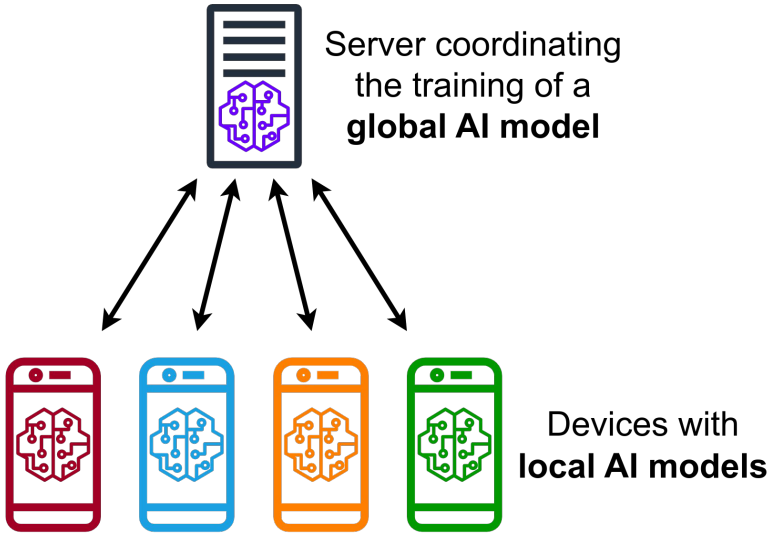
Analysis of Error Feedback in Federated Non-Convex Optimization with Biased Compression: Fast Convergence and Partial Participation

Background

Federated learning (FL) has been an important topic in ML and has seen many applications in 5G/6G wireless communications, Internet of Things (IoT), financial fraud detection, input method editor (IME), advertising (ads), health records, …

In this paper, we consider a standard centralized FL system, with a global server and many local clients (data silos, mobile phones, IoT devices).

In each round, the clients train the models locally, and send back the model updates to the server for aggregation.



Challenges of FL

Three key challenges in FL algorithm design, theory, and deployment:

1. **Communication cost**: Limited wireless bandwidth often cannot afford transmitting full-precision large models.

2. **Data heterogeneity**: Local clients' data are non-iid. Thus, the local training loss (expectation over local data distribution) are different from the global training loss.

3. **Partial participation**: In cross-device FL, clients may drop and join in each round, thus partially participating in FL training.

Reducing the Communication Overhead

To reduce the communication cost, two main categories of strategies are:

1. **Local steps:** we allow clients to run local training for K steps before aggregation, thus reducing the number of communication rounds.
2. **Communication compression:** we compress the local model updates transmitted between the server and the clients.

Common choices of compressor C(x):

- **Unbiased compressor**: E(C(x)) = x
  - **Stochastic quantization, stochastic sparsification …**
  - Can usually be used in place of the full-precision gradients or model updates
- **Biased compressor**: E(C(x)) ≠ x, with bounded deviation $\|\mathcal{C}(x) - x\|^2 \leq q^2\|x\|^2$
  - **Topk, Random-k, sign-SGD, fix quantization …**
  - Need to be applied with an error correction scheme called **Error Feedback (EF) (or variants)**

Our Contributions

Error Feedback (Seide et al. 2014; Stich et al. 2018) has not been studied under the practical FL setting, thoroughly.

- We focus on the **Fed-EF** framework and provide the analysis of EF with **local steps, data heterogeneity, and communication compression**, to achieve a sharp convergence rate compared with state-of-the-art FL methods.
- We propose **Fed-EF-AMS**, the first adaptive (Adam-type) FL algorithm with communication compression.
- We develop the analysis of EF under **partial participation**, showing an extra slow down factor which is related to the client sampling ratio.

[1] **Seide et al.,** 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs, *INTERSPEECH* 2014

[2] **Stich et al.,** Sparsified SGD with memory, *NeurIPS* 2018

Fed-EF Algorithm

Fed-EF-SGD: The server performs SGD updates

Fed-EF-AMS: The server performs AMSGrad (Reddi et al. 2019) updates

For distributed gradient compression with adaptive optimizers, see Li et al. 2022.

