

Mixed-Precision Federated Learning via Multi-Precision Over-the-Air Aggregation

Jinsheng Yuan*

Faculty of Engineering & Applied Sciences
Cranfield University
United Kingdom
jinsheng.yuan@cranfield.ac.uk

Zhuangkun Wei

Department of Engineering
Durham University
United Kingdom
zhuangkun.wei@durham.ac.uk

Weisi Guo

Faculty of Engineering & Applied Sciences
Cranfield University
United Kingdom
weisi.guo@cranfield.ac.uk

Abstract—Over-the-Air Federated Learning (OTA-FL) is a privacy-preserving distributed learning mechanism, by aggregating updates in the electromagnetic channel rather than at the server. A critical research gap in existing OTA-FL research is the assumption of homogeneous client computational bit precision. While in real world application, clients with varying hardware resources may exploit approximate computing (AxC) to operate at different bit precisions optimized for energy and computational efficiency. Model updates with varying precisions among clients present a significant challenge for OTA-FL, as they are incompatible with the wireless modulation superposition process. Here, we propose an mixed-precision OTA-FL framework of clients with multiple bit precisions, demonstrating the following innovations: (i) the superior trade-off for both server and clients within the constraints of varying edge computing capabilities, energy efficiency, and learning accuracy requirements compared to homogeneous client bit precision, and (ii) a multi-precision gradient modulation scheme to ensure compatibility with OTA aggregation and eliminate the overheads of precision conversion. Through case study with real world data, we validate our modulation scheme that enables AxC based mixed-precision OTA-FL. In comparison to homogeneous standard precision of 32-bit and 16-bit, our framework presents more than 10% in 4-bit ultra low precision client performance and over 65% and 13% of energy savings respectively. This demonstrates the great potential of our mixed-precision OTA-FL approach in heterogeneous edge computing environments.

Index Terms—Over-The-Air Computation, Federated Learning, Approximate Computing

I. INTRODUCTION

Federated Learning (FL) [1], see in Fig. 1a, has emerged as a widely studied and applied distributed learning framework that ensures security and privacy by sharing and aggregating model parameters instead of raw data, as is done in centralized learning. Recent advancements in FL research [2] primarily focus on two key aspects: (i) enhancing privacy and security, and (ii) improving efficiency.

Over-the-Air Federated Learning (OTA-FL) [3], see in Fig. 1b, represents a novel paradigm in FL dedicated for wireless networks. By leveraging the inherent randomness of

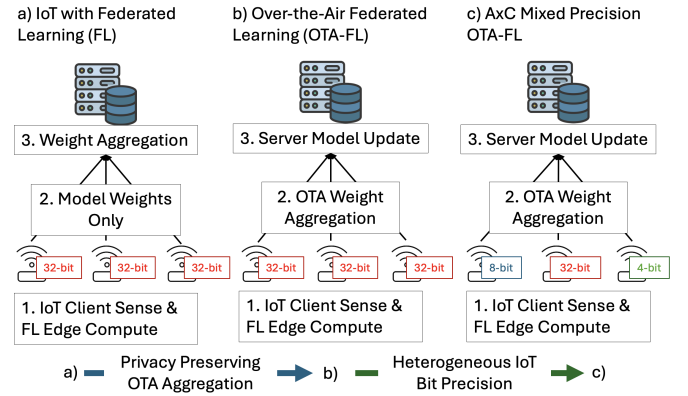


Fig. 1. End-to-end Federated Learning (FL) system moves from (a) FL, to (b) privacy preserving OTA-FL, to (c) energy efficient AxC OTA-FL. The research challenge from (b) to (c) is to achieve heterogeneous OTA weight aggregation to cater for mixed bit precision IoT edge computation.

physical-layer channel states and electromagnetic superposition for aggregating model updates, OTA-FL enhances privacy preservation without the additional computational overhead associated with other approaches, such as those employing differential privacy mechanisms [4]. Recent advancements in OTA-FL encompass transmitting node precoding [5], server beamforming vector optimization [6], and reconfigurable intelligent surface (RIS) phase adjustment [7].

Related Works: In Federated Learning (FL) systems, optimizing the balance between task performance and resource constraints such as communication, computation, and energy has been a key area of focus. Strategies aimed at reducing communication overhead include SCAFFOLD [8], which uses control variates to correct for client drift, and LoSAC [9], which locally updates the estimate for the global full gradient after each local model update to enhance communication efficiency. These methods achieve faster convergence with fewer communication rounds. On the computation side, various approaches have been explored, including hardware acceleration [10], network architecture optimization [11], model slicing that assigns sub-models to clients based on their hardware capabilities [12], and Approximate Computing (AxC) methods.

