Adaptive Top-K in SGD for Communication-Efficient Distributed Learning in Multi-Robot Collaboration

Journal:	*Journal of Selected Topics in Signal Processing*
Manuscript ID	J-STSP-IRSSPI-00199-2023
Manuscript Type:	Special Issue Paper
Date Submitted by the Author:	15-Sep-2023
Complete List of Authors:	Ruan, Mengzhe; City University of Hong Kong, YAN, Guangfeng; City University of Hong Kong Xiao, Yuanzhang; University of Hawai'i at Mānoa, Song, Linqi; City University of Hong Kong, Department of Computer Science xu, weitao; City University of Hong Kong

Adaptive Top-K in SGD for Communication-Efficient Distributed Learning in Multi-Robot Collaboration

Mengzhe Ruan[1,2] Guangfeng Yan[1,2] Yuanzhang Xiao[3] Linqi Song[1,2] Weitao Xu[1,2]

[1] City University of Hong Kong Shenzhen Research Institute

[2] Department of Computer Science, City University of Hong Kong

[3]Hawaii Advanced Wireless Technologies Institute, University of Hawaii at Manoa

***Abstract*—Distributed stochastic gradient descent (D-SGD) with gradient compression has become a popular communication-efficient solution for accelerating optimization procedures in distributed learning systems like multi-robot systems. One commonly used method for gradient compression is Top-K sparsification, which sparsifies the gradients by a fixed degree during model training. However, there has been a lack of an adaptive approach with a systematic treatment and analysis to adjust the sparsification degree to maximize the potential of the model's performance or training speed. This paper proposes a novel adaptive Top-K in Stochastic Gradient Descent framework that enables an adaptive degree of sparsification for each gradient descent step to optimize the convergence performance by balancing the trade-off between communication cost and convergence error with respect to the norm of gradients and the communication budget. Firstly, an upper bound of convergence error is derived for the adaptive sparsification scheme and the loss function. Secondly, we consider communication budget constraints and propose an optimization formulation for minimizing the deep model's convergence error under such constraints. We obtain an enhanced compression algorithm that significantly improves model accuracy under given communication budget constraints. Finally, we conduct numerical experiments on general image classification tasks using the MNIST, CIFAR-10 and CIFAR-100 datasets. For the multi-robot collaboration tasks, we choose the object detection task on the PASCAL VOC dataset. The results demonstrate that the proposed adaptive Top-K algorithm in SGD achieves a significantly better convergence rate compared to state-of-the-art methods, even after considering error compensation.**

***Index Terms*—Distributed Learning, Communication-efficient, Gradient Sparsification, Error Compensation, Multi-Robot Collaboration.**

I. Introduction

As the field of robotics grows, more and more robots appear in human daily life nowadays. The multi-robot systems are well-suited for training learning modes together using the data collected by the robots. There is a rapid emergence of distributed multi-robot collaborative learning algorithms in which local training parameters aggregation accomplishes global learning models [2] [3] [4]. Stochastic gradient descent (SGD) is commonly employed in machine learning for its efficient computational complexity and strong empirical results. Nevertheless, with the overwhelming amount of data today, the vanilla SGD framework has become inadequate. To address this, distributed SGD [5] [6] relied on local user data to construct distributed models and transmit local gradients between distributed nodes and a parameter server until all nodes converge to a global consensus on the learning model have become the core of most distributed learning algorithms.

However, the communication overhead of transmitting gradients often becomes the performance bottleneck due to the limited bandwidth in distributed SGD [7] [8]. Gradient compression, which uses less information to represent the gradients, is an effective and efficient method to solve this problem. The compression methods, however, inevitably introduce compression noise which affects the convergence of the model. Therefore, how to choose the compression methods and the compression level efficiently to balance the trade-off between communication cost and convergence performance remains an open challenge. To address this issue, three primary communication reduction schemes have been proposed to boost the efficiency of distributed SGD. Quantization [29] and sparsification [9] both work by reducing communication overhead through minimizing the uploaded model size. Quantization encodes original gradient vectors into smaller bits, while sparsification discards less informative components. Another approach is to decrease the number of communication rounds between distributed nodes and the server using periodic or less frequent model updates [40] [41] [42]. Some researches use the combination of these three techniques [24] [23]. This work will mainly focus on the sparsification techniques.

The existing literature in this field has yet to thoroughly explore two important aspects. Firstly, the adaptively adjustment of compression levels especially for biased compressors has not been extensively investigated. Most current approaches employ a fixed compression level, such as fixed quantization bits or sparse size, throughout the entire training process, disregarding the fact that the information statistics of model gradients change during training. Conversely, some studies [32] [21] [20] have empirically demonstrated that dynamically adjusting the compression level can lead to improved con-

This work was presented in part at 2023 IEEE Global Communications Conference: Communication Theory (Globecom 2023) [1] vergence compared to fixed schemes but mainly for unbiased compressors like gradient quantization. Nevertheless, unbiased methods need more computing resources than biased methods. Our work, however, provides a theoretical perspective by offering a biased compression level adjustment rule for the training process. Secondly, there is a lack of a systematic framework to characterize the trade-off between explicit communication budget and learning performance under error compensation [30]. Previous research has mainly relied on experimental results, indicating that smaller communication costs (larger compression levels) result in lower model accuracy. An online learning method was proposed in [35] to adaptively adjust the degree of gradient sparsity when the total dataset is non-i.i.d in the federated learning network but lacks a theoretical convergence analysis. In contrast, we consider explicit communication budget constraints, which limit the total number of bits available for transferring gradients during the entire training process and provide the theoretical proof. More importantly, we aim to characterize the trade-off relationship between this budget and the convergence rate with or without the gradient error compensation. By addressing these two aspects, our work contributes to a more comprehensive understanding of adaptive sparsification level and provides insights into the trade-off between communication budget and learning performance for reality training.

In this paper, we propose a novel adaptive Top-K SGD framework for efficient distributed learning in robotics networks, named AdapTop-K, that aims to improve the convergence performance of Top-K while maintaining the same communication cost in distributed learning with or without error compensation. Under the assumption of smoothness, strongly convex and Polyak-Lojasiewicz condition, we derive an upper bound on the gap between the loss function and the optimal loss to characterize the convergence error caused by limited iteration steps, sampling, and adaptive Top-K sparsification. Based on the theoretical analysis, we design an adaptive Top-K method by minimizing the convergence upper bound under the desired total communication cost. The proposed AdapTop-K algorithm adjusts the degree of sparsification by considering the desired model performance, the number of rounds, and the norm of gradients. We validate our theoretical analysis through experiments on classic image classification tasks (e.g. MNIST and CIFAR-10 datasets) and objection detection on the PASCAL VOC dataset. Numerical results show that AdapTop-K outperforms the baseline sparsification methods with or without error compensation.

To summarize, our key contributions are as follows:

- **Convergence analysis**: The theoretical findings of our research contribute to the understanding of the trade-off between communication budget and convergence error. We analyze the optimal convergence rate of the loss function by deriving an upper bound under general Top-K sparsification to gradients over different communication rounds with or without error compensation. We derive the additional term (called the adaptive term) in the convergence rate, which characterizes the impact of the degree of adaptive sparsification in the convergence rate.
- **Adapitive Top-K algorithm**: We solve the optimization problem that minimizes the convergence gap from the convergence rate with the adaptive term under the same communication cost with or without error compensation. We propose a novel adaptive Top-K algorithm to improve the model performance by dynamically adjusting the degree of sparsification in the training process. By quantifying the relationship between communication budget and convergence error, our study offers valuable insights and guidance for designing efficient and effective compression strategies in distributed learning systems.
- **Numerical Experiments**: To empirically validate our theoretical analysis, we conducted a comprehensive set of experiments on various machine learning tasks. In general, we evaluated our proposed approach on common image classification tasks using well-known datasets such as MNIST, CIFAR-10. Additionally, we also implement our methods on object detection using the PASCAL VOC dataset for distributed multi-robot collaborative learning. Our experimental results provide strong evidence of the effectiveness of our approach in mitigating communication costs. We observed significant improvements compared to state-of-the-art gradient compression methods. These improvements not only validate the practicality of our theoretical analysis but also underscore the potential of our approach in real-world scenarios. By achieving substantial reductions in communication costs, our approach contributes to the advancement of large-scale machine learning tasks, enabling more efficient and scalable training processes.

II. Related Work

In recent years, distributed learning has emerged as a promising technique for training deep neural networks, as it enables the use of large-scale datasets and exploiting the parallel computing power of multiple machines. This resurgence of interest in leveraging gradient compression for training deep neural networks highlights the importance of efficient communication schemes in large-scale distributed learning systems.

Fixed Gradient Compression. Several existing traditional studies primarily concentrate on fixed gradient compression strategies, which include gradient quantization and sparsification. In these approaches, the specific quantization bits or sparsification levels utilized during the training process are predetermined and remain unchanged throughout.

Gradient quantization compresses the gradient update by reducing the number of bits used to represent each weight update. For quantization, there are several variants including variance-reduced quantization [36], quantization to a ternary vector [37], and quantization of gradient difference [38]. Besides that, Sign SGD, as proposed in [19], adopts a quantization method that utilizes a single bit to quantize each dimension of the gradients. QSGD [29] and k-level quantization [18] introduce stochastic quantization schemes that enable the quantization of elements into arbitrary bits. These approaches offer flexibility in the choice of quantization levels, allowing for a more fine-grained representation of gradients during the training process.

The gradient sparsification techniques aim to reduce the amount of data transmitted across the network by sending only part of the global update. One common sparsification approach called Rand-K [31] is to randomly filter out part of the elements of gradients and amplify the remaining elements to keep the sparsified gradient unbiased. Alternatively, the Top-K sparsification approach [28] is a biased sparsifier that only keeps the largest k elements of the gradient vector, and sets the rest part to 0, where K is a predefined hyperparameter. In contrast to the unbiased schemes, the biased methods cannot keep the expectation stable. Intuitively, biased methods bring in more compression noise to the optimization process. In Federated Learning (FL), TCS [17] aims to establish a correlation between the sparse representations employed in consecutive iterations. This correlation minimizes the encoding overhead associated with transmitting non-zero positional information. PowerSGD [16] adopts a low-rank linear transformation technique to introduce sparsity into the model, thereby reducing the number of parameters involved. It improved scalability for FL systems while maintaining satisfactory model performance.

In recent research, there has been a growing interest in leveraging both sparsification and quantization techniques [24] [23] to achieve enhanced communication efficiency. Specifically, [24] proposes a method that combines aggressive sparsification with quantization by tracking the difference between the original and compressed gradients to maintain the fidelity of the gradient updates. Similarly, in [23], the importance of gradients is determined by their magnitude. Gradients with magnitudes exceeding a certain threshold are quantized to a fixed number of bits and transmitted, allowing efficient communication while prioritizing significant updates.

Adaptive Gradient Compression. From reality training and observation, adopting a fixed compression level through the entire training duration becomes unwise. Recent research has started to develop adaptive compression schemes that dynamically determine the compression level based on empirical observations or engineering heuristics.

For instance, [21] and [20] determine the compression level based on the size of the gradients. MQGrad [22] formulates the quantization determination as an online learning problem, utilizing historical information from past optimization iterations. AdaComp [15] implements a localized selection approach to gradient residues, automatically adjusting the compression rate based on local training. MIPD [14] adaptively compresses gradients by considering model interpretability and the probability distribution of gradients.

In contrast to the heuristic compression schemes mentioned above, recent works [13] and [12] propose adaptive compression techniques from a theoretical perspective. [13] adaptively adjusts the quantization points to minimize the variance of vector quantization, while [12] dynamically computes scaling factors for integer rounding operators. The consideration of communication budget constraints in adaptive compression occurs in [32]. They unify the dynamic adjust gradient compression methods but withour considering the error compensation. It is crucial to develop practical and effective adaptive compression techniques that not only for pure SGD but also consider error compensation because of a wide range of practical applications.

Error Compensation. To compensate for the compressed gradient errors by adding a memory vector and to accelerate the learning speed, various error compensation techniques have been introduced in the literature. For example, ScaleCom [11] explores the similarity of gradient distribution across clients to provide scalable error compensation for Top-k compressors. CSER [10] employs an error compensation method called error reset to enhance the learning speed of compressors. These techniques contribute to improving the efficiency and performance of distributed learning systems by mitigating compression errors and facilitating faster convergence.

To summarize, the existing works either rely on predetermined fixed compression levels or utilize engineering heuristics to adjust the compression level. However, these approaches sometimes yield contradictory conclusions. For instance, MQGrad [22] and AdaQS [34] suggests using fewer bits in early epochs increasing it in later epochs, whereas Anders [21] suggests the opposite. Additionally, error compensation techniques are employed to accelerate learning speeds for different compressors but no research considers theoretical analysis to add it. Some of the ideas presented in this work were previously discussed in [1], which introduced the theoretical framework for adaptive Top-K without error compensation. This work extends that framework to provide theoretical analysis for AdapTop-K in SGD with error compensation. To the best of our knowledge, our proposed AdapTop-K in SGD is the first to systematically consider the communication budget in adaptively adjusting the compression level based on the gradient norm, the number of training iterations, and the available communication budget and provide convergence analysis with or without error compensation.

III. SYSTEM MODEL

We consider a distributed learning system including a central server and M edge devices (workers). All the workers collaboratively aim to train a shared machine learning model via gradient (or its variant) aggregation with the cooperation of the central server.

A. Learning Model

The learning model is represented by the vector of its parameters $\mathbf{w} \in \mathbb{R}^d$, where d is the model size. The datasets are distributed over the M workers. We use $\mathcal{D}^i$ to denote the local dataset at worker i. The global loss function, denoted by $F : \mathbb{R}^d \to \mathbb{R}$, is defined as

$$F(\mathbf{w}) = \frac{1}{M} \sum_{i=1}^{M} f^i(\mathbf{w}), \quad \text{with } f^i(\mathbf{w}) = \mathbb{E}_{\xi^i \sim \mathcal{D}^i} \left[l^i(\mathbf{w}; \xi^i) \right], \tag{1}$$

where $l^i(\mathbf{w}; \xi^i)$ is the local loss function of the model parameters $\mathbf{w}$ at work i, given the mini-batch ξ^i randomly selected from worker i's local dataset $\mathcal{D}^i$.

The objective of the training is to find a model parameter $\mathbf{w}$ to minimize the global loss function in Eq. (1) :

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} F(\mathbf{w}). \tag{2}$$

The distributed SGD is the most popular method to solve this problem, where each worker i computes its local stochastic gradient $f^t_{i,\xi}(\mathbf{w}) = \nabla l^i(\mathbf{w}_t; \xi^i)$ given parameters $\mathbf{w}_t$ at round t. Then the workers send the local gradient $f^t_{i,\xi}(\mathbf{w})$ to the central server. The server aggregates these gradients to update the model. To reduce the communication cost, we compress the local stochastic gradients before sending them to the server:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{M} \sum_{i=1}^{M} \mathcal{C}^i[f^t_{i,\xi}(\mathbf{w})], \tag{3}$$

where η_t is the learning rate at iteration t, and $\mathcal{C}^i[\cdot]$ is the compression operator. Without the gradient compressor, Eq. (3) reduces to the vanilla distributed SGD with $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{M} \sum_{i=1}^{M} f^t_{i,\xi}(\mathbf{w})$. The same procedure is repeated until the convergence criterion or the maximum number of communication rounds is reached.

