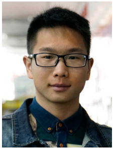
Anarchic Federated Learning

Jia (Kevin) Liu

Assistant Professor
Dept. of Electrical and Computer Engineering
The Ohio State University
Columbus, OH, USA

Joint Work



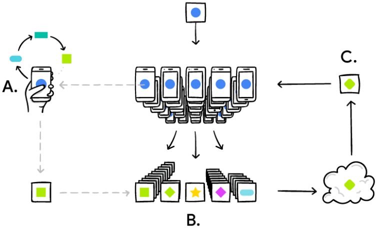
Haibo Yang
OSU

Xin Zhang
ISU/Meta

Prashant Khanduri
OSU/UMN

Jia (Kevin) Liu
OSU

From Distributed Learning to Federated Learning

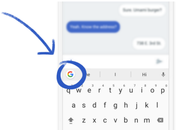


Distributed Learning
Parallelism

→

Federated Learning
Parallelism + Data Privacy + ...

Applications of Federated Learning



Google Gboard

Apple QuickType

Apple “Hey Siri”

- Google: Use FL in Gboard mobile keyboard, featured in Pixel phones, and Android Messages
- Apple: Use FL in QuickType keyboard next word prediction and vocal classifier for “Hey Siri”
- doc.ai uses FL for medical research, Snips uses FL for hotword detection, etc.

Federated Learning vs. Distributed Learning

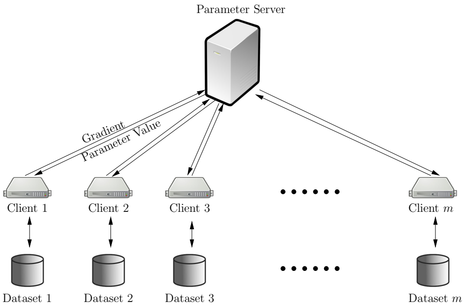
Distributed Learning	Cross-Device FL	Cross-Silo FL
IID dataset	Non-IID dataset	Non-IID dataset
Fast wired communication	Slow wireless communication	Fast communication
Centrally orchestrated	Flexible participation	Centrally orchestrated
Small scale (1 - 1000)	Large scale ($10^6 - 10^{10}$)	Small scale (2 - 100)
Few worker failures	Highly unreliable	Few worker failures
...	...	...

Data Heterogeneity

System Heterogeneity

[1] Kairouz, Peter, et al. "Advances and open problems in federated learning," Foundations and Trends in Machine Learning, 2021.

Sever-Centric Federated Learning



$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) \triangleq \min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i\in[M]} \alpha_i f_i(\mathbf{x}, D_i)$$

f_i : Non-convex loss function

α_i : Data proportion

D_i : Local data $\sim P_i$

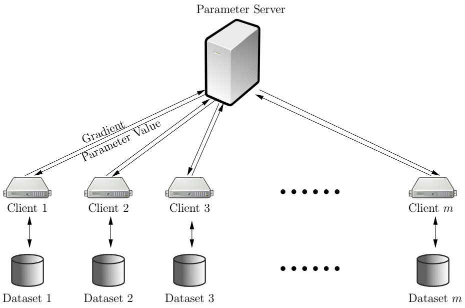
Selection:
- server select m workers to participate

Computation:
- worker makes local updates (K)

Aggregation:
- server aggregates results and updates model

[2] McMahan, H. B., Moore, E., Ramage, and D., Hampson, S., et al., "Communication-efficient learning of deep networks from decentralized data," Proc. AISTATS 2017.

Sever-Centric Federated Learning



$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) \triangleq \min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i\in[M]} \alpha_i f_i(\mathbf{x}, D_i)$$

f_i : Non-convex loss function

α_i : Data proportion

D_i : Local data $\sim P_i$

Selection:
- server select m workers to participate

Computation:
- worker makes local updates (K)

Aggregation:
- server aggregates results and updates model

Server-centric "FedAvg" algorithm (selection-computation-aggregation):
Linear speedup for convergence: $\mathcal{O}(1/\sqrt{mKT})$

[2] McMahan, H. B., Moore, E., Ramage, and D., Hampson, S., et al., "Communication-efficient learning of deep networks from decentralized data," Proc. AISTATS 2017.