*Corresponding author. The work is supported by EPSRC CHEDDAR: Communications Hub for Empowering Distributed cloud computing Applications and Research (EP/X040518/1) (EP/Y037421/1). We acknowledge Dr. Yun Wu at QUB for his inspiration.

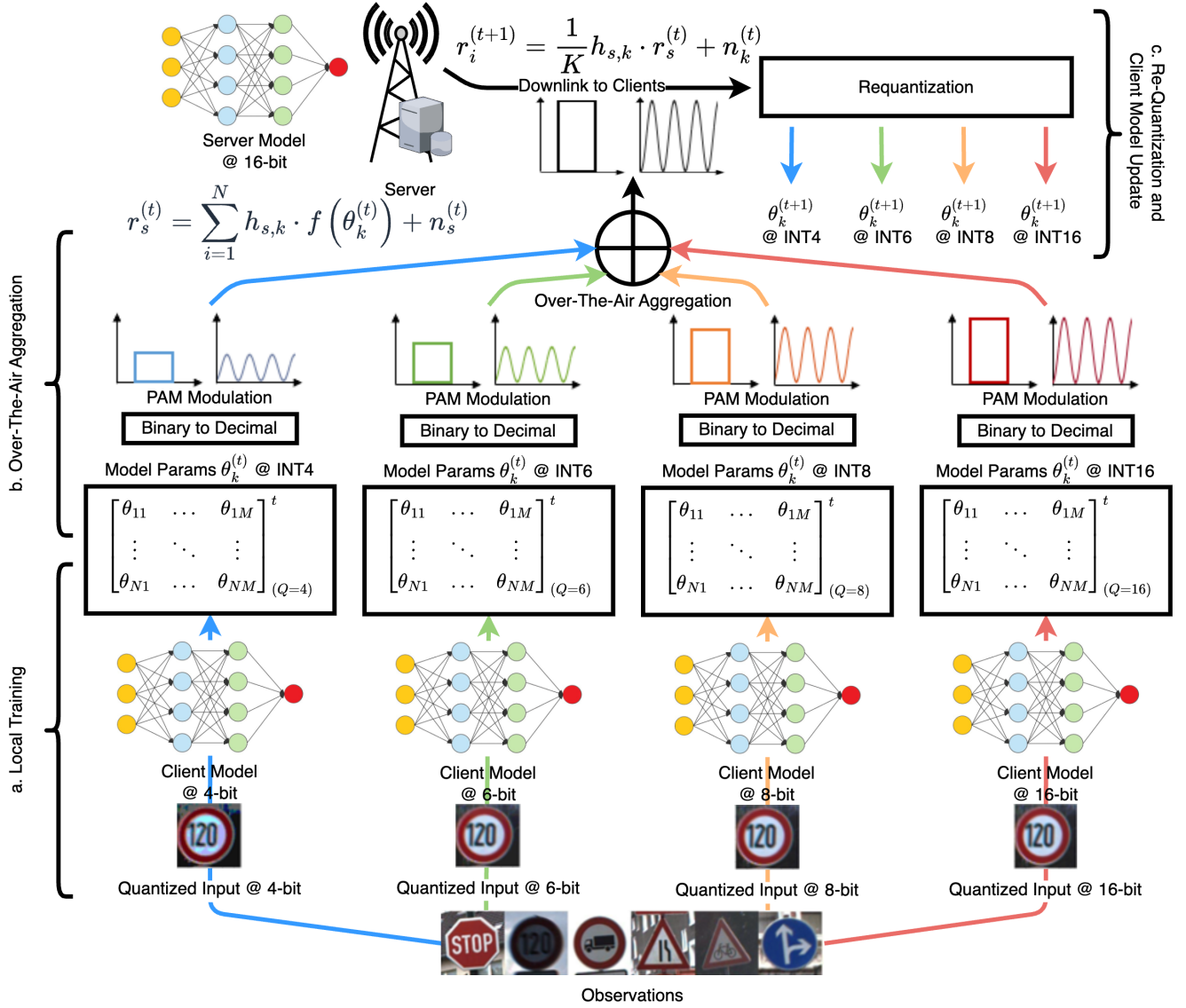


Fig. 2. Structure of our proposed Approximate Computing (AxC) based OTA-FL framework of multi-precision clients and unified multi-precision modulation scheme. The intelligent transport validation case study here is a multi-precision federated supervised traffic sign recognizer. (a) Clients operate end-to-end at their designated computation precisions with their own data and labels. (b) Multi-precision OTA aggregation process (uplink). (c) Downlink, re-quantization and client model update.