A commonly-used compression operator is Top-K, where each worker i keeps only k elements of the gradient $\mathbf{g}^i_t$ with the largest magnitudes and sets the other elements to zero [28]. In this work, we speed up the convergence of Top-K by adaptively choosing the sparsity of the gradient during the convergence process. Specifically, given a total of T rounds of gradient update, our goal is to find the optimal sparsity levels $k_0, \ldots, k_{T-1}$ in each round, so that the final model is as close to the optimal model as possible. It is natural to measure the gap from the optimal model by the difference between the expectation of the final global loss $F(\mathbf{w}_T)$ and the optimal loss $F^* = F(\mathbf{w}^*)$. Note that we need to take expectation of the final loss $F(\mathbf{w}_T)$ due to the stochastic gradient descent. Therefore, our design problem can be formulated as follows

$$\begin{aligned} \min_{k_0,\ldots,k_{T-1}} \quad & \mathbb{E}\left[F(\mathbf{w}_T)\right] - F^* \\ \text{s.t.} \quad & \textstyle\sum_{t=0}^{T-1} k_t \leq K, \\ & k_t \in \{0, 1, \ldots, d\},\ t = 0, \ldots, T-1, \end{aligned} \tag{4}$$

where K is the total budget for the communication overhead during the training. When comparing with other sparsification methods, we can set the communication budget K accordingly.

B. Basic Assumptions on Learning Model

To promote the convergence analysis, we make several basic assumptions on the stochastic gradient and loss functions that are commonly used in the literature [32], [26], and [25].

Assumption 1: (Smoothness). Let $\nabla F(\mathbf{w})$ denote the gradient of the loss function evaluated at parameter $\mathbf{w} \in \mathbb{R}^d$. If $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, there exists a non-negative constant L satisfying:

$$F(\mathbf{x}) - F(\mathbf{y}) - \langle \nabla F(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \leq \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|^2. \tag{5}$$

Assumption 1 guarantees the Lipschitz continuity of the gradient of the loss function, is crucial for the convergence analysis of gradient descent methods. This assumption provides a necessary condition for controlling the rate of change of the loss function with respect to the parameter vector. With the Lipschitz continuity assumption, gradient descent algorithms can ensure that the updates to the parameter vector are small and controlled, allowing for a stepwise approach towards the minimum loss. In the absence of this assumption, the convergence of these algorithms could be jeopardized by overshooting or oscillation, making it difficult to determine a suitable step size for updating the parameters. Therefore, the Lipschitz continuity assumption provides an important condition for the efficient optimization of loss functions using gradient descent methods.

Assumption 2: (Polyak-Lojasiewicz Condition). Let F^* denote the optimal loss function value to Eq. (2). There exists a constant $\mu \geq 0$ such that the global loss function $F(\mathbf{w})$ satisfies the following Polyak-Lojasiewicz condition:

$$\|\nabla F(\mathbf{w})\|^2 \geq 2\mu(F(\mathbf{w}) - F^*). \tag{6}$$

Notice that Assumption 2 is more general than the general assumption of strong convexity (as Assumption 5) [33]. The inequality in Eq. (6) display the crucial property of having gradients that grow at a rate that is at least quadratic when they are not at the optimal function value. By satisfying these assumptions, these loss functions are amenable to optimization using gradient descent algorithms, which can converge to the global minimum with provable guarantees.

Assumption 3: (Unbiasedness and Bounded Variance of Stochastic Gradient). The local stochastic gradients $\nabla f^t_{i,\xi}(\mathbf{w})$ are assumed to be independent and unbiased estimates of the mini-batch gradient $\nabla F(\mathbf{w})$ with bounded variance:

$$\begin{aligned} \mathbb{E}_{\xi \sim \mathcal{D}_i}[\nabla f^i_\xi(\mathbf{w}_t)] = \nabla f^i(\mathbf{w}_t), \\ \mathbb{E}_{\xi \sim \mathcal{D}_i}[\|\nabla f^i_\xi(\mathbf{w}_t) - \nabla f^i(\mathbf{w}_t)\|^2] \leq \sigma^2. \end{aligned} \tag{7}$$

Assumption 4: (Upper Bound of Sample-wise Gradient). At any communication round t on the worker i, the gradient $\nabla f^t_i(\mathbf{w})$ for any training sample is upper bounded by a given contant G as:

$$\mathbb{E}[\left\|\nabla f^i(\mathbf{w}_t)\right\|^2] \leq G^2. \tag{8}$$

Assumption 5: (Strongly Convex). Let $\nabla F(\mathbf{w})$ denote the gradient of the loss function evaluated at parameter $\mathbf{w} \in \mathbb{R}^d$. If $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, there exists a non-negative constant μ satisfying:

$$F(\mathbf{x}) - F(\mathbf{y}) - \langle \nabla F(y), \mathbf{x} - \mathbf{y} \rangle \geq \frac{\mu}{2} \|\mathbf{x} - \mathbf{y}\|^2. \tag{9}$$

Assumption 5 guarantees the μ-stronly convex of the gradient of the loss function, is crucial for the convergence analysis of gradient descent methods. This assumption provides a necessary condition for the gradient must grow faster than a quadratic function as it moves away from the optimal function value.

Notice that the following convergence analysis in Sections IV and V are both based on Assumptions (5) and (7), the Assumption (6) only is used in Section IV, and the Assumption (8) and (9) only is used in Section V, similarly as in prior work [28].

IV. Adaptive Top-K in SGD without error compensation

A. Convergence Rate

In this section, we present a convergence analysis for the AdapTop-K in SGD by using the optimality gap. The standard optimization iterations update is Eq. 3. Inspired by [27], we rewrite the optimization process as:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{C}[\nabla F_\xi(\mathbf{w}_t)], \tag{10}$$

where $\mathcal{C}[\cdot]$ represents the Top-K operator here. We regard the stochastic gradient $\nabla F_\xi(\mathbf{w}_t)$ and the compressed gradient $\mathcal{C}[\nabla F_\xi(\mathbf{w}_t)]$ as:

$$\begin{aligned} \nabla F_\xi(\mathbf{w}_t) &= \nabla F(\mathbf{w}_t) + \mathbf{c}_t(\mathbf{w}_t), \\ \mathcal{C}[\nabla F_\xi(\mathbf{w}_t)] &= \nabla F(\mathbf{w}_t) + \mathbf{b}_t(\mathbf{w}_t) + \mathbf{c}_t(\mathbf{w}_t), \end{aligned} \tag{11}$$

for every variable $\mathbf{a}_t = \sum_{i=1}^{M} a_t^i$, where $\mathbf{c}_t(\mathbf{w}_t)$ is the noise term made by stochastic samples and $\mathbf{b}_t(\mathbf{w}_t)$ is a biased term made by the Top-K method.

By Assumption 3, the noise has zero mean and bounded variance, namely

$$\mathbb{E}[\mathbf{c}_t(\mathbf{w}_t)] = 0 \quad \text{and} \quad \mathbb{E}[\|\mathbf{c}_t(\mathbf{w}_t)\|^2] \leq \sigma^2. \tag{12}$$

Lemma 1: (Bounded Variance of Stochastic Gradient with Top-K sparsification). There exists an assumption for the Top-K sparsification method in gradient update. The variance of the bias $\mathbf{b}_t(\mathbf{w}_t)$ is upper bounded with the mini-batch gradient $\nabla F_\xi(\mathbf{w}_t)$ [28]:

$$||\mathbf{b}_t(\mathbf{w})||^2 \leq (1-\frac{k}{d})\|\nabla F_\xi(\mathbf{w}_t)\|^2. \tag{13}$$

With Lemma 1, we prove an upper bound of the optimality gap under the adaptive sparsity levels of $k_0, \ldots, k_{T-1}$.

Theorem 1: (Upper Bound for Convergence Error). For the problem in Eq. 1 under Assumption 1, 2, and 3 with initial parameter $\mathbf{w}_0$ and stable stepsize $\eta_t = \eta \leq \frac{1}{L}$, using Top-K gradients with Lemma 1 for iterations, the optimality gap of the adaptive Top-K method is upper bounded by:

$$\begin{aligned} \mathbb{E}[F(\mathbf{w}_T)] - F^* \leq & \underbrace{(1-\frac{\eta\mu}{d}k)^T(\mathbb{E}(F(\mathbf{w}_0)-F^*)}_{\mathbf{M}(k)} \\ & \underbrace{+\frac{d}{2k\mu}(1-\frac{k}{d}+\eta L)\sigma^2[1-(1-\frac{\eta\mu}{d}k)^T]}_{\mathbf{N}(k)} \\ & \underbrace{-\sum_{t=0}^{T-1}[(\frac{\eta n_t}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2)(1-\frac{\eta\mu}{d}k)^{T-1-t}]}_{\text{only this term is affected by } n_t}. \end{aligned} \tag{14}$$

where $k = \frac{K}{T}$ is the average sparsity level and $n_t = k_t - k$ is the deviation from the average sparsity level at round t.

Proof: See the appendix. ■

The upper bound in (14) has two parts. The first part is the sum of the first two terms $\mathbf{M}(k) + \mathbf{N}(k)$, which depends only on the average sparsity level k. The second part is the third term, which is the only term that depends on $n_0, \ldots, n_{T-1}$. When $n_t = 0$ for all t, the upper bound reduces to $\mathbf{M}(k) + \mathbf{N}(k)$, namely the bound for the vanilla Top-K method.

B. The Proposed AdapTop-K Algorithm (without error compensation)

We aim to minimize the upper bound of the optimality gap in (14) by choosing $n_0, \ldots, n_{T-1}$. Since only the third term depends on the adjustments $n_0, \ldots, n_{T-1}$, the optimization problem can be formulated as

$$\begin{aligned} \max_{n_0,\ldots,n_{T-1}} \quad & \sum_{t=0}^{T-1} \left(\frac{\eta n_t}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2\right)\left(1-\frac{\eta\mu}{d}k\right)^{T-1-t} \\ \text{s.t.} \quad & \sum_{t=0}^{T-1} n_t \leq 0, \\ & n_t \in \{-k, \ldots, d-k\},\ t = 0, \ldots, T-1, \end{aligned} \tag{15}$$

where the first constraint comes from the constraint on the communication overhead in (4) and the second constraint comes from the fact that $k_t \in \{0, \ldots, d\}$.

Since the objective function is linear in n_t, the optimal solution should assign the largest possible values to the n_t's with the largest coefficients

$$\left(\frac{\eta}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2\right)\left(1-\frac{\eta\mu}{d}k\right)^{T-1-t}, \tag{16}$$

subject to the upper bound $d-k$ and the budget of total communication overhead. However, the major challenge is that the coefficients in (16) depend on the gradients $\nabla F_\xi(\mathbf{w}_t)$, which are stochastic due to the randomly selected mini-batches and are dependent on our choice of sparsity levels $n_0, \ldots, n_{t-1}$ up to round t. Therefore, we cannot solve the optimization problem (15) directly. Instead, we choose to maximize an upper bound of the objective function, which is obtained by bounding the norm of the stochastic gradients $\nabla F_\xi(\mathbf{w}_t)$.

Lemma 2: (Upper Bound for Stochastic Gradient). Under Assumptions 1–3, given the initial parameter $\mathbf{w}_0$ and constant stepsize $\eta_t = \eta \leq \frac{1}{L}$, the stochastic gradient in Eq. (11) can be upper bounded by

$$\mathbb{E}[\|\nabla F_\xi(\mathbf{w}_t)\|^2] \leq \frac{2d}{k\eta} \cdot \frac{F(\mathbf{w}_0)}{t} + \frac{d\sigma^2}{k}(\eta L + 1) \triangleq \frac{\alpha}{t} + \beta \tag{17}$$

where $\alpha \triangleq \frac{2d}{k\eta}F(\mathbf{w}_0)$ and $\beta \triangleq \frac{d\sigma^2}{k}(\eta L + 1)$.

Based on Lemma 2, we obtain the following upper bound of the objective function in (15)

$$\begin{aligned} & \frac{\eta}{2d}\sum_{t=0}^{T-1}\left[\left(\frac{\alpha}{t}+\beta\right)\left(1-\frac{\eta\mu}{d}k\right)^{T-1-t}\right] \cdot n_t \\ \triangleq \quad & \frac{\eta}{2d}\sum_{t=0}^{T-1}(A_t B_t) \cdot n_t, \end{aligned} \tag{18}$$

where $A_t \triangleq \frac{\alpha}{t} + \beta$ and $B_t \triangleq (1-\frac{\eta\mu}{d}k)^{T-1-t}$.

Finally, the optimization problem to solve is

$$\begin{aligned} \max_{n_0,\ldots,n_{T-1}} \quad & \frac{\eta}{2d}\sum_{t=0}^{T-1}(A_t B_t) \cdot n_t \\ \text{s.t.} \quad & \sum_{t=0}^{T-1} n_t \leq 0, \\ & n_t \in \{-k, \ldots, d-k\},\ t = 0, \ldots, T-1. \end{aligned} \tag{19}$$

The objective function in (19) is linear in n_t with coefficient $A_t B_t$. We can prove the following monotonicity results.

Lemma 3: The coefficient A_tB_t first decreases with t and then increases with t. Specifically, we have

$$\begin{aligned} &A_{t+1}B_{t+1} < A_tB_t, \text{ for } t < \hat{t} \triangleq \left\lfloor \frac{-\alpha+\sqrt{\Delta}}{2\beta} \right\rfloor, \text{ and} \\ &A_{t+1}B_{t+1} \geq A_tB_t, \text{ for } t \geq \hat{t}, \end{aligned} \tag{20}$$

where $\Delta \triangleq \alpha^2 - \frac{4\alpha\beta}{lnB}$, $B \triangleq 1 - \frac{\eta\mu}{d}k$, and $\lfloor\cdot\rfloor$ is the floor function.

Given the monotonicity result in Lemma 3, we design the following adaptive sparsity levels

$$\begin{cases} n_t = +\gamma k \Rightarrow k_t = (1+\gamma)k, & t \in [0, \frac{\hat{t}}{2}) \cup [\frac{\hat{t}+T}{2}, T-1] \\ n_t = -\gamma k \Rightarrow k_t = (1-\gamma)k, & t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2}), \end{cases} \tag{21}$$

where γ is the scaling factor (i.e., a hyperparameter). In the above scheme, n_t takes the negative value half the training time and the positive value the other half, which satisfies the communication budget constraint. To maximize the objective function, we set n_t to be positive when A_tB_t is larger.

We can prove that the above adaptive sparsity levels result in a lower convergence error compared to the vanilla Top-K.

Corollary 1: (Convergence Error Bound using AdapTop-K in distributed SGD). Under the adaptive sparsity levels in Eq. (21), the optimality gap is upper bounded by

$$\begin{aligned} \mathbb{E}[F(\mathbf{w}_T)] - F^* &\leq \mathbf{M}(k) + \mathbf{N}(k) \\ &+ \frac{\eta\gamma k}{2d} \underbrace{\left(\sum_{t=\frac{\hat{t}}{2}}^{\frac{\hat{t}+T-1}{2}} A_tB_t - \sum_{t=0}^{\frac{\hat{t}}{2}} A_tB_t - \sum_{t=\frac{\hat{t}+T-1}{2}}^{T-1} A_tB_t \right)}_{\text{always less than 0 because of (9)}} \\ &< \underbrace{\mathbf{M}(k) + \mathbf{N}(k)}_{\text{upper bound for SGD with vanilla Top-K}} . \end{aligned} \tag{22}$$

The pseudo-code of distributed SGD with the proposed AdapTop-K method is provided in Algorithm 1.

