Wyner-Ziv compression is (almost) optimal for distributed optimization

Prathamesh Mayekar[†] [*] Shubham K Jha[*] Himanshu Tyagi[*]

***Abstract*—Consider distributed optimization of smooth convex functions over $\mathbb{R}^d$ where K independent clients can provide estimates of the gradient. Assume that all the gradient estimates are within Euclidean distance σ of the true gradient and that each oracle's output must be compressed to r bits. For this problem, in the centralized setting with one client, the optimal convergence rate using T iterations is known to be roughly $\sqrt{\sigma^2/T}$. We show that in the distributed setting the optimal convergence rate for large K is roughly $\sqrt{\sigma^2/T}\cdot\sqrt{d/Kr}$. Our main contribution is an algorithm which attains this rate by exploiting the fact that the gradient estimates are close to each other. Specifically, our gradient compression scheme first uses half of the parties to form side information and then uses a Wyner-Ziv compression scheme to compress the remaining half of the gradient estimates.**

I. Introduction

In large scale machine learning or federated learning, data is not available at a single processor or location and models are trained by getting gradient estimates from remote clients. In this setting, the gradients are quantized to few bits to reduce the communication delays, which often can become the performance bottleneck. To circumvent this bottleneck, several gradient compression schemes have been proposed (see, for instance, [3]–[5], [9]–[11], [13], [14], [20], [22]–[24], [28]–[31], [34], [35]). In a related different direction, [7], [17], [32] focused on the problem of distributed mean estimation, which is a common primitive used in both distributed optimization and federated learning.

The quantizers used in most of these prior works are for compressing high-dimensional vectors. In a recent thread, an interesting setup has been considered when some "side-information" about the gradients is available at the decoder. Specifically, for distributed mean estimation, [8] and [22] presented several compression schemes when side information is used for decoding each sample vector. [18], [19] build upon some of the ideas presented in these works to exploit the correlation across different clients (spatial) as well as historical gradient data (temporal) to design efficient compression schemes in federated learning. [15], too, propose exploiting spatial and temporal correlations for gradient compression in federated learning.

These schemes are reminiscent of Wyner-Ziv compression in information theory (see, for instance [25], [36], [37]) ; we use this term broadly in the current paper to indicate compression schemes for vectors when the decoder has another vector that is close to the vector being compressed. In this work, we exhibit the role of Wyner-Ziv compression for gradient compression in distributed optimization when the data across different clients is similar, and thereby the gradient updates of different clients are close. At a high-level, our result shows that Wyner-Ziv compression schemes can allow us to exploit this closeness of gradient updates to communicate less and get nearly optimal convergence rates.

Specifically, we consider the setting where a central server can make gradient queries about an unknown smooth convex function over $\mathbb{R}^d$ to K clients each of which have gradients estimates within bounded Euclidean distance σ of the true gradient. The clients can only send r-bits about their gradient estimates. We first show that the error after T iterations of any such algorithm must be at least $\Omega(\sqrt{\sigma^2 d/KrT})$. We then present a scheme that attains this bound for a large Kr setting (though r can be small). In this scheme, we quantize and send gradient estimates from $K/2$ clients to form a preliminary estimate, and then apply a Wyner-Ziv compression scheme to send the gradient estimates from the remaining $K/2$ clients treating the preliminary estimate as side-information. Technically, to apply our Wyner-Ziv scheme, we need to ensure that the preliminary estimate has a subgaussian error with appropriately small variance parameter, which is a more stringent requirement than the expected mean-squared loss needed in prior work.

The rest of the paper is organised as follows. We set up the problem in next section and discuss preliminaries containing the lower bound. We then provide our main result and our scheme in Section III .The analysis of our scheme is provided in Section IV.

II. Setup and Preliminaries

We consider the problem of minimizing an unknown convex function $f\colon \mathcal{X}\to\mathbb{R}$ over its domain $\mathcal{X}\subset\mathbb{R}^d$ using a set of K *clients* who have access to independent noisy gradients of the function. In particular, the optimization algorithm is not directly given access to the function but can get K different gradient estimates of the function at various points of its choice. This class of optimization algorithms includes various descent algorithms, which provide close to optimal convergence rate within the class and are appealing in practice due to their distributed nature.

In our setup, the gradient estimates supplied by the K clients must pass through r-bit quantizers, chosen from a

This work was supported by a grant from Robert Bosch Center for Cyber Physical Systems, Indian Institute of Science, and a grant on Security and Privacy for Smart Cities sponsored by National Security Council, India.

[†]Indian Urban Data Exchange [*]Indian Institute of Science
Email: {prathamesh, shubhamkj, htyagi}@iisc.ac.in

fixed set of quantizers $\mathcal{Q}_r$ [1], and the optimization algorithm π only has access to the quantized outputs. An *r-bit quantizer* consists of randomized mappings (Q^e, Q^d) with the encoder mapping $Q^e : \mathbb{R}^d \to \{0,1\}^r$ and the decoder mapping $Q^d : \{0,1\}^r \to \mathbb{R}^d$. We denote the overall quantization procedure by the composition mapping $Q = Q^d \circ Q^e$.