Inspiration and Motivation: Approximate Computing (AxC) methods encompass both hardware and software domains, aiming to balance task performance with energy efficiency and computational resource optimization. A significant source of inspiration for us within the AxC domain arises from Field Programmable Gate Arrays (FPGA) accelerators [13]. Unlike traditional Central Processing Units (CPUs) and Graphics Processing Units (GPUs), which operate at fixed preset precision levels, FPGAs provide a customizable computation paradigm. This flexibility of reprogrammability allows for highly efficient operations at varying precision levels across diverse applications, including networking, signal processing, multimedia codec operations, and machine learning acceleration [14]. In this work, we investigate the potential of

heterogeneous quantization to address computational and communication trade-offs in Federated Learning (FL). The research focuses on two key questions: (i) what are the potential gains in computation, energy efficiency, and performance when using heterogeneous client precision levels compared to conventional homogeneous approaches, and (ii) how to design an effective modulation scheme that supports multi-precision client updates in Over-the-Air (OTA) aggregation.

Contribution: In this paper, our contributions are threefold:

- 1) We propose an AxC-based OTA-FL framework for multi-precision clients, aimed at bolstering performance and efficiency - see Fig. 1(c).
- 2) We develop heterogeneous gradient resolution modulation schemes to ensure compatibility with physical-

layer OTA aggregation and eliminate the overheads of precision conversion - see Fig. 2(b).

- 3) We conducted a case study with real-world data to demonstrate the effectiveness of our approach in comparison to homogeneous precision OTA-FL systems. Results show a notable improvement in server convergence speed and more than 10% in 4-bit ultra low precision client performance and over 65% and 13% of energy savings respectively compared to FL with homogeneous 32-bit and 16-bit clients. These findings highlight the potential of our framework in resource-constrained, heterogeneous computing environments.

Paper Structure: The remainder of the paper is organized as follows: Section II provides a detailed exploration of the system setup of our mixed-precision OTA-FL. Section III delves into the design specifics of our Ax-C-based OTA-FL framework. Experiments and results are presented in Section IV. Section V concludes the paper, summarizing key findings and contributions.

II. SYSTEM SETUP

A. Federated Learning

We consider an over-the-air federated learning system with N clients, at each communication round, K of them are selected to update denoted by $\mathcal{K} = \{1, \dots, K\}$. Each client k has a local dataset \mathcal{D}_k . For \mathcal{T} communication rounds, the clients collaboratively refine a unified global model, while preserving the privacy of their local data. This process collects the local trained models of clients, and aggregates these models to update the global model. The refined global model is then distributed back to the edge devices for subsequent predictions and further local training iterations, thereby progressively enhancing the global model's accuracy. The model aggregation phase, central to FL, can be mathematically represented as:

$$\theta^{(t+1)} = \frac{1}{K} \sum_{k=1}^K w_k \cdot \theta_k^{(t)}, \quad (1)$$

where $\theta^{(t+1)}$ denotes the parameters of the global model after the $(t+1)$ -th training iteration, $\theta_k^{(t)}$ represents the parameters of the local model from the edge device k at the t -th communication round, and w_k signifies the relative contribution (or weight) of the edge device k , typically proportional to its dataset size.

B. Over-The-Air (OTA) Computation for FL

The principle of Over-The-Air (OTA) computation [15], exploits the natural superposition property inherent to wireless channels. Within the FL framework, this concept finds practical application during the gradient aggregation phase. By utilizing a common uplink bandwidth across all edge devices for gradient transmission, the superposition property of the channel facilitates direct aggregation. Consider a Single-Input Single-Output (SISO) fading channel between the server and an edge device k , characterized by a Rayleigh distributed

random variable $h_{s,k} \in \mathbb{C}$. The OTA aggregation process can then be modeled as:

$$r_s^{(t)} = \sum_{k=1}^K h_{s,k} \cdot f\left(\theta_k^{(t)}\right) + n_s^{(t)}, \quad (2)$$

where $f(\cdot)$ embodies the comprehensive process including source coding, constellation design, up-sampling, modulation, and precoding. Here, $r_s^{(t)}$ represents the aggregated signal received by the server in the t -th upload cycle, and $n_s^{(t)}$ denotes the additive noise.