[1] **Reddi et al.,** On the convergence of Adam and Beyond, *ICLR* 2019

[2] **Li et al.,** On distributed adaptive optimization with gradient compression, *ICLR* 2022

Algorithm 1 Fed-EF: Compressed FL with Error Feedback

Input: learning rates η, η_l; parameters $\beta_1, \beta_2, \epsilon$

Initialize: global model $\theta_1 \in \mathbb{R}^d \subseteq \mathbb{R}^d$; local error accumulator $e_{1,i} = 0$; $m_0 = 0, v_0 = 0, \hat{v}_0 = 0$

for $t = 1, \ldots, T$ **do**

  parallel for worker $i \in [n]$ **do:**

    Receive global model θ_t from server, set $\theta_{t,i}^{(1)} = \theta_t$

    for $k = 1, \ldots, K$ **do**

      Compute stochastic gradient $g_{t,i}^{(k)}$ at $\theta_{t,i}^{(k)}$

      Local update $\theta_{t,i}^{(k+1)} = \theta_{t,i}^{(k)} - \eta_l g_{t,i}^{(k)}$

    end for

    Compute local update $\Delta_{t,i} = \theta_t - \theta_{t,i}^{(K+1)}$

    Send $\widetilde{\Delta}_{t,i} = \mathcal{C}(\Delta_{t,i} + e_{t,i})$ to server

    Update the error $e_{t+1,i} = e_{t,i} + \Delta_{t,i} - \widetilde{\Delta}_{t,i}$

  end parallel

  Central server do:

  Global aggregation $\overline{\widetilde{\Delta}}_t = \frac{1}{n}\sum_{i=1}^n \widetilde{\Delta}_{t,i}$

  Global update $\theta_{t+1} = \theta_t - \eta\overline{\widetilde{\Delta}}_t$ { Fed-EF-SGD }

  $m_t = \beta_1 m_{t-1} + (1-\beta_1)\overline{\widetilde{\Delta}}_t$ { Fed-EF-AMS }

  $v_t = \beta_2 v_{t-1} + (1-\beta_2)\overline{\widetilde{\Delta}}_t^2, \quad \hat{v}_t = \max(v_t, \hat{v}_{t-1})$

  Global update $\theta_{t+1} = \theta_t - \eta\frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}$

end for

Convergence Rates

Contrastive compressor: $\|\mathcal{C}(x) - x\|^2 \leq q^2\|x\|^2$, n: # of clients m: # of active clients

- Fed-SGD with biased compression, **without EF**: $O(\frac{1+q^2}{\sqrt{TKn}} + q^2 \cdot const)$
- **Fed-EF-SGD and Fed-EF-AMS (full participation):**

$$O(\frac{1+q^2}{\sqrt{TKn}}) \Longrightarrow$$ Matches full-precision rates when q=0 (no compression)

- Fed-EF under **partial client participation (uniform sampling assumption)**

$$O(\frac{\sqrt{n}}{\sqrt{m}}\frac{(1+q^2)\sqrt{K}}{\sqrt{Tm}})$$

The full-precision rate under PP is $O(\frac{\sqrt{K}}{\sqrt{Tm}})$ [Yang et al. ICLR'21]

An extra slow-down factor $\frac{\sqrt{n}}{\sqrt{m}}$: **"delayed error compensation"**

Partial participation introduces staleness to the local error accumulator. Updating with the stale information slows down convergence.

