Fundamental Limits of Distributed Optimization over Multiple Access Channel

Shubham K Jha* Prathamesh Mayekar†

***Abstract*—We consider distributed optimization over a d-dimensional space, where K remote clients send coded gradient estimates over an *additive Gaussian Multiple Access Channel (MAC)* with noise variance σ_z^2. Furthermore, the codewords from the K clients must satisfy the average power constraint of P, resulting in a signal-to-noise ratio (SNR) of KP/σ_z^2. In this paper, we study the fundamental limits imposed by MAC on the convergence rate of any distributed optimization algorithm and design optimal communication schemes to achieve these limits. Our first result is a lower bound for the convergence rate showing that compared to the centralized setting, communicating over a MAC imposes a slowdown of $\sqrt{d/\frac{1}{2}\log(1+\texttt{SNR})}$ on any protocol. Next, we design a computationally tractable digital communication scheme that matches the lower bound to a logarithmic factor in K when combined with a projected stochastic gradient descent algorithm. At the heart of our communication scheme is a careful combination of several compression and modulation ideas such as quantizing along random bases, *Wyner-Ziv compression*, *modulo-lattice decoding*, and *amplitude shift keying*. We also show that analog coding schemes, which are popular due to their ease of implementation, can give close to optimal convergence rates at low SNR but experience a slowdown of roughly $\sqrt{d}$ at high SNR.**

I. Introduction

In over-the-air distributed optimization [1], [2], the server wants to minimize an unknown function by getting gradient updates from remote clients. In this setting, the clients must communicate their gradient updates over-the-air, namely through a wireless communication channel, to the server. Due to its applications in federated learning [3], many interesting schemes have been recently proposed for this problem [4]–[11]. However, a clear understanding of the fundamental limits of over-the-air distributed optimization is not present. In this paper, we close this gap by characterizing the fundamental limits imposed on first-order distributed optimization due to over-the-air gradient communication. We also design computationally tractable over-the-air optimization protocols which are almost optimal.

We consider the setting where a server wants to minimize an unknown smooth convex function with domain in $\mathbb{R}^d$ by making gradient queries to K clients. Each of the K clients can generate gradient estimates within a bounded Euclidean distance σ of the true gradient. The clients can communicate their gradient estimates over an *additive Gaussian Multiple Access Channel* (MAC) with variance σ_z^2. Furthermore, each client's communication must also satisfy a power constraint of P, which results in a signal-to-noise ratio (SNR) of KP/σ_z^2. We establish an information-theoretic lower bound on the convergence rate of any over-the-air optimization protocol. Our lower bound shows that there is $\left(\sqrt{\frac{d}{\min(\frac{1}{2}\log(1+\texttt{SNR}),d)}}\right)$ factor slowdown in convergence rate of any over-the-air optimization protocol when compared to that of centralized setting. Next, we design a digital, computationally tractable communication scheme that, combined with the standard *projected stochastic gradient descent* (PSGD) algorithm, almost matches this lower bound.

We elaborate on several key ideas in our communication scheme. In this scheme, we divide the clients into two halves and send the gradients updates from the first half of the clients to form a preliminary estimate. We then employ *Wyner-Ziv* compression to send gradient updates from the second half of clients. This first step is crucial in getting close-to-optimal dependence on the parameter σ in the convergence rate. We also employ quantizing along random bases to get optimal dependence on the dimension d in the convergence rate. Finally, to send a d-dimensional gradient update in a minimum number of channel uses, we use *lattice encoding* and *a modulo lattice decoder*, and *amplitude shift keying* (ASK) modulation.

We also derive tight lower and upper bounds on the performance of analog schemes. Our bounds show that analog schemes are close to the optimal performing schemes at low SNR, but they are highly suboptimal at high SNR and have a slowdown of $\sqrt{d}$ as SNR tends to infinity. Table 1 provides a concise summary of all our results.

Our work is closely related to [12] and [13]. [12], too, studies fundamental limits of over-the-air optimization, but they do so in the single client setting and when the communication channel is the more straightforward additive Gaussian noise channel. The application of distributed optimization considered in [13, Section 5] is similar to ours. However, in their setup, the K remote clients can perfectly communicate any update up to r bits. While the more complicated channel considered in this paper prohibits the direct application of schemes from these papers, we build on the ideas proposed in these two papers to come up with our almost optimal scheme.

In a slightly different direction, distributed optimization with compressed gradient estimates has also been extensively studied in recent years (see, for instance, [14]–[32]). Here gradient compression is employed to mitigate the slowdown in convergence when full gradients are communicated.