In contrast to traditional federated learning aggregation mechanisms, leveraging OTA computation necessitates careful design of $f(\cdot)$ to address two critical challenges: (i) mitigating the effects of channel fading, and (ii) ensuring accurate linear summation of gradients from multiple edges. This becomes particularly complex when considering edges with disparate computation, storage, and operation precision constraints. A notable challenge arises in aggregating gradients quantized at different levels, exemplified by the non-commutative property of quantized modulations:

$$QAM([\theta_j]_{q_j}) + QAM([\theta_k]_{q_k}) \neq QAM([\theta_j]_{q_j} + [\theta_k]_{q_k}), \quad (3)$$

where $QAM(\cdot)$ represent the quadrature modulation function and $[\theta_k]_{q_k}$ indicates model parameters of client k is at q_k quantization levels (in terms of bits) utilized for encoding weights and biases of models at different edge devices.

C. Approximate Computing and Low-Precision ML

Within Ax-C methods, quantization stands out for its universal applicability, and compatibility with existing FL systems. In the context of edge computing, FPGA accelerators become particularly advantageous, as they can be dynamically reprogrammed to efficiently perform computation at any custom precision level including 4-bit and below, thus significantly reducing computation demand and energy consumption in resource-constrained environments [10].

TABLE I
GTSRB CLASSIFICATION PERFORMANCE OF COMMON MODELS ACROSS QUANTIZATION LEVELS

Model	8-bit	6-bit	4-bit	3-bit	2-bit
densenet_161	96.56%	96.45%	91.55%	39.83%	0.00%
efficient_net_b4	94.77%	94.74%	90.55%	42.24%	0.07%
efficient_net_v2m	95.26%	95.19%	85.75%	7.68%	6.22%
reg_net_x_16gf	96.56%	96.11%	80.84%	0.00%	0.00%
reg_net_y_3_2gf	93.53%	92.75%	72.72%	6.97%	0.00%
resnet_50	94.94%	94.33%	65.21%	0.00%	0.83%
squeeze_net_1_0	87.29%	85.41%	72.95%	39.85%	6.64%

Orange: damaged but usable performance (65 – 85%)

Red: unacceptable performance (< 65%).

As quantization can significantly reduce the resource demand for storing, inferring, and training machine learning models [16], [17] at the price of performance degradation, it is essential to demonstrate the trade-offs of low-precision machine learning. We benchmarked the performance degradation of common CNN models in quantization, as shown in

Table. I. All these models are trained in 32-bit floating-point format and then quantize to lower bit-levels. The degradation only becomes noticeable when quantized to 8-bit, and retains acceptable before being quantized below 4-bit.

It is also worth noting that such performance can not be stably achieved via conventional end-to-end training at the same lowest precision level due to the limit of gradient dynamic range and cumulative error of low precision, and additionally, when the size of dataset increases, gradients may require a larger dynamic range to fit the data.

Although end-to-end low precision training can be achieved through dedicated design of format, quantization algorithms, and arithmetic implementation [18], a key advantage of our mixed-precision OTA-FL framework is simplicity for application. We leverage mixed-precision federated learning, to enable the low precision ML on edge devices and hence achieve superior trade-off of computation and performance, and reduction of energy consumption.

III. MIXED-PRECISION OTA FEDERATED LEARNING

The process of our Mixed-Precision OTA Federated Learning framework can be described as Algorithm 1.

Algorithm 1 Mixed-Precision OTA Federated Learning

```

1: Input: Initial global model  $\theta^{(0)}$ , number of communication rounds  $T$ , number of clients  $K$ 
2: Output: Trained global model  $\theta^{(T)}$ 
3: for each round  $t = 1$  to  $T$  do
4:   Step 1: Broadcast global model  $\theta^{(t-1)}$ 
5:   Server broadcasts global model  $\theta^{(t-1)}$  to all clients
6:   Step 2: Local training at clients
7:   for each client  $k = 1$  to  $K$  (in parallel) do
8:     Quantize  $\theta^{(t-1)}$  to designated precision level  $q_k$ 
9:     Local training  $[\theta_k^{(t)}]_{q_k} = \text{Training}([\theta^{(t-1)}]_{q_k}, \mathcal{D}_k)$ 
10:    Calculate update  $\Delta[\theta_k^{(t)}]_{q_k} = [\theta_k^{(t)}]_{q_k} - [\theta^{(t-1)}]_{q_k}$ 
11:   end for
12:   Step 3: Over-the-Air aggregation of mixed-precision model updates
13:   for each client  $k = 1$  to  $K$  (in parallel) do
14:     Convert model update  $\Delta[\theta_k^{(t)}]_{q_k}$  to decimal
15:     Amplitude modulation, channel estimation and uplink
16:   end for
17:   Step 4: Server-side post-processing and downlink
18:   Received signal  $r_s^{(t)} \approx \sum_{k=1}^K \theta_k^{(t)}$ 
19:   Broadcast updated model  $\frac{r_s^{(t)}}{K} \approx \frac{\sum_{k=1}^K \theta_k^{(t)}}{K}$ 
20: end for
21: Return final global model  $\theta^{(T)}$ 

