

**ADVANCED QUANTIZATION TECHNIQUES FOR COMMUNICATION  
EFFICIENCY AND PRIVACY IN FEDERATED LEARNING, AND  
MEMORY-EFFICIENT FINE-TUNING OF LLMS**

A Dissertation  
Presented to  
The Academic Faculty

By

Yeojoon Youn

In Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy in the  
College of Engineering  
School of Electrical and Computer Engineering

Georgia Institute of Technology

Dec 2024

© Yeojoon Youn 2024

**ADVANCED QUANTIZATION TECHNIQUES FOR COMMUNICATION  
EFFICIENCY AND PRIVACY IN FEDERATED LEARNING, AND  
MEMORY-EFFICIENT FINE-TUNING OF LLMS**

Thesis committee:

Dr. Jacob Abernethy  
School of Computer Science  
*Georgia Institute of Technology*

Dr. Justin Romberg  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Vidya Muthukumar  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Dr. Juba Ziani  
School of Industrial and Systems Engineering  
*Georgia Institute of Technology*

Dr. Alexey Tumanov  
School of Computer Science  
*Georgia Institute of Technology*

Dr. Callie Hao  
School of Electrical and Computer Engineering  
*Georgia Institute of Technology*

Date approved: October 28, 2024

For my parents

## ACKNOWLEDGMENTS

I am deeply grateful to my advisor, Dr. Jacob Abernethy, for his invaluable mentorship and unwavering support throughout my PhD journey. Despite my limited background in Machine Learning due to a shift in my research focus, Jake patiently guided me, offering detailed advice on framing both theoretically and practically meaningful research problems. His mentorship helped me develop ideas into viable research projects, and his encouragement to pursue topics driven by my interests allowed me to grow into an independent researcher. I am profoundly thankful for his incredible guidance as my PhD advisor, collaborator, and life mentor over the past six years.

I also wish to extend my heartfelt gratitude to my co-advisor, Dr. Vidya Muthukumar, for her unwavering support during my PhD. My sincere thanks go to Dr. Juba Ziani for his invaluable collaboration and insightful contributions to my privacy-related project. Additionally, I would like to thank the other members of my thesis committee, Dr. Justin Romberg, Dr. Alexey Tumanov, and Dr. Callie Hao, for their generous help and guidance throughout the preparation of this dissertation.

I am also thankful to my co-authors and collaborators: Bhuvesh Kumar, Zihao Hu, Sang Keun Choe, Sehoon Kim, Suhong Moon, and Ce Zhang, for their significant contributions to the research presented in this dissertation. I truly enjoyed working alongside Taesik Na, Xiao Xiao, and Tejaswi Tenneti during my time at Instacart in 2022. Moreover, I am deeply appreciative of my friends in both the US and Korea for their constant support.

Finally, and most importantly, I extend my deepest gratitude to my parents, two sisters, and all my other family members. Their unconditional love and unwavering support have been the foundation that enabled me to successfully complete this PhD journey.

## TABLE OF CONTENTS

<b>Acknowledgments</b> . . . . .	iv
<b>List of Tables</b> . . . . .	viii
<b>List of Figures</b> . . . . .	ix
<b>List of Acronyms</b> . . . . .	xi
<b>Summary</b> . . . . .	xii
<b>Chapter 1: Introduction</b> . . . . .	1
1.1 Communication Efficiency in FL with Quantization . . . . .	4
1.2 Privacy in FL with Quantization . . . . .	5
1.3 LLM Fine-tuning with Quantization . . . . .	7
<b>Chapter 2: Accelerated Federated Optimization with Quantization</b> . . . . .	10
2.1 Introduction . . . . .	10
2.2 Preliminaries . . . . .	13
2.2.1 Related Works . . . . .	13
2.2.2 Problem Setup . . . . .	14
2.3 FedAQ Algorithm . . . . .	16
2.3.1 Multiple Accelerated Local Updates . . . . .	16

2.3.2	Communication with Quantization . . . . .	17
2.4	Convergence Analysis . . . . .	18
2.4.1	Two Parameter Condition Sets . . . . .	19
2.4.2	Proof Sketch of FedAQ Under Condition Set (2.2) . . . . .	20
2.4.3	Proof Details for FedAQ under Condition Set (2.1) . . . . .	24
2.4.4	Proof Details for FedAQ under Condition Set (2.2) . . . . .	43
2.4.5	More Theoretical Details about Remark 2.4.5 and Contribution 2 in Introduction . . . . .	60
2.5	Experiments . . . . .	64
2.5.1	Experimental Setup . . . . .	65
2.5.2	Experimental Results . . . . .	69
<b>Chapter 3: Improving Privacy-Accuracy Trade-offs in Federated Learning via Randomized Quantization Mechanism . . . . .</b>		<b>75</b>
3.1	Introduction . . . . .	75
3.2	Preliminaries . . . . .	78
3.2.1	Related work . . . . .	78
3.2.2	Differential privacy . . . . .	79
3.2.3	User-level privacy . . . . .	80
3.3	Model . . . . .	81
3.4	The Randomized Quantization Mechanism . . . . .	82
3.4.1	Randomized Quantization Mechanism . . . . .	83
3.4.2	Resulting discrete distribution of outcomes . . . . .	86
3.4.3	Analysis of RQM’s privacy guarantees . . . . .	93

3.4.4	Privacy-Accuracy Trade-offs at the User Device Level . . . . .	98
3.5	Federated Learning Experiments . . . . .	98
<b>Chapter 4: Quantization Group Adaptive NormalFloat for Low Bit Fine-tuning of LLMs . . . . .</b>		
		103
4.1	Introduction . . . . .	103
4.2	Preliminaries . . . . .	105
4.2.1	Related Work . . . . .	105
4.2.2	NormalFloat Quantization . . . . .	106
4.2.3	Low-rank Adaptation . . . . .	108
4.3	Method . . . . .	109
4.3.1	Dynamic NormalFloat with a Single Offset . . . . .	109
4.3.2	Quantization Group Adaptive NormalFloat . . . . .	111
4.4	Experiments . . . . .	112
4.4.1	Experimental Setup . . . . .	113
4.4.2	Experimental Results . . . . .	114
<b>Appendices . . . . .</b>		
		118
Appendix A:	More details about the experiments from Chapter 3 . . . . .	119
<b>References . . . . .</b>		
		122

## LIST OF TABLES

2.1	Summary of results on the convergence rate and communication required for linear speedup. $M$ is the number of devices, $T$ is the number of total parallel iterations, and $K$ is the number of communication rounds, $q$ is a quantization parameter (assumption 2.2.1), $d_{\text{quant}}$ is the number of bits used to quantize, $d_{\text{full}}$ is the number of bits required when there is no quantization ( $d_{\text{full}} \gg d_{\text{quant}}$ ). [31] and FedAQ send two iterates per communication round as other algorithms to achieve acceleration (See line 11 in Algorithm 2), we multiply $d_{\text{full}}$ and $d_{\text{quant}}$ by 2 for bits communicated for a linear speedup. The presented results of [30] are newly obtained (section 2.4.5). . . . .	11
4.1	2-bit fine-tuning quantitative results on the GLUE NLU tasks with the DeBERTaV3-base model. We compare our methods, DNF and AdaNF, against two other quantized fine-tuning baselines. N.A. means the model fails to converge. . . . .	115
4.2	Quantitative results on two NLG tasks with LLAMA-2-7B. We compare our methods, DNF and AdaNF, against three other quantized fine-tuning baselines. For reference, LoRA fine-tuning without quantization, which is not included in the table, achieves a perplexity of 5.08 on WikiText-2 and an accuracy of 38.5 on GSM8K. N.A. means the model fails to converge. . .	116

## LIST OF FIGURES

2.1	Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on MNIST with Strongly Convex Settings (first row) and Non-Convex Settings (second row). We observe how the global training loss changes across communication rounds (first column), communicated bits (second column), and human time (third column). FedAQ-I(8bits) and FedAQ(4bits) respectively outperform other algorithms for strongly convex settings and non-convex settings. FedAQ(4bits) sends the same number of communicated bits as FedPAQ(8bits) and FedCOMGATE(8bits) in each communication round, which indicates a fair comparison (See Quantization bits in section 2.5.1).	73
2.2	Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on MNIST with Strongly Convex Settings (first row) and Non-Convex Settings (second row). We observe how the test accuracy changes across communication rounds (first column), communicated bits (second column), and human time (third column). FedAQ-I outperforms other algorithms in all plots for strongly convex settings. Moreover, FedAQ(4bits) outperforms other algorithms in all plots for non-convex settings.	74
2.3	Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on CIFAR-10. We observe how the global training loss and test accuracy change across communication rounds (first column), communicated bits (second column), and human time (third column). We use a CNN model for CIFAR-10. Similar to the MNIST experiment, FedAQ (4 bits) outperforms all other algorithms in every case.	74
3.1	An example of RQM with input $x = c$ and parameters $\Delta = c$ , $m = 16$ .	88
3.2	The results about Numerical Rényi privacy with $\theta = 0.15$ (left), $\theta = 0.25$ (middle), and $\theta = 0.35$ (right). All figures indicate how the Rényi divergence increases as $\alpha$ increases.	97
3.3	Numerical privacy-MSE trade-offs of RQM and PBM with $m = 4$ (left), $m = 16$ (middle), and $m = 64$ (right).	98

3.4	Comparing RQM with PBM ( $\theta = 0.15$ ) and noise-free clipped SGD on EMNIST. All three RQMs with different hyperparameters outperform PBM in both a loss plot (Left) and an accuracy plot (Middle). These RQMs also show better Rényi DP guarantees than PBM (Right).	100
3.5	Comparing RQM with PBM and noise-free clipped SGD on EMNIST (Additional FL experiment with $\theta = 0.25$ ).	100
3.6	Comparing RQM with PBM and noise-free clipped SGD on EMNIST (Additional FL experiment with $\theta = 0.35$ ).	100
3.7	Comparing RQM with PBM, noise-free clipped SGD, centralized continuous Gaussian mechanism, and noise-free no-clipped SGD on CIFAR-100.	102
4.1	The left figure shows the log-scale weight distribution of a specific layer in LLAMA-2-7B. The red points represent the four dequantized values within a particular quantization group using the original 2-bit NormalFloat. With our redesigned NormalFloat, we obtain four blue points that are closer to the center. The right figure illustrates how the L3 norm of the quantization error varies with the CDF offset (See section 4.2.2, section 4.3) within the same quantization group. The four dequantized blue points in the left figure are obtained from an offset of 0.96, resulting in minimal quantization error in this group. By adjusting the offset, we can find the optimal value for each group that minimizes the quantization error.	104
A.1	Both the left ( $\alpha = 2$ ) and the right ( $\alpha = 1000$ ) 2d plot illustrate how the Rényi divergence $D_\alpha(P_{Q(x_1)}  P_{Q(x'_1)})$ changes with respect to the value of $x_1$ and $x'_1$ for the single-device scenario. Here, we follow the hyperparameter choice right above.	119
A.2	Both the left ( $\alpha = 2$ ) and the right ( $\alpha = 1000$ ) plot illustrate how the Rényi divergence $D_\alpha(P_{Q(x_1)}  P_{Q(-c)})$ changes as $x_1$ increases from $-c$ to $c$ for the single-device scenario. Here, we fix $x'_1 = -c$ .	120
A.3	Comparison of the Rényi Divergence of RQM and PBM for the low regimes of $\alpha$ .	120
A.4	Additional numerical privacy-MSE trade-offs of RQM and PBM with $m = 4$ (left), $m = 16$ (middle), and $m = 64$ (right), when $\alpha$ is fixed as 0.5.	121

## LIST OF ACRONYMS

- AdaNF** Quantization Group **Adaptive NormalFloat**
- CDF** Cumulative Density Function
- DP** Differential Privacy
- DPSGD** Differentially Private Stochastic Gradient Descent
- FedAQ** Federated Optimization Algorithm with **A**cceleration and **Q**uantization
- FedAvg** Federated Averaging
- FL** Federated Learning
- LLMs** Large Language Models
- LoRA** Low-Rank Adaptation
- PEFT** Parameter Efficient Fine-Tuning
- RQM** Randomized **Q**uantization **M**echanism

## SUMMARY

Quantization techniques play a crucial role in developing communication-efficient Federated Learning (FL) algorithms, as well as in optimizing training and inference for Large Language Models (LLMs). While it is evident that quantization can significantly enhance the efficiency of modern machine learning systems with thoughtful and sophisticated engineering, the theoretical understanding of why federated optimization algorithms with quantization still maintain convergence guarantees—especially when combined with other optimization techniques—remains unclear. Also, since privacy-sensitive data on local devices necessitates privacy-preserving training in FL, it is imperative to devise a well-designed quantization scheme that not only ensures communication efficiency but also aligns with the guaranteed privacy guarantees. Finally, when it comes to memory-efficient fine-tuning of LLMs, conventional quantization methods like QLoRA fall short for extremely low-bit fine-tuning tasks.

This dissertation addresses these three challenges of quantization from an optimization perspective. Specifically, it presents: 1. **Federated Optimization Algorithm with Acceleration and Quantization (FedAQ)**, which tackles the communication bottleneck in federated learning by combining an accelerated federated averaging method that reduces training and synchronization steps with an efficient quantization scheme, significantly reducing communication complexity while maintaining stronger theoretical guarantees. 2. A new algorithm called the **Randomized Quantization Mechanism (RQM)**, which maps gradients to a randomized discrete grid while preserving Rényi differential privacy. Our experiments demonstrate that this method offers improved privacy-accuracy trade-offs in federated learning compared to the previous state-of-the-art algorithm. 3. A redesign of the NormalFloat quantization data type in QLoRA, introducing Quantization Group **Adaptive NormalFloat (AdaNF)**, which dynamically adjusts the Cumulative Density Function (CDF) offset based on the statistical properties of each quantization group. This makes 2-bit

fine-tuning viable in resource-constrained environments.

# CHAPTER 1

## INTRODUCTION

Federated Learning (FL) is a large-scale distributed machine learning paradigm that allows models to be trained across multiple devices without transferring the raw privacy-sensitive data to a central server [1]. In traditional machine learning approaches, data is collected from various sources and aggregated in a central location for training, which can be impractical or undesirable due to privacy, security, or regulatory constraints. With the ever-growing volume of data generated by distributed devices, such as smartphones, wearable devices, and IoT sensors, and the corresponding increase in computational resources available on these devices, a need for decentralized computing has emerged. Distributed computing addresses the challenge of efficiently leveraging these resources without overwhelming central servers or compromising data security. Federated learning addresses this need by training models locally on each remote device and communicating only the model updates to a central server, thereby mitigating privacy risks and ensuring that sensitive data remains on the user's device. This approach not only alleviates privacy concerns but also leverages the computational power of edge devices effectively, paving the way for its use in numerous applications such as mobile keyboard prediction [2], financial fraud detection [3], and drug discovery [4].

In federated learning, one of the major challenges is the high communication cost, which arises from the frequent exchange of the large amount of data between the central server and numerous local devices. This issue is exacerbated in scenarios with limited network bandwidth, resulting in an even heavier communication burden. The iterative FL procedure aims to address this challenge by reducing the number of communication rounds through a scheme involving multiple local updates [1]. The process involves the following steps: 1) The central server selects a training model. 2) The server broadcasts the initial

model to several clients. 3) Each client trains the model locally using its own data. 4) The server then aggregates the local updates and uses them to update the global model before broadcasting the updated model back to the clients. To further enhance communication efficiency, reducing the number of bits transmitted during each communication is crucial, and *quantization* plays a significant role by enabling the transmission of compressed model updates using fewer bits.

In addition to communication efficiency, data privacy is equally critical in FL due to the sensitive nature of each device's training data. For instance, Google has applied federated learning to improve Google Keyboard (Gboard) query suggestions, ensuring that user interaction text data remains private on each mobile device [5]. At a fundamental level, privacy is preserved by keeping local data stored on each end-user device without sharing it with the central server. However, in some instances, it is possible to partially reconstruct device data from the computed gradients [6]. This potential risk of data leakage can be mitigated by implementing privacy-preserving techniques such as Differential Privacy (DP).

*Differential privacy* is a method that protects individual data points by adding noise, thereby preserving the overall statistical properties of the dataset [7]. In other contexts, such as databases, differential privacy is used to allow statistical analysis while preventing the identification of individual records, even when queries are repeated over time. In the context of machine learning, DP can be implemented using methods like Differentially Private Stochastic Gradient Descent (DPSGD). DPSGD works by adding carefully calibrated noise to the gradient updates, ensuring that information about individual data points is not leaked during training [8]. Integrating differential privacy into the federated learning framework using DPSGD is straightforward, as the technique is applied directly to the gradients computed during local training, before sharing them with the central server. Therefore, to ensure both communication efficiency and data privacy in FL, careful consideration of how to effectively incorporate differential privacy techniques such as DPSGD into the quantization scheme is essential.

Quantization also plays a vital role for Large Language Models (LLMs) in terms of memory-efficient model serving and training, which can directly lead to cost reduction and expand access to LLMs for everyday users with cheaper GPUs. The rise of LLMs has marked a transformative shift in AI, showcasing exceptional capabilities across a broad spectrum of Natural Language Processing (NLP) tasks [9, 10, 11, 12, 13, 14, 15]. LLMs excel as *few-shot learners*, meaning they can perform a variety of downstream tasks through in-context learning [16] with just a few prompted examples. Despite their generalization ability, LLMs often require fine-tuning in specific contexts to achieve optimal results. Fine-tuning involves adapting a pre-trained model to a specific domain by updating its parameters using additional training data, allowing the model to better capture domain-specific nuances. This process is particularly beneficial in highly specialized domains where detailed and specific knowledge is required—knowledge not typically covered in general training corpora [17].

While fine-tuning LLMs is highly effective in improving their performance across a variety of specialized tasks, full fine-tuning has become increasingly impractical in environments with limited GPU resources due to the substantial storage requirements of their vast number of parameters. To overcome these resource limitations, the focus has shifted towards reducing memory usage for model weights, gradients, and optimizer states [18]. Much like in federated learning, where quantization is essential for reducing communication overhead, LLMs also benefit significantly from quantization, particularly in lowering memory demands for model weights. Thus, it is crucial to achieve memory efficiency during the fine-tuning of LLMs, without compromising the performance of downstream tasks.

The contributions of this thesis on advanced quantization schemes can be categorized into three main themes:

- Communication Efficiency in FL with Quantization (**chapter 2**)
- Privacy in FL with Quantization (**chapter 3**)

- LLM Fine-tuning with Quantization (**chapter 4**)

In each respective chapter, we present a novel quantization-based algorithm for each theme. Below, we summarize our contributions and provide a high-level overview of the proposed quantization schemes.

## 1.1 Communication Efficiency in FL with Quantization

The potential for communication bottleneck is significant in FL. The two main methods to tackle this issue are (a) smarter optimization that decreases the frequency of communication rounds and (b) using *compression* techniques such as quantization and sparsification to reduce the number of bits machines need to transmit. In terms of (a), [1] proposes the first and simplest federated optimization algorithm, Federated Averaging (FedAvg), which only requires infrequent synchronization through the multiple local SGD steps of each client. Federated Averaging (FedAvg) (depicted in Algorithm 1) follows the iterative FL procedure mentioned earlier. To be specific, at the beginning of the  $k$ -th round, the central server broadcasts the current global model  $x_k$  to a random subset of clients  $\mathcal{S}^{(k)}$ , and each local model is initialized with  $x_k$ . Each selected client performs  $\tau$  local SGD updates and sends the model changes from multiple local updates  $\Delta_k^i = x_{k,\tau}^i - x_k$  to the server. Here  $x_{k,t}^i$  represents the model parameter of client  $i$  at  $k$ -th round followed by  $t$  local SGD updates. Finally, the server aggregates all local updates and updates the global model as below.

$$x_{k+1} = x_k + \frac{\sum_{i \in \mathcal{S}^{(k)}} p_i \Delta_k^i}{\sum_{i \in \mathcal{S}^{(k)}} p_i} \quad (1.1)$$

The three essential features of FedAvg are partial participation, stochastic approximation, and multiple local updates. Partial participation and multiple local updates scheme are necessary to reduce communication costs.

In chapter 2, we provide a novel algorithm, **F**ederated optimization algorithm with **A**cceleration and **Q**uantization (FedAQ), with improved theoretical guarantees by combin-

---

**Algorithm 1** Federated Averaging (FedAvg) [1]

---

```
1: Input: Initial model  $x_0$ ; local learning rate  $\eta$ 
2: for  $k = 0, 1, \dots, K - 1$  do
3:   Sample a subset  $\mathcal{S}^{(k)}$  of clients
4:   for client  $i \in \mathcal{S}^{(k)}$  in parallel do
5:     Initialize local model  $x_{k,0}^i = x_k$ 
6:     for  $t = 0, \dots, \tau - 1$  do
7:       Compute local stochastic gradient  $g_{k,t}^i \leftarrow \nabla f(x_{k,t}^i, \xi_{k,t}^i)$ ,  $\xi_{k,t}^i \sim \mathcal{D}_i$ 
8:       Perform local update  $x_{k,t+1}^i \leftarrow x_{k,t}^i - \eta g_{k,t}^i$ 
9:     end for
10:    Send  $\Delta_k^i = x_{k,\tau}^i - x_k$  to the server
11:  end for
12:  Update server model  $x_{k+1} \leftarrow x_k + \sum_{i \in \mathcal{S}^{(k)}} p_i \Delta_k^i / \sum_{i \in \mathcal{S}^{(k)}} p_i$ 
13: end for
```

---

ing an accelerated method of federated averaging, reducing the number of training and synchronization steps, with an efficient quantization scheme that significantly reduces communication complexity. We are able to further reduce the number of communication rounds through the acceleration technique. We show that in a homogeneous strongly convex setting, FedAQ achieves a linear speedup in the number of workers  $M$  with only  $\tilde{O}(M^{\frac{1}{3}})$  communication rounds, significantly smaller than what is required by other quantization-based federated optimization algorithms. Moreover, we empirically verify that our algorithm outperforms previous methods. Further details on this quantization-based algorithm are discussed in chapter 2, which draws on the following paper.

- “Accelerated Federated Optimization with Quantization”. Yeojoon Youn, Bhuvish Kumar, and Jacob Abernethy. IEEE Data Engineering Bulletin. 2023

## 1.2 Privacy in FL with Quantization

A primary motivation behind the decentralized approach in FL is data privacy, ensuring that the learner never sees the data of each local source itself. Federated learning then comes with two major challenges: one is handling potentially complex model updates between a server and a large number of data sources; the other is that de-centralization may, in fact, be

insufficient for privacy, as the local updates themselves can reveal information about the sources' data. To address these issues, we consider an approach to federated learning that combines quantization and differential privacy. Absent privacy, Federated Learning often relies on quantization to reduce communication complexity. The conventional approach [19, 20, 21] for integrating quantization with differential privacy involves using discrete additive DP noise alongside quantization. However, these methods face a significant challenge when combined with secure aggregation protocols [22], which are designed to prevent the server from accessing individual local updates. Specifically, the issue of biased estimation arises due to modular clipping. Our focus is on achieving unbiased gradient estimates, as these have been shown to be more effective for gradient-descent-based techniques in federated learning [23].

We introduce a novel quantization-based DP mechanism that achieves unbiased estimation without requiring explicit discrete noise. We employ *randomized rounding* for quantization, a scheme that stochastically truncates bits from floating point numbers in an unbiased manner. While randomized rounding inherently reduces the information encoded about the original input, quantization alone is insufficient for privacy because it always maps an input to a fixed set of two quantization levels in a deterministic manner.

In chapter 3, to address this limitation, we propose a new algorithm called the **R**andomized **Q**uantization **M**echanism (RQM), which ensures privacy through a two-level randomization process. More precisely, our RQM algorithm is comprised of three key components: (1) enlarging the output range beyond the input range and setting up evenly spaced quantization bins, (2) sub-sampling realized quantization levels, and (3) performing a randomized rounding procedure on the *sub-sampled* (and only those) discrete levels to map an input value to a quantization level. Each of these steps is crucial in ensuring the Rényi DP guarantees of the RQM. Step (1) is necessary to ensure that the quantization output for the maximum input is not always equal to the input itself, thereby preventing significant information leakage about the input. The combination of steps (2) and (3) represents the two-level randomization

process, allowing any input to be mapped to all different quantization levels with some probability. Without step (2), each input would always be deterministically mapped to a fixed set of two quantization levels, compromising privacy.

We are able to establish that our results preserve *Rényi differential privacy* (Rényi DP), a notion we’ll explore in more detail in chapter 3. We empirically study the performance of our algorithm and demonstrate that compared to previous work it yields improved privacy-accuracy trade-offs for DP federated learning. To the best of our knowledge, this is the first study that solely relies on randomized quantization without incorporating explicit discrete noise to achieve Rényi DP guarantees in Federated Learning systems. More details on this method are elaborated in chapter 3. The material of chapter 3 is based on the following paper.

- “Randomized Quantization is All You Need for Differential Privacy in Federated Learning”. Yeojoon Youn, Zihao Hu, Juba Ziani, and Jacob Abernethy. ICML 2023 Workshop in Federated Learning.

### **1.3 LLM Fine-tuning with Quantization**

To achieve memory-efficient fine-tuning of LLMs, it is essential to reduce memory consumption for weights, gradients, and optimizer states. As previously discussed, memory usage for weights can be significantly reduced by applying quantization to model weights. Additionally, the widely adopted approach for reducing memory usage related to gradients and optimizer states is Low-Rank Adaptation (LoRA). LoRA [24] is a Parameter Efficient Fine-Tuning (PEFT) technique that reduces the memory required for storing gradients and optimizer states by introducing a small set of trainable parameters while keeping the main model parameters fixed. These trainable parameters, known as adapters, are represented by factorized projections that augment the original model, thereby enabling a modified forward

pass that can be described as follows:

$$W' = W + \alpha BA$$

where  $W \in \mathbb{R}^{d \times k}$  represents the pre-trained weight matrix,  $B \in \mathbb{R}^{d \times r}$ ,  $A \in \mathbb{R}^{r \times k}$ , and  $\alpha$  is a scaling factor. It is important to note that the rank  $r$  is much smaller than  $\min(d, k)$ . During backpropagation, gradients flow through the fixed base model weights, which remain unchanged, while only the small set of parameters in the low-rank adapters are updated. Remarkably, the fine-tuning performance of LoRA closely matches that of full fine-tuning, making it an efficient alternative without compromising model effectiveness.

The integration of Quantization and LoRA [24] presents a promising avenue for the memory-efficient fine-tuning of LLMs within GPU memory constraints. QLoRA [25] successfully demonstrates high-fidelity 4-bit LoRA fine-tuning using a datatype called NormalFloat. The NormalFloat (NF) data type is based on Quantile Quantization [26], an information-theoretically optimal data type that ensures each quantization bin contains an equal number of values from the input tensor. QLoRA also uses group quantization, which involves dividing the input tensor into smaller chunks that are independently quantized, indirectly reducing the number of outliers in each group and thereby leading to smaller quantization errors. However, challenges arise with lower-bit fine-tuning, such as 2-bit, where QLoRA often struggles with convergence due to significant information loss from quantization. This occurs because, in 2-bit quantization (only 4 dequantized values), the presence of an outlier in a quantization group causes dequantization to be dominated by the group’s maximum value, resulting in unrepresentative quantized values and high quantization error. Although group quantization helps to reduce outlier effects, it is insufficient to address the severe issues encountered in 2-bit quantization. Therefore, adjusting the dequantized values to bring them closer to the center—ensuring that neither the maximum nor the minimum of the new dequantized values are outliers—can significantly reduce quantization

error and improve convergence.

In chapter 4, we address these challenges by introducing an updated version of the NormalFloat data type, which we call Dynamic NormalFloat. This new version adjusts the dequantized values based on the ratio of the quantile output of the reference CDF offset to the quantile output of our chosen CDF offset, where the CDF offset controls the range of probabilities from the standard normal distribution used to generate quantized values. By selecting a lower CDF offset, the original dequantized values are brought closer to the center through this redesigned data type.

Furthermore, we introduce quantization group **Adaptive NormalFloat** (AdaNF), which dynamically determines an optimal CDF offset for the Dynamic NormalFloat data type within each quantization group to minimize quantization error. Since each quantization group has unique statistical properties, adapting the CDF offset for each group allows for more effective information preservation during low-bit quantization compared to using a fixed offset. The quantization error between the original weight tensor and the dequantized weight tensor is measured using the  $L_p$  norm, and this error metric is employed to find the optimal CDF offset for each group through grid search. Our empirical investigations across various models and downstream tasks in the low-bit fine-tuning regime confirm that our method achieves performance comparable to existing approaches, effectively overcoming the limitations of previous techniques. Further details on this research are discussed in chapter 4, which is based on the following paper.

- “AdaNF: Quantization Group Adaptive NormalFloat for Low Bit Fine-tuning of LLMs”. Yejoon Youn, Sehoon Kim, Suhong Moon, Sang Keun Choe, and Ce Zhang. ICML 2024 Workshop on Efficient Systems for Foundation Models II.

## CHAPTER 2

### ACCELERATED FEDERATED OPTIMIZATION WITH QUANTIZATION

#### 2.1 Introduction

This chapter provides a novel algorithm, FedAQ, with improved theoretical guarantees by combining an accelerated method of federated averaging, reducing the number of training and synchronization steps, with an efficient quantization scheme that significantly reduces communication complexity. In this work, we focus on the challenge of communication efficiency that is of primary interest in cross-device settings when there is a heavy communication burden with many edge computing devices and limited network bandwidth. The two effective, widely used methods to reduce the communication cost are federated averaging optimization and randomized compression techniques.

In federated averaging (FedAvg) [1], also called *local SGD*, each client locally updates its model with multiple SGD steps, and a server aggregates model updates of clients. The server updates its own model parameters by averaging client models and then broadcasts the server parameters to all clients. This enables FL systems to achieve high communication efficiency with infrequent synchronization while showing better performance than distributed large mini-batch SGD [27]. Due to the significant empirical success of FedAvg, researchers have proposed an interesting theoretical question: To what extent can we minimize the number of synchronizations in order to both guarantee convergence and achieve linear speedup in the number of workers  $M$ <sup>1</sup>? For the strongly-convex and homogeneous settings, [28] was able to achieve a linear speedup in  $M$  with  $\tilde{O}(M)$  communication rounds, which is the state-of-the-art result for FedAvg convergence analysis. However, even with this progress on theoretical guarantees of FedAvg, it remains unclear whether further improvements on

---

<sup>1</sup>Linear speedup in the number of workers is a desirable property in parallel computing which implies that the task takes half as much time if the number of workers are doubled.

Table 2.1: Summary of results on the convergence rate and communication required for linear speedup.  $M$  is the number of devices,  $T$  is the number of total parallel iterations, and  $K$  is the number of communication rounds,  $q$  is a quantization parameter (assumption 2.2.1),  $d_{\text{quant}}$  is the number of bits used to quantize,  $d_{\text{full}}$  is the number of bits required when there is no quantization ( $d_{\text{full}} \gg d_{\text{quant}}$ ). [31] and FedAQ send two iterates per communication round as other algorithms to achieve acceleration (See line 11 in Algorithm 2), we multiply  $d_{\text{full}}$  and  $d_{\text{quant}}$  by 2 for bits communicated for a linear speedup. The presented results of [30] are newly obtained (section 2.4.5).

Algorithm	Convergence rate	Communication rounds for $\tilde{\mathcal{O}}(\frac{1}{T})$ convergence with linear speedup	Bits communicated for linear speedup
[29]	$\mathcal{O}(\frac{1+q}{K} + \frac{T}{K^2})$	Not possible	Not possible
[30]	$\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1}{TK})$	$\tilde{\mathcal{O}}(\frac{M}{1+q})$	$\tilde{\mathcal{O}}(\frac{M}{1+q}) \cdot d_{\text{quant}}$
[31]	$\tilde{\mathcal{O}}(\frac{1}{MT} + \frac{1}{TK^3})$	$\tilde{\mathcal{O}}(M^{\frac{1}{3}})$	$\tilde{\mathcal{O}}(M^{\frac{1}{3}}) \cdot 2d_{\text{full}}$
<b>FedAQ</b>	$\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$	$\tilde{\mathcal{O}}(M^{\frac{1}{3}})$	$\tilde{\mathcal{O}}(M^{\frac{1}{3}}) \cdot 2d_{\text{quant}}$

convergence time and communication efficiency can be achieved.

Applying acceleration methods to FL has led to improved convergence, with [31] providing a faster version of FedAvg with provably stronger bounds. For the strongly-convex and homogeneous setting, their algorithm achieves a linear speedup in  $M$  with only  $\tilde{\mathcal{O}}(M^{\frac{1}{3}})$  communication rounds. Hence, the accelerated version of federated averaging requires a much smaller number of communication rounds than FedAvg to achieve the same accuracy. At present, this remains the best result for strongly-convex and homogeneous local data distribution settings. In addition to reducing the required number of communication rounds, another powerful way to build communication-efficient FL systems is to reduce the number of bits that need to be transmitted at each synchronization. [29, 30] have shown that such compression techniques, which include *quantization*, reduce communication costs and guarantee convergence (See Table 2.1).

In this work, we provide a novel algorithm, **F**ederated optimization algorithm with **A**cceleration and **Q**uantization (FedAQ), to solve the severe communication bottleneck problem in FL systems. FedAQ is the first federated optimization algorithm that successfully incorporates *multiple local update schemes*, *acceleration*, and *quantization* for master-worker topology. Although these three key desiderata of Federal Learning systems have

individually been shown to build communication-efficient FL systems, it is not obvious if or how acceleration techniques can lead to faster convergence even for quantization based methods. We answer this question by showing that FedAQ converges for strongly-convex and homogeneous local data distribution settings without any additional strong assumptions.

Let  $T$  be the number of total parallel iterations,  $K$  be the number of total communication rounds. We compare our results to previous methods in Table 2.1, and highlight the following contributions:

1. FedAQ has a convergence rate of  $\tilde{O}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$  which is better than the  $\tilde{O}(\frac{1+q}{MT} + \frac{1}{TK})$  convergence of [30], the state of the art in quantization based methods. Here  $q$  is a parameter that measures the effectiveness of the quantization scheme (see assumption 2.2.1). This allows FedAQ to obtain linear speedup with only  $\tilde{O}(M^{\frac{1}{3}})$  communication rounds whereas [30] requires  $\tilde{O}(\frac{M}{1+q})$  rounds. The faster convergence in number of communication rounds also implies that FedAQ can achieve better convergence than [30] by using many fewer communication rounds. Thus, although FedAQ sends two iterates in each communication round, that is the bits communicated in each round are twice many compared to [30] for the same level of quantization, FedAQ requires much smaller total communication costs due to the large reduction in synchronization rounds.
2. When comparing FedAQ to Accelerated Federated learning, we observe that FedAQ has similar convergence and requires the same number of communication rounds as [31]. In each communication round of [31], every client sends the complete iterates to the server without any quantization. To effectively obtain a convergence rate of  $\tilde{O}(\frac{1}{MT})$ , it needs to send each value with a precision of  $\tilde{O}(\frac{1}{MT})$ , requiring  $d_{\text{full}} = \mathcal{O}(\log(MT))$  bits. In comparison, if we use the low precision quantizer (Example 1) given by [32], FedAQ needs to send only  $d_{\text{quant}} = \mathcal{O}(\log \frac{1}{q})$  bits for each value. Since  $q$  is a constant,  $d_{\text{quant}} \ll d_{\text{full}}$ . The extra  $1 + q$  term in the convergence for FedAQ can be offset by scaling the number of local updates by  $1 + q$ , which is cheaper

than expensive data communication. Thus, FedAQ obtains the same convergence as [31] using as many communication rounds but by sending many fewer bits per round.

Finally, we empirically verify that our algorithm exhibits better performance than baselines, FedPAQ [29], FedCOMGATE [30], FedAC [31], and FedAvg [1] on classical vision datasets such as MNIST and CIFAR-10.

## 2.2 Preliminaries

### 2.2.1 Related Works

The first guarantee for FedAvg, showing that it converges at the same rate as mini-batch SGD in strongly convex scenarios, was shown by [33] in the IID setting. The further convergence analysis of FedAvg for non-convex functions was laid out in a number of published works [34, 35, 36]. Followup work has managed to remove unnecessary assumptions, such as uniformly bounded gradients, to achieve better convergence rates [34, 37, 38, 28, 39]. Moreover, [40, 41, 42, 28, 43] define scenarios that depart from the IID framework, analyzing the convergence of FedAvg and its variants in settings with heterogeneous data distributions.

Reducing the transmitted bits between a server and clients through compression techniques is pivotal to saving communication costs in federated learning. This motivates researchers to develop various compression techniques such as sparsification and quantization without significantly sacrificing accuracy [44, 32, 45, 46, 47, 48, 49, 50, 51, 52]. [29] show near-optimal theoretical guarantees of the first federated optimization algorithm that incorporates federated averaging, partial node participation, and quantization in homogeneous local data distribution settings. [30] further provide improved convergence rates for both homogeneous and heterogeneous settings.

We can achieve better communication efficiency by applying acceleration methods into client updates. [31] have proposed the first provable acceleration of FedAvg that achieves a

linear speedup with the fewest communication rounds. Several other works aim to achieve communication efficiency by using momentum or adaptive optimizers [53, 54, 55]. It is important to note that our work is not the first to combine acceleration and quantization. [56, 57], for example, propose compressed and accelerated distributed optimization methods that are neither stochastic nor FedAvg variants. [58] propose communication efficient momentum SGD for decentralized optimization. [59, 60] show that distributed and federated versions of adaptive optimizers along with gradient compression can lead to similar convergence rates as their non-compressed counterparts. But these works do not achieve the core result of our work, which is the reduced communication complexity via a faster convergence rate and a linear speedup with the small number of communication rounds. To the best of our knowledge, FedAQ is the first accelerated version of federated averaging for master-worker topology that successfully integrates a quantization scheme and provides rigorous convergence guarantees.

### 2.2.2 Problem Setup

In this research, we build our algorithm based on federated learning with captain-worker topology where  $M$  local devices contain their own local data, and a server aggregates local parameter updates without sharing any data during synchronization rounds. Since we focus on *homogeneous* local data distribution settings for the convergence analysis of our algorithm, we define the distributed stochastic optimization problem as below.

$$\min_{w \in \mathbb{R}^d} F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$$

In our convergence analysis, we assume  $F$  is *strongly-convex*. Each client can access  $F$  at  $w$  via oracle  $\nabla f(w; z)$  because all clients have the same loss function  $f$ . Also, every local device has the same local data distribution  $\mathcal{D}$ . Moreover, we use the *full participation* of nodes for local updates and synchronizations.

### Assumptions

Let us clarify assumptions on the unbiased quantizer  $Q$ , the global objective function  $F$ , and the unbiased gradient estimator  $\nabla f$ .

**Assumption 2.2.1.** The variance of the unbiased quantizer  $Q$  is bounded by the squared of  $l_2$ -norm of its argument, i.e.,  $\mathbb{E}[Q(x)|x] = x$ ,  $\mathbb{E}[\|Q(x) - x\|^2|x] \leq q\|x\|^2$ .

For example, a well-known randomized quantizer which satisfies assumption 2.2.1 is low-precision quantizer in [32].

**Example 1.** (Low-precision quantizer) Given  $x \in \mathbb{R}^d$ , the quantizer  $Q : \mathbb{R}^d \rightarrow \mathbb{R}^d$  is defined by

$$Q_i(x) = \text{sign}(x_i) \cdot \|x\| \cdot \xi_i(x, s), \quad i \in [d]$$

$\xi_i$  is defined as below.

$$\xi_i(x, s) = \begin{cases} \frac{l+1}{s}, & \text{with probability } \frac{|x_i|}{\|x\|}s - l \\ \frac{l}{s}, & \text{o/w} \end{cases}$$

$s$  is the number of quantization levels.  $l \in [0, s)$  is an integer which satisfies  $\frac{|x_i|}{\|x\|} \in [\frac{l}{s}, \frac{l+1}{s})$ .