V. Adaptive Top-K in SGD with Error Compensation

A. Convergence Rate

In this section, we present a convergence analysis for the AdapTop-K with error compensation in SGD by using the optimality gap. The standard SGD optimization iterations update is Eq. 3. However, the SGD with error compensation is different with standard version. We consider the following optimization algorithm for parameter $0 < k \leq d$, using a compression factor $\mathcal{C}_k : \mathbb{R}^d \to \mathbb{R}^d$ which is a Top-K compression operator.

$$\begin{cases} \mathbf{w}_{t+1} = \mathbf{w}_t - \mathcal{C}(\mathbf{g}_t) \\ \mathbf{g}_t = \mathbf{m}_t + \eta\nabla F_\xi(\mathbf{w}_t) \\ \mathbf{m}_{t+1} = \mathbf{g}_t - \mathcal{C}(\mathbf{g}_t), \end{cases} \tag{23}$$

where the $\nabla F_\xi(\mathbf{w}_t)$ is stochastic gradient of loss function, $\mathbf{m}$ is the memory vector using in error compensation with $\mathbf{m}_0 := 0$ and η denotes a sequence of stepsizes. Note that the gradients get multiplied with the stepsize η at the timestep t when they are put into memory, and not when they are retrieved from the memory.

Algorithm 1 AdapTop-K in Distributed SGD

```
Input: Maximum iterations number $T$, learning rate $\eta$, initial
    point $\mathbf{w}_0 \in \mathbb{R}^d$, fixed $k$ value, adjusted scale factor $\gamma$,
    hyper-parameters $\hat{t}$
Output: $\mathbf{w}_t$
for $t = 0, 1, ...T-1$ do
    On each worker $i = 1, ..., M$:
    Compute stochastic local gradient $\nabla f_\xi^i(\mathbf{w}_t)$
    if $t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2})$ then
        Set $k_t$ to $k - \gamma k$
    else
        Set $k_t$ to $k + \gamma k$
    end if
    Compress gradient $\nabla f_\xi^i(\mathbf{w}_t)$ to $\mathcal{C}_{k_t}[\nabla f_\xi^i(\mathbf{w}_t)]$
    Send $\mathcal{C}_{k_t}[\nabla f_\xi^i(\mathbf{w}_t)]$ to server
    Receive $\mathbf{w}_{t+1}$ from server
    On server:
    Collect $M$ compressed gradients $\mathcal{C}_{k_t}[\nabla f_\xi^i(\mathbf{w}_t)]$ from
    workers
    Aggregation: $\mathcal{C}_{k_t}[\nabla F_\xi(\mathbf{w}_t)] = \sum_{i=1}^{M} \mathcal{C}_{k_t}[\nabla f_\xi^i(\mathbf{w}_t)]$
    Update global parameters: $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta}{M}\mathcal{C}_{k_t}[\nabla F_\xi(\mathbf{w}_t)]$
    Send $\mathbf{w}_{t+1}$ back to all workers
end for
```

For the convergence analysis, we first need the perturbed iterated analysis. Inspired by the perturbed iterate framework in [28] and [39], we define a virtual sequence $\{\tilde{\mathbf{w}}_t\}_{t\geq 0}$ in the following way to analysis the convergence rate at first:

$$\tilde{\mathbf{w}}_0 = \mathbf{w}_0 \,, \qquad \tilde{\mathbf{w}}_{t+1} = \tilde{\mathbf{w}}_t - \eta\nabla F_i(\mathbf{w}_t) \,, \tag{24}$$

where the sequences $\{\mathbf{w}_t\}_{t\geq 0}$, variable η are the same as in (23). Notice that

$$\tilde{\mathbf{w}}_t - \mathbf{w}_t = \Big(\mathbf{w}_0 - \textstyle\sum_{j=0}^{t-1} \eta\nabla f_i(\mathbf{w}_j)\Big) - \Big(\mathbf{w}_0 - \textstyle\sum_{j=0}^{t-1} \mathcal{C}(\mathbf{g}_j)\Big) = \mathbf{m}_t \,. \tag{25}$$

Lemma 4: Let $\{\mathbf{w}_t\}_{t\geq 0}$ and $\{\tilde{\mathbf{w}}_t\}_{t\geq 0}$ be defines as in (23) and (24) and let the loss function f_i be L-smooth and f be μ-strongly convex with $\mathbb{E}\|\nabla F_i(\mathbf{w}_t)\|^2 \leq G^2$. Then we have

$$\begin{aligned} \mathbb{E}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^*\|^2 \leq (1 - \frac{\mu\eta}{2})\mathbb{E}\|\tilde{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta^2G^2 \\ - \eta(\mathbb{E}[F(\mathbf{w}_t)] - F^*) + \eta(\mu + 2L)\mathbb{E}\|\mathbf{m}_t\|^2. \end{aligned} \tag{26}$$

We know that the compression ratio only affects the memory vectors. From the above theorem, we separated the terms influenced by the memory vector from the stable terms in the convergence rate.

We state the precise convergence result for Top-K with error compensation in Theorem 2 below.

Lemma 5: (Initial Upper Bound for Convergence Rates of Top-K with error compensation). For the problem in Eq. 1 under Assumption 1, 4, 5 for $t \in [0, T]$ with initial parameter $\mathbf{w}_0$ and stable stepsize $\eta_t = \eta \leq \frac{1}{L}$, using Top-K gradients

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_T)] - F^* \le & \underbrace{(1-\frac{\mu\eta}{2})^T(\frac{1}{\eta}-\frac{\mu}{2})\mathbb{E}\|\mathbf{w}_0-\mathbf{w}^*\|^2 + \eta G^2 + (\frac{1}{\eta}-\frac{\mu}{2})(\frac{2}{\mu\eta}-1)[1-(1-\frac{\eta\mu}{2})^T]\eta^2 G^2}_{\mathbf{P}} \\ & \underbrace{+(\mu+2L)(1-\frac{k}{d})\mathbb{E}\|\mathbf{g}_{T-1}\|^2 + (1-\frac{\mu\eta}{2})(\mu+2L)(1-\frac{k}{d})\sum_{t=1}^{T-1}(1-\frac{\mu\eta}{2})^{T-t}\mathbb{E}\|\mathbf{g}_{t-1}\|^2}_{\mathbf{Q}(k)} \\ & \underbrace{-(\mu+2L)\frac{n_{T-1}}{d}\mathbb{E}\|\mathbf{g}_{T-1}\|^2 + (1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})\sum_{t=1}^{T-1}(1-\frac{\mu\eta}{2})^{T-t}n_{t-1}\mathbb{E}\|\mathbf{g}_{t-1}\|^2}_{\text{only this term is affected by } n_t}. \qquad (29)\end{aligned}$$

with error compensation satisfied Lemma 4, we can upper bound the convergence error by:

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_T)] - F^* \le & (1-\frac{\mu\eta}{2})^T(\frac{1}{\eta}-\frac{\mu}{2})\mathbb{E}\|\mathbf{w}_0-\mathbf{w}^*\|^2 + \eta G^2 \\ & +(\mu+2L)\mathbb{E}\|\mathbf{m}_T\|^2 + (\frac{1}{\eta}-\frac{\mu}{2})(\frac{2}{\mu\eta}-1)[1-(1-\frac{\eta\mu}{2})^T]\eta^2 G^2 \\ & + (1-\frac{\mu\eta}{2})(\mu+2L)\sum_{t=0}^{T-1}(1-\frac{\mu\eta}{2})^{T-t}\mathbb{E}\|\mathbf{m}_t\|^2\end{aligned} \qquad (27)$$

From convergence rate equation (27), it becomes clear that we should derive an upper bound on $\mathbb{E}\|\mathbf{m}_t\|^2$.

Lemma 6: (Bounded Variance of Stochastic Gradient with Top-K sparsification with error compensation). There exists an assumption for the Top-K sparsification method in gradient update with error compensation. Because we import the memory vector $\mathbf{m}$, the biased term $(\mathbf{g}_t - \mathcal{C}(\mathbf{g}_t))$ is assumed to have a bounded variance with the mini-batch gradient $\nabla F_\xi(\mathbf{w}_t)$ [28]:

$$||\mathbf{m}_{t+1}(\mathbf{w})||^2 \le (1-\frac{k}{d})\|\mathbf{g}_t\|^2 - \frac{n_t}{d}\|\mathbf{g}_t\|^2. \qquad (28)$$

We successfully separate the fixed k and dynamic n_t in the convergence error bound with error compensation to find which part is influenced by the dynamic term n_t. The algorithm degrades to the vanilla Top-K method with error compensation when $n_t = 0$.

Theorem 2: (Upper Bound for Convergence Rates of Top-K with error compensation). Based on the conditions proposed in Lemma 5 and Lemma 6, the upper bound of the convergence error using Top-K gradients with error compensation satisfies the inequality (29):

Proof: See the appendix. ■

The upper bound in (29) has two parts. The first part is the sum of the first two terms $\mathbf{P}$ and $\mathbf{Q}(k)$, which depends only on the hyperparameters and the average sparsity level k. The second part is the third term, which is the only term that depends on $n_0, \ldots, n_{T-1}$. When $n_t = 0$ for all t, the upper bound reduces to $\mathbf{P} + \mathbf{Q}(k)$, namely the bound for the vanilla Top-K method with error compensation.

B. The Proposed AdapTop-K Algorithm (with error compensation)

Similarly, we aim to design the AdapTop-K algorithm with error compensation to improve the convergence performance under fixed communication cost. We aim to minimize the upper bound of the optimality gap in (29) by choosing $n_0, \ldots, n_{T-1}$. Since only the third term depends on the adjustments $n_0, \ldots, n_{T-1}$, the optimization problem can be formulated as

$$\begin{aligned}\max_{n_0,\ldots,n_{T-1}} \; & (\mu+2L)\frac{n_{T-1}}{d}\mathbb{E}\|\mathbf{g}_{T-1}\|^2 \\ & +(1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})\sum_{t=1}^{T-1}(1-\frac{\mu\eta}{2})^{T-t}n_{t-1}\mathbb{E}\|\mathbf{g}_{t-1}\|^2 \\ \text{s.t.} \; & \sum_{t=0}^{T}(k+n_t) = K \Leftrightarrow \sum_{t=0}^{T} n_t = 0, \\ & n_t \in \{-k, \ldots, d-k\}, \; t = 0, \ldots, T-1,\end{aligned} \qquad (30)$$

where the first constraint comes from the constraint on the communication overhead in (4) and the second constraint comes from the fact that $k_t \in \{0, \ldots, d\}$. The K is the total communication budget. When considering the number of the total communication cost by bits, the budget K is equal to $(k+n_t)(32+log_2 d)$ because the number of bits to represent a float number is 32.

Since the objective function is linear in n_t, the optimal solution should assign the largest possible values to the n_t's with the largest coefficients

$$(1-\frac{\mu\eta}{2})^{T-t-1}\mathbb{E}\|\mathbf{g}_t\|^2, \qquad (31)$$

subject to the upper bound $d-k$ and the budget of total communication overhead. However, the major challenge is that the coefficients in (31) depend on the gradients $\mathbf{g}_t$, which are the combination of stochastic gradients using the randomly selected mini-batches and the memory vector. Those are dependent on our choice of sparsity levels $n_0, \ldots, n_{t-1}$ up to round t. Therefore, we cannot solve the optimization problem (30) directly. Instead, we choose to maximize an upper bound of the objective function, which is obtained by bounding the norm of the stochastic gradients $\mathbf{g}_t$.

Lemma 7: (Upper Bound for Stochastic Gradient with error compensation). Under Assumptions 1–3, given the initial parameter $\mathbf{w}_0$ and constant stepsize $\eta_t = \eta \le \frac{1}{L}$, the stochastic gradient in Eq. (23) can be upper bounded by

$$\begin{aligned}\mathbb{E}[\|\hat{\mathbf{g}}_t\|^2] &\le \frac{2d}{k\eta}\cdot\frac{F(\mathbf{w}_0)}{t} + \frac{d\sigma^2}{k}(\eta L+1) \\ \mathbb{E}[\|\mathbf{g}_t\|^2] &= \eta^2\mathbb{E}[\|\hat{\mathbf{g}}_t\|^2] \\ &\le \frac{2d\eta}{k}\cdot\frac{F(\mathbf{w}_0)}{t} + \frac{d\sigma^2\eta^2}{k}(\eta L+1) \triangleq \frac{\alpha}{t} + \beta\end{aligned} \qquad (32)$$

where $\alpha \triangleq \frac{2d\eta}{k}F(\mathbf{w}_0)$, $\beta \triangleq \frac{d\sigma^2\eta^2}{k}(\eta L+1)$ and $\hat{\mathbf{g}_t} = \frac{1}{\eta}\mathbf{g}_t$. The proof and more details are in the appendix.

Based on Lemma 7, we obtain the following upper bound of the objective function in (30)

$$
\begin{aligned}
&(1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})\sum_{t=0}^{T-1}\left[\left(\frac{\alpha}{t}+\beta\right)\left(1-\frac{\eta\mu}{2}\right)^{T-1-t}\right]\cdot n_t \\
\triangleq\ &(1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})\sum_{t=0}^{T-1}(A_tB_t)\cdot n_t, \qquad (33)
\end{aligned}
$$

where $A_t \triangleq \frac{\alpha}{t}+\beta$ and $B_t \triangleq (1-\frac{\eta\mu}{2})^{T-1-t}$.

Finally, the optimization problem to solve is

$$
\begin{aligned}
\max_{n_0,\ldots,n_{T-1}} \quad & (1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})\sum_{t=0}^{T-1}(A_tB_t)\cdot n_t \qquad (34)\\
\text{s.t.} \quad & \sum_{t=0}^{T-1} n_t \le 0,\\
& n_t \in \{-k,\ldots,d-k\}\,,\ t=0,\ldots,T-1.
\end{aligned}
$$

The objective function in (34) is linear in n_t with coefficient A_tB_t. We can prove the following monotonicity results.

Lemma 8: The coefficient A_tB_t first decreases with t and then increases with t. Specifically, we have

$$
\begin{aligned}
&A_{t+1}B_{t+1} < A_tB_t,\ \text{for } t < \hat{t} \triangleq \left\lfloor \frac{-\alpha+\sqrt{\Delta}}{2\beta} \right\rfloor,\ \text{and}\\
&A_{t+1}B_{t+1} \ge A_tB_t,\ \text{for } t \ge \hat{t}, \qquad (35)
\end{aligned}
$$

where $\Delta \triangleq \alpha^2 - \frac{4\alpha\beta}{lnB}$, $B \triangleq 1-\frac{\eta\mu}{2}$, and $\lfloor\cdot\rfloor$ is the floor function.

Given the monotonicity result in Lemma 8, we design the following adaptive sparsity levels

$$
\begin{cases}
n_t = +\gamma k \Rightarrow k_t = (1+\gamma)k, & t \in [0,\frac{\hat{t}}{2}) \cup [\frac{\hat{t}+T}{2}, T-1]\\
n_t = -\gamma k \Rightarrow k_t = (1-\gamma)k, & t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2}),
\end{cases} \qquad (36)
$$

where γ is the scaling factor (i.e., a hyperparameter). The above scheme is similar with Eq. (21) and n_t takes the negative value half the training time and the positive value the other half, which satisfies the communication budget constraint. To maximize the objective function, we set n_t to be positive when A_tB_t is larger.

We can prove that the above adaptive sparsity levels result in a lower convergence error compared to the vanilla Top-K with error compensation.