Communication Complexity

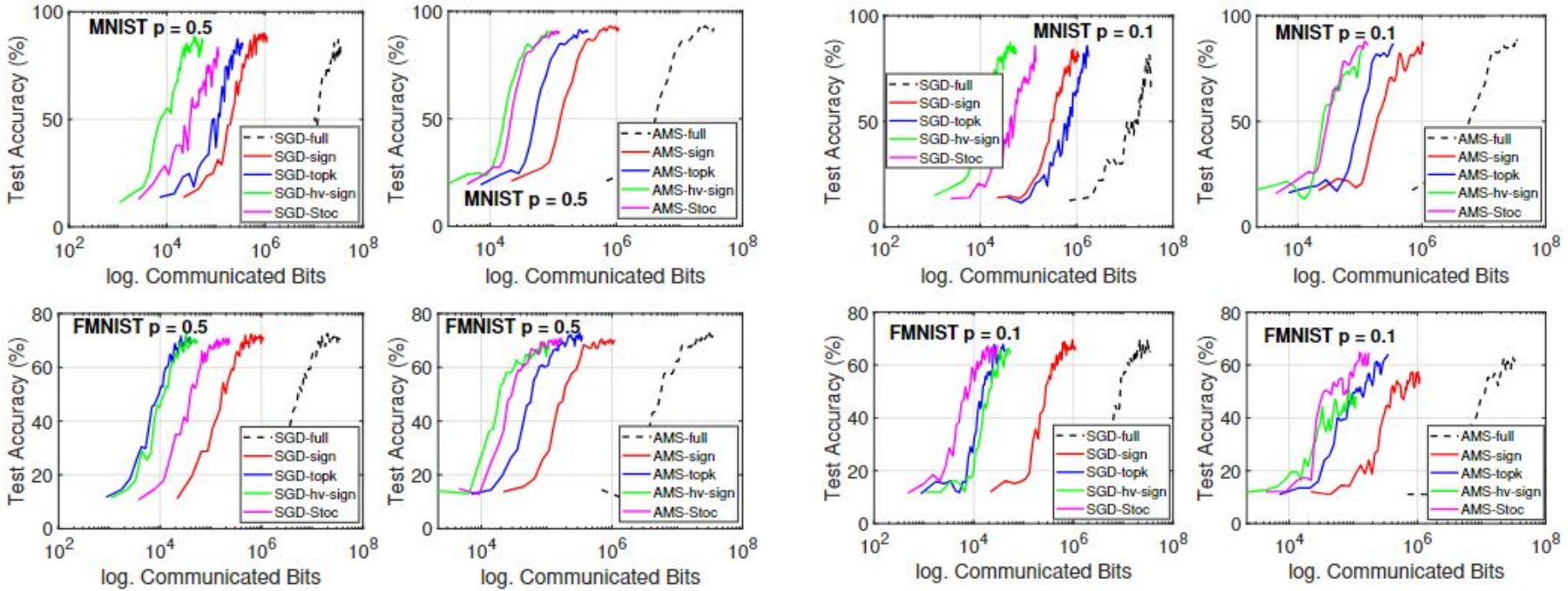
Table 1. Summary of theoretical convergence results from some existing works on distributed and federated learning with communication compression for non-convex optimization. "PP" stands for "partial participation", and "# of Rounds" is the number of communication rounds required to achieve linear speedup, which is a common measure of the communication complexity of FL algorithms. T is the number of communication rounds, K is the number of local steps, and n is the total number of clients.

Reference	Local Step	Non-iid Data	PP	Adaptive Opt.	Compression	# of Rounds
Jiang and Agrawal (2018) [a]		✓			Unbiased	$T = \mathcal{O}(n)$
Li et al. (2022b)		✓		✓	Biased + EF	$T = \mathcal{O}(n^3)$
Reisizadeh et al. (2020)	✓				Unbiased	– [b]
Haddadpour et al. (2021)	✓	✓			Unbiased	$T = \mathcal{O}(Kn)$
Basu et al. (2019)	✓				Biased + EF	$T = \mathcal{O}(K^3n^3)$
Gao et al. (2021)	✓				Biased + EF	$T = \mathcal{O}(Kn^3)$
Fed-EF (our paper) [c]	✓	✓	✓	✓	Biased + EF	$\boldsymbol{T = \mathcal{O}(Kn)}$

Our algorithm and analysis covers local steps, data heterogeneity, partial participation and adaptive optimizer.

