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 \(\sigma\) 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 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 \(\sigma\) 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 \(\Omega(\sqrt{\sigma^2d/KrT})\). 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 \(f: \mathcal{X} \to \mathbb{R}\) 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
fixed set of quantizers $Q_r$, and the optimization algorithm $\pi$ 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 $Q_{1:t}$, $\forall i \in [K], t \in [T]$, and an optimization algorithm $\pi$ 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 $t$th iteration may depend on all the previous quantized outputs. Specifically, denoting by $Y_{i,t}$ the $i$th client’s quantized output at time $t$, the adaptive quantizer selection strategy $S := (S_1, \ldots, S_T)$ over $T$ iterations consists of mappings $S_t: \mathbb{R}^{dK(t-1)} \to Q_{r}^K$ that take $\{Y_{i,t'}\}_{i \in [K], t' \in [t-1]}$ as input and outputs a tuple of $K$ quantizers $\{Q_{i,t}\}_{i \in [K]} \in Q_r^K$. We write $S_{Q_{1:T}}$ 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 $\pi$ 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 $Q_{i,t} \in Q_r$ chosen based on strategy $S$, and the output $Y_{i,t}$ is observed by the first-order optimization algorithm $\pi$. 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 $\Pi_T$ be the set of all first-order optimization algorithms that make $T$ queries to $\mathcal{C}$ and for the $t$th query $x_t$, get back the outputs $\{Y_{i,t}\}_{i \in [K]}$. We measure the performance of an optimization protocol $\pi$ 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

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

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 $Q_r$ and the number of iterations $T$, we define the minimax optimization error as

$$\mathcal{E}^*(\mathcal{X}, \mathcal{O}, T, Q_r) = \inf_{\pi \in \Pi_T} \inf_{S \in \mathcal{S}_{Q_{1:T}}} \sup_{(f, \mathcal{C}) \in \mathcal{O}} \mathcal{E}(f, \mathcal{C}, \pi, S).$$

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 $\pi$.

1The set of $r$-bit quantizers $Q_r$ 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, $\forall \lambda \in [0, 1], \forall x, y \in \mathbb{R}^d$,

$$f(\lambda x + (1 - \lambda) y) \leq \lambda f(x) + (1 - \lambda) f(y),$$

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L \|x - y\|_2,$$

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:

$$\mathbb{E}[\hat{g}_i(x) \mid x] = \nabla f(x),$$

(unbiased estimates)

$$\|\hat{g}_i(x) - \nabla f(x)\|_2^2 \leq \sigma^2,$$

(maximum deviation bound)

$$\max_{x \in \mathcal{X}} \|\hat{g}_i(x)\|_2^2 \leq B^2,$$

(a.s. bounded estimate)

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 $\mathcal{O}_{sc}$ 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 $\mathcal{X}_2(D) := \{\mathcal{X} \subseteq \mathbb{R}_d^d : \max_{x, y \in \mathcal{X}} \|x - y\|_2 \leq D\}$ be the collection of subsets of $\mathbb{R}^d$ whose $\ell_2$ 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 $T \geq d/(6K)$,

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

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

$$\Gamma_\mathcal{X}(y) := \min_{x \in \mathcal{X}} \|x - y\|, \forall y \in \mathbb{R}^d.$$
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

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

$$\beta(Q) := \sup_{\forall x,i \in [K], \bar{g}_t \in \mathbb{R}^d} \| \mathbb{E} \left[ (Q(\bar{Y}) - \nabla f(x)) \right] \|, \quad (7)$$

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 $\tilde{x}_T$ of optimization algorithm given in Algorithm 1 satisfies

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

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

**D. Sub-gaussian norm and random rotation**

For our analysis, it will be convenient to recall the definition of subgaussian norm $\sigma$ of a random variable.

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

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 $R := 1/\sqrt{d}HD'$, 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 $2\| \cdot \|_{\psi_2}$ as the norm.
Theorem III.1 and relegates the dependence of convergence rate on \( B \) to only second order terms.

