Communication-Efficient Adaptive Federated Learning

Yujia Wang[1] Lu Lin[2] Jinghui Chen[1]

[1]Penn State University
[2]University of Virginia

Federated Learning: Background

- Federated learning has recently become a popular machine learning training paradigm that enables multiple clients to jointly learn a machine learning model without sharing their own data.
- Federated learning can be formulized as the following nonconvex optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} = \underbrace{\mathbb{E}_{\xi^{(i)}\sim\mathcal{D}_i} F_i(\mathbf{x};\xi^{(i)})}_{:=F_i(\mathbf{x})},$$

where we assume $F_i(\mathbf{x})$ is the local *nonconvex* loss function on worker i.

Federated Learning: Optimization

- FedAvg (McMahan et al., 2017) is a commonly used optimization approach to solve the former optimization problem:

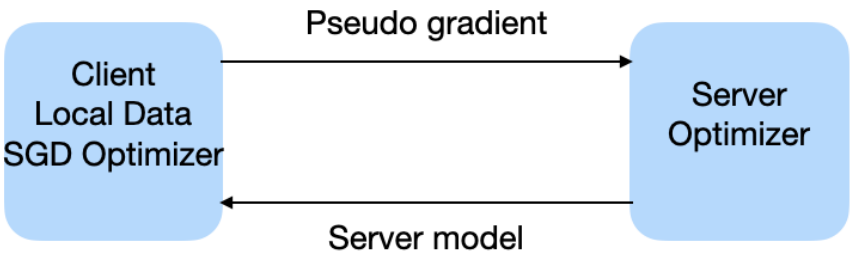
Client i: receives $\mathbf{x}_t$,
updates $\mathbf{x}_{t,K}^i$ by K steps of SGD
with learning rateη_l,
obtains $\Delta_t^i = \mathbf{x}_{t,K}^i - \mathbf{x}_t$.

Δ_t^i →

← $\mathbf{x}_{t+1}$

Server: receives $\Delta_t^i, i \in [1, m]$,
updates $\Delta_t = \frac{1}{m}\sum_{i=1}^m \Delta_t^i$,
updates $\mathbf{x}_{t+1} = \mathbf{x}_t + \Delta_t$.

- The key idea of federated optimization is to use the *pseudo gradient* Δ_t^i for aggregation and update.



Federated Learning: Challenges

- Large communication overhead
- Lack of adaptivity
- Various attempts have been made to solve the challenges individually or in pairs:
  - quantization or compression strategies
  - partial participation in training rounds
  - adaptive federated optimization

Federated Learning: Challenges

✔ Quantization or Compression: clients send $\mathcal{Q}(\Delta_t^i)$ or $\mathcal{C}(\Delta_t^i)$ instead of Δ_t^i to the server.

✔ Partial participation: allows part of the clients (usually a small amount) to participate training process in each round.

✔ Adaptive federated optimization (e.g., FedAdam): the server updates global model using Δ_t via adaptive (e.g., Adam) optimizer:

$$\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1-\beta_1)\Delta_t, \mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1-\beta_2)\Delta_t^2,$$
$$\mathbf{x}_{t+1} = \mathbf{x}_t + \eta \frac{\mathbf{m}_t}{\sqrt{\mathbf{v}_t} + \epsilon}.$$

✖ Partial participation settings in adaptive federated optimization.

✖ Quantization or compression with adaptive federated optimization.

Motivations and Contributions

Can we simultaneously overcome the challenges through *various aspects*, i.e., achieve *communication-efficient adaptive federated optimization* with rigorous convergence guarantees?

In this work, we will show that both the **gradient compression** and the **partial participation** can be applied to **adaptive federated optimization** for overcoming the existing challenges in federated learning.

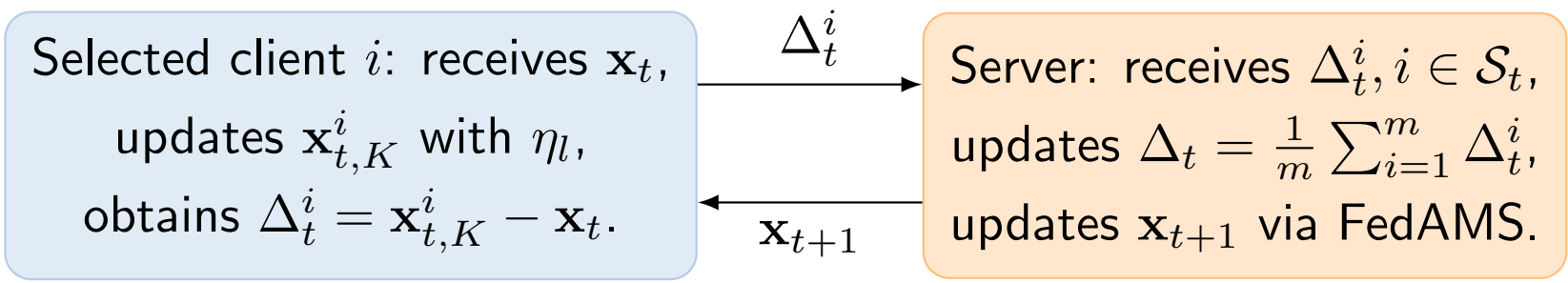
FedAMS: Federated AMSGrad with Max Stablization