Limitations of Server-Centric Federated Learning

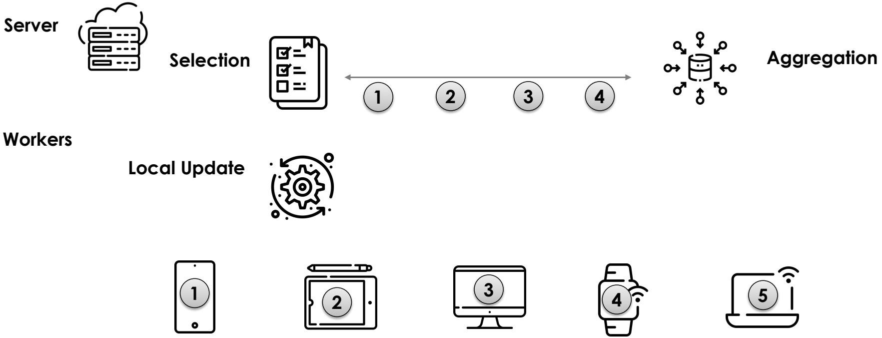
Server-centric FL (Selection-Computation-Aggregation):

Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …

Limitations of Server-Centric Federated Learning

Server-centric FL (Selection-Computation-Aggregation):

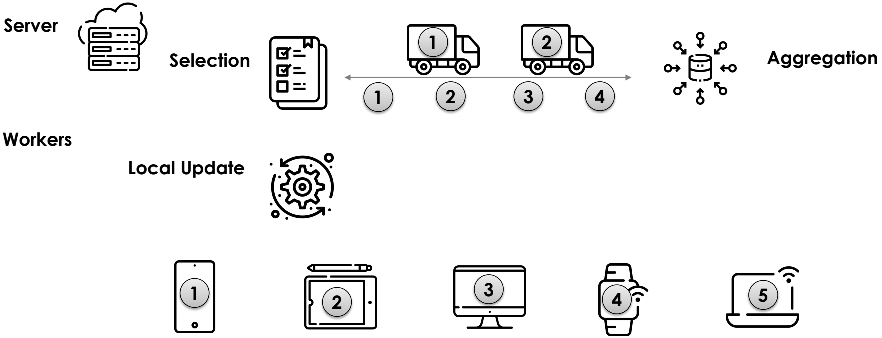
Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …



Limitations of Server-Centric Federated Learning

Server-centric FL (Selection-Computation-Aggregation):

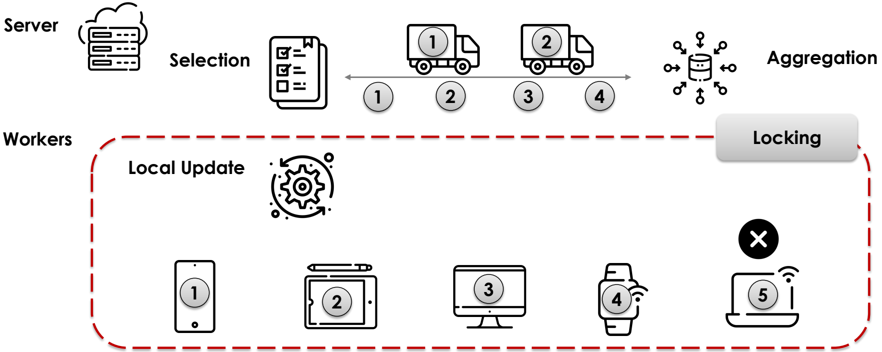
Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …



Limitations of Server-Centric Federated Learning

Server-centric FL (Selection-Computation-Aggregation):

