DAGC: Data-Aware Adaptive Gradient Compression

Rongwei Lu[†], Jiajun Song[‡‡*], Bin Chen[‡], Laizhong Cui[§††], Zhi Wang[†¶**]
[†]Tsinghua-Berkeley Shenzhen Institute, Tsinghua Shenzhen International Graduate School
[‡‡]School of Software Technology, Dalian University of Technology
[‡]Harbin Institute of Technology, Shenzhen
[§]College of Computer Science and Software Engineering, Shenzhen University
[¶]Peng Cheng Laboratory
[††]Guangdong Laboratory of Artificial Intelligence and Cyber Economics (SZ)
lrw21@mails.tsinghua.edu.cn, 1372118616@mail.dlut.edu.cn,
chenbin2021@hit.edu.cn, cuilz@szu.edu.cn, wangzhi@sz.tsinghua.edu.cn

***Abstract*—Gradient compression algorithms are widely used to alleviate the communication bottleneck in distributed ML. However, existing gradient compression algorithms suffer from accuracy degradation in Non-IID scenarios, because a uniform compression scheme is used to compress gradients at workers with *different data distributions and volumes*, since workers with larger volumes of data are forced to adapt to the same aggressive compression ratios as others. Assigning different compression ratios to workers with different data distributions and volumes is thus a promising solution. In this study, we first derive a function from capturing the correlation between the number of training iterations for a model to converge to the same accuracy, and the compression ratios at different workers; This function particularly shows that workers with larger data volumes should be assigned with higher compression ratios[1] to guarantee better accuracy. Then, we formulate the assignment of compression ratios to the workers as an n-variables chi-square nonlinear optimization problem under fixed and limited total communication constrain. We propose an adaptive gradient compression strategy called DAGC, which assigns each worker a different compression ratio according to their data volumes. Our experiments confirm that DAGC can achieve better performance facing highly imbalanced data volume distribution and restricted communication.**

***Index Terms*—Distributed Machine Learning, Non-IID, Data-aware Adaptive Gradient Compression**

I. Introduction

Lossy gradient compression algorithms cause worse model convergence in Non-IID scenarios compared to the same setting with IID datasets [2], [3]. For example, the same gradient compression algorithm [4] (with the hyperparameter set to 10%) reduces the accuracy by only 0.7% compared to the bulk synchronous parallel (BSP) [5] as a baseline in IID scenarios, but the same setting reduces the accuracy by 10.4% in the Non-IID scenarios [3]. The reason for the accuracy degradation is that they use the same aggressive gradient compression ratios for different workers, while different workers usually have different data volumes and distributions [6]–[8]. For example, in Flickr-mammal (denoted as Flickr in the following) [3], the worker with the largest data volume has 78% more samples than the worker with the second largest data volume (divided by subcontinent). In Google Landmark Dataset v2 [9], the worker with the largest data volumes has at least 213% more images than other workers (divided by continent).

For convenience, we denote workers with large (as well as small) data volumes as *large (small) workers* and the number of local samples as the *worker size*. Most of the existing designs of gradient compression algorithms neglect the differences in the *worker size*; For studies that have proposed adaptive algorithms for adjusting the gradient compression ratios according to differences in the data distributions (like SkewScout in [3]), *large workers* still use the same aggressive compression ratios as *small workers*, and a lot of critical information that can speed up the convergence is lost when transmitting gradients.

Based on a measurement study, we reveal the following insights for designing a data-aware gradient compression strategy. First, the uniform compression strategy is not optimal in a communication-constrained Non-IID environment because workers with different data distributions and volumes expect different compression ratios. Second, to converge to the same accuracy, the gradient compression strategy that sets a higher compression ratio for workers with larger local training samples, can reduce a certain amount of training time compared to the uniform compression. Based on these insights, we propose to design an adaptive gradient compression algorithm according to the *worker size* to achieve the optimal compression ratio setting with fixed and limited communication costs.

The technical challenge in designing this algorithm is: given the total amount of communication, how to determine the compression ratio of each worker? To answer this question, we first derive the convergence rate of distributed SGD with error-feedback (denoted as D-EF-SGD), under relative compression strategy and different compression ratios. We derive the correlation between the number of training iterations for a model to converge to the same accuracy, and the compression ratios. This correlation covers previous studies [10]–[12], where they restrict each worker with the same compression ratio and

[*] Jiajun Song has been pre-admitted to Tsinghua Shenzhen International Graduate School when doing this work.

[**]Corresponding author.

[1]In this work, the compression ratio is equal to the compressed data divided by the uncompressed data, referring to the gradient compression part of Sec. II in [1].

training weight. Secondly, we find the dominant term (*i.e.*, the term that mostly affects the convergence rate) under the limited-communication environment. We denote the number of workers as n, the compression ratio of the i-th worker as δ_i, and the dominant term as $\Phi(\delta_1, \ldots, \delta_n)$ (abbreviated as Φ). Φ is linearly related to the number of iterations for the model to converge to the same accuracy. Thirdly, we formulate finding a proper assignment of δ_i as an n-variable chi-square nonlinear optimization problem with one constraint. Since it is not possible to solve this problem directly with the Lagrange multiplier method (the optimal solution under this method is outside the domain of δ_i), we divide this problem into an n-variable chi-square nonlinear optimization problem with one constraint, which can be solved by the Lagrange multiplier method, and a problem of finding the minimal value of a one-dimensional function. Finally, we find the optimal δ_i by traversing the minimum value of Φ for n cases.

Based on these analyses, we propose DAGC (Fig. 1 is a graphical illustration), a low-cost data-aware adaptive gradient compression algorithm that sets different compression ratios depending on the *worker size*. We denote the local dataset size divided by the global dataset size as p_i, which is equal to the training weight of the i-th worker [2], [13]. We have $\frac{\delta_i}{\delta_j} \approx (\frac{p_i}{p_j})^{2/3}$ in DAGC. This supports the the conclusion of the measurement study, which suggests that large compression ratios should be assigned to workers with larger p_i. The time complexity of DAGC to find the optimal δ_i is $\mathcal{O}(n)$.

Our contributions are as follows:

- We experimentally reveal that setting higher compression ratios to *large workers* converges faster than the uniform compression under the fixed and limited communication volume, saving up to 33.75% iterations for Logistic on FMNIST.
- We reveal that the D-EF-SGD algorithm using a relative compressor with uneven compression ratios suffers from a linear slow-down in Φ in the communication-constraint Non-IID environment. We propose DAGC by solving an n-variable chi-square nonlinear asymmetric optimization problem with a communication constraint, which theoretically achieves the minimum of Φ.
- We employ the DAGC in both the real-world Non-IID and artificially partitioned Non-IID datasets. The experimental results confirm the correctness of our theory and show that the DAGC can save up to 26.19% of iterations to converge to the same accuracy compared to the uniform compression.

The remainder of this paper is organized as follows. We introduce the preliminaries in Sec. II. We empirically demonstrate a faster convergence shown when *large workers* are set high compression ratios in Sec. III. We describe the optimal compression ratios formulation and propose DAGC in Sec. IV. We show experimental results in Sec. V. We discuss the related work in Sec. VI, and present the conclusion in Sec. VII.