Our objective is to select quantizers $Q_{i,t}, \forall i \in [K], t \in [T]$, and an optimization algorithm π to guarantee the minimum worst-case optimization error. In our setting, we allow for *adaptive gradient processing,* whereby, the quantizer $Q_{i,t}$ selected in tth iteration may depend on all the previous quantized outputs. Specifically, denoting by $Y_{i,t}$ the ith client's quantized output at time t, the *adaptive quantizer selection strategy* $S := (S_1, \ldots, S_T)$ over T iterations consists of mappings $S_t : \mathbb{R}^{dK(t-1)} \to \mathcal{Q}_r^K$ that take $\{Y_{i,t'}\}_{i\in[K],t'\in[t-1]}$ as input and outputs a tuple of K quantizers $\{Q_{i,t}\}_{i\in[K]} \in \mathcal{Q}_r^K$. We write $\mathcal{S}_{\mathcal{Q}_r,T}$ for the collection of all such quantizer selection strategies. The entire framework can be summarized as follows:

1) At iteration t, the first-order optimization algorithm π makes a query for point x_t to clients $C_1, \ldots C_K$.
2) Upon receiving the point $x_t \in \mathcal{X}$, the client $i \in [K]$ outputs $\hat{g}_i(x_t)$, an unbiased estimate of $\nabla f(x_t)$.
3) The gradient estimate $\hat{g}_i(x_t)$ is passed through a quantizer $Q_{i,t} \in \mathcal{Q}_r$ chosen based on strategy S, and the output $Y_{i,t}$ is observed by the first-order optimization algorithm π. The algorithm then uses all the messages $\{Q_{i,t'}(x_{t'})\}_{i\in[K],t'\in[t]}$ to further update x_t to x_{t+1}.

Denote by $\mathcal{C}$ the collection of K clients $(C_1, \ldots, C_K)$. Let Π_T be the set of all first-order optimization algorithms that make T queries to $\mathcal{C}$ and for the tth query x_t, get back the outputs $\{Y_{i,t}\}_{i\in[K]}$. We measure the performance of an optimization protocol π and a quantizer selection strategy S for a given function f and clients C_i, $i \in [K]$, using the metric $\mathcal{E}(f, \mathcal{C}, \pi, S)$ defined as

$$\mathcal{E}(f, \mathcal{C}, \pi, S) = \mathbb{E}\left[f(\bar{x}_T) - \min_{x\in\mathcal{X}} f(x)\right],$$

where $\bar{x}_T := \frac{1}{T}\sum_{t\in[T]} x_t$ and the expectation is over the randomness in $\bar{x}_T$.

For a set of various function and client pairs above, denoted by $\mathcal{O}$, the set of r-bit quantizers $\mathcal{Q}_r$ and the number of iterations T, we define the *minimax optimization error* as

$$\mathcal{E}^*(\mathcal{X}, \mathcal{O}, T, \mathcal{Q}_r) = \inf_{\pi\in\Pi_T} \inf_{S\in\mathcal{S}_{\mathcal{Q}_r,T}} \sup_{(f,\mathcal{C})\in\mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, S).$$

Below we will describe the class $\mathcal{O}$ of interest to us.

A. Function classes

We now define the class of functions and state the assumptions related to the stochastic clients accessible to the algorithm π.

[1]The set of r-bit quantizers $\mathcal{Q}_r$ is used to model the communication constraints in a distributed setting.

a) Convex and smooth function family: Throughout, we restrict ourselves to convex and L-smooth functions over $\mathcal{X} \subset \mathbb{R}^d$, i.e., functions satisfying, $\forall \lambda \in [0,1], \forall x, y \in \mathbb{R}^d$,

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y), \tag{1}$$
$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2, \tag{2}$$

where $\nabla f(x) \in \mathbb{R}^d$ denotes the gradient of f at input x.

b) Stochastic gradients: We assume that the output $\hat{g}_i(x)$ by client $C_i, 1 \leq i \leq K$, when a point $x \in \mathcal{X}$ is queried satisfies the following conditions:

$$\mathbb{E}[\hat{g}_i(x) \mid x] = \nabla f(x), \quad \text{(unbiased estimates)} \tag{3}$$
$$\|\hat{g}_i(x) - \nabla f(x)\|_2^2 \leq \sigma^2, \text{ (maximum deviation bound)} \tag{4}$$
$$\max_{x\in\mathcal{X}} \|\hat{g}_i(x)\|_2^2 \leq B^2. \quad \text{(a.s. bounded estimate)} \tag{5}$$

Assumption (3) is standard in stochastic optimization literature ($cf.$ [27], [26], [6]. However, it is enough to assume a bound on the variance of stochastic gradients instead of (4) to prove convergence guarantees for smooth stochastic optimization without any communication constraints. The stronger assumption made here is to aid a much tighter analysis under communication constraints. In Section III-A, we provide a scheme which can operate under the standard variance bound.

Denote by $\mathcal{O}_{\mathtt{sc}}$ the set of tuples of function and K clients, $(f, \mathcal{C})$, satisfying (1), (2), (3), (4) and (5).