**Assumption 2.2.2.**  $F$  is  $\mu$ -strongly convex, i.e.,  $F(w_1) \geq F(w_2) + \langle \nabla F(w_2), w_1 - w_2 \rangle + \frac{1}{2}\mu\|w_1 - w_2\|^2$  for any  $w_1, w_2 \in \mathbb{R}^d$ .

**Assumption 2.2.3.**  $F$  is  $L$ -smooth, i.e.,  $F(w_1) \leq F(w_2) + \langle \nabla F(w_2), w_1 - w_2 \rangle + \frac{1}{2}L\|w_1 - w_2\|^2$  for any  $w_1, w_2 \in \mathbb{R}^d$ .

**Assumption 2.2.4.**  $\nabla f(w; \xi)$  is unbiased and variance bounded, i.e.,  $\mathbb{E}_\xi[\nabla f(w; \xi)] = \nabla F(w)$ ,  $\mathbb{E}_\xi[\|\nabla f(w; \xi) - \nabla F(w)\|^2] \leq \sigma^2$  for any  $w \in \mathbb{R}^d$ .

## Notation

We use  $\tau, K$  to respectively denote the number of local updates and total communication rounds, which means the total number of iterations  $T$  at each node satisfies  $T = K\tau$ . Since we consider a strongly-convex case, we can find the optimal point  $w^*$  and denote the optimal function value as  $F^* := F(w^*)$ . The local parameter  $w_{k,t}^m$  indicates the parameter of the  $m$ -th local model after  $k$ th synchronization followed by  $t$  local SGD updates. There are other types of parameters such as  $w_{k,t}^{\text{ag},m}$  and  $w_{k,t}^{\text{md},m}$ , and we obtain two types of parameters  $w_k$  and  $w_k^{\text{ag}}$  in the server side after  $k$ th synchronization. More details on these parameters will be discussed in the next section.

## 2.3 FedAQ Algorithm

We propose a novel communication efficient algorithm that combines an accelerated variant of federated averaging and an efficient quantization scheme. Our FedAQ algorithm has two main parts: (1) multiple accelerated local updates and (2) communication with quantization. Both components contribute to achieving better communication efficiency than other previous federated algorithms. The entire process is summarized in Algorithm 2.

### 2.3.1 Multiple Accelerated Local Updates

The FedAvg algorithm, proposed by [1], is widely used for federated learning to improve communication efficiency by reducing communication rounds with multiple local SGD updates. [31] provide FedAC that replaces the stochastic gradient updates of FedAvg by accelerated version of SGD by [61] resulting in a linear speedup in  $M$  with fewer communication rounds than FedAvg.

Thus, we apply the FedAC scheme to multiple updates of each local model. Since previous quantization-based federated optimization algorithms are FedAvg variants with no acceleration, the accelerated method enables our algorithm to gain better communication

efficiency than others.

As you can see in Algorithm 2, we need two more local parameters  $w_{k,t}^{\text{ag},m}$  and  $w_{k,t}^{\text{md},m}$  for acceleration in addition to the main local parameter  $w_{k,t}^m$ .  $w_{k,t}^{\text{ag},m}$  aggregates the past iterates, and the gradients are queried at the auxiliary parameter  $w_{k,t}^{\text{md},m}$ . While typical FL algorithms without acceleration only have a learning rate  $\eta$  as their hyperparameter, the general acceleration scheme makes our algorithm flexible due to four hyperparameters  $\alpha, \beta, \eta, \gamma$ .  $\alpha, \beta$  are hyperparameters related to coupling coefficients, and  $\eta, \gamma$  stand for learning rates respectively for  $w_{k,t}^{\text{ag},m}, w_{k,t}^m$ . The flexibility of hyperparameters enables the fast convergence speed of FedAQ, but naively chosen hyperparameters also cause unstable training of FedAQ. We discuss the exact choice of hyperparameters in section 2.4. Unlike FedAC, that requires each client to communicate the exact iterates to the server with high precision, we discuss in the following subsection how FedAQ incorporates quantization techniques to reduce communication cost.

### 2.3.2 Communication with Quantization

In cross-device federated learning, a large amount of communicated messages from a number of devices and the limited communication bandwidth can lead to severe communication bottlenecks. Therefore, in this scenario, an efficient quantization scheme can significantly reduce the size of communicated messages and make communication between local devices and a server faster. We apply the same unbiased quantizer used in [30] that satisfies assumption 2.2.1.

In contrast with other quantization-based federated optimization algorithms [29, 30], the algorithmic novelty of FedAQ is based on applying quantization to two model parameter updates, which is required in order to simultaneously reduce the frequency of communication and the volume of communicated bits. To the best of our knowledge, this is the first quantization-based method that achieves the accelerated rate with the dramatic reduction in communication cost. To be specific on the communication process, after each client  $m$

obtains  $w_{k,\tau}^m, w_{k,\tau}^{\text{ag},m}$  through  $\tau$  accelerated local iterations, each client quantizes the difference between  $w_{k,\tau}^m, w_{k,\tau}^{\text{ag},m}$  and the most recent server models  $w_k, w_k^{\text{ag}}$ . Then, a server aggregates  $Q(w_{k,\tau}^m - w_k), Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})$  from all clients. After dequantizing those messages, the server obtains the following new models  $w_{k+1}, w_{k+1}^{\text{ag}}$  and broadcasts them back to each client.

---

**Algorithm 2** Federated Accelerated SGD with Quantization (FedAQ)

---

```

1: Input:  $\alpha, \beta, \eta, \gamma$ , initial vector  $w_0 = w_{0,0}^{\text{ag},m} = w_{0,0}^m$  for all devices  $m \in [M]$ 
2: for  $k = 0, \dots, K - 1$  do
3:   for each client  $m$  in parallel do
4:      $w_{k,0}^m \leftarrow w_k, w_{k,0}^{\text{ag},m} \leftarrow w_k^{\text{ag}}$ 
5:     for  $t = 0, \dots, \tau - 1$  do
6:        $w_{k,t}^{\text{md},m} \leftarrow \beta^{-1}w_{k,t}^m + (1 - \beta^{-1})w_{k,t}^{\text{ag},m}$ 
7:        $g_{k,t}^m \leftarrow \nabla f(w_{k,t}^{\text{md},m}, \xi_{k,t}^m)$ 
8:        $w_{k,t+1}^{\text{ag},m} \leftarrow w_{k,t}^{\text{md},m} - \eta g_{k,t}^m$ 
9:        $w_{k,t+1}^m \leftarrow (1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma g_{k,t}^m$ 
10:    end for
11:    send  $Q(w_{k,\tau}^m - w_k), Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})$ 
12:  end for
13:  server finds  $w_{k+1} \leftarrow w_k + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^m - w_k), w_{k+1}^{\text{ag}} \leftarrow w_k^{\text{ag}} + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})$ 
14: end for

```

---

## 2.4 Convergence Analysis

The rigorous theoretical guarantees of reducing communication complexity under strongly-convex and homogeneous assumptions should come first to ensure the significance of FedAQ as one of the standards of communication-efficient federated optimization algorithms. Proving convergence guarantees of FedAQ even under these assumptions requires careful consideration of the approximation error induced by the quantization scheme combined with the convergence analysis of acceleration based methods. To recall, in FedAQ the server aggregates two quantized local updates  $Q(w_{k,\tau}^m - w_k), Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})$  from all clients (See line 11 in Algorithm 2) in each round. If we simply try to generalize the convergence guarantee of FedAC to incorporate the quantization variance costs, the proof

techniques from earlier quantization-based methods cannot be directly applied, as we now have two additional quantization error terms that contribute to the overall cost. A significant amount of additional effort is required in order to account for this new quantization error.

In this section, we first define two condition sets of hyperparameters used for the convergence analysis of FedAQ. Then, we provide the proof sketch of FedAQ under one such condition set that leads to the better convergence rate  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$ . The full proofs of lemmas, theorems, and corollaries under both condition sets are elaborated in section 2.4.3 and section 2.4.4. Finally, we discuss how we obtain the new convergence rate for [30] and look into more theoretical details on contribution item 2 in Introduction.

#### 2.4.1 Two Parameter Condition Sets

We carefully determine two parameter condition sets that theoretically ensure the convergence guarantees. The first one is

$$\eta, \gamma \in \left(0, \frac{1}{L}\right], \gamma = \max\left(\sqrt{\frac{\eta}{\mu\tau}}, \eta\right), \alpha = \frac{1}{\gamma\mu}, \beta = \alpha + 1 \quad (2.1)$$

We add one more condition  $\gamma \in (0, \frac{1}{L}]$  to the FedAC-I condition [31] and create our parameter condition set (eq. (2.1)). The second one is

$$\eta, \gamma \in \left(0, \frac{1}{L}\right], \gamma = \max\left(\sqrt{\frac{\eta}{\mu\tau}}, \eta\right), \alpha = \frac{3}{2\gamma\mu} - \frac{1}{2}, \beta = \frac{2\alpha^2 - 1}{\alpha - 1}, \gamma\mu \leq \frac{3}{4} \quad (2.2)$$

We add two more conditions  $\gamma \in (0, \frac{1}{L}]$  and  $\gamma\mu \leq \frac{3}{4}$  to the FedAC-II condition to build our parameter condition set (eq. (2.2)). Even though quantization adds complexity to the algorithm, these weak assumptions are the only additional requirements for showing the convergence of FedAQ. Moreover, although the better convergence rate  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$  is obtained from the condition set (eq. (2.2)), we also analyze the convergence of FedAQ under the condition set (eq. (2.1)) because this set empirically leads to more stable training and better performance in experiments than the condition set (eq. (2.2)) (See Strongly convex

case in section 2.5.2). The intuition of the less stable training of FedAQ under the condition set (eq. (2.2)) comes from larger  $\alpha, \beta$  than those of the condition set (eq. (2.1)). If  $\alpha, \beta$  are too large,  $\alpha^{-1}, \beta^{-1}$  in Algorithm 2 cannot be used as proper coupling coefficients for local parameters  $w_{k,t}^m, w_{k,t}^{\text{ag},m}, w_{k,t}^{\text{md},m}$ . This results in aggressive updates and less stable training behavior.

#### 2.4.2 Proof Sketch of FedAQ Under Condition Set (2.2)

The decentralized potential  $\Phi_{k,t}$  [31] is used for our convergence analysis. People commonly use this potential for acceleration analysis [62].

$$\Phi_{k,t} = F(\bar{w}_{k,t}^{\text{ag}}) - F^* + \frac{1}{6}\mu\|\bar{w}_{k,t} - w^*\|^2$$

$\bar{w}_{k,t}$  and  $\bar{w}_{k,t}^{\text{ag}}$  is respectively the average of  $w_{k,t}^m$  and  $w_{k,t}^{\text{ag},m}$  for all  $m$ . Here, we additionally define  $\Phi_k$  as below.

$$\Phi_k := \Phi_{k,0} = F(w_k^{\text{ag}}) - F^* + \frac{1}{6}\mu\|w_k - w^*\|^2$$

Since  $w_k$  and  $w_k^{\text{ag}}$  are parameters obtained after  $k$ th synchronization in a server side,  $\Phi_k$  can be considered as the potential of server models.  $\Phi_k$  is essential to show the convergence of FedAQ because there is the computation of the quantizer between  $\Phi_{k-1,\tau}$  and  $\Phi_{k,0}$ . Thus, we should not naively track  $\Phi_{k,t}$  but track  $\Phi_k$  for our analysis. Obtaining  $\Phi_k \leq \epsilon$  would imply that  $F(w_k^{\text{ag}}) - F^* \leq \epsilon$  and since  $F^* \leq F(w_k^{\text{ag}})$ , it would also imply that  $\|w_k - w^*\|^2 = O(\epsilon)$ , thus obtaining convergence in terms of both the objective value and the iterate.

Our goal is to show the convergence of FedAQ and derive the simplified convergence rate so that we can get the number of communication rounds to achieve a linear speedup in  $M$ . As the first step to show this, we prove lemma 2.4.1 which represents the relationship between two consecutive server potential functions  $\Phi_k$  and  $\Phi_{k+1}$ . The quantization scheme amplifies the instability to the convergence of FedAQ in addition to the effect of acceleration.

Despite this challenge, we derive lemma 2.4.1 with the help of subtle Propositions (See section 2.4.4).

**Lemma 2.4.1.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.1, assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{3}{2\gamma\mu} - \frac{1}{2}$ ,  $\beta = \frac{2\alpha^2-1}{\alpha-1}$ ,  $\gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ ,  $\eta, \gamma \in (0, \frac{1}{L}]$ ,  $\gamma\mu \leq \frac{3}{4}$ ,  $\tau \geq 2$ , FedAQ yields*

$$\begin{aligned} \mathbb{E}[\Phi_{k+1}] &\leq \\ D(\gamma, \tau)\mathbb{E}[\Phi_k] &+ \left(\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}\right) \frac{\tau \sigma^2}{M} + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{md}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{md,m})\|^2] \\ &+ \underbrace{\frac{q}{M} \left(\frac{\gamma^2 \mu}{3} + \eta^2 L\right) \tau \sigma^2 + \frac{q}{2M} \left((\gamma - \eta)^2 \gamma^2 \mu^2 \left(\frac{\mu}{3} + \frac{L}{4}\right) + \gamma^4 \left(\frac{\mu}{3} + L\right)^2 L\right) \tau^3 \sigma^2}_{\text{additional terms due to quantization}} \end{aligned}$$

Where  $D(\gamma, \tau)$  is defined as

$$D(\gamma, \tau) = \left(1 - \frac{1}{3}\gamma\mu\right)^\tau + \underbrace{\frac{q}{M} \left(\gamma^2 \mu \left(\frac{8}{3}\mu + 2L\right) + 2\gamma^2 L \left(\frac{\mu}{3} + L\right)\right) \tau^2}_{\text{additional terms due to quantization}}$$

We get the inequality between  $\Phi_k$  and  $\Phi_{k+1}$  by finding the upper bounds of error terms due to multiple( $\tau$ ) local steps and the quantization step. The upper bound of the error caused by multiple local steps is obtained with the help of the analysis in [31] (See proposition 2.4.15). Also, we get the tight upper bound of the error due to quantization with our new proof techniques (See proposition 2.4.16, proposition 2.4.17, proposition 2.4.18). The key challenge in bounding the quantization error terms comes from representing the upper bound of variances of the quantizer  $Q$  on two local updates  $w_{k,\tau}^m - w_k, w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}$  in the form of a server potential  $\Phi_k$ . Some terms in lemma 2.4.1 are similar to those in Lemma C.2 of the FedAC paper [31], but our lemma contains additional terms that emerge from the quantization scheme.

For the next step, by telescoping lemma 2.4.1, we obtain the main theoretical result theorem 2.4.2. theorem 2.4.2 represents how  $\Phi_K$  decreases from the initial potential  $\Phi_0$  as a

communication round  $K$  increases. Since we aim to telescope lemma 2.4.1,  $D(\gamma, \tau)$  should be smaller than 1. Specifically, we show  $D(\gamma, \tau) \leq 1 - \frac{1}{6}\gamma\mu\tau$  with condition (eq. (2.3)) (See section 2.4.4). That's why theorem 2.4.2 requires the learning rate  $\gamma$  to satisfy the certain condition (eq. (2.3)).

**Theorem 2.4.2.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.1, assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for the parameter condition set (eq. (2.2)),  $\tau \geq 2$ , if the learning rate  $\gamma$  satisfies*

$$\left(\frac{1}{9}\mu^2 + \frac{q}{M}\left(\mu\left(\frac{8}{3}\mu + 2L\right) + 2L\left(\frac{\mu}{3} + L\right)\right)\right)\gamma\tau \leq \frac{1}{6}\mu \quad (2.3)$$

*FedAQ yields*

$$\begin{aligned} \mathbb{E}[\Phi_K] \leq & \exp\left(-\frac{1}{6}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Phi_0 + \frac{2(2q+1)\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{8(q+25)\eta^2L^2\tau\sigma^2}{\mu} \\ & + \frac{3q\left(\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + L\left(\frac{\mu}{3} + L\right)^2\right)\eta^{\frac{3}{2}}\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{5}{2}}M} + \frac{3qL\left(\frac{\mu}{3} + L\right)^2\eta^3\tau^2\sigma^2}{\mu M} \end{aligned}$$

We get the convergence rate of FedAQ with respect to  $\eta$  under the condition set (eq. (2.2)). The final step is to tune  $\eta$  appropriately and obtain a more intuitive form of convergence rate that we can easily analyze a linear speedup in  $M$ . The exact form of this can be found in corollary 2.4.19. Here, we introduce the simplified form of corollary 2.4.19.

**Corollary 2.4.3.** *(Simplified form of corollary 2.4.19) Note that  $T = K\tau$ . For  $\eta = \min(\frac{1}{L}, \tilde{\Theta}(\frac{\tau}{\mu T^2}))$ , FedAQ yields*

$$\begin{aligned} \mathbb{E}[\Phi_K] \leq & \min\left(\exp\left(-\frac{\mu T}{6L}\right), \exp\left(-\frac{\mu^{\frac{1}{2}}T}{6L^{\frac{1}{2}}\tau^{\frac{1}{2}}}\right)\right)\Phi_0 \\ & + \tilde{\mathcal{O}}\left(\underbrace{\frac{(1+q)\sigma^2}{\mu MT}}_I + \underbrace{\frac{(1+q)L^2\tau^3\sigma^2}{\mu^3 T^4}}_{II} + \underbrace{\frac{qL^3\tau^2\sigma^2}{\mu^4 MT^3}}_{III}\right) \end{aligned}$$

The convergence rate of FedAQ under the condition set (eq. (2.1)) is obtained in a

similar way. The convergence analysis under the condition set (eq. (2.1)) is elaborated as lemma 2.4.6, theorem 2.4.12, and corollary 2.4.13 in section 2.4.3.

*Remark 2.4.4.* The above convergence rate is worse than the convergence rate of FedAC-II according to Theorem C.13 in [31] because there are additive terms related to the quantization noise  $q$  in our case. Let's figure out the dominant terms with  $\tilde{\mathcal{O}}$  notation from the above convergence rate. Here, we replace  $\tau$  with  $\frac{T}{K}$ . At first, we can ignore the first term because it decreases exponentially. The second term I would be  $\tilde{\mathcal{O}}(\frac{1+q}{MT})$ . Then, the third term II becomes  $\tilde{\mathcal{O}}(\frac{(1+q)\tau^3}{T^4}) = \tilde{\mathcal{O}}(\frac{1+q}{TK^3})$ . Finally, the last term III turns into  $\tilde{\mathcal{O}}(\frac{q\tau^2}{MT^3}) = \tilde{\mathcal{O}}(\frac{q}{MTK^2})$ . Thus, the overall convergence rate of FedAQ under the condition set (eq. (2.2)) would be  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$ . Similarly, we obtain the simplified convergence rate of FedAQ under the condition set (eq. (2.1)) from three terms (2.14), (2.15), (2.16) of corollary 2.4.13. In this case, the convergence rate of FedAQ is  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1}{TK^2})$ , and the required number of communication rounds to achieve a linear speedup in  $M$  is  $\tilde{\mathcal{O}}((\frac{M}{1+q})^{\frac{1}{2}})$ .

*Remark 2.4.5.* As we mention above, FedAQ converges at rate  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1+q}{TK^3})$ , which is better than the convergence rate of [30]  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1}{TK})$ . To our knowledge, [30] obtain the best convergence rate among previous quantization-based federated optimization algorithms. Actually, in the strongly-convex and homogeneous case, [30] provide different convergence rate  $\mathcal{O}(\frac{1}{\gamma^2\tau} + \frac{(q+1)}{(\frac{q}{M}+1)\tau M}) = \mathcal{O}(\frac{K}{\gamma^2T} + \frac{(q+1)K}{(\frac{q}{M}+1)TM})$ , where  $\gamma$  is a learning rate for the server updates. They achieve this convergence rate by tuning  $\eta = \frac{1}{2L(\frac{q}{M}+1)\tau\gamma}$ . However, we cannot say this algorithm achieves a linear speedup in this scenario. That's why we provide a new convergence rate  $\tilde{\mathcal{O}}(\frac{1+q}{MT} + \frac{1}{TK})$  for [30] by tuning  $\eta$  in a different way. This new  $\eta$  makes this algorithm achieve a linear speedup. Why the original  $\eta$  cannot achieve a linear speedup and how we get new  $\eta$  can be found in section 2.4.5.

### 2.4.3 Proof Details for FedAQ under Condition Set (2.1)

Before diving into proof details, we define  $\bar{w}_{k,\tau}, \bar{w}_{k,\tau}^{\text{ag}}, \Psi_{k,t}^m, \Psi_{k,t}, \Psi_k, A_{k,t}^m$  as below.

$$\begin{aligned}\bar{w}_{k,\tau} &= \frac{1}{M} \sum_{m=1}^M w_{k,\tau}^m, \quad \bar{w}_{k,\tau}^{\text{ag}} = \frac{1}{M} \sum_{m=1}^M w_{k,\tau}^{\text{ag},m} \\ \Psi_{k,t}^m &= F(w_{k,t}^{\text{ag},m}) - F^* + \frac{1}{2}\mu \|w_{k,t}^m - w^*\|^2 \\ \Psi_{k,t} &= \frac{1}{M} \sum_{m=1}^M F(w_{k,t}^{\text{ag},m}) - F^* + \frac{1}{2}\mu \|\bar{w}_{k,t} - w^*\|^2 \\ \Psi_k &:= \Psi_{k,0} = F(w_k^{\text{ag}}) - F^* + \frac{1}{2}\mu \|w_k - w^*\|^2 \\ A_{k,t}^m &= \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma\mu)^2} \|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + \gamma^2 (\mu + L) \frac{2L}{1 + \gamma\mu} \Psi_{k,t}^m\end{aligned}$$

The above notations are essential to our convergence analysis. Intuitively, if the FedAQ algorithm converges to the optimal point,  $\bar{w}_{k,\tau}, \bar{w}_{k,\tau}^{\text{ag}}$  become  $w^*$ , and  $\Psi_{k,t}^m, \Psi_{k,t}, \Psi_k, A_{k,t}^m$  become 0. In order to denote the  $\sigma$ -algebra generated by  $\{w_{k',t'}^m, w_{k',t'}^{\text{ag},m}\}_{(k' < k) \text{ or } (k'=k, t' \leq t), m \in [M]}$ , we use  $\mathcal{F}_{k,t}$ .

*Proof of lemma 2.4.6*

**Lemma 2.4.6.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.1, assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{1}{\gamma\mu}, \beta = \alpha + 1, \gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}], \eta, \gamma \in (0, \frac{1}{L}], \tau \geq 2$ , FedAQ yields*

$$\begin{aligned}\mathbb{E}[\Psi_{k+1}] &\leq C(\gamma, \tau) \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M}) \tau \sigma^2 + \gamma \mu L \tau \\ &\quad \cdot \max_{0 \leq t < \tau} \mathbb{E} \left[ \frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{md} - w_{k,t}^{md,m}\| \left\| \frac{1}{1 + \gamma\mu} (\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu} (\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m}) \right\| \right] \\ &\quad + \underbrace{\frac{q}{M} (\gamma^2 \mu + \eta^2 L) \tau \sigma^2 + \frac{q}{2M} \left( \frac{(\gamma - \eta)^2 \gamma^2 \mu^2 (\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4 (\mu + L)^2 L}{1 + \gamma\mu} \right)}_{\text{Additional terms due to quantization}} \tau^3 \sigma^2\end{aligned}$$

Where  $C(\gamma, \tau)$  is defined as

$$C(\gamma, \tau) = (1 - \gamma\mu)^\tau + \underbrace{\frac{q}{M} \left( \frac{4\gamma^2\mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu} \right)}_{\text{Additional terms due to quantization}} \tau^2$$

In this section, we first introduce five crucial Propositions for proving lemma 2.4.6. Then, we prove lemma 2.4.6 by using Propositions in the last part of this section.

**Proposition 2.4.7.** *Let assumption 2.2.1 hold and consider any  $k$  synchronization round. Then, we can decompose the expectation as follows:*

$$\begin{aligned} \mathbb{E}[\|w_{k+1} - w^*\|^2] &= \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] + \mathbb{E}[\|\bar{w}_{k,\tau} - w^*\|^2] \\ \mathbb{E}[F(w_{k+1}^{ag}) - F^*] &= \mathbb{E}[F(w_{k+1}^{ag}) - \frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{ag,m})] + \mathbb{E}[\frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{ag,m}) - F^*] \end{aligned}$$

*Proof of proposition 2.4.7* The second equality is trivial. Let's focus on the first equality. By assumption 2.2.1, the quantizer  $Q$  is unbiased and we get,

$$\mathbb{E}_Q[w_{k+1}] = w_k + \frac{1}{M} \sum_{m=1}^M \mathbb{E}_Q Q(w_{k,\tau}^m - w_k) = \frac{1}{M} \sum_{m=1}^M w_{k,\tau}^m = \bar{w}_{k,\tau}$$

Thus, we finally obtain

$$\begin{aligned} \mathbb{E}[\|w_{k+1} - w^*\|^2] &= \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau} + \bar{w}_{k,\tau} - w^*\|^2] \\ &= \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] + \mathbb{E}[\|\bar{w}_{k,\tau} - w^*\|^2] \end{aligned}$$

**Proposition 2.4.8.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{1}{\gamma\mu}, \beta = \alpha + 1, \gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}], \eta \in (0, \frac{1}{L}], \text{FedAQ}$*

yields

$$\begin{aligned} \mathbb{E}[\Psi_{k,\tau}] &\leq (1 - \gamma\mu)^\tau \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 + \gamma\mu L\tau \\ &\quad \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{md} - w_{k,t}^{md,m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{ag} - w_{k,t}^{ag,m})\|] \end{aligned}$$

*Proof of proposition 2.4.8* We refer to the proof of Lemma B.2 in [31]. There is no quantization between  $\Psi_{k,\tau}$  and  $\Psi_k$ . Thus, we can directly apply useful inequalities in the proof of Lemma B.2 in [31] to our proof. Then, we obtain

$$\begin{aligned} \mathbb{E}[\Psi_{k,t+1} | \mathcal{F}_{k,t}] &\leq (1 - \gamma\mu)\Psi_{k,t} + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\sigma^2 + \gamma\mu L \\ &\quad \cdot \frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{md} - w_{k,t}^{md,m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{ag} - w_{k,t}^{ag,m})\| \end{aligned}$$

From the above relationship between  $\Psi_{k,t+1}$  and  $\Psi_{k,t}$ , we get

$$\begin{aligned} \mathbb{E}[\Psi_{k,\tau}] &\leq (1 - \gamma\mu)^\tau \mathbb{E}[\Psi_k] + \left( \sum_{t=0}^{\tau-1} (1 - \gamma\mu)^t \right) \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\sigma^2 + \gamma\mu L \cdot \sum_{t=0}^{\tau-1} \left\{ (1 - \right. \\ &\quad \left. \gamma\mu)^{\tau-t-1} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{md} - w_{k,t}^{md,m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{ag} - w_{k,t}^{ag,m})\|] \right\} \\ &\leq (1 - \gamma\mu)^\tau \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 + \gamma\mu L\tau \\ &\quad \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{md} - w_{k,t}^{md,m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{ag} - w_{k,t}^{ag,m})\|] \end{aligned}$$

**Proposition 2.4.9.** *Let assumption 2.2.1 hold. Then, we have*

$$\begin{aligned} \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] &\leq \frac{q}{M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] \\ \mathbb{E}[F(w_{k+1}^{ag}) - \frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{ag,m})] &\leq \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{ag,m} - w_k^{ag}\|^2] \end{aligned}$$

*Proof of proposition 2.4.9* First, let's consider the first inequality. According to

assumption 2.2.1, we get

$$\begin{aligned}
\mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] &= \mathbb{E}[\|w_k + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^m - w_k) - \frac{1}{M} \sum_{m=1}^M w_{k,\tau}^m\|^2] \\
&= \mathbb{E}[\|\frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^m - w_k) - (w_{k,\tau}^m - w_k)\|^2] \\
&= \frac{1}{M^2} \sum_{m=1}^M \mathbb{E}[\|Q(w_{k,\tau}^m - w_k) - (w_{k,\tau}^m - w_k)\|^2] \leq \frac{q}{M^2} \sum_{m=1}^M \mathbb{E}\|w_{k,\tau}^m - w_k\|^2
\end{aligned}$$

The third equality comes from the unbiasedness of  $Q$ , and the last inequality stems from the variance assumption of  $Q$ . Similarly, we obtain

$$\begin{aligned}
&\mathbb{E}[F(w_{k+1}^{\text{ag}}) - \frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{\text{ag},m})] \\
&= \mathbb{E}[F(w_k^{\text{ag}} + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})) - \frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{\text{ag},m})] \\
&= \mathbb{E}[\frac{1}{M} \sum_{m=1}^M F(w_k^{\text{ag}} + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})) - F(w_{k,\tau}^{\text{ag},m})] \\
&\leq \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \langle \nabla F(w_{k,\tau}^{\text{ag},m}), \frac{1}{M} \sum_{m=1}^M (Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})) \rangle \\
&\quad + \frac{L}{2} \|\frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})\|^2] \\
&= \frac{L}{2} \mathbb{E}[\|\frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})\|^2] \\
&= \frac{L}{2M^2} \sum_{m=1}^M \mathbb{E}[\|Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})\|^2] \leq \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2]
\end{aligned}$$

**Proposition 2.4.10.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{1}{\gamma\mu}, \beta = \alpha + 1, \gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}], \eta, \gamma \in (0, \frac{1}{L}]$ , we*

get

$$\mathbb{E}[A_{k,t}^m] \leq \mathbb{E}[A_{k,0}^m] + \left( \frac{(\gamma - \eta)^2(\mu + L)}{1 + \gamma\mu} + \frac{\gamma^2(\mu + L)^2 L}{\mu^2} \right) \cdot \left( 1 - \left( 1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu} \right)^t \right) \sigma^2$$

*Proof of proposition 2.4.10* From the notation mentioned in the beginning of section 2.4.3,

$$\begin{aligned} & \mathbb{E}[A_{k,t+1}^m | \mathcal{F}_{k,t}] \\ &= \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma\mu)^2} \mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] + \gamma^2 (\mu + L) \frac{2L}{1 + \gamma\mu} \mathbb{E}[\Psi_{k,t+1}^m | \mathcal{F}_{k,t}] \end{aligned} \quad (2.4)$$

Thus, let's sequentially compute  $\mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}]$  and  $\mathbb{E}[\Psi_{k,t+1}^m | \mathcal{F}_{k,t}]$ .

$$\begin{aligned} & \mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] \\ &= \mathbb{E}[\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma g_{k,t}^m - w_{k,t}^{\text{md},m} + \eta g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \\ &= \mathbb{E}[\|(1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}) - (\gamma - \eta)g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \quad (\leftarrow \gamma \geq \eta) \\ &= \|(1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}) - (\gamma - \eta)\nabla F(w_{k,t}^{\text{md},m})\|^2 \\ &\quad + (\gamma - \eta)^2 \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m}) - g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \\ &\leq (1 - \alpha^{-1})^2 \|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 + (\gamma - \eta)^2 \|\nabla F(w_{k,t}^{\text{md},m})\|^2 \\ &\quad + (\gamma - \eta)^2 \sigma^2 - 2(\gamma - \eta) \langle (1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}), \nabla F(w_{k,t}^{\text{md},m}) \rangle \\ &\leq (1 - \alpha^{-1})^2 (1 + \gamma\mu) \|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 \\ &\quad + (\gamma - \eta)^2 \left( 1 + \frac{1}{\gamma\mu} \right) \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + (\gamma - \eta)^2 \sigma^2 \\ &= \frac{(1 - \gamma\mu)^2}{1 + \gamma\mu} \|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + (\gamma - \eta)^2 \frac{1 + \gamma\mu}{\gamma\mu} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + (\gamma - \eta)^2 \sigma^2 \end{aligned}$$

Here, we need to bound  $\|\nabla F(w_{k,t}^{\text{md},m})\|^2$ .

$$\begin{aligned}
\|\nabla F(w_{k,t}^{\text{md},m})\|^2 &\leq 2L(F(w_{k,t}^{\text{md},m}) - F^*) \quad (\because \text{assumption 2.2.3}) \\
&\leq 2L\left(\beta^{-1}(F(w_{k,t}^m) - F(w^*)) + (1 - \beta^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*)\right) \\
&\leq \beta^{-1}L^2\|w_{k,t}^m - w^*\|^2 + 2(1 - \beta^{-1})L(F(w_{k,t}^{\text{ag},m}) - F^*) \\
&= \frac{\gamma\mu L^2}{1 + \gamma\mu}\|w_{k,t}^m - w^*\|^2 + \frac{2L}{1 + \gamma\mu}(F(w_{k,t}^{\text{ag},m}) - F^*) \\
&\leq \frac{\mu L}{1 + \gamma\mu}\|w_{k,t}^m - w^*\|^2 + \frac{2L}{1 + \gamma\mu}(F(w_{k,t}^{\text{ag},m}) - F^*) = \frac{2L}{1 + \gamma\mu}\Psi_{k,t}^m \quad (2.5)
\end{aligned}$$

The last inequality comes from the fact  $\gamma \in [0, \frac{1}{L})$ . Therefore, we finally get

$$\begin{aligned}
&\mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] \\
&\leq \frac{(1 - \gamma\mu)^2}{1 + \gamma\mu}\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + (\gamma - \eta)^2 \frac{1 + \gamma\mu}{\gamma\mu} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + (\gamma - \eta)^2 \sigma^2 \\
&\leq \frac{(1 - \gamma\mu)^2}{1 + \gamma\mu}\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + (\gamma - \eta)^2 \frac{1 + \gamma\mu}{\gamma\mu} \left( \frac{2L}{1 + \gamma\mu} \Psi_{k,t}^m \right) + (\gamma - \eta)^2 \sigma^2 \quad (2.6)
\end{aligned}$$

Now, let's compute  $\mathbb{E}[\Psi_{k,t+1}^m | \mathcal{F}_{k,t}]$ . We need to compute  $\mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}]$  and  $\mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}]$  first.

$$\begin{aligned}
\mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}] &= \mathbb{E}[\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma g_{k,t}^m - w^*\|^2 | \mathcal{F}_{k,t}] \\
&\leq \|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*\|^2 + \gamma^2 \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \gamma^2 \sigma^2 \\
&\quad - 2\gamma \langle (1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle \\
&\leq (1 - \alpha^{-1})\|w_{k,t}^m - w^*\|^2 + \alpha^{-1}\|w_{k,t}^{\text{md},m} - w^*\|^2 + \gamma^2 \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \gamma^2 \sigma^2 \\
&\quad - 2\gamma \langle (1 - \alpha^{-1}(1 - \beta^{-1}))w_{k,t}^m + \alpha^{-1}(1 - \beta^{-1})w_{k,t}^{\text{ag},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle \\
&= (1 - \gamma\mu)\|w_{k,t}^m - w^*\|^2 + \gamma\mu\|w_{k,t}^{\text{md},m} - w^*\|^2 + \gamma^2 \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \gamma^2 \sigma^2 \\
&\quad - 2\gamma \langle \frac{1}{1 + \gamma\mu}w_{k,t}^m + \frac{\gamma\mu}{1 + \gamma\mu}w_{k,t}^{\text{ag},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle
\end{aligned}$$

$$\begin{aligned}
& \mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}] \\
& \leq \mathbb{E}[F(w_{k,t}^{\text{md},m}) + \langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t+1}^{\text{ag},m} - w_{k,t}^{\text{md},m} \rangle + \frac{L}{2} \|w_{k,t+1}^{\text{ag},m} - w_{k,t}^{\text{md},m}\|^2 - F^* | \mathcal{F}_{k,t}] \\
& \leq F(w_{k,t}^{\text{md},m}) - F^* - \eta \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2 \\
& \leq F(w_{k,t}^{\text{md},m}) - F^* - \frac{\eta}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2 \left( \because 1 - \frac{\eta L}{2} \geq \frac{1}{2} \leftarrow \eta \in [0, \frac{1}{L}] \right) \\
& = (1 - \alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) + \alpha^{-1}(F(w_{k,t}^{\text{md},m}) - F^*) \\
& \quad + (1 - \alpha^{-1})(F(w_{k,t}^{\text{md},m}) - F(w_{k,t}^{\text{ag},m})) - \frac{\eta}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2
\end{aligned}$$

$$\begin{aligned}
& \leq (1 - \alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) - \frac{\mu \alpha^{-1}}{2} \|w_{k,t}^{\text{md},m} - w^*\|^2 + \alpha^{-1} \langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t}^{\text{md},m} - w^* \rangle \\
& \quad + (1 - \alpha^{-1}) \langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t}^{\text{md},m} - w_{k,t}^{\text{ag},m} \rangle - \frac{\eta}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2 \\
& = (1 - \alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) - \frac{\mu \alpha^{-1}}{2} \|w_{k,t}^{\text{md},m} - w^*\|^2 - \frac{\eta}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2 \\
& \quad + \alpha^{-1} \langle \nabla F(w_{k,t}^{\text{md},m}), \alpha \beta^{-1} w_{k,t}^m + (1 - \alpha \beta^{-1}) w_{k,t}^{\text{ag},m} - w^* \rangle \\
& = (1 - \gamma \mu)(F(w_{k,t}^{\text{ag},m}) - F^*) - \frac{\gamma \mu^2}{2} \|w_{k,t}^{\text{md},m} - w^*\|^2 - \frac{\eta}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2} \sigma^2 \\
& \quad + \gamma \mu \left\langle \frac{1}{1 + \gamma \mu} w_{k,t}^m + \frac{\gamma \mu}{1 + \gamma \mu} w_{k,t}^{\text{ag},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \right\rangle
\end{aligned}$$

Then, we bound  $\mathbb{E}[\Psi_{k,t+1}^m | \mathcal{F}_{k,t}]$  by using the above results.

$$\begin{aligned}
\mathbb{E}[\Psi_{k,t+1}^m | \mathcal{F}_{k,t}] & = \frac{\mu}{2} \mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}] + \mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}] \\
& \leq (1 - \gamma \mu) \Psi_{k,t}^m - \frac{\eta - \gamma^2 \mu}{2} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\gamma^2 \mu + \eta^2 L}{2} \sigma^2 \\
& \leq (1 - \gamma \mu) \Psi_{k,t}^m + \frac{\gamma^2 \mu + \eta^2 L}{2} \sigma^2 \left( \because \gamma \leq \sqrt{\frac{\eta}{\mu}} \right) \\
& \leq (1 - \gamma \mu) \Psi_{k,t}^m + \frac{\gamma^2 (\mu + L)}{2} \sigma^2 \tag{2.7}
\end{aligned}$$

Plugging Equation 2.6, Equation 2.7 in Equation 2.4 yields,

$$\begin{aligned}
\mathbb{E}[A_{k,t+1}^m | \mathcal{F}_{k,t}] &\leq \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma \mu)^2} \left( \frac{(1 - \gamma \mu)^2}{1 + \gamma \mu} \|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + (\gamma - \eta)^2 \frac{1 + \gamma \mu}{\gamma \mu} \right. \\
&\quad \left. \left( \frac{2L}{1 + \gamma \mu} \Psi_{k,t}^m \right) + (\gamma - \eta)^2 \sigma^2 \right) + \gamma^2 (\mu + L) \frac{2L}{1 + \gamma \mu} \left( (1 - \gamma \mu) \Psi_{k,t}^m + \frac{\gamma^2 (\mu + L)}{2} \sigma^2 \right) \\
&= \frac{(1 - \gamma \mu)^2}{1 + \gamma \mu} \cdot \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma \mu)^2} \|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + \left( \frac{\gamma \mu (\gamma - \eta)^2 (\mu + L)}{1 + \gamma \mu} \right. \\
&\quad \left. + \gamma^2 (\mu + L) (1 - \gamma \mu) \right) \frac{2L}{1 + \gamma \mu} \Psi_{k,t}^m + \left( \frac{\gamma^2 \mu^2 (\gamma - \eta)^2 (\mu + L)}{(1 + \gamma \mu)^2} + \frac{\gamma^4 (\mu + L)^2 L}{1 + \gamma \mu} \right) \sigma^2
\end{aligned} \tag{2.8}$$