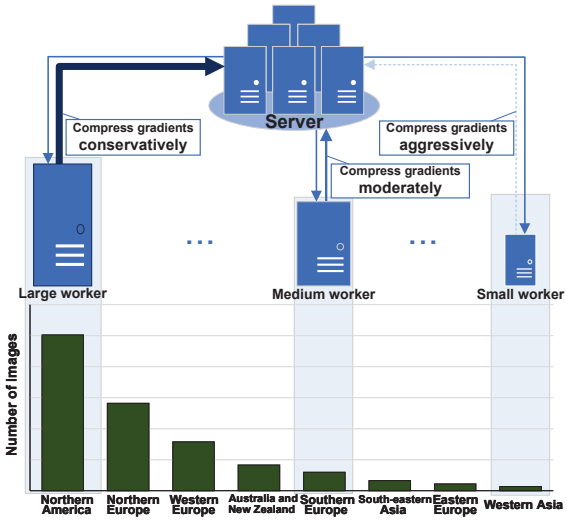


Fig. 1: High-level design of DAGC. DAGC sets different compression ratios to workers depending on the *worker size*. *Large workers* (*i.e.*, the workers with large data volumes and similarly to *small* and *medium workers*) are assigned conservative compression ratios, and *small workers* adopt aggressive compression ratios. The bar chart of the number of samples owned by each worker refers to Fig. 13 in [3].

II. Preliminaries

We focus on distributed ML in the data-parallel mode via D-EF-SGD algorithms with relative compressors in Non-IID scenarios. We give a brief description of distributed ML, Non-IID, gradient compression, and relative compressors in turn and highlight the focus of this study.

Distributed ML: We consider the distribution problem in this work.

$$f^* := \min_{\mathbf{x}} \left[f(\mathbf{x}) := \sum_{i=1}^{n} p_i f_i(\mathbf{x}) \right],$$

where the objective function f is split among n terms $f_i, i \in \{1, 2, \ldots, n\}$, and p_i is the training weight of the i-th worker such that $p_i \geq 0$ and $\sum_{i=1}^{n} p_i = 1$. For convenience, we set the serial number of workers according to the training weight[2], which means $p_i \geq p_{i+1}, \forall i \in \{1, 2, \ldots, n-1\}$.

Non-IID: The Non-IID scenario refers to the size and classification of datasets stored by each worker on distributed workers, which is a primary challenge in modern DML. Some workers have more samples and categories, while others have fewer. This phenomenon is usually due to the workers' geographical location and user-oriented differences. Skewed distribution of data labels across devices/locations is a common type of Non-IID [3], which is considered in this work. In the theoretical analysis, we use ζ to measure the size of data heterogeneity, following **Assumption 4** in [10].

Gradient Compression: Referring to the compression strategy [14], this technique can be divided into (i) quantization

[2]The following part of the paper also maintains the order of the training weight.

[15]–[17], which maps high precision to low precision, thereby reducing the number of bits transmitted; (ii) sparsification [18], [19], which keeps only some elements of the gradient, and set the rest to 0; (iii) low-rank [20], which decomposes the gradient matrix to obtain two or more low-rank matrices.

D-EF-SGD with relative compressors: Relative compressors are very popular in sparsification compressors [10]. We denote the compression ratio as δ and the relative compressor as $\mathbf{C}_\delta$. $\mathbf{C}_\delta$ is a mapping: $\mathbb{R}^d \to \mathbb{R}^d$, having the property:

$$\mathbb{E}_{\mathbf{C}_\delta}\|\mathbf{C}_\delta(\mathbf{x}) - \mathbf{x}\|^2 \leq (1-\delta)\|\mathbf{x}\|^2.$$

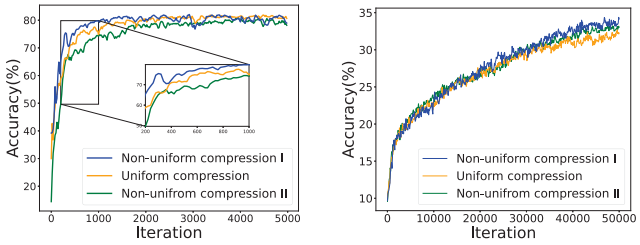
D-EF-SGD with relative compressors [13] and distributed quantized SGD (D-QSGD) [21], [22] are two dominant compressors in modern distributed ML. D-EF-SGD has two advantages in communication-constrained Non-IID scenarios. Firstly, it can achieve lower compression ratios[3] (*i.e.*, transferring less information), which is beneficial for resource-constrained scenarios. Secondly, D-EF-SGD is less dependent on ζ and therefore more suitable for high-skewness Non-IID scenarios [10]. For these reasons, we focus on D-EF-SGD with relative compressors.

III. Motivating Examples

In this section, we aim to give some motivating experiments to demonstrate that 1) A compression strategy that sets different compression ratios for workers with different sizes converges faster than uniform compression in a communication-constrained environment. 2) Strategies that set a higher compression ratio for *large workers* tend to converge faster than those for small ones, saving up to 33.75% iterations. To this end, We propose two non-uniform compression strategies: 1) *Non-uniform compression I gives a higher compression ratio to the large worker* and a lower compression ratio to small workers; 2) *Non-uniform compression II gives a lower compression ratio to the large worker* and a higher compression ratio to small workers.

Empirically, we conduct both Image Classification (Logistic on Fashion-MNIST, abbreviated as FMNIST [26]) and Speech Recognition (LSTM on Speech Commands, abbreviated as SCs [27]) tasks to ensure the generality of the experimental results. We dichotomize the workers into *large* and *small workers* in order to better investigate the experimental results. We set 11 workers, consisting of one *large worker*, whose local dataset accounts for 50% of the global dataset, and 10 *small workers*, whose datasets each account for 5%. In these experiments, communication is constrained (the average compression ratio referring to the aggressive compression ratio $k_{min} = 0.1\%$ in the appendix experiments in [19]) and the total communication volume is consistent (*i.e.*, fixing $\sum_{i=1}^{n}\delta_i$).

[3]The smallest compression ratio of D-QSGD can be $\frac{1}{32}$ [23], [24], nearly 3.4%, while the compression ratio of spasification can be less than 0.1% [25]. Meanwhile, the compression ratio of D-QSGD is discrete, and this makes adaptively adjusting ratios more complicated. This issue is also discussed in the design of DC2 [1].



(a) Logsitic@FMNIST (b) LSTM@SCs

Fig. 2: Convergence rate when using different compression strategies during the training of Logistic@FMNIST (a) and LSTM@SCs (b). Non-uniform compression scheme I (as well as scheme II) gives a higher (lower) compression ratio to the worker with large data volumes, and uniform compression gives each worker the same compression ratio. Non-uniform compression scheme I perform the best among the three strategies.

Logistic on FMNIST: Fig. 2(a) shows that the Non-uniform compression I ($\delta_1 = 1\%$, and $\delta_i = 0.01\%$, $i \in [2, n]$) achieves faster in the early iterations than two other strategies. To converge to the same accuracy (80%), the Non-uniform compression I ($\delta_i = 0.1\%$, $i \in [1, n]$) can reduce up to 33.75% of the number of iterations (from $1,600$ iterations to $1,060$ iterations) compared to uniform compression. FMNIST converges more easily, and a compression ratio of 0.1% corresponds to the conservative compression, under which Non-uniform compression I performs optimally and Non-uniform compression II ($\delta_1 = 0.01\%$, and $\delta_i = 0.11\%$, $i \in [2, n]$) performs worst.

LSTM on SCs: Fig. 2(b) shows that the model using non-uniform compression strategy I converge the fastest under the condition that the communication volume is fixed. To converge to the same accuracy (32%), Non-uniform compression I can reduce up to 17.59% and 11.44% of training time compared to uniform compression (from $43,200$ iterations to $35,600$ iterations) and Non-uniform compression II (from $40,200$ iterations to $35,600$ iterations), respectively. The SCs dataset is more challenging to converge than FMNIST and CIAFR-10, and a compression ratio of 0.1% is extremely aggressive for SCs. We can see that the training has not fully converged after $50,000$ iterations. Under such conditions, Non-uniform compression I and II both perform better than uniform compression. Detailed compression ratios are the same as Logistic on FMNIST.