At each iteration \( t \), \( \text{WZ} - \text{SGD} \) uses the clients in \( C_1 \) to form the side information estimate \( Z_t \) at the server and then uses the clients in \( C_2 \) to estimate the gradient for the gradient descent step, where \( C_1 := \{ C_1, \ldots, C_{K/2} \} \), \( C_2 := \mathcal{C} \setminus C_1 \).

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 \( B \) rate on \( \text{Theorem III.1} \) and relegates the dependence of convergence to \( \mathcal{C} \) by using the clients in \( \mathcal{C} \) at each iteration \( t \). Then, the average of the quantized outputs from all its associated clients is encoded by client \( c \). For each block, we then form the side information by taking \( \ell_1 \)-level CUQ to quantize the associated \( B \)-bit coordinate. Thus, the overall communication constraint.

b) The Wyner-Ziv gradient estimate \( Q_{x,x} \): We use the clients in \( C_2 \) to form the actual gradient estimate. The clients encode the stochastic gradients using a subsampled RMQ quantizer from \cite{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 \cite{8} and also used in \cite{22}. We follow the description given in \cite[Section 3.1]{22}. 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 a predefined distance between \( x \) and \( y \), the precision \( \log \ell_2 \), and lattice parameter \( c \). The encoder of MQ first randomly quantizes \( x \) to either \( [x/c] \) or \( [x/c]\) such that the output is an unbiased estimate of \( x/c \). Then, a modulo-\( k \) operation is performed on that output and it is sent to the decoder. That is,

\[
Q_M^x(x) = z \mod k, \quad \text{where} \quad z = \begin{cases} \lfloor x/c \rfloor, & \text{w.p. } x/c - \lfloor x/c \rfloor \\ \lfloor x/c \rfloor, & \text{w.p. } \lfloor x/c \rfloor - x/c. \end{cases}
\]

The decoder is described as follows.

\[
Q_M(x,y) = \min \{ z \ell_2 + Q_M^x(x) z - \lfloor y \rfloor : z \in \mathbb{Z} \}.
\]

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 \( g_j(x) \), the output encoded by client \( C_j \) using subsampled RMQ is described as follows:

\[
Q_{xz,j}(\hat{g}_j(x_j)) = \{ Q_M^x(\hat{g}_j(x_j)(i)) : i \in \mathcal{D}_j \}.
\]

At the server, the communication for all \( C_j \in C_2 \) is decoded as follows:

\[
Q_{xz,j}(\hat{g}_j(x_j), Z_t) = R_i^{-1} \left( \sum_{i \in \mathcal{D}_j} (\hat{g}_j - R_j Z_t) e_i + R_j Z_t \right)
\]

where \( \hat{g}_j(i) = Q_M(R_j \hat{g}_j(x_j)(i), R_j Z_t(i)) \). Finally, the server averages over all the quantized gradient estimates of clients in \( C_2 \) to get \( Q \), which in turn is used to make the PSGD step in line 2 of Algorithm 1. That is,

\[
Q_t = \frac{\sum_{j=K/2+1}^K Q_{xz,j}(\hat{g}_j(x_j), Z_t)}{K/2}.
\]

We are now ready to present our main result: the convergence rate of \( \text{WZ} - \text{SGD} \) 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 \( C_2 \). Let \( \pi \) be the optimization algorithm described in Algorithm 1 which uses \( Q_t \) in (11) to make the PSGD step after the \( t \)-th 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 \( K \geq c_2 d^2 \log (B/\sigma) \), we have

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

Thus, in the setting where the number of clients \( K \) is large, we match the lower bound in Theorem II.1 up to a \( \log \log KT \) factor.
IV. ANALYSIS OF WZ-SGD

a) Side information is close to gradient estimates:
We begin by noting that side-information $Z^o$ is close to the stochastic gradient estimates computed by clients in $C_2$. 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 $C_1$, we get the following.