Corollary 2: (Convergence Error Bound using AdapTop-K in distributed SGD with error compensation). Under the adaptive sparsity levels in Eq. (36), the D is $(1-\frac{\mu\eta}{2})(\frac{\mu+2L}{d})$ and the optimality gap is upper bounded by:

$$
\begin{aligned}
&\mathbb{E}[F(\mathbf{w}_T)] - F^* \le \mathbf{P} + \mathbf{Q}(k)\\
&\quad +\gamma k D \underbrace{\left(\sum_{t=\frac{\hat{t}}{2}}^{\frac{\hat{t}+T-1}{2}} A_tB_t - \sum_{t=0}^{\frac{\hat{t}}{2}} A_tB_t - \sum_{t=\frac{\hat{t}+T-1}{2}}^{T-1} A_tB_t\right)}_{\text{always less than 0 because of (35)}}\\
&\quad < \underbrace{\mathbf{P} + \mathbf{Q}(k)}_{\text{upper bound for SGD with vanilla Top-K}} . \qquad (37)
\end{aligned}
$$

Algorithm 2 AdapTop-K with error compensation in Distributed SGD

```
Input: Maximum iterations number $T$, learning rate $\eta$, initial
    point $\mathbf{w}_0 \in \mathbb{R}^d$, fixed $k$ value, adjusted scale factor $\gamma$,
    hyper-parameters $\hat{t}$, initial memory vector $\mathbf{m}_0 = 0$
Output: $\mathbf{w}_t$
for $t = 0, 1, ...T-1$ do
    On each worker $i = 1, ..., M$:
    Compute stochastic local gradient $g_i^t$
    if $t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2})$ then
        Set $k_t$ to $k - \gamma k$
    else
        Set $k_t$ to $k + \gamma k$
    end if
    $\mathbf{g}_{i,t} \leftarrow \mathbf{m}_{i,t} + \eta\nabla f_{i,\xi}^t(\mathbf{w})$
    Compress gradient $\mathbf{g}_{i,t}$ to $\mathcal{C}_{k_t}[\mathbf{g}_{i,t}]$
    Send $\mathcal{C}_{k_t}[\mathbf{g}_{i,t}]$ to server
    Update memory vector $\mathbf{m}_{i,t+1} \leftarrow \mathbf{m}_{i,t} + \eta\nabla f_{i,\xi}^t(\mathbf{w}) -$
        $\mathbf{g}_{i,t}$
    Receive $\mathbf{w}_{t+1}$ from server
    On server:
    Collect $M$ compressed gradients $\mathcal{C}[\mathbf{g}_{i,t}]$ from workers
    Aggregation: $\mathcal{C}_{k_t}[\mathbf{g}_t] = \sum_{i=1}^{M} \mathcal{C}_{k_t}[\mathbf{g}_{i,t}]$
    Update global parameters: $\mathbf{w}_{t+1} = \mathbf{w}_{t+} - \frac{1}{M}\mathcal{C}_{k_t}[\mathbf{g}_t]$
    Send $\mathbf{w}_{t+1}$ back to all workers
end for
```

The pseudo-code of distributed SGD with the proposed AdapTop-K method adding error compensation is provided in Algorithm 2.

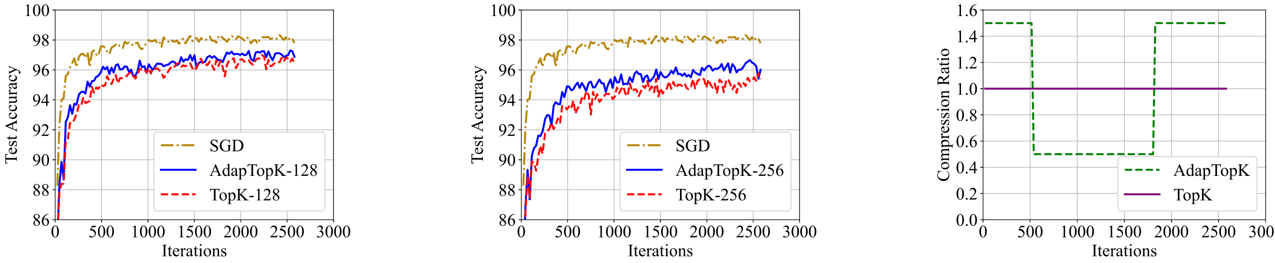
VI. Evaluation

In this section, we conduct experiments on two different tasks using several different kinds of datasets. In multi-robot collaboration, the vision ability is useful and important. Therefore, we choose the computer vision tasks in our evaluation to validate the effectiveness of our proposed AdapTop-K method. The first task is classic image classification based on widely used datasets, including MNIST, CIFAR-10, and CIFAR-100. The second task is object detection based on PASCAL VOC datasets because object detection is one of the most important tasks for robotics.

For classic image classification, we choose M = 8/16 workers and use canonical networks to evaluate the performance using different algorithms: the fully-connected network on the MNIST dataset, Resnet18 on the CIFAR-10 dataset

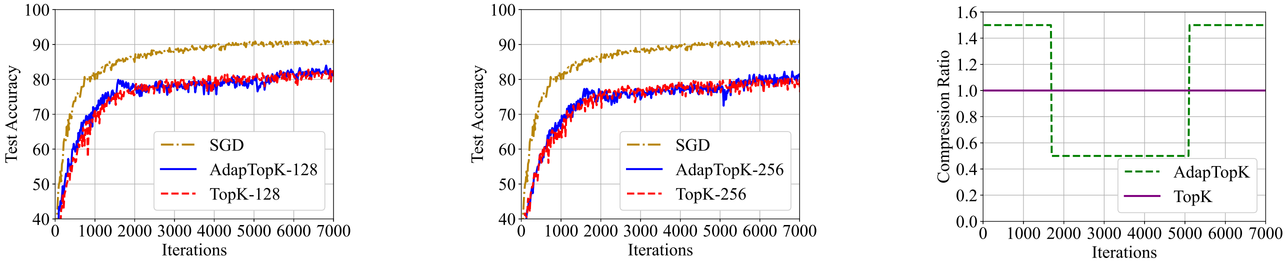
Dataset	MNIST	CIFAR-10	CIFAR-100	PASCAL VOC 2007+2012
Networks	fully-connected network	ResNet18	ResNet34	SSD [43] (based on VGG16)
Model Size	$d = 785$	$d = 1 \times 10^7$	$d = 3 \times 10^7$	$d = 2 \times 10^8$
Learning Rate	0.1	0.05	0.05	0.01
Batch Size	32	32	32	32
Workers	8	8	16	16
Iterations	3,000	7,000	7,000	24000
Compression Ratio	128/256/512	128/256/512	128/256/512	128/256/512
γ	0.5	0.5	0.5	0.5

TABLE I: Experimental Setting.



(a) Accuracy ($\frac{d}{k}$=128) (b) Accuracy ($\frac{d}{k}$=256) (c) Compression level on MNIST Dataset

Fig. 1: Evaluation results of different methods on MNIST Dataset.



(a) Accuracy ($\frac{d}{k}$=128) (b) Accuracy ($\frac{d}{k}$=256) (c) Compression level on CIFAR-10 Dataset

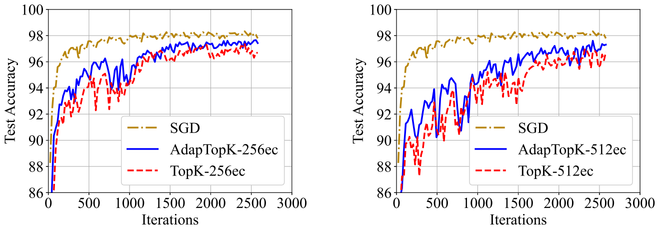
Fig. 2: Evaluation results of different methods on CIFAR-10.

and Resnet34 on the CIFAR-100 dataset. The above datasets are the databases commonly used for training various image processing systems. Other parameters information is shown in Table I. We use test accuracy to measure the learning performance. We compare our proposed AdapTop-K in SGD with the vanilla Top-K with or without error compensation.

Fig. 1 shows the comparison results of the classific Top-K algorithm and our proposed AdapTop-K on the MNIST dataset. Fig. 1a and Fig. 1b show the test accuracy curves and the training loss curves on the MNIST dataset. It shows how the model performance changes with iterations for several different values of the sparsification factor (128 or 256). The accuracy of the original distributed SGD reaches 98.02%. In Fig. 1a, the AdapTop-K achieves 97.03% accuracy which is better than 96.64% from Top-K. In Fig. 1b, the AdapTop-K achieves 96.21% accuracy which is higher than 95.41% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 128 and 256, respectively.

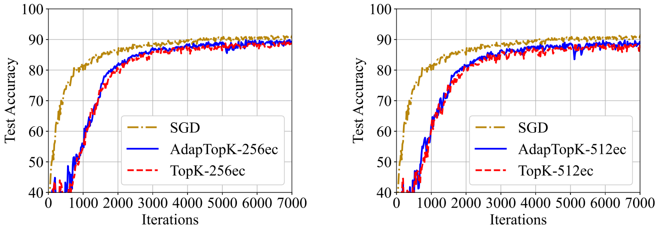
Similarly, Fig. 2 shows the comparison results of the fixed Top-K and our proposed AdapTop-K on CIFAR-10 dataset. Fig. 2a and Fig. 2b show the test accuracy curves and the training loss curves. It shows how the model performance changes with iterations for several different values of the sparsification factor (128 or 256). The accuracy of the original distributed SGD reaches 90.92%. In Fig. 2a, the AdapTop-K achieves 82.11% accuracy which is better than 81.36% from Top-K. In Fig. 2b, the AdapTop-K achieves 80.31% accuracy which is higher than 79.30% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 128 and 256, respectively. We keep the communication cost of the AdapTop-K stable compared with the classic Top-K in the total training process. It can be seen that our adaptive sparsification strategy can effectively improve the convergence rate and model performance with the pure Top-K algorithm. Fig. 1c and Fig. 2c both show the gradient sparsification level in the training process of AdapTop-K on different datasets. We can see that AdapTop-K significantly increases the bits assigned at the early stage and the late stage of training and improves the gradient accuracy as the training goes on.

After that, we add the error compensation [30] (abbreviated as ec) in Fig. 3 and Fig. 4 in our experiments, because it is a popular technique to improve the performance of distributed SGD with gradient compression. It shows how the model performance changes with iterations for several



(a) Accuracy with ec ($\frac{d}{k}$=256) (b) Accuracy with ec ($\frac{d}{k}$=512)

Fig. 3: Evaluation with error compensation on MNIST.

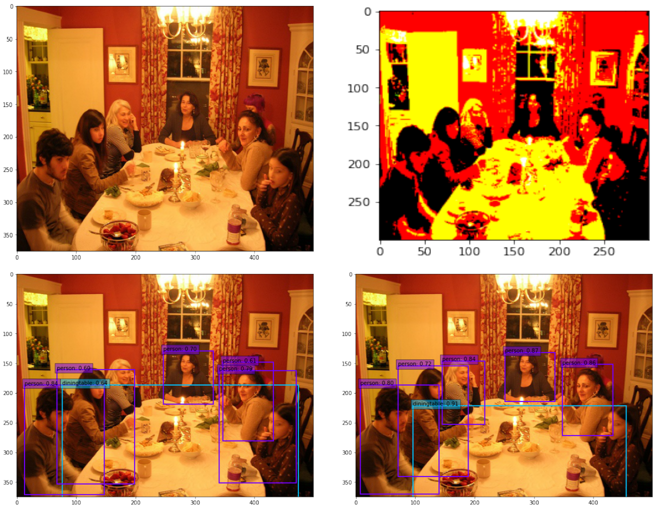


(a) Accuracy with ec ($\frac{d}{k}$=256) (b) Accuracy with ec ($\frac{d}{k}$=512)

Fig. 4: Evaluation with error compensation on CIFAR-10.

different values of the sparsification factor (256 or 512) when we add the error compensation. In these experiments, we use the bigger compression ratios (e.g., 256 and 512) because error compensation may reduce optimization errors in the training process to improve the total performance. Fig. 3 and Fig. 4 show the comparison results of the classific Top-K algorithm and our proposed AdapTop-K (all with error compensation) on MNIST and CIFAR-20 datasets. In Fig. 3a, the AdapTop-K achieves 97.50% accuracy which is higher than 96.71% from Top-K. In Fig. 3b, the AdapTop-K achieves 97.10% accuracy which is better than 96.24% from Top-K. In Fig. 4a, the AdapTop-K achieves 89.18% accuracy which is better than 88.66% from Top-K. In Fig. 4b, the AdapTop-K achieves 88.68% accuracy which is higher than 87.64% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 256 and 512, respectively. The results show that the AdapTop-K algorithm with error compensation achieves better performance under stable communication cost.

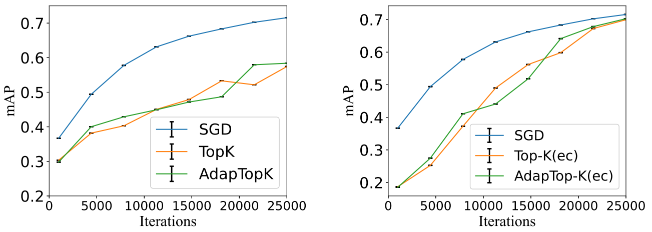
In multi-robot collaborative object detection, we choose M = 16 workers and use the SSD [43] framework of neural networks to evaluate the performance on PASCAL VOC datasets. For object detection task, the AP (Average precision) is a popular metric in measuring the accuracy of object detectors like SSD. Average precision computes the average precision value for recall value over 0 to 1. We always use mAP (mean average precision), which is averaged AP over all categories, to evaluate the model performance. In Fig. 6, we show the model performance for the concrete picture. At first, Fig. 6a shows the initial picture, and Fig. 6b shows the result after pre-processing. The other two pictures (Fig. 5c and Fig. 5d) show the model performance difference in the different training stages intuitively.



(c) After 12000 iterations (d) After 24000 iterations

Fig. 5: Example Picture in Total Training Process

Fig. 6 shows the comparison results of the classific Top-K algorithm and our proposed AdapTop-K using the SSD algorithm. Fig. 6a show the mAP curves on PASCAL VOC datasets. It shows how the model performance changes with iterations when the sparsification factor is 256. The mAP of the original distributed SGD reaches 0.715. In Fig. 1a, the AdapTop-K achieves 0.583 which is better than 0.574 from Top-K. In Fig. 1b, we add the error compensation and the AdapTop-K achieves 0.702 which is higher than 0.699 from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) is 256.



(a) mAP without ec (b) mAP with ec

Fig. 6: Evaluation for SSD when $\frac{d}{k}$=256.

Overall, the evaluation results demonstrate that the AdapTop-K outperforms the baselines.

VII. Conclusion

This paper proposes AdapTop-K, a novel adaptive gradient sparsification strategy in distributed SGD for multi-robot collaborative learning. The proposed method adjusts the sparsification levels adaptively by considering the gradient and the current iteration step. The experimental results for image classification show that AdapTop-K is superior to the state-of-the-art gradient compression methods in reducing communication cost. Our proposed method can significantly improve the communication efficiency in distributed robotics networks with strict proof.