Experiments

p = m/n is the client participation rate



Fed-EF matches the performance of full-precision training with substantially reduced communication cost (30 - 100x)

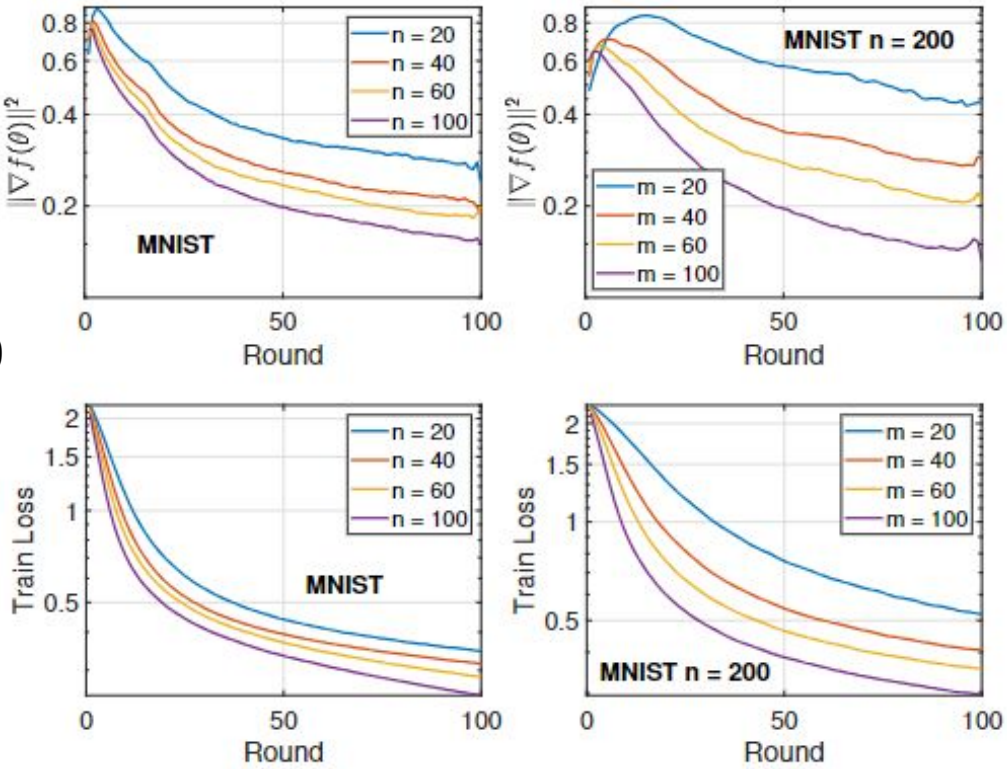
Speedup

Fed-EF with Topk-0.01 compression

Left: full participation, n = 20, 40, 60, 100

Right: Partial participation, n = 200

m = 20, 40, 60, 100



Linear speedup with *n* in full participation case

Faster speedup with *m* in partial participation (validating the extra $\frac{\sqrt{n}}{\sqrt{m}}$ factor)

Our prior works on distributed, adaptive, or federated optimization/learning

ICML'23, Analysis of Error Feedback in Federated Non-Convex Optimization with Biased Compression
JMLR'23, Sharper Analysis for Minibatch Stochastic Proximal Point Method: Stability, Smoothness, and Deviation
UAI'23, Fed-LAMB: Layer-wise and Dimension-wise Locally Adaptive Federated Learning
ICLR'23, Improved Convergence of Differential Private SGD with Gradient Clipping
ICLR'22, On Distributed Adaptive Optimization with Gradient Compression
NeurIPS'22, On Convergence of FedProx: Local Dissimilarity Invariant Bounds, Non-smoothness and Beyond
BIGDATA'22, Communication-Efficient TeraByte-Scale Model Training Framework for Online Advertising
ACML'22, On the Convergence of Decentralized Adaptive Gradient Methods
ACML'21, An Optimistic Acceleration of AMSGrad for Nonconvex Optimization
FODS'20, Toward Communication Efficient Adaptive Gradient Method
JMLR'20, On Convergence of Distributed Approximate Newton Methods: Globalization, Sharper Bounds and
NeurIPS'20, Towards Better Generalization of Adaptive Gradient Methods
MLSys'20, Distributed Hierarchical GPU Parameter Server for Massive Scale Deep Learning Ads Systems
CIKM'19, AIBox: CTR Prediction Model Training on a Single Node

Thank you!