In summary, experiment results shown above confirm the fact that uniform compression is not the optimal strategy when facing the difference in worker size. Instead, the proposed strategy that assigns higher compression ratios to *large workers* is an effective approach for improving training speed.

IV. Theoretical Analysis

In this section, we solve the core challenge in this study: **Given the total amount of communication, how to theoret-**

Algorithm 1: D-EF-SGD with the relative compressor and different compression ratios

Input: number of workers n, initial parameters $\mathbf{x}_0$, step-size γ, relative compressor $\mathbf{C}$, initial local error $\mathbf{e}_0^i = \mathbf{0}_d$, training weight p_i, compression ratios $\delta_1, \ldots, \delta_n$

Output: $\mathbf{x}_T$

for $t = 0, \ldots, T-1$ **do**
  /* Worker side */
  for $i = 1, \ldots, n$ **do**
    Download $\mathbf{x}_t$ from the server;
    $\mathbf{g}_t^i := \mathbf{g}^i(\mathbf{x}_t)$;
    $\hat{\Delta}_t^i := \mathbf{C}_{\delta_i}(\mathbf{e}_t^i + \mathbf{g}_t^i)$;
    $\mathbf{e}_{t+1}^i := \mathbf{e}_t^i + \mathbf{g}_t^i - \hat{\Delta}_t^i$;
    Upload $\hat{\Delta}_t^i$;
  end
  /* Server side */
  Gather all $\hat{\Delta}_t^i$;
  $\mathbf{x}_{t+1} := \mathbf{x}_t - \gamma \sum_{i=1}^n p_i \hat{\Delta}_t^i$;
  Broadcast $\mathbf{x}_{t+1}$ to all workers
end
Return $\mathbf{x}_T$;

ically determine the compression ratio of each worker?

First, we show the pseudo-code of D-EF-SGD with the relative compressor and different compression ratios (denoted as Algorithm 1).

Second, we derive the convergence rate of Algorithm 1, which shows the correlation between the number of iterations for the model to converge to the same accuracy and δ_i, considering the non-convex, convex, and strongly convex cases. (**Theorem 1, 2, 3** in Sec. IV-B and the proof in Sec. IV-F)

Third, we analyze the convergence rate and find the dominant term $\Phi(\delta_1, \ldots, \delta_n)$. (Sec. IV-C)

Fourth, we derive the optimal δ_i by solving the minimum of $\Phi(\delta_1, \ldots, \delta_n)$, which is an n-variable chi-square nonlinear optimization problem with one constraint, *i.e.*, under fixed and limited total communication constrain. (**Theorem 4** in Sec. IV-D and the proof in Sec. IV-G)

Finally, we propose DAGC based on **Theorem 4**, which adaptive assigns δ_i based on p_i and the average compression ratio. (Algorithm 2 in Sec. IV-E)

A. Regularity assumptions

We follow Assumptions 1-4 as [10]. That is, we assume L-smooth functions with gradient noise of SGD assumed to have zero mean and variance σ^2. We measure data heterogeneity with constants $\zeta_i^2 > 0, Z^2 \geq 1$ that bound the variance across the n workers. We assume objective functions μ-convex in **Theorem 2, 3**.

B. Convergence rate of Algorithm 1

Theorem 1 (Non-convex convergence rate of Algorithm 1). Let $f : \mathbb{R}^d \to \mathbb{R}$ be L-smooth. Then there exists a stepsize $\gamma \leq \frac{\delta_{min}}{4LZ\sqrt{nC_Z}}$, where $C_Z = \sum_{i=1}^n \frac{\delta_{min}}{\delta_i} p_i^2$, such that at most

$$\mathcal{O}\Big(\frac{\sigma^2 \sum_{i=1}^n p_i^2}{\epsilon^2} + \frac{\sqrt{n}(\zeta \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}} + \sigma\sqrt{\sum_{i=1}^n p_i^2})}{\epsilon^{\frac{3}{2}}\sqrt{\delta_{min}}} + \frac{\sqrt{n}Z \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}}}{\epsilon\sqrt{\delta_{min}}}\Big) \cdot LF_0 \tag{1}$$

iterations of Algorithm 1 it holds $\mathbb{E}f(\mathbf{x}_{out}) - f^* \leq \epsilon$, where $F_0 \geq f(\mathbf{x}_0) - f^*$, and $\mathbf{x}_{out} = \mathbf{x}_t$ denotes an iterate $\mathbf{x}_t \in \{\mathbf{x}_0, \ldots, \mathbf{x}_{T-1}\}$, chosen at random uniformly.

Theorem 2 (Convex convergence rate of Algorithm 1, *i.e.*, $\mu = 0$). Let $f : \mathbb{R}^d \to \mathbb{R}$ be L-smooth and μ-convex. Then there exists a stepsize $\gamma \leq \frac{\delta_{min}}{14LZ\sqrt{nC_Z}}$, where $C_Z = \sum_{i=1}^n \frac{\delta_{min}}{\delta_i} p_i^2$, such that at most

$$\mathcal{O}\Big(\frac{\sigma^2 \sum_{i=1}^n p_i^2}{\epsilon^2} + \frac{\sqrt{nL}(\zeta \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}} + \sigma\sqrt{\sum_{i=1}^n p_i^2})}{\epsilon^{3/2}\sqrt{\delta_{min}}} + \frac{\sqrt{n}LZ \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}}}{\epsilon\sqrt{\delta_{min}}}\Big) \cdot R_0^2 \tag{2}$$

iterations of Algorithm 1 it holds $\mathbb{E}f(\mathbf{x}_{out}) - f^* \leq \epsilon$, where $R_0^2 \geq \|\mathbf{x}_0 - \mathbf{x}_*\|$, and $\mathbf{x}_{out} = \mathbf{x}_t$ denotes an iterate $\mathbf{x}_t \in \{\mathbf{x}_0, \ldots, \mathbf{x}_{T-1}\}$, chosen at random uniformly.

Theorem 3 (Strong convex convergence rate of Algorithm 1, *i.e.*, $\mu > 0$). Let $f : \mathbb{R}^d \to \mathbb{R}$ be be L-smooth and μ-convex. Then there exists a stepsize $\gamma \leq \frac{\delta_{min}}{14LZ\sqrt{nC_Z}}$, where $C_Z = \sum_{i=1}^n \frac{\delta_{min}}{\delta_i} p_i^2$, such that at most

$$\tilde{\mathcal{O}}\Big(\frac{\sigma^2 \sum_{i=1}^n p_i^2}{\mu\epsilon} + \frac{\sqrt{nL}(\zeta \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}} + \sigma\sqrt{\sum_{i=1}^n p_i^2})}{\mu\sqrt{\delta_{min}\epsilon}} + \frac{\sqrt{n}LZ \sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}}}{\mu\sqrt{\delta_{min}}}\Big) \tag{3}$$

iterations of Algorithm 1 it holds $\mathbb{E}f(\mathbf{x}_{out}) - f^* \leq \epsilon$, and $\mathbf{x}_{out} = \mathbf{x}_t$ denotes an iterate $\mathbf{x}_t \in \{\mathbf{x}_0, \ldots, \mathbf{x}_{T-1}\}$, chosen at random with probability proportional to $(1 - \min\{\frac{\mu\gamma}{2}, \frac{\delta_{min}}{4}\})^{-t}$.

In the following analysis of the convergence rate, we mainly focus on **Theorem 3**, and the rest are similar.

