACCURACY OF THE ALGORITHMS ON CIFAR-10

ALGORITHMS	D-SGD	SCALED SIGNSGD	1-BIT L_2 NORM QSGD	1-BIT L_∞ NORM QSGD	STO-SIGNSGD	EF-STO-SIGNSGD
ACCURACY	77.67±0.83%	22.21±2.67%	47.08±0.48%	74.48±0.53%	78.52±0.47%	78.73±0.68%

3) Error-Feedback: Fig. 7 shows the performance of EF-STO-SIGNSGD against the "flip sign" attack. It can be observed from Fig. 7 that the error-feedback variant does not necessarily perform better under attack. More specifically, EF-STO-SIGNSGD performs worse than STO-SIGNSGD on MNIST, CIFAR-10, and Tiny-ImageNet in the presence of 1, 6, and 6 Byzantine workers, respectively. This may be because as the gradients decrease, the probability of wrong aggregation increases. In this case, the error-feedback mechanism may carry the wrong aggregations to future iterations and harm the learning process. However, EF-STO-SIGNSGD outperforms STO-SIGNSGD on all the examined datasets without Byzantine workers, which validates its effectiveness.

C. Extension to Partial Worker Participation

In this section, we examine the performance of the compressor sto-sign in the partial worker participation scenario. More specifically, we consider a scenario of $M = 100$ normal workers with the training data on each worker drawn independently with class labels following a Dirichlet distribution Dir(α) with $\alpha = 0.1$. During each communication round, 31 workers out of 100 are randomly selected to perform the training. We examine four baselines: "D-SGD," "Scaled SIGNSGD" [24], "1-bit L_2 norm QSGD" [2] and "1-bit L_∞ norm QSGD" [2]. More specifically, in "D-SGD," the workers share the gradients with the parameter server in full precision; in "Scaled SIGNSGD," each worker m shares $((||\boldsymbol{g}_m^{(t)}||_1)/d)\text{sign}(\boldsymbol{g}_m^{(t)})$, in which $\boldsymbol{g}_m^{(t)}$ is the local stochastic gradient; in "1-bit L_2 norm QSGD," each entry of the gradient $\boldsymbol{g}_m^{(t)}$ is mapped to $\text{sign}((\boldsymbol{g}_m^{(t)})_i)$ with probability $|(\boldsymbol{g}_m^{(t)})_i|/||\boldsymbol{g}_m^{(t)}||_2$, and 0 otherwise; in "1-bit L_∞ norm QSGD," each entry of the gradient $\boldsymbol{g}_m^{(t)}$ is mapped to $\text{sign}((\boldsymbol{g}_m^{(t)})_i)$ with probability $|(\boldsymbol{g}_m^{(t)})_i|/||\boldsymbol{g}_m^{(t)}||_\infty$, and 0 otherwise. We note that the stochastic quantizer in L_2 norm QSGD is one of the most commonly adopted compressors in the literature (e.g., [6], [16]). In all the baselines, the parameter server performs the aggregation by taking the average over all the compressed gradients from the workers.

We run the algorithms for 3500 communication rounds, and the corresponding results are presented in Table VII. It can be observed that the proposed STO-SIGNSGD and EF-STO-SIGNSGD achieve comparable performance to "D-SGD" which shares the gradients in full precision while improving the communication efficiency. In addition, the proposed algorithms attain higher accuracy than the other baselines concerning the communication rounds.

VIII. CONCLUSION

In this work, we derive a sufficient condition for the convergence of sign-based gradient descent methods, based on which a novel gradient compressor that can deal with heterogeneous data distributions is proposed. The proposed algorithms are proven to converge in the heterogeneous data distribution scenario. Then, the Byzantine resilience of the proposed algorithm is shown analytically. Besides, we further improve the learning performance of the proposed method by incorporating the error-feedback scheme. The parameter $\boldsymbol{b}$ plays a crucial role in the performance of the proposed compressor, and we develop heuristic approaches to select $\boldsymbol{b}$ in this work. Designing adaptive schemes for the selection of $\boldsymbol{b}$ and further incorporating privacy-preserving techniques remain interesting future directions.

ACKNOWLEDGMENT

The views expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] J. Dean et al., "Large scale distributed deep networks," in *Proc. NeurIPS*, 2012, pp. 1223–1231.

[2] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Proc. NeurIPS*, 2017, pp. 1709–1720.