B. Lower bound

Before proceeding further, the following bound will serve as a basic benchmark for our problem. Let $D > 0$ and $\mathbb{X}_2(D) := \{\mathcal{X} \subseteq \mathbb{R}^d : \max_{x,y\in\mathcal{X}} \|x - y\|_2 \leq D\}$ be the collection of subsets of $\mathbb{R}^d$ whose ℓ_2 diameter is at most D.

Theorem II.1. *There exists an absolute constant $0 \leq c_0 \leq 1$ such that for $r \in \mathbb{N}$ and $T \geq d/(6Kr)$,*

$$\sup_{\mathcal{X}\in\mathbb{X}_2(D)} \mathcal{E}^*(\mathcal{X}, \mathcal{O}_{\mathtt{sc}}, T, \mathcal{Q}_r) \geq \frac{c_0 D\sigma}{\sqrt{KT}} \cdot \sqrt{\frac{d}{d \wedge r}}.$$

Note that affine function are 0-smooth and admitted in the class of L smooth functions. We use affine functions as difficult functions and proceed in the same manner as in the lower bounds for convex, Lipschitz optimization under communication constraints ([1, Section 4.5]; see, also, [2]), since the lower bounds for convex Lipschitz optimization also use affine functions as difficult functions.

C. A general convergence bound

We present a general convergence bound based on a non-adaptive channel strategy. In particular, we fix same quantization process in every iteration, and the quantized outputs $\{Y_{i,t}\}_{i\in[K]}$ are passed through a mapping $Q : \mathbb{R}^{Kd} \to \mathbb{R}^d$ in order to update the query.

We use PSGD as the first-order optimization algorithm; the overall optimization procedure is described in Algorithm 1. PSGD proceeds as SGD, with the additional projection step where it projects the updates back to domain $\mathcal{X}$ using the map $\Gamma_{\mathcal{X}}(y) := \min_{x\in\mathcal{X}} \|x - y\|, \forall y \in \mathbb{R}^d$.

1: **for** $t=0$ to $T-1$ **do**
2: $\quad x_{t+1}=\Gamma_{\mathcal{X}}\left(x_t-\eta_t Q(Y_{1,t},...,Y_{K,t})\right)$
3: Output $\bar{x}_T=\frac{1}{T}\sum_{t=1}^{T}x_t$

Algorithm 1: PSGD with quantizers $\{Q_i\}_{i\in[K]}$

The convergence rate of Algorithm 1 is controlled by the worst-case L_2-norm $\alpha(Q)$ and the worst-case bias $\beta(Q)$ defined as

$$\alpha(Q):=\sup_{\substack{\{\forall x, i\in[K],\hat{g}_i\in\mathbb{R}^d:\\ \|\hat{g}_i-\nabla f(x)\|^2\le\sigma^2\}}}\sqrt{\mathbb{E}\left[\|Q(\bar{Y})-\nabla f(x)\|^2\right]},\tag{6}$$

$$\beta(Q):=\sup_{\substack{\{\forall x, i\in[K],\hat{g}_i\in\mathbb{R}^d:\\ \|\hat{g}_i-\nabla f(x)\|^2\le\sigma^2\}}}\|\mathbb{E}\left[(Q(\bar{Y})-\nabla f(x)\right]\|,\tag{7}$$

where for $i\in[K], \bar{Y}=(Y_{1,t},...,Y_{K,t})$. Using a slight modification of the standard proof of convergence for PSGD, we can derive the following lemma.

Lemma II.2. *For any mapping Q and set of quantizers $\{Q_i\}_{i\in[K]}$ defined above, the output $\bar{x}_T$ of optimization algorithm given in Algorithm 1 satisfies*

$$\sup_{(f,\mathcal{C})\in\mathcal{O}}\mathcal{E}(f,\mathcal{C},\pi,S)$$
$$\le\frac{\sqrt{2}\alpha(Q)D}{\sqrt{T}}+\beta(Q)\left(D+\frac{DB}{\alpha(Q)\sqrt{2T}}\right)+\frac{LD^2}{2T},$$

with the learning rate $\eta_t=\min\{\frac{1}{L},\frac{D}{\alpha(Q)\sqrt{2T}}\},\forall t\in[T.]$

D. Sub-gaussian norm and random rotation

For our analysis, it will be convenient to recall the definition of subgaussian norm[2] of a random variable.

Definition II.3 ([33])**.** A subgaussian norm of a subgaussian random variable X, denoted $\|X\|_{\psi_2}$, is defined as $\|X\|_{\psi_2}:=\inf\{t>0:\mathbb{E}\left[e^{X^2/t^2}\right]\le 2\}$. It follows that for a centered subgaussian random variable X, $\Pr(|X|)\ge t)\le 2e^{-\frac{t^2}{\|X\|^2_{\psi_2}}}$.

In addition to this, we require a random Hadamard matrix to perform random rotation. Specifically, denoting by H the $d\times d$ Walsh-Hadamard Matrix (See [12]), define

$$R:=1/\sqrt{d}HD',\tag{8}$$

where D' is a diagonal matrix with each diagonal entry generated uniformly from $\{-1,+1\}$.