Since  $\eta \leq \gamma$ , we get  $(\gamma - \eta)^2 \leq \gamma^2$ . By using this fact, we obtain

$$\begin{aligned}
\frac{\gamma \mu (\gamma - \eta)^2 (\mu + L)}{1 + \gamma \mu} + \gamma^2 (\mu + L) (1 - \gamma \mu) &\leq \frac{\gamma^3 \mu (\mu + L)}{1 + \gamma \mu} + \gamma^2 (\mu + L) (1 - \gamma \mu) \\
&= \gamma^2 (\mu + L) \left( 1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} \right)
\end{aligned} \tag{2.9}$$

It is easy to show that  $1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} < 1$ . Also, we get

$$\frac{(1 - \gamma \mu)^2}{1 + \gamma \mu} < 1 - \gamma \mu < 1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} \tag{2.10}$$

From Equation 2.8, Equation 2.9, and Equation 2.10 we finally get

$$\mathbb{E}[A_{k,t+1}^m | \mathcal{F}_{k,t}] \leq \left( 1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} \right) A_{k,t}^m + \left( \frac{\gamma^2 \mu^2 (\gamma - \eta)^2 (\mu + L)}{(1 + \gamma \mu)^2} + \frac{\gamma^4 (\mu + L)^2 L}{1 + \gamma \mu} \right) \sigma^2$$

From this relationship between  $A_{k,t+1}^m$  and  $A_{k,t}^m$ , we obtain the result of proposition 2.4.10.

$$\begin{aligned}
\mathbb{E}[A_{k,t}^m] &\leq \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)^t \mathbb{E}[A_{k,0}^m] + \left(\frac{\gamma^2\mu^2(\gamma - \eta)^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2L}{1 + \gamma\mu}\right)\sigma^2 \\
&\quad \cdot \frac{1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)^t}{1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)} \\
&= \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)^t \mathbb{E}[A_{k,0}^m] + \left(\frac{(\gamma - \eta)^2(\mu + L)}{1 + \gamma\mu} + \frac{\gamma^2(\mu + L)^2L}{\mu^2}\right)\sigma^2 \\
&\quad \cdot \left(1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)^t\right) \\
&\leq \mathbb{E}[A_{k,0}^m] + \left(\frac{(\gamma - \eta)^2(\mu + L)}{1 + \gamma\mu} + \frac{\gamma^2(\mu + L)^2L}{\mu^2}\right) \cdot \left(1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1 + \gamma\mu}\right)^t\right)\sigma^2
\end{aligned}$$

**Proposition 2.4.11.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{1}{\gamma\mu}, \beta = \alpha + 1, \gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}], \eta, \gamma \in (0, \frac{1}{L}], \tau \geq 2$ , FedAQ yields*

$$\begin{aligned}
\frac{\mu}{2}\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] &+ \frac{L}{2}\mathbb{E}[\|w_{k,\tau}^{ag,m} - w_k^{ag}\|^2] \\
&\leq \left(\frac{4\gamma^2\mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu}\right)\tau^2\mathbb{E}[\Psi_k] + (\gamma^2\mu + \eta^2L)\tau\sigma^2 \\
&\quad + \left(\frac{(\gamma - \eta)^2\gamma^2\mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2L}{1 + \gamma\mu}\right)\frac{\tau^3\sigma^2}{2}
\end{aligned}$$

*Proof of proposition 2.4.11* Let's first bound  $\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2]$  and  $\mathbb{E}[\|w_{k,\tau}^{ag,m} - w_k^{ag}\|^2]$  individually.

$$\begin{aligned}
\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] &= \mathbb{E}[\|(w_{k,\tau}^m - w_{k,\tau-1}^m) + \dots + (w_{k,1}^m - w_{k,0}^m)\|^2] \\
&= \mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1} \left((1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{md,m} - w_{k,t}^m - \gamma g_{k,t}^m\right)\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\alpha^{-1}\sum_{t=0}^{\tau-1} (w_{k,t}^{md,m} - w_{k,t}^m) - \gamma\sum_{t=0}^{\tau-1} g_{k,t}^m\right\|^2\right]
\end{aligned}$$

Then, this is upper bounded by

$$\begin{aligned}
&\leq 2\alpha^{-2}\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(w_{k,t}^{\text{md},m} - w_{k,t}^m)\right\|^2\right] + 2\gamma^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}g_{k,t}^m\right\|^2\right] \\
&\leq 2\alpha^{-2}\tau\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|w_{k,t}^{\text{md},m} - w_{k,t}^m\|^2\right] + 2\gamma^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}\nabla F(w_{k,t}^{\text{md},m})\right\|^2\right] \\
&\quad + 2\gamma^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(g_{k,t}^m - \nabla F(w_{k,t}^{\text{md},m}))\right\|^2\right] \\
&\leq 2\alpha^{-2}(1 - \beta^{-1})^2\tau\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2\right] + 2\gamma^2\tau\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|\nabla F(w_{k,t}^{\text{md},m})\|^2\right] \\
&\quad + 2\gamma^2\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|g_{k,t}^m - \nabla F(w_{k,t}^{\text{md},m})\|^2\right] \\
&= \tau\left(\sum_{t=0}^{\tau-1}2\alpha^{-2}(1 - \beta^{-1})^2\mathbb{E}\left[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2\right] + 2\gamma^2\mathbb{E}\left[\|\nabla F(w_{k,t}^{\text{md},m})\|^2\right]\right) + 2\tau\gamma^2\sigma^2
\end{aligned}$$

For  $\mathbb{E}\left[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2\right]$ , we similarly obtain

$$\begin{aligned}
\mathbb{E}\left[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2\right] &= \mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(w_{k,t+1}^{\text{ag},m} - w_{k,t}^{\text{ag},m})\right\|^2\right] = \mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(w_{k,t}^{\text{md},m} - w_{k,t}^{\text{ag},m} - \eta g_{k,t}^m)\right\|^2\right] \\
&\leq 2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(w_{k,t}^{\text{md},m} - w_{k,t}^{\text{ag},m})\right\|^2\right] + 2\eta^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}g_{k,t}^m\right\|^2\right] \\
&= 2\beta^{-2}\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(w_{k,t}^m - w_{k,t}^{\text{ag},m})\right\|^2\right] + 2\eta^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}\nabla F(w_{k,t}^{\text{md},m})\right\|^2\right] \\
&\quad + 2\eta^2\mathbb{E}\left[\left\|\sum_{t=0}^{\tau-1}(g_{k,t}^m - \nabla F(w_{k,t}^{\text{md},m}))\right\|^2\right] \\
&\leq 2\beta^{-2}\tau\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2\right] + 2\eta^2\tau\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|\nabla F(w_{k,t}^{\text{md},m})\|^2\right] \\
&\quad + 2\eta^2\sum_{t=0}^{\tau-1}\mathbb{E}\left[\|g_{k,t}^m - \nabla F(w_{k,t}^{\text{md},m})\|^2\right] \\
&= \tau\left(\sum_{t=0}^{\tau-1}2\beta^{-2}\mathbb{E}\left[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2\right] + 2\eta^2\mathbb{E}\left[\|\nabla F(w_{k,t}^{\text{md},m})\|^2\right]\right) + 2\tau\eta^2\sigma^2
\end{aligned}$$

Thus, by using the above results, we get

$$\begin{aligned}
& \frac{\mu}{2} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
& \leq \tau \sum_{t=0}^{\tau-1} \left\{ \left( \mu \alpha^{-2} (1 - \beta^{-1})^2 + L \beta^{-2} \right) \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] \right. \\
& \quad \left. + (\gamma^2 \mu + \eta^2 L) \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m})\|^2] \right\} + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2 \\
& \leq \tau \sum_{t=0}^{\tau-1} \left\{ \left( \mu \alpha^{-2} (1 - \beta^{-1})^2 + L \beta^{-2} \right) \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] + (\gamma^2 \mu + \eta^2 L) \frac{2L}{1 + \gamma \mu} \mathbb{E}[\Psi_{k,t}^m] \right\} \\
& \quad + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2 \quad (\because \text{Equation 2.5}) \\
& \leq \tau \sum_{t=0}^{\tau-1} \left\{ \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma \mu)^2} \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] + \gamma^2 (\mu + L) \frac{2L}{1 + \gamma \mu} \mathbb{E}[\Psi_{k,t}^m] \right\} \\
& \quad + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2 = \tau \left( \sum_{t=0}^{\tau-1} \mathbb{E}[A_{k,t}^m] \right) + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2
\end{aligned}$$

By proposition 2.4.10 and the fact  $\Psi_{k,0}^m = \Psi_k$ , we obtain

$$\begin{aligned}
& \frac{\mu}{2} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \leq \tau \left\{ \sum_{t=0}^{\tau-1} \mathbb{E}[A_{k,0}^m] + \left( \frac{(\gamma - \eta)^2 (\mu + L)}{1 + \gamma \mu} \right. \right. \\
& \quad \left. \left. + \frac{\gamma^2 (\mu + L)^2 L}{\mu^2} \right) \cdot \left( 1 - \left( 1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} \right)^t \right) \sigma^2 \right\} + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2 \\
& = \tau^2 \left( \frac{\gamma^2 \mu^2 (\mu + L)}{(1 + \gamma \mu)^2} \mathbb{E}[\|w_k - w_k^{\text{ag}}\|^2] + \gamma^2 (\mu + L) \frac{2L}{1 + \gamma \mu} \mathbb{E}[\Psi_k] \right) + \tau \left( \frac{(\gamma - \eta)^2 (\mu + L)}{1 + \gamma \mu} \right. \\
& \quad \left. + \frac{\gamma^2 (\mu + L)^2 L}{\mu^2} \right) \left( \sum_{t=0}^{\tau-1} 1 - \left( 1 - \gamma \mu + \frac{\gamma \mu}{1 + \gamma \mu} \right)^t \right) \sigma^2 + (\gamma^2 \mu + \eta^2 L) \tau \sigma^2
\end{aligned}$$

Before we get to the final result, let's find the upper bound for  $\|w_k - w_k^{\text{ag}}\|^2$ ,  $\sum_{t=0}^{\tau-1} \left( 1 - \right.$

$$\left(1 - \gamma\mu + \frac{\gamma\mu}{1+\gamma\mu}\right)^t$$

$$\begin{aligned} \|w_k - w_k^{\text{ag}}\|^2 &= \|w_k - w^* - (w_k^{\text{ag}} - w^*)\|^2 \leq 2\|w_k - w^*\|^2 + 2\|w_k^{\text{ag}} - w^*\|^2 \\ &\leq 2\|w_k - w^*\|^2 + 2 \cdot \frac{2}{\mu} \left( F(w_k^{\text{ag}}) - F^* - \langle \nabla F(w^*), w_k^{\text{ag}} - w^* \rangle \right) \\ &= 2\|w_k - w^*\|^2 + \frac{4}{\mu} (F(w_k^{\text{ag}}) - F^*) = \frac{4}{\mu} \Psi_k \end{aligned}$$

$$\begin{aligned} \sum_{t=0}^{\tau-1} \left(1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1+\gamma\mu}\right)^t\right) &= \tau - \sum_{t=0}^{\tau-1} \left(1 - \gamma\mu + \frac{\gamma\mu}{1+\gamma\mu}\right)^t \\ &= \tau - \frac{1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1+\gamma\mu}\right)^\tau}{1 - \left(1 - \gamma\mu + \frac{\gamma\mu}{1+\gamma\mu}\right)} \\ &\leq \tau - \frac{1 - \left(1 - \frac{\gamma^2\mu^2}{1+\gamma\mu}\tau + \left(\frac{\gamma^2\mu^2}{1+\gamma\mu}\right)^2 \frac{\tau(\tau-1)}{2}\right)}{\frac{\gamma^2\mu^2}{1+\gamma\mu}} \\ &= \frac{\gamma^2\mu^2}{1+\gamma\mu} \cdot \frac{\tau(\tau-1)}{2} \leq \frac{\gamma^2\mu^2}{1+\gamma\mu} \cdot \frac{\tau^2}{2} \end{aligned}$$

Therefore, we conclude as below

$$\begin{aligned} &\frac{\mu}{2} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\ &\leq \left( \frac{4\gamma^2\mu(\mu+L)}{(1+\gamma\mu)^2} + \frac{2L\gamma^2(\mu+L)}{1+\gamma\mu} \right) \tau^2 \mathbb{E}[\Psi_k] + (\gamma^2\mu + \eta^2 L) \tau \sigma^2 \\ &\quad + \left( \frac{(\gamma-\eta)^2 \gamma^2 \mu^2 (\mu+L)}{(1+\gamma\mu)^2} + \frac{\gamma^4 (\mu+L)^2 L}{1+\gamma\mu} \right) \frac{\tau^3 \sigma^2}{2} \end{aligned}$$

*Proof of lemma 2.4.6* By the definition of  $\Psi_k$ ,  $\Psi_{k,t}$  and proposition 2.4.7,

$$\mathbb{E}[\Psi_{k+1}] = \mathbb{E}[\Psi_{k,\tau}] + \frac{\mu}{2} \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] + \mathbb{E}[F(w_{k+1}^{\text{ag}}) - \frac{1}{M} \sum_{m=1}^M F(w_{k,\tau}^{\text{ag},m})]$$

Applying proposition 2.4.8 and proposition 2.4.9, we have

$$\begin{aligned}
\mathbb{E}[\Psi_{k+1}] &\leq (1 - \gamma\mu)^\tau \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 + \gamma\mu L\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} \\
&\quad - w_{k,t}^{\text{md},m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m})\|] \\
&\quad + \frac{q\mu}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
&\leq (1 - \gamma\mu)^\tau \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 + \gamma\mu L\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} - w_{k,t}^{\text{md},m}\| \\
&\quad \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m})\|] + \frac{q}{M} \left[ \left( \frac{4\gamma^2 \mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu} \right) \right. \\
&\quad \left. \tau^2 \mathbb{E}[\Psi_k] + (\gamma^2 \mu + \eta^2 L)\tau\sigma^2 + \left( \frac{(\gamma - \eta)^2 \gamma^2 \mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2 L}{1 + \gamma\mu} \right) \frac{\tau^3 \sigma^2}{2} \right] \\
&= \left\{ (1 - \gamma\mu)^\tau + \frac{q}{M} \left( \frac{4\gamma^2 \mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu} \right) \tau^2 \right\} \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 \\
&\quad + \frac{q}{M}(\gamma^2 \mu + \eta^2 L)\tau\sigma^2 + \frac{q}{2M} \left( \frac{(\gamma - \eta)^2 \gamma^2 \mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2 L}{1 + \gamma\mu} \right) \tau^3 \sigma^2 + \gamma\mu L\tau \\
&\quad \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} - w_{k,t}^{\text{md},m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m})\|]
\end{aligned}$$

The second inequality comes from proposition 2.4.11. Then, let's define  $C(\gamma, \tau)$  as

$$C(\gamma, \tau) = (1 - \gamma\mu)^\tau + \frac{q}{M} \left( \frac{4\gamma^2 \mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu} \right) \tau^2$$

Finally, we obtain

$$\begin{aligned}
\mathbb{E}[\Psi_{k+1}] &\leq C(\gamma, \tau) \mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2 L + \frac{\gamma^2 \mu}{M})\tau\sigma^2 + \frac{q}{M}(\gamma^2 \mu + \eta^2 L)\tau\sigma^2 \\
&\quad + \frac{q}{2M} \left( \frac{(\gamma - \eta)^2 \gamma^2 \mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2 L}{1 + \gamma\mu} \right) \tau^3 \sigma^2 + \gamma\mu L\tau \\
&\quad \cdot \max_{0 \leq t < \tau} \mathbb{E}[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} - w_{k,t}^{\text{md},m}\| \|\frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m})\|]
\end{aligned}$$

*Proof of theorem 2.4.12*

**Theorem 2.4.12.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.1, assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{1}{\gamma\mu}$ ,  $\beta = \alpha + 1$ ,  $\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}})$ ,  $\eta, \gamma \in (0, \frac{1}{L}]$ ,  $\tau \geq 2$ , if the learning rate  $\gamma$  satisfies*

$$\left(\mu^2 + \frac{q}{M}(\mu + L)(4\mu + 2L)\right)\gamma\tau \leq \frac{1}{2}\mu \quad (2.11)$$

*FedAQ yields*

$$\begin{aligned} \mathbb{E}[\Psi_K] &\leq \exp\left(-\frac{1}{2}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Psi_0 + (2q + 1)\left(\frac{\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{\eta\sigma^2}{M}\right) + 14\eta^2L\tau\sigma^2 \\ &\quad + \frac{(780 + \frac{2q}{M})\eta^{\frac{3}{2}}L\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}} + \frac{(\mu + L)(\mu^2 + \mu L + L^2)q\eta^{\frac{3}{2}}\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{5}{2}}M} + \frac{q\eta^3\tau^2(\mu + L)^2L\sigma^2}{\mu M} \end{aligned}$$

*Proof of theorem 2.4.12* At first, due to the condition (eq. (2.11)) in theorem 2.4.12, we get

$$\begin{aligned} C(\gamma, \tau) &= (1 - \gamma\mu)^\tau + \frac{q}{M}\left(\frac{4\gamma^2\mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{2L\gamma^2(\mu + L)}{1 + \gamma\mu}\right)\tau^2 \\ &\leq 1 - \gamma\mu\tau + \gamma^2\mu^2\tau^2 + \frac{q}{M}\gamma^2(\mu + L)(4\mu + 2L)\tau^2 \\ &= 1 - \gamma\mu\tau + \left(\mu^2 + \frac{q}{M}(\mu + L)(4\mu + 2L)\right)\gamma^2\tau^2 \\ &\leq 1 - \frac{1}{2}\gamma\mu\tau \quad (\because \text{condition (2.11)}) \end{aligned}$$

The first inequality comes from the fact that  $(1 - \gamma\mu)^\tau \leq e^{-\gamma\mu\tau} \leq 1 - \gamma\mu\tau + \gamma^2\mu^2\tau^2$  when  $0 \leq \gamma\mu \leq 1$ . Also, it is trivial that  $\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}}) \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ . Thus, we can use

lemma 2.4.6. By using lemma 2.4.6 and the above result, we obtain

$$\begin{aligned}
\mathbb{E}[\Psi_{k+1}] &\leq (1 - \frac{1}{2}\gamma\mu\tau)\mathbb{E}[\Psi_k] + \frac{1}{2}(\eta^2L + \frac{\gamma^2\mu}{M})\tau\sigma^2 + \frac{q}{M}(\gamma^2\mu + \eta^2L)\tau\sigma^2 \\
&\quad + \frac{q}{2M}\left(\frac{(\gamma - \eta)^2\gamma^2\mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2L}{1 + \gamma\mu}\right)\tau^3\sigma^2 + \gamma\mu L\tau \\
&\quad \cdot \max_{0 \leq t < \tau} \mathbb{E}\left[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} - w_{k,t}^{\text{md},m}\| \left\| \frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m}) \right\| \right]
\end{aligned} \tag{2.12}$$

By the Lemma B.3 in [31], we know that the below quantity is bounded.

$$\begin{aligned}
\max_{0 \leq t < \tau} \mathbb{E}\left[\frac{1}{M} \sum_{m=1}^M \|\bar{w}_{k,t}^{\text{md}} - w_{k,t}^{\text{md},m}\| \left\| \frac{1}{1 + \gamma\mu}(\bar{w}_{k,t} - w_{k,t}^m) + \frac{\gamma\mu}{1 + \gamma\mu}(\bar{w}_{k,t}^{\text{ag}} - w_{k,t}^{\text{ag},m}) \right\| \right] &\leq B \\
B &= \begin{cases} 7\eta\gamma\tau\sigma^2 \left(1 + \frac{2\gamma^2\mu}{\eta}\right)^{2\tau}, & \text{if } \gamma \in \left(\eta, \sqrt{\frac{\eta}{\mu}}\right] \\ 7\eta^2\tau\sigma^2, & \text{if } \gamma = \eta \end{cases}
\end{aligned}$$

Telescoping Equation 2.12 yields

$$\begin{aligned}
\mathbb{E}[\Psi_K] &\leq (1 - \frac{1}{2}\gamma\mu\tau)^K \Psi_0 + \left(\sum_{k'=0}^{K-1} (1 - \frac{1}{2}\gamma\mu\tau)^{k'}\right) \cdot \left[\frac{1}{2}(\eta^2L + \frac{\gamma^2\mu}{M})\tau\sigma^2 + \gamma\mu L\tau B \right. \\
&\quad \left. + \frac{q}{M}(\gamma^2\mu + \eta^2L)\tau\sigma^2 + \frac{q}{2M}\left(\frac{(\gamma - \eta)^2\gamma^2\mu^2(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^4(\mu + L)^2L}{1 + \gamma\mu}\right)\tau^3\sigma^2\right] \\
&\leq \exp\left(-\frac{\gamma\mu\tau K}{2}\right) \Psi_0 + \frac{\eta^2L\sigma^2}{\gamma\mu} + \frac{\gamma\sigma^2}{M} + 2LB + 2q\left(\frac{\gamma\sigma^2}{M} + \frac{\eta^2L\sigma^2}{\gamma\mu M}\right) \\
&\quad + \frac{q}{M}\left(\frac{(\gamma - \eta)^2\gamma\mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^3(\mu + L)^2L}{(1 + \gamma\mu)\mu}\right)\tau^2\sigma^2
\end{aligned}$$

The last inequality comes from the fact that  $\sum_{k'=0}^{K-1} (1 - \frac{1}{2}\gamma\mu\tau)^{k'} \leq \frac{2}{\gamma\mu\tau}$ . Since we plug in

$\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}})$ , we can use Lemma B.4 in [31]. Therefore, we obtain

$$\begin{aligned} \mathbb{E}[\Psi_K] &\leq \exp\left(-\frac{1}{2}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Psi_0 + \frac{\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{\eta\sigma^2}{M} + \frac{780\eta^{\frac{3}{2}}L\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}} \\ &\quad + 14\eta^2L\tau\sigma^2 + \max\left(\frac{2q\eta^{\frac{1}{2}}\sigma^2}{M\mu^{\frac{1}{2}}\tau^{\frac{1}{2}}}, \frac{2q\eta\sigma^2}{M}\right) + \min\left(\frac{2q\eta^{\frac{3}{2}}\tau^{\frac{1}{2}}L\sigma^2}{M\mu^{\frac{1}{2}}}, \frac{2q\eta L\sigma^2}{M\mu}\right) \\ &\quad + \frac{q\tau^2\sigma^2}{M}\max\left(\frac{\eta^{\frac{3}{2}}\mu(\mu+L)}{\mu^{\frac{3}{2}}\tau^{\frac{3}{2}}}, \frac{\eta^{\frac{3}{2}}(\mu+L)^2L}{\mu^{\frac{5}{2}}\tau^{\frac{3}{2}}}, \frac{\eta^3(\mu+L)^2L}{\mu}\right) \end{aligned}$$

The first term stems directly from Lemma B.4 in [31]. Also, the last term comes from the fact that

$$\frac{(\gamma - \eta)^2\gamma\mu(\mu + L)}{(1 + \gamma\mu)^2} + \frac{\gamma^3(\mu + L)^2L}{(1 + \gamma\mu)\mu} \leq \begin{cases} \gamma^3\mu(\mu + L) + \frac{\gamma^3(\mu+L)^2L}{\mu}, & \text{if } \gamma \neq \eta \\ \frac{\eta^3(\mu+L)^2L}{\mu}, & \text{if } \gamma = \eta \end{cases}$$

Therefore, by simple inequalities such as  $\max(a, b) \leq a + b$  and  $\min(a, b) \leq a$ , we ultimately get

$$\begin{aligned} \mathbb{E}[\Psi_K] &\leq \exp\left(-\frac{1}{2}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Psi_0 + \frac{(2q+1)\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{(2q+1)\eta\sigma^2}{M} + 14\eta^2L\tau\sigma^2 \\ &\quad + \frac{(780 + \frac{2q}{M})\eta^{\frac{3}{2}}L\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}} + \frac{(\mu+L)(\mu^2 + \mu L + L^2)q\eta^{\frac{3}{2}}\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{5}{2}}M} + \frac{q\eta^3\tau^2(\mu+L)^2L\sigma^2}{\mu M} \end{aligned} \tag{2.13}$$

*Proof of corollary 2.4.13*

**Corollary 2.4.13.** *Let  $C_1, C_2$ , and  $\eta_0$  as below. Note that  $T = K\tau$ .*

$$\begin{aligned} C_1 &= \frac{(\mu+L)(\mu^2 + \mu L + L^2)q}{\mu^{\frac{5}{2}}}, \quad C_2 = \frac{q(\mu+L)^2L}{\mu} \\ \eta_0 &= \frac{4\tau}{\mu T^2} \log^2\left(e + \min\left(\frac{\mu MT\Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3\Psi_0}{L\tau^2\sigma^2}, \frac{\mu^3 MT^3\Psi_0}{(\mu^{\frac{3}{2}}C_1 + 8C_2)\tau^2\sigma^2}\right)\right) \end{aligned}$$

Then for  $\eta = \min(\frac{1}{L}, \eta_0)$ , FedAQ yields

$$\begin{aligned} \mathbb{E}[\Psi_K] &\leq \min\left(\exp(-\frac{\mu T}{2L}), \exp(-\frac{\mu^{\frac{1}{2}} T}{2L^{\frac{1}{2}} \tau^{\frac{1}{2}}})\right) \Psi_0 \\ &+ \frac{7(2q+1)\sigma^2}{\mu MT} \log^2\left(e + \frac{\mu MT \Psi_0}{(2q+1)\sigma^2}\right) \end{aligned} \quad (2.14)$$

$$+ \frac{(6465 + \frac{16q}{M})L\tau^2\sigma^2}{\mu^2 T^3} \log^4\left(e + \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right) \quad (2.15)$$

$$+ \frac{9(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}{\mu^3 MT^3} \log^6\left(e + \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right) \quad (2.16)$$

*Proof of corollary 2.4.13* Let's decompose the final result Equation 2.13 of the theorem 2.4.12 into a decreasing term and an increasing term. We denote the decreasing term  $\psi_1$  and the increasing term  $\psi_2$  as below.

$$\begin{aligned} \psi_1(\eta) &= \exp\left(-\frac{1}{2} \max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})T\right) \Psi_0 \\ \psi_2(\eta) &= \frac{(2q+1)\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}} M \tau^{\frac{1}{2}}} + \frac{(2q+1)\eta\sigma^2}{M} + \frac{(780 + \frac{2q}{M})\eta^{\frac{3}{2}} L \tau^{\frac{1}{2}} \sigma^2}{\mu^{\frac{1}{2}}} + 14\eta^2 L \tau \sigma^2 \\ &+ \frac{(\mu+L)(\mu^2 + \mu L + L^2)q\eta^{\frac{3}{2}} \tau^{\frac{1}{2}} \sigma^2}{\mu^{\frac{5}{2}} M} + \frac{q\eta^3 \tau^2 (\mu+L)^2 L \sigma^2}{\mu M} \end{aligned}$$

Since  $\psi_1$  is the decreasing term, we have

$$\psi_1(\eta) \leq \psi_1\left(\frac{1}{L}\right) + \psi_1(\eta_0) \quad (2.17)$$

where

$$\begin{aligned} \psi_1\left(\frac{1}{L}\right) &= \min\left(\exp(-\frac{\mu T}{2L}), \exp(-\frac{\mu^{\frac{1}{2}} T}{2L^{\frac{1}{2}} \tau^{\frac{1}{2}}})\right) \Psi_0 \\ \psi_1(\eta_0) &\leq \exp\left(-\frac{1}{2} \sqrt{\frac{\eta_0 \mu}{\tau}} T\right) = \left(e + \min\left(\frac{\mu MT \Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right.\right. \\ &\left.\left., \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right)\right)^{-1} \Psi_0 \leq \frac{(2q+1)\sigma^2}{\mu MT} + \frac{L\tau^2\sigma^2}{\mu^2 T^3} + \frac{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}{\mu^3 MT^3} \end{aligned}$$

Since  $\psi_2$  is the increasing term, we have

$$\begin{aligned}
\psi_2(\eta) &\leq \psi_2(\eta_0) \\
&\leq \frac{2(2q+1)\sigma^2}{\mu MT} \log\left(e + \frac{\mu MT \Psi_0}{(2q+1)\sigma^2}\right) + \frac{4(2q+1)\tau\sigma^2}{\mu MT^2} \log^2\left(e + \frac{\mu MT \Psi_0}{(2q+1)\sigma^2}\right) \\
&\quad + \frac{8(780 + \frac{2q}{M})L\tau^2\sigma^2}{\mu^2 T^3} \log^3\left(e + \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right) + \frac{224L\tau^3\sigma^2}{\mu^2 T^4} \log^4\left(e + \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right) \\
&\quad + \frac{8C_1\tau^2\sigma^2}{\mu^{\frac{3}{2}} MT^3} \log^3\left(e + \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right) + \frac{64C_2\tau^5\sigma^2}{\mu^3 MT^6} \log^6\left(e + \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right) \\
&\leq \frac{6(2q+1)\sigma^2}{\mu MT} \log^2\left(e + \frac{\mu MT \Psi_0}{(2q+1)\sigma^2}\right) + \frac{(6464 + \frac{16q}{M})L\tau^2\sigma^2}{\mu^2 T^3} \log^4\left(e + \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right) \\
&\quad + \frac{8(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}{\mu^3 MT^3} \log^6\left(e + \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right) \tag{2.18}
\end{aligned}$$

The last inequality comes from  $\frac{\tau}{T} \leq 1$ . Therefore, by combining Equation 2.17 and Equation 2.18, we finally get

$$\begin{aligned}
\mathbb{E}[\Psi_K] &\leq \psi_1(\eta) + \psi_2(\eta) \leq \psi_1\left(\frac{1}{L}\right) + \psi_1(\eta_0) + \psi_2(\eta_0) \\
&\leq \min\left(\exp\left(-\frac{\mu T}{2L}\right), \exp\left(-\frac{\mu^{\frac{1}{2}} T}{2L^{\frac{1}{2}}\tau^{\frac{1}{2}}}\right)\right) \Psi_0 + \frac{7(2q+1)\sigma^2}{\mu MT} \log^2\left(e + \frac{\mu MT \Psi_0}{(2q+1)\sigma^2}\right) \\
&\quad + \frac{(6465 + \frac{16q}{M})L\tau^2\sigma^2}{\mu^2 T^3} \log^4\left(e + \frac{\mu^2 T^3 \Psi_0}{L\tau^2\sigma^2}\right) \\
&\quad + \frac{9(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}{\mu^3 MT^3} \log^6\left(e + \frac{\mu^3 MT^3 \Psi_0}{(\mu^{\frac{3}{2}} C_1 + 8C_2)\tau^2\sigma^2}\right)
\end{aligned}$$

*Why the Condition (eq. (2.11)) is Satisfied*

The synchronization rounds  $K$  required for linear speedup in  $M$  for FedAQ is  $\tilde{O}\left(\left(\frac{M}{1+q}\right)^{\frac{1}{2}}\right)$  (See remark 2.4.4). Since we derive this result from theorem 2.4.12, we should show that  $K = \tilde{O}\left(\left(\frac{M}{1+q}\right)^{\frac{1}{2}}\right)$  satisfies the condition (eq. (2.11)) in theorem 2.4.12.

$$\left(\mu^2 + \frac{q}{M}(\mu + L)(4\mu + 2L)\right)\gamma\tau \leq \frac{1}{2}\mu$$

We rewrite the above condition as below.

$$\gamma\tau \leq \frac{\mu}{2\mu^2 + \frac{2q}{M}(\mu + L)(4\mu + 2L)} \quad (2.19)$$

We know  $\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}})$  and  $\eta = \min(\frac{1}{L}, \eta_0)$ . Since  $\eta_0$  becomes smaller and smaller as  $T$  increases, we assume  $\eta = \eta_0$  here. Therefore, we get

$$\begin{aligned} \gamma\tau &= \max(\eta_0\tau, \sqrt{\frac{\eta_0\tau}{\mu}}) \\ &= \max\left(\frac{4\tau^2}{\mu T^2} \log^2\left(e + \min\left(\frac{\mu MT\Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3\Psi_0}{L\tau^2\sigma^2}, \frac{\mu^3 MT^3\Psi_0}{(\mu^{\frac{3}{2}}C_1 + 8C_2)\tau^2\sigma^2}\right)\right), \right. \\ &\quad \left. \frac{2\tau}{\mu T} \log\left(e + \min\left(\frac{\mu MT\Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3\Psi_0}{L\tau^2\sigma^2}, \frac{\mu^3 MT^3\Psi_0}{(\mu^{\frac{3}{2}}C_1 + 8C_2)\tau^2\sigma^2}\right)\right)\right) \end{aligned}$$

Note that  $K = \frac{T}{\tau} = \tilde{O}((\frac{M}{1+q})^{\frac{1}{2}}) = C(\frac{M}{1+q})^{\frac{1}{2}} \log(T)$  because  $\tilde{O}$  contains hidden multiplicative polylog factors with respect to  $T$ . We can assume  $T$  is sufficiently large here. Then, we have

$$\begin{aligned} \gamma\tau &= \max\left(\frac{4(1+q)}{\mu C^2 M \log^2(T)} \log^2\left(e + \min\left(\frac{\mu MT\Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3\Psi_0}{L\tau^2\sigma^2}, \frac{\mu^3 MT^3\Psi_0}{(\mu^{\frac{3}{2}}C_1 + 8C_2)\tau^2\sigma^2}\right)\right), \right. \\ &\quad \left. \frac{2(1+q)^{\frac{1}{2}}}{\mu C M^{\frac{1}{2}} \log(T)} \log\left(e + \min\left(\frac{\mu MT\Psi_0}{(2q+1)\sigma^2}, \frac{\mu^2 T^3\Psi_0}{L\tau^2\sigma^2}, \frac{\mu^3 MT^3\Psi_0}{(\mu^{\frac{3}{2}}C_1 + 8C_2)\tau^2\sigma^2}\right)\right)\right) \\ &\leq \max\left(\frac{4(1+q)}{\mu C^2 M \log^2(T)} \log^2\left(\frac{2\mu MT\Psi_0}{(2q+1)\sigma^2}\right), \frac{2(1+q)^{\frac{1}{2}}}{\mu C M^{\frac{1}{2}} \log(T)} \log\left(\frac{2\mu MT\Psi_0}{(2q+1)\sigma^2}\right)\right) \end{aligned}$$

For an arbitrary constant  $k_1 > 0$ , it is easy to show that  $\lim_{T \rightarrow \infty} \frac{\log(k_1 T)}{\log(T)} = 1$ . Thus, we obtain

$$\begin{aligned} \gamma\tau &\leq \max\left(\frac{4(1+q)}{\mu C^2 M \log^2(T)} \log^2\left(\frac{2\mu MT\Psi_0}{(2q+1)\sigma^2}\right), \frac{2(1+q)^{\frac{1}{2}}}{\mu C M^{\frac{1}{2}} \log(T)} \log\left(\frac{2\mu MT\Psi_0}{(2q+1)\sigma^2}\right)\right) \\ &\simeq \max\left(\frac{4(1+q)}{\mu C^2 M}, \frac{2(1+q)^{\frac{1}{2}}}{\mu C M^{\frac{1}{2}}}\right) \leq \frac{\mu}{2\mu^2 + \frac{2q}{M}(\mu + L)(4\mu + 2L)} \end{aligned}$$

Finally, we conclude that there exists a constant  $C$  that meets the last inequality. Therefore,  $K = \tilde{O}(\left(\frac{M}{1+q}\right)^{\frac{1}{2}})$  satisfies the condition (eq. (2.11)).

#### 2.4.4 Proof Details for FedAQ under Condition Set (2.2)

We use notations defined in section 2.4.3 here as well. We newly define  $\Phi_{k,t}^m, \Phi_{k,t}, \Phi_k, B_{k,t}^m$  as below.

$$\begin{aligned}\Phi_{k,t}^m &= F(w_{k,t}^{\text{ag},m}) - F^* + \frac{1}{6}\mu\|w_{k,t}^m - w^*\|^2, \quad \Phi_{k,t} = F(\bar{w}_{k,t}^{\text{ag}}) - F^* + \frac{1}{6}\mu\|\bar{w}_{k,t} - w^*\|^2 \\ \Phi_k &:= \Phi_{k,0} = F(w_k^{\text{ag}}) - F^* + \frac{1}{6}\mu\|w_k - w^*\|^2 \\ B_{k,t}^m &= \left(\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2}\right)\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2 + \gamma^2\left(\frac{\mu}{3} + L\right)\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\Phi_{k,t}^m\end{aligned}$$

The flow of proof is similar to section 2.4.3. We need one more condition  $\gamma\mu \leq \frac{3}{4}$  to show the convergence of FedAQ under the parameter condition set (eq. (2.2)).

##### *Proof of lemma 2.4.1*

In order to prove lemma 2.4.1, we first introduce five crucial Propositions for proving lemma 2.4.1. Then, we prove lemma 2.4.1 by using Propositions in the last part of this section.

**Proposition 2.4.14.** *Let assumption 2.2.1 hold and consider any  $k$  synchronization round.*

*Then, we can decompose the expectation as follows:*

$$\begin{aligned}\mathbb{E}[\|w_{k+1} - w^*\|^2] &= \mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] + \mathbb{E}[\|\bar{w}_{k,\tau} - w^*\|^2] \\ \mathbb{E}[F(w_{k+1}^{\text{ag}}) - F^*] &= \mathbb{E}[F(w_{k+1}^{\text{ag}}) - F(\bar{w}_{k,\tau}^{\text{ag}})] + \mathbb{E}[F(\bar{w}_{k,\tau}^{\text{ag}}) - F^*]\end{aligned}$$

*Proof of proposition 2.4.14* The second equality is trivial. The first equality is the same as one in proposition 2.4.7.

**Proposition 2.4.15.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{3}{2\gamma\mu} - \frac{1}{2}$ ,  $\beta = \frac{2\alpha^2-1}{\alpha-1}$ ,  $\gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ ,  $\eta \in (0, \frac{1}{L}]$ , FedAQ yields*

$$\begin{aligned} \mathbb{E}[\Phi_{k,\tau}] &\leq (1 - \frac{1}{3}\gamma\mu)^\tau \mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\tau \sigma^2}{M} \\ &\quad + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] \end{aligned}$$

*Proof of proposition 2.4.15* We refer to the proof of Lemma C.2 in [31]. There is no quantization between  $\Phi_{k,\tau}$  and  $\Phi_k$ . Thus, we can directly apply useful inequalities in the proof of Lemma C.2 in [31] to our proof. Then, we obtain

$$\begin{aligned} &\mathbb{E}[\Phi_{k,t+1} | \mathcal{F}_{k,t}] \\ &\leq (1 - \frac{1}{3}\gamma\mu) \Phi_{k,t} + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\sigma^2}{M} + \gamma \|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2 \end{aligned}$$

From the above relationship between  $\Phi_{k,t+1}$  and  $\Phi_{k,t}$ , we get

$$\begin{aligned} \mathbb{E}[\Phi_{k,\tau}] &\leq (1 - \frac{1}{3}\gamma\mu)^\tau \mathbb{E}[\Phi_k] + \left( \sum_{t=0}^{\tau-1} (1 - \frac{1}{3}\gamma\mu)^t \right) \cdot (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\sigma^2}{M} \\ &\quad + \gamma \sum_{t=0}^{\tau-1} \left\{ (1 - \frac{1}{3}\gamma\mu)^{\tau-t-1} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] \right\} \\ &\leq (1 - \frac{1}{3}\gamma\mu)^\tau \mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\tau \sigma^2}{M} \\ &\quad + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] \end{aligned}$$

**Proposition 2.4.16.** *Let assumption 2.2.1 hold. Then, we have*

$$\begin{aligned}\mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] &\leq \frac{q}{M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] \\ \mathbb{E}[F(w_{k+1}^{\text{ag}}) - F(\bar{w}_{k,\tau}^{\text{ag}})] &\leq \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2]\end{aligned}$$

*Proof of proposition 2.4.16* The first inequality is the same as one in proposition 2.4.9.