[3] W. Wen et al., "TernGrad: Ternary gradients to reduce communication in distributed deep learning," in *Proc. NeurIPS*, 2017, pp. 1509–1519.

[4] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signSGD: Compressed optimisation for non-convex problems," in *Proc. ICML*, 2018, pp. 560–569.

[5] J. Wu, W. Huang, J. Huang, and T. Zhang, "Error compensated quantized SGD and its applications to large-scale distributed optimization," in *Proc. ICML*, 2018, pp. 5325–5333.

[6] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization," in *Proc. Int. Conf. Artif. Intell. Stat. (AISTATS)*, Aug. 2020, pp. 2021–2031.

[7] M. Safaryan and P. Richtárik, "Stochastic sign descent methods: New algorithms and better theory," in *Proc. ICML*, 2021, pp. 9224–9234.

[8] X. Chen, T. Chen, H. Sun, S. Z. Wu, and M. Hong, "Distributed training with heterogeneous data: Bridging median- and mean-based algorithms," in *Proc. NeurIPS*, vol. 33, 2020, pp. 21616–21626.

[9] K. Yue, R. Jin, C.-W. Wong, and H. Dai, "Federated learning via plurality vote," *IEEE Trans. Neural Netw. Learn. Syst.*, 2022.

[10] S. M. Shah and V. K. N. Lau, "Model compression for communication efficient federated learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 5937–5951, Sep. 2021.

[11] J. Xu, W. Du, Y. Jin, W. He, and R. Cheng, "Ternary compression for communication-efficient federated learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 3, pp. 1162–1176, Mar. 2022.

[12] J. Bernstein, J. Zhao, K. Azizzadenesheli, and A. Anandkumar, "signSGD with majority vote is communication efficient and Byzantine fault tolerant," in *Proc. ICLR*, 2019.

[13] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, "Communication compression for decentralized training," in *Proc. NeurIPS*, 2018, pp. 7652–7662.

[14] P. Jiang and G. Agrawal, "A linear speedup analysis of distributed deep learning with sparse and quantized communication," in *Proc. NeurIPS*, 2018, pp. 2525–2536.

[15] H. Wang, S. Sievert, S. Liu, Z. Charles, D. Papailiopoulos, and S. Wright, "ATOMO: Communication-efficient learning via atomic sparsification," in *Proc. NeurIPS*, 2018, pp. 9850–9861.

[16] F. Haddadpour, M. M. Kamani, A. Mokhtari, and M. Mahdavi, "Federated learning with compression: Unified analysis and sharp guarantees," in *Proc. Int. Conf. AISTATS*, 2021, pp. 2350–2358.

[17] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, Sep. 2020.

[18] D. Basu, D. Data, C. Karakus, and S. Diggavi, "Qsparse-local-SGD: Distributed SGD with quantization, sparsification and local computations," in *Proc. NeurIPS*, vol. 32, 2019, pp. 14695–14706.

[19] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, Sep. 2014.

[20] D. Carlson, Y.-P. Hsieh, E. Collins, L. Carin, and V. Cevher, "Stochastic spectral descent for discrete graphical models," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 2, pp. 296–311, Mar. 2016.

[21] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, no. 1, pp. 1–210, 2021.

[22] S. U. Stich, J. B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proc. NeurIPS*, 2018, pp. 4447–4458.

[23] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli, "The convergence of sparsified gradient methods," in *Proc. NeurIPS*, 2018, pp. 5973–5983.

[24] S. P. Karimireddy, Q. Rebjock, S. Stich, and M. Jaggi, "Error feedback fixes signSGD and other gradient compression schemes," in *Proc. ICML*, 2019, pp. 3252–3261.

[25] S. Zheng, Z. Huang, and J. Kwok, "Communication-efficient distributed blockwise momentum SGD with error-feedback," in *Proc. NeurIPS*, 2019, pp. 11446–11456.

[26] H. Tang, X. Lian, T. Zhang, and J. Liu, "DoubleSqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression," in *Proc. ICML*, 2019, pp. 6155–6165.

[27] P. Blanchard, E. M. E. Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. NeurIPS*, 2017, pp. 119–129.

[28] D. Alistarh, Z. Allen-Zhu, and J. Li, "Byzantine stochastic gradient descent," in *Proc. NeurIPS*, 2018, pp. 4613–4623.

[29] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *Proc. ICML*, 2018, pp. 5650–5659.