C. Analysis of convergence rate of Algorithm 1

To simplify the analysis, we define a n-variable function $\Phi(\delta_1, \ldots, \delta_n) = \frac{\sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}}}{\sqrt{\delta_{min}}}$, and add an extra assumption denoted as **Assumption***, which applies within the context of this work.

Assumption*: In the communication-constraint Non-IID environment, we assume that the order of magnitude of $(\sum_{i=1}^n \frac{p_i}{\sqrt{\delta_i}})\zeta$ is greater than $\sqrt{\sum_{i=1}^n p_i^2}\sigma$.

This assumption applies to the communication-constraint Non-IID environment, which is the scope of this work. In this case, firstly, because gradient compression suffers from accuracy degradation in Non-IID scenarios, data heterogeneity ζ is not orders of magnitude lower than gradient noise σ. Second,

δ_i increases ζ by an order of magnitude. So **Assumption*** applies to this work.

We analyze the convergence rate in two cases:

• *Without gradient noise ($\sigma = 0$)*: Algorithm 1 converges sublinearly at the rate of $\mathcal{O}(\frac{\sqrt{nL}\zeta}{\mu\sqrt{\epsilon}}\Phi(\delta_1,\ldots,\delta_n))$. Given $\zeta > 0$, $\Phi(\delta_1,\ldots,\delta_n)$ directly affects the convergence.

• *With gradient noise ($\sigma \neq 0$)*: When ϵ is a higher order infinitesimal of δ_{min}, it converges at rates $\mathcal{O}(\frac{\sigma^2\sum_{i=1}^n p_i^2}{\mu\epsilon})$, independent of δ_{min} and ζ. When the order of magnitude of ϵ is greater than or equal to δ_{min}, the term $\mathcal{O}(\frac{\sqrt{nL}\zeta}{\mu\sqrt{\epsilon}}\Phi(\delta_1,\ldots,\delta_n))$ has a dominant effect due to **Assumption***.

In the following analysis, the scenarios we analyze are $\sigma = 0$ and $\sigma > 0$ while ϵ and δ have the same order of magnitude cases, where Algorithm 1 converges at the rate of $\mathcal{O}(\frac{\sqrt{nL}\zeta}{\mu\sqrt{\epsilon}}\Phi(\delta_1,\ldots,\delta_n))$. That is, in communication-constraint Non-IID environment, the convergence rate is slowing down linearly in $\Phi(\delta_1,\ldots,\delta_n)$. For the case where ϵ is a higher order of δ with $\sigma > 0$, previous work has analyzed [11], [12], [28], and we will not repeat here.

D. The optimal compression ratios

In this subsection, we give the theoretical basis (**Theorem 4**) to find the optimal compression ratios.

Theorem 4 (Optimal compression ratios). We have the following equation under the condition that the total communication traffic is determined, *i.e.*, $\sum_{i=1}^n \delta_i = n\bar{\delta}$. We set $\delta_j = \delta_{min} = \min\{\delta_1,\ldots,\delta_n\}$ and denote $P := \sum_{i=1}^n p_i^{2/3}$ Then the minimal of $\Phi(\delta_1,\ldots,\delta_n)$ can be divided into two different cases:

• *The first case is $j \neq n$*

$$\Phi(\delta_1,\ldots,\delta_n) \geq \frac{1}{n\bar{\delta_j}}(p_j(1+Q_j) + p_n Q_j(1+Q_j)), \quad (4)$$

where $Q_j = \frac{P - p_j^{2/3}}{p_n^{2/3}}$. (4) takes the equality at $\delta_j = \frac{n\bar{\delta}}{Q_j+1}$ and $\delta_i = \frac{n\bar{\delta}}{Q_j+1}\frac{p_i^{2/3}}{p_n^{2/3}}, i \neq j$.

• *The second case is $j = n$*

$$\Phi(\delta_1,\ldots,\delta_n) \geq \frac{1}{n\bar{\delta}}(p_j(1+Q_j) + p_{n-1} Q_j(1+Q_j)), \quad (5)$$

where $Q_j = \frac{P - p_j^{2/3}}{p_{n-1}^{2/3}}$. (5) takes the equality at $\delta_j = \frac{n\bar{\delta}}{Q_j+1}$ and $\delta_i = \frac{n\bar{\delta}}{Q_j+1}\frac{p_i^{2/3}}{p_{n-1}^{2/3}}, i \neq j$.

E. DAGC

Algorithm 2 provides the pseudo-code of DAGC. DAGC is designed as follows: 1) it solves for n local optimal solutions by traversing j in **Theorem 4** ($\delta_j = \min\{\delta_1,\ldots,\delta_n\}$) from 1 to n, and 2) it updates δ_i, $i \in \{1,\ldots,n\}$ when $\Phi(\delta_1,\ldots,\delta_n)$ takes a smaller value. The latest compression ratios are the optimal parameters.

This function is computationally inexpensive and the results can be used multiple times. The time complexity required for one traversal is $\mathcal{O}(n)$. For a given Non-IID partition, we only need to compute the optimal δ_i assigned to each worker once and use it multiple times. If $\bar{\delta}$ changes into a new compression ratio, denoted as $\bar{\delta}_{new}$, we can use $\delta_i = \delta_i\frac{\bar{\delta}_{new}}{\bar{\delta}}$ to update the compression ratios.

Algorithm 2: DAGC

Input: number of workers n, training weight p_i, average compression ratio $\bar{\delta}$
Output: compression ratio $\delta_1,\ldots,\delta_n$

```
/* φ_j is the minimum of Φ(δ_1,...,δ_n)
   with δ_j = min{δ_i}                    */
```

Initialize $\phi_{min} = +\infty$;
for $j = n, n-1, \ldots, 1$ **do**
  if $j == n$ **then**
    Use the right hand of (5) to calculate ϕ_j;
  else
    if $p_j == p_{j-1}$ **then**
      /* If weights are duplicated, skip this calculation */
      $\phi_j = \phi_{min}$;
    else
      Use the right hand of (4) to calculate ϕ_j;
    end
  end
  if $\phi_j < \phi_{min}$ **then**
    $\phi_{min} = \phi_j$ and update the optimal $\delta_1,\ldots,\delta_n$;
  end
end
Return $\delta_1,\ldots,\delta_n$;

*F. Proof of **Theorem 1, 2, 3***

We follow [11] and define a virtual sequence:

$$\tilde{\mathbf{x}}_0 = \mathbf{x}_0, \quad \tilde{\mathbf{x}}_{t+1} := \tilde{\mathbf{x}}_t - \gamma\sum_{i=1}^n p_i g_t^i.$$

The error term that indicates the distance from the virtual sequence to the actual sequence is

$$\tilde{\mathbf{x}}_t - \mathbf{x}_t = \gamma\sum_{i=1}^n p_i \mathbf{e}_t^i.$$

We will use the notations $\tilde{F}_t := \mathbf{E}f(\tilde{\mathbf{x}}_t) - f^*$, $F_t := \mathbf{E}f(\mathbf{x}_t) - f^*$, $G_t := \mathbf{E}\|\nabla f(x_t)\|^2$, $E_t = \sum_{i=1}^n p_i^2\mathbb{E}\|\mathbf{e}_t^i\|^2$.

