



HAL
open science

Compression Study on Federated Learning

Lucas Grativol Ribeiro, Mathieu Leonardon, Guillaume Muller, Fresse, Virginie, Matthieu Arzel

► **To cite this version:**

Lucas Grativol Ribeiro, Mathieu Leonardon, Guillaume Muller, Fresse, Virginie, Matthieu Arzel. Compression Study on Federated Learning. Assemblée Générale 2023 du GdR ISIS, May 2023, Lyon, France. hal-04251991

HAL Id: hal-04251991

<https://hal.science/hal-04251991>

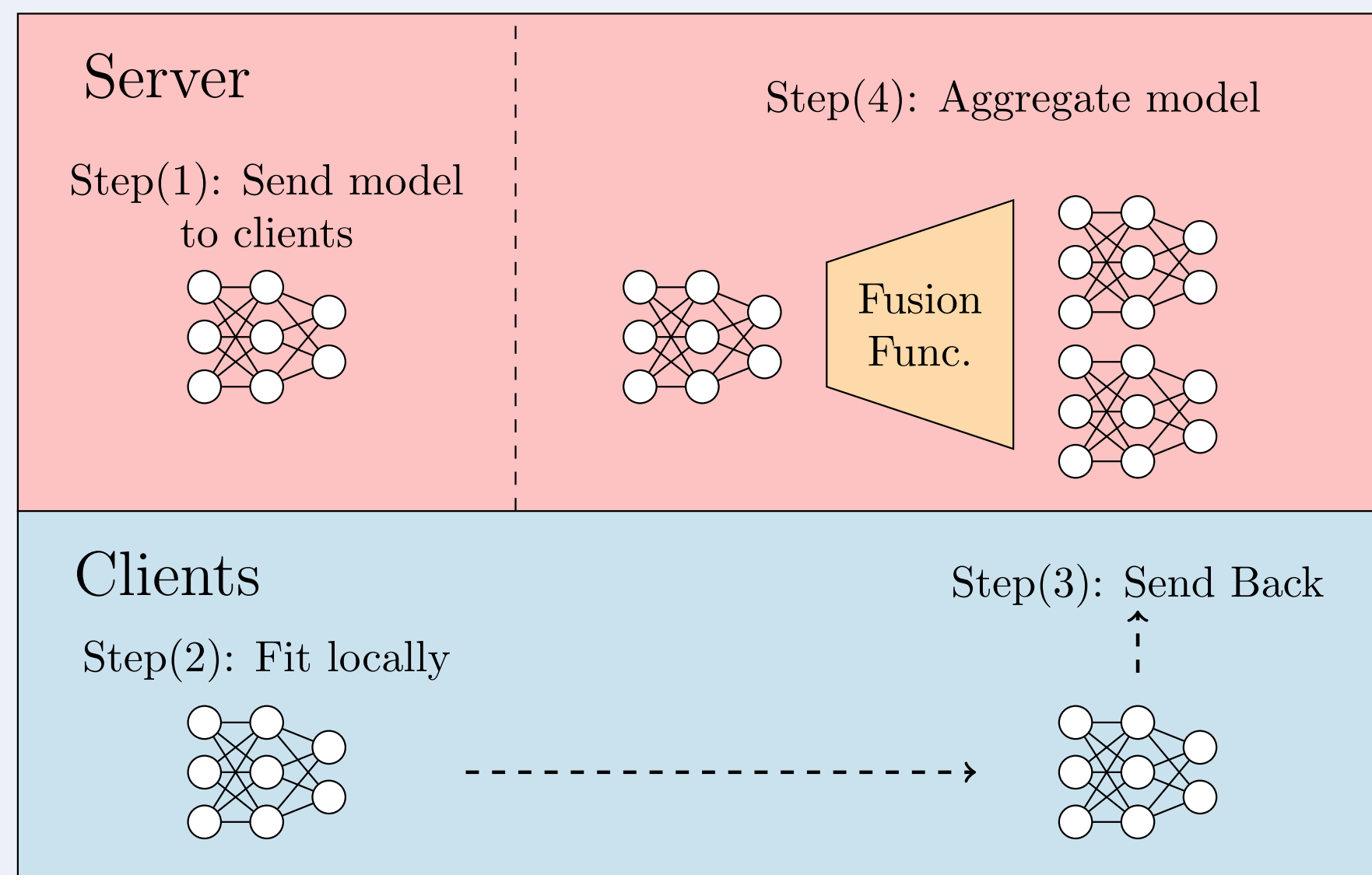
Submitted on 20 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Context

Federated Learning (FL) is used distributed/collaborative learning algorithm, for a more private machine learning.