*Indian Institute of Science. Email: shubhamkj@iisc.ac.in
†National University of Singapore. Email: pratha22@nus.edu.sg

An extended draft version containing all the detailed proofs can be accessed here: https://drive.google.com/file/d/1LoMPThrVMhos-pvWfzvWf6X--IIZ_A1N/view?usp=share_link

Lower Bound (General)	Proposed Scheme (General)	Lower Bound (Analog)	Proposed Scheme (Analog)
$\frac{D\sigma}{\sqrt{KN}} \cdot \sqrt{\frac{d}{\frac{1}{2}\log(1+\mathtt{SNR})}}$	$\frac{D\sqrt{B\sigma}}{\sqrt{KN}} \cdot \sqrt{\frac{d(\log K + \log\log N)}{\frac{1}{2}\log(1+\mathtt{SNR})}}$	$\frac{D\sigma}{\sqrt{KN}} \cdot \sqrt{\frac{d}{\mathtt{SNR}}}$	$\frac{DB}{\sqrt{KN}} \cdot \sqrt{\frac{d}{\mathtt{SNR}}}$
(Theorem III.3)	(Theorem IV.3)	(Theorem V.2)	(Theorem V.3)

Table 1: Convergence rates of our proposed schemes for large K, N, and for $\frac{1}{2}\log(1+\mathtt{SNR})$ less than d.

II. Setup

Consider the following distributed optimization problem. A *server* wants to minimize an unknown convex function $f\colon \mathcal{X} \to \mathbb{R}$ over its domain $\mathcal{X} \subset \mathbb{R}^d$ using gradient updates from K remote *clients*. At each iteration, the server queries the clients for gradient estimates of the unknown function. On receiving the query, each of the K clients generates a stochastic gradient estimate of the function at the queried point, encodes it, and transmits it over a MAC. The output of this channel is available to the server, which it first decodes and then uses it to update the query point for the next iteration using a first-order optimization algorithm (such as Stochastic Gradient Descent). This setting models practical distributed optimization scenarios arising in federated learning and is of independent theoretical interest.

Our goal is twofold: 1) To understand the fundamental limits imposed by communicating gradients over a MAC on the convergence rate; 2) To design the encoding algorithms at the clients, and the decoding and optimization algorithm at the server to come close to the aforementioned fundamental limit.

A. Functions and gradient estimates

a) Convex and smooth function family: We assume that the server wants to minimize an unknown function f which is convex and L-smooth functions. That is, for all[1] $x, y \in \mathcal{X}$,

$$f(\lambda x + (1-\lambda)y) \le \lambda f(x) + (1-\lambda) f(y), \tag{1}$$

$$f(y) - f(x) \le \nabla f(x)^\top (y-x) + \frac{L}{2}\|y-x\|^2. \tag{2}$$

b) Stochastic gradient estimates: We assume that client $C_k, k \in [K]$, outputs a noisy gradient $\hat{g}_k(x)$ at a query point $x \in \mathcal{X}$ which satisfies the following standard conditions:

$$\mathbb{E}\left[\hat{g}_k(x)|x\right] = \nabla f(x), \text{ (unbiasedness)} \tag{3}$$

$$\mathbb{E}\left[\|\hat{g}_k(x) - \nabla f(x)\|^2 | x\right] \le \sigma^2, \text{ (bounded deviation)} \tag{4}$$

$$\|\hat{g}_k(x)\|^2 \le B^2. \text{ (a.s. bounded estimate)} \tag{5}$$

Denote by $\mathcal{O}$ the set of tuple $(f, \mathcal{C})$ of functions and clients satisfying the conditions (1), (2), (3), (4) and (5).

B. Communication schemes and the multiple access channel

For the tth query x_t made by the server, each of the K clients generates gradient estimates $\{\hat{g}_{k,t}\}_{k=1}^K$. In our setting, the gradient estimates are not directly available to the server. They are first encoded by the clients for error correction and then sent over MAC, and only the output of the channel is available to the server. For all the clients, we consider encoders of length ℓ with average power less than P. That is, the encoder $\varphi_k\colon \mathbb{R}^d \times \mathcal{U} \to \mathbb{R}^\ell$ used by client C_k satisfies the power constraint

$$\mathbb{E}\left[\|\varphi_k(\hat{g}_{k,t}, U)\|^2\right] \le \ell P \quad \forall k \in [K], \tag{6}$$

where U is string of public randomness available to all the k clients' encoders and the server's decoder, and $\mathcal{U}$ is the space of such random strings. For convenience, we will drop the argument U from the rest of the paper.

The encoded codewords $\{\varphi_k(\hat{g}_{k,t})\}_{k=1}^K$ are sent over MAC using ℓ channel uses. The server sees the channel output $Y_t \in \mathbb{R}^\ell$ given by

$$Y_t(j) = \sum_{k=1}^K \varphi_k(\hat{g}_{k,t})(j) + Z_t(j) \qquad \forall j \in [\ell], \tag{7}$$

where $Z_t(j)$ is Gaussian distributed with mean 0 and variance σ_z^2. We denote the *signal-to-noise ratio* by $\mathtt{SNR} \coloneqq \dfrac{KP}{\sigma_z^2}$.