The proof of the second inequality is similar to proposition 2.4.9 as well.

$$\begin{aligned}\mathbb{E}[F(w_{k+1}^{\text{ag}}) - F(\bar{w}_{k,\tau}^{\text{ag}})] &= \mathbb{E}[F(w_k^{\text{ag}} + \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})) - F(\frac{1}{M} \sum_{m=1}^M w_{k,\tau}^{\text{ag},m})] \\ &\leq \mathbb{E}\left[\langle \nabla F(\frac{1}{M} \sum_{m=1}^M w_{k,\tau}^{\text{ag},m}), \frac{1}{M} \sum_{m=1}^M (Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})) \rangle \right. \\ &\quad \left. + \frac{L}{2} \left\| \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) \right\|^2 \right] \\ &= \frac{L}{2} \mathbb{E}\left[\left\| \frac{1}{M} \sum_{m=1}^M Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) \right\|^2\right] \\ &= \frac{L}{2M^2} \sum_{m=1}^M \mathbb{E}[\|Q(w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}) - (w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}})\|^2] \leq \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2]\end{aligned}$$

**Proposition 2.4.17.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{3}{2\gamma\mu} - \frac{1}{2}$ ,  $\beta = \frac{2\alpha^2-1}{\alpha-1}$ ,  $\gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ ,  $\eta, \gamma \in (0, \frac{1}{L}]$ ,  $\gamma\mu \leq \frac{3}{4}$ , we get*

$$\begin{aligned}\mathbb{E}[B_{k,t}^m] &\leq \mathbb{E}[B_{k,0}^m] + \left( \left( \frac{\mu}{3} \left( \frac{2\alpha-1}{2\alpha^2-1} \right)^2 + L \left( \frac{\alpha-1}{2\alpha^2-1} \right)^2 \right) \cdot (\gamma - \eta)^2 \right. \\ &\quad \left. + \gamma^4 \left( \frac{\mu}{3} + L \right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L \right) \cdot \frac{1 + \frac{1}{2}\alpha^{-1}}{\frac{1}{4}\alpha^{-2}} \cdot \left( 1 - \left( 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \right)^t \right) \sigma^2\end{aligned}$$

*Proof of proposition 2.4.17* From the notation mentioned in the beginning of sec-

tion 2.4.4,

$$\begin{aligned}\mathbb{E}[B_{k,t+1}^m | \mathcal{F}_{k,t}] &= \left( \frac{\mu\alpha^{-2}}{3} (1 - \beta^{-1})^2 + L\beta^{-2} \right) \mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] \\ &\quad + \gamma^2 \left( \frac{\mu}{3} + L \right) \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L \mathbb{E}[\Phi_{k,t+1}^m | \mathcal{F}_{k,t}]\end{aligned}\tag{2.20}$$

Thus, let's sequentially compute  $\mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}]$  and  $\mathbb{E}[\Phi_{k,t+1}^m | \mathcal{F}_{k,t}]$ .

$$\begin{aligned}\mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] &= \mathbb{E}[\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma g_{k,t}^m - w_{k,t}^{\text{md},m} + \eta g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \\ &= \mathbb{E}[\|(1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}) - (\gamma - \eta)g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \quad (\leftarrow \gamma \geq \eta) \\ &= \|(1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}) - (\gamma - \eta)\nabla F(w_{k,t}^{\text{md},m})\|^2 \\ &\quad + (\gamma - \eta)^2 \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m}) - g_{k,t}^m\|^2 | \mathcal{F}_{k,t}] \\ &\leq (1 - \alpha^{-1})^2 \|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 + (\gamma - \eta)^2 \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + (\gamma - \eta)^2 \sigma^2 \\ &\quad - 2(\gamma - \eta) \langle (1 - \alpha^{-1})(w_{k,t}^m - w_{k,t}^{\text{md},m}), \nabla F(w_{k,t}^{\text{md},m}) \rangle \\ &\leq (1 - \alpha^{-1})^2 (1 + 2\alpha^{-1}) \|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 + (\gamma - \eta)^2 \left(1 + \frac{\alpha}{2}\right) \|\nabla F(w_{k,t}^{\text{md},m})\|^2 \\ &\quad + (\gamma - \eta)^2 \sigma^2\end{aligned}$$

Here, we need to bound  $\|\nabla F(w_{k,t}^{\text{md},m})\|^2$ .

$$\begin{aligned}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 &\leq 2L(F(w_{k,t}^{\text{md},m}) - F^*) \quad (\because \text{assumption 2.2.3}) \\ &\leq 2L\left(\beta^{-1}(F(w_{k,t}^m) - F(w^*)) + (1 - \beta^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*)\right) \\ &\leq \beta^{-1}L^2\|w_{k,t}^m - w^*\|^2 + 2(1 - \beta^{-1})L(F(w_{k,t}^{\text{ag},m}) - F^*) \\ &= \frac{\alpha - 1}{2\alpha^2 - 1}L^2\|w_{k,t}^m - w^*\|^2 + 2L \cdot \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1}(F(w_{k,t}^{\text{ag},m}) - F^*) \\ &\leq \frac{\mu}{3} \frac{(2\alpha^2 - \alpha)}{2\alpha^2 - 1}L\|w_{k,t}^m - w^*\|^2 + 2L \cdot \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1}(F(w_{k,t}^{\text{ag},m}) - F^*) = \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\Phi_{k,t}^m\end{aligned}\tag{2.21}$$

It is easy to show  $(\alpha - 1)L \leq \frac{\mu}{3}(2\alpha^2 - \alpha)$  by using the fact  $\gamma L \leq 1$ . Therefore, we finally get

$$\begin{aligned}
\mathbb{E}[\|w_{k,t+1}^m - w_{k,t+1}^{\text{ag},m}\|^2 | \mathcal{F}_{k,t}] &\leq (1 - \alpha^{-1})^2(1 + 2\alpha^{-1})\|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 \\
&\quad + (\gamma - \eta)^2(1 + \frac{\alpha}{2})\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + (\gamma - \eta)^2\sigma^2 \\
&\leq (1 - \alpha^{-1})^2(1 + 2\alpha^{-1})\|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 + (\gamma - \eta)^2(1 + \frac{\alpha}{2})\left(\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\Phi_{k,t}^m\right) \\
&\quad + (\gamma - \eta)^2\sigma^2 \tag{2.22}
\end{aligned}$$

Now, let's compute  $\mathbb{E}[\Phi_{k,t+1}^m | \mathcal{F}_{k,t}]$ . We need to compute  $\mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}]$  and  $\mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}]$  first.

$$\begin{aligned}
\mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}] &= \mathbb{E}[\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma g_{k,t}^m - w^*\|^2 | \mathcal{F}_{k,t}] \\
&\leq \|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma \nabla F(w_{k,t}^{\text{md},m}) - w^*\|^2 + \gamma^2\sigma^2 \\
&\leq (1 + \frac{1}{2}\alpha^{-1})\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - \gamma \nabla F(w_{k,t}^{\text{md},m}) - w^*\|^2 + \gamma^2\sigma^2 \\
&= (1 + \frac{1}{2}\alpha^{-1})\|(1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*\|^2 + \gamma^2(1 + \frac{1}{2}\alpha^{-1})\|\nabla F(w_{k,t}^{\text{md},m})\|^2 \\
&\quad - 2\gamma(1 + \frac{1}{2}\alpha^{-1})\langle (1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle + \gamma^2\sigma^2 \\
&\leq (1 + \frac{1}{2}\alpha^{-1})\left(\|(1 - \alpha^{-1})w_{k,t}^m - w^*\|^2 + \alpha^{-1}\|w_{k,t}^{\text{md},m} - w^*\|^2\right) + \gamma^2(1 + \frac{1}{2}\alpha^{-1}) \\
&\quad \|\nabla F(w_{k,t}^{\text{md},m})\|^2 - 2\gamma(1 + \frac{1}{2}\alpha^{-1})\langle (1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle + \gamma^2\sigma^2
\end{aligned}$$

It is easy to show  $(1 + \frac{1}{2}\alpha^{-1})(1 - \alpha^{-1}) < 1 - \frac{1}{2}\alpha^{-1}$ ,  $1 + \frac{1}{2}\alpha^{-1} \leq \frac{3}{2}$ . Due to these facts, we

obtain

$$\begin{aligned}
\mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}] &\leq (1 - \frac{1}{2}\alpha^{-1})\|w_{k,t}^m - w^*\|^2 + \frac{3}{2}\alpha^{-1}\|w_{k,t}^{\text{md},m} - w^*\|^2 + \frac{3}{2}\gamma^2\|\nabla \\
&\quad F(w_{k,t}^{\text{md},m})\|^2 - 2\gamma(1 + \frac{1}{2}\alpha^{-1})\langle (1 - \alpha^{-1})w_{k,t}^m + \alpha^{-1}w_{k,t}^{\text{md},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle + \gamma^2\sigma^2 \\
&\leq (1 - \frac{1}{2}\alpha^{-1})\|w_{k,t}^m - w^*\|^2 + \frac{3}{2}\alpha^{-1}\|w_{k,t}^{\text{md},m} - w^*\|^2 + \frac{3}{2}\gamma^2\|\nabla F(w_{k,t}^{\text{md},m})\|^2 - 2\gamma(1 \\
&\quad + \frac{1}{2}\alpha^{-1})\langle (1 - \alpha^{-1}(1 - \beta^{-1}))w_{k,t}^m + \alpha^{-1}(1 - \beta^{-1})w_{k,t}^{\text{ag},m} - w^*, \nabla F(w_{k,t}^{\text{md},m}) \rangle + \gamma^2\sigma^2
\end{aligned}$$

Next, we compute the upper bound of  $\mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}]$ .

$$\begin{aligned}
&\mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}] \\
&\leq \mathbb{E}[F(w_{k,t}^{\text{md},m}) + \langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t+1}^{\text{ag},m} - w_{k,t}^{\text{md},m} \rangle + \frac{L}{2}\|w_{k,t+1}^{\text{ag},m} - w_{k,t}^{\text{md},m}\|^2 - F^* | \mathcal{F}_{k,t}] \\
&\leq F(w_{k,t}^{\text{md},m}) - F^* - \eta\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\sigma^2 \\
&\leq F(w_{k,t}^{\text{md},m}) - F^* - \frac{\eta}{2}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\sigma^2 (\because 1 - \frac{\eta L}{2} \geq \frac{1}{2} \leftarrow \eta \in [0, \frac{1}{L}]) \\
&= (1 - \frac{1}{2}\alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) + \frac{1}{2}\alpha^{-1}(F(w_{k,t}^{\text{md},m}) - F^*) \\
&\quad + (1 - \frac{1}{2}\alpha^{-1})(F(w_{k,t}^{\text{md},m}) - F(w_{k,t}^{\text{ag},m})) - \frac{\eta}{2}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\sigma^2 \\
&\leq (1 - \frac{1}{2}\alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) - \frac{\mu\alpha^{-1}}{4}\|w_{k,t}^{\text{md},m} - w^*\|^2 + \frac{1}{2}\alpha^{-1}\langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t}^{\text{md},m} \\
&\quad - w^* \rangle + (1 - \frac{1}{2}\alpha^{-1})\langle \nabla F(w_{k,t}^{\text{md},m}), w_{k,t}^{\text{md},m} - w_{k,t}^{\text{ag},m} \rangle - \frac{\eta}{2}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\sigma^2 \\
&= (1 - \frac{1}{2}\alpha^{-1})(F(w_{k,t}^{\text{ag},m}) - F^*) - \frac{\mu\alpha^{-1}}{4}\|w_{k,t}^{\text{md},m} - w^*\|^2 - \frac{\eta}{2}\|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{\eta^2 L}{2}\sigma^2 \\
&\quad + \frac{1}{2}\alpha^{-1}\langle \nabla F(w_{k,t}^{\text{md},m}), 2\alpha\beta^{-1}w_{k,t}^m + (1 - 2\alpha\beta^{-1})w_{k,t}^{\text{ag},m} - w^* \rangle
\end{aligned}$$

It is easy to show  $\frac{1}{2}\alpha^{-1} = \frac{\gamma\mu}{3}(1 + \frac{1}{2}\alpha^{-1})$ . Then, we bound  $\mathbb{E}[\Phi_{k,t+1}^m | \mathcal{F}_{k,t}]$  by using the above

results.

$$\begin{aligned}
\mathbb{E}[\Phi_{k,t+1}^m | \mathcal{F}_{k,t}] &= \frac{\mu}{6} \mathbb{E}[\|w_{k,t+1}^m - w^*\|^2 | \mathcal{F}_{k,t}] + \mathbb{E}[F(w_{k,t+1}^{\text{ag},m}) - F^* | \mathcal{F}_{k,t}] \\
&\leq (1 - \frac{1}{2}\alpha^{-1})\Phi_{k,t}^m - \frac{2\eta - \gamma^2\mu}{4} \|\nabla F(w_{k,t}^{\text{md},m})\|^2 + \frac{1}{2}(\frac{\gamma^2\mu}{3} + \eta^2 L)\sigma^2 \\
&\leq (1 - \frac{1}{2}\alpha^{-1})\Phi_{k,t}^m + \frac{1}{2}(\frac{\gamma^2\mu}{3} + \eta^2 L)\sigma^2 (\because \gamma \leq \sqrt{\frac{\eta}{\mu}}) \\
&\leq (1 - \frac{1}{2}\alpha^{-1})\Phi_{k,t}^m + \frac{\gamma^2}{2}(\frac{\mu}{3} + L)\sigma^2
\end{aligned} \tag{2.23}$$

Plugging Equation 2.22, (refineqD.5-4 in Equation 2.20 yields,

$$\begin{aligned}
&\mathbb{E}[B_{k,t+1}^m | \mathcal{F}_{k,t}] \\
&\leq \left(\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2}\right) \left((1 - \alpha^{-1})^2(1 + 2\alpha^{-1})\|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2\right. \\
&\quad \left.+ (\gamma - \eta)^2(1 + \frac{\alpha}{2}) \cdot \left(\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\Phi_{k,t}^m\right) + (\gamma - \eta)^2\sigma^2\right) \\
&\quad \left.+ \gamma^2(\frac{\mu}{3} + L)\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\left((1 - \frac{1}{2}\alpha^{-1})\Phi_{k,t}^m + \frac{\gamma^2}{2}(\frac{\mu}{3} + L)\sigma^2\right)\right) \\
&= (1 - \alpha^{-1})^2(1 + 2\alpha^{-1})\left(\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2}\right)\|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2 + \left(\left(\frac{\mu\alpha^{-2}}{3}\right.\right. \\
&\quad \left.\left.(1 - \beta^{-1})^2 + L\beta^{-2}\right)(\gamma - \eta)^2(1 + \frac{\alpha}{2}) + (1 - \frac{1}{2}\alpha^{-1})\gamma^2(\frac{\mu}{3} + L)\right) \cdot \left(\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1}\right. \\
&\quad \left.\cdot 2L\Phi_{k,t}^m\right) + \left(\left(\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2}\right)(\gamma - \eta)^2 + \gamma^4(\frac{\mu}{3} + L)^2\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1}L\right)\sigma^2
\end{aligned} \tag{2.24}$$

We can show that both coefficients of  $\|w_{k,t}^m - w_{k,t}^{\text{md},m}\|^2$  and  $\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\Phi_{k,t}^m$  are upper bounded by  $1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}$ .

$$\begin{aligned}
(1 - \alpha^{-1})^2(1 + 2\alpha^{-1}) &\leq 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} (< 1) \\
\Leftrightarrow 1 - \frac{1}{4}\alpha^{-2} + \frac{1}{2}\alpha^{-1} - (1 - \alpha^{-1})^2(1 + 2\alpha^{-1})(1 + \frac{1}{2}\alpha^{-1}) &\geq 0
\end{aligned} \tag{2.25}$$

Let's define  $g_1(\alpha^{-1}) = 1 - \frac{1}{4}\alpha^{-2} + \frac{1}{2}\alpha^{-1} - (1 - \alpha^{-1})^2(1 + 2\alpha^{-1})(1 + \frac{1}{2}\alpha^{-1})$ . Then, it is

easy to check that  $g_1(\alpha^{-1}) \geq 0$  for  $0 < \alpha^{-1} \leq 1$ . Moreover, we would like to show the below inequality.

$$\begin{aligned}
& \left( \frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2} \right) (\gamma - \eta)^2 \left( 1 + \frac{\alpha}{2} \right) + \left( 1 - \frac{1}{2}\alpha^{-1} \right) \gamma^2 \left( \frac{\mu}{3} + L \right) \\
& \leq \left( \frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2} \right) \gamma^2 \left( 1 + \frac{\alpha}{2} \right) + \left( 1 - \frac{1}{2}\alpha^{-1} \right) \gamma^2 \left( \frac{\mu}{3} + L \right) \\
& \leq \left( 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \right) \gamma^2 \left( \frac{\mu}{3} + L \right) \tag{2.26}
\end{aligned}$$

Since  $\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2} = \frac{\mu}{3} \left( \frac{2\alpha-1}{2\alpha^2-1} \right)^2 + L \left( \frac{\alpha-1}{2\alpha^2-1} \right)^2 \leq \left( \frac{\mu}{3} + \frac{L}{4} \right) \left( \frac{2\alpha-1}{2\alpha^2-1} \right)^2$ , it is enough to show

$$\left( \frac{\mu}{3} + \frac{L}{4} \right) \left( \frac{2\alpha-1}{2\alpha^2-1} \right)^2 \gamma^2 \left( 1 + \frac{\alpha}{2} \right) \leq \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \gamma^2 \left( \frac{\mu}{3} + L \right)$$

We also know that  $\frac{\frac{\mu}{3} + L}{\frac{\mu}{3} + \frac{L}{4}} = 4 - \frac{1}{\frac{1}{3} + \frac{L}{\mu} \cdot \frac{1}{4}} > \frac{16}{7}$  ( $\because \frac{L}{\mu} > 1$ ). Then, we only need to show

$$\begin{aligned}
& \left( \frac{2\alpha-1}{2\alpha^2-1} \right)^2 \left( 1 + \frac{\alpha}{2} \right) \leq \frac{16}{7} \cdot \frac{\frac{1}{2}}{\alpha + \frac{1}{2}} \\
& \Leftrightarrow \frac{8}{7} (2\alpha^2 - 1)^2 - (2\alpha - 1)^2 \left( 1 + \frac{\alpha}{2} \right) \left( \alpha + \frac{1}{2} \right) \geq 0
\end{aligned}$$

Let's define  $g_2(\alpha) = \frac{8}{7}(2\alpha^2 - 1)^2 - (2\alpha - 1)^2 \left( 1 + \frac{\alpha}{2} \right) \left( \alpha + \frac{1}{2} \right)$ . Then, it is easy to check  $g_2(\alpha) \geq 0$  for  $\alpha \geq \frac{3}{2}$ . As we assume  $\gamma\mu \leq \frac{3}{4}$ , we can say  $\alpha = \frac{3}{2\gamma\mu} - \frac{1}{2} \geq \frac{3}{2}$ . This indicates that the inequality (eq. (2.26)) is satisfied. Thus, from Equation 2.24, Equation 2.25, and Equation 2.26 we finally get

$$\begin{aligned}
\mathbb{E}[B_{k,t+1}^m | \mathcal{F}_{k,t}] & \leq \left( 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \right) B_{k,t}^m \\
& + \left( \left( \frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2} \right) (\gamma - \eta)^2 + \gamma^4 \left( \frac{\mu}{3} + L \right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L \right) \sigma^2
\end{aligned}$$

From this relationship between  $B_{k,t+1}^m$  and  $B_{k,t}^m$ , we obtain the result of proposition 2.4.17.

$$\begin{aligned}
\mathbb{E}[B_{k,t}^m] &\leq \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)^t \mathbb{E}[B_{k,0}^m] + \left(\left(\frac{\mu\alpha^{-2}}{3}(1 - \beta^{-1})^2 + L\beta^{-2}\right)(\gamma - \eta)^2\right. \\
&\quad \left. + \gamma^4\left(\frac{\mu}{3} + L\right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L\right) \sigma^2 \cdot \frac{1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)^t}{1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)} \\
&\leq \mathbb{E}[B_{k,0}^m] + \left(\left(\frac{\mu}{3}\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2 + L\left(\frac{\alpha - 1}{2\alpha^2 - 1}\right)^2\right) \cdot (\gamma - \eta)^2 + \gamma^4\left(\frac{\mu}{3} + L\right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L\right) \\
&\quad \cdot \frac{1 + \frac{1}{2}\alpha^{-1}}{\frac{1}{4}\alpha^{-2}} \cdot \left(1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)^t\right) \sigma^2
\end{aligned}$$

**Proposition 2.4.18.** *Let  $F$  be  $\mu$ -strongly convex, and assume assumption 2.2.2, assumption 2.2.3, assumption 2.2.4, then for  $\alpha = \frac{3}{2\gamma\mu} - \frac{1}{2}$ ,  $\beta = \frac{2\alpha^2 - 1}{\alpha - 1}$ ,  $\gamma \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ ,  $\eta, \gamma \in (0, \frac{1}{L}]$ ,  $\gamma\mu \leq \frac{3}{4}$ ,  $\tau \geq 2$ , FedAQ yields*

$$\begin{aligned}
&\frac{\mu}{6} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
&\leq \left(\gamma^2\mu\left(\frac{8}{3}\mu + 2L\right) + 2\gamma^2L\left(\frac{\mu}{3} + L\right)\right) \tau^2 \mathbb{E}[\Phi_k] + \left(\frac{\gamma^2\mu}{3} + \eta^2L\right) \tau \sigma^2 \\
&\quad + \left((\gamma - \eta)^2\gamma^2\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + \gamma^4\left(\frac{\mu}{3} + L\right)^2L\right) \frac{\tau^3\sigma^2}{2}
\end{aligned}$$

*Proof of proposition 2.4.18* We use the same upper bounds for  $\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2]$  and  $\mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2]$  as in proposition 2.4.11.

$$\begin{aligned}
\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] &\leq \tau \left( \sum_{t=0}^{\tau-1} 2\alpha^{-2}(1 - \beta^{-1})^2 \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] + 2\gamma^2 \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m})\|^2] \right) \\
&\quad + 2\tau\gamma^2\sigma^2 \\
\mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] &\leq \tau \left( \sum_{t=0}^{\tau-1} 2\beta^{-2} \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] + 2\eta^2 \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m})\|^2] \right) + 2\tau\eta^2\sigma^2
\end{aligned}$$

Thus, by using the above results, we get

$$\begin{aligned}
& \frac{\mu}{6} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
& \leq \tau \sum_{t=0}^{\tau-1} \left\{ \left( \frac{\mu\alpha^{-2}}{3} (1 - \beta^{-1})^2 + L\beta^{-2} \right) \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] \right. \\
& \quad \left. + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \mathbb{E}[\|\nabla F(w_{k,t}^{\text{md},m})\|^2] \right\} + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \tau \sigma^2 \\
& \leq \tau \sum_{t=0}^{\tau-1} \left\{ \left( \frac{\mu\alpha^{-2}}{3} (1 - \beta^{-1})^2 + L\beta^{-2} \right) \mathbb{E}[\|w_{k,t}^m - w_{k,t}^{\text{ag},m}\|^2] \right. \\
& \quad \left. + \gamma^2 \left( \frac{\mu}{3} + L \right) \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} 2L \mathbb{E}[\Phi_{k,t}^m] \right\} + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \tau \sigma^2 (\because \text{Equation 2.21}) \\
& = \tau \left( \sum_{t=0}^{\tau-1} \mathbb{E}[B_{k,t}^m] \right) + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \tau \sigma^2
\end{aligned}$$

By proposition 2.4.17 and the fact  $\Phi_{k,0}^m = \Phi_k$ , we obtain

$$\begin{aligned}
& \frac{\mu}{6} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
& \leq \tau \left\{ \sum_{t=0}^{\tau-1} \mathbb{E}[B_{k,t}^m] + \left( \left( \frac{\mu}{3} \left( \frac{2\alpha - 1}{2\alpha^2 - 1} \right)^2 + L \left( \frac{\alpha - 1}{2\alpha^2 - 1} \right)^2 \right) (\gamma - \eta)^2 + \gamma^4 \left( \frac{\mu}{3} + L \right)^2 \right. \right. \\
& \quad \left. \left. \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L \right) \frac{1 + \frac{1}{2}\alpha^{-1}}{\frac{1}{4}\alpha^{-2}} \left( 1 - \left( 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \right)^t \right) \sigma^2 \right\} + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \tau \sigma^2 \\
& = \tau^2 \left( \left( \frac{\mu\alpha^{-2}}{3} (1 - \beta^{-1})^2 + L\beta^{-2} \right) \mathbb{E}[\|w_k - w_k^{\text{ag}}\|^2] + \gamma^2 \left( \frac{\mu}{3} + L \right) \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L \mathbb{E}[\Phi_k] \right) \\
& \quad + \tau \left( \left( \frac{\mu}{3} \left( \frac{2\alpha - 1}{2\alpha^2 - 1} \right)^2 + L \left( \frac{\alpha - 1}{2\alpha^2 - 1} \right)^2 \right) \cdot (\gamma - \eta)^2 + \gamma^4 \left( \frac{\mu}{3} + L \right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L \right) \frac{1 + \frac{1}{2}\alpha^{-1}}{\frac{1}{4}\alpha^{-2}} \\
& \quad \cdot \left( \sum_{t=0}^{\tau-1} 1 - \left( 1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}} \right)^t \right) \sigma^2 + \left( \frac{\gamma^2\mu}{3} + \eta^2 L \right) \tau \sigma^2
\end{aligned}$$

Before we get to the final result, let's find the upper bound for  $\|w_k - w_k^{\text{ag}}\|^2$ ,  $\sum_{t=0}^{\tau-1} \left( 1 - \right.$

$$\left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)t$$

$$\begin{aligned} \|w_k - w_k^{\text{ag}}\|^2 &= \|w_k - w^* - (w_k^{\text{ag}} - w^*)\|^2 \\ &\leq \left(1 + \frac{1}{3}\right)\|w_k - w^*\|^2 + (1 + 3)\|w_k^{\text{ag}} - w^*\|^2 \\ &\leq \frac{4}{3}\|w_k - w^*\|^2 + 4 \cdot \frac{2}{\mu} \left(F(w_k^{\text{ag}}) - F^* - \langle \nabla F(w^*), w_k^{\text{ag}} - w^* \rangle\right) \\ &= \frac{4}{3}\|w_k - w^*\|^2 + \frac{8}{\mu} (F(w_k^{\text{ag}}) - F^*) = \frac{8}{\mu} \Phi_k \end{aligned}$$

$$\begin{aligned} \sum_{t=0}^{\tau-1} \left(1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)t\right) &= \tau - \sum_{t=0}^{\tau-1} \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)^t \\ &= \tau - \frac{1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)^\tau}{1 - \left(1 - \frac{1}{2}\alpha^{-1} + \frac{\frac{1}{2}\alpha^{-1}}{1 + \frac{1}{2}\alpha^{-1}}\right)} \\ &\leq \tau - \frac{1 - \left(1 - \frac{\frac{1}{4}\alpha^{-2}}{1 + \frac{1}{2}\alpha^{-1}}\tau + \left(\frac{\frac{1}{4}\alpha^{-2}}{1 + \frac{1}{2}\alpha^{-1}}\right)^2 \frac{\tau(\tau-1)}{2}\right)}{\frac{\frac{1}{4}\alpha^{-2}}{1 + \frac{1}{2}\alpha^{-1}}} \\ &= \frac{\frac{1}{4}\alpha^{-2}}{1 + \frac{1}{2}\alpha^{-1}} \cdot \frac{\tau(\tau-1)}{2} \leq \frac{\frac{1}{4}\alpha^{-2}}{1 + \frac{1}{2}\alpha^{-1}} \cdot \frac{\tau^2}{2} \end{aligned}$$

Therefore, we obtain

$$\begin{aligned} &\frac{\mu}{6} \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2} \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\ &\leq \left(\frac{8}{3}\alpha^{-2}(1 - \beta^{-1})^2 + \frac{8L}{\mu}\beta^{-2} + \gamma^2\left(\frac{\mu}{3} + L\right)\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L\right)\tau^2 \mathbb{E}[\Phi_k] + \left(\frac{\gamma^2\mu}{3} + \eta^2 L\right)\tau\sigma^2 \\ &\quad + \left(\left(\frac{\mu}{3}\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2 + L\left(\frac{\alpha - 1}{2\alpha^2 - 1}\right)^2\right) \cdot (\gamma - \eta)^2 + \gamma^4\left(\frac{\mu}{3} + L\right)^2 \frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} L\right) \cdot \frac{\tau^3\sigma^2}{2} \end{aligned} \tag{2.27}$$

Moreover, we can simplify the above inequality by replacing  $\alpha, \beta$  with  $\gamma, \mu$ . It is easy to

show  $\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \leq 1$ ,  $\frac{2\alpha - 1}{2\alpha^2 - 1} \leq \frac{1}{\alpha} = \frac{2\gamma\mu}{3 - \gamma\mu} \leq \gamma\mu$ . Then, we can further show

$$\begin{aligned}
& \frac{8}{3}\alpha^{-2}(1 - \beta^{-1})^2 + \frac{8L}{\mu}\beta^{-2} + \gamma^2\left(\frac{\mu}{3} + L\right)\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L \\
&= \frac{8}{3}\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2 + \frac{8L}{\mu}\left(\frac{\alpha - 1}{2\alpha^2 - 1}\right)^2 + \gamma^2\left(\frac{\mu}{3} + L\right)\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1} \cdot 2L \\
&\leq \left(\frac{8}{3} + \frac{2L}{\mu}\right)\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2 + \gamma^2\left(\frac{\mu}{3} + L\right)2L \\
&\leq \left(\frac{8}{3} + \frac{2L}{\mu}\right)\alpha^{-2} + \gamma^2\left(\frac{\mu}{3} + L\right)2L \\
&\leq \gamma^2\mu\left(\frac{8}{3}\mu + 2L\right) + 2\gamma^2L\left(\frac{\mu}{3} + L\right)
\end{aligned} \tag{2.28}$$

We also get

$$\begin{aligned}
& \left(\frac{\mu}{3}\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2 + L\left(\frac{\alpha - 1}{2\alpha^2 - 1}\right)^2\right) \cdot (\gamma - \eta)^2 + \gamma^4\left(\frac{\mu}{3} + L\right)^2\frac{2\alpha^2 - \alpha}{2\alpha^2 - 1}L \\
&\leq \left(\frac{\mu}{3} + \frac{L}{4}\right)\left(\frac{2\alpha - 1}{2\alpha^2 - 1}\right)^2(\gamma - \eta)^2 + \gamma^4\left(\frac{\mu}{3} + L\right)^2L \\
&\leq (\gamma - \eta)^2\gamma^2\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + \gamma^4\left(\frac{\mu}{3} + L\right)^2L
\end{aligned} \tag{2.29}$$

Finally, from Equation 2.27, Equation 2.28, and Equation 2.29, we conclude as below

$$\begin{aligned}
& \frac{\mu}{6}\mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{L}{2}\mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
&\leq \left(\gamma^2\mu\left(\frac{8}{3}\mu + 2L\right) + 2\gamma^2L\left(\frac{\mu}{3} + L\right)\right)\tau^2\mathbb{E}[\Phi_k] + \left(\frac{\gamma^2\mu}{3} + \eta^2L\right)\tau\sigma^2 \\
&\quad + \left((\gamma - \eta)^2\gamma^2\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + \gamma^4\left(\frac{\mu}{3} + L\right)^2L\right)\frac{\tau^3\sigma^2}{2}
\end{aligned}$$

*Proof of lemma 2.4.1* By the definition of  $\Phi_k$ ,  $\Phi_{k,t}$  and proposition 2.4.14,

$$\mathbb{E}[\Phi_{k+1}] = \mathbb{E}[\Phi_{k,\tau}] + \frac{\mu}{6}\mathbb{E}[\|w_{k+1} - \bar{w}_{k,\tau}\|^2] + \mathbb{E}[F(w_{k+1}^{\text{ag}}) - F(\bar{w}_{k,\tau}^{\text{ag}})]$$

Applying proposition 2.4.15 and proposition 2.4.16, we have

$$\begin{aligned}
& \mathbb{E}[\Phi_{k+1}] \\
& \leq (1 - \frac{1}{3}\gamma\mu)^\tau \mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\tau \sigma^2}{M} + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) \\
& \quad - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] + \frac{q\mu}{6M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^m - w_k\|^2] + \frac{qL}{2M^2} \sum_{m=1}^M \mathbb{E}[\|w_{k,\tau}^{\text{ag},m} - w_k^{\text{ag}}\|^2] \\
& \leq (1 - \frac{1}{3}\gamma\mu)^\tau \mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\tau \sigma^2}{M} + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) \\
& \quad - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] + \frac{q}{M} \left[ (\gamma^2 \mu (\frac{8}{3}\mu + 2L) + 2\gamma^2 L (\frac{\mu}{3} + L)) \tau^2 \mathbb{E}[\Phi_k] \right. \\
& \quad \left. + (\frac{\gamma^2 \mu}{3} + \eta^2 L) \tau \sigma^2 + \left( (\gamma - \eta)^2 \gamma^2 \mu^2 (\frac{\mu}{3} + \frac{L}{4}) + \gamma^4 (\frac{\mu}{3} + L)^2 L \right) \frac{\tau^3 \sigma^2}{2} \right] \\
& = D(\gamma, \tau) \mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6}) \frac{\tau \sigma^2}{M} + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m}) \\
& \quad \|^2] + \frac{q}{M} (\frac{\gamma^2 \mu}{3} + \eta^2 L) \tau \sigma^2 + \frac{q}{2M} \left( (\gamma - \eta)^2 \gamma^2 \mu^2 (\frac{\mu}{3} + \frac{L}{4}) + \gamma^4 (\frac{\mu}{3} + L)^2 L \right) \tau^3 \sigma^2
\end{aligned}$$

The second inequality comes from proposition 2.4.18.  $D(\gamma, \tau)$  is defined as below.

$$D(\gamma, \tau) = (1 - \frac{1}{3}\gamma\mu)^\tau + \frac{q}{M} \left( \gamma^2 \mu (\frac{8}{3}\mu + 2L) + 2\gamma^2 L (\frac{\mu}{3} + L) \right) \tau^2$$

*Proof of theorem 2.4.2*

*Proof of theorem 2.4.2* At first, due to the condition (eq. (2.3)) in theorem 2.4.2, we get

$$\begin{aligned}
D(\gamma, \tau) &= (1 - \frac{1}{3}\gamma\mu)^\tau + \frac{q}{M} \left( \gamma^2 \mu (\frac{8}{3}\mu + 2L) + 2\gamma^2 L (\frac{\mu}{3} + L) \right) \tau^2 \\
&\leq 1 - \frac{1}{3}\gamma\mu\tau + \frac{1}{9}\gamma^2 \mu^2 \tau^2 + \frac{q}{M} \gamma^2 \left( \mu (\frac{8}{3}\mu + 2L) + 2L (\frac{\mu}{3} + L) \right) \tau^2 \\
&= 1 - \frac{1}{3}\gamma\mu\tau + \left( \frac{1}{9}\mu^2 + \frac{q}{M} \left( \mu (\frac{8}{3}\mu + 2L) + 2L (\frac{\mu}{3} + L) \right) \right) \gamma^2 \tau^2 \\
&\leq 1 - \frac{1}{6}\gamma\mu\tau \quad (\because \text{condition (eq. (2.3))})
\end{aligned}$$

It is trivial that  $\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}}) \in [\eta, \sqrt{\frac{\eta}{\mu}}]$ . Thus, we can use lemma 2.4.1. By using lemma 2.4.1 and the above result, we obtain

$$\begin{aligned}
\mathbb{E}[\Phi_{k+1}] &\leq (1 - \frac{1}{6}\gamma\mu\tau)\mathbb{E}[\Phi_k] + (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6})\frac{\tau\sigma^2}{M} \\
&\quad + \gamma\tau \cdot \max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] + \frac{q}{M}(\frac{\gamma^2 \mu}{3} + \eta^2 L)\tau\sigma^2 \\
&\quad + \frac{q}{2M} \left( (\gamma - \eta)^2 \gamma^2 \mu^2 (\frac{\mu}{3} + \frac{L}{4}) + \gamma^4 (\frac{\mu}{3} + L)^2 L \right) \tau^3 \sigma^2 \tag{2.30}
\end{aligned}$$

By the Lemma C.14 in [31], we know that the below quantity is bounded.

$$\begin{aligned}
\max_{0 \leq t < \tau} \mathbb{E}[\|\nabla F(\bar{w}_{k,t}^{\text{md}}) - \frac{1}{M} \sum_{m=1}^M \nabla F(w_{k,t}^{\text{md},m})\|^2] &\leq B' \\
B' &= \begin{cases} 4\eta^2 L^2 \tau \sigma^2 \left(1 + \frac{\gamma^2 \mu}{\eta}\right)^{2\tau}, & \text{if } \gamma \in \left(\eta, \sqrt{\frac{\eta}{\mu}}\right] \\ 4\eta^2 L^2 \tau \sigma^2, & \text{if } \gamma = \eta \end{cases}
\end{aligned}$$

Telescoping Equation 2.30 yields

$$\begin{aligned}
\mathbb{E}[\Phi_K] &\leq (1 - \frac{1}{6}\gamma\mu\tau)^K \Phi_0 + \left( \sum_{k'=0}^{K-1} (1 - \frac{1}{6}\gamma\mu\tau)^{k'} \right) \cdot \left[ (\frac{\eta^2 L}{2} + \frac{\gamma^2 \mu}{6})\frac{\tau\sigma^2}{M} + \frac{q}{M}(\frac{\gamma^2 \mu}{3} \right. \\
&\quad \left. + \eta^2 L)\tau\sigma^2 + \frac{q}{2M} \left( (\gamma - \eta)^2 \gamma^2 \mu^2 (\frac{\mu}{3} + \frac{L}{4}) + \gamma^4 (\frac{\mu}{3} + L)^2 L \right) \tau^3 \sigma^2 + \gamma\tau B' \right] \\
&\leq \exp\left(-\frac{\gamma\mu\tau K}{6}\right) \Phi_0 + \frac{3\eta^2 L\sigma^2}{\gamma\mu M} + \frac{\gamma\sigma^2}{M} + \frac{6B'}{\mu} + 2q\left(\frac{\gamma\sigma^2}{M} + \frac{3\eta^2 L\sigma^2}{\gamma\mu M}\right) \\
&\quad + \frac{3q}{M} \left( (\gamma - \eta)^2 \gamma\mu (\frac{\mu}{3} + \frac{L}{4}) + \frac{\gamma^3 (\frac{\mu}{3} + L)^2 L}{\mu} \right) \tau^2 \sigma^2
\end{aligned}$$

The last inequality comes from the fact that  $\sum_{k'=0}^{K-1} (1 - \frac{1}{6}\gamma\mu\tau)^{k'} \leq \frac{6}{\gamma\mu\tau}$ . Since we plug in  $\gamma = \max(\eta, \sqrt{\frac{\eta}{\mu\tau}})$ , we can use Lemma C.15 in [31]. Therefore, we obtain

$$\begin{aligned} \mathbb{E}[\Phi_K] &\leq \exp\left(-\frac{1}{6}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Phi_0 + \frac{2(2q+1)\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{4(2q+1)\eta^2L^2\tau\sigma^2}{\mu} \\ &\quad + \frac{24e^2\eta^2L^2\tau\sigma^2}{\mu} + \frac{3q\tau^2\sigma^2}{M} \max\left(\frac{\eta^{\frac{3}{2}}\mu(\frac{\mu}{3} + \frac{L}{4})}{\mu^{\frac{3}{2}}\tau^{\frac{3}{2}}}, \frac{\eta^{\frac{3}{2}}(\frac{\mu}{3} + L)^2L}{\mu^{\frac{5}{2}}\tau^{\frac{3}{2}}}, \frac{\eta^3(\frac{\mu}{3} + L)^2L}{\mu}\right) \end{aligned}$$