Federated Learning standard setup.

Clients are usually embedded devices, with low hardware specifications but access to real-world, possibly private, data. In this scenario, **how can neural networks compression techniques improve the FL framework?**

Setup

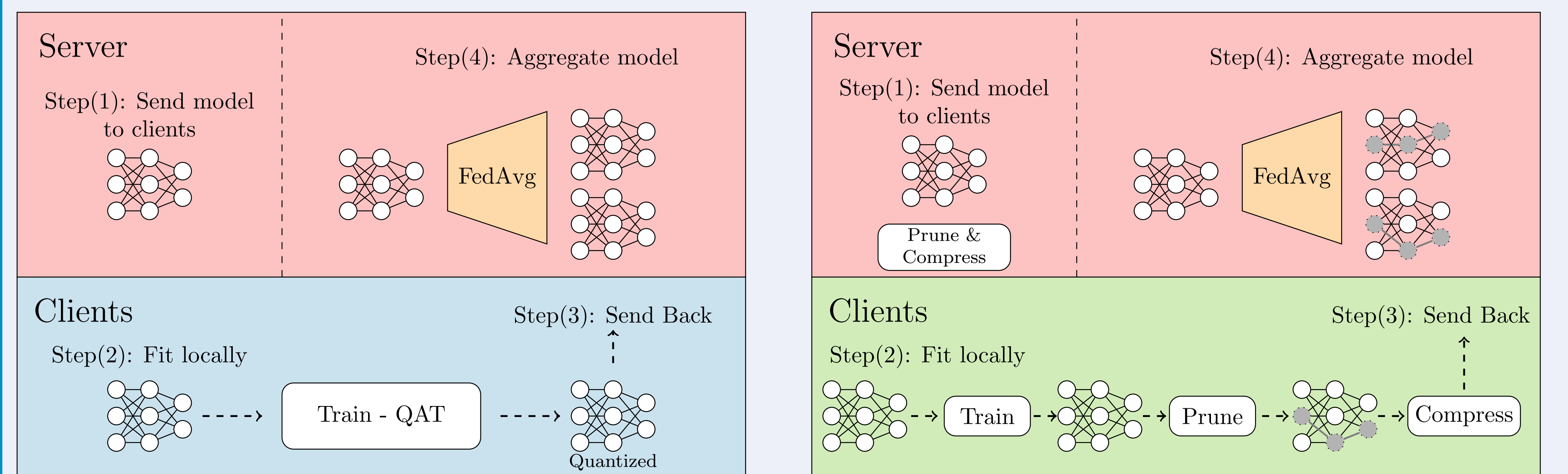
FL Setup: We used Flower's framework [1] to simulate 10 FL clients, with 40% sampling rate, for 100 rounds of communication. The client's learning rate was fixed to 0.0316 and a batch-size of 20, as per [2]. On the server side, we use FedAvg as aggregation strategy.

Simulation: The experiments are done for image classification on CIFAR-10 and CIFAR-100, where each client holds a fraction of the trainset, accordingly to a Latent Dirichlet Allocation [3] with parameter equals to 100. Each client trains a ResNet-12, with GroupNorm instead of a BatchNorm.

Quantization: We chose to work with quantization levels that are easily exploitable on embedded systems, 1-bit, 4-bit and 8-bit (weights only). For 4- and 8-bit, we use the brevitas framework [4] to implement QAT.