- The FedAMS framework follows the same momentum $\mathbf{m}_t$ and variance $\mathbf{v}_t$ update rule as FedAdam. In addition, FedAMS provides two options for max stabilization:

$$\text{Option 1: } \widehat{\mathbf{v}}_t = \max(\widehat{\mathbf{v}}_{t-1}, \mathbf{v}_t, \epsilon), \mathbf{x}_{t+1} = \mathbf{x}_t + \eta \frac{\mathbf{m}_t}{\sqrt{\widehat{\mathbf{v}}_t}}.$$

$$\text{Option 2: } \widehat{\mathbf{v}}_t = \max(\widehat{\mathbf{v}}_{t-1}, \mathbf{v}_t), \mathbf{x}_{t+1} = \mathbf{x}_t + \eta \frac{\mathbf{m}_t}{\sqrt{\widehat{\mathbf{v}}_t} + \epsilon}.$$

- FedAMS supports partial participation settings: only a subset $\mathcal{S}_t$ of clients participate in the tth round.



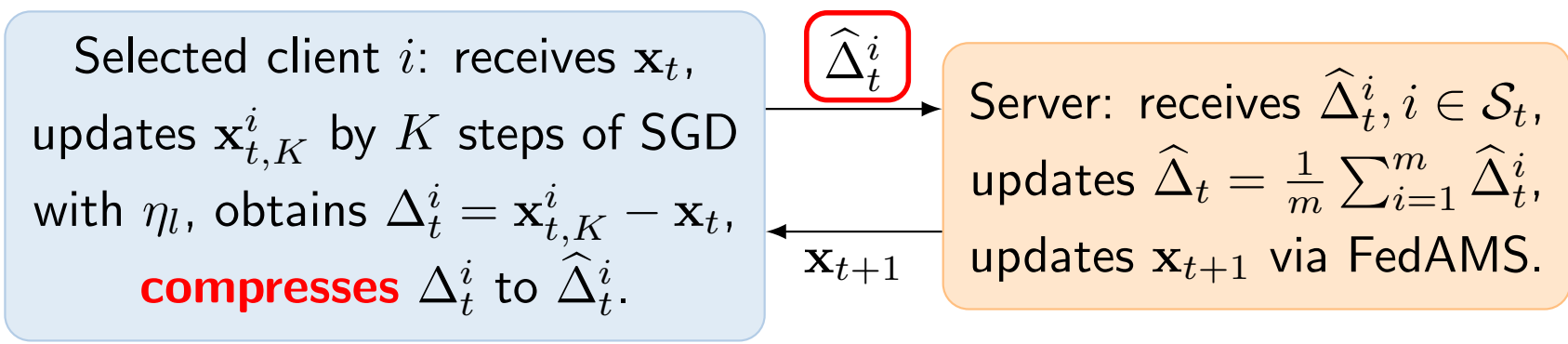
FedCAMS: Federated Communication-Compressed AMSGrad

- FedCAMS compresses the model difference Δ_t^i to $\widehat{\Delta}_t^i$ via compression strategy with error feedback:

$$\widehat{\Delta}_t^i = \mathcal{C}(\Delta_t^i + \mathbf{e}_t^i); \mathbf{e}_{t+1}^i = \Delta_t^i + \mathbf{e}_t^i - \widehat{\Delta}_t^i.$$

  Each client sends the **compressed** $\widehat{\Delta}_t^i$ to the server.

- FedCAMS is also compatible with partial participation settings by keeping the stale cumulative compression error for inactive clients.