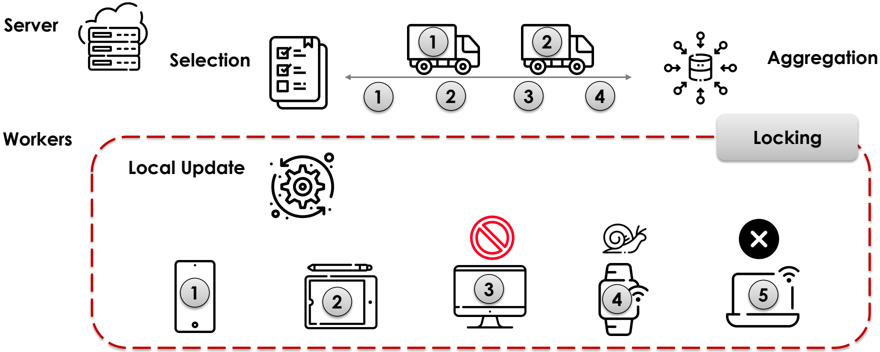
Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …



Limitations of Server-Centric Federated Learning

Server-centric FL (Selection-Computation-Aggregation):

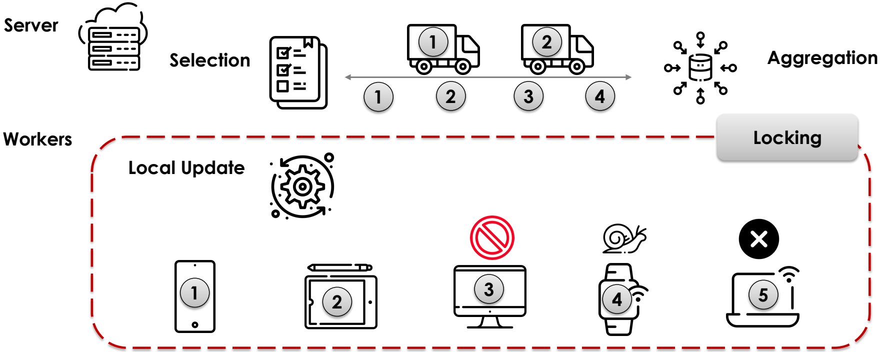
Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …



Limitations of Server-Centric Federated Learning

Server-centric FL (Selection-Computation-Aggregation):

Tight worker-server coupling: 1) straggler, 2) energy waste, 3) bias/fairness …



Our Solution: Anarchic Federated Learning

General Framework of AFL

General Framework of AFL

At the Server (Concurrently with Workers):

1. (Concurrent Thread) Collect local updates returned from the workers.
2. (Concurrent Thread) Aggregate the workers' results and update global model following some server-side optimization process.

General Framework of AFL

At the Server (Concurrently with Workers):

1. (Concurrent Thread) Collect local updates returned from the workers.
2. (Concurrent Thread) Aggregate the workers' results and update global model following some server-side optimization process.

At Each Worker (Concurrently with Server):

1. Once decided to participate in the training, pull the global model with current timestamp.
2. Perform (multiple) local update steps following some worker-side optimization process.
3. Return the result and the associated pulling timestamp to the server, with extra processing if so desired.

Fundamental Questions

Fundamental Questions

1) Is it possible to design algorithms that converge under AFL?

Fundamental Questions

1) Is it possible to design algorithms that converge under AFL?

2) If the answer to 1) is “yes,” then under what condition and how fast could the algorithms converge?

Fundamental Questions

1) Is it possible to design algorithms that converge under AFL?
2) If the answer to 1) is "yes," then under what condition and how fast could the algorithms converge?
3) If 2) can be resolved, could the highly desirable "linear speedup effect" still be achievable under AFL?

Fundamental Questions

1) Is it possible to design algorithms that converge under AFL?

2) If the answer to 1) is "yes," then under what condition and how fast could the algorithms converge?

3) If 2) can be resolved, could the highly desirable "linear speedup effect" still be achievable under AFL?

The answers to all these questions are affirmative under AFL!