The first term stems directly from Lemma C.15 in [31]. Also, the last term comes from the fact that

$$(\gamma - \eta)^2\gamma\mu\left(\frac{\mu}{3} + \frac{L}{4}\right) + \frac{\gamma^3(\frac{\mu}{3} + L)^2L}{\mu} \leq \begin{cases} \gamma^3\mu\left(\frac{\mu}{3} + \frac{L}{4}\right) + \frac{\gamma^3(\frac{\mu}{3} + L)^2L}{\mu}, & \text{if } \gamma \neq \eta \\ \frac{\eta^3(\frac{\mu}{3} + L)^2L}{\mu}, & \text{if } \gamma = \eta \end{cases}$$

Therefore, by simple inequalities such as  $\max(a, b) \leq a + b$  and  $\min(a, b) \leq a$ , we ultimately get

$$\begin{aligned} \mathbb{E}[\Phi_K] &\leq \exp\left(-\frac{1}{6}\max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}})K\tau\right)\Phi_0 + \frac{2(2q+1)\eta^{\frac{1}{2}}\sigma^2}{\mu^{\frac{1}{2}}M\tau^{\frac{1}{2}}} + \frac{8(q+25)\eta^2L^2\tau\sigma^2}{\mu} \\ &\quad + \frac{3q\left(\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + L\left(\frac{\mu}{3} + L\right)^2\right)\eta^{\frac{3}{2}}\tau^{\frac{1}{2}}\sigma^2}{\mu^{\frac{5}{2}}M} + \frac{3qL\left(\frac{\mu}{3} + L\right)^2\eta^3\tau^2\sigma^2}{\mu M} \end{aligned} \quad (2.31)$$

*Proof of corollary 2.4.19*

**Corollary 2.4.19.** *Let  $D_1, D_2$ , and  $\eta_0$  as below. Note that  $T = K\tau$ .*

$$\begin{aligned} D_1 &= \frac{\left(\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + L\left(\frac{\mu}{3} + L\right)^2\right)q}{\mu^{\frac{5}{2}}}, \quad D_2 = \frac{q\left(\frac{\mu}{3} + L\right)^2L}{\mu} \\ \eta_0 &= \frac{36\tau}{\mu T^2} \log^2\left(e + \min\left(\frac{\mu MT\Phi_0}{(2q+1)\sigma^2}, \frac{\mu^3 T^4\Phi_0}{(q+25)L^2\tau^3\sigma^2}, \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}\right)\right) \end{aligned}$$

Then for  $\eta = \min(\frac{1}{L}, \eta_0)$ , FedAQ yields

$$\begin{aligned} \mathbb{E}[\Phi_K] &\leq \min\left(\exp\left(-\frac{\mu T}{6L}\right), \exp\left(-\frac{\mu^{\frac{1}{2}} T}{6L^{\frac{1}{2}} \tau^{\frac{1}{2}}}\right)\right) \Phi_0 \\ &\quad + \frac{13(2q+1)\sigma^2}{\mu MT} \log^2\left(e + \frac{\mu MT \Phi_0}{(2q+1)\sigma^2}\right) \end{aligned} \quad (2.32)$$

$$+ \frac{10369(q+25)L^2 \tau^3 \sigma^2}{\mu^3 T^4} \log^4\left(e + \frac{\mu^3 T^4 \Phi_0}{(q+25)L^2 \tau^3 \sigma^2}\right) \quad (2.33)$$

$$+ \frac{649(\mu^{\frac{3}{2}} D_1 + 216 D_2) \tau^2 \sigma^2}{\mu^3 M T^3} \log^6\left(e + \frac{\mu^3 M T^3 \Phi_0}{(\mu^{\frac{3}{2}} D_1 + 216 D_2) \tau^2 \sigma^2}\right) \quad (2.34)$$

*Proof of corollary 2.4.19* Let's decompose the final result Equation 2.31 of the theorem 2.4.2 into a decreasing term and an increasing term. We denote the decreasing term  $\phi_1$  and the increasing term  $\phi_2$  as below.

$$\begin{aligned} \phi_1(\eta) &= \exp\left(-\frac{1}{6} \max(\eta\mu, \sqrt{\frac{\eta\mu}{\tau}} T)\right) \Phi_0 \\ \phi_2(\eta) &= \frac{2(2q+1)\eta^{\frac{1}{2}} \sigma^2}{\mu^{\frac{1}{2}} M \tau^{\frac{1}{2}}} + \frac{8(q+25)\eta^2 L^2 \tau \sigma^2}{\mu} + \frac{3q\left(\mu^2\left(\frac{\mu}{3} + \frac{L}{4}\right) + L\left(\frac{\mu}{3} + L\right)^2\right)\eta^{\frac{3}{2}} \tau^{\frac{1}{2}} \sigma^2}{\mu^{\frac{5}{2}} M} \\ &\quad + \frac{3qL\left(\frac{\mu}{3} + L\right)^2 \eta^3 \tau^2 \sigma^2}{\mu M} \end{aligned}$$

Since  $\phi_1$  is the decreasing term, we have

$$\phi_1(\eta) \leq \phi_1\left(\frac{1}{L}\right) + \phi_1(\eta_0) \quad (2.35)$$

where

$$\begin{aligned}
\phi_1\left(\frac{1}{L}\right) &= \min\left(\exp\left(-\frac{\mu T}{6L}\right), \exp\left(-\frac{\mu^{\frac{1}{2}}T}{6L^{\frac{1}{2}}\tau^{\frac{1}{2}}}\right)\right)\Phi_0 \\
\phi_1(\eta_0) &\leq \exp\left(-\frac{1}{6}\sqrt{\frac{\eta_0\mu}{\tau}}T\right) \\
&= \left(e + \min\left(\frac{\mu MT\Phi_0}{(2q+1)\sigma^2}, \frac{\mu^3 T^4\Phi_0}{(q+25)L^2\tau^3\sigma^2}, \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}\right)\right)^{-1}\Phi_0 \\
&\leq \frac{(2q+1)\sigma^2}{\mu MT} + \frac{(q+25)L^2\tau^3\sigma^2}{\mu^3 T^4} + \frac{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}{\mu^3 MT^3}
\end{aligned}$$

Since  $\phi_2$  is the increasing term, we have

$$\begin{aligned}
\phi_2(\eta) &\leq \phi_2(\eta_0) \\
&\leq \frac{12(2q+1)\sigma^2}{\mu MT} \log\left(e + \frac{\mu MT\Phi_0}{(2q+1)\sigma^2}\right) + \frac{8 \cdot 36^2(q+25)L^2\tau^3\sigma^2}{\mu^3 T^4} \log^4\left(e + \frac{\mu^3 T^4\Phi_0}{(q+25)L^2\tau^3\sigma^2}\right) \\
&\quad + \frac{3 \cdot 6^3 D_1 \tau^2 \sigma^2}{\mu^{\frac{3}{2}} MT^3} \log^3\left(e + \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}\right) \\
&\quad + \frac{3 \cdot 36^3 D_2 \tau^5 \sigma^2}{\mu^3 MT^6} \log^6\left(e + \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}\right) \\
&\leq \frac{12(2q+1)\sigma^2}{\mu MT} \log\left(e + \frac{\mu MT\Phi_0}{(2q+1)\sigma^2}\right) + \frac{8 \cdot 36^2(q+25)L^2\tau^3\sigma^2}{\mu^3 T^4} \log^4\left(e + \frac{\mu^3 T^4\Phi_0}{(q+25)L^2\tau^3\sigma^2}\right) \\
&\quad + \frac{3 \cdot 6^3(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}{\mu^3 MT^3} \log^6\left(e + \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 6^3D_2)\tau^2\sigma^2}\right)
\end{aligned} \tag{2.36}$$

The last inequality comes from  $\frac{\tau}{T} \leq 1$ . Therefore, by combining Equation 2.35 and

Equation 2.36, we finally get

$$\begin{aligned}
\mathbb{E}[\Phi_K] &\leq \phi_1(\eta) + \phi_2(\eta) \leq \phi_1\left(\frac{1}{L}\right) + \phi_1(\eta_0) + \phi_2(\eta_0) \\
&\leq \min\left(\exp\left(-\frac{\mu T}{6L}\right), \exp\left(-\frac{\mu^{\frac{1}{2}}T}{6L^{\frac{1}{2}}\tau^{\frac{1}{2}}}\right)\right)\Phi_0 + \frac{13(2q+1)\sigma^2}{\mu MT} \log^2\left(e + \frac{\mu MT\Phi_0}{(2q+1)\sigma^2}\right) \\
&\quad + \frac{10369(q+25)L^2\tau^3\sigma^2}{\mu^3 T^4} \log^4\left(e + \frac{\mu^3 T^4\Phi_0}{(q+25)L^2\tau^3\sigma^2}\right) \\
&\quad + \frac{649(\mu^{\frac{3}{2}}D_1 + 216D_2)\tau^2\sigma^2}{\mu^3 MT^3} \log^6\left(e + \frac{\mu^3 MT^3\Phi_0}{(\mu^{\frac{3}{2}}D_1 + 216D_2)\tau^2\sigma^2}\right)
\end{aligned}$$

#### 2.4.5 More Theoretical Details about Remark 2.4.5 and Contribution 2 in Introduction

##### *Why Haddadpour et al. (2021) Cannot Achieve a Linear Speedup*

It is hard to say that [30] achieves a linear speedup in  $M$  in strongly-convex and homogeneous settings. Let's first recap Corollary D.8 in [30]. They let  $\eta\gamma\mu\tau \leq \frac{1}{2}$ ,  $\kappa = \frac{L}{\mu}$ ,  $\gamma \geq M$  and tune  $\eta$  as  $\eta = \frac{1}{2L(\frac{q}{M}+1)\tau\gamma}$ . Here,  $\eta$  is the client learning rate, and  $\gamma$  is the server learning rate. Other parameters are the same as we defined. Then, they obtain the below result.

$$\begin{aligned}
\mathbb{E}[F(w_K) - F^*] &\leq \exp(-\eta\gamma\mu\tau K)(F(w_0) - F^*) + \frac{1}{\mu} \left[ \frac{1}{2}\tau L^2\eta^2\sigma^2 + (1+q)\frac{\gamma\eta L\sigma^2}{2M} \right] \\
&\tag{2.37}
\end{aligned}$$

$$\begin{aligned}
&\leq \mathcal{O}\left(\exp\left(-\frac{K}{2(\frac{q}{M}+1)\kappa}\right)(F(w_0) - F^*) + \frac{\sigma^2}{\gamma^2\mu\tau} + \frac{(q+1)\sigma^2}{\mu(\frac{q}{M}+1)\tau M}\right) \\
&= \mathcal{O}\left(\exp\left(-\frac{K}{2(\frac{q}{M}+1)\kappa}\right)(F(w_0) - F^*) + \frac{\sigma^2 K}{\gamma^2\mu T} + \frac{(q+1)K\sigma^2}{\mu(\frac{q}{M}+1)TM}\right)
\end{aligned}$$

Let's focus on the second and third term. We assume  $M$  is large enough and represent them only with  $\gamma, K, T, M$  to easily check the linear speedup of this convergence rate. Then, we obtain

$$\mathcal{O}\left(\frac{K}{\gamma^2 T} + \frac{K}{MT}\right) \leq \mathcal{O}\left(\frac{K}{M^2 T} + \frac{K}{MT}\right) (\because \gamma \geq M) \tag{2.38}$$

Thus, it seemingly achieves a linear speedup in  $M$  when  $K$  is just a constant. However, we are missing the critical point in this analysis. To be specific, let's consider the case when  $\gamma = 1$ . Then, the convergence rate (eq. (2.38)) changes into  $\mathcal{O}\left(\frac{K}{T} + \frac{K}{MT}\right)$  that cannot achieve a linear speedup in  $M$ . This is implausible because the convergence rate (eq. (2.37)) becomes tighter when  $\gamma = 1$  than  $\gamma \geq M$  (See the last term of (eq. (2.37))). Actually, we can achieve a linear speedup in  $M$  when  $\gamma = 1$  if we tune  $\eta = \frac{1}{2L(\frac{q}{M}+1)\tau M}$ . However, this is not an appropriate tuning because there is  $M$  in the denominator. Similarly, [30] tunes  $\eta = \frac{1}{2L(\frac{q}{M}+1)\tau\gamma}$  where  $\gamma \geq M$ . Even though there is no  $M$  in the denominator, the condition  $\gamma \geq M$  forcibly makes the convergence rate achieve a linear speedup without any theoretical benefits of the algorithm. Therefore, we cannot say their  $\eta$  makes their algorithm achieve a linear speedup in  $M$ . We should tune in a different way that does not contain  $M$  in a denominator. For reference, our tuning parameter  $\eta$  for the FedAQ algorithm does not contain  $M$  in the denominator (See corollary 2.4.13 and corollary 2.4.19).

#### *New Convergence Rate for Haddadpour et al. (2021)*

We propose new  $\eta$  and convergence rate for [30]. This new  $\eta$  makes the algorithm achieve a linear speedup in  $M$ . Let's denote  $\Phi_0 = F(w_0) - F^*$ . We also know that  $T = K\tau$ . Then, we choose  $\eta$  as

$$\eta = \frac{1}{\gamma\mu T} \log \left( e + \min\left(\frac{\gamma^2\mu^3T^2\Phi_0}{\tau L^2\sigma^2}, \frac{\mu^2MT\Phi_0}{(1+q)L\sigma^2}\right) \right)$$

We plug in this  $\eta$  to (eq. (2.37)). We bound the first term as below.

$$\begin{aligned} \exp(-\eta\gamma\mu\tau K)(F(w_0) - F^*) &= \left( e + \min\left(\frac{\gamma^2\mu^3T^2\Phi_0}{\tau L^2\sigma^2}, \frac{\mu^2MT\Phi_0}{(1+q)L\sigma^2}\right) \right)^{-1} \Phi_0 \\ &\leq \frac{\tau L^2\sigma^2}{\gamma^2\mu^3T^2} + \frac{(1+q)L\sigma^2}{\mu^2MT} \end{aligned}$$

The another terms are bounded as below.

$$\begin{aligned} \frac{1}{\mu} \left[ \frac{1}{2} \tau L^2 \eta^2 \sigma^2 + (1+q) \frac{\gamma \eta L \sigma^2}{2M} \right] &\leq \frac{\tau L^2 \sigma^2}{2\gamma^2 \mu^3 T^2} \log^2 \left( e + \frac{\gamma^2 \mu^3 T^2 \Phi_0}{\tau L^2 \sigma^2} \right) \\ &+ \frac{(1+q) L \sigma^2}{2\mu^2 M T} \log \left( e + \frac{\mu^2 M T \Phi_0}{(1+q) L \sigma^2} \right) \end{aligned}$$

Thus, we obtain a new convergence rate by combining the above two bounds.

$$\begin{aligned} \mathbb{E}[F(w_K) - F^*] &\leq \exp(-\eta \gamma \mu \tau K) (F(w_0) - F^*) + \frac{1}{\mu} \left[ \frac{1}{2} \tau L^2 \eta^2 \sigma^2 + (1+q) \frac{\gamma \eta L \sigma^2}{2M} \right] \\ &\leq \frac{3\tau L^2 \sigma^2}{2\gamma^2 \mu^3 T^2} \log^2 \left( e + \frac{\gamma^2 \mu^3 T^2 \Phi_0}{\tau L^2 \sigma^2} \right) + \frac{3(1+q) L \sigma^2}{2\mu^2 M T} \log \left( e + \frac{\mu^2 M T \Phi_0}{(1+q) L \sigma^2} \right) \end{aligned}$$

Here, we replace  $\tau$  with  $\frac{T}{K}$ . Then, we represent the above convergence rate with only  $T, K, M, q$ .

$$\tilde{O}\left(\frac{1}{TK} + \frac{1+q}{MT}\right)$$

This is the new convergence rate we propose in remark 2.4.5. We also get  $K = \tilde{O}\left(\frac{M}{1+q}\right)$  communication rounds make this algorithm achieve a linear speedup in  $M$ .

*More Details on Contribution item 2 in Introduction*

**More Details on  $d_{\text{quant}}$**  This paragraph explains why FedAQ needs to send only  $d_{\text{quant}} = O(\log \frac{1}{q})$  bits for each value. We use the result of Lemma 3.1 in [32]. They show the below result with a low-precision quantizer (Example 1 in section 2.2.2)

$$\mathbb{E}[\|Q(x, s) - x\|_2^2] \leq \min\left(\frac{n}{s^2}, \frac{\sqrt{n}}{s}\right) \|x\|_2^2$$

where  $n$  is the dimension of  $x$ , and  $s$  is the number of quantization levels. Then, we regard  $q$  as

$$q = \frac{\sqrt{n}}{s} = \frac{\sqrt{n}}{2^{d_{\text{quant}}}} \quad (2.39)$$

Thus, we obtain the following conclusion.

$$d_{\text{quant}} = \frac{\frac{1}{2} \log n + \log \frac{1}{q}}{\log 2} = O\left(\log \frac{1}{q}\right)$$

**Comparing FedAQ to FedAC** We compare computation and communication efficiency of FedAC-II and FedAQ under the condition set (eq. (2.2)) to achieve the same error. Let's recall the convergence rate of FedAC and FedAQ. The convergence rate of FedAC and FedAQ is respectively  $\tilde{O}\left(\frac{1}{MT} + \frac{1}{TK^3}\right)$  and  $\tilde{O}\left(\frac{1+q}{MT} + \frac{1+q}{TK^3}\right)$ . Let's say FedAC requires  $T$  iterations and  $K = M^{\frac{1}{3}}$  communication rounds to achieve the error  $\frac{1}{MT}$ . Then, FedAQ requires

$$T' = (1 + q)T, \quad K' = M^{\frac{1}{3}}$$

to achieve the same error  $\frac{1}{MT}$ . This means FedAQ needs  $1 + q$  times more local steps and the same number of communication rounds to achieve the same error of FedAC. These local steps do not require any communication with the server hence can be performed without any additional communication overhead.

From discussion in the previous section, if we use the simple low-precision quantizer, we need only  $d_{\text{quant}} = O\left(\log \frac{1}{q}\right)$  bits for communicating values with enough precision that can lead to an error rate of  $O\left(\frac{1}{MT}\right)$ . In comparison, FedAC would require  $O(\log(MT))$  bits to maintain enough precision to achieve the same error rate. In a majority of tasks in the real world, 32 bits are usually enough for  $d_{\text{full}}$  to achieve enough precision as we usually don't need converge to a very small error rate. Nonetheless, even if we compare FedAQ(8bits) with

to FedAC(32bits), we argue that the overall benefit from less communication by quantization is more influential than the slowdown effect from quantization.

For example, if we consider a  $l_2$ -regularized logistic regression model for MNIST (strongly convex experiment) and quantize from 32 bits to  $d_{\text{quant}} = 8$  bits. Here,  $n = 784 \times 10$ . We get the following results by using Equation 2.39.

$$1 + q = 1 + \frac{\sqrt{n}}{2^{d_{\text{quant}}}} = 1 + \frac{\sqrt{7840}}{2^8} \simeq 1.346,$$

On the other hand, the ratio of data communicated by FedAC and FedAQ is

$$\frac{32}{d_{\text{quant}}} = 4$$

In contribution item 2, we claim  $1 + q \ll \frac{d_{\text{full}}}{d_{\text{quant}}}$  because  $d_{\text{full}}$  is unbounded as  $T$  goes to infinity. In the real world example,  $\frac{d_{\text{full}}}{d_{\text{quant}}} = 4$  is still much greater than  $1 + q$ . Furthermore, since the local computation is much cheaper than data communication, we conclude that the benefit from less communication by quantization (4 times less bits) overwhelm the slowdown effect from quantization ( $(1 + q)$  times more local computation).

## 2.5 Experiments

In this section, we provide experimental results of FedAQ in homogeneous local data distribution settings. We compare FedAQ with other quantization-based federated optimization algorithms, FedPAQ [29] and FedCOMGATE [30]. FedAvg [1] and FedAC [31], federated optimization algorithms without quantization, are also our baselines. We empirically validate the performance of 5 algorithms on classical classification tasks on MNIST[63] and CIFAR-10[64] datasets in the distributed learning environment. We consider three objective functions i) A strongly convex objective of  $l_2$ -regularized logistic regression model on the MNIST dataset, ii) A non convex objective of training a multilayer perceptron on the MNIST data, and iii) A non convex objective of training a convolution neural network (CNN) on the

CIFAR-10 dataset.

### 2.5.1 Experimental Setup

**Implementation Environment.** We follow the implementation setup in [30]. We use the Distributed library of PyTorch to implement our algorithm because this library allows us to simulate real-world communication and distributed training. The 18 cores of Intel Xeon E5-2676 CPU are used as computing sources. Each core is considered as one local client. We use 16 cores for strongly convex MNIST, 18 cores for the non-convex MNIST, and 8 cores for the CIFAR-10. For MNIST, the strongly convex experiment and the non-convex one respectively run for 300 rounds of communication with 20 local updates and 50 rounds of communication with 100 local updates. The CIFAR-10 experiment runs for 100 rounds of communication with 100 local updates.

**Datasets.** For image classification tasks, we choose two main classical image datasets: MNIST and CIFAR-10. Since we assume homogeneous settings, data is distributed homogeneously among clients, which also means each device has access to all 10 classes.

**Hyperparameter Choice.** The important hyperparameters in our experiments are learning rates for each algorithm. For the client learning rate  $\eta$ , we respectively use 0.002, 0.1, and 0.01 for strongly convex MNIST, non-convex MNIST, and CIFAR-10 for all algorithms. For FedAQ and FedAC, once we set the value of  $\mu$ , other hyperparameters ( $\gamma, \alpha, \beta$ ) are automatically determined (See condition set (eq. (2.1)) and (eq. (2.2))). Thus, we choose 0.1, 0.01, and 0.2 for  $\mu$  value for strongly convex MNIST, non-convex MNIST, and CIFAR-10. Since too large  $\mu$  leads to slow convergence and too small  $\mu$  leads to unstable training, we get these  $\mu$  values by tuning  $\mu$  appropriately. FedCOMGATE has a server learning rate, and we set this value as 1 for all experiments.

**Quantization Bits.** We have three quantization-based federated algorithms: FedAQ, FedPAQ, FedCOMGATE. We quantize the updates from 32 bits to 8 bits for all quantization-based algorithms in both MNIST and CIFAR-10. Additionally, particularly for FedAQ in non-convex experiments, we consider 4 bits quantization as well. Since FedAQ sends twice as many messages as FedPAQ or FedCOMGATE at every synchronization when we use 8 bits quantization for all cases, we apply 4 bits quantization to FedAQ to let FedAQ send the same amount of information in each communication round as other quantization-based algorithms for a fair comparison.

**New Time Metric.** In our experiments, communication between CPU cores is very fast, so it is hard to say that the environment of our experiments fully reflects the real-world federated learning when there is a heavy communication burden. Thus, we use a linear model to estimate the execution time  $T_{\text{round}}(\mathcal{A})$  between two consecutive communication rounds for real federated learning scenarios [65].

$$T_{\text{round}}(\mathcal{A}) = T_{\text{comm}}(\mathcal{A}) + T_{\text{comp}}(\mathcal{A}), \quad T_{\text{comm}}(\mathcal{A}) = \frac{S_{\text{down}(\mathcal{A})}}{B_{\text{down}}} + \frac{S_{\text{up}(\mathcal{A})}}{B_{\text{up}}}$$

$$T_{\text{comp}}(\mathcal{A}) = \max_j T_{\text{client}}^j(\mathcal{A}) + T_{\text{server}}(\mathcal{A}), \quad T_{\text{client}}^j(\mathcal{A}) = R_{\text{comp}} T_{\text{sim}}^j(\mathcal{A}) + C_{\text{comp}}$$

Since  $T_{\text{server}}(\mathcal{A})$  is relatively smaller than  $T_{\text{client}}^j(\mathcal{A})$ , we ignore  $T_{\text{server}}(\mathcal{A})$  in our experiments. We get client download size  $S_{\text{down}(\mathcal{A})}$  and upload size  $S_{\text{up}(\mathcal{A})}$  from the number of neural network parameters.  $\max_j T_{\text{sim}}^j(\mathcal{A})$  is the computation time in our simulation.

$$B_{\text{down}} \sim 0.75\text{MB/secs}, \quad B_{\text{up}} \sim 0.25\text{B/secs}, \quad R_{\text{comp}} \sim 7, \quad C_{\text{comp}} \sim 10\text{secs}$$

[65] estimate each value of the above parameters from a real world cross-device FL system. The upload bandwidth  $B_{\text{up}}$  is generally smaller than download bandwidth  $B_{\text{down}}$ . We define human time as the parallel time estimated by this new time metric.

## Training Models

For MNIST, we use a  $l_2$ -regularized logistic regression model for the strongly convex case and a multilayer perceptron (MLP) with two hidden layers for the non-convex case. For CIFAR-10, we use a Convolutional Neural Network (CNN). Here, we note that the number of parameters in a neural network model is directly related to the number of communicated bits. We discuss more details as follows.

**MLP Model for MNIST.** We use a multilayer perceptron (MLP) with two hidden layers. Each hidden layer consists of 200 neurons with ReLU activations. Thus, we compute the total number of parameters in this MLP model as below.

$$\begin{aligned} (\# \text{ of MLP parameters}) &= (\# \text{ of input features}) \times (\# \text{ of neurons in the 1st layer}) \\ &+ (\# \text{ of neurons in the 1st layer}) \times (\# \text{ of neurons in the 2nd layer}) \\ &+ (\# \text{ of neurons in the 2nd layer}) \times (\# \text{ of MNIST classes}) \\ &+ (\# \text{ of neurons in the 1st layer}) + (\# \text{ of neurons in the 2nd layer}) \\ &+ (\# \text{ of MNIST classes}) \\ &= 28 \times 28 \times 200 + 200 \times 200 + 200 \times 10 + 200 + 200 + 10 = 199210 \end{aligned}$$

Finally, we derive  $S_{\text{up}}(\mathcal{A})(= S_{\text{down}}(\mathcal{A}))$ , defined in section 2.5.1 (New time metric), by using the above fact. We use 32 bits floating-point if there is no quantization.

$$\begin{aligned} S_{\text{up}}(\mathcal{A}) &= (\# \text{ of device}) \times (\# \text{ of MLP parameters}) \times (\# \text{ of bits}) \\ &= 18 \times 199210 \times 32 = 114744960 \end{aligned}$$

The FedAvg algorithm follows the above calculation. If we use 8 bits quantization for FedPAQ, FedCOMGATE, and FedAQ, ( $\#$  of bits) in the above equation will respectively be 8, 8, and 16. Since FedAQ sends twice as many messages as others at every communication

round, (# of bits) for FedAQ is 16. Similarly, (# of bits) for FedAC, which has no quantization, is 64.

**CNN Model for CIFAR-10.** We use a CNN model, which consists of two 2-dimensional convolutional layers, two max pooling layers, and two fully connected layers. The ReLU activations are used in this CNN model. Let's clarify (# of input channel, # of output channel, kernel size, stride) for convolutional layers. We respectively use (3, 20, 5, 1), (20, 50, 5, 1) for the 1st and 2nd convolutional layer. Let's denote each convolutional layer and fully connected layer as CONV1, CONV2, FC3, FC4. At first, the activation shape of input layer for CIFAR-10 is (32, 32, 3). Then, we get the activation shape after CONV1 and the number of parameters for CONV1.

$$\begin{aligned}
 (\text{width of activation shape}) &= \frac{(\text{width of previous activation shape}) - \text{kernel size} + 1}{\text{stride}} \\
 &= \frac{32 - 5 + 1}{1} = 28 \Rightarrow \text{activation shape} = (28, 28, 20) \\
 (\# \text{ of CONV1 parameters}) &= \left( \text{kernel size} \times \text{kernel size} \right. \\
 &\quad \times (\# \text{ of filters in the previous layer}) + 1 \left. \right) \\
 &\quad \times (\# \text{ of filters in the current layer}) = (5 \times 5 \times 3 + 1) \times 20 = 1520
 \end{aligned}$$

The activation shape becomes (14, 14, 20) after max pooling. There are no learnable parameters in pooling layers. We do similar calculation for CONV2.

$$\begin{aligned}
 (\text{width of activation shape}) &= \frac{(\text{width of previous activation shape}) - \text{kernel size} + 1}{\text{stride}} \\
 &= \frac{14 - 5 + 1}{1} = 10 \Rightarrow \text{activation shape} = (10, 10, 50) \\
 (\# \text{ of CONV2 parameters}) &= \left( \text{kernel size} \times \text{kernel size} \right. \\
 &\quad \times (\# \text{ of filters in the previous layer}) + 1 \left. \right) \times (\# \text{ of filters in the current layer}) \\
 &= (5 \times 5 \times 20 + 1) \times 50 = 25050
 \end{aligned}$$

The activation shape becomes (5, 5, 50) after second max pooling. Then, we calculate the number of parameters in FC3 and FC4 similar to the MLP case.

$$\begin{aligned} (\# \text{ of FC3 parameters} ) &= (5 \times 5 \times 50) \times 512 + 512 = 640512 \\ (\# \text{ of FC4 parameters} ) &= 512 \times 10 + 10 = 5130 \end{aligned}$$

Thus, the total number of parameters in this CNN model is

$$\begin{aligned} (\# \text{ of CNN parameters} ) &= (\# \text{ of CONV1 parameters} ) + (\# \text{ of CONV2 parameters} ) \\ &+ (\# \text{ of FC3 parameters} ) + (\# \text{ of FC4 parameters} ) \\ &= 1520 + 25050 + 640512 + 5130 = 672212 \end{aligned}$$

Finally, we derive  $S_{\text{up}}(\mathcal{A})(= S_{\text{down}}(\mathcal{A}))$  in this case.

$$\begin{aligned} S_{\text{up}}(\mathcal{A}) &= (\# \text{ of device} ) \times (\# \text{ of CNN parameters} ) \times (\# \text{ of bits} ) \\ &= 8 \times 672212 \times 32 = 172086272 \end{aligned}$$

We can do the similar discussion in the MLP case when it comes to applying this to quantization-based federated optimization algorithms.

### 2.5.2 Experimental Results

In our experiments on both MNIST and CIFAR-10, we verify how the global training loss and test accuracy of five algorithms change with respect to communication rounds, the number of bits communicated between one client and the server during the uplink, and human time. We provide both qualitative analysis and quantitative results for plots.

### *Qualitative Analysis*

**Strongly Convex Case.** In this experiment, we compare FedAQ under the condition set (eq. (2.1)) and set (eq. (2.2)) with FedAvg, FedPAQ, FedCOMGATE, and FedAC-I. We denote each FedAQ as FedAQ-I and FedAQ-II. As we observe the theoretical benefits of FedAQ over other methods in section 2.4, FedAQ-I outperforms all other quantization-based federated optimization algorithms and FedAC-I in all plots (See each first row of Figure 2.1, Figure 2.2). However, although FedAQ-II shows the fast convergence speed, the training process is unstable. Thus, we only use FedAQ-I for further non-convex experiments. FedAC and FedAQ in non-convex experiments indicate FedAC-I and FedAQ-I.

**Non-Convex Case.** Each second row of Figure 2.1, Figure 2.2, and Figure 2.3 clearly demonstrates that FedAQ with 4 bits quantization outperforms other algorithms in all plots. In terms of communication rounds, accelerated algorithms, FedAQ and FedAC, converge faster than other algorithms. We also observe that quantization does not lead to slower convergence, which means we can apply an efficient quantization scheme to make communication efficient FL systems without sacrificing convergence speed. The plots related to communicated bits are helpful to interpret how algorithms work well in situations with heavy communication. FedAQ with 8 bits quantization shows comparable performance relative to FedPAQ and FedCOMGATE with the help of acceleration, even though FedAQ sends more updates during every synchronization. When we use 4 bits quantization for FedAQ to make the number of communicated bits the same for all quantization-based algorithms during synchronization, FedAQ shows a much faster convergence speed with regard to the number of communicated bits. However, plots of communicated bits fail to reflect how algorithms converge in real estimated time for FL scenarios, which consists of both communication and computation. Thus, we further analyze algorithms with human time. We observe that FedAQ with 8 quantization bits performs slightly better than FedPAQ and FedCOMGATE for both MNIST and CIFAR-10. This occurs because while all quantization-

based algorithms send the same number of communicated bits, the number of communication rounds for FedAQ is much smaller than others. Then, this also indicates that FedAQ takes less computation time than other methods while reaching the same accuracy.

### *Quantitative Results*

We provide quantitative results to help readers understand plots better. To be specific, for all plots, we observe the number of communication rounds, the number of communicated bits, and the human time required to achieve a particular test accuracy by each federated optimization algorithm.

For the strongly convex experiment on MNIST (See the first row of Figure 2.2), the number of communication rounds required to achieve 90.28% test accuracy by FedAvg, FedPAQ(8bits), FedCOMGATE(8bits), FedAC-I, FedAQ-I(8bits), FedAQ-II(8bits) are respectively 217, 216, 260, 28, 26, 99. The number of communicated bits required to achieve the same accuracy are respectively  $5.4e7$ ,  $1.4e7$ ,  $1.6e7$ ,  $1.4e7$ ,  $3.3e6$ ,  $1.2e7$ . Lastly, the required human time are respectively 3220s, 2760s, 3336s, 484s, 344s, 1323s. In this experiment, FedAQ-I(8bits) requires the smallest number of communication rounds, the smallest number of communicated bits, and the shortest human time to achieve the same test accuracy. These experimental results support the validity of our theoretical analysis on strongly convex cases.

For the non-convex experiment on MNIST (See the second row of Figure 2.2), the number of communication rounds required to achieve 97.6% test accuracy by FedAvg, FedPAQ(8bits), FedCOMGATE(8bits), FedAC, FedAQ(8bits), FedAQ(4bits) are respectively 23, 48, 38, 18, 18, 16. The number of communicated bits required to achieve the same accuracy are respectively  $1.5e8$ ,  $7.6e7$ ,  $6.1e7$ ,  $2.3e8$ ,  $5.7e7$ ,  $2.5e7$ . Finally, the required human time are respectively 2424s, 2311s, 1834s, 3327s, 1248s, 805s. Thus, we conclude that FedAQ(4bits) outperforms other algorithms, and even FedAQ(8bits) needs smaller number of communicated bits/less human time to achieve the goal accuracy than

FedPAQ(8bits)/FedCOMGATE(8bits).

For the non-convex experiment on CIFAR-10 (See Figure 2.3), the number of communication rounds required to achieve 65.4% test accuracy by FedAvg, FedPAQ(8bits), FedCOMGATE(8bits), FedAC, FedAQ(8bits), FedAQ(4bits) are respectively 98, 89, 95, 49, 50, 48. The number of communicated bits required to achieve the same accuracy are respectively  $2.1e9$ ,  $4.8e8$ ,  $5.1e8$ ,  $2.1e9$ ,  $5.4e8$ ,  $2.6e8$ . Finally, the required human time are respectively 31798s, 11526s, 12240s, 28720s, 9902s, 6464s. As with the non-convex experiment on MNIST, FedAQ(4bits) outperforms other algorithms, and even FedAQ(8bits) requires less human time to achieve the same accuracy than FedPAQ(8bits)/FedCOMGATE(8bits).

*Remark 2.5.1.* Our current experimental setup only allows us to scale the number of clients up to the number of CPU cores in our machine. Since FedAQ achieves linear speed up in the number of workers with much fewer communication rounds than other quantization based methods, we expect FedAQ to outperform other methods by an even larger margin as we scale the number of workers.

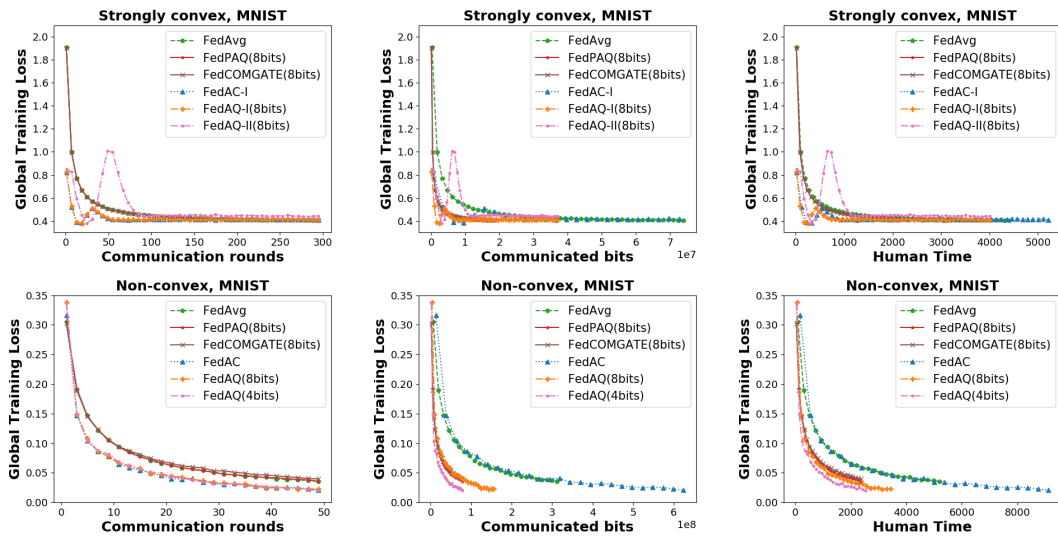


Figure 2.1: Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on MNIST with Strongly Convex Settings (first row) and Non-Convex Settings (second row). We observe how the global training loss changes across communication rounds (first column), communicated bits (second column), and human time (third column). FedAQ-I(8bits) and FedAQ(4bits) respectively outperform other algorithms for strongly convex settings and non-convex settings. FedAQ(4bits) sends the same number of communicated bits as FedPAQ(8bits) and FedCOMGATE(8bits) in each communication round, which indicates a fair comparison (See Quantization bits in section 2.5.1).

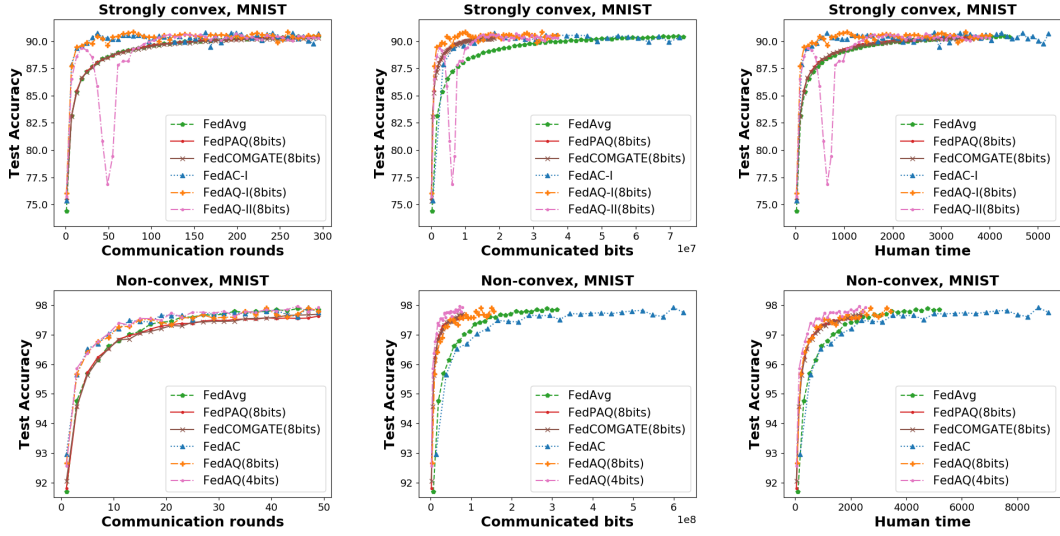


Figure 2.2: Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on MNIST with Strongly Convex Settings (first row) and Non-Convex Settings (second row). We observe how the test accuracy changes across communication rounds (first column), communicated bits (second column), and human time (third column). FedAQ-I outperforms other algorithms in all plots for strongly convex settings. Moreover, FedAQ(4bits) outperforms other algorithms in all plots for non-convex settings.