The decoder $\psi\colon \mathbb{R}^\ell \times \mathcal{U} \to \mathbb{R}^d$ at the server projects back the ℓ-length channel output to a vector in $\mathbb{R}^d$, which the optimization algorithm uses to update the query point.

We say that Q is a (d, ℓ, P, K)-*communication scheme* if it is a tuple $(\varphi_1, \ldots, \varphi_k, \psi)$, where φ_k, $k \in [K]$, and ψ are as described above. Denote by $\mathcal{Q}_\ell$ the set of all possible (d, ℓ, P, K)-communication schemes.

C. Over-the-air optimization

We now describe the optimization algorithm π interacting with the tuple $(\varphi_1, \ldots, \varphi_k, \psi) \in \mathcal{Q}_\ell$. At iteration t, the optimization algorithm uses all the previous query points, $\{x_{t'}\}_{t'=1}^{t-1}$, and the decoded gradient estimates, $\{\psi(Y_{t'})\}_{t'=1}^{t-1}$, to decide on the query point $x_t \in \mathcal{X}$. The server then queries the clients at the point x_t, resulting in a gradient estimate $\psi(Y_t)$. This continues for T iterations, after which the algorithm outputs a point $x_T \in \mathcal{X}$.

Denote by $\Pi_{T,\ell}$ the set of all optimization algorithms π making T queries to the clients and interacting with a (d, ℓ, P, K)-communication scheme.

For an optimization algorithm $\pi \in \Pi_{T,\ell}$ and a communication scheme $Q \in \mathcal{Q}_\ell$, we call the tuple (π, Q) an *over-the-air optimization protocol*. For a tuple of function and clients $(f, \mathcal{C}) \in \mathcal{O}$, we measure the performance of any over-the-air optimization protocol (π, Q) by the convergence error

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

[1] $\|\cdot\|$ refers to the standard euclidean norm.

We will study this error when the total number of channel uses, $T\ell$, is restricted to be at most N. We can use communication schemes of arbitrary length ℓ. Note, however, that to increase the length ℓ, we must decrease the number of queries T, since the total number of channel uses is limited to N. Conversely, to increase the number of queries, we must decrease the length of the communication schemes. Let $\Lambda(N) := \{(\pi, Q) : \pi \in \Pi_{T,\ell}, Q \in \mathcal{Q}_\ell, T\ell \leq N\}$ be the set of over-the-air optimization protocols with at most N channel uses. The smallest worst-case error possible over all such protocols is given by

$$\mathcal{E}^*(N, K, \texttt{SNR}, \mathcal{X}) := \inf_{(\pi,Q)\in\Lambda(N)} \sup_{(f,\mathcal{C})\in\mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, Q).$$

Let $\mathbb{X} := \{\mathcal{X} \colon \sup_{x,y\in\mathcal{X}} \|x-y\| \leq D\}$. In this paper, we will characterize[2] $\mathcal{E}^*(N, K, \texttt{SNR}) := \sup_{\mathcal{X}\in\mathbb{X}} \mathcal{E}^*(N, K, \texttt{SNR}, \mathcal{X})$.

III. Preliminaries and an information theoretic lower bound

A. A benchmark from prior results

We recall the results for the centralized case, which we can model by setting $\texttt{SNR} = \infty$. In this case, clients can perfectly communicate the gradient estimates in only one channel use. A direct application of [33, Theorem 6.3] leads to the following upper bound on $\mathcal{E}^*(N, K, \infty)$ which serves as a basic benchmark for our results in this paper.

Theorem III.1. $\mathcal{E}^*(N, K, \infty) \leq \frac{\sqrt{2}D\sigma}{\sqrt{KN}} + \frac{LD^2}{2N}$.

B. A general convergence bound

Throughout the paper, we will use projected stochastic gradient descent (PSGD) as the first-order optimization algorithm π; the overall over-the-air optimization protocol is described in Algorithm 2. PSGD proceeds as stochastic gradient descent 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: $\quad x_{t+1} = \Gamma_{\mathcal{X}}\left(x_t - \eta_t \psi(Y_t)\right)$
3: Output $\bar{x}_T = \frac{1}{T}\sum_{t=1}^{T} x_t$