III. Main Result: An optimal upper bound for distributed optimization

A. Baseline: `Parallel SGD`

We begin by presenting the convergence result for the baseline scheme in our setup: the `Parallel SGD` algorithm. In `Parallel SGD`, all clients compress their stochastic gradient estimates to r bits using an efficient quantizer for the Euclidean ball and send it to the server, which then takes the average of the quantized gradients for the projected gradient descent step. We choose subsampled RATQ ([24]) for this efficient quantizer. We denote by $Q_{\texttt{RATQ}}$ the subsampled version of RATQ using r bits, which is described in [24, Section 3.5]. Denote by Q_t the average of all the quantized gradients, i.e.,

$$Q_t=\frac{1}{K}\sum_{i=1}^{K}Q_{\texttt{RATQ}}(\hat{g}_j(x_t)).\tag{9}$$

We use Q_t to make the projected descent step as seen in line 2 of Algorithm 1.

Theorem III.1. *Let S be the quantizer selection strategy which fixes the quantizer to be $Q_{\texttt{RATQ}}$ for all clients at all iterations. Let π be the optimization algorithm described in Algorithm 1 where Q_t as described in (9) is used to make the PSGD step after the tth query. Then, for universal constants c_1 and c_2, and r such that $d\ge r\ge c_1\log\log^* d$, we have*

$$\mathcal{E}(f,\mathcal{C},\pi,S)\le\frac{c_2D}{\sqrt{KT}}\sqrt{\sigma^2+\frac{c_2dB^2\log\log^* d}{r}}+\frac{LD^2}{T}.$$

We note that the term $\frac{dB^2\log\log^* d}{r}$ illustrates the slowdown in convergence due to quantization error. This is nearly the best rate which can be achieved when one uses r-bit quantizers without any side information[3]. Note that for the cases when B is large relative to σ^2, the slowdown due to this term can be significant, and the algorithm maybe far away from our lower bound in Theorem II.1.

B. `WZ-SGD`*: An almost optimal algorithm for distributed optimization*

We are now ready to present our main algorithm: `WZ-SGD`. `WZ-SGD` significantly improves over the convergence rate of

for Clients $i\in[K]$ **do**
$\quad$ **if** $i\in\mathcal{C}_1$ **then** $Q_i=Q_{\texttt{u}}$
$\quad$ **else** $Q_i=Q_{\texttt{WZ},i}$
Intialize $x_1\in\mathcal{X}$
for $t\in[T]$ **do**
$\quad$ **for** Server **do**
$\quad\quad$ Broadcast x_t to clients
$\quad$ **for** Clients $i\in[K]$ **do** $\qquad$ ▷ Encoding
$\quad\quad$ Compute $\hat{g}_i(x_t)$
$\quad\quad$ Send $Q_i^{\texttt{e}}(\hat{g}_i(x_t))$ to server
$\quad$ **for** Server **do** $\qquad$ ▷ Decoding
$\quad\quad$ **for** $i\in\mathcal{C}_1$ **do**
$\quad\quad\quad$ $Q_i(\hat{g}_i(x_t))=Q_i^{\texttt{d}}(Q_i^{\texttt{e}}(\hat{g}_i(x_t)))$
$\quad\quad$ $Z_t=\frac{2}{K}\sum_{i\in\mathcal{C}_1}Q_i(\hat{g}_i(x_t))$ $\qquad$ ▷ Side-information
$\quad\quad$ **for** $i\in\mathcal{C}_2$ **do**
$\quad\quad\quad$ $Q_i(\hat{g}_i(x_t),Z_t)=Q_i^{\texttt{d}}(Q_i^{\texttt{e}}(\hat{g}_i(x_t)),Z_t)$
$\quad\quad$ $x_{t+1}=\Gamma_{\mathcal{X}}\left(x_t-\eta_t\cdot\frac{2}{K}\sum_{i\in\mathcal{C}_2}Q_i(\hat{g}_i(x_t),Z_t)\right)$
At **Server Output:** $\bar{x}_T=\frac{1}{T}\sum_{t\in[T]}x_{t+1}$

Algorithm 2: `WZ − SGD` algorithm

[2] $\|\cdot\|_{\psi_2}$ is indeed a norm.

[3] Similar convergence bounds (upto $\log\log d$ factor) for parallel SGD can be achieved by using subsampled version of rotated quantizer in [32] or the subsampled version of uniform quantizer after preprocessing due to Kashin's representation ($cf.$ [16], [21]).

Theorem III.1 and relegates the dependence of convergence rate on B to only second order terms.