```

A. Multi-Precision Over-The-Air Aggregation

Over-The-Air Federated Learning (OTA-FL) encompasses a three-step process during each update round, denoted as the t -th round. The sequence begins with each client k conducting local training and producing a model update $\Delta[\theta_k^{(t)}]_{q_k}$. Subsequently, as shown in Fig. 2 (b), the client prepares for

transmission by converting binary parameters of its designated precision into decimal equivalents. These decimal values are then modulated onto carrier waves through amplitude modulation, creating signals ready for bandwidth transmission. Our modulation function can be described as follows.

$$M([\theta_k]_{q_k}) = [\theta_k]_{q_k} \cdot \cos 2\pi f_c t \quad (4)$$

where $M(\cdot)$ represent the modulation function, $[\theta_i]_{q_i}$ indicates model parameters of client k is at q_k quantization levels, and f_c is the channel frequency.

In the upload phase, the client k performs the channel estimation to determine the communication link, denoted as $h_{s,k}$, between the server and itself. This estimated channel information is crucial for implementing transmission beamforming, a technique employed for channel compensation. Such compensation is integral to facilitating efficient OTA aggregation of transmitted data. The procedural intricacies of this approach are elucidated below.

1) *Channel Estimation at Clients:* Channel estimation between the server and each client is done by broadcasting a predefined pilot sequences u from the server. Then, client k can estimate the channel between k and the server $h_{s,k}$ as follows.

$$\hat{h}_{s,k} = y_s \cdot \frac{u^*}{|u|^2} \approx h_{s,k}, \quad (5)$$

where $y_s = h_{s,k} \cdot u + n_s$ is the received signals at server with receiving noise n_s .

2) *Uplink Design:* After channel estimation at each client k , the clients modulate model updates to carrier frequency and compensate for channel distortion via its estimated channel. The base-band transmitted signal is designed as follows.

$$f(\theta_k^{(t)}) = \hat{h}_{s,k}^{-1} \cdot \theta_k^{(t)}. \quad (6)$$

Hence, aggregation can be done via the natural superposition of the electromagnetic wave, i.e., $r_s^{(t)} \approx \sum_{k=1}^K \theta_k^{(t)}$.

3) *Downlink Design:* After OTA aggregation, the server broadcasts the updated model, i.e., $r_s^{(t)}/K \approx \sum_{k=1}^K \theta_k^{(t)}/K$ back to clients. The received signal at each client k is:

$$r_k^{(t+1)} = \frac{1}{K} h_{s,k} \cdot r_s^{(t)} + n_k^{(t)}. \quad (7)$$

The client i then recovers the aggregated gradient via the estimated channel, as follows.

$$\tilde{\theta}_k^{(t+1)} = h_{s,k}^{-1} \cdot r_k^{(t+1)} \approx \frac{1}{K} \sum_{k=1}^K \theta_k^{(t)}, \quad (8)$$

where $\tilde{\theta}_k^{(t+1)}$ is the updated gradient of client k , and will be local trained for the $(t+1)$ th round.

Furthermore, during the OTA aggregation process, the mixed-precision quantization scheme, which remains inherently opaque to potential adversaries, significantly enhances the system's security against attacks targeting the aggregation process.

B. Approximate Computing via Quantization

Algorithm 2 Quantization Function

```

1: Input: Tensor  $W \in \mathbb{R}^{m \times n}$ , type: "fixed-point" or "floating-point", bit-width  $b$ 
2: Output: Quantized tensor  $Q \in \mathbb{R}^{m \times n}$ 
3: if type is "fixed" then
4:    $w_{\min} = \min(W)$ ,  $w_{\max} = \max(W)$ 
5:    $scale = \frac{w_{\max} - w_{\min}}{2^b - 1}$ ,  $zero\_point = -\frac{w_{\min}}{scale}$ 
6:   for each  $w_{ij}$  in  $W$  do
7:      $q_{ij} = \max(0, \min(2^b - 1, \lfloor \frac{w_{ij}}{scale} + zero\_point \rfloor))$ 
8:   end for
9: else if type is "floating-point" then
10:  for each  $w_{ij}$  in  $W$  do
11:    Truncate mantissa and exponent to fit  $b$  bits
12:  end for
13: end if
14: Return  $Q$ 

```

To ensure broad applicability, we employ a simple and efficient quantization algorithm, as outlined in Algorithm 2. For precision levels of 8-bit and higher, both fixed-point and floating-point formats are supported. However, fixed-point format is preferred for lower precision levels due to the limited dynamic range of floating-point formats under 8-bit representation.