Lemma 1. Let f be L-smooth. If the stepsize $\gamma \leq \frac{1}{4L}$, then it holds for the iterates of Algorithm 1:

$$\tilde{F}_{t+1} \leq \tilde{F}_t - \frac{\gamma}{4}G_t + \gamma^2\frac{L\sum_{i=1}^n p_i^2\sigma^2}{2} + \gamma^3\frac{nL^2}{2}E_t. \quad (6)$$

If f is in addition μ-convex, we have

$$X_{t+1} \leq (1 - \frac{\gamma\mu}{2})X_t - \frac{\gamma}{2}F_t + \gamma^2\sum_{i=1}^n p_i^2\sigma^2 + 3\gamma^3 nLE_t. \quad (7)$$

Proof. Similar to the analysis in [10], [11], we have

$$\begin{aligned}\tilde{F}_{t+1} &\leq \tilde{F}_t - \frac{\gamma}{4}G_t \\ &+ \gamma^2\frac{L}{2}\mathbb{E}\|\sum_{i=1}^n p_i\xi_t^i\|^2 + \gamma^3\frac{L^2}{2}\mathbb{E}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2, \\ X_{t+1} &\leq (1-\frac{\gamma\mu}{2})X_t - \frac{\gamma}{2}F_t \\ &+ \gamma^2\mathbb{E}\|\sum_{i=1}^n p_i\xi_t^i\|^2 + 3\gamma^3 L\mathbb{E}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2.\end{aligned}$$

With the independent ξ_t^i and **Assumption 3** in [10], we have

$$\mathbb{E}_{\xi_t}\|\sum_{i=1}^n p_i\xi_t^i\|^2 = \sum_{i=1}^n p_i^2\mathbb{E}_{\xi_t}\|\xi_t^i\|^2 \leq \sum_{i=1}^n p_i^2\sigma^2.$$

Moreover, we have

$$\mathbb{E}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2 = \mathbb{E}\|\sum_{i=1}^n p_i e_t^i\|^2 \leq {}^4 n\sum_{i=1}^n p_i^2\mathbb{E}\|e_t^i\|^2 = nE_t.$$

Finally, We get the desired results.

Lemma 2. It holds

$$E_{t+1} \leq (1-\frac{\delta_{min}}{2})E_t + \frac{2}{\delta_{min}}(C_\zeta\zeta^2 + C_Z Z^2 G_t) + \sum_{i=1}^n p_i^2\sigma^2, \tag{8}$$

where $C_\zeta = C_Z = \sum_{i=1}^n \frac{\delta_{min}}{\delta_i}p_i^2$.

Proof. With the analysis in [11] and [10], it follows

$$\begin{aligned}\mathbb{E}_{\xi_t^i, C_\delta}\|\mathbf{e}_{t+1}^i\|^2 &\leq (1-\frac{\delta}{2})\|\mathbf{e}_t^i\|^2 + \frac{2}{\delta}\|\nabla f_i(\mathbf{x}_t)\|^2 + (1-\delta)\sigma^2 \\ &\leq (1-\frac{\delta}{2})\|\mathbf{e}_t^i\|^2 + \frac{2}{\delta}(\zeta_i^2 + Z^2\|\nabla f(\mathbf{x}_t)\|^2) + \sigma^2,\end{aligned} \tag{9}$$

and the last inequality is followed by **Assumption 4** in [10].

Then we substitute the different compression ratio of the different workers into Eq. 9 and sum the result:

$$\begin{aligned}E_{t+1} &\leq (1-\frac{\delta_{min}}{2})\sum_{i=1}^n p_i^2\|\mathbf{e}_t^i\|^2 \\ &+ \frac{2}{\delta_{min}}(\sum_{i=1}^n \frac{\delta_{min}}{\delta_i}p_i^2)(\zeta^2 + Z^2 G_t) + \sum_{i=1}^n p_i^2\sigma^2,\end{aligned}$$

where $\zeta = \max\{\zeta_1, \ldots, \zeta_n\}, \delta_{min} = \min\{\delta_1, \ldots, \delta_n\}$.

Lemma 3. (Lyapunov function). Let f be L-smooth and $\gamma \leq \frac{\delta_{min}}{4LZ\sqrt{nC_Z}}$. Then it holds

$$\begin{aligned}\Xi_{t+1} &\leq \Xi_t - \frac{\gamma}{8}G_t + \gamma^2\frac{L\sum_{i=1}^n p_i^2\delta^2}{2} + \\ &\gamma^3\left(\frac{L^2 n}{\delta_{min}}\right)\left(\frac{2C_\zeta\zeta^2}{\delta_{min}} + \sum_{i=1}^n p_i^2\sigma^2\right),\end{aligned} \tag{10}$$

where $\Xi_t := \tilde{F}_t + bE_t, b = \frac{\gamma^3 L^2 n}{\delta_{min}}$. Furthermore, letting f be L-smooth, μ-convex and $\gamma \leq \frac{\delta_{min}}{14LZ\sqrt{nC_Z}}$, we have

$$\begin{aligned}\Psi_{t+1} &\leq \left(1 - \min\left\{\frac{\gamma\mu}{2}, \frac{\delta}{4}\right\}\right)\Psi_t - \frac{1}{8L}G_t + \gamma^2\sum_{i=1}^n p_i^2\sigma^2 \\ &+\gamma^3\left(\frac{12Ln}{\delta_{min}}\right)\left(\frac{2C_\zeta\zeta^2}{\delta_{min}} + \sum_{i=1}^n p_i^2\sigma^2\right),\end{aligned} \tag{11}$$

where $\Psi_t := X_t + aE_t$ with $a = \frac{12\gamma^3 nL}{\delta_{min}}$.

Proof. For smooth functions, we bring Eq. 6 and 8 into the right-hand side of $\Xi_{t+1} := \tilde{F}_{t+1} + bE_{t+1}$.

For convex functions, we bring Eq. 7 and 8 into the right-hand side of $\Psi_{t+1} := X_{t+1} + aE_{t+1}$, thus completing the whole proof.

For the non-convex function, we bring Eq. 10 into Appendix F. Lemma 27 of [10], then **Theorem 1** is proven. For the convex function with $\mu = 0$, we get Eq. 11 into Appendix F. Lemma 27 of [10], then **Theorem 2** is proven. For the strong convex function with $\mu > 0$, we get Eq. 11 into Appendix F. Lemma 25 of [10], then **Theorem 3** is proven.

G. Proof of *Theorem 4*

Note that the original problem is equivalent to finding the local optimal solution of $\Phi(\delta_1, \ldots, \delta_n) = \frac{\frac{p_1}{\sqrt{\delta_1}} + \ldots + \frac{p_n}{\sqrt{\delta_n}}}{\sqrt{\delta_{min}}}$ under the constraint $\sum_{i=1}^n \delta_i = n\bar{\delta}$ and $\delta_i > 0, \forall i \in \{1, \ldots, n\}$. We divide the proof of **Theorem 4** into two steps. First, the n-variable optimization problem with one constraint is transformed into a one-variable optimization problem (by Eq. 12). Second, the minimum of the one-variable optimization problem is solved.

Lemma 4. Suppose that $a_i, b_i > 0, \forall i \in \{1, \ldots, n\}$ with $\sum_{i=1}^n a_i = A$ (A is a constant), b_i are constants, we have

$$\sum_{i=1}^n \frac{b_i}{\sqrt{a_i}} \geq A^{-\frac{1}{2}}(\sum_{i=1}^n b_i^{\frac{2}{3}})^{\frac{3}{2}}. \tag{12}$$

The inequality takes equal if $a_i = Ab_i^{\frac{2}{3}}(\sum_{i=1}^n b_i^{\frac{2}{3}})^{-1}$.

Proof. With the equality constrain on a_i, we define a Lagrangian function as follows:

$$\mathbb{L} = \sum_{i=1}^n \frac{b_i}{\sqrt{a_i}} + \lambda(\sum_{i=1}^n a_i - A).$$

By the optimality condition, we have

$$\begin{cases} \frac{\partial\mathbb{L}}{\partial a_i} &= -\frac{1}{2}b_i a_i^{-\frac{3}{2}} + \lambda = 0, \forall i \in \{1, \ldots, n\} \\ \frac{\partial\mathbb{L}}{\partial\lambda} &= \sum_{i=1}^n a_i - A = 0 \end{cases}.$$

By solving system of equations above, we can obtain the desired result.