APPENDIX A
PROOF IN SECTION IV

A. *Proof of Lemma 1*

Inspired by [28] and [27], we can get:

$$\|\mathbf{b}_t(\mathbf{w})\|^2 \leq (1-\frac{k}{d})\|\mathbf{g}_t\|^2.$$

B. *Proof for Lemma 2*

Using Eq. 10 and Assumption 1, we get:

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_{t+1})] &\leq F(\mathbf{w}_t) - \eta\langle\nabla F(\mathbf{w}_t), \mathcal{C}(\nabla F_\xi(\mathbf{w}_t))\rangle \\ &\quad + \frac{\eta^2 L}{2} E\|C(\nabla F_\xi(\mathbf{w}_t))\|^2 \\ \text{use} \quad \mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t))\|^2 &= \mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) \\ &\quad - [\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))]\|^2 \\ &\quad + \mathbb{E}\|\mathbb{E}[\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))]\|^2\end{aligned}$$

and Assumption 3, we get:

$$\begin{aligned}&\leq F(\mathbf{w}_t) - \eta\langle\nabla F(\mathbf{w}_t), \mathbb{E}[\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))]\rangle \\ &\quad + \frac{\eta^2 L}{2}(\sigma^2 + \mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))\|^2) \\ &\leq F(\mathbf{w}_t) + \frac{\eta}{2}\left(\mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))\|^2\right) + \frac{\eta^2 L}{2}\sigma^2 \\ &- \eta\left\langle\nabla F(\mathbf{w}_t), \mathbb{E}[\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - (\nabla F_\xi(\mathbf{w}_t) - \nabla F(\mathbf{w}_t))]\right\rangle \ (\eta \leq \frac{1}{L})\end{aligned}$$

from $\mathbb{E}\|\nabla F(\mathbf{w}_t) + \mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - \nabla F_\xi(\mathbf{w}_t)\|^2 = \mathbb{E}\|\nabla F(\mathbf{w}_t)\|^2$ $+\mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - \nabla F_\xi(\mathbf{w}_t)\|^2 + 2\mathbb{E}\langle\nabla F(\mathbf{w}_t), \mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - \nabla F_\xi(\mathbf{w}_t))\rangle$

$$\leq F(\mathbf{w}_t) + \frac{\eta}{2}(\mathbb{E}\|\mathcal{C}(\nabla F_\xi(\mathbf{w}_t)) - \nabla F_\xi(\mathbf{w}_t)\|^2 - \mathbb{E}\|\nabla F(\mathbf{w}_t)\|^2) + \frac{\eta^2 L}{2}\sigma^2 \tag{38}$$

$$\begin{aligned}\text{from} \quad Eq.\ (13) \quad \mathbb{E}\|\mathbf{b}_t(\mathbf{w})\|^2 &\leq \mathbb{E}[(1-\frac{k_t}{d})\|\nabla F_\xi(\mathbf{w}_t)\|^2] \\ &\leq \mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2 + \sigma^2 - \frac{k_t}{d}\|\nabla F_\xi(\mathbf{w}_t)\|^2] \\ \leq F(\mathbf{w}_t) - \frac{\eta k_t}{2d}\mathbb{E}\|\nabla F_\xi(\mathbf{w}_t)\|^2 &+ \frac{\eta}{2}\sigma^2 + \frac{\eta^2 L}{2}\sigma^2\end{aligned}$$

After the recursion: $\sum_{i=0}^{t}\mathbb{E}[\|\nabla F_\xi(\mathbf{w}_i)\|^2]$

$$\begin{aligned}&\leq \frac{2d}{k\eta}(F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_{t+1})]) + \frac{d\sigma^2(\eta L+1)(t+1)}{k} \\ \mathbb{E}[\|\nabla F_\xi(\mathbf{w}_t)\|^2] &\leq \frac{2d}{k}\cdot\frac{F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_{t+1})]}{\eta t} + \frac{d\sigma^2}{k}(\eta L+1) \\ &\leq \frac{2d}{k}\cdot\frac{F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_T)]}{\eta t} + \frac{d\sigma^2}{k}(\eta L+1)\end{aligned}$$

when $t \in [0, T-1]$, $\mathbb{E}[F(\mathbf{w}_T)] \leq \mathbb{E}[F(\mathbf{w}_{t+1})]$, then

We define $\mathbb{E}[\|\nabla F_\xi(\mathbf{w}_t)\|^2] \triangleq \frac{1}{t}\alpha + \beta$

$$\alpha = \frac{2d}{k}(F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_T)]), \beta = \frac{d\sigma^2}{k}(\eta L+1).$$

C. *Proof for Theorem 1*

Using Eq. (13), we assume that $k_t = k + n_t$, we have:

$$\begin{aligned}\mathbb{E}\|\mathbf{b}_t(\mathbf{w})\|^2 &\leq \mathbb{E}[(1-\frac{k_t}{d})\|\nabla F_\xi(\mathbf{w}_t)\|^2] \\ &\leq \mathbb{E}[(1-\frac{k}{d})\|\nabla F(\mathbf{w}_t)\|^2 + (1-\frac{k}{d})\sigma^2 - \frac{n_t}{d}\|\nabla F_\xi(\mathbf{w}_t)\|^2],\end{aligned}$$

then put this equation back to our above derivation Eq. (38):

$$\leq F(\mathbf{w}_t) - \frac{\eta k}{2d}\|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta}{2}(1-\frac{k}{d}+\eta L)\sigma^2 - \frac{\eta n_t}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2.$$

Therefore, we use Assumption 2 and get convergence rate as

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_{t+1})] - F^* &\leq (1-\frac{\eta k\mu}{d})(\mathbb{E}(F(\mathbf{w}_t) - F^*) \\ &\quad + \frac{\eta}{2}(1-\frac{k}{d}+\eta L)\sigma^2 - \frac{\eta n_t}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2.\end{aligned}$$

After recursion and simplification, we get:

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_T)] - F^* &\leq (1-\frac{\eta\mu}{d}k)^T[\mathbb{E}[F(\mathbf{w}_0)] - F^*] \\ &\quad + \frac{d}{2k\mu}(1-\frac{k}{d}+\eta L)\sigma^2[1-(1-\frac{\eta\mu}{d}k)^T] \\ &\quad - \sum_{t=0}^{T-1}[(\frac{\eta n_t}{2d}\|\nabla F_\xi(\mathbf{w}_t)\|^2)(1-\frac{\eta\mu}{d}k)^{T-1-t}].\end{aligned}$$

D. *Proof for Corollary 1*

According to Theorem 1, Eq. 20 and Eq. 21, we have:

$$\sum_{t=0}^{T-1}A_t B_t n_t = \gamma k(\sum_{t=\frac{\hat{t}}{2}}^{\frac{\hat{t}+T-1}{2}}A_t B_t - \sum_{t=0}^{\frac{\hat{t}}{2}}A_t B_t - \sum_{t=\frac{\hat{t}+T-1}{2}}^{T-1}A_t B_t) < 0$$

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* < \mathbf{M}(k) + \mathbf{N}(k)$$

APPENDIX B
PROOF IN SECTION V

A. *Proof for Lemma 4*

Using the update equation (24) we have

$$\begin{aligned}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2 &= \|\tilde{\mathbf{w}}_t - \mathbf{w}^\star\|^2 + \eta^2\|\nabla f_{i_t}(\mathbf{w}_t)\|^2 \\ &\quad - 2\eta\langle\mathbf{w}_t - \mathbf{w}^\star, \nabla f_{i_t}(\mathbf{w}_t)\rangle + 2\eta\langle\mathbf{w}_t - \tilde{\mathbf{w}}_t, \nabla f_{i_t}(\mathbf{w}_t)\rangle.\end{aligned}$$

And by applying expectation

$$\begin{aligned}\mathbb{E}_{i_t}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2 &\leq \|\tilde{\mathbf{w}}_t - \mathbf{w}^\star\|^2 + \eta_t^2 G^2 \\ &\quad - 2\eta_t\langle\mathbf{w}_t - \mathbf{w}^\star, \nabla f(\mathbf{w}_t)\rangle + 2\eta_t\langle\mathbf{w}_t - \tilde{\mathbf{w}}_t, \nabla f(\mathbf{w}_t)\rangle.\end{aligned}$$

To upper bound the third term, we use the same estimates as in [39, Appendix C.3], and the strong convexity Assumption 5, hence

$$-\langle\mathbf{w}_t - \mathbf{w}^\star, \nabla F(\mathbf{w}_t)\rangle \leq -(F(\mathbf{w}_t) - F^\star) - \frac{\mu}{2}\|\mathbf{w}_t - \mathbf{w}^\star\|^2$$

and with $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ we further have

$$-\|\mathbf{w}_t - \mathbf{w}^\star\|^2 \leq \|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2 - \frac{1}{2}\|\tilde{\mathbf{w}}_t - \mathbf{w}^\star\|^2 .$$

Putting these two estimates together, we can get upper bound as follows:

$$\begin{aligned}\mathbb{E}_{i_t}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2 &\leq \left(1 - \frac{\eta_t\mu}{2}\right)\|\tilde{\mathbf{w}}_t - \mathbf{w}^\star\|^2 + \eta_t^2 G^2 \\ -2\eta_t e_t + \eta_t\mu\|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2 &+ 2\eta_t\langle\mathbf{w}_t - \tilde{\mathbf{w}}_t, \nabla F(\mathbf{w}_t)\rangle ,\end{aligned} \tag{39}$$

where $e_t = \mathbb{E}[F(\mathbf{w}_t)] - F^\star$. We now estimate the last term. As each F_i is L-smooth satisfies Assumption 1. Together with $2\langle a, b\rangle \leq \gamma \|a\|^2 + \gamma^{-1}\|b\|^2$ we have

$$\begin{aligned}\langle \mathbf{w}_t - \tilde{\mathbf{w}}_t, \nabla F(\mathbf{w}_t)\rangle &\leq \frac{1}{2}\left(2L\|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2 + \frac{1}{2L}\|\nabla F(\mathbf{w}_t)\|^2\right) \\ &= L\|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2 + \frac{1}{4L}\|\nabla F(\mathbf{w}_t) - \nabla F(\mathbf{w}^\star)\|^2 \\ &\leq L\|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2 + \frac{1}{2}\left(F(\mathbf{w}_t) - F^\star\right).\end{aligned}$$

Combining with (39) we have

$$\begin{aligned}\mathbb{E}_{i_t}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^\star\|^2 \leq \left(1 - \frac{\eta_t \mu}{2}\right)\|\tilde{\mathbf{w}}_t - \mathbf{w}^\star\|^2 + \eta_t^2 G^2 \\ - \eta_t e_t + \eta_t(\mu + 2L)\|\mathbf{w}_t - \tilde{\mathbf{w}}_t\|^2,\end{aligned}$$

and the claim follows with (26).

B. Proof for Lemma 5

According to Lemma 4, we can rewrite it as:

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_t)] - F^* &\leq (\frac{1}{\eta} - \frac{\mu}{2})\mathbb{E}\|\tilde{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta G^2 \\ &\quad - \mathbb{E}\|\tilde{\mathbf{w}}_{t+1} - \mathbf{w}^*\|^2 + (\mu + 2L)\mathbb{E}\|\mathbf{m}_t\|^2 \\ &\leq (\frac{1}{\eta} - \frac{\mu}{2})\mathbb{E}\|\tilde{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta G^2 + (\mu + 2L)\mathbb{E}\|\mathbf{m}_t\|^2\end{aligned}$$

We can also do recursion for Lemma 4 and get:

$$\begin{aligned}\mathbb{E}\|\tilde{\mathbf{w}}_t - \mathbf{w}^*\|^2 &\leq (1 - \frac{\mu\eta}{2})^t \mathbb{E}\|\tilde{\mathbf{w}}_0 - \mathbf{w}^*\|^2 \\ &\quad + \sum_{i=0}^{t-1}(1 - \frac{\mu\eta}{2})^{t-i}(\eta^2 G^2 + \eta(\mu + 2L)\mathbb{E}\|\mathbf{m}_i\|^2).\end{aligned}$$

Combine the above two equations, we could get the Eq. (27):

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_T)] - F^* &\leq (1 - \frac{\mu\eta}{2})^T(\frac{1}{\eta} - \frac{\mu}{2})\mathbb{E}\|\mathbf{w}_0 - \mathbf{w}^*\|^2 + \eta G^2 \\ &\quad + (\mu + 2L)\mathbb{E}\|\mathbf{m}_T\|^2 + (\frac{1}{\eta} - \frac{\mu}{2})(\frac{2}{\mu\eta} - 1)[1 - (1 - \frac{\eta\mu}{2})^T]\eta^2 G^2 \\ &\quad + (1 - \frac{\mu\eta}{2})(\mu + 2L)\sum_{t=0}^{T-1}(1 - \frac{\mu\eta}{2})^{T-t}\mathbb{E}\|\mathbf{m}_t\|^2\end{aligned}$$

C. Proof for Lemma 6

It's similiar with the proof of Lemma 1, inspired by [28], we can get:

$$\begin{aligned}\|\mathbf{m}_{t+1}(\mathbf{w})\|^2 &= \|\mathbf{g}_t - \mathcal{C}(\mathbf{g}_t)\|^2 \\ &\leq (1 - \frac{k_t}{d})\|\mathbf{g}_t\|^2 \\ &\leq (1 - \frac{k}{d})\|\mathbf{g}_t\|^2 - \frac{n_t}{d}\|\mathbf{g}_t\|^2.\end{aligned}$$

D. Proof for Theorem 2

After get the Lemma 5 and Lemma 6, we just combine them together to get the convergence rate. The exact form of result is too complex, please see Eq. 29.

E. Proof for Lemma 7

Firstly, the sparsification operation ($\mathcal{C}(\cdot)$) has the property as:

$$\begin{cases}\mathcal{C}(\hat{\mathbf{g}}_t) = \frac{1}{\eta}\mathcal{C}_k(\mathbf{g}_t), \\ \hat{\mathbf{g}_t} = \frac{1}{\eta}\mathbf{m}_t + \nabla F_{i,\xi}(\mathbf{w}_t) = \frac{1}{\eta}\mathbf{g}_t.\end{cases}$$

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_{t+1})] &\leq F(\mathbf{w}_t) - \langle \nabla F_\xi(\mathbf{w}_t), \mathcal{C}(\mathbf{g}_t)\rangle + \frac{L}{2}\mathbb{E}\|\mathcal{C}(\mathbf{g}_t)\|^2 \\ &\leq F(\mathbf{w}_t) - \eta\langle \nabla F_\xi(\mathbf{w}_t), \mathcal{C}(\hat{\mathbf{g}}_t)\rangle + \frac{\eta^2 L}{2}\mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t)\|^2\end{aligned}$$

use $\mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t)\|^2 \leq \sigma^2 + \mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - (\hat{\mathbf{g}}_t - (\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)))\|^2$,

$$\begin{aligned}\text{we get: } &\leq F(\mathbf{w}_t) - \eta\langle \nabla F_\xi(\mathbf{w}_t), \mathcal{C}(\hat{\mathbf{g}}_t)\rangle \\ &\quad + \frac{\eta^2 L}{2}(\sigma^2 + \mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - (\hat{\mathbf{g}}_t - (\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)))\|^2)\end{aligned}$$

from $\mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - (\hat{\mathbf{g}}_t - (\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)))\|^2 \leq \mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - \hat{\mathbf{g}}_t\|^2$ $-\mathbb{E}\|\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)\| + 2\mathbb{E}\langle \nabla F_\xi(\mathbf{w}_t), \mathcal{C}(\hat{\mathbf{g}_t})\rangle$, we get:

$$\leq F(\mathbf{w}_t) + \frac{\eta}{2}(\mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - \hat{\mathbf{g}}_t\|^2 - \mathbb{E}\|\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)\| + \frac{\eta^2 L}{2}\sigma^2 (\eta \leq \frac{1}{L})$$

$$\begin{aligned}\text{from} \quad Eq.\ (28) \quad \mathbb{E}\|\mathcal{C}(\hat{\mathbf{g}}_t) - \hat{\mathbf{g}}_t\|^2 &\leq (1 - \frac{k}{d})\|\hat{\mathbf{g}}_t\|^2 \\ &\leq \mathbb{E}\|\frac{\mathbf{m}_t}{\eta} + \nabla F_\xi(\mathbf{w}_t)\| + \sigma^2 - \frac{k_t}{d}\mathbb{E}\|\hat{\mathbf{g}}_t\|^2 \\ \leq F(\mathbf{w}_t) - \frac{\eta k_t}{2d}\mathbb{E}\|\hat{\mathbf{g}}_t\|^2 + \frac{\eta}{2}\sigma^2 + \frac{\eta^2 L}{2}\sigma^2\end{aligned}$$

After the recursion: $\sum_{i=0}^{t}\mathbb{E}[\|\hat{\mathbf{g}}_i\|^2]$

$$\begin{aligned}&\leq \frac{2d}{k\eta}(F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_{t+1})]) + \frac{d\sigma^2(\eta L + 1)(t+1)}{k} \\ \mathbb{E}[\|\hat{\mathbf{g}}_t\|^2] &\leq \frac{2d}{k}\cdot\frac{F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_{t+1})]}{\eta t} + \frac{d\sigma^2}{k}(\eta L + 1) \\ &\leq \frac{2d}{k}\cdot\frac{F(\mathbf{w}_0) - \mathbb{E}[F(\mathbf{w}_T)]}{\eta t} + \frac{d\sigma^2}{k}(\eta L + 1)\end{aligned}$$

when $t \in [0, T-1]$, $\mathbb{E}[F(\mathbf{w}_T)] \leq \mathbb{E}[F(\mathbf{w}_{t+1})]$, then

We define $\mathbb{E}[\|\mathbf{g}_t\|^2] = \eta^2\mathbb{E}[\|\hat{\mathbf{g}}_t\|^2]$

$$\begin{aligned}&\leq \frac{2d\eta}{k}\cdot\frac{F(\mathbf{w}_0)}{t} + \frac{d\sigma^2\eta^2}{k}(\eta L + 1) \triangleq \frac{\alpha}{t} + \beta \\ &\alpha \triangleq \frac{2d\eta}{k}F(\mathbf{w}_0), \beta \triangleq \frac{d\sigma^2\eta^2}{k}(\eta L + 1)\end{aligned}$$

References

[1] M. Ruan, G. Yan, Y. Xiao, L. Song, and W. Xu, "Adaptive top-k in sgd for communication-efficient distributed learning," in *2023 IEEE Global Communications Conference: Communication Theory (Globecom 2023)*.