At each iteration t, $\texttt{WZ-SGD}$ uses the clients in $\mathcal{C}_1$ to form the side information estimate Z_t at the server and then uses the clients in $\mathcal{C}_2$ to estimate the gradient for the gradient descent step, where[4] $\mathcal{C}_1 :=\{C_1, \ldots, C_{K/2}\}, \mathcal{C}_2 :=\mathcal{C} \setminus \mathcal{C}_1$.

a) The side information estimate Z_t: The side information is formed as follows. Under the r-bit communication constraint, we divide the coordinates into blocks of dimension r_1, where $r_1{:=}d/\log \ell_1$, and $\log \ell_1$ denotes the precision bits used by clients to represent each coordinate in the assigned block. This way we have d/r_1 blocks. We assign each block to $Kr_1/(2d)$ clients to form the side information for the coordinates represented by that block. To quantize the coordinates within any block, the clients assigned to that block will use a coordinate-wise uniform quantizer (CUQ). CUQ is an unbiased, uniform quantizer that has appeared recently in many works on gradient quantization. We denote by $Q_{\texttt{u}} : [-B, B] \to \{-B + 2B\cdot(i-1)/(\ell_1-1) : i \in [\ell_1]\}$ the ℓ_1-level CUQ quantizer. For a scalar input $x \in [-B, B]$,

$$Q_{\texttt{u}}(x) = \begin{cases} \left\lceil \frac{x(\ell_1-1)}{2B} \right\rceil \cdot \frac{2B}{\ell_1-1}, & \text{w.p.} \quad \frac{x - \lfloor \frac{x(\ell_1-1)}{2B} \rfloor}{\frac{2B}{\ell_1-1}}, \\ \lfloor \frac{x(\ell_1-1)}{2B} \rfloor \cdot \frac{2B}{\ell_1-1}, & \text{w.p.} \quad \frac{\left\lceil \frac{x(\ell_1-1)}{2B}\right\rceil - x}{\frac{2B}{\ell_1-1}}. \end{cases} \tag{10}$$

Each client uses an ℓ_1-level CUQ to quantize the associated block of coordinates separately. Thus, the overall communication by each client is $r_1 \cdot \log \ell_1 = r$ and satisfies the communication constraint.

For each block, we then form the side information by taking the average of the quantized outputs from all its associated clients. Denote by Z_t the side-information formed at the server by using the clients in $\mathcal{C}_1$ at iteration t. Then, from the description of our scheme, for all coordinates $i \in \{r_1(j-1)+1, \ldots, r_1 j\}$ and for all $j \in [d/r_1]$ we have

$$Z_t(i) = \frac{2d}{Kr_1} \sum_{k \in \mathcal{S}_j} Q_{\texttt{u}}(\hat{g}_k(x_t)(i)),$$

where $\mathcal{S}_j$ denotes the set of $\frac{Kr_1}{2d}$ clients assigned to form the side information for the coordinates $\{r_1(j-1)+1, \ldots, r_1 j\}$, i.e., $\mathcal{S}_j = \{C_{(Kr_1/(2d))\cdot(j-1)+1}, \ldots, C_{(Kr_1/(2d))\cdot j}\}$.

We remark that to decode each quantized gradient estimate sent by clients in $\mathcal{C}_2$, we will use Z_t as side information. However, Z_t will not be used as is but a version which is rotated[5] using a random matrix (8) will be used.

b) The Wyner-Ziv gradient estimate $Q_{\texttt{WZ}}$: We use the clients in $\mathcal{C}_2$ to form the actual gradient estimate. The clients encode the stochastic gradients using a subsampled RMQ quantizer from [22, Section 3.3].

In subsampled RMQ, before compressing the computed gradients, each client preprocesses the stochastic gradients by randomly rotating them using *iid* versions of R given in (8), which in turn is generated using public randomness between the client and the server.

Then, each coordinate of the rotated vector is quantized to $\log \ell_2$ bits using Modulo quantizer (MQ). MQ was recently proposed for distributed mean estimation with side information in [8] and also used in [22]. We follow the description given in [22, Section 3.1]. Specifically, MQ is a uniform quantizer used to quantize input vector $x \in \mathbb{R}$ with side-information y available for decoding. As additional inputs, MQ needs Δ', an estimate on distance between x and y, the precision $\log \ell_2$, and lattice parameter ϵ. The encoder of MQ first randomly quantizes x to either $\lceil x/\epsilon \rceil$ or $\lfloor x/\epsilon \rfloor$ such that the output is an unbiased estimate of x/ϵ. Then, a modulo- k operation is performed on that output and it is sent to the decoder. That is,

$$Q^{\mathtt{e}}_{\texttt{M}}(x) = \tilde{z} \bmod k, \text{where } \tilde{z} = \begin{cases} \lceil x/\varepsilon \rceil, & w.p.\ x/\varepsilon - \lfloor x/\varepsilon \rfloor \\ \lfloor x/\varepsilon \rfloor, & w.p.\ \lceil x/\varepsilon \rceil - x/\varepsilon. \end{cases}$$

The decoder is described as follows.

$$Q_{\texttt{M}}(x, y) = \min\{|(z\ell_2 + Q^{\mathtt{e}}_{\texttt{M}}(x))\varepsilon - y| : z \in \mathbb{Z}\}.$$

Then, each client C_j independently samples a set $\mathcal{D}_j \in [d]$ of cardinality $r_2 := r/\log \ell_2$ uniformly at random. Once again, uniform sampling is done by using the public randomness shared between the client and the server. Only the output of MQ corresponding to coordinates in the set $\mathcal{D}_j$ is sent to the server. Therefore, for stochastic gradient $\hat{g}_j(x_t)$, the output encoded by client C_j using subsampled RMQ is described as follows: $Q^{\mathtt{e}}_{\texttt{WZ},j}(\hat{g}_j(x_t)) = \{Q^{\mathtt{e}}_{\texttt{M}}(R_j\hat{g}_j(x_t)(i)) : i \in \mathcal{D}_j\}$.