Algorithm 2: PSGD for over-the-air optimization

The convergence rate of Algorithm 2 is controlled by the square root of worst-case mean square error (MSE) $\alpha(Q)$ and the worst-case bias $\beta(Q)$ of the gradient estimates decoded by the server. They are defined as follows:

$$\alpha(Q) := \sup_{\substack{\forall x,k\in[K],\hat{g}_k\in\mathbb{R}^d:\\ \mathbb{E}\|\hat{g}_k - \nabla f(x)\|^2 \leq \sigma^2}} \sqrt{\mathbb{E}\left[\|\psi(Y) - \nabla f(x)\|^2\right]},$$

$$\beta(Q) := \sup_{\substack{\forall x,k\in[K],\hat{g}_k\in\mathbb{R}^d:\\ \mathbb{E}\|\hat{g}_k - \nabla f(x)\|^2 \leq \sigma^2}} \|\mathbb{E}\left[\psi(Y)\right] - \nabla f(x)\|,$$

[2]While our upper bound techniques can handle an arbitrary, fixed $\mathcal{X}$, the supremum over $\mathbb{X}$ is to ensure that the lower bounds are independent of the geometry of set $\mathcal{X}$.

where for $i \in [d]$, $Y(i)$ satisfies (7) and the expectation is taken over all the randomness in the set up. We now recall a lemma from [13] that upper bounds the convergence rate of Algorithm 2 in terms of $\alpha(Q)$ and $\beta(Q)$.

Lemma III.2 ([13, Lemma II.2])**.** *Let π be the PSGD algorithm making T queries to the clients and Q be any communication scheme in $\mathcal{Q}_\ell$. Then, we have*

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

with the learning rate $\eta_t = \min\{\frac{1}{L}, \frac{D}{\alpha(Q)\sqrt{T}}\}, \forall t \in [T]$. Further, the over-the-air optimization protocol uses the MAC channel $N = T \cdot \ell$ times.

Thus it is enough to control the mean square error and bias of the communication scheme to upper bound the convergence rate of the corresponding over-the-air optimization protocol.

C. Lower bound for over-the-air optimization

We now present an information-theoretic lower bound for any over-the-air optimization protocol. We note that [12] shows a similar lower bound in the single client setting. We build on their proof and extend the result to the more general setting of K clients. The key step involves showing that over-the-air optimization over parallel independent additive Gaussian noise channel is much easier than over MAC and then proceeding as in [12].

Theorem III.3. *For some universal constant $c \in (0, 1)$ and $N \geq \frac{d}{K\log(1+\texttt{SNR})}$, we have*

$$\mathcal{E}^*(N, K, \texttt{SNR}) \geq \frac{cD\sigma}{\sqrt{KN}}\sqrt{\frac{d}{\min\{d, \frac{1}{2}\log(1+\texttt{SNR})\}}}.$$

Our lower bound states that, except for very high values of SNR, any over-the-air optimization protocol will experience a slowdown by a factor of $\sqrt{\frac{d}{\frac{1}{2}\log(1+\texttt{SNR})}}$ over the convergence rate of centralized setting.

IV. A digital communication scheme for over-the-air optimization

In this section, we present our main result: a digital communication scheme that, combined with PSGD, will almost match the lower bound in Theorem III.3. Our scheme below is "universal" in the sense that the clients don't require the knowledge of σ for the transmission of gradient estimates.

A. Warm-up scheme UQ-OTA

For ease of presentation, we first present a warm-up scheme based on uniform quantization. We will build on the components described below to present our final digital scheme. Throughout the description of our schemes, we omit the subscript t for convenience.

a) Uniform quantization: Each client k first divides the gradient estimate $\hat{g}_k$ by the number of clients K to form $\tilde{g}_k$ and quantizes it using an unbiased v-level *coordinate-wise uniform quantizer* $v-\texttt{CUQ}$. The $v-\texttt{CUQ}$ takes ith coordinate $\tilde{g}_k(i) \in [-\frac{B}{K}, \frac{B}{K}]$ as input and outputs $z_{k,i} \in \{0, ..., v-1\}$ as per the following rule:

$$z_{k,i} = \begin{cases} \left\lceil \frac{(v-1)(K\tilde{g}_k(i)+B)}{2B} \right\rceil, & \text{w.p. } \frac{\tilde{g}_k(i) - \lfloor \frac{\tilde{g}_k(i)K(v-1)}{2B} \rfloor}{2B/(K(v-1))} \\ \left\lfloor \frac{(v-1)(K\tilde{g}_k(i)+B)}{2B} \right\rfloor, & \text{w.p. } \frac{\left\lceil \frac{\tilde{g}_k(i)K(v-1)}{2B} \right\rceil - \tilde{g}_k(i)}{2B/(K(v-1))} \end{cases}.$$

Note that the $z_{k,i}$ suffices to form an unbiased estimate of $\tilde{g}_k(i)$. Define $\mathtt{Q}_k := \{z_{k,i} : i \in [d]\}$ as the quantized output for client k.

b) Lattice encoding and ASK modulation using $\mathcal{M}(\mathtt{Q}_k, v, p)$: Client k sends $\{z_{k,i}\}_{i\in[d]}$ over MAC by first encoding them as one-dimensional lattice points and then modulating each lattice point onto an ASK code. This entire procedure is denoted by $\mathcal{M}(\mathtt{Q}_k, v, p)$, where parameters v and p will be specified later, and is described below.