[2] J. Yu, J. A. Vincent, and M. Schwager, "Dinno: Distributed neural network optimization for multi-robot collaborative learning," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1896–1903, 2022.

[3] S. H. Alsamhi, O. Ma, and M. S. Ansari, "Survey on artificial intelligence based techniques for emerging robotic communication," *Telecommunication Systems*, vol. 72, pp. 483–503, 2019.

[4] L. Qian, P. Yang, M. Xiao, O. A. Dobre, M. Di Renzo, J. Li, Z. Han, Q. Yi, and J. Zhao, "Distributed learning for wireless communications: Methods, applications and challenges," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 3, pp. 326–342, 2022.

[5] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang *et al.*, "Large scale distributed deep networks," in *in Proceedings of Conference in Neural In- formation Processing Systems (NeurIPS)*, 2012.

[6] R. Bekkerman, M. Bilenko, and J. Langford, *Scaling up machine learning: Parallel and distributed approaches*. Cambridge University Press, 2011.

[7] X. Cao, T. Bacsar, S. Diggavi, Y. C. Eldar, K. B. Letaief, H. V. Poor, and J. Zhang, "Communication-efficient distributed learning: An overview," *IEEE journal on selected areas in communications*, 2023.

[8] W. Xu, Z. Yang, D. W. K. Ng, M. Levorato, Y. C. Eldar, and M. Debbah, "Edge learning for b5g networks with distributed signal processing: Semantic communication, edge computing, and wireless sensing," *IEEE journal of selected topics in signal processing*, vol. 17, no. 1, pp. 9–39, 2023.

[9] Y. Chen, R. S. Blum, M. Takáč, and B. M. Sadler, "Distributed learning with sparsified gradient differences," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 3, pp. 585–600, 2022.

[10] C. Xie, S. Zheng, O. Koyejo, I. Gupta, M. Li, and H. Lin, "Cser: communication-efficient sgd with error reset," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 12 593–12 603.

[11] C.-Y. Chen, J. Ni, S. Lu, X. Cui, P.-Y. Chen, X. Sun, N. Wang, S. Venkataramani, V. Srinivasan, W. Zhang *et al.*, "Scalecom: scalable sparsified gradient compression for communication-efficient distributed training," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, pp. 13 551–13 563.

[12] K. Mishchenko, B. Wang, D. Kovalev, and P. Richtárik, "Intsgd: Adaptive floatless compression of stochastic gradients," in *International Conference on Learning Representations*, 2021.

[13] F. Faghri, I. Tabrizian, I. Markov, D. Alistarh, D. M. Roy, and A. Ramezani-Kebrya, "Adaptive gradient quantization for data-parallel sgd," *Advances in neural information processing systems*, vol. 33, pp. 3174–3185, 2020.

[14] Z. Zhang and C. Wang, "Mipd: An adaptive gradient sparsification framework for distributed dnns training," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 3053–3066, 2022.

[15] C.-Y. Chen, J. Choi, D. Brand, A. Agrawal, W. Zhang, and K. Gopalakrishnan, "Adacomp: Adaptive residual gradient compression for data-parallel distributed training," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[16] T. Vogels, S. P. Karimireddy, and M. Jaggi, "Powersgd: Practical low-rank gradient compression for distributed optimization," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[17] E. Ozfatura, K. Ozfatura, and D. Gündüz, "Time-correlated sparsification for communication-efficient federated learning," in *2021 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2021, pp. 461–466.

[18] A. T. Suresh, X. Y. Felix, S. Kumar, and H. B. McMahan, "Distributed mean estimation with limited communication," in *International conference on machine learning*. PMLR, 2017, pp. 3329–3337.

[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 *The 15th Annual Conference of the International Speech Communication Association*, 2014.

[20] S. Agarwal, H. Wang, K. Lee, S. Venkataraman, and D. Papailiopoulos, "Adaptive gradient communication via critical learning regime identification," *Proceedings of Machine Learning and Systems*, vol. 3, pp. 55–80, 2021.

[21] A. Øland and B. Raj, "Reducing communication overhead in distributed learning by an order of magnitude (almost)," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 2219–2223.

[22] G. Cui, J. Xu, W. Zeng, Y. Lan, J. Guo, and X. Cheng, "Mqgrad: Reinforcement learning of gradient quantization in parameter server," in *Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval*, 2018, pp. 83–90.

[23] Y. Lin, S. Han, H. Mao, Y. Wang, and B. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *International Conference on Learning Representations*, 2018.

[24] D. Basu, D. Data, C. Karakus, and S. N. Diggavi, "Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 217–226, 2020.

[25] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 40, no. 1, pp. 342–358, 2021.

[26] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the Convergence of FedAvg on Non-IID Data," in *Proceedings of International Conference on Learning Representations (ICLR)*, 2020.

[27] A. Ajalloeian and S. U. Stich, "On the convergence of SGD with biased gradients," in *Proceedings of Workshop in International Conference on Machine Learning (ICML)*, 2020.

[28] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[29] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding," in *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[30] J. Wu, W. Huang, J. Huang, and T. Zhang, "Error compensated quantized SGD and its applications to large-scale distributed optimization," in *Proceedings of International Conference on Machine Learning (ICML)*. PMLR, 2018, pp. 5325–5333.

[31] J. Wangni, J. Wang, J. Liu, and T. Zhang, "Gradient sparsification for communication-efficient distributed optimization," in *Proceedings of Conference in Neural Information Processing Systems (NeurIPS)*, 2018.

[32] G. Yan, T. Li, S.-L. Huang, T. Lan, and L. Song, "AC-SGD: Adaptively Compressed SGD for Communication-Efficient Distributed Learning," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 40, no. 9, pp. 2678–2693, 2022.

[33] H. Karimi, J. Nutini, and M. Schmidt, "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Lojasiewicz Condition," in *Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2016, pp. 795–811.

[34] J. Guo, W. Liu, W. Wang, J. Han, R. Li, Y. Lu, and S. Hu, "Accelerating distributed deep learning by adaptive gradient quantization," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 1603–1607.

[35] P. Han, S. Wang, and K. K. Leung, "Adaptive gradient sparsification for efficient federated learning: An online learning approach," in *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2020, pp. 300–310.

[36] H. Zhang, J. Li, K. Kara, D. Alistarh, J. Liu, and C. Zhang, "ZipML: Training linear models with end-to-end low precision, and a little bit of deep learning," in *Proceedings of International Conference on Machine Learning (ICML)*. PMLR, 2017, pp. 4035–4043.

[37] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li, "Terngrad: Ternary gradients to reduce communication in distributed deep learning," in *Proceedings of Conference in Neural Information Processing Systems (NeurIPS)*, 2017.

[38] K. Mishchenko, E. Gorbunov, M. Takáč, and P. Richtárik, "Distributed learning with compressed gradient differences," *arXiv preprint arXiv:1901.09269*, 2019.

[39] D. P. B. R. K. R. Horia Mania, Xinghao Pan and M. I. Jordan, "Perturbed iterate analysis for asynchronous stochastic optimization," *SIAM Journal on Optimization, 27(4):2202–2229, 2017.*, 2017.

[40] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[41] S. U. Stich, "Local sgd converges fast and communicates little," in *International Conference on Learning Representations*, 2018.

[42] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi, "Don't use large mini-batches, use local sgd," in *International Conference on Learning Representations*, 2019.

[43] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

Adaptive Top-K in SGD for Communication-Efficient Distributed Learning

Mengzhe Ruan[1,2] Guangfeng Yan[1,2] Yuanzhang Xiao[3] Linqi Song[1,2] Weitao Xu[1,2,*]
[1] City University of Hong Kong Shenzhen Research Institute, Shenzhen, China
[2] Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China
[3]Hawaii Advanced Wireless Technologies Institute, University of Hawaii at Manoa, Honolulu, HI, USA

Abstract—**Distributed stochastic gradient descent (SGD) with gradient compression has become a popular communication-efficient solution for accelerating distributed learning. One commonly used method for gradient compression is Top-K sparsification, which sparsifies the gradients by a fixed degree during model training. However, there has been a lack of an adaptive approach to adjust the sparsification degree to maximize the potential of the model's performance or training speed. This paper proposes a novel adaptive Top-K in SGD framework that enables an adaptive degree of sparsification for each gradient descent step to optimize the convergence performance by balancing the trade-off between communication cost and convergence error. Firstly, an upper bound of convergence error is derived for the adaptive sparsification scheme and the loss function. Secondly, an algorithm is designed to minimize the convergence error under the communication cost constraints. Finally, numerical results on the MNIST and CIFAR-10 datasets demonstrate that the proposed adaptive Top-K algorithm in SGD achieves a significantly better convergence rate compared to state-of-the-art methods, even after considering error compensation.**

I. Introduction

Nowadays, with extensive data collected in distributed networks, there is an increasing need for distributed learning algorithms that aggregate local gradients to learn a global models. Distributed stochastic gradient descent (SGD) is the core of most distributed learning algorithms [1]. In practical networks, however, the communication overhead of transmitting gradients often becomes the performance bottleneck due to the limited bandwidth. Gradient compression, which uses less information to represent the gradients, is an effective and efficient method to solve this problem. The compression methods, however, inevitably introduce compression noise which affects the convergence of the model. Therefore, how to choose the compression methods and the compression level efficiently to balance the trade-off between communication cost and convergence performance remains an open challenge.

Traditional compression methods often compress parameters with a *fixed* compression factor for all the training iterations, which may not be optimal. To further improve communication efficiency, an online learning method was proposed in [11] to adaptively adjust the degree of gradient sparsity when the total dataset is non-i.i.d distributed in the federated learning network. Unfortunately, there lacks a theoretical convergence analysis in their research. In [9], an adaptive quantization method is proposed and its theoretical guarantee has also been proved. Nevertheless, the quantization method needs more computing resources than sparsification methods, which simply keep some components of the gradient and set others to zero. Therefore, we would like to investigate the adaptive sparsification methods in distributed SGD. We will improve upon Top-K, the most commonly-used biased sparsification method, which keeps only a few coordinates of the stochastic gradient with the largest magnitudes.

In this paper, we propose a novel adaptive Top-K SGD framework, named AdapTop-K, that aims to improve the convergence performance of Top-K while maintaining the same communication cost. Under the assumption of smoothness and Polyak-Lojasiewicz condition [10], we derive an upper bound on the gap between the loss function and the optimal loss to characterize the convergence error caused by limited iteration steps, sampling, and adaptive Top-K sparsification. Based on the theoretical analysis, we design an adaptive Top-K method by minimizing the convergence upper bound under the desired total communication cost. The proposed AdapTop-K algorithm adjusts the degree of sparsification by considering the desired model performance, the number of rounds, and the norm of gradients. We validate our theoretical analysis through experiments on image classification tasks on the MNIST and CIFAR-10 datasets. Numerical results show that AdapTop-K outperforms the baseline sparsification methods.

To summarize, our key contributions are as follows:

- We propose a novel framework to characterize the trade-off between the communication cost and the convergence rate by adaptively adjusting the gradient sparsification levels in distributed learning. We analyze the convergence error of the loss function under Top-K sparsification for gradients over different communication rounds. We isolate our bound on the convergence error to characterize the impact of adaptive sparsification on the convergence rate.
- We solve the optimization problem that minimizes the convergence error while keeping the same communication cost as Top-K. To achieve this, we propose a novel adaptive Top-K algorithm called AdapTop-K, which dynamically adjusts the degree of gradient sparsification during training to improve model performance.
- We validate the proposed AdapTop-K on the popular datasets and machine learning models, demonstrating that our proposed AdapTop-K outperforms state-of-the-art gradient

Weitao Xu is the corresponding author.

sparsification methods.

II. Related Work

There are two main approaches to compress SGD to reduce communication cost: quantization and sparsification. Quantization compresses gradients by limiting the number of bits representing floating point numbers during communication. The gradient quantization was proposed in [6]. There are several variants of quantization, including error compensation [7], variance-reduced quantization [12], quantization to a ternary vector [13], and quantization of gradient difference [14]. Sparsification methods aim to reduce the number of non-zero entries in the stochastic gradients [8]. An aggressive sparsification method (Top-K) [5] is to keep very few coordinates of the stochastic gradient with the largest magnitudes. The methods can also be classified based on whether the compression is biased or unbiased. The unbiased methods could keep the expectation of compressed gradients as that of the true gradients [6] and [13]. In contrast, the biased methods introduce bias in the compression and more compression noise to the optimization process [5]. These methods can compress the gradient efficiently to speed up distributed training. However, they do not consider adaptively changing the degree of compression during training, which is the key difference between our method and existing methods.

III. System Model

We consider a distributed learning system with a central server and M edge devices (workers). The workers collaborate to train a shared machine learning model by aggregating the gradient or its variant in cooperation with the central server.

The learning model is represented by the vector of its parameters $\mathbf{w} \in \mathbb{R}^d$, where d is the model size. The datasets are distributed over the M workers. We use $\mathcal{D}^i$ to denote the local dataset at worker i. The global loss function, denoted by $F : \mathbb{R}^d \to \mathbb{R}$, is defined as

$$F(\mathbf{w}) = \frac{1}{M}\sum_{i=1}^{M} f^i(\mathbf{w}), \qquad \text{with } f^i(\mathbf{w}) = \mathbb{E}_{\xi^i \sim \mathcal{D}^i}\left[l^i(\mathbf{w};\xi^i)\right], \tag{1}$$

where $l^i(\mathbf{w};\xi^i)$ is the local loss function of the model parameters $\mathbf{w}$ at work i, given the mini-batch ξ^i randomly selected from worker i's local dataset $\mathcal{D}^i$.