Convergence Analysis: Assumptions

- **Smoothness** For any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, assume that

$$\left| f_i(\mathbf{x}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \right| \leq \frac{L}{2} \|\mathbf{x} - \mathbf{y}\|_2^2.$$

- **Bounded Gradient** For any $\mathbf{x} \in \mathbb{R}^d$ and any ξ, assume that

$$\|\nabla f_i(\mathbf{x}; \xi)\|_2 \leq G, \qquad \|\nabla f_i(\mathbf{x}; \xi)\|_\infty \leq G_\infty.$$

- **Bounded Variances** For any $\mathbf{x} \in \mathbb{R}^d$, assume that $\mathbf{g}^{(i)} = \nabla f_i(\mathbf{x}, \xi^{(i)})$ has a bounded variance, i.e.,

$$\mathbb{E}_{\xi^{(i)} \sim \mathcal{D}_i} \left\| \nabla f_i(\mathbf{x}, \xi^{(i)}) - \nabla f_i(\mathbf{x}) \right\|^2 \leq \sigma^2;$$

we also assume the loss function has a global variance bound:

$$\frac{1}{m} \sum_{i=1}^{m} \|\nabla F_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \leq \sigma_g^2.$$

Convergence Analysis for FedAMS

- Under the aforementioned assumptions, by choosing $\eta = \Theta(\sqrt{Km})$ and $\eta_l = \Theta(\frac{1}{\sqrt{TK}})$, the convergence rate for FedAMS under *full participation* settings satisfies

$$\min_{t\in[T]} \mathbb{E}\big[\|\nabla f(\mathbf{x}_t)\|^2\big] = \mathcal{O}\bigg(\frac{1}{\sqrt{TKm}}\bigg),$$

  where T is the total iterations, K denotes the number of local updates, and m denotes the number of workers.

- This matches the result for general federated nonconvex optimization methods such as FedAdam and SCAFFOLD.

Convergence Analysis for FedAMS

- Under the aforementioned assumptions, by choosing $\eta = \Theta(\sqrt{Kn})$ and $\eta_l = \Theta(\frac{1}{\sqrt{TK}})$, the convergence rate for FedAMS under *partial participation* settings (only n of m clients participate in each round) satisfies

$$\min_{t\in[T]} \mathbb{E}\left[\|\nabla f(\mathbf{x}_t)\|^2\right] = \mathcal{O}\left(\frac{\sqrt{K}}{\sqrt{Tn}}\right).$$

- This convergence rate is consistent with the partial participation result of FedAvg in the non i.i.d. case.

Convergence Analysis: Assumptions (FedCAMS only)

Besides assumptions of **Smoothness**, **Bounded Gradient** and **Bounded Variances**, FedCAMS needs one more assumption for the compressor $\mathcal{C}(\cdot)$.

- **Biased Compressor** Consider a biased compressor $\mathcal{C} : \mathbb{R}^d \to \mathbb{R}^d$, there exists constant $0 \leq q \leq 1$ such that

$$\mathbb{E}[\|\mathcal{C}(\mathbf{x}) - \mathbf{x}\|] \leq q\|\mathbf{x}\|, \forall \mathbf{x} \in \mathbb{R}^d.$$

Several widely used compressors satisfying this assumption such as the scaled-sign compressor and the top-k compressor.

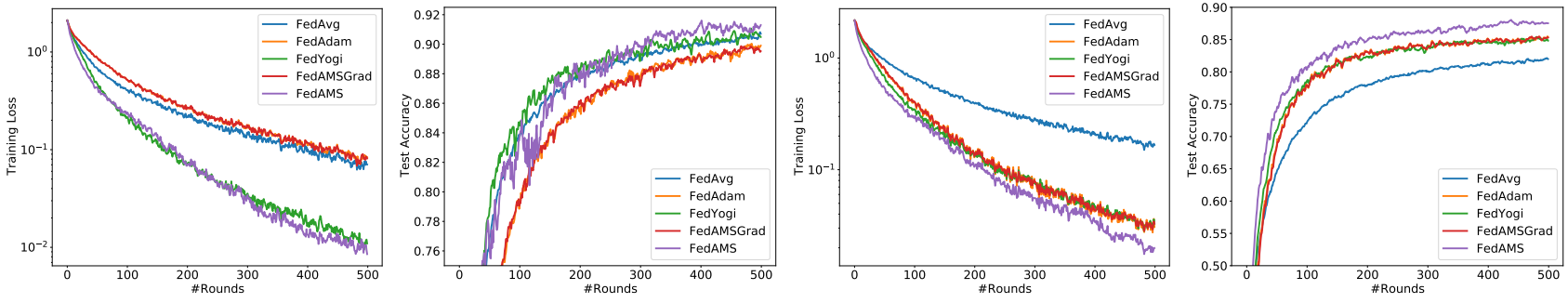
Convergence Analysis for FedCAMS

- Under the aforementioned assumptions, by choosing $\eta = \Theta(\sqrt{Km})$ and $\eta_l = \Theta(\frac{1-q}{\sqrt{TK}})$, the convergence rate for FedCAMS under *full participation* settings satisfies

$$\min_{t\in[T]} \mathbb{E}\left[\|\nabla f(\mathbf{x}_t)\|^2\right] = \mathcal{O}\left(\frac{1}{(1-q)\sqrt{TKm}}\right).$$