For some parameter[3] $p \leq d$, the set of coordinates $[d]$ is equally partitioned into d/p blocks. For $j \in [d/p]$, the jth block is given by $\mathcal{B}_j := \{(j-1)p+1, ..., (j-1)p+p\}$. For each $\mathcal{B}_j$, the corresponding quantized values are mapped onto an one-dimensional lattice generated by bases $\{w^0, ..., w^{p-1}\}$. Denote by $\tau_{k,j}$ the lattice point corresponding to block $\mathcal{B}_j$, of the kth client. Symbolically,

$$\tau_{k,j} = w^0 \mathtt{Q}_k(\mathcal{B}_j(1)) + \cdots + w^{p-1}\mathtt{Q}_k(\mathcal{B}_j(p)),$$

where $w = K(v-1)+1$. Our choice of w is to ensure a successful recovery of the sum of client updates at the server.

To satisfy the power constraints of MAC, we then modulate each $\tau_{k,j}$ to $[-\sqrt{P}, \sqrt{P}]$ using an ASK code.

Definition IV.1. A code is an *Amplitude Shift Keying (ASK) code* satisfying the average power constraint (6) if the range $\mathcal{A}$ of the encoder mapping is given by $\mathcal{A} := \left\{-\sqrt{P} + (i-1)\cdot\frac{2\sqrt{P}}{r-1} : i \in [r]\right\}$, for some $r \in \mathbb{N}$. Note that this is a code of length 1.

Since each $\tau_{k,j}$ takes values in $\{0, ..., \frac{w^p-1}{K}\}$, we set size of ASK code $r = \frac{w^p-1}{K} + 1$ to establish one-to-one correspondence. Consequently, we have $\ell = d/p$ and $\varphi_k(j) = \mathcal{A}(\tau_{k,j}+1), \forall j \in [d/p], k \in [K]$ in (7).

c) Lattice decoding at server $\mathcal{L}(Y, v, p)$: On the server side, our goal is to compute an unbiased estimate of sum $\sum_k \tilde{g}_k$ from Y. Therefore, it simply suffices to recover just the sum $\sum_k \mathtt{Q}_k$, instead of individual $\mathtt{Q}_k$s.

Towards that, each coordinate $Y(j), j \in \ell$, is first fed into a coordinate-wise minimum-distance (MD) decoder, thereby locating the nearest possible point $\hat{Y}(j)$ in $\{-K\sqrt{P} + \frac{2(i-1)\sqrt{P}}{r-1} : i \in [r]\}$. Using the one-to-one correspondence, the decoded point $\hat{Y}(j)$ is then mapped back to the same lattice generated using $\{w^0, ..., w^{p-1}\}$ to decode the sum of transmitted lattice points $\sum_k \tau_{k,j}$. Denote by $\hat{\tilde{\tau}}_j$ the decoded lattice point can be expressed as

$$\hat{\tilde{\tau}}_j = w^0\lambda(\mathcal{B}_j(1)) + ...w^{p-1}\lambda(\mathcal{B}_j(p)),$$

for some vector $\lambda \in \{0, ..., K(v-1)\}^d$. Therefore, to recover the desired sum, the server uses a *modulo-lattice decoder* for each $\mathcal{B}_j, j \in [d/p]$, that successively outputs the coordinates of λ. In particular $\forall i \in [p]$,

$$\lambda(\mathcal{B}_j(i)) = \frac{\hat{\tilde{\tau}}_j - \lambda(\mathcal{B}_j(1)) ... - w^{i-2}\lambda(\mathcal{B}_j(i-1))}{w^{i-1}} \mod w.$$

Note that such recovery is possible with the current choice of w since every coordinate of $\sum_{k\in[K]} \mathtt{Q}_k$ is less than w. The value λ obtained above is finally used to form $\psi(Y)$ to be used in Algorithm 2 as

$$\psi(Y) = -B + \lambda \cdot \frac{2B}{K(v-1)}. \tag{8}$$

Theorem IV.2. *Let π be the optimization algorithm described in Algorithm 2, where $\psi(Y)$ is obtained in (8) with $v = \sqrt{d}+1$. Then, for a universal constant $c_1 > 0$ and integers p, K such that $d \geq p \geq 1$ and $K \geq B^2/\sigma^2$, we have*

$$\sup_{(f,\mathcal{C})\in\mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, Q) \leq \frac{c_1 DB}{\sqrt{KN}}\sqrt{\frac{d}{p}} + \frac{LD^2 d}{2Np},$$

where $p = \lfloor \frac{\log\left(1+\sqrt{\frac{2K\mathrm{SNR}}{\ln(KN^{1.5})}}\right)}{\log(Kd)} \rfloor$.