On the client side, the quantization function is systematically applied to every layer of the CNN model, spanning from input to output, and is integrated into both the forward and backward passes. This approach ensures a unified precision level throughout the end-to-end system.

C. Energy Consumption Estimation

Along with compute savings, as an equally important product of quantization, we also measure the energy savings which originate from the higher throughput of operations at lower precisions. We estimate the energy consumption of training a ResNet-50 model at multiple precision levels between 32-bit and 4-bit on 9 Xilinx FPGA platforms of varying specifications. Then we use the average relative energy saving compared to 32-bit at each precision levels of all these platforms to estimate the total savings for our mixed-precision clients.

However, the energy consumption for the same task of FPGA platforms can vary significantly based on the program design of the same hardware. In FPGA focused researches, accurate energy consumption can be obtained through the official power analysis toolkit [19]. For this paper, we provide a modest energy consumption estimation for the clients based on official data sheets [20] of typical FPGA edge platforms with Equation 9 below to showcase the potential of energy savings of our approach.

$$E_{ML} = \frac{D_{ML}}{F_{DSP} \cdot N_{DSP} \cdot N_{MAC}} \cdot E_{Package} \quad (9)$$

Where E_{ML} and D_{ML} represent the energy consumption and computation demand by operations per communication

round of ML task respectively, F_{DSP} is the frequency of DSP slices, N_{DSP} is the number of onboard DSP slices, and N_{MAC} is the number of multiply-accumulate (MAC) operations each DSP slice can carry out per cycle, and $E_{Package}$ represent typical package energy consumption [21].

IV. EXPERIMENTS

This section details the experiment setup and results addressing the research questions posited earlier, focusing on (i) the efficacy of multi-precision OTA aggregation, and (ii) the formulation of client quantization schemes to optimize performance across both server and client dimensions.

A. Settings

Our experimental framework emulates an OTA-FL system of 15 clients working at designated quantization levels. The model structure is ResNet-50 with ImageNet [22] pre-trained weights initialization. The FL operates over 100 communication rounds with 5-30dB of emulated Gaussian noise.

1) *Data:* We utilize the German Traffic Sign Recognition Benchmark (GTSRB) [23] as the dataset for our experiments. This dataset consists of 39,209 training samples and 12,630 testing samples across 43 traffic sign classes, captured in real-world driving environments. It reflects the variability of real-world conditions, including changes in lighting, weather, perspectives, and occlusions. This diversity ensures the dataset's authenticity and makes it highly suitable for advancing research and development in smart transportation systems. In our experimental setup, each client is assigned an equal subset of the data.

2) *Quantization Schemes:* We assign quantization levels to the 15 clients by a group of 5. Each scheme consists of 3 precision levels, and each precision level is assigned to 5 clients. Quantization levels are chosen from [32, 24, 16, 12, 8, 6, 4].

3) *Performance Metrics:* Evaluation metrics for the server include convergence speed, measured by the number of communication rounds the system took to converge, and final performance of the aggregated model. For clients, we assess their performances after aggregation and re-quantization.

4) *Energy Efficiency:* We compare the energy savings of our mix-precision clients with homogeneous precision clients at standard 32-bit, 16-bit, 8-bit and 4-bit.

B. Results

TABLE II
ESTIMATED ENERGY CONSUMPTION PER SAMPLE FOR RESNET-50
FORWARD PASS AND RELATIVE SAVINGS COMPARED TO 32-BIT

	32-bit	16-bit	12-bit	8-bit	6-bit	4-bit
Energy Cost (J)	0.36	0.17	0.16	0.022	0.021	0.0056
Saving (%)	0	52.58	56.15	93.89	94.17	98.45

1) *Energy Efficiency Estimation:* Based on Eq. 9, we estimate energy consumption of forward passing one sample through a ResNet-50 network, across following quantization levels [32, 24, 16, 12, 8, 6, 4] on 9 platforms of different hardware resources such as logic cells and DSP slices. In Table II,

we present the average energy cost and relative savings to 32-bit of these platforms. Notably, due to under-utilization of hardware, quantizing to 16-bit and 12-bit share very similar degree energy saving, and the same applies to 8-bit and 6-bit. From the table, we can also see the diminishing energy saving gain when further quantizing from low precision like 8-bit to ultra low ones like 4-bit.