[30] L. Chen, H. Wang, Z. Charles, and D. Papailiopoulos, "DRACO: Byzantine-resilient distributed training via redundant gradients," in *Proc. ICML*, 2018, pp. 903–912.

[31] S. Rajput, H. Wang, Z. Charles, and D. Papailiopoulos, "DETOX: A redundancy-based framework for faster and more robust gradient aggregation," in *Proc. NeurIPS*, vol. 32, 2019.

[32] L. Li, W. Xu, T. Chen, G. B. Giannakis, and Q. Ling, "RSA: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1544–1551.

[33] C. Xie, S. Koyejo, and I. Gupta, "SLSGD: Secure and efficient distributed on-device machine learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2019, pp. 213–228.

[34] D. Data and S. Diggavi, "Byzantine-resilient high-dimensional SGD with local iterations on heterogeneous data," in *Proc. ICML*, 2021, pp. 2478–2488.

[35] J. Steinhardt, M. Charikar, and G. Valiant, "Resilience: A criterion for learning in the presence of arbitrary outliers," in *Proc. Innov. Theor. Comput. Sci. Conf.*, vol. 94, 2018, pp. 45:1–45:21.

[36] S. P. Karimireddy, L. He, and M. Jaggi, "Byzantine-robust learning on heterogeneous datasets via bucketing," in *Proc. ICLR*, 2022.

[37] S. P. Karimireddy, L. He, and M. Jaggi, "Learning from history for Byzantine robust optimization," in *Proc. ICML*, 2021, pp. 5311–5319.

[38] S. Liu, N. Gupta, and N. H. Vaidya, "Redundancy in cost functions for Byzantine fault-tolerant federated learning," in *Proc. 1st Workshop Syst. Challenges Reliable Secure Federated Learn.*, Oct. 2021, pp. 4–6.

[39] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *Proc. ICML*, 2021, pp. 6357–6368.

[40] J. Xu, S.-L. Huang, L. Song, and T. Lan, "Byzantine-robust federated learning through collaborative malicious gradient filtering," in *Proc. IEEE 42nd Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2022, pp. 1223–1235.

[41] W. Wan, S. Hu, J. Lu, L. Y. Zhang, H. Jin, and Y. He, "Shielding federated learning: Robust aggregation with adaptive client selection," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 753–760.

[42] A. Ghosh, R. K. Maity, S. Kadhe, A. Mazumdar, and K. Ramchandran, "Communication-efficient and Byzantine-robust distributed learning with error feedback," *IEEE J. Sel. Areas Inf. Theory*, vol. 2, no. 3, pp. 942–953, Sep. 2021.

[43] H. Zhu and Q. Ling, "Byzantine-robust distributed learning with compression," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 9, pp. 280–294, 2023.

[44] Q. Xia, Z. Tao, Q. Li, and S. Chen, "Byzantine tolerant algorithms for federated learning," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 6, pp. 3172–3183, Nov./Dec. 2023.

[45] W. Bao and J. He, "BOBA: Byzantine-robust federated learning with label skewness," 2022, *arXiv:2208.12932*.

[46] X. Cao, M. Fang, J. Liu, and N. Z. Gong, "FLTrust: Byzantine-robust federated learning via trust bootstrapping," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2021.

[47] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. AISTATS*, 2017, pp. 1273–1282.

[48] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, Sep. 1951.

[49] K. Bonawitz et al., "Towards federated learning at scale: System design," in *Proc. Mach. Learn. Syst.*, vol. 1, 2019, pp. 374–388.

[50] C. Lee, S. S. Sarwar, P. Panda, G. Srinivasan, and K. Roy, "Enabling spike-based backpropagation for training deep neural network architectures," *Frontiers Neurosci.*, vol. 14, Feb. 2020.

[51] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, "Escaping the big data paradigm with compact transformers," 2021, *arXiv:2104.05704*.

[52] T.-M. Harry Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," 2019, *arXiv:1909.06335*.

[53] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. NeurIPS*, vol. 32, 2019, pp. 14774–14784.

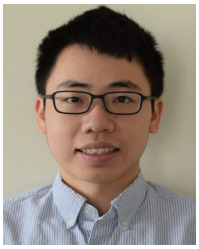
[54] G. Baruch, M. Baruch, and Y. Goldberg, "A little is enough: Circumventing defenses for distributed learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 8635–8645.