At the server, the communication for all $C_j \in \mathcal{C}_2$ is decoded as follows:

$$Q_{\texttt{WZ},j}(\hat{g}_j(x_t), Z_t) {=} R_j^{-1}\left(\frac{d}{r_2}\sum_{i\in\mathcal{D}_j}(\tilde{g}_j - R_j Z_t(i))\, e_i + R_j Z_t\right)$$

where $\tilde{g}_j(i) = Q_{\texttt{M}}(R_j\hat{g}_j(x_t)(i), R_j Z_t(i))$. Finally, the server averages over all the quantized gradient estimates of clients in $\mathcal{C}_2$ to get Q, which in turn is used to make the PSGD step in line 2 of Algorithm 1. That is,

$$Q_t = \frac{\sum_{j=K/2+1}^{K} Q_{\texttt{WZ},j}(\hat{g}_j(x_t), Z_t)}{K/2}. \tag{11}$$

We are now ready to present our main result: the convergence rate of $\texttt{WZ} - \texttt{SGD}$ algorithm.

Theorem III.2. *Let S be the communication protocol which uses the CUQ quantizer for clients C_1 and the subsampled RMQ quantizer for clients in C_2. Let π be the optimization algorithm described in Algorithm 1 which uses Q_t in (11) to make the PSGD step after the tth query. Then, for universal constants c_1, c_2, and c_3 and r, K such that $d \geq r \geq c_1 \max\{\log\log KT, \log(B/\sigma)\}$ and $Kr \geq c_2 d^2 \log(B/\sigma)$, we have*

$$\mathcal{E}(f, \mathcal{C}, \pi, S) \leq \frac{c_3 D\sigma}{\sqrt{KT}} \cdot \sqrt{\frac{d \log\log KT}{r}} + \frac{LD^2}{2T}.$$

Thus, in the setting where the number of clients K is large, we match the lower bound in Theorem II.1 upto a $\log\log KT$ factor.

[4] For simplicity, we assume that $K/2$ and d/r_1 are integers such that d/r_1 divides $K/2$.

[5] For decoding each quantized gradient sent by clients in $\mathcal{C}_2$, Z_t will be rotated using independent and identical versions of matrix R

IV. Analysis of WZ-SGD

a) Side information is close to gradient estimates: We begin by noting that side-information Z[6] is close to the stochastic gradient estimates computed by clients in $\mathcal{C}_2$. Specifically, setting the parameters as $\log \ell_1 = \lceil \log \frac{2B}{\sigma} + 1 \rceil$ and $r_1 = r / \lceil \log 2B/\sigma + 1 \rceil$ for clients in $\mathcal{C}_1$, we get the following.

Lemma IV.1. *For all $x \in \mathbb{R}^d$, $j \in \mathcal{C}_2$, and $i \in [d]$, we have*

$$\Pr(|R\hat{g}_j(x)(i) - RZ(i)| \geq t) \leq 2e^{-c \min\{\frac{t^2}{4\sigma'^2}, \frac{t\sqrt{d}}{2\sigma'}\}} + 2e^{-c\frac{t^2 d}{4\sigma^2}},$$

where R is a random Hadamard matrix (8) and for a universal constant c

$$\sigma'^2 = \frac{c8d\sigma^2 \lceil \log(2B/\sigma + 1) \rceil}{Kr}. \quad (12)$$

Remark 1. In the analysis for RMQ in [22], the difference between the coordinates of the rotated input and rotated side information had subgaussian tails. However, note that in Lemma IV.1, we can only prove a slightly weaker concentration result.

Towards proving Lemma IV.1, we begin by showing the following result which holds from the subgaussian properties of uniform quantizer error and standard properties of subgaussian random variables.

Lemma IV.2. *For all $x \in \mathbb{R}^d$ and $i \in [d]$ we have*

$$\|Z(i) - \nabla f(x)(i)\|_{\psi_2}^2 \leq \sigma'^2.$$

Remark 2. In order to quantize a d-dimensional gradient to $r \leq d$ bits, the technique of uniform sampling has been used in recent papers on distributed optimization (*cf.* [32], [24]). However, this only gives small quantization error in the mean square sense, which will not suffice for our Wyner-Ziv compression algorithm.

Next, using standard properties of subgaussian random variables (*cf.* [33, Lemma 2.7.7 and Theorem 2.8.1]), we can show the following.

Lemma IV.3. *For all $x \in \mathbb{R}^d$ and $i \in [d]$ we have*

$$\Pr(|RZ(i) - R\nabla f(x)(i)| \geq t) \leq 2e^{\left(-c \min\{t^2/\sigma'^2, t\sqrt{d}/\sigma'\}\right)}.$$

Finally, using similar proof techniques as in [24, Lemma 5.8], we can show that the rotated gradient estimates of clients in $\mathcal{C}_2$ is close to the rotation of the true gradient.