2) *Federated Training and Server Performance*: The convergence velocity, as depicted in Fig. 3, indicates that setups of uniform 4-bit clients, or a mixed-precision schema of [12, 4, 4] bits, exhibit slower and more erratic initial convergence, even when the latter has a better random start in training. In contrast, setups incorporating clients with 16-bit precision or higher demonstrate more rapid and stable convergence, achieving approximately 90% accuracy within 10 communication rounds. Notably, for clients of high resource capacity, 32-bit or 24-bit precision only offers marginal training gains compared to 16-bit precision. The server model performance of all quantization schemes reached 97% top-1 accuracy within a tight 0.3% margin after 100 communication rounds, underscoring the effectiveness of the federated learning framework in achieving high accuracy with mixed-precision clients.

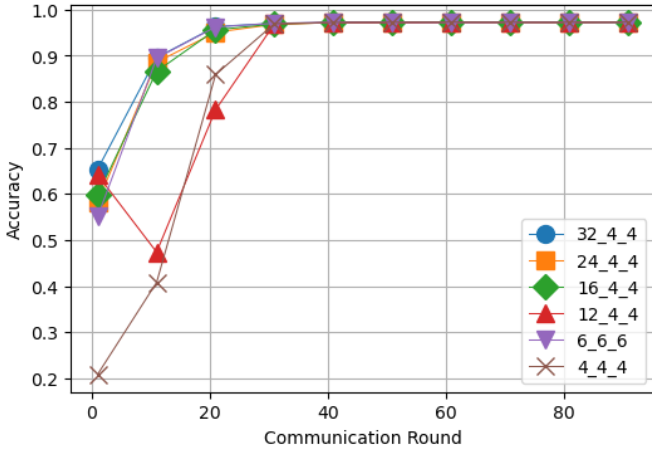


Fig. 3. Training accuracy in 100 communication rounds, with ImageNet pre-trained weights initialization, scheme [4, 4, 4] (denoted by brown ‘X’), and scheme [12, 4, 4] (denoted by red upright triangle) converge significantly slower than other schemes, even when the latter one has a better random start.

3) *Client Performance*: After 100 rounds, the final global model, θ^{100} , is broadcast to clients. Here, we focus on the clients at the lowest precision, 4-bit, for higher counterparts have better performance and minor degradation from well converged global model as illustrated in Table I and Fig. 3.

As shown in Fig. 4, in comparison to FL systems of homogeneous clients at 32-bit and 16-bit, based on our estimation in Table II, our FL with mixed-precision clients models can save over 65% and 13% of energy consumption respectively, while gaining more than 10% in accuracy on clients at 4-bit. Notably, for those under schemes incorporating 16-bit precision or higher, when re-quantized for 4-bit clients, attain around 5% higher accuracy, and this performance boost for lower precision

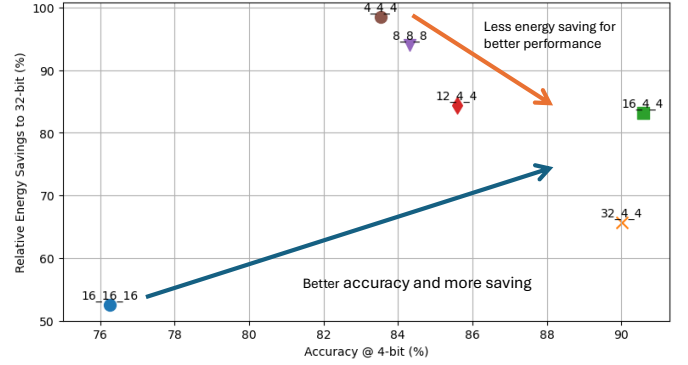


Fig. 4. Trade-offs between accuracy of model quantized to 4-bit and energy savings in comparison to homogeneous 32-bit and 16-bit clients, schemes near bottom right corner presents superior trade-off towards accuracy.

clients from higher precision counterparts shows diminishing returns beyond 16-bit precision. While comparing to FL of homogeneous clients at 8-bit and 4-bit, our mixed-precision FL can trade mere 10% of energy savings for 5% of accuracy.