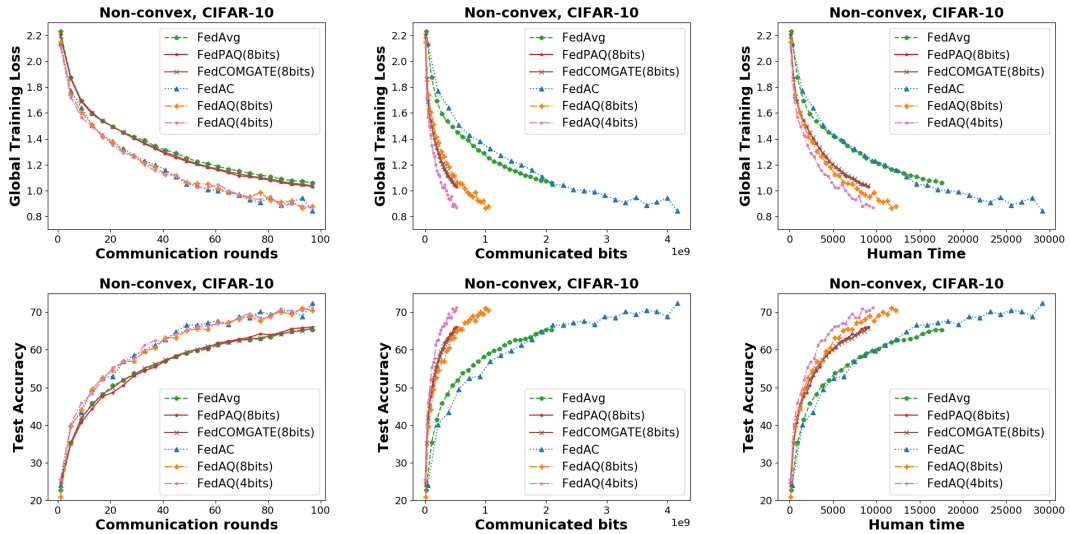


Figure 2.3: Comparing FedAQ with FedAvg, FedPAQ, FedCOMGATE, and FedAC on CIFAR-10. We observe how the global training loss and test accuracy change across communication rounds (first column), communicated bits (second column), and human time (third column). We use a CNN model for CIFAR-10. Similar to the MNIST experiment, FedAQ (4 bits) outperforms all other algorithms in every case.

## CHAPTER 3

### IMPROVING PRIVACY-ACCURACY TRADE-OFFS IN FEDERATED LEARNING VIA RANDOMIZED QUANTIZATION MECHANISM

#### 3.1 Introduction

In chapter 2, we provided the FedAQ algorithm by combining accelerated method of federated averaging with an efficient quantization scheme to solve the communication bottleneck problem in Federated Learning (FL). Besides communication efficiency, equally important in the context of FL is data privacy of each local device. At a basic level, privacy is maintained by storing local data on each end-user device without sharing it with the server. However, in some cases, device or local device data can be partially reconstructed from computed gradients [6]. This potential data leakage from gradients can be addressed through the use of privacy-preserving techniques. Building on the extensive communication demands in edge computing FL systems, as discussed in chapter 2, this chapter devises a training scheme that not only preserves privacy but also aligns with the requirements of efficient communication within FL systems. What is perhaps surprising, however, is that these two objectives are not necessarily in tension and *can even be aligned!* One way to improve communication overhead is to reduce bit complexity through stochastic rounding schemes, but we show that these randomization procedures, if designed carefully, provide additional benefits to data privacy.

Past studies, such as the work of [19, 20, 21], have sought to tackle this issue by employing various forms of discrete additive DP noise in conjunction with quantization; this is in part because quantization has immediate communication complexity benefits over continuous noise. However, when these discrete additive noise methods are coupled with secure aggregation protocols [22], aimed at preventing a server from inspecting individual

local device updates, they encounter a challenge of biased estimation due to modular clipping. To solve this, [23] introduce the Poisson Binomial Mechanism (PBM), bypassing the use of additive noise and instead directly mapping continuous inputs to discrete values in an unbiased fashion.

Losses in accuracy compared to noise-free gradient updates that do not protect privacy in the strong sense afforded by differential privacy are essentially unavoidable. The addition of a privacy requirement inevitably constraints the learner’s problem, and privacy must be traded-off with accuracy. Yet, the solution provided by [23], while providing good performance, may still enjoy sub-optimal privacy-accuracy trade-offs. Can we develop new mechanisms with improved privacy-accuracy trade-off compared to the mechanism of [23]?

Our starting point to address this question is to note that much research focusing on quantization in federated learning for the sake of communication complexity absent privacy [32, 29, 30, 66] reveal that performance degradation from quantization alone is somewhat minimal. Furthermore, quantization itself inherently reduces the amount of information encoded about the original input. While quantization in itself is insufficient for privacy, we posit that a two-stage approach, selecting a *randomized* quantization scheme followed by randomized rounding, can provide a viable approach to obtaining low communication complexity with formal *differential privacy* guarantees while still enjoying good performance. Thus:

*Can we harness randomization in quantization schemes to further improve privacy-accuracy trade-offs in differentially private federated learning?*

To address this question, we introduce what we call the *Randomized Quantization Mechanism*, or *RQM* for short. RQM achieves privacy entirely through randomly subsampling quantization levels followed by a (randomized) rounding procedure to a close-by quantization level.

**Summary of contributions** As mentioned above, our research studies mechanisms for releasing gradients while satisfying Rényi differential privacy, and how our proposed mech-

anisms can be integrated in standard federated learning frameworks.

- In section 3.4, we introduce our *Randomized Quantization Mechanism* that maps gradients to a randomized discrete grid in a way that preserves both standard and Rényi differential privacy.
- In section 3.4.3, we provide theoretical evidence that our proposed Randomized Quantization Mechanism exhibits  $\alpha$ -Rényi differential privacy guarantees “locally”, at the level of each single end-user device. Our theoretical guarantees hold for  $\alpha \rightarrow +\infty$ , implying in particular that they hold not just for Rényi but also for traditional  $(\epsilon, 0)$ -differential privacy. Also, we show that for any given  $\alpha$ , RQM provides lower Rényi divergence hence better Rényi DP guarantees than Poisson Binomial Mechanism (PBM) introduced by [23] via the numerical Rényi DP computation approach. Further, in section 3.4.4, we discuss the performance of RQM compared to PBM in terms of numerical privacy-accuracy trade-offs by using the mean-squared error (MSE) as an accuracy measure.
- In section 3.5, we provide federated learning experiments that highlight the performance of our mechanism. In particular, we show that RQM outperforms the state-of-the-art PBM in terms of privacy-accuracy trade-offs. We demonstrate that incorporating RQM into the standard differentially private federated learning framework results in higher model accuracy compared to PBM, while using the same hyperparameters that led to improved numerical Rényi DP for RQM in section 3.4.3. This indicates that the accuracy improvement is achieved without compromising privacy. To be specific, we show this by experimentally exploring possible values of the parameter  $\theta$  used in PBM, and showing that for each value of  $\theta$ , there exists an instantiation of the parameters of our algorithm, RQM, that leads to better accuracy and privacy guarantees *simultaneously*.

## 3.2 Preliminaries

### 3.2.1 Related work

Both communication complexity and privacy concerns have been driving forces behind the development of Federated Learning. Federated optimization often uses two types of privacy-preserving techniques hand-in-hand. One is secure multi-party computation, which protects the communication between local devices and the learner, preventing an attacker from intercepting messages sent between them [67, 22]. One is information-theoretic privacy guarantees such as differential privacy [68] that prevent inference of any given single local device’s data from observed summary outputs (such as local gradient updates or the learner’s model itself). For example, [69] and [70] add a calibrated amount of Gaussian noise to the average of clipped local device updates based on the FedAvg [1] algorithm.

In this work, we focus on providing robust Rényi differential privacy guarantees in federated optimization while maintaining high communication efficiency and good accuracy. Previous methods have often used an approach based on quantization followed by the addition of discrete noise to achieve both differential privacy guarantees and low communication efficiency. [19] introduces the first communication-efficient federated optimization algorithm with differential privacy by incorporating quantization with the binomial mechanism. [20] and [21] employ discrete Gaussian and Skellam mechanisms, respectively, in conjunction with quantization and secure aggregation for enhanced privacy. However, the above methods lead to biased estimation due to the necessity of modular clipping. To address this issue, [23] and [71] propose unbiased mechanisms with improved privacy-accuracy trade-offs. [23] encodes local devices’ gradients into a parameter of the binomial distribution, allowing their mechanism to generate a sample from this distribution without the need for additive discrete noise. In contrast, rather than using known privacy mechanisms, [71] introduces the *Minimum Variance Unbiased mechanism* (MVU) to enhance the privacy-utility trade-off by solving an optimization problem designed to minimize the output variance of the mechanism

while adhering to local differential privacy and unbiasedness constraints. Enhancing this model, [72] propose a more scalable MVU mechanism with better privacy-utility trade-off, achieved through a new interpolation procedure in the numerical design process. Despite their progress in improving the privacy-utility trade-off, these methods do not fully exploit the privacy advantages offered by randomized quantization itself.

Our research is not the first to leverage compression techniques to achieve both communication efficiency and provable privacy benefits without incorporating additive discrete noise [73, 74]. [73] assume a Gaussian input vector distribution for their sketching algorithms to ensure differential privacy guarantees, which might not be strictly necessary. [74] ultimately first quantize the gradients updates, then randomize the quantized gradients via differential private mechanisms such as randomized response or Rappor [75]. However, and to the best of our knowledge, our Randomized Quantization Mechanism is the first investigation that exclusively utilizes randomization of the quantization itself to attain improved Rényi DP guarantees within Federated Learning frameworks.

### 3.2.2 Differential privacy

The main privacy technique for our Randomized Quantization Mechanism is differential privacy, defined as below.

**Definition 3.2.1.** ((Approximate) Differential Privacy [7]) For  $\epsilon, \delta \geq 0$ , a randomized mechanism  $\mathcal{M} : \mathcal{D} \rightarrow \mathcal{R}$  satisfies  $(\epsilon, \delta)$ -differential privacy if for any neighbor dataset  $D, D' \in \mathcal{D}$  differing by the addition or removal of a single user’s records, it holds that

$$\Pr(\mathcal{M}(D) \in E) \leq e^\epsilon \cdot \Pr(\mathcal{M}(D') \in E) + \delta$$

for all events  $E \subset \mathcal{R}$ .

In this work, we also consider a variant of standard differential privacy called *Rényi Differential Privacy* (or *Rényi DP*), introduced in the seminal work of [76]. We develop

mechanisms that guarantee Rényi DP and by extension traditional DP. The use of Rényi DP allows for tight privacy accounting throughout the training iterations. Rényi differential privacy relies on first understanding the notion of *Rényi divergence*:

**Definition 3.2.2.** (Rényi Divergence [77]) Let  $P$  and  $Q$  be probability distributions defined over  $\mathcal{R}$ . The Rényi divergence of order  $\alpha > 1$  is defined as

$$D_\alpha(P||Q) := \frac{1}{\alpha - 1} \log \mathbb{E}_{x \sim Q} \left[ \left( \frac{P(x)}{Q(x)} \right)^\alpha \right].$$

Then, Rényi differential privacy is defined as follows:

**Definition 3.2.3.** (Rényi Differential Privacy [76]) A randomized mechanism  $\mathcal{M} : \mathcal{D} \rightarrow \mathcal{R}$  satisfies  $(\alpha, \epsilon)$ -Rényi differential privacy if for any neighbor dataset  $D, D' \in \mathcal{D}$  it holds that

$$D_\alpha(P_{\mathcal{M}(D)} || P_{\mathcal{M}(D')}) \leq \epsilon. \tag{3.1}$$

When  $\alpha \rightarrow \infty$ ,  $(\alpha, \epsilon)$ -Rényi DP in fact recovers standard  $(\epsilon, 0)$ -DP. However, Rényi DP provides a finer-grained definition of privacy in that its guarantees can be tailored to the specific value of  $\alpha$  and corresponding Rényi divergence that one considers. We now state a major property of the Rényi divergence that is useful to our theoretical analysis.

**Lemma 3.2.4.** (*Monotonicity*)  $D_\alpha$  is nondecreasing in  $\alpha$ . I.e.,  $D_\alpha(P||Q) \leq D_{\alpha'}(P||Q)$  for all  $1 \leq \alpha \leq \alpha' \leq \infty$ .

### 3.2.3 User-level privacy

In the context of federated learning, we employ differential privacy to mask the contribution of any individual local device, making it challenging for a potential adversary to discern whether a local device’s dataset was utilized in the training process. As such, we need to extend the traditional item-level definition of differential privacy (definition 3.2.1) by redefining what we mean by neighboring datasets. In this context, two datasets are

considered neighboring if one dataset can be created by changing any subset of data points of a single user from the other dataset. This user-level perspective is relatively standard and is the same as the one studied by [69] and [78].

**Definition 3.2.5.** (User-level DP [78]) For  $\epsilon, \delta \geq 0$ , a randomized mechanism  $\mathcal{M} : \mathcal{D} \rightarrow \mathcal{R}$  satisfies  $(\epsilon, \delta)$ -user level DP if for any neighbor dataset  $D, D' \in \mathcal{D}$  satisfying  $d_{\text{user}}(D, D') \leq 1$ , it holds that

$$\Pr(\mathcal{M}(D) \in E) \leq e^\epsilon \cdot \Pr(\mathcal{M}(D') \in E) + \delta$$

for all events  $E \subset \mathcal{R}$ , where  $d_{\text{user}}$  is defined with  $n$  users as

$$D = (D_1, \dots, D_n), \text{ where } D_i = \{z_{i,1}, \dots, z_{i,m_i}\} \rightarrow d_{\text{user}}(D, D') := \sum_{i=1}^n 1\{D_i \neq D'_i\}$$

### 3.3 Model

We consider a federated learning set-up comprised of three types of entities: there are  $n$  end-user devices, one secure aggregator called *SecAgg*, and one learner. The learner's goal is to learn a machine learning model, parameterized by a  $f$ -dimensional vector  $w \in \mathbb{R}^f$ , using the data on the devices through Stochastic Gradient Descent (SGD). However, the learner does not access the data from the devices directly, both for communication efficiency and privacy reasons. Rather, at each time step  $t$ :

1. Each end-user device  $i$  computes a coordinate wise L-inf clipped gradient  $g_t^i \in [-c, c]^f$  locally using the data on that device only. This gradient is then encoded into an integer  $z_t^i$ . This integer can be seen as the index of a discrete level in a discretization of the space of potential gradients.
2. The secure aggregator receives one message  $z_t^i$  from each device  $i$ , which encodes information about the gradient  $g_t^i$  computed by  $i$ . The aggregator aggregates them into

a single message  $z_t = \sum_i z_t^i$ .

3. The server decodes  $z_t$ , computes the corresponding gradient  $\hat{g}_t$ , and takes a gradient step  $w_{t+1} \leftarrow w_t - \eta \hat{g}_t$ .

The traditional approach to federated learning releases gradients exactly; this approach is, however, i) inefficient from a communication complexity perspective and ii) vulnerable when it comes to privacy. We address i) by discretizing (or “quantizing”) the space of possible values of the gradients to a grid of size  $m$  per coordinate of the gradient; in turn, we require only  $f \times \log m$  bits to represent a single update by a single device. Regarding ii), we note that it is well-understood that releasing exact gradients can lead to privacy violations in that the secure aggregator and the learner can recover information about device  $i$ ’s dataset  $D_i$  through the gradient itself. To address this issue, instead of releasing the gradient  $g_t^i$  directly, device  $i$  releases a noisy quantization level  $z_t^i = RQM(g_t^i)$ , where RQM is a Randomized Quantization Mechanism that must satisfy Rényi differential privacy. The entire setup is described formally in Algorithm 3.

What Algorithm 3 describes is essentially the well-known, generic *Differentially Private Stochastic Gradient Descent* approach to federated learning [69]. The focus and novelty of our work, however, come from the design of the private mechanism RQM itself. We propose our new mechanism in section 3.4, characterize its privacy guarantees in terms of standard and Rényi differential privacy theoretically in section 3.4.3 and empirically in the Numerical Privacy Guarantees part of section 3.4.3, and study the accuracy of the resulting model when using privately-released gradient updates to train it in section 3.5.

### 3.4 The Randomized Quantization Mechanism

In this section, we introduce our main building block for privacy in federated learning. This building block provides a mechanism for privately releasing a scalar aggregate statistic of a single user’s data in the form of a new algorithm called the *Randomized Quantization*

---

**Algorithm 3** Distributed DP-SGD with RQM

---

- 1: **Input:**  $N$  local devices, each local device dataset  $D_i \in \mathcal{D}$  ( $i = 1, \dots, N$ ), coordinate wise L-inf clipping threshold  $c$ , RQM parameters  $(\Delta, m, q)$ , server learning rate  $\eta$ , initial vector  $w_0$ , loss function  $f(w, D)$
  - 2: **for**  $t = 0, \dots, T - 1$  **do**
  - 3:   Server broadcasts  $w_t$  to  $n$  sampled local devices from total  $N$  local devices;
  - 4:   **for** each local device  $i$  in parallel **do**
  - 5:      $g_t^i \leftarrow \text{Clip}(\nabla f(w_t, D_i))$ ;
  - 6:      $z_t^i \leftarrow \text{RQM}(g_t^i)$ ;
  - 7:     send  $z_t^i$  to the secure aggregator SecAgg.
  - 8:   **end for**
  - 9:   SecAgg outputs  $z_t = \sum_{i=1}^n z_t^i$ ;
  - 10:   server decodes  $\hat{g}_t \leftarrow -(c + \Delta) + \frac{2z_t(c+\Delta)}{n(m-1)}$ ;
  - 11:   server finds  $w_{t+1} \leftarrow w_t - \eta \hat{g}_t$ .
  - 12: **end for**
- 

*Mechanism* (RQM). We remark that when dealing with  $f$ -dimensional vectors instead, we apply our Randomized Quantization Mechanism independently to each vector coordinate.

We first formally present our RQM mechanism, outlined in Algorithm 4. Since our mechanism relies on a discrete probability distribution to choose the quantization, we show how this probability distribution over the quantizations translates into a probability distribution over outcomes of our mechanism on any given input  $x$ ; this distribution over outcomes is crucial to characterize the level of privacy obtained by our mechanism. Then, we theoretically analyze the both standard and Rényi differential privacy guarantees of RQM by using this distribution over outcomes and empirically show that RQM achieves better numerical privacy guarantees than PBM. Finally, we demonstrate the superiority of RQM over PBM in terms of numerical privacy-accuracy trade-offs.

### 3.4.1 Randomized Quantization Mechanism

In this section, we assume that each user outputs a continuous scalar input  $x$  computed from their data; this can be viewed as the simplest case of local updates.

Our RQM algorithm is then comprised of three key components: (1) enlarging the output range beyond the input range and setting up evenly spaced quantization bins, (2) sub-

sampling realized quantization levels, and (3) performing a randomized rounding procedure on the *sub-sampled* (and only those) discrete levels to map a value  $x$  to a quantization level. Each of these steps is crucial in ensuring the Rényi DP guarantees of the RQM, as we describe below. Formally, in each step, we perform the following operations:

1. We establish the output range of our mechanism by first augmenting the size of the input range. We do so by adding  $\Delta$  to the upper bound  $c$  and subtracting  $\Delta$  from the lower bound  $-c$  on the input data. This augmentation of the range is necessary for privacy: if we use the same range for the output, the quantization output for the maximum input ( $x = c$ ) would subsequently always be  $c$  subsequently, leaking a lot of information about  $x$ . After this, we establish  $m$  initial, evenly spaced quantization levels ( $B(0), B(1), \dots, B(m - 1)$ ) within this output range, which will be potential outputs of our mechanism.
2. Instead of using the entire set of quantization levels from step (1), we randomly sub-sample feasible quantization levels. We do so by including each discrete level for quantization with a carefully chosen probability  $q$ . The randomization of the quantization levels is necessary for privacy; otherwise, a value of  $x$  would always map to the fixed set of two quantization levels deterministically. This immediately breaks differential privacy.
3. We perform quantization on the sub-sampled discrete levels (and these sub-sampled levels only), achieving both robust privacy and unbiased estimation. We identify the quantization bin that houses the input  $x$  and perform randomized rounding on  $x$  within this interval. The specific probabilities employed for randomized rounding can be reviewed in Algorithm 4.

The randomized quantization and rounding procedures described above are also illustrated later on in Figure 3.1a.

---

**Algorithm 4** Randomized Quantization Mechanism

---

- 1: **Input:**  $c > 0, x \in [-c, c]$ , extend the upper bound and lower bound by  $\Delta$ , the maximum number of quantization levels  $m$ , include a certain quantization level with probability  $q$
- 2: Set  $X^{\max}$ :  $X^{\max} = c + \Delta$ , max and min value of quantization levels is respectively  $X^{\max}, -X^{\max}$ .
- 3: Quantization bins:  $i = 0, 1, \dots, m - 1 \rightarrow B(i) = -X^{\max} + \frac{2iX^{\max}}{m-1}$ .
- 4: sub-sample feasible quantization levels:
- 5: Always include  $B(0), B(m - 1)$  &  $i = 1, 2, \dots, m - 2 \rightarrow$  include  $B(i)$  with probability  $q$ .
- 6: sub-sampled indices of quantization levels  $\rightarrow i_1(= 0), i_2, \dots, i_l(= m - 1)$
- 7: Quantization step:
- 8: Find  $i_{j^*}(i_1 \leq i_{j^*} \leq i_l)$  that satisfies  $x \in [B(i_{j^*}), B(i_{j^*+1})]$ .
- 9: Do randomized rounding on  $x$  in this interval.
- 10:

$$z = \begin{cases} i_{j^*+1}, & \text{with probability } \frac{x - B(i_{j^*})}{B(i_{j^*+1}) - B(i_{j^*})} \\ i_{j^*}, & \text{o/w} \end{cases}$$

11: **return**  $z$

---

A major desiderata of our algorithm, shared with the Poisson Binomial Mechanism, is that it is an unbiased estimator of  $x$ . Unbiasedness is a desirable property of SGD-based algorithms, as bias can cause issues with convergence, for example in traditional DP-SGD.

*Claim 3.4.1. (Unbiasedness of RQM)* Let  $x \in [-c, c]$ . Conditional on the realization of  $i_{j^*}$ , the quantization level at the output  $z$  of Algorithm 4 is unbiased. I.e.,

$$\mathbb{E}[B(z)|i_{j^*}] = x,$$

where the expectation is taken over the randomization in the rounding step in line 10 of Algorithm 4.

*Proof of claim 3.4.1* Since  $B(z)$  is unbiased conditional on the realization of  $i_{j^*}$ , it

follows immediately that Algorithm 4 itself is also unbiased. In more details, we have that

$$\begin{aligned}
& \mathbb{E}[B(z)|x \in [B(i_{j^*}), B(i_{j^*+1})]] \\
&= B(i_{j^*+1}) \times \frac{x - B(i_{j^*})}{B(i_{j^*+1}) - B(i_{j^*})} + B(i_{j^*}) \times \frac{B(i_{j^*+1}) - x}{B(i_{j^*+1}) - B(i_{j^*})} \\
& \hspace{15em} \text{(Line 10 of Algorithm 4)} \\
&= \frac{x B(i_{j^*+1}) - B(i_{j^*+1}) B(i_{j^*}) + B(i_{j^*+1}) B(i_{j^*}) - x B(i_{j^*})}{B(i_{j^*+1}) - B(i_{j^*})} \\
&= \frac{x B(i_{j^*+1}) - x B(i_{j^*})}{B(i_{j^*+1}) - B(i_{j^*})} \\
&= x.
\end{aligned}$$

We now provide a quick remark on the additional flexibility of parameter choices offered by our framework, RQM, over the Poisson Binomial Mechanism, and which will be crucial when it comes to improving privacy-accuracy trade-offs:

*Remark 3.4.2.* The hyperparameters within our RQM algorithm offer enhanced flexibility, allowing for a more nuanced hyperparameter optimization when compared to PBM. RQM has in fact three hyperparameters  $\Delta, q, m$ , while PBM has two hyperparameters  $\theta, m$  (See Algorithm 2 in [23]). At a fixed number of discrete levels  $m$ , i.e. at a fixed level of communication complexity, this allows us to search over a bigger space of output distributions of quantization levels than [23] through the choice of  $(q, \Delta)$ . In section 3.5, we show that this leads to RQM achieving better privacy-accuracy trade-offs than PBM.

### 3.4.2 Resulting discrete distribution of outcomes

Given an input  $x$  and parameters  $m, q, \Delta$ , we can compute the discrete probability distribution of outputs  $Q(x)$  of RQM over the set of potential quantization levels  $B(0), B(1), \dots, B(m-1)$ . This discrete probability distribution is given in lemma 3.4.3:

**Lemma 3.4.3.** *Let  $m \in \mathbb{N}$ , and  $q \in (0, 1)$  be parameters of Randomized Quantization Mechanism  $Q$ . Define evenly spaced  $m$  quantization levels  $B(0), \dots, B(m-1)$  as in*

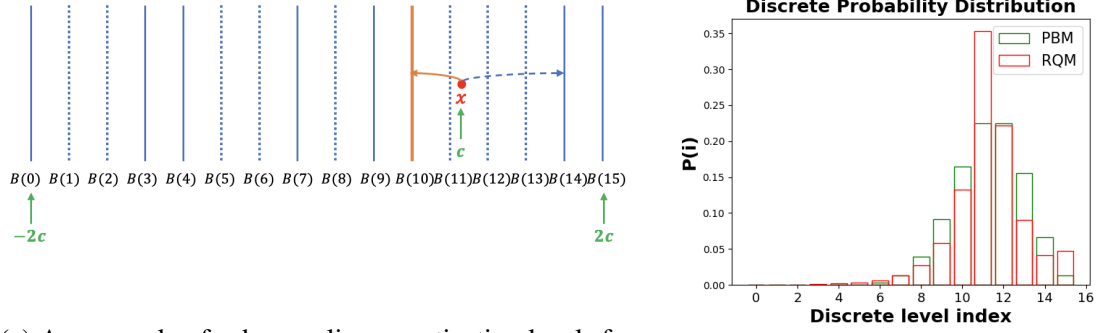
*Algorithm 4. Let  $j$  be the unique integer such that  $x \in [B(j), B(j + 1))$ . The probability distribution of outcomes of the Randomized Quantization Mechanism is given by:*

$$\Pr(Q(x) = i) = \begin{cases} (1 - q)^{j-i} \left( (1 - q)^{m-j-2} \frac{B(m-1)-x}{B(m-1)-B(i)} + \sum_{k=j+1}^{m-2} q(1 - q)^{k-j-1} \frac{B(k)-x}{B(k)-B(i)} \right), & i = 0, \\ q(1 - q)^{j-i} \left( (1 - q)^{m-j-2} \frac{B(m-1)-x}{B(m-1)-B(i)} + \sum_{k=j+1}^{m-2} q(1 - q)^{k-j-1} \frac{B(k)-x}{B(k)-B(i)} \right), & 0 < i \leq j, \\ q(1 - q)^{i-j-1} \left( (1 - q)^j \frac{x-B(0)}{B(i)-B(0)} + \sum_{k=1}^j q(1 - q)^{j-k} \frac{x-B(k)}{B(i)-B(k)} \right), & j + 1 \leq i < m - 1, \\ (1 - q)^{i-j-1} \left( (1 - q)^j \frac{x-B(0)}{B(i)-B(0)} + \sum_{k=1}^j q(1 - q)^{j-k} \frac{x-B(k)}{B(i)-B(k)} \right), & i = m - 1. \end{cases} \quad (3.2)$$

Equation 3.2 exhibits different cases for  $\Pr(Q(x) = i)$ . These cases depend on i) how the  $i$ -th quantization level compares to  $j$ , where  $j$  is defined to be such that  $x \in [B(j), B(j + 1))$  and ii) on the two special cases  $i = 0$  or  $i = m - 1$ . The probabilities corresponding to these two extreme values of  $i$  differ from the rest in that we always incorporate the 0-th and  $(m - 1)$ -th discrete level, which influences our probability calculations.

Figure 3.1a and Figure 3.1b provide some insights into how to derive the distribution of outputs and gives some intuition for lemma 3.4.3. In Figure 3.1a, the solid lines corresponding to the (0, 3, 4, 7, 9, 10, 14, 15)-th discrete levels have been selected for quantization, while the dotted lines (1, 2, 5, 6, 8, 11, 12, 13)-th discrete levels have been thrown away. To have  $Q(x) = 10$ , the 10-th discrete level must always be chosen while the 11-th discrete level must not be chosen by the sub-sampling step of our algorithm; the probability of this happening is  $q(1 - q)$ . The probability of the next quantization level bigger than  $x$  being the 14-th level, as shown in Figure 3.1a, is similarly given by  $q(1 - q)^2$  (levels 12 and 13 must not be sub-sampled, but 14 must be). Then, the likelihood of  $x$  transitioning to the 10-th discrete level due to randomized rounding between the 10-th and 14-th levels is  $\frac{B(14)-x}{B(14)-B(10)}$ . I.e., the situation described in Figure 3.1a happens with probability

$q^2(1 - q)^3 \frac{B(14)-x}{B(14)-B(10)}$ . For a complete analysis, we must also account for randomized rounding intervals  $[B(10), B(12)]$ ,  $[B(10), B(13)]$ ,  $[B(10), B(15)]$  and aggregate all these probabilities.



(a) An example of sub-sampling quantization levels for RQM.

(b) Distribution of outputs  $Q(x)$  under PBM and RQM.

Figure 3.1: An example of RQM with input  $x = c$  and parameters  $\Delta = c$ ,  $m = 16$ .

Figure 3.1b provides some insights on what the distribution induced by our quantization mechanism looks like, and seems to evidence that its shape differs from that of the Poisson Binomial Mechanism. Next, we present a complete proof of lemma 3.4.3.

*Proof of lemma 3.4.3* In lemma 3.4.3,  $j$  is defined as  $x \in [B(j), B(j + 1))$ . We divide the range of  $i$  into four cases- $0 < i \leq j$ ,  $i = 0$ ,  $j + 1 \leq i < m - 1$ ,  $i = m - 1$ - and compute the discrete probability  $\Pr(Q(x) = i)$  for each case. The core proof idea of lemma 3.4.3 is centered on evaluating the probability of each potential interval that can be used for randomized rounding for  $x$ . Subsequently, the probability that  $Q(x) = i$  arises due to randomized rounding within a given interval is computed. Thus, when  $i \leq j$  and  $k \geq j + 1$ , we define the event  $E_i$  and  $F_k$  as below to use this notation for calculating the probability of each potential interval that can be used for the randomized rounding step in Algorithm 4.

$E_i$  : the event of  $i$ -th discrete level being used for randomized rounding

$F_k$  : the event of  $k$ -th discrete level being used for randomized rounding (3.3)

From the above definition of two events,  $E_i \cap F_k$  indicates an event of the interval

$[B(i), B(k)]$  being used for randomized rounding. In this event, this also means  $i_{j^*} = i$  and  $i_{j^*+1} = k$  in Algorithm 4. Now, let's deep dive into how we can exactly calculate  $\Pr(Q(x) = i)$  for each case of four ranges.

(I)  $0 < i \leq j$ :

First, Let us consider the case when  $0 < i \leq j$ . Similar to the logic in section 3.4.2, to have  $Q(x) = i$ , the  $i$ -th discrete level must always be chosen while the  $(i + 1)$ -th,  $\dots$ ,  $j$ -th discrete levels must not be chosen by the sub-sampling step of our algorithm. The probability of this happening is  $q(1 - q)^{j-i}$ . Thus, we can use the definition of the event  $E_i$  in eq. (3.3) for this case.

$$\Pr(E_i) = \Pr(i : \text{chosen}, (i + 1, \dots, j) : \text{not chosen}) = q(1 - q)^{j-i} \quad (3.4)$$

Let us denote  $k$  as an index of the next quantization level bigger than  $x$ . The possible  $k$ s are  $j + 1, \dots, m - 1$ . When  $k \in [j + 1, m - 2]$ , the probability of the next quantization level bigger than  $x$  being the  $k$ -th level is similarly given by  $q(1 - q)^{k-j-1}$ . Thus, we can use the definition of the event  $F_k$  in eq. (3.3) for this case.

$$\Pr(F_k) = \Pr(k : \text{chosen}, (j + 1, \dots, k - 1) : \text{not chosen}) = q(1 - q)^{k-j-1} \quad (3.5)$$

Then, the likelihood of  $x$  transitioning to the  $i$ -th discrete level due to the randomized rounding between  $i$ -th and  $k$ -th levels is  $\frac{B(k)-x}{B(k)-B(i)}$ . This means

$$\Pr(Q(x) = i | E_i \cap F_k) = \frac{B(k) - x}{B(k) - B(i)} \quad (3.6)$$

Therefore, for  $k \in [j + 1, m - 2]$ , by combining Equation 3.4, Equation 3.5, Equation 3.6,

we get

$$\begin{aligned}
& \Pr((Q(x) = i) \cap E_i \cap F_k) \\
&= \Pr(E_i \cap F_k) \cdot \Pr(Q(x) = i | E_i \cap F_k) \\
&= \Pr(E_i) \cdot \Pr(F_k) \cdot \Pr(Q(x) = i | E_i \cap F_k) (\because E_i, F_k : \text{independent}) \\
&= q(1 - q)^{j-i} \cdot q(1 - q)^{k-j-1} \cdot \frac{B(k) - x}{B(k) - B(i)} \tag{3.7}
\end{aligned}$$

We can perform a similar computation for  $k = m - 1$ . However, the probability of event  $F_{m-1}$  is different from that of Equation 3.5 because the  $(m - 1)$ -th level is always chosen by Algorithm 4. Thus, we have

$$\Pr(F_{m-1}) = \Pr(m - 1 : \text{chosen}, (j + 1, \dots, m - 2) : \text{not chosen}) = (1 - q)^{m-j-2} \tag{3.8}$$

Therefore, for  $k = m - 1$ , by combining Equation 3.4, Equation 3.8, and Equation 3.6, we obtain

$$\begin{aligned}
& \Pr((Q(x) = i) \cap E_i \cap F_{m-1}) \\
&= \Pr(E_i \cap F_{m-1}) \cdot \Pr(Q(x) = i | E_i \cap F_{m-1}) \\
&= \Pr(E_i) \cdot \Pr(F_{m-1}) \cdot \Pr(Q(x) = i | E_i \cap F_{m-1}) \\
&= q(1 - q)^{j-i} \cdot (1 - q)^{m-j-2} \cdot \frac{B(m - 1) - x}{B(m - 1) - B(i)} \tag{3.9}
\end{aligned}$$

Finally, by combining Equation 3.7 and Equation 3.9, we get

$$\begin{aligned}
& \Pr(Q(x) = i) \\
&= \sum_{k=j+1}^{m-1} \Pr((Q(x) = i) \cap E_i \cap F_k) \\
&= q(1-q)^{j-i} \left( (1-q)^{m-j-2} \frac{B(m-1) - x}{B(m-1) - B(i)} + \sum_{k=j+1}^{m-2} q(1-q)^{k-j-1} \frac{B(k) - x}{B(k) - B(i)} \right)
\end{aligned} \tag{3.10}$$

(II)  $i = 0$ :

We can compute  $\Pr(Q(x) = 0)$  in a similar way as in case (I). However, the probability of event  $E_0$  is different from that of Equation 3.4 because the 0-th level is always chosen by Algorithm 4. Thus, we have

$$\Pr(E_0) = \Pr(0 : \text{chosen}, (1, \dots, j) : \text{not chosen}) = (1-q)^j \tag{3.11}$$

Therefore, in Equation 3.10, by substituting  $E_i$  into  $E_0$ , we get

$$\begin{aligned}
& \Pr(Q(x) = 0) \\
&= \sum_{k=j+1}^{m-1} \Pr((Q(x) = 0) \cap E_0 \cap F_k) \\
&= (1-q)^j \left( (1-q)^{m-j-2} \frac{B(m-1) - x}{B(m-1) - B(0)} + \sum_{k=j+1}^{m-2} q(1-q)^{k-j-1} \frac{B(k) - x}{B(k) - B(0)} \right)
\end{aligned} \tag{3.12}$$

(III)  $j + 1 \leq i < m - 1$ :

For  $i$  within this range, we can similarly compute  $\Pr(Q(x) = i)$  as in (I). To obtain  $Q(x) = i$ , the  $i$ -th discrete level must always be chosen while the  $(j + 1)$ -th,  $\dots$ ,  $(i - 1)$ -th discrete levels must not be chosen by the sub-sampling step of our algorithm. Thus, since  $i \geq j + 1$ , the probability of  $i$ -th discrete level being used for randomized rounding can be

expressed by using  $F_i$  (refer to Equation 3.5).

$$\Pr(F_i) = \Pr(i : \text{chosen}, (j + 1, \dots, i - 1) : \text{not chosen}) = q(1 - q)^{i-j-1} \quad (3.13)$$

Let us denote  $k$  as an index of the just previous level less than  $x$ . The possible  $k$ 's are  $0, \dots, j$ . Then, for  $k \in [1, j]$ , the probability of the  $k$ -th discrete level being used for randomized rounding can be represented by utilizing  $E_k$  (refer to Equation 3.4).

$$\Pr(E_k) = \Pr(k : \text{chosen}, (k + 1, \dots, j) : \text{not chosen}) = q(1 - q)^{j-k} \quad (3.14)$$

Then, the likelihood of  $x$  transitioning to the  $i$ -th discrete level due to the randomized rounding between  $k$ -th and  $i$ -th levels is  $\frac{x-B(k)}{B(i)-B(k)}$ . This means

$$\Pr(Q(x) = i | E_k \cap F_i) = \frac{x - B(k)}{B(i) - B(k)} \quad (3.15)$$

Therefore, for  $k \in [1, j]$ , by combining Equation 3.13, Equation 3.14, Equation 3.15, we get

$$\begin{aligned} & \Pr((Q(x) = i) \cap E_k \cap F_i) \\ &= \Pr(F_i) \cdot \Pr(E_k) \cdot \Pr(Q(x) = i | E_k \cap F_i) \\ &= q(1 - q)^{i-j-1} \cdot q(1 - q)^{j-k} \cdot \frac{x - B(k)}{B(i) - B(k)} \end{aligned} \quad (3.16)$$

We can similarly calculate for  $k = 0$  by using Equation 3.11.

$$\begin{aligned} & \Pr((Q(x) = i) \cap E_0 \cap F_i) \\ &= \Pr(F_i) \cdot \Pr(E_0) \cdot \Pr(Q(x) = i | E_0 \cap F_i) \\ &= q(1 - q)^{i-j-1} \cdot (1 - q)^j \cdot \frac{x - B(0)}{B(i) - B(0)} \end{aligned} \quad (3.17)$$

Finally, by combining Equation 3.16 and Equation 3.17

$$\begin{aligned}
& \Pr(Q(x) = i) \\
&= \sum_{k=0}^j \Pr((Q(x) = i) \cap E_k \cap F_i) \\
&= q(1-q)^{i-j-1} \left( (1-q)^j \frac{x-B(0)}{B(i)-B(0)} + \sum_{k=1}^j q(1-q)^{j-k} \frac{x-B(k)}{B(i)-B(k)} \right) \quad (3.18)
\end{aligned}$$

(IV)  $i = m - 1$ :

We can calculate  $\Pr(Q(x) = m - 1)$  in a similar way compared to case (III). However, the  $(m - 1)$ -th level should be always chosen by Algorithm 4, we rely on Equation 3.8 rather than Equation 3.13. We obtain:

$$\begin{aligned}
& \Pr(Q(x) = m - 1) \\
&= \sum_{k=0}^j \Pr((Q(x) = m - 1) \cap E_k \cap F_{m-1}) \\
&= (1-q)^{m-j-2} \left( (1-q)^j \frac{x-B(0)}{B(m-1)-B(0)} + \sum_{k=1}^j q(1-q)^{j-k} \frac{x-B(k)}{B(m-1)-B(k)} \right) \quad (3.19)
\end{aligned}$$

Therefore, we finally get Equation 3.2 of lemma 3.4.3 from combining cases (I), (II), (III), and (IV).

### 3.4.3 Analysis of RQM's privacy guarantees

We now provide a theoretical analysis of the level of standard differential privacy achieved by our single-dimensional RQM mechanism.