Richeng Jin (Member, IEEE) received the B.S. degree in information and communication engineering from Zhejiang University, Hangzhou, China, in 2015, and the Ph.D. degree in electrical engineering from North Carolina State University, Raleigh, NC, USA, in 2020.

He was a Post-Doctoral Researcher in electrical and computer engineering at North Carolina State University, from 2021 to 2022. He is currently a Faculty Member of the Department of Information and Communication Engineering, Zhejiang University, Hangzhou, China. His research interests include general area of wireless AI, game theory, and security and privacy in machine learning/artificial intelligence, and wireless networks.

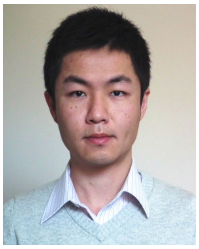
Yuding Liu (Student Member, IEEE) received the B.S. degree in microelectronics and communication engineering from Chongqing University, Chongqing, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Information and Communication Engineering, Zhejiang University, Hangzhou, China.

His current research interests include federated learning and wireless communication.



Yufan Huang received the B.E. degree in electromagnetic engineering from Beihang University, Beijing, China, in 2012, the M.S. and Ph.D. degrees in electrical and computer engineering from North Carolina State University, Raleigh, NC, USA, in 2014 and 2019, respectively.

His research interests are in the areas of mobile networks, social networks, and multilayer networks. His current research interests include information spreading in multiplex networks.



Xiaofan He (Senior Member, IEEE) received the B.S. degree in electronics and information engineering from Huazhong University of Science and Technology, Wuhan, China, in 2008, the M.S. degree in electrical and computer engineering from McMaster University, Hamilton, ON, Canada, in 2011, and the Ph.D. degree in electrical and computer engineering from North Carolina State University, Raleigh, NC, USA, in 2015.

He was a tenure-track Assistant Professor of electrical engineering with Lamar University, Beaumont, TX, USA, from 2016 to 2018. He moved back to his hometown Wuhan, in 2019. He is currently a Faculty Member of the School of Electronic Information, Wuhan University, Wuhan. His research interests are in the general areas of wireless communications, networking, and computing. His current research mainly focuses on (distributed) edge computing and the associated optimization, scheduling, learning, and statistical analysis.

Dr. He received the Exemplary Reviewer Award of IEEE TRANSACTIONS ON COMMUNICATIONS in 2014 and 2015, and a Distinguished Member of Technical Program Committee Award of IEEE INFOCOM in 2018. He is currently serving as an Associate Editor for IEEE ACCESS.

Tianfu Wu (Member, IEEE) received the Ph.D. degree in statistics from UCLA, Los Angeles, CA, USA, in 2011.

He is an Associate Professor with the Department of Electrical and Computer Engineering, NC State University (NCSU), Raleigh, NC, USA, and Leads the laboratory for interpretable visual modeling, computing and learning (iVMCL). His research focuses on computer vision, often motivated by the tasks of pursuing a unified framework for AI to ask, learn, test, explain and refine (ALTER) in a trustworthy, robust and responsive way for AI generated content and ground-truth (AIGCGT).

Huaiyu Dai (Fellow, IEEE) received the B.E. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1996 and 1998, respectively, and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, USA, in 2002.

He was with Bell Labs, Lucent Technologies, Holmdel, NJ, USA, in Summer 2000, and with AT&T Labs-Research, Middletown, NJ, USA, in Summer 2001. He is currently a Professor with Electrical and Computer Engineering, NC State University, Raleigh, NC, USA, holding the title of University Faculty Scholar. His research interests are in the general areas of communication systems and networks, advanced signal processing for digital communications, communication theory, and information theory. His current research interests include on networked information processing and cross layer design in wireless networks, cognitive radio networks, network security, and associated information-theoretic and computation theoretic analysis.

Dr. Dai was the Co-Chair of the Signal Processing for Communications Symposium of IEEE Globecom 2013, the Communications Theory Symposium of IEEE ICC 2014, and the Wireless Communications Symposium of IEEE Globecom 2014. He was a co-recipient of Best Paper Awards at 2010 IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS 2010), the 2016 IEEE INFOCOM BIGSECURITY Workshop, and 2017 IEEE International Conference on Communications (ICC 2017). He has served as an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. Currently, he is an Area Editor in Charge of Wireless Communications for IEEE TRANSACTIONS ON COMMUNICATIONS, and a member of the Executive Editorial Committee for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.