Lemma IV.1. For all $x \in \mathbb{R}^d$, $j \in C_2$, and $i \in [d]$, we have
$$\Pr(|R\hat{g}_j(x)(i) - RZ(i)| \geq t) \leq 2e^{-c \min\{\frac{t^2}{d\sigma^2} - \frac{\sigma^2}{d}, \frac{t^2}{4}\}} + 2e^{-c \frac{t^2}{d\sigma^2}}$$
where $R$ is a random Hadamard matrix (8) and for a universal constant $c$
$$\sigma^2 = \frac{c8d\sigma^2 \lceil \log(2B/\sigma + 1) \rceil}{Kr}.$$ (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
$$\Pr(|RZ(i) - R\nabla f(x)(i)| \geq t) \leq 2e^{-c \min\{t^2/d\sigma^2, t\sqrt{\sigma}\}}.$$ 

Finally, using similar proof techniques as in [24, Lemma 5.8], we can show that the rotated gradient estimates of clients in $C_2$ is close to the rotation of the true gradient.

Lemma IV.4. For all $x \in \mathbb{R}^d$ we have
$$\|R\hat{g}_j(x)(i) - R\nabla f(x)(i)\|_{\psi_2}^2 \leq \sigma^2/d.$$ 

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

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

Remark 3. Note that such a choice of $\varepsilon$ ensures that whenever a coordinate of the rotated vector $R\hat{g}_j(x)$ is within $\Delta'$ of the corresponding coordinate of the rotated side information there is no error in decoding. Therefore, the output of modulo quantizer is unbiased and $\varepsilon$ 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 $Q_{\text{RMQ},j}$ the rotated modulo quantizer without any subsampling for client $j \in C_2$. That is,
$$Q_{\text{RMQ},j}(\hat{g}_j(x), Z) = R_j^{-1} \left( \sum_{i \in [d]} (\hat{g}_j - R_j Z(i)) e_i + R_j Z \right)$$
where $\hat{g}_j(i) = Q_{\text{RMQ}}(R_j \hat{g}_j(x)(i))$.

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 $x \in \mathbb{R}^d$, $j \in C_2$, and $\delta \in (0, 2\sigma/\sqrt{\ell})$ that
$$\mathbb{E} \left[ \|Q_{\text{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2 \right] \leq \frac{144\sigma^2}{c(l_2 - 2)\varepsilon} \left( \frac{\log \frac{2\sigma}{\varepsilon\delta}}{\sqrt{\varepsilon\delta}} \right)^2 + 251\delta^2.$$ 

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
$$\mathbb{E} \left[ \|Q_{\text{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2 \right] \leq d\varepsilon^2 + d\sum_{i=1}^d \mathbb{E} \left[ (Q_{\text{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x))(i)^2 \mathbb{1}_{|R_j \hat{g}_j(x) - Z(i)| \geq \Delta'} \right]$$
$$\leq d\varepsilon^2 + 2\ell_2 \varepsilon^2 \sum_{i=1}^d \Pr(|R_j \hat{g}_j(x) - Z(i)| \geq \Delta')$$
$$+ 2d \sum_{i=1}^d \mathbb{E} \left[ (R_j \hat{g}_j(x) - Z(i))^2 \mathbb{1}_{|R_j \hat{g}_j(x) - Z(i)| \geq \Delta'} \right].$$

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
$$\mathbb{E} \left[ \|Q_{\text{RMQ},j}(\hat{g}_j(x), Z) - \hat{g}_j(x)\|_2^2 \mathbb{1}_{|R_j \hat{g}_j(x) - Z(i)| \geq \Delta'} \right]$$

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 $\alpha(Q_1)$ and $\beta(Q_1)$ for $\log \ell_2 = \varepsilon \log \log KT$ and $\delta \leq \frac{\varepsilon}{\sqrt{\ell}}$;
$$\alpha^2(Q_1) \leq c_1 \frac{\sigma^2 \log \log KT}{\varepsilon^2}, \quad \beta^2(Q_1) \leq c_2 \frac{\sigma^2}{\varepsilon^2}.$$ The convergence proof of Theorem III.2 can be completed by using the bounds on $\alpha$, $\beta$ and using Lemma II.2.