The objective of the training is to find a model parameter $\mathbf{w}$ to minimize the global loss function in Eq. (1) :

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} F(\mathbf{w}). \tag{2}$$

The distributed SGD is the most popular method to solve this problem, where each worker i computes its local stochastic gradient $\mathbf{g}_t^i = \nabla l^i(\mathbf{w}_t;\xi^i)$ given parameters $\mathbf{w}_t$ at round t. Then the workers send the local gradient $\mathbf{g}_t^i$ to the central server. The server aggregates these gradients to update the model. To reduce the communication cost, we compress the local stochastic gradients before sending them to the server:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{M}\sum_{i=1}^{M} \mathcal{C}^i[\mathbf{g}_t^i], \tag{3}$$

where η_t is the learning rate at iteration t, and $\mathcal{C}^i[\cdot]$ is the compression operator. Without the gradient compressor, Eq. (3) reduces to the vanilla distributed SGD with $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{M}\sum_{i=1}^{M}\mathbf{g}_t^i$. The same procedure is repeated until the convergence criterion or the maximum number of communication rounds is reached.

A commonly-used compression operator is Top-K, where each worker i keeps only k elements of the gradient $\mathbf{g}_t^i$ with the largest magnitudes and sets the other elements to zero [5]. In this work, we speed up the convergence of Top-K by adaptively choosing the sparsity of the gradient during the convergence process. Specifically, given a total of T rounds of gradient update, our goal is to find the optimal sparsity levels $k_0, \ldots, k_{T-1}$ in each round, so that the final model is as close to the optimal model as possible. It is natural to measure the gap from the optimal model by the difference between the expectation of the final global loss $F(\mathbf{w}_T)$ and the optimal loss $F^* = F(\mathbf{w}^*)$. Note that we need to take expectation of the final loss $F(\mathbf{w}_T)$ due to the stochastic gradient descent. Therefore, our design problem can be formulated as follows

$$\begin{aligned} \min_{k_0,\ldots,k_{T-1}} \quad & \mathbb{E}\left[F(\mathbf{w}_T)\right] - F^* \\ \text{s.t.} \quad & \textstyle\sum_{t=0}^{T-1} k_t \le K, \\ & k_t \in \{0,1,\ldots,d\},\ t = 0,\ldots,T-1, \end{aligned} \tag{4}$$

where K is the total budget for the communication overhead during the training. When comparing with other sparsification methods, we can set the communication budget K accordingly.

IV. Proposed Algorithm

In this section, we first provide convergence analysis of AdapTop-K given a sequence of sparsity levels $k_0, \ldots, k_{T-1}$. Based on the analysis, we then propose a practical algorithm for finding a sequence $k_0, \ldots, k_{T-1}$ that guarantees to outperform the standard Top-K method.

A. Convergence Analysis

For the convergence analysis, we make standard assumptions on the stochastic gradient and the loss function that are commonly used in the literature [9], [3], and [2].

Assumption 1: (Smoothness). There exists a non-negative constant L such that for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$,

$$F(\mathbf{x}) - F(\mathbf{y}) - \langle \nabla F(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle \le \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \tag{5}$$

where $\nabla F(\mathbf{y})$ is the gradient of the loss function $F(\cdot)$ at $\mathbf{y}$.

Assumption 2: (Polyak-Lojasiewicz Condition). There exists a constant $\mu \ge 0$ such that for any $\mathbf{w} \in \mathbb{R}^d$, we have

$$\|\nabla F(\mathbf{w})\|^2 \ge 2\mu(F(\mathbf{w}) - F^*). \tag{6}$$

Note that Assumption 2 is milder than the assumption of strong convexity [10].

Assumption 3: (Unbiasedness and Bounded Variance of Stochastic Gradient). The local stochastic gradients $\mathbf{g}^i$ are assumed to be independent and unbiased estimates of the local gradient $\nabla f^i(\mathbf{w}_t)$ with bounded variance:

$$\begin{aligned} \mathbb{E}_{\xi^i \sim \mathcal{D}^i}\left[\mathbf{g}_t^i\right] &= \nabla f^i(\mathbf{w}_t), \qquad (7)\\ \mathbb{E}_{\xi^i \sim \mathcal{D}^i}\left[\|\mathbf{g}_t^i - \nabla f^i(\mathbf{w}_t)\|^2\right] &\leq \sigma^2. \end{aligned}$$

As proven in [4], the gradient update in Eq. (3) can be rewritten as

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathcal{C}[\mathbf{g}_t], \tag{8}$$

where $\mathbf{g}_t$ is the stochastic gradient of the global loss function

$$\mathbf{g}_t = \nabla F(\mathbf{w}_t) + \mathbf{m}_t(\mathbf{w}_t), \tag{9}$$

and $\mathcal{C}[\cdot]$ is the aggregate Top-K operator

$$\mathcal{C}[\mathbf{g}_t] = \nabla F(\mathbf{w}_t) + \mathbf{m}_t(\mathbf{w}_t) + \mathbf{b}_t(\mathbf{w}_t), \tag{10}$$

where $\mathbf{m}_t(\mathbf{w}_t)$ is the noise in SGD and $\mathbf{b}_t(\mathbf{w}_t)$ is the bias introduced by sparsification.

By Assumption 3, the noise has zero mean and bounded variance, namely

$$\mathbb{E}[\mathbf{m}_t(\mathbf{w}_t)] = 0 \quad \text{and} \quad \mathbb{E}[\|\mathbf{m}_t(\mathbf{w}_t)\|^2] \leq \sigma^2. \tag{11}$$

An upper bound of the variance of the bias is given in [5]. We summarize the result as a lemma here.

Lemma 1: (Bounded Variance of Stochastic Gradient with Top-K sparsification). The variance of the bias $\mathbf{b}_t(\mathbf{w}_t)$ is upper bounded by the mini-batch gradient $\mathbf{g}_t$ as follows: [5]

$$\|\mathbf{b}_t(\mathbf{w}_t)\|^2 \leq \left(1 - \frac{k}{d}\right)\|\mathbf{g}_t\|^2. \tag{12}$$

With Lemma 1, we prove an upper bound of the optimality gap under the adaptive sparsity levels of $k_0, \ldots, k_{T-1}$.

Theorem 1: (Upper Bound for Convergence Error). Under Assumptions 1–3, given the initial parameter $\mathbf{w}_0$ and constant stepsize $\eta_t = \eta \leq \frac{1}{L}$, the optimality gap of the adaptive Top-K method is upper bounded by

$$\begin{aligned} \mathbb{E}[F(\mathbf{w}_T)] - F^* \leq\ & \underbrace{\left(1 - \frac{\eta\mu}{d}k\right)^T [F(\mathbf{w}_0) - F^*]}_{\mathbf{M}(k)} \qquad (13)\\ &+ \underbrace{\frac{d\sigma^2}{2k\mu}\left(1 - \frac{k}{d} + \eta L\right)\left[1 - \left(1 - \frac{\eta\mu}{d}k\right)^T\right]}_{\mathbf{N}(k)}\\ &- \underbrace{\sum_{t=0}^{T-1}\left[\left(\frac{\eta n_t}{2d}\|\mathbf{g}_t\|^2\right)\left(1 - \frac{\eta\mu}{d}k\right)^{T-1-t}\right]}_{\text{the only term that depends on } n_t}, \end{aligned}$$

where $k = \frac{K}{T}$ is the average sparsity level and $n_t = k_t - k$ is the deviation from the average sparsity level at round t.

Proof: See the appendix. The proofs of all the other results can be found in our technical report [15]. ■

The upper bound in (13) has two parts. The first part is the sum of the first two terms $\mathbf{M}(k) + \mathbf{N}(k)$, which depends only on the average sparsity level k. The second part is the third term, which is the only term that depends on $n_0, \ldots, n_{T-1}$. When $n_t = 0$ for all t, the upper bound reduces to $\mathbf{M}(k) + \mathbf{N}(k)$, namely the bound for the vanilla Top-K method.

B. The Proposed AdapTop-K Algorithm

We aim to minimize the upper bound of the optimality gap in (13) by choosing $n_0, \ldots, n_{T-1}$. Since only the third term depends on the adjustments $n_0, \ldots, n_{T-1}$, the optimization problem can be formulated as

$$\begin{aligned} \max_{n_0, \ldots, n_{T-1}} \quad & \sum_{t=0}^{T-1}\left[\left(\frac{\eta n_t}{2d}\|\mathbf{g}_t\|^2\right)\left(1 - \frac{\eta\mu}{d}k\right)^{T-1-t}\right] \qquad (14)\\ \text{s.t.} \quad & \sum_{t=0}^{T-1} n_t \leq 0,\\ & n_t \in \{-k, \ldots, d-k\}\,,\ t = 0, \ldots, T-1, \end{aligned}$$

where the first constraint comes from the constraint on the communication overhead in (4) and the second constraint comes from the fact that $k_t \in \{0, \ldots, d\}$.

Since the objective function is linear in n_t, the optimal solution should assign the largest possible values to the n_t's with the largest coefficients

$$\left(\frac{\eta}{2d}\|\mathbf{g}_t\|^2\right)\left(1 - \frac{\eta\mu}{d}k\right)^{T-1-t}, \tag{15}$$

subject to the upper bound $d - k$ and the budget of total communication overhead. However, the major challenge is that the coefficients in (15) depend on the gradients $\mathbf{g}_t$, which are stochastic due to the randomly selected mini-batches and are dependent on our choice of sparsity levels $n_0, \ldots, n_{t-1}$ up to round t. Therefore, we cannot solve the optimization problem (14) directly. Instead, we choose to maximize an upper bound of the objective function, which is obtained by bounding the norm of the stochastic gradients $\mathbf{g}_t$.

Lemma 2: (Upper Bound for Stochastic Gradient). Under Assumptions 1–3, given the initial parameter $\mathbf{w}_0$ and constant stepsize $\eta_t = \eta \leq \frac{1}{L}$, the stochastic gradient in Eq. (9) can be upper bounded by

$$\mathbb{E}[\|\mathbf{g}_t\|^2] \leq \frac{2d}{k\eta} \cdot \frac{F(\mathbf{w}_0)}{t} + \frac{d\sigma^2}{k}(\eta L + 1) \triangleq \frac{\alpha}{t} + \beta \tag{16}$$

where $\alpha \triangleq \frac{2d}{k\eta}F(\mathbf{w}_0)$ and $\beta \triangleq \frac{d\sigma^2}{k}(\eta L + 1)$.

Based on Lemma 2, we obtain the following upper bound of the objective function in (14)

$$\begin{aligned} & \frac{\eta}{2d}\sum_{t=0}^{T-1}\left[\left(\frac{\alpha}{t} + \beta\right)\left(1 - \frac{\eta\mu}{d}k\right)^{T-1-t}\right] \cdot n_t\\ \triangleq\ & \frac{\eta}{2d}\sum_{t=0}^{T-1}(A_t B_t) \cdot n_t, \qquad (17) \end{aligned}$$

where $A_t \triangleq \frac{\alpha}{t} + \beta$ and $B_t \triangleq (1 - \frac{\eta\mu}{d}k)^{T-1-t}$.

Finally, the optimization problem to solve is

$$\begin{aligned} \max_{n_0, \ldots, n_{T-1}} \quad & \frac{\eta}{2d}\sum_{t=0}^{T-1}(A_t B_t) \cdot n_t \qquad (18)\\ \text{s.t.} \quad & \sum_{t=0}^{T-1} n_t \leq 0,\\ & n_t \in \{-k, \ldots, d-k\}\,,\ t = 0, \ldots, T-1. \end{aligned}$$

The objective function in (18) is linear in n_t with coefficient A_tB_t. We can prove the following monotonicity results.

Lemma 3: The coefficient A_tB_t first decreases with t and then increases with t. Specifically, we have

$$\begin{aligned} &A_{t+1}B_{t+1} < A_tB_t, \text{ for } t < \hat{t} \triangleq \left\lfloor \frac{-\alpha + \sqrt{\Delta}}{2\beta} \right\rfloor, \text{ and} \\ &A_{t+1}B_{t+1} \geq A_tB_t, \text{ for } t \geq \hat{t}, \end{aligned} \tag{19}$$

where $\Delta \triangleq \alpha^2 - \frac{4\alpha\beta}{lnB}$, $B \triangleq 1 - \frac{\eta\mu}{d}k$, and $\lfloor\cdot\rfloor$ is the floor function.

Given the monotonicity result in Lemma 3, we design the following adaptive sparsity levels

$$\begin{cases} n_t = +\gamma k \Rightarrow k_t = (1+\gamma)k, & t \in [0, \frac{\hat{t}}{2}) \cup [\frac{\hat{t}+T}{2}, T-1] \\ n_t = -\gamma k \Rightarrow k_t = (1-\gamma)k, & t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2}), \end{cases} \tag{20}$$

where γ is the scaling factor (i.e., a hyperparameter). In the above scheme, n_t takes the negative value half the training time and the positive value the other half, which satisfies the communication budget constraint. To maximize the objective function, we set n_t to be positive when A_tB_t is larger.

We can prove that the above adaptive sparsity levels result in a lower convergence error compared to the vanilla Top-K.

Corollary 1: (Convergence Error Bound using AdapTop-K in distributed SGD). Under the adaptive sparsity levels in Eq. (20), the optimality gap is upper bounded by

$$\begin{aligned} &\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq \mathbf{M}(k) + \mathbf{N}(k) \\ &\quad + \frac{\eta\gamma k}{2d} \underbrace{\left(\sum_{t=\frac{\hat{t}}{2}}^{\frac{\hat{t}+T-1}{2}} A_tB_t - \sum_{t=0}^{\frac{\hat{t}}{2}} A_tB_t - \sum_{t=\frac{\hat{t}+T-1}{2}}^{T-1} A_tB_t \right)}_{\text{always less than 0 because of (19)}} \\ &\quad < \underbrace{\mathbf{M}(k) + \mathbf{N}(k)}_{\text{upper bound for SGD with vanilla Top-K}} . \end{aligned} \tag{21}$$

The pseudo-code of distributed SGD with the proposed AdapTop-K method is provided in Algorithm 1.

V. Evaluation

In this section, we conduct experiments on two widely used datasets, namely MNIST and CIFAR-10, to validate the effectiveness of our proposed AdapTop-K method. We conduct experiments for M = 8 workers and use canonical networks to evaluate the performance on the image classification task using different algorithms: fully-connected network on the MNIST dataset, and Resnet18 on the CIFAR-10 dataset. The above datasets are the database commonly used for training various image processing systems. Other parameters information is shown in Table I. We use test accuracy to measure the learning performance. We compare our proposed AdapTop-K in SGD with the vanilla Top-K.

Fig. 1 shows the comparison results of the classific Top-K algorithm and our proposed AdapTop-K on the MNIST dataset. Fig. 1a and Fig. 1b show the test accuracy curves and the training loss curves on the MNIST dataset. It shows how the model performance changes with iterations for several different values of the sparsification factor (128 or 256). The accuracy of the original distributed SGD reaches 98.02%. In Fig. 1a, the AdapTop-K achieves 97.03% accuracy which is better than 96.64% from Top-K. In Fig. 1b, the AdapTop-K achieves 96.21% accuracy which is higher than 95.41% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 128 and 256, respectively.