- This result matches the rate for the uncompressed counterpart, FedAMS, w.r.t. T, K, m
- A larger q ($q \to 1$) corresponds to a stronger compression, leading to a worse convergence due to heavier information loss.

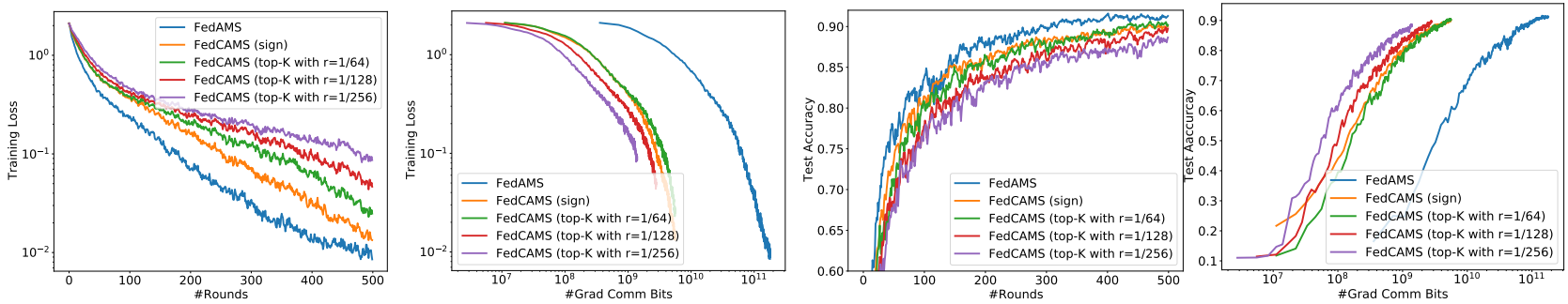
Experiments



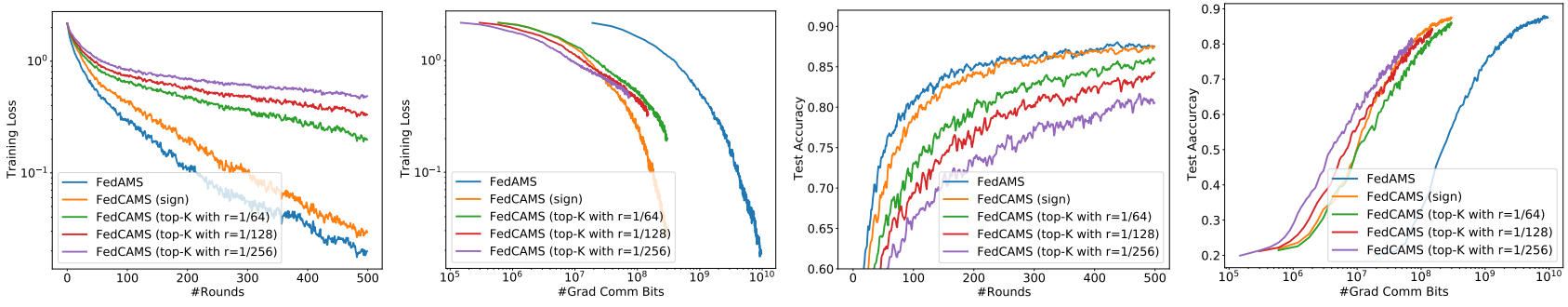
(a) ResNet-18 (b) ConvMixer-256-8

The learning curves for FedAMS and other federated learning baselines on training CIFAR-10 data, (a) shows the results for the ResNet-18 model and (b) shows the results for the ConvMixer-256-8 model. We denote FedAMS for Option 1 and FedAMSGrad for Option 2 in the proposed FedAMS framework.

Experiments



(a) ResNet-18 model



(b) ConvMixer-256-8 model

The learning curves for FedCAMS and uncompressed FedAMS on training CIFAR-10 data on (a) ResNet-18 model and (b) ConvMixer-256-8 model.

Thank you

References I

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.