Proof sketch: It is easy to see that under perfect decoding $\beta(Q) = 0$ and $\alpha(Q) \leq \frac{4B\sqrt{d}}{\sqrt{K}(v-1)}$, where Q is $\texttt{UQ-OTA}$. We can then use Lemma III.2 to upper bound the expected optimization error under perfect decoding. To upper bound the expected optimization error when there is a channel decoding error, note that the optimization error is trivially bounded by DB. It only remains to bound the probability of channel decoding. The proof is then completed by noting that our choice of size of ASK code r, leads to the probability of channel decoding error being at the most $\frac{1}{K\sqrt{N}}$.

B. Wyner-Ziv digital scheme $\texttt{WZ-OTA}$

We are now ready to present our main digital scheme $\texttt{WZ-OTA}$ which significantly improves over the performance of $\texttt{UQ-OTA}$ and is almost optimal.

In this scheme, we partition the clients $\mathcal{C}$ equally into two sets $\mathcal{C}_1$ and $\mathcal{C}_2$. In each iteration t, the clients in $\mathcal{C}_1$ construct the side information at the server, and the remaining clients in $\mathcal{C}_2$ exploit this information to form a Wyner-Ziv estimate of $\nabla f(x_t)$ at the server.

a) Side information construction: The clients in $\mathcal{C}_1$ use the previously described $\texttt{UQ-OTA}$ communication scheme to form a preliminary estimate (8) at the server. This requires $\ell = d/p$ channel uses. Note that the clients in $\mathcal{C}_2$ send 0 during these transmissions.

The server divides this preliminary estimate by $K/2$ to form S and then rotates it by a random matrix $\mathbf{R}$ to form the side information $\mathbf{R}S$. Here $\mathbf{R} = 1/\sqrt{d}\mathbf{H}\mathbf{D}'$ where $\mathbf{H}$ is the Walsh-Hadamard[4] matrix [34], and $\mathbf{D}'$ is a random diagonal matrix

[3] For simplicity, we assume p divides d.

[4] Without loss of generality, we assume d is a power of 2. If not, we can zero-pad the gradient estimates and make the resulting dimension power of 2; this only adds a constant multiplicative factor to our upper bounds.

with non-zero entries generated uniformly and independently from $\{-1,+1\}$.

b) The Wyner-Ziv estimate: The clients in $\mathcal{C}_2$ use a Wyner-Ziv estimator boosted DAQ from [13] to construct the final estimate, while those in $\mathcal{C}_1$ tranmit 0 in all channel uses. The boosted DAQ uses the idea of correlated sampling between the input and the side information to reduce quantization error. Specifically, for an input $|x| \leq M$ at the encoder and a corresponding side information $|y| \leq M$ at the decoder, the boosted DAQ estimate is given by

$$\hat{X} = (2M/I) \sum_{i\in[I]} \left(\mathbb{1}_{\{U_i \leq x\}} - \mathbb{1}_{\{U_i \leq y\}}\right) + y, \qquad (9)$$

where each $U_i \sim \texttt{unif}[-M, M]$ is a uniform random variable. Note that $\hat{X}$ is an unbiased estimate of x with MSE at most $2M|x-y|/I$.

In our setting, each client $k \in \mathcal{C}_2$ first pre-processes its noisy estimate as $\tilde{g}_k = \frac{2\hat{g}_k}{K}$ and uses shared randomness to draw I uniform random vectors $U_{k,i} \in [-M, M]^d, i \in [I]$, independently. The choice of M and I are crucial for our scheme and will be specified later. Using shared randomness again, each $\tilde{g}_k$ is rotated using the same random matrix $\mathbf{R}$ used earlier. Each coordinate of this rotated vector is then quantized to an element in $\{0, ..., I\}$ as

$$\mathtt{Q}_k(j) = \sum_{i\in[I]} \mathbb{1}_{\{U_{k,i}(j) \leq \mathbf{R}\tilde{g}_k(j)\}}, \ \forall j \in [d].$$

As an aside, it is instructive to note that under the event $\mathcal{V}_j = \{|\mathbf{R}S(j)| \leq M, |\mathbf{R}\tilde{g}_k(j)| \leq M\}$, $\mathtt{Q}_k(j)$ suffices to form an unbiased estimate of $\mathbf{R}\tilde{g}_k(j)$ using boosted DAQ (see (9)). Coming back to our scheme, each client k transmits the quantized vector $\mathtt{Q}_k$ over the MAC channel by first using the lattice encoder and then using ASK modulation. The entire operation is described by the function $\mathcal{M}(\mathtt{Q}_k, v', p')$ (see Section IV-A) with $v' = I + 1$ and p' to be specified shortly. Note that there are $\ell = d/p'$ channel uses per iteration.