V. CONCLUSION

In this study, we introduce a framework for Over-the-Air Federated Learning (OTA-FL) that incorporates Approximate Computing (AxC) to accommodate clients of multiple precision levels. Our novel mixed-precision OTA aggregation mechanism enables the enhancement of overall performance, computational and energy efficiency within federated learning systems, especially for those of ultra low precision. Our framework is universally applicable and compatible with existing FL systems, unveiling the huge energy saving potential of incorporating clients at ultra low precision while having their performance improved. These advantages of leveraging multi-precision client configurations in OTA-FL systems, in both performance and energy savings, are particularly valuable in resource-diverse and heterogeneous edge computing environments. This study may serve as a foundational guideline for the architectural design and optimization of future green and sustainable multi-precision OTA-FL systems, at the cost of a less efficient mixed-precision modulation scheme for OTA aggregation. In addition to FL systems, the mixed-precision modulation scheme can also facilitate other distributed computation applications tailored for wireless networks.

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 *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, vol. 54, 2017, pp. 1273–1282.
- [2] H. Chen, H. Wang, Q. Long, D. Jin, and Y. Li, “Advancements in federated learning: Models, methods, and privacy,” *ACM Comput. Surv.*, vol. 57, no. 2, Nov. 2024.
- [3] K. Yang, T. Jiang, Y. Shi, and Z. Ding, “Federated learning via over-the-air computation,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, 2020.
- [4] A. El Ouadrhiri and A. Abdelhadi, “Differential privacy for deep and federated learning: A survey,” *IEEE Access*, vol. 10, 2022.

- [5] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3796–3811, 2021.
- [6] M. Kim, A. L. Swindlehurst, and D. Park, "Beamforming vector design and device selection in over-the-air federated learning," *IEEE Transactions on Wireless Communications*, 2023.
- [7] J. Zheng, H. Tian, W. Ni, W. Ni, and P. Zhang, "Balancing accuracy and integrity for reconfigurable intelligent surface-aided over-the-air federated learning," *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10964–10980, 2022.
- [8] 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.
- [9] H. Chen, H. Wang, Q. Yao, Y. Li, D. Jin, and Q. Yang, "Losac: An efficient local stochastic average control method for federated optimization," *ACM Transactions on Knowledge Discovery from Data*, vol. 17, no. 4, pp. 1–28, 2023.
- [10] C. Zhang, P. Li, G. Sun, Y. Guan, B. Xiao, and J. Cong, "Optimizing fpga-based accelerator design for deep convolutional neural networks," in *Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, 2015, p. 161–170.
- [11] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *Journal of Machine Learning Research*, vol. 20, no. 55, 2019.
- [12] R. Lee, J. Fernandez-Marques, S. X. Hu, D. Li, S. Laskaridis, Ł. Dudziak, T. Hospedales, F. Huszár, and N. D. Lane, "Recurrent early exits for federated learning with heterogeneous clients," *arXiv preprint arXiv:2405.14791*, 2024.
- [13] G. Flegar, F. Scheidegger, V. Novaković, G. Mariani, A. E. Tomás, A. C. I. Malossi, and E. S. Quintana-Ortí, "Floatx: A c++ library for customized floating-point arithmetic," *ACM Transactions on Mathematical Software*, vol. 45, no. 4, dec 2019.
- [14] S. Mittal, "A survey of techniques for architecting and managing asymmetric multicore processors," *ACM Computing Surveys (CSUR)*, vol. 48, no. 3, pp. 1–38, 2016.
- [15] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Transactions on information theory*, vol. 53, no. 10, 2007.
- [16] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep learning with limited numerical precision," in *International conference on machine learning*. PMLR, 2015, pp. 1737–1746.
- [17] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016.
- [18] X. Sun, N. Wang, C.-Y. Chen, J. Ni, A. Agrawal, X. Cui, S. Venkataramani, K. El Maghraoui, V. V. Srinivasan, and K. Gopalakrishnan, "Ultra-low precision 4-bit training of deep neural networks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020.
- [19] AMD, "Xilinx power estimator (xpe)," 2024. [Online]. Available: <https://www.xilinx.com/products/technology/power/xpe.html>
- [20] AMD, "Virtex ultrascale+ fpga data sheet (ds923)," 2021. [Online]. Available: <https://docs.amd.com/v/u/en-US/ds923-virtex-ultrascale-plus>
- [21] R. B. Abdelhamid, G. Kuwazawa, and Y. Yamaguchi, "Quantitative study of floating-point precision on modern fpgas," in *Proceedings of the 13th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies*, 2023, pp. 49–58.
- [22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.
- [23] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, no. 0, pp. –, 2012.