Fundamental Convergence Error Lower Bound

Theorem 1 (Convergence Error Lower Bound)

- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$
- *Unbiased stochastic gradients:* $\mathbb{E}[\nabla f_i(\mathbf{x}_i, \xi_k^i)] = \nabla f_i(\mathbf{x}_k)$
- *Bounded dissimilarity for non-i.i.d. data across workers:* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_i, \xi_k^i) - \nabla f_i(\mathbf{x}_k)\|^2] \leq \sigma_L^2$ *and* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_k) - \nabla f(\mathbf{x}_k)\|^2] \leq \sigma_G^2$

Fundamental Convergence Error Lower Bound

Theorem 1 (Convergence Error Lower Bound)

- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$
- *Unbiased stochastic gradients:* $\mathbb{E}[\nabla f_i(\mathbf{x}_i, \xi_k^i)] = \nabla f_i(\mathbf{x}_k)$
- *Bounded dissimilarity for non-i.i.d. data across workers:* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_i, \xi_k^i) - \nabla f_i(\mathbf{x}_k)\|^2] \leq \sigma_L^2$ *and* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_k) - \nabla f(\mathbf{x}_k)\|^2] \leq \sigma_G^2$
- *Then, under general worker information arrival processes, there exists a loss function (and its stochastic gradient estimator) such that the output* $\tilde{\mathbf{x}}$ *of any state-less AFL algorithm satisfies:*

$$\mathbb{E}[\|\nabla f(\tilde{\mathbf{x}})\|^2] = \Omega(\sigma_G^2).$$

Anarchic FedAvg for Cross-Device (AFA-CD)

Anarchic FedAvg for Cross-Device (AFA-CD)

At the Server (Concurrently with Workers):

1. In t-th round, collect m local updates $\{\mathbf{G}_i(\mathbf{x}_{t-\tau_{t,i}}), i \in \mathcal{M}_t\}$ from workers to form set $\mathcal{M}_t$, where $\tau_{t,i}$ is the random delay of worker i, $i \in \mathcal{M}_t$.
2. Aggregate and update: $\mathbf{G}_t = \frac{1}{m}\sum_{i\in\mathcal{M}_t}\mathbf{G}_i(\mathbf{x}_{t-\tau_{t,i}}), \quad \mathbf{x}_{t+1} = \mathbf{x}_t - \eta\mathbf{G}_t.$

Anarchic FedAvg for Cross-Device (AFA-CD)

At the Server (Concurrently with Workers):

1. In t-th round, collect m local updates $\{\mathbf{G}_i(\mathbf{x}_{t-\tau_{t,i}}), i \in \mathcal{M}_t\}$ from workers to form set $\mathcal{M}_t$, where $\tau_{t,i}$ is the random delay of worker i, $i \in \mathcal{M}_t$.
2. Aggregate and update: $\mathbf{G}_t = \frac{1}{m} \sum_{i \in \mathcal{M}_t} \mathbf{G}_i(\mathbf{x}_{t-\tau_{t,i}}), \quad \mathbf{x}_{t+1} = \mathbf{x}_t - \eta \mathbf{G}_t$.

At Each Worker (Concurrently with Server):

1. Once decided to participate in the training, retrieve the parameter $\mathbf{x}_\mu$ from the server and its timestamp, set local model: $\mathbf{x}^i_{\mu,0} = \mathbf{x}_\mu$.
2. Choose a local step number $K_{t,i}$ (can be time-varying & device-dependent). Let $\mathbf{x}^i_{\mu,k+1} = \mathbf{x}^i_{\mu,k} - \eta_L \mathbf{g}^i_{\mu,k}$, where $\mathbf{g}^i_{\mu,k} = \nabla f_i(\mathbf{x}^i_{\mu,k}, \xi^i_{\mu,k})$, $k = 0, \ldots, K_{t,i} - 1$.
3. Sum & scale stochastic gradients: $\mathbf{G}_i(\mathbf{x}_\mu) = \frac{1}{K_{t,i}} \sum_{j=0}^{K_{t,i}-1} \mathbf{g}^i_{\mu,j}$. Return $\mathbf{G}_i(\mathbf{x}_\mu)$.