At the server, the channel output $Y \in \mathbb{R}^{d/p'}$ is passed through $\mathcal{L}(Y, v', p')$ to obtain λ. Following the boosted DAQ estimator (9), the final output $\psi(Y)$ is given by

$$\psi(Y) = (2M/I)\mathbf{R}^{-1} \sum_{j\in[d]} (\lambda(j) - \omega(j))\, e_j + (K/2)S, \quad (10)$$

where each $\omega(j) = \sum_{k\in\mathcal{C}_2} \sum_{i\in[I]} \mathbb{1}_{\{U_{k,i}(j) \leq \mathbf{R}S(j)\}}$. We next characterise the performance of WZ-OTA.

Theorem IV.3. *Let c_2, c_3 be positive universal constants and π be the optimization algorithm described in Algorithm 2, where $\psi(Y)$ is obtained using (10) with $v = 7, M = \frac{c_2 B}{K\sqrt{d}}\sqrt{\ln(K^{1.5}N)}$ and $I = c_2\sqrt{\ln(K^{1.5}N)}$. Then, for integers p, p' and K such that $K \geq B^2 d/\sigma^2$ and $d \geq p, p' \geq 1$, we have*

$$\sup_{(f,\mathcal{C})\in\mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, Q) \leq \frac{c_3 D\sqrt{B\sigma}}{\sqrt{KN}}\sqrt{\frac{d}{q} + \frac{LD^2 d}{2Nq}},$$

where $\frac{1}{q} = \frac{1}{p} + \frac{1}{p'}$ with $p = \lfloor \frac{\log\left(1+\sqrt{\frac{K\mathtt{SNR}}{2\ln(KN^{1.5})}}\right)}{\log K} \rfloor$ and $p' = \lfloor \frac{\log\left(1+\sqrt{\frac{K\mathtt{SNR}}{2\ln(KN^{1.5})}}\right)}{\log K + \log\log(N)} \rfloor$.

Proof sketch: We follow a similar strategy as in proof of Theorem IV.2 to upper bound the expected optimization error. Under channel decoding error, the optimization error is bounded similarly as in the previous proof.

It only remains to bound optimization error under perfect decoding for which we employ Lemma III.2. The key step involved in bounding α and β is to show that the preliminary estimate is close to the true gradient. In addition, we note that under the event $\mathcal{V}_j$ as defined earlier, clients in $\mathcal{C}_2$ communicate unbiased gradient estimates, and the MSE can be bounded using MSE bounds for boosted DAQ. At last, our choice of M ensures sufficiently small probability for $\mathcal{V}_j^c$, thus completing the proof.

Remark 1. For large K, N, we remark that the WZ-OTA combined with PSGD is off only by $\sqrt{B/\sigma\,(\log(K) + \log\log N)}$ factor from our lower bound. In comparison, from Theorem IV.2, UQ-OTA combined with PSGD is off by $B/\sigma\sqrt{(\log(K) + \log(d) + \log\log N)}$. Quantization along random bases and Wyner-ziv compression allows WZ-OTA to improve by factors $\log d$ and $\sqrt{B/\sigma}$ over UQ-OTA.

V. Performance of Analog Schemes

Definition V.1. A communication scheme is an *analog scheme* if the encoder mapping φ is linear, i.e., $\varphi(x) = \mathbf{A}x$ for $\mathbf{A} \in \mathbb{R}^{\ell\times d}$ and $\ell \leq d$. We allow random entries for $\mathbf{A}$ as long as the randomness is independent of x. For the class of (d, ℓ, P, K)-communication schemes restricted to using such analog schemes, we denote by $\mathcal{E}^*_{\mathtt{analog}}(N, K, \mathtt{SNR})$ the corresponding min-max optimization error. Clearly, $\mathcal{E}^*_{\mathtt{analog}}(N, K, \mathtt{SNR}) \geq \mathcal{E}^*(N, K, \mathtt{SNR})$.

We begin by proving a lower bound for analog communication schemes.

Theorem V.2. *For some universal constant $c \in (0, 1)$, and $N \geq \frac{d}{K}(\sigma^2 + \frac{\sigma^2}{\mathtt{SNR}})$, we have $\mathcal{E}^*_{\mathtt{analog}}(N, K, \mathtt{SNR}) \geq \frac{cD}{\sqrt{KN}}\sqrt{d\sigma^2 + \frac{d\sigma^2}{\mathtt{SNR}}}$.*

The following lower bound also uses affine functions as difficult functions and builds on a class of Gaussian oracles proposed, recently, towards proving a similar result in [12].

For our upper bound, we use the well-known *scaled transmission* scheme from [7]. In this scheme, the gradient estimates are scaled-down by $\sqrt{dP}/B$ by every client $C_k \in \mathcal{C}$ to satisfy the power constraint in (6), sent coordinate-by-coordinate over d channel uses, and then scaled-up by $B/\sqrt{dP}$ and averaged at the server before using it in a gradient descent procedure. It is not difficult to see the following upper bound.