With **Lemma 4**, we can convert $\Phi(\delta_1, \ldots, \delta_n)$ into a one-dimensional function: We assume that $\delta_{min} = \delta_j \leq$

[4]The inequality follows from the fact that $\|\sum_{i=1}^k a_i\|^2 \leq k\sum_{i=1}^k\|a_i\|^2$.

$\min\{\delta_i\}, i \in \{1,\dots,n\} \setminus \{j\}$ and set $b_i = p_i$ if $i \in [1, j-1]$, otherwise $b_i = p_{i+1}$. We also set $a_i = \delta_i$ and $A = (n\bar{\delta} - \delta_j)$.

$$\Phi(\delta_1,\dots,\delta_n) \geq \frac{p_j}{\delta_j} + \frac{(P - p_j^{\frac{2}{3}})^{\frac{3}{2}}}{\sqrt{(n\bar{\delta} - \delta_j)\delta_j}}, \text{ where } P = \sum_{i=1}^{n} p_i^{\frac{2}{3}}, \tag{13}$$

and Eq. 13 holds true if and only if $\delta_i = (n\bar{\delta} - \delta_j)p_i^{\frac{2}{3}}(P - p_j^{\frac{2}{3}})^{-1}, i \neq j$. Since p_i is sorted in descending order, $\min\{\delta_i\}$ equals δ_n if $j \in \{1,\dots,n-1\}$, otherwise $\min\{\delta_i\}$ equals δ_{n-1}.

Note that the minimum of the right-hand side of Eq. 13 depends on the range of δ_j, we discuss the cases of $j \in [1, n-1]$ and $j = n$, respectively.

• If $j \in [1, n-1]$, we have

$$\delta_{min} = \delta_j = \frac{(n\bar{\delta} - \delta_j)p_n^{\frac{2}{3}}}{P - p_j^{\frac{2}{3}}}.$$

We set $Q_j = \frac{P - p_j^{2/3}}{p_n^{2/3}}, j \in [1, n-1]$ and use $\delta_j \leq \min\{\delta_i\}$. Then we get the range of $\delta_j \in (0, \frac{n\bar{\delta}}{Q_j+1}]$. We set $H(\delta_j) = \frac{p_j}{\delta_j} + \frac{(P - p_j^{\frac{2}{3}})^{\frac{3}{2}}}{\sqrt{(n\bar{\delta} - \delta_j)\delta_j}}$ and derive the derivative for $H(\delta_j)$:

$$H'(\delta_j) = -p_j\delta_j^{-2} - \frac{1}{2}(P - p_j^{\frac{2}{3}})^{\frac{3}{2}}[(n\bar{\delta} - \delta_j)\delta_j]^{-\frac{3}{2}}(n\bar{\delta} - 2\delta_j) < 0.$$

Thus we get the minimum of $H(\delta_j)$ at $\delta_j = \frac{n\bar{\delta}}{Q_j+1}$:

$$H(\delta_j) \geq H(\frac{n\bar{\delta}}{Q_j+1}) = \frac{1}{n\bar{\delta}}(p_j(1+Q_j) + p_nQ_j(1+Q_j)). \tag{14}$$

We combine Eq. 13 and 14 and complete the first case ($j \neq n$) in the proof.

• If $j = n$, we set $Q_n = \frac{P - p_n^{2/3}}{p_{n-1}^{2/3}}$ and have

$$\min\{\delta_i\} = \delta_{n-1} = \frac{n\bar{\delta} - \delta_n}{Q_n}.$$

We get the range of δ_n is $(0, \frac{n\bar{\delta}}{Q_n+1}]$. In this range, $H'(\delta_j) < 0$ (the proof process is the same as $j \neq n$). We have

$$H(\delta_j) \geq H(\frac{n\bar{\delta}}{Q_n+1}) = \frac{1}{n\bar{\delta}}(p_n(1+Q_n) + p_{n-1}Q_n(1+Q_n)). \tag{15}$$

We combine Eq. 13 and 15 and complete the second case ($j = n$) in the proof.

V. Evalutaion Experiments

This evaluation answers the following questions:

• Does DAGC outperform uniform compression in real-world datasets, if the total compression ratio is limited and fixed? (Fig. 3 in Sec. V-B)

• Will DAGC perform better as the size distribution becomes more imbalanced and the compression becomes more aggrassively? (Table II in Sec. V-C)

We demonstrate the faster convergence of DAGC in both real-world Non-IID and artificially partitioned Non-IID datasets, especially facing the highly imbalanced size distribution and restricted communication.

A. Experimental settings

Environment: Our experiments are implemented on a server running on Ubuntu 18.04.6 LTS system, which is equipped with an Intel Xeon Silver 4210 CPU @2.20GHz and 4 Nvidia GeForce GTX 3090 with 24GB memory. The Python version is 3.8.12, and other used libraries are all based on the Python version. We use PyTorch 1.11.0 with CUDA 11.3 as the ML toolkit.

Non-IID type: We run the experiments in two Non-IID types:

• *Artificial Non-IID data partition:* To simulate the label imbalance, we allocate a proportion of the samples of each label obeying the Dirichlet distribution to each worker and set the concentration parameter to 0.5. This partitioning strategy to generate Non-IID is broadly used [30]–[32].

• *Real-world datasets:* We use Flickr [3] as the real-world dataset. We download the datasets from `https://doi.org/10.5281/zenodo.3676081`, and divide the images according to the subcontinent they belong to. Excluding damaged images and network limitations that prevented downloading images, we get a total of 15 workers, and the data distribution is shown in Fig. 3(a).

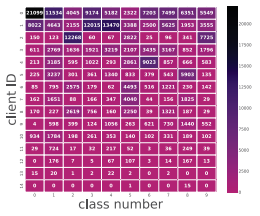
Experiment tasks: Table I lists the four types of experimental settings used in this work, with tasks for both image classification and speech recognition. The CNN used here is a convolution neural network with four layers referring to [6]. The VGG-11s is a simplified version of VGG-11 [33] from [34]. The LSTM has 2 hidden layers of size 128. All these models remove the batch normalization layer to alleviate the accuracy loss caused by Non-IID [3]. The dataset for the Speech Recognition is Speech commands [27] (abbreviated as SCs). We select the 10 categories with the largest number of samples in SCs, and $4,000$ samples are taken from each of these categories, $3,000$ samples as training set and the remaining $1,000$ samples as test set.

Baselines: We compare DAGC with Top-δ[5] and ACCORDION [35]. Top-δ represents transmitting the δ largest value of gradients (in absolute value), which is a relative gradient compressor widely studied in the field of (adaptive) gradient compression [11], [19]. ACCORDION is the state-of-the-art sparsified adaptive gradient compression algorithm, which compresses the gradient using either Top-δ_{min} or Top-δ_{max}. If the training is in the critical regime, ACCORDION uses Top-δ_{min}. Otherwise, ACCORDION uses Top-δ_{max}. For comparison, we define the value of δ_{max} (as well as δ_{min}) in ACCORDION as equal to the maximal (minimal) δ_i in DAGC. Both baselines adopt the uniform compression strategy.

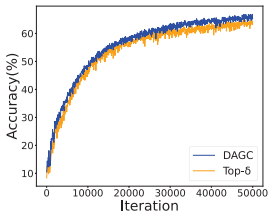
[5]This algorithm is also written as 'Top-k' in other literature [18], [25].

TABLE I: Summary of the experiment settings used in this work.