**Theorem 3.4.4.** *(Standard DP) Let  $c, \Delta > 0$ ,  $m \in \mathbb{N}$ , and  $q \in (0, 1)$  be parameters of Algorithm 4. Consider two scalars  $x$  and  $x'$  in  $[-c, c]$ ,  $P_{Q(x)}$  the distribution of outputs of RQM ran on scalar  $x$ , and  $P_{Q(x')}$  the distribution of outputs of RQM ran on scalar  $x'$ . We*

have:

$$D_\infty(P_{Q(x)}||P_{Q(x')}) \leq \log \left( 2(1-q)^2 \left( 1 + \frac{c}{\Delta} \right) \right) + m \log \frac{1}{1-q} = \epsilon. \quad (3.20)$$

which indicates RQM is  $(\epsilon, 0)$  differentially private.

We show that RQM is  $(\epsilon, 0)$  differentially private by leveraging the equivalence between  $(\epsilon, 0)$ -differential privacy and  $(\infty, \epsilon)$ -Rényi differential privacy. There, we note that the privacy level  $\epsilon = \log \left( 2(1-q)^2 \left( 1 + \frac{c}{\Delta} \right) \right) + m \log \frac{1}{1-q}$  that we obtain increases linearly on  $m$ , the number of quantizations level. This makes sense as a large number of quantization levels allows one to encode more information about the initial scalar  $x$ , in turn leading to less privacy and higher  $\epsilon$ 's. We also note that as  $\Delta$  increases,  $\epsilon$  decreases, and we obtain more privacy; once again, this follows the intuition from section 3.4.1 that when we increase the output range, we better protect the privacy of extreme values of  $x$  that are close to  $c$  or  $-c$ . As expected, when  $\Delta = 0$ ,  $\epsilon \rightarrow +\infty$  and our privacy guarantees are trivial, highlighting the fact that augmenting the range of output values beyond  $[-c, c]$  is an unavoidable step to obtain reasonable privacy guarantees. The full proof of theorem 3.4.4 is provided in the following.

*Proof of theorem 3.4.4* Let's find an upper bound on  $D_\infty(P_{Q(x)}||P_{Q(x')})$ .

$$\begin{aligned} D_\infty(P_{Q(x)}||P_{Q(x')}) &= \sup_{i \in \{0, 1, \dots, m-1\}} \log \left( \frac{\Pr(Q(x) = i)}{\Pr(Q(x') = i)} \right) \\ &= \max \left( \sup_{i \in \{0, m-1\}} \log \left( \frac{\Pr(Q(x) = i)}{\Pr(Q(x') = i)} \right), \sup_{i \in \{1, \dots, m-2\}} \log \left( \frac{\Pr(Q(x) = i)}{\Pr(Q(x') = i)} \right) \right) \\ &\leq \max \left( \log \left( \frac{1}{\min_{i \in \{0, m-1\}} \Pr(Q(x') = i)} \right), \log \left( \frac{q}{\min_{i \in \{1, \dots, m-2\}} \Pr(Q(x') = i)} \right) \right) \end{aligned}$$

The second inequality comes from  $\Pr(Q(x) = i) \leq 1$  for any  $i$  and  $\Pr(Q(x) = i) \leq q$  for  $i \in \{1, 2, \dots, m-2\}$ .  $\Pr(Q(x) = i)$  is less than or equal to  $q$  for  $i \in \{1, 2, \dots, m-2\}$

because

$$\begin{aligned}
\Pr(Q(x) = i) &= \Pr(Q(x) = i | i : \text{chosen}) \Pr(i : \text{chosen}) \\
&= \Pr(Q(x) = i | i : \text{chosen}) \times q \\
&\leq q(\Pr(Q(x) = i | i : \text{chosen}) + \Pr(Q(x) \neq i | i : \text{chosen})) = q
\end{aligned}$$

We establish a value for  $j$  that makes it so that  $-c$  falls within the range of values between  $B(j)$  and  $B(j+1)$ . Since  $\min_{i \in \{0, m-1\}} \Pr(Q(x') = i) \geq \Pr(Q(-c) = m-1)$  and  $\min_{i \in \{1, \dots, m-2\}} \Pr(Q(x') = i) \geq \Pr(Q(-c) = m-2)$ , we obtain

$$\begin{aligned}
D_\infty(P_{Q(x)} || P_{Q(x')}) &\leq \max \left( \log \left( \frac{1}{\min_{i \in \{0, m-1\}} \Pr(Q(x') = i)} \right), \log \left( \frac{q}{\min_{i \in \{1, \dots, m-2\}} \Pr(Q(x') = i)} \right) \right) \\
&\leq \max \left( \log \left( \frac{1}{\Pr(Q(-c) = m-1)} \right), \log \left( \frac{q}{\Pr(Q(-c) = m-2)} \right) \right) \\
&= \max \left( \log \left( \frac{1}{(1-q)^{m-2} \cdot \frac{-c-B(0)}{B(m-1)-B(0)} + \sum_{k=1}^j q(1-q)^{m-2-k} \cdot \frac{-c-B(k)}{B(m-1)-B(k)}} \right), \right. \\
&\quad \left. \log \left( \frac{q}{q \left( (1-q)^{m-3} \frac{-c-B(0)}{B(m-2)-B(0)} + \sum_{k=1}^j q(1-q)^{m-3-k} \frac{-c-B(k)}{B(m-2)-B(k)} \right)} \right) \right) \\
&\leq \log \left( \frac{1}{(1-q)^{m-2} \cdot \frac{-c-B(0)}{B(m-1)-B(0)}} \right) \\
&= \log \frac{1}{(1-q)^{m-2} \cdot \frac{\Delta}{2c+2\Delta}} \\
&= \log \left( \frac{2(1-q)^2(c+\Delta)}{\Delta} \right) + m \log \frac{1}{1-q}
\end{aligned}$$

To go from the third to the fourth and fifth line, we used lemma 3.4.3.

**Corollary 3.4.5.** *The upper bound in theorem 3.4.4 immediately applies to Rényi differential*

privacy. For any  $\alpha$ , RQM is  $(\alpha, \epsilon)$  Rényi differentially private:

$$D_\alpha(P_{Q(x)}||P_{Q(x')}) \leq D_\infty(P_{Q(x)}||P_{Q(x')}) \leq \log \left( 2(1 - q)^2 \left( 1 + \frac{c}{\Delta} \right) \right) + m \log \frac{1}{1 - q} = \epsilon. \quad (3.21)$$

We obtain corollary 3.4.5 by using lemma 3.2.4.

*Remark 3.4.6.* In corollary 3.4.5, we show the theoretical Rényi DP guarantees of RQM for the single dimensional case. For the multi dimensional case, the Rényi DP upper bound of corollary 3.4.5 grows linearly in the dimension  $f$  due to the composition theorems for Rényi DP [76].

**Numerical Privacy Guarantees** In theorem 3.4.4, we characterized the privacy guarantee of our Randomized Quantization Mechanism in the special case in which  $\alpha \rightarrow +\infty$ . We consider a *local* differential privacy benchmark, evaluating privacy against a strong adversary that can see the output  $Q(x_i)$  of *each* device  $i$  (but not the input data  $x_i$ )<sup>1</sup>. To demonstrate the superior Rényi DP guarantees of RQM over PBM for arbitrary  $\alpha$ , we use Equation 3.2 in lemma 3.4.3 to numerically compute and plot the upper bound of Rényi divergence  $D_\alpha(P_{Q(x)}||P_{Q(x')})$  for finite  $\alpha$  in Figure 3.2. We compare it to the Rényi divergence of the Poisson Binomial Mechanism of [23]; we note that we do not compare to the upper bound provided by [23] as the privacy measure that may not be tight, but instead to the actual Rényi divergence computed numerically and *exactly*. In both cases, we plot the nearly *worst-case* (over  $x, x'$ ) Rényi divergence, which is approximately maximized when  $x = c$  and  $x' = -c^2$ .

We fix the number of discrete levels  $m$  as 16 for both RQM and PBM to compare privacy guarantees between the two algorithms *at equal communication complexity*. We set the value  $c$  to be  $1.5^3$ . We provide the numerical Rényi DP experiments comparing our results to

<sup>1</sup>This is in contrast with central privacy, where the adversary can only see the aggregated gradients. The central privacy model requires trust in the aggregator, while the strong local privacy model does not.

<sup>2</sup>More details on this are discussed in section A.1.1.

<sup>3</sup>Our mechanism is in fact scale-invariant for DP guarantees, and the choice of  $c$  itself does not matter at a

PBM for a wide range of  $\theta \in [0, 0.4]$  values considered by [23]. Specifically, we select three different  $\theta$  values: 0.15, 0.25, and 0.35. For each  $\theta$ , the corresponding parameter pairs  $(\Delta, q)$  for RQM are  $(2.33c, 0.42)$ ,  $(c, 0.42)$ , and  $(0.429c, 0.49)$ , respectively. Figure 3.2 compares the Rényi divergence of PBM and RQM for a large range of  $\alpha \in [0, 1000]$ ; we see significant disparities in the levels of Rényi privacy guaranteed by PBM and RQM for all different  $\theta$ s, with RQM vastly outperforming (i.e., guaranteeing a lower Rényi divergence hence a better privacy guarantee than) PBM, with the gap in privacy guarantees increasing as  $\alpha \rightarrow +\infty$ .

We have demonstrated that for typical ranges of parameters for PBM, we can find an implementation of our framework, RQM, that leads to significantly enhanced privacy. However, we note that getting better privacy on its own is easy (one could for example simply not use the data, and get perfect privacy at the cost of accuracy). In section 3.4.4, we demonstrate that this improvement in privacy does not come at a cost in accuracy at the level of each single device, i.e. for  $n = 1$ . Then, in section 3.5, we run larger scale federated learning experiments under multiple devices that further highlight that this improvement in privacy does not come at a cost in accuracy. Our experiments demonstrate that the same ranges of parameters also show accuracy improvements for our RQM method over PBM in a large-scale federated learning experiment on the EMNIST dataset.

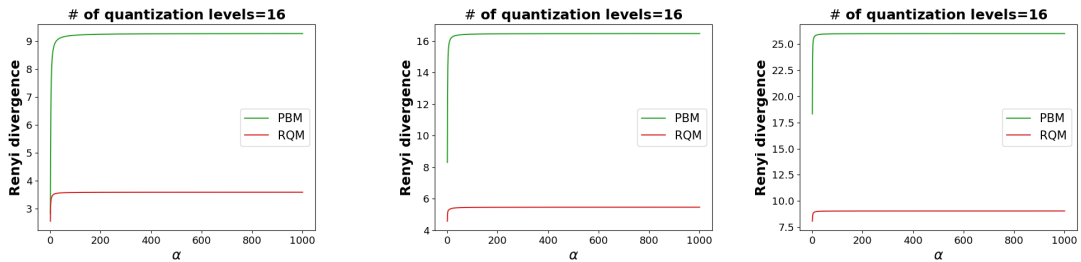


Figure 3.2: The results about Numerical Rényi privacy with  $\theta = 0.15$  (left),  $\theta = 0.25$  (middle), and  $\theta = 0.35$  (right). All figures indicate how the Rényi divergence increases as  $\alpha$  increases.

---

given constant ratio between  $\Delta$  and  $c$ .

### 3.4.4 Privacy-Accuracy Trade-offs at the User Device Level

We present a numerical analysis of the privacy-accuracy trade-offs of our framework, highlighting the improved performance of RQM compared to PBM, in Figure 3.3.

To compute the privacy guarantee we achieve for a given set of parameters, we follow the numerical Rényi DP computation approach elaborated in section 3.4.3.

Our accuracy measure is the mean-squared error (MSE). The MSE is calculated as the average MSE over 30 equally spaced scalar input values  $x$  within the range  $[-c, c]$ ; the MSE for each value  $x$  is based on 100,000 samples to deal with privacy noise. In each plot in Figure 3.3, we fix  $\alpha (= 1.5)$ ,  $c (= 1.5)$ , and  $m$  (variable), ensuring that the same values are used for both PBM and RQM. Three privacy-MSE plots are shown in Figure 3.3, each corresponding to different values of  $m$  (4, 16, and 64, mimicking [23]).

Finally, to generate the privacy-MSE curves, we sweep through  $\theta$  for PBM and  $q$  for RQM, while keeping  $\Delta = c$  constant for RQM; different values of  $\theta$  for PBM and of  $q$  for RQM correspond to different points on the curve. Across all plots in Figure 3.3, we consistently observe better privacy-MSE trade-offs with RQM compared to PBM. More numerical results with different  $\alpha$  are provided in section A.1.3.

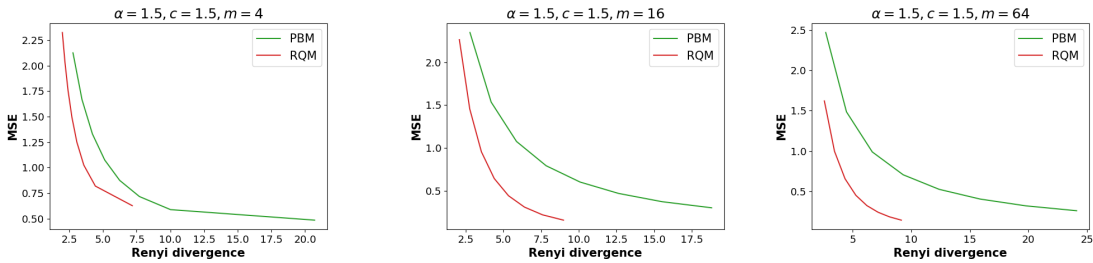


Figure 3.3: Numerical privacy-MSE trade-offs of RQM and PBM with  $m = 4$  (left),  $m = 16$  (middle), and  $m = 64$  (right).

## 3.5 Federated Learning Experiments

In this section, we conduct experiments designed to complement our theoretical results and illustrate how RQM performs compared to PBM in terms of the privacy-accuracy

trade-off on a larger scale Federated Learning experiment. We expand our experiments to provide insights beyond the privacy guarantees of the Randomized Quantization Mechanism itself, and to take into account how RQM integrates with the rest of the federated learning framework described in section 3.3. We implement multi-dimensional RQM within the federated DP-SGD algorithm described in Algorithm 3. We employ the same parameters that yielded improved Rényi DP in the Numerical Privacy Guarantees part of section 3.4.3 to demonstrate that RQM also excels in accuracy in our federated learning experiment.

We evaluate the privacy-accuracy trade-off of our algorithms against the previous leading approach, the Poisson Binomial Mechanism [23], and an ideal noise-free clipped SGD benchmark that does not provide any differential privacy guarantee. The classification task for our federated learning experiment is performed on the EMNIST dataset [79].

**Implementation environment.** We adopt the same implementation setup as outlined in [23]. To implement our algorithm, we utilize TensorFlow [80] and the TensorFlow Federated (TFF) library. Our computational resources include 2 NVIDIA RTX A5000 GPUs. We simulate a federated learning scenario involving a total of 3,400 local devices, with  $n = 40$  local devices participating in each round. The total number of communication rounds is set to 2,000.

**Dataset & training model.** We perform image classification on the EMNIST dataset, which is comprised of 62 classes. We employ a Convolutional Neural Network (CNN) as the learning model for our training purposes.

**Hyperparameter choice.** We adhere to the same hyperparameters for our FL experiments as those of the Numerical Privacy Guarantees part in section 3.4.3:  $m = 16$ ,  $\theta = 0.15, 0.25, 0.35$  for PBM,  $(\Delta, q) = (2.33c, 0.42), (c, 0.42), (0.429c, 0.49)$  for RQM. To highlight the flexibility of the choice of hyperparameters for RQM (remark 3.4.2), we also plot results of two more  $(\Delta, q)$  pairs for each  $\theta$ . For  $\theta = 0.15$ , we add two more

pairs  $(\Delta, q) = (4c, 0.5)$  and  $(\Delta, q) = (c, 0.23)$ ; for  $\theta = 0.25$ , we add  $(\Delta, q) = (2c, 0.57)$  and  $(\Delta, q) = (0.66c, 0.33)$ ; and for  $\theta = 0.35$ , we include  $(\Delta, q) = (c, 0.65)$  and  $(\Delta, q) = (0.25c, 0.37)$ . For clipping threshold  $c$ , we choose  $2.9731 \times 10^{-5}$ .

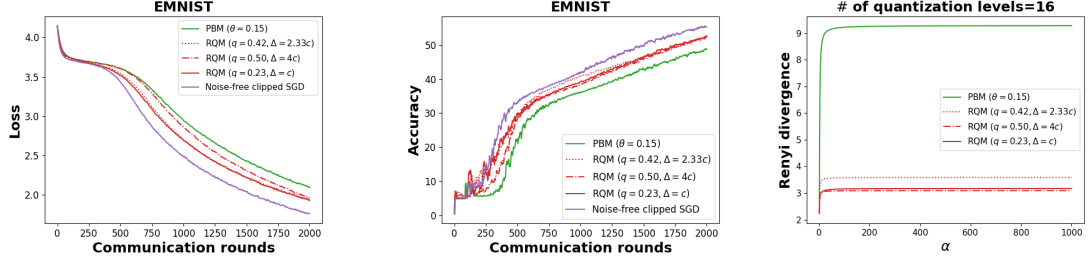


Figure 3.4: Comparing RQM with PBM ( $\theta = 0.15$ ) and noise-free clipped SGD on EMNIST. All three RQMs with different hyperparameters outperform PBM in both a loss plot (Left) and an accuracy plot (Middle). These RQMs also show better Rényi DP guarantees than PBM (Right).

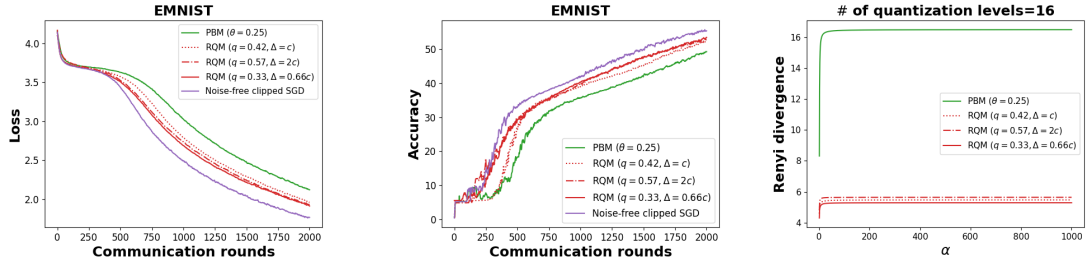


Figure 3.5: Comparing RQM with PBM and noise-free clipped SGD on EMNIST (Additional FL experiment with  $\theta = 0.25$ ).

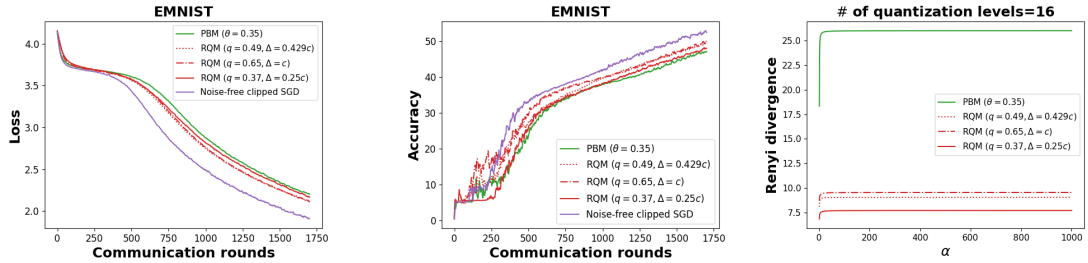


Figure 3.6: Comparing RQM with PBM and noise-free clipped SGD on EMNIST (Additional FL experiment with  $\theta = 0.35$ ).

**Experimental results** The left and middle plots in Figure 3.4, where  $\theta = 0.15$ , clearly demonstrate that all three RQMs with different hyperparameter pairs show improved perfor-

mance (in terms of loss and accuracy) on the EMNIST dataset than PBM. All three RQMs achieve similar accuracy. The performance of our three RQMs are still worse than noise-free clipped SGD: this is unavoidable because noise-free clipped SGD only focus on accuracy without providing any privacy guarantees, and is an ideal, impossible-to-achieve benchmark with privacy. Similarly, Figure 3.5 and Figure 3.6, corresponding to  $\theta$  values of 0.25 and 0.35 respectively, also show that all three RQMs outperform PBM in terms of performance. In Figure 3.5, among the three RQMs, the pair  $(\Delta, q) = (0.66c, 0.33)$  achieves the highest accuracy. In Figure 3.6, the pair  $(\Delta, q) = (c, 0.65)$  achieves the highest accuracy among the three RQMs.

The right plot in Figure 3.4<sup>4</sup> replicates experiment of the Numerical Privacy Guarantees part in section 3.4.3 that were aimed at showcasing the privacy level achieved by RQM compared to PBM. The figure shows that the improved accuracy of the three RQMs compared to PBM in the left and middle figures does not come at the cost of privacy. In fact, the three plots together demonstrate that all three instantiations of RQM provide both better performance and better Rényi DP guarantees than PBM. I.e., in our experiments, RQM improves the *privacy-accuracy trade-off* of federated differentially private stochastic gradient descent compared to the previous state of the art.

**More experimental results on CIFAR-100** In Figure 3.7, we provide additional experimental results on CIFAR-100 [64]. We use  $m = 16$ ,  $\theta = 0.25$  for PBM and  $\Delta = c$ ,  $q = 0.42$  for RQM (same hyperparameters as those for EMNIST in section 3.5), and clipping threshold  $c = 5.3680 \times 10^{-5}$ . We add the centralized continuous Gaussian mechanism and noise-free no-clipped SGD as the baselines. In Figure 3.7, we provide the evidence that our RQM achieves better performance in both loss and accuracy compared to the other methods including PBM, except for noise-free no-clipped SGD. Since we use the same hyperparameters for PBM and RQM as those in Figure 3.5 (dotted red line for RQM), the privacy

---

<sup>4</sup>The same conclusion can be drawn from Figure 3.5 and Figure 3.6, but we highlight Figure 3.4 as a representative figure.

level achieved by RQM is better than PBM. Thus, RQM shows improved privacy-accuracy trade-off compared to PBM for CIFAR-100 as well.

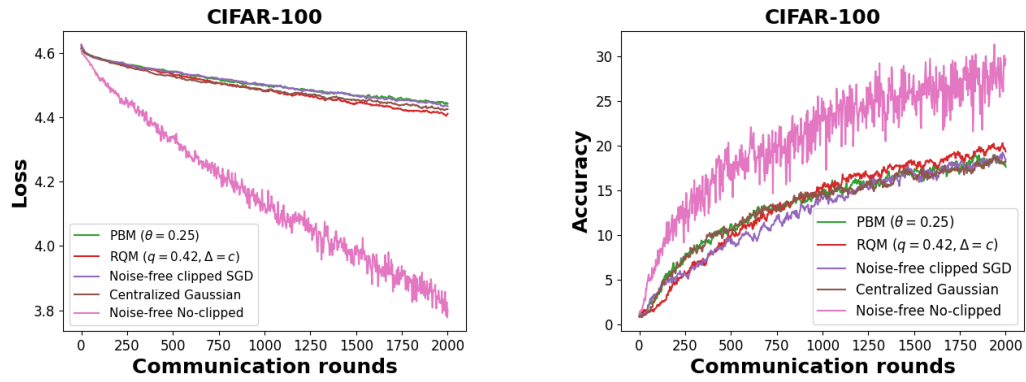


Figure 3.7: Comparing RQM with PBM, noise-free clipped SGD, centralized continuous Gaussian mechanism, and noise-free no-clipped SGD on CIFAR-100.

## CHAPTER 4

### QUANTIZATION GROUP ADAPTIVE NORMALFLOAT FOR LOW BIT FINE-TUNING OF LLMS

#### 4.1 Introduction

In chapter 2 and chapter 3, we explored how quantization improves communication efficiency and privacy-accuracy trade-offs in Federated Learning. Similarly, in the context of Large Language Models (LLMs), quantization plays a critical role in enabling memory-efficient fine-tuning, especially in resource-constrained environments. This chapter focuses on the design of a quantization scheme optimized for extremely low-bit fine-tuning, specifically exploring how 2-bit quantization can be effectively applied to fine-tune LLMs from an optimization perspective.

While fine-tuning LLMs is highly effective for enhancing their performance on various specific tasks, full fine-tuning has become impractical in scenarios with limited graphics processing unit (GPU) resources due to the substantial storage demands of the massive parameters. To address these resource constraints, the focus has been on reducing memory usage for optimizer states, gradients, and model weights. One widely adopted method to reduce memory consumption for optimizer states and gradients is Low-Rank Adaptation (LoRA) [24]. LoRA achieves this by significantly reducing the number of trainable parameters, representing the difference between frozen pre-trained weights and fully fine-tuned weights using only trainable low-rank matrices.

Further reductions in memory usage can be achieved through the quantization of model weights. QLoRA [25] successfully combines LoRA with quantization for the first time, demonstrating high performance in 4-bit quantized fine-tuning while significantly reducing GPU memory requirements without incurring additional costs. Nevertheless, when it comes

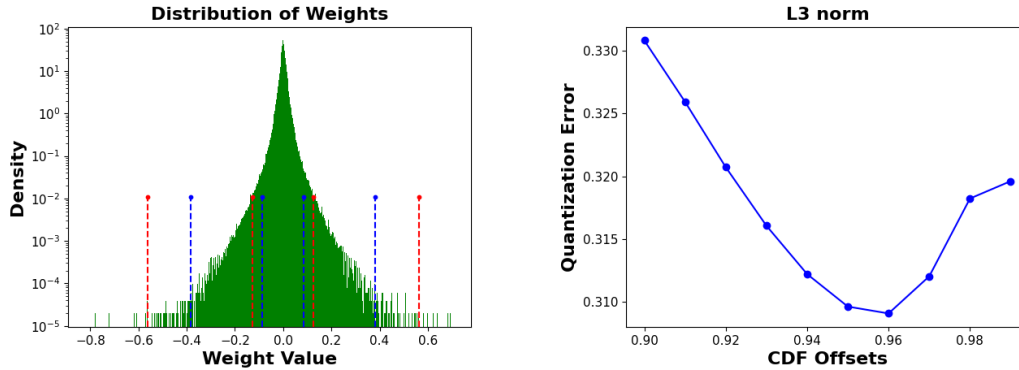


Figure 4.1: The left figure shows the log-scale weight distribution of a specific layer in LLAMA-2-7B. The red points represent the four dequantized values within a particular quantization group using the original 2-bit NormalFloat. With our redesigned NormalFloat, we obtain four blue points that are closer to the center. The right figure illustrates how the L3 norm of the quantization error varies with the CDF offset (See section 4.2.2, section 4.3) within the same quantization group. The four dequantized blue points in the left figure are obtained from an offset of 0.96, resulting in minimal quantization error in this group. By adjusting the offset, we can find the optimal value for each group that minimizes the quantization error.

to extremely low bit fine-tuning regimes, such as 2-bit, QLoRA often fails to converge on many downstream tasks [81]. This suggests that LoRA fine-tuning alone is insufficient to recover from the substantial information loss caused by low-bit model quantization. Even though some efforts have been made to mitigate this information loss through the strategic initialization of LoRA components [81, 82], a fundamental redesign of the NormalFloat quantization data type used in QLoRA is necessary to make low-bit fine-tuning more practical.

The motivation for redesigning the NormalFloat data type stems from understanding why the original NormalFloat experiences significant information loss in 2-bit fine-tuning. As illustrated in Figure 4.1, when an outlier is present in a quantization group, the original NormalFloat bases its dequantization on the group’s maximum value, which corresponds to the outlier. Consequently, the four dequantized values (red points in the left figure of Figure 4.1) are not representative, as 50% of them are outliers, leading to high quantization error. Therefore, in the 2-bit regime, it is crucial to bring these dequantized values closer to

the center (blue points in the left figure of Figure 4.1), as this adjustment can significantly reduce the quantization error.

To address this issue, we first introduce an updated version of the NormalFloat data type, which we call Dynamic NormalFloat. This new version adjusts the dequantized values based on the ratio of the quantile output of the reference CDF offset <sup>1</sup> to the quantile output of our chosen CDF offset, where the CDF offset controls the range of probabilities from the standard normal distribution that are used to generate quantized values. By selecting a lower CDF offset, the adjustment from the red points to the blue points in Figure 4.1 is achieved through this redesigned data type. Additionally, since each quantization group has unique statistical characteristics, we propose the quantization group **Adaptive NormalFloat** (AdaNF), which identifies the optimal CDF offset for each quantization group through grid search by minimizing the Lp norm of the quantization error (see the right figure of Figure 4.1). We evaluate our quantization framework through experiments on various models and downstream tasks in the low-bit fine-tuning regime. Our method outperforms existing approaches for 2-bit fine-tuning and shows comparable performance for 3-bit and 4-bit fine-tuning.

## 4.2 Preliminaries

### 4.2.1 Related Work

Due to the significant computational and memory demands of Large Language Models (LLMs), numerous parameter-efficient fine-tuning (PEFT) methods have been developed. These methods reduce memory and computational costs by optimizing a much smaller subset of parameters compared to the original LLMs [24, 83, 84, 85, 86, 87, 17, 88, 89, 90]. Among these, LoRA [24] is the most widely adopted. It trains low-rank adapter layers on top of the frozen base model, offering stable training due to the implicit regularization of

---

<sup>1</sup>The detailed relationship between CDF offset and the dequantized values is mathematically elaborated in section 4.2.2 and section 4.3.

low-rank adaptation. Furthermore, LoRA facilitates easy modular adaptation across different tasks. Many LoRA variants have since been introduced to closely match the performance of full fine-tuning on more challenging downstream tasks [91, 92, 93, 94, 95].

Combining model quantization with LoRA can further reduce GPU memory consumption during the fine-tuning of LLMs. While many model quantization methods have been developed mainly for inference purposes [96, 97, 98, 99, 100, 101, 102, 103], QLoRA [25] is the first to demonstrate that fine-tuning a quantized 4-bit model with minimal performance degradation is possible by combining NormalFloat quantization with a small set of learnable low-rank adapter weights. However, in extremely low-bit scenarios like the 2-bit regime, QLoRA suffers from significant performance degradation due to substantial weight information loss. To mitigate this, [81, 82] explore strategic initialization of LoRA matrices. [104] proposes a different approach to quantized fine-tuning by focusing on minimizing activation error instead of weight error. [105] addresses information loss from low-bit quantization by calibrating a bias constant groupwisely based on information entropy maximization. Nevertheless, to the best of our knowledge, our AdaNF quantization is the first to highlight the importance of CDF offset initialization in NormalFloat for low-bit fine-tuning, further improving performance through adaptive initialization of the NormalFloat offset for each quantization group.

#### 4.2.2 NormalFloat Quantization

Given that we employ NormalFloat as the framework for our quantization in the low bit LLM fine-tuning, we first present the definition of NormalFloat [25].

**Definition 4.2.1.** (Symmetric NormalFloat)  $Q$  is the quantile function of the standard normal distribution  $N(0, 1)$ , also known as the inverse cumulative distribution function (CDF). Then, for the CDF offset  $c_{\text{offset}} \in (0.5, 1.0)$ , each  $i$ th quantized value of the  $k$ -bit

symmetric NormalFloat data type is represented as

$$q_i = Q\left(1 - c_{\text{offset}} + \frac{2c_{\text{offset}} - 1}{2^k - 1} \times (i - 1)\right) \quad (4.1)$$

for all  $i = 1, 2, \dots, 2^k$ .

The NormalFloat (NF) data type is based on Quantile Quantization [26], an information-theoretically optimal data type that ensures each quantization bin contains an equal number of values from the input tensor. Equation 4.1 in definition 4.2.1 means the  $2^k$  equally spaced quantiles over the range of probabilities  $[1 - c_{\text{offset}}, c_{\text{offset}}]$ . Similarly, we can define the asymmetric NormalFloat that includes 0,  $(2^{k-1} - 1)$  negative values, and  $2^{k-1}$  positive values.

**Definition 4.2.2.** (Asymmetric NormalFloat)  $Q$  is the quantile function of the standard normal distribution  $N(0, 1)$ . Then, for the CDF offset  $c_{\text{offset}} \in (0.5, 1.0)$ , each  $i$ th quantized value of the  $k$ -bit asymmetric NormalFloat data type is represented as

$$q_i = \begin{cases} Q\left(1 - c_{\text{offset}} + \frac{0.5 - (1 - c_{\text{offset}})}{2^{k-1} - 1}(i - 1)\right) & \text{if } i \in [1, 2^{k-1}] \\ Q(0.5)(= 0) & \text{if } i = 2^{k-1} \\ Q\left(0.5 + \frac{c_{\text{offset}} - 0.5}{2^{k-1}}(i - 2^{k-1})\right) & \text{if } i \in (2^{k-1}, 2^k] \end{cases} \quad (4.2)$$

For Equation 4.2 in definition 4.2.2, the first case means  $2^{k-1}$  equally spaced quantiles over the range of probabilities  $[1 - c_{\text{offset}}, 0.5]$ , and the third case means  $2^{k-1} + 1$  equally spaced quantiles over the range of probabilities  $[0.5, c_{\text{offset}}]$ . After obtaining discrete values from either symmetric NormalFloat or asymmetric NormalFloat, we normalize them to the range  $[-1, 1]$  by dividing each value by the maximum value  $Q(c_{\text{offset}})$ . Thus, the exact values of the normalized  $k$ -bit NormalFloat data type are as follows:

$$q_{\text{NF}}^k = \frac{q}{Q(c_{\text{offset}})} = \frac{[q_1, q_2, \dots, q_{2^k}]}{Q(c_{\text{offset}})} \quad (4.3)$$

This allows us to quantize the input weight parameters by normalizing them into the same range  $[-1, 1]$  via absolute maximum rescaling.

### *Group Quantization*

In addition to using NormalFloat as our quantization data type, our framework relies on group quantization to effectively handle outlier issues in the weight parameters. Group quantization involves dividing the input tensor into smaller chunks that are independently quantized. This approach indirectly reduces the number of outliers in each group, leading to smaller quantization errors. Group quantization can be implemented by dividing the weight tensor  $W \in \mathbb{R}^{d \times h}$  into  $n_g$  contiguous groups of size  $G$ . This is done by flattening the weight tensor into a vector  $W^{\text{flat}} \in \mathbb{R}^{dh \times 1}$  and then slicing this vector into  $n_g = \frac{d \times h}{G}$  quantization groups. When we define the  $k$ -bit NormalFloat quantization function as  $Q_{\text{NF}}^k$  and denote the  $i$ -th group tensor as  $W_i^{\text{flat}}$  for  $1 \leq i \leq n_g$ , the quantized output  $W_{q,i}^{\text{flat}}$  can be expressed as

$$W_{q,i}^{\text{flat}} = Q_{\text{NF}}^k \left( \frac{W_i^{\text{flat}}}{\text{absmax}(W_i^{\text{flat}})} \right) \quad (4.4)$$

### 4.2.3 Low-rank Adaptation

Low-Rank Adaptation (LoRA) [24] is a Parameter Efficient Fine-Tuning (PEFT) method that reduces the memory needed for optimizer state and gradient storage by utilizing a small set of trainable parameters, while keeping the main full model parameters fixed. These finetunable parameters, known as adapters, are implemented as factorized projections that augment the original base model. This allows the forward pass to be modified through the adapted model, which can be expressed as:

$$W' = W + \alpha BA$$

where  $W \in \mathbb{R}^{d \times k}$  is a pre-trained weight matrix,  $B \in \mathbb{R}^{d \times r}$ ,  $A \in \mathbb{R}^{r \times k}$ , and  $\alpha$  is a scalar. Here, we note that the rank  $r$  is much smaller than  $\min(d, k)$ . During backpropagation, the gradients flow through the fixed base model weights, which do not receive updates. Instead, only the small number of parameters in the low-rank adapters are updated. We use LoRA as the learnable parameters in our framework.

### 4.3 Method

In this section, we propose quantization group **Adaptive NormalFloat** (AdaNF), an advantageous quantization data type for low-bit LLM fine-tuning. We begin by introducing the concept of Dynamic NormalFloat with a single offset, which involves adjusting the NormalFloat initialization based on the ratio between the quantile output, determined by a specific CDF offset, and a reference quantile value. This adjustment aims to reduce information loss from quantized weight parameters. Additionally, we describe how to adaptively determine the improved NormalFloat initialization for each quantization group using this newly defined Dynamic NormalFloat.

#### 4.3.1 Dynamic NormalFloat with a Single Offset

In the original NormalFloat [25], they set a default CDF offset  $c_{\text{offset}}$  to 0.9677083. Then, they obtain the normalized quantization map within the range  $[-1, 1]$  by dividing all quantile values from definition 4.2.1 or definition 4.2.2 by the maximum quantile value  $Q(c_{\text{offset}})$ , as explained in section 4.2.2. Using this  $k$ -bit normalized NormalFloat quantization map  $q_{\text{NF}}^k$  (see Equation 4.3), the quantized output of the  $i$ -th group tensor is obtained as shown in Equation 4.4. The dequantized weight tensor is then calculated from this quantized tensor

as follows:

$$\begin{aligned} W_{\text{deq},i}^{\text{flat}} &= \text{absmax}(W_i^{\text{flat}}) \cdot W_{q,i}^{\text{flat}} \\ &= \text{absmax}(W_i^{\text{flat}}) \cdot Q_{\text{NF}}^k \left( \frac{W_i^{\text{flat}}}{\text{absmax}(W_i^{\text{flat}})} \right) \end{aligned}$$

Since the maximum and minimum outputs of the function  $Q_{\text{NF}}^k$ <sup>2</sup> are 1 and -1, respectively,  $\text{absmax}(W_i^{\text{flat}})$  and  $-\text{absmax}(W_i^{\text{flat}})$  will be the maximum and minimum among the possible dequantized values for the  $i$ -th quantization group. In low-bit fine-tuning, such as 2-bit, all four possible dequantized values should be representative within the  $i$ -th quantization group. However, if there is an extreme outlier in the group,  $\text{absmax}(W_i^{\text{flat}})$  and  $-\text{absmax}(W_i^{\text{flat}})$  may not be representative, leading to significant quantization errors.

To address this issue, we propose Dynamic NormalFloat, which adjusts the maximum and minimum dequantized values inward (i.e., reduces their absolute values) compared to  $\text{absmax}(W_i^{\text{flat}})$  and  $-\text{absmax}(W_i^{\text{flat}})$ . This ensures that all possible dequantized values are more representative within the group. This adjustment is achieved by further reducing the quantization output range of  $Q_{\text{NF}}^k$  from  $[-1, 1]$ . The updated quantization data type is defined as follows:

**Definition 4.3.1.** (Dynamic NormalFloat) Let  $Q$  be the quantile function of the standard normal distribution  $N(0, 1)$ . For a CDF offset  $c_{\text{offset}} \in (0.5, 1.0)$ , each  $i$ -th quantized value of the  $k$ -bit NormalFloat data type, denoted as  $q_i$ , is derived from either definition 4.2.1 or definition 4.2.2. The reference offset is  $c_{\text{ref}}$ . The  $k$ -bit Dynamic NormalFloat data type  $q_{\text{DNF}}$  is then represented as follows:

$$q_{\text{DNF}}^k|_{c_{\text{offset}}, c_{\text{ref}}} = \frac{q|_{c_{\text{offset}}}}{Q(c_{\text{ref}})} = \frac{[q_1, q_2, \dots, q_{2^k}]}{Q(c_{\text{ref}})} \quad (4.5)$$

The range of  $q_{\text{DNF}}$  is  $\left[ -\frac{Q(c_{\text{offset}})}{Q(c_{\text{ref}})}, \frac{Q(c_{\text{offset}})}{Q(c_{\text{ref}})} \right]$ . Therefore, if  $c_{\text{offset}}$  is less than  $c_{\text{ref}}$ , the range

---

<sup>2</sup>We use nearest rounding for the quantization function.

of  $q_{\text{DNF}}$  becomes smaller than  $[-1, 1]$ . This means that the absolute value of a dequantized weight parameter must be less than  $\text{absmax}(W_i^{\text{flat}})$ . For the  $j$ -th element of the dequantized weight tensor in group  $i$ , we have:

$$\begin{aligned} |W_{\text{deq},i}^{\text{flat}}(j)| &= \text{absmax}(W_i^{\text{flat}}) \left| Q_{\text{DNF}}^k \left( \frac{W_i^{\text{flat}}(j)}{\text{absmax}(W_i^{\text{flat}})} \right) \right| \\ &\leq \text{absmax}(W_i^{\text{flat}}) \frac{Q(c_{\text{offset}})}{Q(c_{\text{ref}})} < \text{absmax}(W_i^{\text{flat}}), \end{aligned}$$

where  $Q_{\text{DNF}}^k$  is the  $k$ -bit Dynamic NormalFloat quantization function defined by  $q_{\text{DNF}}^k$ . Thus, with a properly chosen  $c_{\text{offset}}$ ,  $q_{\text{DNF}}^k$  can potentially result in less quantization error compared to the original quantization map  $q_{\text{NF}}^k$ .