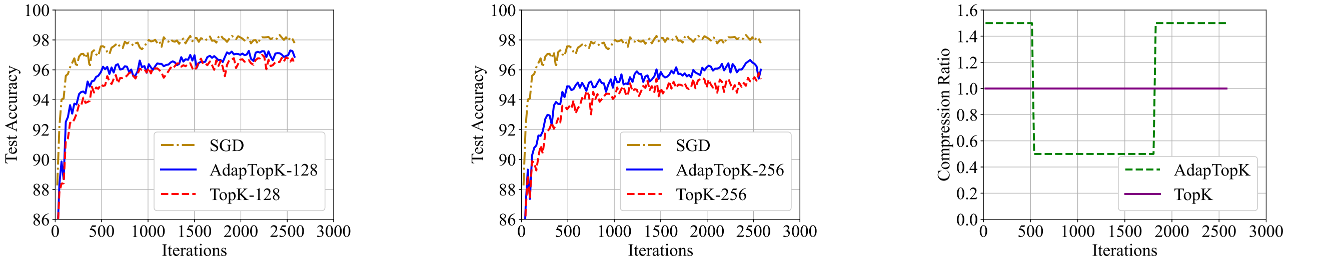
Similarly, Fig. 2 shows the comparison results of the fixed Top-K and our proposed AdapTop-K on CIFAR-10 dataset. Fig. 2a and Fig. 2b show the test accuracy curves and the training loss curves. It shows how the model performance changes with iterations for several different values of the sparsification factor (128 or 256). The accuracy of the original distributed SGD reaches 90.92%. In Fig. 2a, the AdapTop-K achieves 82.11% accuracy which is better than 81.36% from Top-K. In Fig. 2b, the AdapTop-K achieves 80.31% accuracy which is higher than 79.30% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 128 and 256, respectively. We keep the communication

Algorithm 1 AdapTop-K in Distributed SGD

```
Input: Maximum iterations number $T$, learning rate $\eta$, initial point $\mathbf{w}_0 \in \mathbb{R}^d$, fixed $k$ value, adjusted scale factor $\gamma$, hyper-parameters $\hat{t}$
Output: $\mathbf{w}_t$
for $t = 0, 1, ...T-1$ do
    On each worker $i = 1, ..., M$:
    Compute stochastic local gradient $\mathbf{g}_t^i$
    if $t \in [\frac{\hat{t}}{2}, \frac{\hat{t}+T}{2})$ then
        Set $k_t$ to $k - \gamma k$
    else
        Set $k_t$ to $k + \gamma k$
    end if
    Compress gradient $\mathbf{g}_t^i$ to $\mathcal{C}_{k_t}[\mathbf{g}_t^i]$
    Send $\mathcal{C}_{k_t}[\mathbf{g}_t^i]$ to server
    Receive $\mathbf{w}_{t+1}$ from server
    On server:
    Collect $M$ compressed gradients $\mathcal{C}_{k_t}[\mathbf{g}_t^i]$ from workers
    Aggregation: $\mathcal{C}_{k_t}[\mathbf{g}_t] = \sum_{i=1}^{M} \mathcal{C}_{k_t}[\mathbf{g}_t^i]$
    Update global parameters: $\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta}{M}\mathcal{C}_{k_t}[\mathbf{g}_t]$
    Send $\mathbf{w}_{t+1}$ back to all workers
end for
```

Dataset	MNIST	CIFAR-10
Networks	fully-connected network	ResNet18
Model Size	$d = 785$	$d = 1 \times 10^7$
Learning Rate	0.1	0.1
Batch Size	32	32
Workers	8	8
Iterations	3,000	7,000
Compression Ratio	128/256/512	128/256/512
γ	0.5	0.5

TABLE I: Experimental Setting.

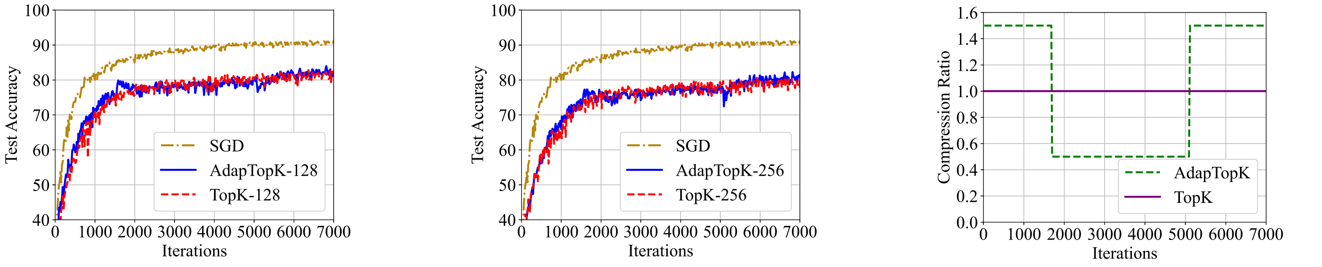


(a) Accuracy ($\frac{d}{k}$=128)

(b) Accuracy ($\frac{d}{k}$=256)

(c) Compression level on MNIST Dataset

Fig. 1: Evaluation results of different methods on MNIST Dataset.

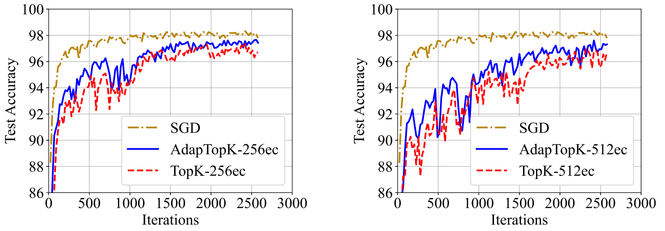


(a) Accuracy ($\frac{d}{k}$=128)

(b) Accuracy ($\frac{d}{k}$=256)

(c) Compression level on CIFAR-10 Dataset

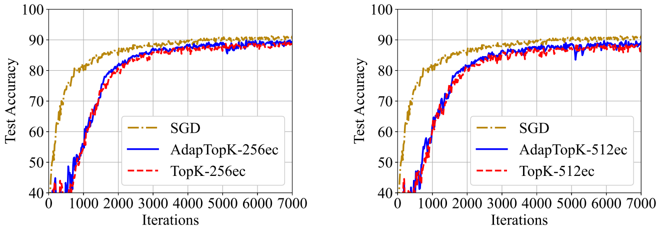
Fig. 2: Evaluation results of different methods on CIFAR-10.



(a) Accuracy with ec ($\frac{d}{k}$=256)

(b) Accuracy with ec ($\frac{d}{k}$=512)

Fig. 3: Evaluation with error compensation on MNIST.



(a) Accuracy with ec ($\frac{d}{k}$=256)

(b) Accuracy with ec ($\frac{d}{k}$=512)

Fig. 4: Evaluation with error compensation on CIFAR-10.

cost of the AdapTop-K stable compared with the classic Top-K in the total training process. It can be seen that our adaptive sparsification strategy can effectively improve the convergence rate and model performance with the pure Top-K algorithm. Fig. 1c and Fig. 2c both show the gradient sparsification level in the training process of AdapTop-K on different datasets. We can see that AdapTop-K significantly increases the bits assigned at the early stage and the late stage of training and improves the gradient accuracy as the training goes on.

After that, we add the error compensation [7] (abbreviated as ec) in Fig. 3 and Fig. 4 in our experiments, because it is a popular technique to improve the performance of distributed SGD with gradient compression. It shows how the model performance changes with iterations for several different values of the sparsification factor (256 or 512) when we add the error compensation. In these experiments, we use the bigger compression ratios (e.g., 256 and 512) because error compensation may reduce optimization errors in the training process to improve the total performance. Fig. 3 and Fig. 4 show the comparison results of the classific Top-K algorithm and our proposed AdapTop-K (all with error compensation) on MNIST and CIFAR-20 datasets. In Fig. 3a, the AdapTop-K achieves 97.50% accuracy which is higher than 96.71% from Top-K. In Fig. 3b, the AdapTop-K achieves 97.10% accuracy which is better than 96.24% from Top-K. In Fig. 4a, the AdapTop-K achieves 89.18% accuracy which is better than 88.66% from Top-K. In Fig. 4b, the AdapTop-K achieves 88.68% accuracy which is higher than 87.64% from Top-K. The curve corresponding to the AdapTop-K achieves better performance than fixed Top-K compression when the compression ratios ($\frac{d}{k}$) are 256 and 512, respectively. The results show that the AdapTop-K algorithm with error compensation achieves better performance under stable communication cost. Overall, the evaluation results demonstrate that the AdapTop-K outperforms the baselines.

VI. Conclusion

This paper proposes AdapTop-K, a novel adaptive gradient sparsification strategy for distributed SGD. The proposed method adjusts the sparsification levels adaptively by considering the gradient and the current iteration step. The experimental results for image classification show that AdapTop-K is superior to the state-of-the-art gradient compression methods in reducing the communication cost.

Acknowledgment

The work described in this paper was substantially sponsored by the project 62101471 supported by NSFC and was partially supported by the Shenzhen Research Institute, City University of Hong Kong. The work was also partially supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 21201420 and CityU 11201422), Shenzhen Science and Technology Funding Fundamental Research Program (Project No. 2021Szvup126), Shandong Provincial Natural Science Foundation (Project No. ZR2021LZH010). This work was supported in part by the CityU grants 7005660, 7005849, InnoHK initiative, the Government of the HKSAR, Laboratory for AI-Powered Financial Technologies.

VII. Appendix

A. *Proof for Theorem 1*

Using Eq. (8) and Assumption 1, we get:

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \le F(\mathbf{w}_t) - \eta\langle\nabla F(\mathbf{w}_t), \mathcal{C}(\mathbf{g}_t)\rangle + \frac{\eta^2 L}{2}\mathbb{E}\|\mathcal{C}(\mathbf{g}_t)\|^2$$

(use $\mathbb{E}\|\mathcal{C}(\mathbf{g}_t)\|^2 = \mathbb{E}\|\mathcal{C}(\mathbf{g}_t) - [\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))]\|^2 +$ $\mathbb{E}\|\mathbb{E}[\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))]\|^2$and Assumption 3)

$$\le F(\mathbf{w}_t) - \eta\langle\nabla F(\mathbf{w}_t), \mathbb{E}[\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))]\rangle + \frac{\eta^2 L}{2}(\sigma^2 + \mathbb{E}\|\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))\|^2)$$

$$\le F(\mathbf{w}_t) + \frac{\eta}{2}(\mathbb{E}\|\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))\|^2) - 2\langle\nabla F(\mathbf{w}_t), \mathbb{E}[\mathcal{C}(\mathbf{g}_t) - (\mathbf{g}_t - \nabla F(\mathbf{w}_t))]\rangle) + \frac{\eta^2 L}{2}\sigma^2 \ (\eta \le \frac{1}{L})$$

(from $\mathbb{E}\|\nabla F(\mathbf{w}_t) + \mathcal{C}(\mathbf{g}_t) - \mathbf{g}_t\|^2 = \mathbb{E}\|\nabla F(\mathbf{w}_t)\|^2 +$ $\mathbb{E}\|\mathcal{C}(\mathbf{g}_t) - \mathbf{g}_t\|^2 + 2\mathbb{E}\langle\nabla F(\mathbf{w}_t), \mathcal{C}(\mathbf{g}_t) - \mathbf{g}_t\rangle)$)

$$\le F(\mathbf{w}_t) + \frac{\eta}{2}(\mathbb{E}\|\mathcal{C}(\mathbf{g}_t) - \mathbf{g}_t\|^2 - \mathbb{E}\|\nabla F(\mathbf{w}_t)\|^2) + \frac{\eta^2 L}{2}\sigma^2$$

(from $Eq.$ (12) and assume that $k_t = k + n_t$, we have:

$$\mathbb{E}\|b_t(\mathbf{w})\|^2 = \mathbb{E}\|\mathbf{g}_t - \mathcal{C}(\mathbf{g}_t)\|^2 \le \mathbb{E}[(1 - \frac{k_t}{d})\|\mathbf{g}_t\|^2]$$
$$\le \mathbb{E}[(1 - \frac{k}{d})\|\nabla F(\mathbf{w}_t)\|^2 + (1 - \frac{k}{d})\sigma^2 - \frac{n_t}{d}\|\mathbf{g}_t\|^2],$$

then put this equation back to our above derivation)

$$\le F(\mathbf{w}_t) - \frac{\eta k}{2d}\|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta}{2}(1 - \frac{k}{d} + \eta L)\sigma^2 - \frac{\eta n_t}{2d}\|\mathbf{g}_t\|^2.$$

Therefore, we use Assumption 2 and get convergence rate as

$$\mathbb{E}[F(\mathbf{w}_{t+1})] - F^* \le (1 - \frac{\eta k \mu}{d})(\mathbb{E}(F(\mathbf{w}_t) - F^*) + \frac{\eta}{2}(1 - \frac{k}{d} + \eta L)\sigma^2 - \frac{\eta n_t}{2d}\|\mathbf{g}_t\|^2.$$

After recursion and simplification, we get:

$$\begin{aligned}\mathbb{E}[F(\mathbf{w}_T)] - F^* \le{}& (1 - \frac{\eta\mu}{d}k)^T[\mathbb{E}[F(\mathbf{w}_0)] - F^*] \\ &+ \frac{d}{2k\mu}(1 - \frac{k}{d} + \eta L)\sigma^2[1 - (1 - \frac{\eta\mu}{d}k)^T] \\ &- \sum_{t=0}^{T-1}[(\frac{\eta n_t}{2d}\|\mathbf{g}_t\|^2)(1 - \frac{\eta\mu}{d}k)^{T-1-t}].\end{aligned}$$

References

[1] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang *et al.*, "Large scale distributed deep networks," in *in Proceedings of Conference in Neural In- formation Processing Systems (NeurIPS)*, 2012.

[2] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 40, no. 1, pp. 342–358, 2021.

[3] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the Convergence of FedAvg on Non-IID Data," in *Proceedings of International Conference on Learning Representations (ICLR)*, 2020.

[4] A. Ajalloeian and S. U. Stich, "On the convergence of SGD with biased gradients," in *Proceedings of Workshop in International Conference on Machine Learning (ICML)*, 2020.

[5] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[6] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-Efficient SGD via Gradient Quantization and Encoding," in *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[7] J. Wu, W. Huang, J. Huang, and T. Zhang, "Error compensated quantized SGD and its applications to large-scale distributed optimization," in *Proceedings of International Conference on Machine Learning (ICML)*. PMLR, 2018, pp. 5325–5333.

[8] J. Wangni, J. Wang, J. Liu, and T. Zhang, "Gradient sparsification for communication-efficient distributed optimization," in *Proceedings of Conference in Neural Information Processing Systems (NeurIPS)*, 2018.

[9] G. Yan, T. Li, S.-L. Huang, T. Lan, and L. Song, "AC-SGD: Adaptively Compressed SGD for Communication-Efficient Distributed Learning," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 40, no. 9, pp. 2678–2693, 2022.

[10] H. Karimi, J. Nutini, and M. Schmidt, "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Lojasiewicz Condition," in *Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2016, pp. 795–811.

[11] P. Han, S. Wang, and K. K. Leung, "Adaptive gradient sparsification for efficient federated learning: An online learning approach," in *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2020, pp. 300–310.

[12] H. Zhang, J. Li, K. Kara, D. Alistarh, J. Liu, and C. Zhang, "ZipML: Training linear models with end-to-end low precision, and a little bit of deep learning," in *Proceedings of International Conference on Machine Learning (ICML)*. PMLR, 2017, pp. 4035–4043.

[13] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li, "Terngrad: Ternary gradients to reduce communication in distributed deep learning," in *Proceedings of Conference in Neural Information Processing Systems (NeurIPS)*, 2017.

[14] K. Mishchenko, E. Gorbunov, M. Takáč, and P. Richtárik, "Distributed learning with compressed gradient differences," *arXiv preprint arXiv:1901.09269*, 2019.

[15] M. Ruan, G. Yan, Y. Xiao, L. Song, and W. Xu, "Adaptive Top-K in SGD for Communication-Efficient Distributed Learning," *arXiv preprint arXiv:2210.13532*, 2022.