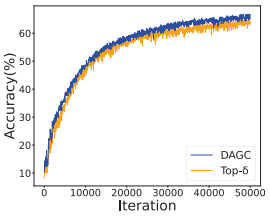
Task	Model	Dataset	Non-IID type	Quality metric	Training iterations	Experiment Section
Image Classification	Logistic	FMNIST [26]	Artificially partitioned Non-IID	Top-1 Accuracy	5000	Sec. III
	CNN	CIFAR-10 [29]	Artificially partitioned Non-IID		$20,000$	Sec. V-B
	VGG11s	Flickr [3]	Real-world Non-IID		$50,000$	Sec. V-B
Speech Recognition	LSTM	SCs [27]	Artificially partitioned Non-IID	Top-1 Accuracy	$10,000$	Sec. III, V-B



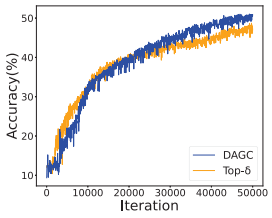
(a) The label distribution for Flickr



(b) $\bar{\delta} = 10\%$



(c) $\bar{\delta} = 1\%$



(d) $\bar{\delta} = 0.1\%$

Fig. 3: The label distribution for Flickr (a) and convergence rate when using different compression algorithms during the training of VGG11s on Flickr with the average compression ratio $\bar{\delta} = 10\%$ (b), 1% (c), 0.1% (d). DAGC outperforms Top-δ with fixed and limited communication.

The number of workers and the worker size: This setting is for artificial Non-IID data partition only. We set the number of workers equal to 10. The *worker size* does not take a dichotomous division (used in Sec. III for simplicity). p_i is an arithmetic series. In order to increase the randomness of the series and to closely match the real-world datasets, we add a Dirichlet distribution (the concentration parameter is 0.5) to p_i, $i \in [2, n-1]$, making it an approximate arithmetic series, still in descending order. This refers to generating artificially Non-IID datasets with different worker sizes in Federated Learning [31], [36]. We define the skew ratio (abbrived as SR) as p_1/p_n, to measure the imbalance of datasets.

B. Real-world Non-IID Scenarios

Our experimental results show that the training results of DAGC outperform Top-δ with the uniform compression in real-world scenarios with fixed communication volumes.

Fig. 3(a) shows the data distribution of Flickr [3] (divided by subcontinent), where the skew ratio is $4,997$ ($\approx \frac{79958}{16}$). We take out 10 categories with the largest number of images. Then, we take these 10 kinds of images from these 15 workers as the training dataset.

TABLE II: Accuracy of different gradient compression algorithms under different SR and average compression ratios $\bar{\delta}$. SR increasing corresponds to the size distribution of datasets becoming more imbalanced. $\bar{\delta}$ measures the degree of restricted communication. The results show that DAGC outperforms Top-δ in both Image Classification and Speech Recognition tasks especially when the size distribution of workers is highly imbalanced and the communication is restricted.

Model @Dataset	SR	$\bar{\delta}$	**DAGC**	Top-δ	ACCORDION
CNN @CIFAR-10	10	10%	73.12%	**73.37%**	73.17%
		1%	71.56%	**71.97%**	71.10%
		0.1%	**67.40%**	66.86%	62.50%
	100	10%	72.73%	**73.47%**	72.64%
		1%	**72.18%**	72.12%	71.41%
		0.1%	**67.32%**	66.04%	62.50%
	$1,000$	10%	**72.60%**	72.31%	71.84%
		1%	**71.84%**	71.61%	69.92%
		0.1%	**67.10%**	65.84%	62.24%
LSTM @SCs	10	10%	75.52%	75.52%	**75.57%**
		1%	73.46%	**73.66%**	72.33%
		0.1%	**59.94%**	58.83%	53.71%
	100	10%	**75.11%**	74.33%	74.32%
		1%	73.49%	**74.12%**	70.62%
		0.1%	**59.55%**	57.01%	52.56%
	$1,000$	10%	**74.84%**	74.42%	74.80%
		1%	**73.12%**	72.57%	70.92%
		0.1%	**60.56%**	57.78%	53.53%

Fig. 3(b) shows that DAGC achieves a faster convergence rate than Top-δ, with the average compression $\bar{\delta} = 10\%$. To converge to the same accuracy (64%), DAGC can reduce up to 23.24% of the number of iterations (from $41,480$ iterations to $31,840$ iterations) compared to the uniform compression.

Fig. 3(c) shows Top-δ converges faster in the early iterations but is overtaken by DAGC in the late iterations, with $\bar{\delta} = 1\%$. DAGC lags behind Top-δ in the early stage probably because of the loss fluctuation. In the later stage, DAGC shows superior performance compared to uniform compression, thus achieving the reversal. DAGC can save up to 26.19% of training time (from $37,880$ iterations to $27,960$ iterations) with the same accuracy (43%) in the later stages compared to uniform compression.

The curve in Fig. 3(d) fluctuates considerably due to too aggressive compression slowing down the model converging. Even so, DAGC still outperforms uniform compression in most of the iterations.

Overall, DAGC surpasses the uniform compression in the normal compression interval with fixed communication volumes, both in the aggressive and conservative compression intervals.

C. Artificially Partitioned Non-IID Scenarios

Our experimental results show that DAGC achieves better performance than both Top-δ and ACCORDION under the datasets with a high skew ratio. The specific experimental results are shown in Table II.

Comparison among different skew ratios: With increasing skew ratios, the advantage of DAGC becomes more obvious in both tasks. For a skew ratio of 10, the performance of DAGC is not significantly different from Top-δ and ACCORDION in both tasks, except when $\delta = 0.1\%$. But when SR ≥ 100, DAGC achieves higher accuracy than Top-δ after the same number of iterations.

DAGC performs better when the skew ratio is large, because the reduction of the dominant term Φ is more obvious. When the skew ratio is small (equal to or less than 10), the differences between DAGC and Top-δ in the compression ratio setting and the value of Φ are extremely small, so there is almost no difference in the accuracy between DAGC and Top-δ. But when the skew ratio is larger, the compression setting of DAGC differs greatly from Top-δ, and the reduction of the dominant term is effectively reflected in the rate of model convergence. This also explains the superiority of DAGC on VGG11s@Flickr, as the skew ratio is nearly $5,000$.

Comparison among different $\bar{\delta}$: DAGC converges faster than Top-δ and ACCORDION with lower $\bar{\delta}$. In CNN@CIAFR-10, DAGC becomes more superior to uniform algorithms as $\bar{\delta}$ decreases, when SR is fixed. In LSTM@SCs, when the skew ratio is $1,000$, the same trend occurs. ACCORDION performs poorly when $\bar{\delta} = 0.1\%$, regardless of the skew ratio and the task. These results show that DAGC is suitable for highly constrained communication conditions.

Experimental results are consistent with the theoretical analysis. As $\bar{\delta}$ becomes smaller, the greater the impact of the dominant term Φ having on the convergence rate, which fits better with **Assumption*** in Sec. IV-C. So DAGC performs well with limited and fixed communication.

In summary, DAGC performs better on both real-world Non-IID and aritificially partitioned Non-IID datasets with highly imbalanced size distribution and limited communication, which is consistent with our theoretical analysis.

VI. Related Works

Adaptive gradient compression can adaptively adjust compression parameters used in traditional algorithms [37], [38]. improving their robustness in specific scenarios such as dynamic network conditions and data heterogeneity. The work [35] proposed ACCORDION to identify the current training stage, improving the model quality. DC2, a network latency-based compression control system, was proposed in the work [1] to enable model training to complete on time in a dynamic network environment. The work [3] proposed SkewScout to make algorithms robust to Non-IID scenes. This system-level approach dynamically adjusts the compression ratio based on the loss difference between workers, which is actually difficult to measure. This makes SkewScout hard to implement.