Theorem V.3. *Let π be the PSGD optimization algorithm and Q be the scaled transmission communication scheme described above. Then, we have $\sup_{(f,\mathcal{C})\in\mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, Q) \leq \frac{\sqrt{2}D}{\sqrt{KN}}\sqrt{d\sigma^2 + \frac{dB^2}{\mathtt{SNR}}} + \frac{dLD^2}{2N}$.*

Remark 2. For $\mathtt{SNR} \geq B^2/\sigma^2$, Theorem V.2 shows that compared to the centralized setting discussed in Theorem III.1, analog schemes will have a slowdown of $\sqrt{d}$. However, for

small values of SNR, an analog communication scheme combined with PSGD gives close optimal performance. It matches the lower bound in Theorem III.3 up to a factor of B/σ. This observation follows by noting that $\log(1+\mathtt{SNR}) \approx \mathtt{SNR}$ for small values of SNR.

VI. Conclusion

We provide an almost complete characterization of the min-max convergence rate of over-the-air distributed optimization. Our bounds show that a simple analog coding scheme is optimal at low values of SNR, but they can be far from optimal at high values of SNR (Remark 2). This observation mirrors the observation made by [12], albeit in the single client setting. Furthermore, we design an explicit digital communication scheme based on lattice coding to match our lower bound for all values of SNR. We hope our work inspires other explicit communication schemes for similar distributed optimization problems. Our upper bound matches our lower bound up to a nominal $\sqrt{\log K + \log\log N}$ factor (Theorem IV.3). Further closing the gap between our upper and lower bound would lead to new communication schemes or lower bound techniques for distributed optimization and is an exciting research direction.

References

[1] M. M. Amiri and D. Gündüz, "Over-the-Air Machine Learning at the Wireless Edge," in *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, 2019.

[2] M. M. Amiri and D. Gündüz, "Machine Learning at the Wireless Edge: Distributed Stochastic Gradient Descent Over-the-Air," in *IEEE International Symposium on Information Theory (ISIT)*, pp. 1432–1436, 2019.

[3] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *NIPS Workshop on Private Multi-Party Machine Learning*, 2016.

[4] A. Sahin and R. Yang, "A survey on over-the-air computation," *https://arxiv.org/abs/2210.11350*, Nov 2022.

[5] W.-T. Chang and R. Tandon, "Communication Efficient Federated Learning over Multiple Access Channels," *https://arxiv.org/abs/2001.08737*, 2020.

[6] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "COTAF: Convergent Over-the-Air Federated Learning," in *IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2020.

[7] T. Sery and K. Cohen, "On Analog Gradient Descent Learning Over Multiple Access Fading Channels," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2897–2911, 2020.

[8] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated Learning via Over-the-Air Computation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, 2020.

[9] J. Zhang, N. Li, and M. Dedeoglu, "Federated Learning over Wireless Networks: A Band-limited Coordinated Descent Approach," *https://arxiv.org/abs/2102.07972*, 2021.

[10] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-Bit Over-the-Air Aggregation for Communication-Efficient Federated Edge Learning: Design and Convergence Analysis," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2120–2135, 2021.

[11] 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)*, pp. 5055–5059, IEEE, 2021.

[12] S. K. Jha, P. Mayekar, and H. Tyagi, "Fundamental limits of over-the-air optimization: Are analog schemes optimal?," *IEEE Journal on Selected Areas in Information Theory*, vol. 3, no. 2, pp. 217–228, 2022.

[13] P. Mayekar, S. K. Jha, A. T. Suresh, and H. Tyagi, "Wyner-ziv estimators for distributed mean estimation with side information and optimization," *https://arxiv.org/abs/2011.12160.v2*, 2022.

[14] 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.

[15] 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*, Proceedings of Machine Learning Research, pp. 2197–2205, PMLR, 2021.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] P. Mayekar and H. Tyagi, "RATQ: A universal fixed-length quantizer for stochastic optimization," *IEEE Transactions on Information Theory*, 2020.

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

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

[23] 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)*, pp. 3110–3114, 2021.

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

[25] 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*, pp. 2021–2031, PMLR, 2020.

[26] 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.

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

[28] J. Acharya, C. Canonne, P. Mayekar, and H. Tyagi, "Information-constrained optimization: can adaptive processing of gradients help?," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 7126–7138, 2021.

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

[30] 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.

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

[32] P. Davies, V. Gurunanthan, N. Moshrefi, S. Ashkboos, and D.-A. Alistarh, "New bounds for distributed mean estimation and variance reduction," in *9th International Conference on Learning Representations*, 2021.

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

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