### 4.3.2 Quantization Group Adaptive NormalFloat

---

#### Algorithm 5 Grid Search for AdaNF

---

- 1: **Input:** original weight of a group  $W$ , reference CDF offset  $c_{\text{ref}}$ , number of grids  $n$ , start grid for CDF offset  $c_{\text{start}}$ , end grid for CDF offset  $c_{\text{end}}$ , order of the norm  $p$
  - 2: Initialize  $c^* = c_{\text{start}}$ ,  $E^* = \text{inf}$
  - 3: **for**  $i = 1, \dots, n$  **do**
  - 4:    $c_{\text{offset}} = c_{\text{start}} + \frac{c_{\text{end}} - c_{\text{start}}}{n-1} (i - 1)$
  - 5:   create the  $k$ -bit Dynamic NormalFloat  $q_{\text{DNF}}^k|_{c_{\text{offset}}, c_{\text{ref}}}$
  - 6:   perform nearest rounding with  $q_{\text{DNF}}^k|_{c_{\text{offset}}, c_{\text{ref}}}$ :  
 $W_q = Q_{\text{DNF}}^k|_{c_{\text{offset}}, c_{\text{ref}}} \left( \frac{W}{\text{absmax}(W)} \right)$
  - 7:   get the dequantized weight  $W_{\text{deq}} = \text{absmax}(W)W_q$
  - 8:   compute the quantization error  $E = \|W - W_{\text{deq}}\|_p$
  - 9:   **if**  $E < E^*$  **then**
  - 10:      $c^* \leftarrow c_{\text{offset}}$ ,  $E^* \leftarrow E$
  - 11:   **end if**
  - 12: **end for**
  - 13: **return**  $c^*$
- 

We now introduce quantization group **Adaptive NormalFloat** (AdaNF), which dynamically determines an appropriate CDF offset for the Dynamic NormalFloat data type in each quantization group to minimize quantization error. Since each quantization group has unique statistical characteristics, adjusting the CDF offset for each group can more effectively

preserve information during low-bit quantization compared to using a single offset. We measure the quantization error between the original weight tensor and the dequantized weight tensor using the  $L_p$  norm. This error metric is then used to identify the optimal CDF offset for each group through grid search. Finding an optimal order of the norm  $p$  for each case is crucial. A  $p$  value that is too large will cause the quantization to be overly influenced by outliers, while a  $p$  value that is too small will ignore outliers entirely. Therefore, it is essential to strike a balance by selecting an appropriate  $p$  that adequately considers outliers without being dominated by them. Our algorithm addresses this by exploring and tuning  $p$  to achieve this balance. We empirically find that  $p$  values within the range of  $[2, 3]$  strike this good balance, and the specific  $p$  values used in our experiments are detailed in section 4.4.1. The detailed algorithm is provided in Algorithm 5.

After obtaining the optimal CDF offset for each quantization group through Algorithm 5, we initialize each Dynamic NormalFloat data type with the corresponding offset. Subsequently, the weight parameters are quantized using group quantization, as detailed in section 4.2.2, and these quantized weights remain fixed during the fine-tuning process. For the fine-tuning, we employ the LoRA method, as described in section 4.2.3.

## 4.4 Experiments

In this section, we present experimental results for Dynamic NormalFloat (DNF) and AdaNF<sup>3</sup> on Natural Language Understanding (NLU) and Natural Language Generation (NLG) tasks. We compare our algorithm with QLoRA [25] and other low-bit fine-tuning methods, such as LoftQ [81] and ApiQ [104]. Additionally, we use full fine-tuning and full precision LoRA [24] for reference. For the NLU task, we empirically assess the performance of these algorithms by quantizing the encoder-only DeBERTaV3-base model [106] and fine-tuning it on the General Language Understanding Evaluation (GLUE) benchmark [107]. For the NLG task, we evaluate the performance by quantizing the decoder-only LLAMA-2-7B

---

<sup>3</sup>The more accurate terminology would be QLoRA with DNF and QLoRA with AdaNF, as DNF and AdaNF refer to quantization data types. However, for simplicity, we use the terms DNF and AdaNF.

model [12] and fine-tuning it on two NLG datasets: WikiText-2 [108] and GSM8k [109].

#### 4.4.1 Experimental Setup

**Implementation Details** We follow the implementation setup from [81], with our work largely based on the HuggingFace Transformers codebase [110]. In our model implementation, we retain the original weight matrices in a frozen state and incorporate low-rank adapters into the weight matrices within the all Multi-Head Attention (MHA) and Feed-forward Neural Network (FNN) layers. For the GLUE NLU task, we also quantize the embedding layer of DeBERTaV3-base. For LoRA, we use ranks of 32 for DeBERTaV3-base and 64 for LLAMA-2-7B. Model quantization is then applied to the weight matrices augmented with low-rank adapters. We perform 2-bit fine-tuning for the NLU task and 2, 3, and 4-bit fine-tuning for the NLG task. We use symmetric NormalFloat (see definition 4.2.1) for our DNF and AdaNF. The NVIDIA H100 80GB GPUs are used as computing resources.

**Datasets** For NLU, we use total 8 tasks in GLUE, which includes three natural language inference tasks: MNLI [111], QNLI [112], RTE [113], two single sentence classification tasks: SST-2 [114], CoLA [115], and three similarity and paraphrase tasks: MRPC [116], STS-B [117], QQP. For NLG, we utilize WikiText-2, a dataset derived from Wikipedia articles, and GSM8K, also known as the Grade School Math 8K, a specialized benchmark designed to evaluate the arithmetic reasoning capabilities of language models.

#### *Hyperparameter Choice*

When utilizing DNF quantization, we need to set two hyperparameters: the reference CDF offset  $c_{\text{ref}}$  and a specific CDF offset  $c_{\text{offset}}$ . For AdaNF quantization, five hyperparameters are required:  $c_{\text{ref}}$ , number of grids  $n$ , starting grid for CDF offset  $c_{\text{start}}$ , ending grid for CDF offset  $c_{\text{end}}$ , and the norm order  $p$  (see Algorithm 5). For all experiments,  $c_{\text{ref}}$  is set to 0.995.

In the NLU experiments with the DeBERTaV3-base model, for DNF, we set  $c_{\text{offset}}$  to 0.9 for the QNLI, SST-2, MRPC, CoLA, QQP, and STS-B tasks, and to 0.88 for the MNLI and RTE tasks. In the same NLU experiments, for AdaNF, we measure quantization error using L2.5 and L3 norms (see line 8 in Algorithm 5), meaning  $p$  can be 2.5 or 3. For  $p = 2.5$ , the hyperparameters  $(n, c_{\text{start}}, c_{\text{end}})$  are set to (10, 0.9, 0.99) for MNLI, QNLI, SST-2, and CoLA, and (15, 0.85, 0.99) for RTE, MRPC, QQP, and STS-B. For  $p = 3$ , we use (15, 0.85, 0.99) only for CoLA and (10, 0.9, 0.99) for all other tasks.

In the NLG experiments with the LLAMA-2-7B model, for DNF,  $c_{\text{offset}}$  is set to 0.95, 0.98, and 0.99 for 2-bit, 3-bit, and 4-bit, respectively. For AdaNF in the same NLG experiments, we measure quantization error using L2 and L3 norms. For  $p = 2$ , the hyperparameters  $(n, c_{\text{start}}, c_{\text{end}})$  are set to (10, 0.9, 0.99), (10, 0.9, 0.99), and (15, 0.95, 0.9967) for 2-bit, 3-bit, and 4-bit, respectively. For  $p = 3$ , the hyperparameters are (10, 0.9, 0.99), (15, 0.95, 0.9967), and (15, 0.95, 0.9967) for 2-bit, 3-bit, and 4-bit, respectively.

In all our experiments, we set the quantization group size  $G$  to 64 for DNF, AdaNF, and all baseline methods. Also, regarding the choice of learning rate, for NLU experiments, we follow the setup in [81], except for RTE, where we use  $1 \times 10^{-4}$ . For all NLG experiments, we use  $4 \times 10^{-4}$  as the learning rate.

#### 4.4.2 Experimental Results

##### *NLU with DeBERTaV3-base*

We begin with Natural Language Understanding (NLU) experiments utilizing the relatively smaller DeBERTaV3-base model. The outcomes of these experiments are detailed in Table 4.1. We assess the 2-bit fine-tuning effectiveness of our quantization methods, DNF with a single CDF offset and AdaNF, in comparison to two baselines: QLoRA and LoftQ, across 8 different tasks in the GLUE benchmark. For AdaNF, we explore two scenarios: one employing the L2.5 norm and the other using the L3 norm to evaluate quantization error. In terms of evaluation metrics, a higher score indicates better performance across all 8 tasks.

Table 4.1: 2-bit fine-tuning quantitative results on the GLUE NLU tasks with the DeBERTaV3-base model. We compare our methods, DNF and AdaNF, against two other quantized fine-tuning baselines. N.A. means the model fails to converge.

	MNLI Acc(mm)	QNLI Acc	RTE Acc	SST Acc	MRPC Acc	CoLA Matt	QQP Acc	STS P/S Corr
Full fine-tuning	90.6	94.0	82.0	95.3	89.5/93.3	69.2	92.4/89.8	91.6/91.1
LoRA	90.5	94.6	85.1	95.1	89.9/93.6	69.9	92.0/89.4	91.7/91.1
QLoRA	78.7	80.4	56.7	86.9	73.8/82.7	N.A.	87.1/82.7	83.6/83.3
LoftQ	86.1	89.9	<b>61.7</b>	92.0	<b>83.6/87.2</b>	<b>47.5</b>	<b>91.0/87.9</b>	<b>87.5/87.0</b>
<b>DNF</b>	85.1	88.2	52.3	89.0	73.5/82.0	25.7	89.9/86.4	83.2/82.9
<b>AdaNF (L2.5 norm)</b>	31.8	<b>91.0</b>	58.1	<b>92.9</b>	75.0/83.5	39.4	90.3/87.1	85.6/85.3
<b>AdaNF (L3 norm)</b>	<b>87.0</b>	89.6	58.5	91.9	79.7/86.3	30.3	89.6/86.0	85.5/85.2

When comparing the best results of our methods with QLoRA for each task, our methods outperform QLoRA across all 8 tasks in the GLUE benchmark. This confirms that our redesigned quantization data types, DNF and AdaNF, are indeed improved versions of the original NormalFloat. Notably, AdaNF with the L3 norm consistently surpasses QLoRA in all tasks, demonstrating that the L3 norm is particularly effective for measuring quantization error. This norm effectively balances the influence of outliers within each quantization group, leading to reduced information loss and superior performance in 2-bit fine-tuning (for more detailed insights, see section 4.3.2).

When comparing our best results to those of LoftQ, our methods show better performance in three tasks: MNLI, QNLI, and SST-2. This suggests that our quantization approach is already competitive with the current state-of-the-art LoftQ. Moreover, our methods have the potential for further enhancement through additional tuning of many hyperparameters in Algorithm 5.

Additionally, when comparing our methods internally, AdaNF generally outperforms DNF in nearly all cases, except for the MNLI task when comparing DNF with AdaNF using the L2.5 norm. This observation supports the underlying intuition of the AdaNF algorithm: adaptively finding the optimal CDF offset for each quantization group based on minimal quantization error with the  $L_p$  norm leads to better performance than applying the same

offset to all quantization groups.

*NLG with LLAMA-2-7B*

Table 4.2: Quantitative results on two NLG tasks with LLAMA-2-7B. We compare our methods, DNF and AdaNF, against three other quantized fine-tuning baselines. For reference, LoRA fine-tuning without quantization, which is not included in the table, achieves a perplexity of 5.08 on WikiText-2 and an accuracy of 38.5 on GSM8K. N.A. means the model fails to converge.

	WikiText-2 (Perplexity ↓)			GSM8K (Accuracy ↑)		
	4bit	3bit	2bit	4bit	3bit	2bit
QLoRA	5.70	5.73	N.A.	<b>38.2</b>	32.1	N.A.
LoftQ	5.24	5.63	7.85	38.0	<b>36.2</b>	26.5
ApiQ	5.28	5.53	7.46	36.4	36.0	26.0
<b>DNF</b>	5.21	5.55	6.93	35.4	33.7	<b>27.6</b>
<b>AdaNF (L2)</b>	<b>5.19</b>	<b>5.48</b>	6.88	36.7	32.4	22.8
<b>AdaNF (L3)</b>	<b>5.19</b>	<b>5.48</b>	<b>6.80</b>	35.8	33.5	25.5

To assess the scalability of our methods, we also conducted Natural Language Generation (NLG) experiments using the larger LLAMA-2-7B model. The results of these experiments are summarized in Table 4.2. We compare the low-bit fine-tuning performance of our quantization algorithms, DNF with a single CDF offset and AdaNF, against three baselines: QLoRA, LoftQ, and ApiQ, on WikiText-2 and GSM8k. For AdaNF, we evaluate two cases: one using the L2 norm and another using the L3 norm to measure quantization error. The evaluation metrics used are perplexity for WikiText-2 and accuracy for GSM8K.

For the WikiText-2 experiments, all our methods demonstrate improved perplexity than the three other baselines across 2-bit, 3-bit, and 4-bit settings, with the exception of the DNF 3-bit case. However, even in this instance, the perplexity of DNF 3-bit is only slightly higher than that of ApiQ 3-bit, the most recent of the three baselines. Notably, in the challenging 2-bit scenario, AdaNF with L3 norm achieves the best perplexity score of 6.80, where QLoRA fails to converge. Overall, AdaNF with L3 norm consistently shows the best performance on WikiText-2, indicating that the L3 norm effectively captures quantization error and

aids AdaNF in finding the optimal CDF offset for each quantization group, minimizing information loss. Additionally, it is evident that both versions of AdaNF outperform DNF in terms of perplexity across all bit settings.

For the GSM8K experiments, our method demonstrates outstanding performance in the challenging 2-bit case. Specifically, DNF with a single CDF offset achieves an accuracy of 27.6, surpassing the 26.5 achieved by LoftQ [81], the current state-of-the-art for 2-bit fine-tuning on GSM8K. While DNF outperforms all baselines in the 2-bit fine-tuning scenario, its accuracy for 3-bit and 4-bit fine-tuning is lower compared to other methods. For instance, in the 3-bit setting, both DNF and AdaNF improve upon the original QLoRA but still trail behind LoftQ and ApiQ. Further optimization of our algorithm, such as finer hyperparameter tuning, could enhance these results (see Algorithm 5).

# **Appendices**

## APPENDIX A

### MORE DETAILS ABOUT THE EXPERIMENTS FROM CHAPTER 3

#### A.1 Numerical DP experiment

##### A.1.1 Nearly worst-case Rényi divergence

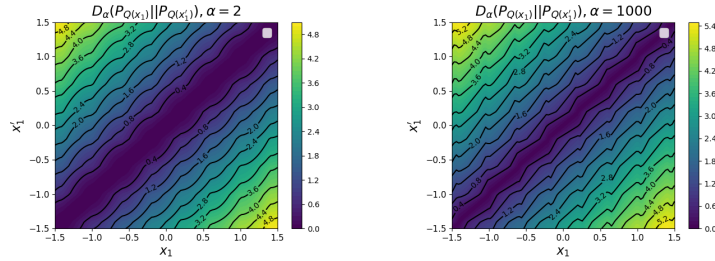


Figure A.1: Both the left ( $\alpha = 2$ ) and the right ( $\alpha = 1000$ ) 2d plot illustrate how the Rényi divergence  $D_\alpha(P_{Q(x_1)} || P_{Q(x'_1)})$  changes with respect to the value of  $x_1$  and  $x'_1$  for the single-device scenario. Here, we follow the hyperparameter choice right above.

To numerically compute Rényi divergence, we use  $\theta = 0.25$  for PBM and  $(\Delta, q) = (c, 0.42)$  for our RQM. Under a single-device case, the peak Rényi divergence  $D_\alpha(P_{Q(x_1)} || P_{Q(x'_1)})$  occurs predominantly around  $(x_1, x'_1) = (c, -c)$  and  $(-c, c)$  (See Figure A.1). Furthermore, when we retain  $x'_1$  at  $-c$ , as per Figure A.2, it's discernible that Rényi divergence  $D_\alpha(P_{Q(x_1)} || P_{Q(-c)})$  increases as  $x_1$  transitions from  $-c$  to  $c$ . In instances of larger  $\alpha$ , minor fluctuations at quantization levels are observed, followed by a swift incline in the Rényi divergence. However, considering these fluctuations as negligible, we deduce that the distance between distributions  $P_{Q(x_1)}$  and  $P_{Q(x'_1)}$  rises almost monotonically as  $x_1$  distances itself from  $x'_1$ . Furthermore, for a more quantitative analysis, the Rényi divergence computed at  $(x_1, x'_1) = (c, -c)$  is 5.46838, as shown in the right plot of Figure A.2. At a quantization level near  $x_1 = c$ , where the Rényi divergence is locally maximized, the value is 5.46190, which is slightly less than the Rényi divergence at  $x_1 = c$ . Thus, in a single-device situation, we can judiciously choose  $x_1 = c$  and  $x'_1 = -c$  to represent the scenario of worst-case

Rényi divergence.

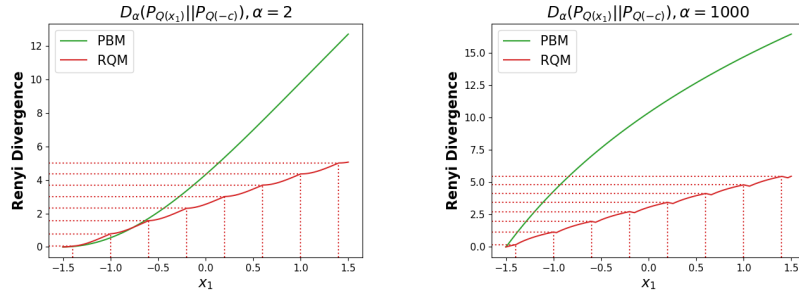


Figure A.2: Both the left ( $\alpha = 2$ ) and the right ( $\alpha = 1000$ ) plot illustrate how the Rényi divergence  $D_\alpha(P_{Q(x_1)}||P_{Q(-c)})$  changes as  $x_1$  increases from  $-c$  to  $c$  for the single-device scenario. Here, we fix  $x'_1 = -c$ .

### A.1.2 Zooming in on the leftmost plot of Figure 3.2

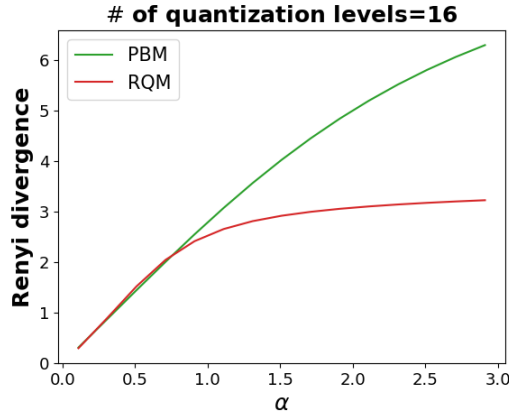


Figure A.3: Comparison of the Rényi Divergence of RQM and PBM for the low regimes of  $\alpha$ .

In Figure A.3, we provide a closer examination of the leftmost plot from Figure 2, which highlights that RQM remains competitive in the very low  $\alpha$  regimes. Additionally, RQM outperforms PBM for moderate  $\alpha$  values, even before the convergence towards pure privacy is reached. Furthermore, the middle and right plots in Figure 2 reveal a distinct advantage of RQM over PBM, with a noticeable performance gap between the two.

### A.1.3 More results on privacy-MSE trade-offs

We conduct further numerical analysis on the privacy-MSE trade-offs in the lower  $\alpha$  regime, where RQM and PBM exhibit similar Rényi divergence, as shown in Figure A.3. In Figure A.4, we fix  $\alpha$  at 0.5, while keeping all other hyperparameters the same as those in section 3.4.4. Even in the lower  $\alpha$  regime, we observe that RQM outperforms PBM in terms of privacy-MSE trade-offs.

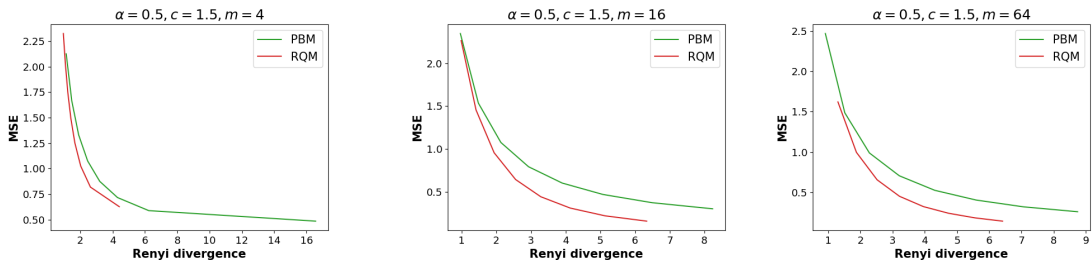


Figure A.4: Additional numerical privacy-MSE trade-offs of RQM and PBM with  $m = 4$  (left),  $m = 16$  (middle), and  $m = 64$  (right), when  $\alpha$  is fixed as 0.5.

## REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Artificial intelligence and statistics*, PMLR, 2017, pp. 1273–1282.
- [2] A. Hard *et al.*, “Federated learning for mobile keyboard prediction,” *arXiv preprint arXiv:1811.03604*, 2018.
- [3] Intel and Consilient, *Intel and consilient join forces to fight financial fraud with ai*, <https://newsroom.intel.com/news/intel-consilient-join-forces-fight-financial-fraud-ai/>, Dec. 2020.
- [4] MELLODDY, *Melloddy project meets its year one objective: Deployment of the world’s first secure platform for multi-task federated learning in drug discovery among 10 pharmaceutical companies*, <https://www.melloddy.eu/y1announcement>, Sep. 2020.
- [5] T. Yang *et al.*, “Applied federated learning: Improving google keyboard query suggestions,” *arXiv preprint arXiv:1812.02903*, 2018.
- [6] L. Zhu, Z. Liu, and S. Han, “Deep leakage from gradients,” *Advances in neural information processing systems*, vol. 32, 2019.
- [7] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, “Our data, ourselves: Privacy via distributed noise generation,” in *Advances in Cryptology-EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28-June 1, 2006. Proceedings 25*, Springer, 2006, pp. 486–503.
- [8] M. Abadi *et al.*, “Deep learning with differential privacy,” in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 308–318.
- [9] T. Brown *et al.*, “Language models are few-shot learners,” *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [10] J. Achiam *et al.*, “Gpt-4 technical report,” *arXiv preprint arXiv:2303.08774*, 2023.
- [11] H. Touvron *et al.*, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.
- [12] H. Touvron *et al.*, “Llama 2: Open foundation and fine-tuned chat models,” *arXiv preprint arXiv:2307.09288*, 2023.

- [13] A. Chowdhery *et al.*, “Palm: Scaling language modeling with pathways,” *Journal of Machine Learning Research*, vol. 24, no. 240, pp. 1–113, 2023.
- [14] G. Team *et al.*, “Gemini: A family of highly capable multimodal models,” *arXiv preprint arXiv:2312.11805*, 2023.
- [15] A. Q. Jiang *et al.*, “Mistral 7b,” *arXiv preprint arXiv:2310.06825*, 2023.
- [16] Q. Dong *et al.*, “A survey on in-context learning,” *arXiv preprint arXiv:2301.00234*, 2022.
- [17] H. Liu *et al.*, “Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 1950–1965, 2022.
- [18] S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He, “Zero: Memory optimizations toward training trillion parameter models,” in *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, IEEE, 2020, pp. 1–16.
- [19] N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan, “cpSGD: Communication-efficient and differentially-private distributed SGD,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
- [20] P. Kairouz, Z. Liu, and T. Steinke, “The distributed discrete gaussian mechanism for federated learning with secure aggregation,” in *International Conference on Machine Learning*, PMLR, 2021, pp. 5201–5212.
- [21] N. Agarwal, P. Kairouz, and Z. Liu, “The skellam mechanism for differentially private federated learning,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 5052–5064, 2021.
- [22] K. Bonawitz *et al.*, “Practical secure aggregation for privacy-preserving machine learning,” in *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1175–1191.
- [23] W.-N. Chen, A. Ozgur, and P. Kairouz, “The poisson binomial mechanism for unbiased federated learning with secure aggregation,” in *International Conference on Machine Learning*, PMLR, 2022, pp. 3490–3506.
- [24] E. J. Hu *et al.*, “Lora: Low-rank adaptation of large language models,” *arXiv preprint arXiv:2106.09685*, 2021.

- [25] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “Qlora: Efficient finetuning of quantized llms,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [26] T. Dettmers, M. Lewis, S. Shleifer, and L. Zettlemoyer, “8-bit optimizers via block-wise quantization,” *arXiv preprint arXiv:2110.02861*, 2021.
- [27] T. Lin, S. U. Stich, K. K. Patel, and M. Jaggi, “Don’t use large mini-batches, use local sgd,” *arXiv preprint arXiv:1808.07217*, 2018.
- [28] A. Khaled, K. Mishchenko, and P. Richtárik, “Tighter theory for local sgd on identical and heterogeneous data,” in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 4519–4529.
- [29] 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.
- [30] F. Haddadpour, M. M. Kamani, A. Mokhtari, and M. Mahdavi, “Federated learning with compression: Unified analysis and sharp guarantees,” in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 2350–2358.
- [31] H. Yuan and T. Ma, “Federated accelerated stochastic gradient descent,” *Advances in Neural Information Processing Systems*, vol. 33, 2020.
- [32] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, “Qsgd: Communication-efficient sgd via gradient quantization and encoding,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 1709–1720, 2017.
- [33] S. U. Stich, “Local sgd converges fast and communicates little,” *arXiv preprint arXiv:1805.09767*, 2018.
- [34] J. Wang and G. Joshi, “Cooperative sgd: A unified framework for the design and analysis of communication-efficient sgd algorithms,” *arXiv preprint arXiv:1808.07576*, 2018.
- [35] F. Haddadpour, M. M. Kamani, M. Mahdavi, and V. Cadambe, “Trading redundancy for communication: Speeding up distributed sgd for non-convex optimization,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 2545–2554.
- [36] H. Yu, S. Yang, and S. Zhu, “Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5693–5700.

- [37] S. U. Stich and S. P. Karimireddy, “The error-feedback framework: Better rates for sgd with delayed gradients and compressed communication,” *arXiv preprint arXiv:1909.05350*, 2019.
- [38] F. Haddadpour, M. M. Kamani, M. Mahdavi, and V. Cadambe, “Local sgd with periodic averaging: Tighter analysis and adaptive synchronization,” in *Advances in Neural Information Processing Systems*, 2019, pp. 11 082–11 094.
- [39] B. Woodworth *et al.*, “Is local sgd better than minibatch sgd?” In *International Conference on Machine Learning*, PMLR, 2020, pp. 10 334–10 343.
- [40] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” *arXiv preprint arXiv:1812.06127*, 2018.
- [41] F. Haddadpour and M. Mahdavi, “On the convergence of local descent methods in federated learning,” *arXiv preprint arXiv:1910.14425*, 2019.
- [42] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, “On the convergence of fedavg on non-iid data,” *arXiv preprint arXiv:1907.02189*, 2019.
- [43] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, “Scaffold: Stochastic controlled averaging for federated learning,” in *International Conference on Machine Learning*, PMLR, 2020, pp. 5132–5143.
- [44] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, “Federated learning: Strategies for improving communication efficiency,” *arXiv preprint arXiv:1610.05492*, 2016.
- [45] A. T. Suresh, X. Y. Felix, S. Kumar, and H. B. McMahan, “Distributed mean estimation with limited communication,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 3329–3337.
- [46] J. Wangni, J. Wang, J. Liu, and T. Zhang, “Gradient sparsification for communication-efficient distributed optimization,” *arXiv preprint arXiv:1710.09854*, 2017.
- [47] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, “Signsgd: Compressed optimisation for non-convex problems,” in *International Conference on Machine Learning*, PMLR, 2018, pp. 560–569.
- [48] H. Wang, S. Sievert, Z. Charles, S. Liu, S. Wright, and D. Papailiopoulos, “Atomo: Communication-efficient learning via atomic sparsification,” *arXiv preprint arXiv:1806.04090*, 2018.

- [49] T. Vogels, S. P. Karinireddy, and M. Jaggi, “Powersgd: Practical low-rank gradient compression for distributed optimization,” *Advances In Neural Information Processing Systems 32 (Nips 2019)*, vol. 32, no. CONF, 2019.
- [50] S. Horvath, C.-Y. Ho, L. Horvath, A. N. Sahu, M. Canini, and P. Richtárik, “Natural compression for distributed deep learning,” *arXiv preprint arXiv:1905.10988*, 2019.
- [51] D. Basu, D. Data, C. Karakus, and S. Diggavi, “Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations,” *arXiv preprint arXiv:1906.02367*, 2019.
- [52] D. Rothchild *et al.*, “Fetchsgd: Communication-efficient federated learning with sketching,” in *International Conference on Machine Learning*, PMLR, 2020, pp. 8253–8265.
- [53] H. Yu, R. Jin, and S. Yang, “On the linear speedup analysis of communication efficient momentum sgd for distributed non-convex optimization,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 7184–7193.
- [54] S. P. Karimireddy *et al.*, “Mime: Mimicking centralized stochastic algorithms in federated learning,” *arXiv preprint arXiv:2008.03606*, 2020.
- [55] J. Wang, Z. Xu, Z. Garrett, Z. Charles, L. Liu, and G. Joshi, “Local adaptivity in federated learning: Convergence and consistency,” *arXiv preprint arXiv:2106.02305*, 2021.
- [56] Z. Li, D. Kovalev, X. Qian, and P. Richtárik, “Acceleration for compressed gradient descent in distributed and federated optimization,” *arXiv preprint arXiv:2002.11364*, 2020.
- [57] Z. Li and P. Richtárik, “Canita: Faster rates for distributed convex optimization with communication compression,” *arXiv preprint arXiv:2107.09461*, 2021.
- [58] N. Singh, D. Data, J. George, and S. Diggavi, “Squarm-sgd: Communication-efficient momentum sgd for decentralized optimization,” *IEEE Journal on Selected Areas in Information Theory*, 2021.
- [59] X. Li, B. Karimi, and P. Li, “On distributed adaptive optimization with gradient compression,” *arXiv preprint arXiv:2205.05632*, 2022.
- [60] Y. Wang, L. Lin, and J. Chen, “Communication-efficient adaptive federated learning,” *arXiv preprint arXiv:2205.02719*, 2022.

- [61] S. Ghadimi and G. Lan, “Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework,” *SIAM Journal on Optimization*, vol. 22, no. 4, pp. 1469–1492, 2012.
- [62] N. Bansal and A. Gupta, “Potential-function proofs for gradient methods,” *Theory of Computing*, vol. 15, no. 1, pp. 1–32, 2019.
- [63] Y. LeCun, “The mnist database of handwritten digits,” <http://yann.lecun.com/exdb/mnist/>, 1998.
- [64] A. Krizhevsky, G. Hinton, *et al.*, “Learning multiple layers of features from tiny images,” Manuscript, 2009.
- [65] J. Wang *et al.*, “A field guide to federated optimization,” *arXiv preprint arXiv:2107.06917*, 2021.
- [66] Y. Youn, B. Kumar, and J. Abernethy, “Accelerated federated optimization with quantization,” in *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*, 2022.
- [67] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, “Secure single-server aggregation with (poly) logarithmic overhead,” in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 1253–1269.
- [68] C. Dwork, A. Roth, *et al.*, “The algorithmic foundations of differential privacy,” *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.
- [69] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, “Learning differentially private recurrent language models,” *arXiv preprint arXiv:1710.06963*, 2017.
- [70] R. C. Geyer, T. Klein, and M. Nabi, “Differentially private federated learning: A client level perspective,” *arXiv preprint arXiv:1712.07557*, 2017.
- [71] K. Chaudhuri, C. Guo, and M. Rabbat, “Privacy-aware compression for federated data analysis,” in *Uncertainty in Artificial Intelligence*, PMLR, 2022, pp. 296–306.
- [72] C. Guo, K. Chaudhuri, P. Stock, and M. Rabbat, “Privacy-aware compression for federated learning through numerical mechanism design,” in *International Conference on Machine Learning*, PMLR, 2023, pp. 11 888–11 904.
- [73] T. Li, Z. Liu, V. Sekar, and V. Smith, “Privacy for free: Communication-efficient learning with differential privacy using sketches,” *arXiv preprint arXiv:1911.00972*, 2019.

- [74] V. Gandikota, D. Kane, R. K. Maity, and A. Mazumdar, “vqSGD: Vector quantized stochastic gradient descent,” in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 2197–2205.
- [75] Ú. Erlingsson, V. Pihur, and A. Korolova, “Rappor: Randomized aggregatable privacy-preserving ordinal response,” in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, 2014, pp. 1054–1067.
- [76] I. Mironov, “Rényi differential privacy,” in *2017 IEEE 30th computer security foundations symposium (CSF)*, IEEE, 2017, pp. 263–275.
- [77] A. Rényi, “On measures of entropy and information,” in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, University of California Press, vol. 4, 1961, pp. 547–562.
- [78] D. Levy *et al.*, “Learning with user-level privacy,” *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [79] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, “Emnist: Extending mnist to handwritten letters,” in *2017 international joint conference on neural networks (IJCNN)*, IEEE, 2017, pp. 2921–2926.
- [80] Martín Abadi *et al.*, *TensorFlow: Large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015.
- [81] Y. Li *et al.*, “Loftq: Lora-fine-tuning-aware quantization for large language models,” *arXiv preprint arXiv:2310.08659*, 2023.
- [82] H. Guo, P. Greengard, E. P. Xing, and Y. Kim, “Lq-lora: Low-rank plus quantized matrix decomposition for efficient language model finetuning,” *arXiv preprint arXiv:2311.12023*, 2023.
- [83] N. Houlsby *et al.*, “Parameter-efficient transfer learning for nlp,” in *International conference on machine learning*, PMLR, 2019, pp. 2790–2799.
- [84] G. Qin and J. Eisner, “Learning how to ask: Querying lms with mixtures of soft prompts,” *arXiv preprint arXiv:2104.06599*, 2021.
- [85] B. Lester, R. Al-Rfou, and N. Constant, “The power of scale for parameter-efficient prompt tuning,” *arXiv preprint arXiv:2104.08691*, 2021.
- [86] X. L. Li and P. Liang, “Prefix-tuning: Optimizing continuous prompts for generation,” *arXiv preprint arXiv:2101.00190*, 2021.

- [87] S. An *et al.*, “Input-tuning: Adapting unfamiliar inputs to frozen pretrained models,” *arXiv preprint arXiv:2203.03131*, 2022.
- [88] E. B. Zaken, S. Ravfogel, and Y. Goldberg, “Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models,” *arXiv preprint arXiv:2106.10199*, 2021.
- [89] Y.-L. Sung, V. Nair, and C. A. Raffel, “Training neural networks with fixed sparse masks,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 193–24 205, 2021.
- [90] R. Karimi Mahabadi, J. Henderson, and S. Ruder, “Compacter: Efficient low-rank hypercomplex adapter layers,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 1022–1035, 2021.
- [91] Q. Zhang *et al.*, “Adaptive budget allocation for parameter-efficient fine-tuning,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [92] A. Renduchintala, T. Konuk, and O. Kuchaiev, “Tied-lora: Enhancing parameter efficiency of lora with weight tying,” *arXiv preprint arXiv:2311.09578*, 2023.
- [93] D. J. Kopiczko, T. Blankevoort, and Y. M. Asano, “Vera: Vector-based random matrix adaptation,” *arXiv preprint arXiv:2310.11454*, 2023.
- [94] W. Xia, C. Qin, and E. Hazan, “Chain of lora: Efficient fine-tuning of language models via residual learning,” *arXiv preprint arXiv:2401.04151*, 2024.
- [95] M. Nikdan, S. Tabesh, and D. Alistarh, “Rosa: Accurate parameter-efficient fine-tuning via robust adaptation,” *arXiv preprint arXiv:2401.04679*, 2024.
- [96] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, “Gptq: Accurate post-training quantization for generative pre-trained transformers,” *arXiv preprint arXiv:2210.17323*, 2022.
- [97] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, “Optq: Accurate quantization for generative pre-trained transformers,” in *The Eleventh International Conference on Learning Representations*, 2022.
- [98] W. Shao *et al.*, “Omniquant: Omnidirectionally calibrated quantization for large language models,” *arXiv preprint arXiv:2308.13137*, 2023.
- [99] G. Xiao, J. Lin, M. Seznec, H. Wu, J. Demouth, and S. Han, “Smoothquant: Accurate and efficient post-training quantization for large language models,” in *International Conference on Machine Learning*, PMLR, 2023, pp. 38 087–38 099.

- [100] T. Dettmers *et al.*, “Spqr: A sparse-quantized representation for near-lossless llm weight compression,” *arXiv preprint arXiv:2306.03078*, 2023.
- [101] S. Kim *et al.*, “Squeezellm: Dense-and-sparse quantization,” *arXiv preprint arXiv:2306.07629*, 2023.
- [102] J. Lin, J. Tang, H. Tang, S. Yang, X. Dang, and S. Han, “Awq: Activation-aware weight quantization for llm compression and acceleration,” *arXiv preprint arXiv:2306.00978*, 2023.
- [103] J. Chee, Y. Cai, V. Kuleshov, and C. M. De Sa, “Quip: 2-bit quantization of large language models with guarantees,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [104] B. Liao and C. Monz, “Apiq: Finetuning of 2-bit quantized large language model,” *arXiv preprint arXiv:2402.05147*, 2024.
- [105] H. Qin *et al.*, “Accurate lora-finetuning quantization of llms via information retention,” *arXiv preprint arXiv:2402.05445*, 2024.
- [106] P. He, J. Gao, and W. Chen, “Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing,” *arXiv preprint arXiv:2111.09543*, 2021.
- [107] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” *arXiv preprint arXiv:1804.07461*, 2018.
- [108] S. Merity, C. Xiong, J. Bradbury, and R. Socher, “Pointer sentinel mixture models,” *arXiv preprint arXiv:1609.07843*, 2016.
- [109] K. Cobbe *et al.*, “Training verifiers to solve math word problems,” *arXiv preprint arXiv:2110.14168*, 2021.
- [110] A. Paszke *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *Advances in neural information processing systems*, vol. 32, 2019.
- [111] A. Williams, N. Nangia, and S. R. Bowman, “A broad-coverage challenge corpus for sentence understanding through inference,” *arXiv preprint arXiv:1704.05426*, 2017.
- [112] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “Squad: 100,000+ questions for machine comprehension of text,” *arXiv preprint arXiv:1606.05250*, 2016.

- [113] I. Dagan, O. Glickman, and B. Magnini, “The pascal recognising textual entailment challenge,” in *Machine learning challenges workshop*, Springer, 2005, pp. 177–190.
- [114] R. Socher *et al.*, “Recursive deep models for semantic compositionality over a sentiment treebank,” in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.
- [115] A. Warstadt, A. Singh, and S. R. Bowman, “Neural network acceptability judgments,” *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 625–641, 2019.
- [116] B. Dolan and C. Brockett, “Automatically constructing a corpus of sentential paraphrases,” in *Third international workshop on paraphrasing (IWP2005)*, 2005.
- [117] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, “Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation,” *arXiv preprint arXiv:1708.00055*, 2017.