Improving algorithm robustness: Some works try to address the Non-IID quagmire by considering theoretical compression analysis, which can reduce the reliance on the data heterogeneity [22]. The work [10] compared Q-SGD and D-EF-SGD in Non-IID scenarios and added bias correction to them for improving their data-depencence. The work [19] theoretical analysed the robustness of the hard threshold sparsification algorithm, which only sends absolute gradient values above a constant hard threshold, and found it to be more robust to the Non-IID problem compared to Top-δ.

Data-aware methods: There has been work proposing data-aware node selection in distributed ML. The work [39] proposed a method based on data volumes to select workers in Federated Learning, which differs from gradient-based methods [40]. This work experimentally concludes that the data-volume-based node selection method outperforms the uniform selection strategy in Non-IID scenarios, suggesting that allowing *large workers* to convey more information may be beneficial. However, there has been no work proposing data-aware algorithms in the field of gradient compression. In this work, we propose a data-aware gradient compression algorithm and give the corresponding theoretical analysis.

VII. Conclusion

In this paper, we propose a novel gradient compression algorithm to be more robust to Non-IID datasets from a new perspective, *i.e.*, the non-uniformity of the data size. Firstly, we empirically demonstrate a faster convergence when workers with larger data volumes are set higher compression ratios. Secondly, we derive the convergence rate of D-EF-SGD with a relative compressor and different compression ratios, which is a linear slow-down at the dominant term Φ. We propose DAGC, where sets $\frac{\delta_i}{\delta_j} \approx (\frac{p_i}{p_j})^{2/3}$, by solving the minimum of Φ. DAGC can achieve a faster convergence rate than the uniform compresssion in a communication-constraint Non-IID environment, suggesting that we should let the large worker transfer more information in the communication optimization problems in Non-IID scenarios. We evaluate DAGC on both the real-world and artificially partitioned Non-IID datasets. The evaluation experiments show that DAGC can save up to 26.19% of iterations on Flickr and improve the accuracy by 2.78% on artificially partitioned Non-IID datasets.

Acknowledgment

Rongwei Lu would like to thank Xuanyu Zhu, Yinan Mao, Chen Tang, Shuzhao Xie and Yifei Zhu for their help in making this work possible. This work is supported in part by NSFC under Grant 61872215, the Guangdong Basic and Applied Basic Research Foundation under Grant 2021A1515110066, the GXWD 20220811172936001, and Shenzhen Science and Technology Program under Grant RCYX20200714114523079 and JCYJ20220818101012025.

REFERENCES

[1] A. M. Abdelmoniem and M. Canini, "Dc2: Delay-aware compression control for distributed machine learning," in *IEEE Conference on Computer Communications 2021*, 2021, pp. 1–10. DOI: 10.1109/INFOCOM42981.2021.9488810.

[2] 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.

[3] K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons, "The non-iid data quagmire of decentralized machine learning," in *International Conference on Machine Learning*, PMLR, 2020, pp. 4387–4398.

[4] K. Hsieh, A. Harlap, N. Vijaykumar, *et al.*, "Gaia:{geo-distributed} machine learning approaching {lan} speeds," in *14th USENIX Symposium on Networked Systems Design and Implementation*, 2017, pp. 629–647.

[5] L. G. Valiant, "A bridging model for parallel computation," *Communications of the ACM*, vol. 33, no. 8, pp. 103–111, 1990.

[6] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[7] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7611–7623, 2020.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[9] T. Weyand, A. Araujo, B. Cao, and J. Sim, "Google landmarks dataset v2 - a large-scale benchmark for instance-level recognition and retrieval," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020.

[10] S. U. Stich, "On communication compression for distributed optimization on heterogeneous data," *arXiv preprint arXiv:2009.02388*, 2020.

[11] S. U. Stich and S. P. Karimireddy, "The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates," *Journal of Machine Learning Research*, vol. 21, pp. 1–36, 2020.

[12] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified sgd with memory," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[13] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Transactions on Neural Networks and Learning systems*, vol. 31, no. 9, pp. 3400–3413, 2019.

[14] H. Xu, C.-Y. Ho, A. M. Abdelmoniem, *et al.*, "Grace: A compressed communication framework for distributed machine learning," in *IEEE International Conference on Distributed Computing Systems*, 2021.

[15] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "Signsgd: Compressed optimisation for non-convex problems," in *International Conference on Machine Learning*, PMLR, 2018, pp. 560–569.

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

[17] D. Basu, D. Data, C. Karakus, and S. Diggavi, "Qsparse-local-sgd: Distributed sgd with quantization, sparsification and local computations," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli, "The convergence of sparsified gradient methods," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[19] A. Sahu, A. Dutta, A. M Abdelmoniem, T. Banerjee, M. Canini, and P. Kalnis, "Rethinking gradient sparsification as total error minimization," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

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

[21] L. Cui, X. Su, Y. Zhou, and Y. Pan, "Slashing communication traffic in federated learning by transmitting clustered model updates," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2572–2589, 2021.

[22] M. M. Amiri, D. Gunduz, S. R. Kulkarni, and H. V. Poor, "Federated Learning With Quantized Global Model Updates," *arXiv e-prints*, arXiv:2006.10672, arXiv:2006.10672, Jun. 2020. arXiv: 2006.10672 [cs.IT].

[23] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "SignSGD: Compressed optimisation for non-convex problems," in *International Conference on Machine Learning*, J. Dy and A. Krause, Eds., ser. Machine Learning Research, vol. 80, PMLR, Oct. 2018, pp. 560–569. [Online]. Available: https://proceedings.mlr.press/v80/bernstein18a.html.

[24] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "Qsgd: Communication-efficient sgd via gradient quantization and encoding," *Advances in Neural Information Processing Systems*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/6c340f25839e6acdc73414517203f5f0-Paper.pdf.

[25] A. F. Aji and K. Heafield, "Sparse communication for distributed gradient descent," *arXiv preprint arXiv:1704.05021*, 2017.

[26] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[27] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[28] S. P. Karimireddy, Q. Rebjock, S. Stich, and M. Jaggi, "Error feedback fixes signsgd and other gradient compression schemes," in *International Conference on Machine Learning*, PMLR, 2019, pp. 3252–3261.

[29] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," 2009.

[30] Q. Li, B. He, and D. Song, "Practical one-shot federated learning for cross-silo setting," *arXiv preprint arXiv:2010.01017*, 2020.

[31] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," in *IEEE International Conference on Data Engineering*, 2022.

[32] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," *arXiv preprint arXiv:2002.06440*, 2020.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[34] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Sparse binary compression: Towards distributed deep learning with minimal communication," in *International Joint Conference on Neural Networks*, IEEE, 2019, pp. 1–8.

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

[36] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7611–7623, 2020.

[37] J. Guo, W. Liu, W. Wang, *et al.*, "Accelerating distributed deep learning by adaptive gradient quantization," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2020, pp. 1603–1607.

[38] J. Wang and G. Joshi, "Adaptive communication strategies to achieve the best error-runtime trade-off in local-update sgd," *Machine Learning and Systems*, vol. 1, pp. 212–229, 2019.

[39] C. Dupuy, T. G. Roosta, L. Long, C. Chung, R. Gupta, and S. Avestimehr, "Learnings from federated learning in the real world," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022. [Online]. Available: https://www.amazon.science/publications/learnings-from-federated-learning-in-the-real-world.

[40] H. Wu and P. Wang, "Node selection toward faster convergence for federated learning on non-iid data," *IEEE Transactions on Network Science and Engineering*, 2022.