Lemma IV.4. *For all $x \in \mathbb{R}^d$ we have*

$$\|R\hat{g}_j(x)(i) - R\nabla f(x)(i)\|_{\psi_2}^2 \leq c\sigma^2/d.$$

Lemma IV.1 follows from Lemmas IV.3 and IV.4.

b) Bounds on $\alpha(Q)$ and $\beta(Q)$: Let the grid size ε of modulo quantizer be set as follows: $\varepsilon = \frac{2\Delta'}{\ell_2 - 2}$, where $\Delta' = \frac{3\sigma}{\sqrt{cd}} \cdot \log(\frac{2\sigma}{\sqrt{c}\delta})$ for some parameter δ to be specified later.

Remark 3. Note that such a choice of ε ensures that whenever a coordinate of the rotated vector $R\hat{g}_j(x)$ is within Δ' of the corresponding coordinate of the rotated side information there is no error in decoding. Therefore, the output of modulo quantizer is unbiased and ε close to input under this event. Also, note that because of the minimum distance decoding at the decoder, each coordinate decoded by modulo quantizer is always $\ell\varepsilon$ close to the side information.

Denote by $Q_{\texttt{RMQ},j}$ the rotated modulo quantizer without any subsampling for client $j \in \mathcal{C}_2$. That is,

$$Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z) = R_j^{-1}\left(\sum_{i \in [d]} (\tilde{g}_j - R_j Z(i))\, e_i + R_j Z\right)$$

where $\tilde{g}_j(i) = Q_{\texttt{M}}(R_j \hat{g}_j(x)(i))$.

The key step of the proof is bounding MSE and bias of RMQ. Towards that, we have the following lemma.

Lemma IV.5. *Under the condition that $Kr \geq c_2 d^2 \log(B/\sigma)$, we have for all $x \in \mathbb{R}^d$, $j \in \mathcal{C}_2$, and $\delta \in (0, 2\sigma/\sqrt{c})$ that*

$$\mathbb{E}\left[\|Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2\right] \leq \frac{144\sigma^2}{c(\ell_2 - 2)^2}\left(\ln \frac{2\sigma}{\sqrt{c}\delta}\right)^2 + 251\delta^2$$

$$\|\mathbb{E}\left[Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z)\right] - \hat{g}_j(x)\|_2^2 \leq 251\delta^2.$$

Proof. By considering events $\{|R_j(\hat{g}_j(x) - Z)(i)| \leq \Delta'\}$ and $\{|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta'\}$, and then using the facts in Remark 3 for modulo quantizer, we have

$$\begin{aligned}
&\mathbb{E}\left[\|Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2\right] \\
&\leq d\varepsilon^2 + \\
&\sum_{i=1}^d \mathbb{E}\left[(Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x))(i)^2 \mathbb{1}_{\{|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta'\}}\right] \\
&\leq d\varepsilon^2 + 2\ell^2\varepsilon^2 \sum_{i=1}^d \Pr(|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta') \\
&+ 2\sum_{i=1}^d \mathbb{E}\left[(R_j(\hat{g}_j(x) - Z)(i))^2 \mathbb{1}_{\{|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta'\}}\right].
\end{aligned}$$

Note that the terms $\Pr(|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta')$ and $\mathbb{E}\left[(R_j(\hat{g}_j(x) - Z)(i))^2 \mathbb{1}_{\{|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta'\}}\right]$ can be bounded appropriately using the concentration bound in Lemma IV.1. Due to space constraints we skip the details.

The bound on bias follows by noting that it is bounded by $\mathbb{E}\left[\|Q_{\texttt{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2 \mathbb{1}_{\{|R_j(\hat{g}_j(x) - Z)(i)| \geq \Delta'\}}\right]$ $\square$

Remark 4. The condition on Kr is needed to remove any B dependence from the MSE upper bound.

Then using standard bounds for subsampling and averaging of vectors (see, for instance, [22, Lemmas 2.1 and 3.3]), we can extend the above result and show the following bounds for $\alpha(Q_t)$ and $\beta(Q_t)$ for $\log \ell_2 = \lceil c \log\log KT \rceil$ and $\delta \leq \frac{2\sigma}{KT}$: $\alpha^2(Q_t) \leq c_1 \frac{\sigma^2 \log\log KT}{K} \cdot \frac{d}{r}, \beta^2(Q_t) \leq \frac{c_2\sigma^2}{KT}$. The convergence proof of Theorem III.2 can be completed by using the bounds on α, β and using Lemma II.2.

[6]For convenience, we drop the iteration subscript t in this Section.

REFERENCES

[1] J. Acharya, C. L. Canonne, P. Mayekar, and H. Tyagi, "Information-constrained optimization: can adaptive processing of gradients help?" *Advances in Neural Information Processing Systems*, 2021.

[2] J. Acharya, C. L. Canonne, Z. Sun, and H. Tyagi, "Unified lower bounds for interactive high-dimensional estimation under information constraints," *http://arxiv.org/abs/2010.06562v5*, 2020.

[3] J. Acharya, C. De Sa, D. J. Foster, and K. Sridharan, "Distributed Learning with Sublinear Communication," *International Conference on Machine Learning*, 2019.

[4] 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*, pp. 1709–1720, 2017.

[5] 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*, 2019.