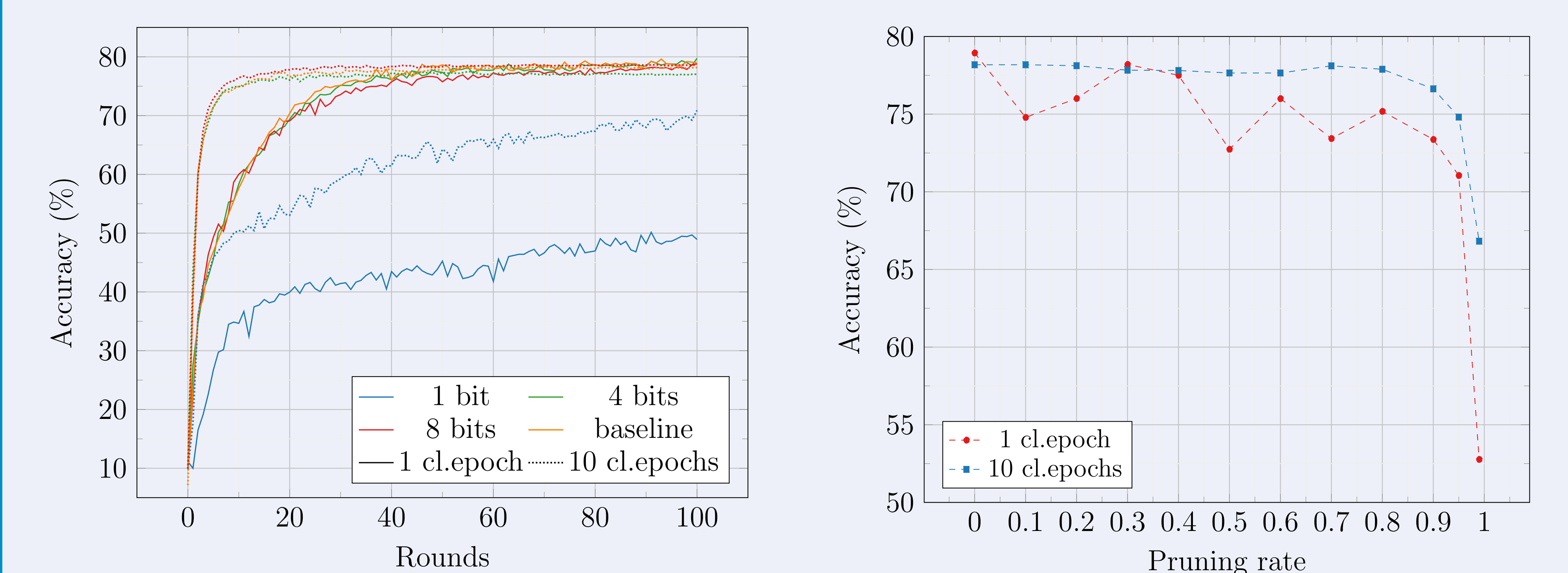
Pruning: We considered an unstructured magnitude based pruning method. To keep coherence between up and down streams, server and clients prune the same percentage of weights each round.

Our scheme



Proposed setups. On the left, each client trains a model using QAT (Quantization-Aware-Training). On the right, both client and server use magnitude prune to compress the model.

Experimental results – CIFAR10



Accuracy evolution comparison, for the quantization technique, between baseline (FP 32-bit), 1-bit, 4-bit and 8-bit, on the left. Pruning effect on the accuracy in function of the pruning rate, where the rate indicates the % of total parameters pruned, on the right. Experiments done for 1 and 10 cl.epochs (clients epochs)

Compression expectation

Technique	Accuracy (%)		Message (MiB)
	1 Lc. Epoch	10 Lc. Epochs	
Baseline	78.94	78.18	2.97
Pruning			
10 %	74.79	78.18	2.57
20 %	76.01	78.12	2.34
30 %	78.20	77.83	2.10
40 %	77.50	77.81	1.85
50 %	72.74	77.65	1.57
60 %	76.00	77.65	1.29
70 %	73.43	78.11	1.01
80 %	75.18	77.89	0.70
90 %	73.37	76.63	0.37
95 %	71.05	74.81	0.19
99 %	52.77	66.82	0.04
Quantization			
8-bits	78.80	78.58	0.75
4-bits	79.74	77.04	0.38
1-bit	48.93	70.89	0.10

Summary of message size and accuracy for the CIFAR-10 dataset.

Through the integration of common compression techniques to FL, where, depending on the compression used, it is possible to reduce the size of the message to be sent from 13% (10% pruning) to 96% (99%/1-bit pruning), after data compression. Where the trade-off between accuracy and message size is the main point.

Conclusions

- Takeaways:** As seen in centralized training, reduced precision can achieve comparable accuracy to an FP32 model. Furthermore, in the quantization experiment, reaching 78% accuracy takes approximately 40 rounds with 1 local epoch, whereas with 10, it is achieved in just 10 rounds. By increasing the number of local epochs, it is possible to reduce communication, but at the same time, it increases device training time/energy.
- Perspectives:** In order to go further with the adaptation of hardware constraints and the machine learning model, it could be interesting to integrate heterogeneous Federated Learning.

References

- [1] Beutel, Daniel J and et al.: *Flower: A friendly federated learning research framework*, arXiv (2020)
- [2] Reddi, Sashank and et al.: *Adaptive federated optimization*, ICLR (2021)
- [3] Hsu, Tzu-Ming Harry and et al.: *Measuring the effects of non-identical data distribution for federated visual classification*, Neurips Workshop on Federated Learning (2019)
- [4] Alessandro Pappalardo.: *Xilinx/brevitas*.