[6] S. Bubeck, "Convex optimization: Algorithms and complexity," *Foundations and Trends® in Machine Learning*, vol. 8, no. 3-4, pp. 231–357, 2015.

[7] W.-N. Chen, P. Kairouz, and A. Özgür, "Breaking the communication-privacy-accuracy trilemma," *Neural Information Processing Systems (NeurIPS)*, 2020.

[8] P. Davies, V. Gurunathan, N. Moshrefi, S. Ashkboos, and D. Alistarh, "Distributed variance reduction with optimal communication," *arXiv e-prints*, pp. arXiv–2002, 2020.

[9] F. Faghri, I. Tabrizian, I. Markov, D. Alistarh, D. Roy, and A. Ramezani-Kebrya, "Adaptive gradient quantization for data-parallel sgd," *Advances in Neural Information Processing Systems*, 2020.

[10] V. Gandikota, D. Kane, R. Kumar Maity, and A. Mazumdar, "vqsgd: Vector quantized stochastic gradient descent," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research. PMLR, 2021, pp. 2197–2205.

[11] A. Ghosh, R. K. Maity, and A. Mazumdar, "Distributed newton can communicate less and resist byzantine workers," *Advances in Neural Information Processing Systems*, 2020.

[12] K. J. Horadam, *Hadamard matrices and their applications*. Princeton university press, 2012.

[13] Z. Huang, W. Yilei, K. Yi *et al.*, "Optimal sparsity-sensitive bounds for distributed mean estimation," *Advances in Neural Information Processing Systems*, pp. 6371–6381, 2019.

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

[15] D. Jhunjhunwala, A. Mallick, A. H. Gadhikar, S. Kadhe, and G. Joshi, "Leveraging spatial and temporal correlations in sparsified mean estimation," in *Advances in Neural Information Processing Systems*, 2021.

[16] B. Kashin, "Section of some finite-dimensional sets and classes of smooth functions (in russian) izv," *Acad. Nauk. SSSR*, vol. 41, pp. 334–351, 1977.

[17] J. Konečnỳ and P. Richtárik, "Randomized distributed mean estimation: Accuracy vs. communication," *Frontiers in Applied Mathematics and Statistics*, vol. 4, p. 62, 2018.

[18] K. Liang and Y. Wu, "Improved communication efficiency for distributed mean estimation with side information," in *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 3185–3190.

[19] K. Liang, H. Zhong, H. Chen, and Y. Wu, "Wyner-Ziv Gradient Compression for Federated Learning," *https://arxiv.org/abs/2111.08277*, 2021.

[20] C.-Y. Lin, V. Kostina, and B. Hassibi, "Differentially Quantized Gradient Descent," in *IEEE International Symposium on Information Theory (ISIT)*, 2021.

[21] Y. Lyubarskii and R. Vershynin, "Uncertainty principles and vector quantization," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3491–3501, 2010.

[22] P. Mayekar, A. T. Suresh, and H. Tyagi, "Wyner-Ziv estimators: Efficient distributed mean estimation with side-information," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 3502–3510.

[23] P. Mayekar and H. Tyagi, "Limits on gradient compression for stochastic optimization," *Proceedings of the IEEE International Symposium of Information Theory (ISIT' 20)*, 2020.

[24] ——, "RATQ: A universal fixed-length quantizer for stochastic optimization," *IEEE Transactions on Information Theory*, 2020.

[25] N. Merhav and J. Ziv, "On the wyner-ziv problem for individual sequences," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 867–873, 2006.

[26] A. Nemirovsky, "Information-based complexity of convex programming," 1995, Available Online http://www2.isye.gatech.edu/ne-mirovs/Lec_EMCO.pdf.

[27] A. Nemirovsky and D. B. Yudin, "Problem complexity and method efficiency in optimization." *Wiley series in Discrete Mathematics and Optimization*, 1983.

[28] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2021–2031.

[29] R. Saha, S. Rini, M. Rao, and A. Goldsmith, "Decentralized optimization over noisy, rate-constrained networks: How we agree by talking about how we disagree," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5055–5059.

[30] 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," *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

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

[32] A. T. Suresh, F. X. Yu, S. Kumar, and H. B. McMahan, "Distributed mean estimation with limited communication," *Proceedings of the International Conference on Machine Learning (ICML' 17)*, vol. 70, pp. 3329–3337, 2017.

[33] R. Vershynin, *High-dimensional probability: An introduction with applications in data science*. Cambridge university press, 2018, vol. 47.

[34] H. Wang, S. Sievert, S. Liu, Z. Charles, D. Papailiopoulos, and S. Wright, "Atomo: Communication-efficient learning via atomic sparsification," *Advances in Neural Information Processing Systems*, pp. 9850–9861, 2018.

[35] 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," *Advances in Neural Information Processing Systems*, pp. 1509–1519, 2017.

[36] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on information Theory*, vol. 22, no. 1, pp. 1–10, 1976.

[37] H. Yamamoto, "Wyner - ziv theory for a general function of the correlated sources (corresp.)," *IEEE Transactions on Information Theory*, vol. 28, no. 5, pp. 803–807, 1982.