Convergence Performance of AFA-CD

Theorem 2 (AFA-CD w/ General Worker Info Arrival Processes)

- *Bounded maximum delay:* $\exists\ \tau := \max_{t \in [T], i \in \mathcal{M}_t} \{\tau_{t,i}\} < \infty$
- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$
- *Unbiased stochastic gradients:* $\mathbb{E}[\nabla f_i(\mathbf{x}_i, \xi_k^i)] = \nabla f_i(\mathbf{x}_k)$
- *Bounded dissimilarity for non-i.i.d. data across workers:* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_i, \xi_k^i) - \nabla f_i(\mathbf{x}_k)\|^2] \leq \sigma_L^2$ *and* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_k) - \nabla f(\mathbf{x}_k)\|^2] \leq \sigma_G^2$

Convergence Performance of AFA-CD

Theorem 2 (AFA-CD w/ General Worker Info Arrival Processes)

- *Bounded maximum delay:* $\exists\, \tau := \max_{t \in [T], i \in \mathcal{M}_t} \{\tau_{t,i}\} < \infty$
- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \le L\|\mathbf{x} - \mathbf{y}\|$
- *Unbiased stochastic gradients:* $\mathbb{E}[\nabla f_i(\mathbf{x}_i, \xi_k^i)] = \nabla f_i(\mathbf{x}_k)$
- *Bounded dissimilarity for non-i.i.d. data across workers:* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_i, \xi_k^i) - \nabla f_i(\mathbf{x}_k)\|^2] \le \sigma_L^2$ *and* $\mathbb{E}[\|\nabla f_i(\mathbf{x}_k) - \nabla f(\mathbf{x}_k)\|^2] \le \sigma_G^2$
- *Then output sequence $\{\mathbf{x}_t\}$ generated by AFA-CD with general worker information arrival processes satisfies:*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla f(\mathbf{x}_t)\|^2 \le \frac{4(f_0 - f_*)}{\eta\eta_L T} + 4\big(\alpha_L \sigma_L^2 + \alpha_G \sigma_G^2\big),$$

where the constants α_L and α_G are problem-dependent constants.

Convergence Performance of AFA-CD

Corollary 3 (Linear Speedup to an Error Ball)

With a constant local step K, let $\eta_L = \frac{1}{\sqrt{T}}$, and $\eta = \sqrt{mK}$, the convergence rate of AFA-CD with general worker information arrival processes is:

$$\mathcal{O}\left(\frac{1}{\sqrt{mKT}}\right) + \mathcal{O}\left(\frac{\tau^2}{T}\right) + \mathcal{O}\left(\frac{K^2}{T}\right) + \mathcal{O}(\sigma_G^2).$$

Anarchic Federated Averaging for Cross-Silo (AFA-CS)

At the Server (Concurrently w/ Workers):

1. In $t-$th round, collect m local updates.
2. Update worker i's information in memory using the returned local update $\mathbf{G}_i$.
3. Aggregate and update: $\mathbf{G}_t = \frac{1}{M}\sum_{i\in[M]}\mathbf{G}_i, \quad \mathbf{x}_{t+1} = \mathbf{x}_t - \eta\mathbf{G}_t.$

At Each Worker (Concurrently w/ Server): Same as AFA-CD.

Convergence Performance of AFA-CS

Theorem 4

- *Bounded maximum delay:* $\exists\ \tau := \max_{t\in[T], i\in\mathcal{M}_t}\{\tau_{t,i}\} < \infty$
- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x}-\mathbf{y}\|$
- *Choose* η *and* η_L *as such that* $6\eta_L^2(2K_{t,i}^2 - 3K_{t,i} + 1)L^2 \leq 1, \forall t, i,$ $(\frac{\eta\eta_L(M-m^{'})^2L^2\tau^2}{M^2} + \frac{L}{2})\eta\eta_L \leq \frac{1}{4}$, *and* $\frac{30L^2\eta_L^2\tau}{M}(\sum_{i\in[M]} K_{t,i}^2) \leq \frac{1}{4}$.

Convergence Performance of AFA-CS

Theorem 4

- *Bounded maximum delay:* $\exists\ \tau := \max_{t\in[T], i\in\mathcal{M}_t}\{\tau_{t,i}\} < \infty$
- *L-Lipschitz smoothness:* $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x}-\mathbf{y}\|$
- *Choose η and η_L as such that* $6\eta_L^2(2K_{t,i}^2 - 3K_{t,i} + 1)L^2 \leq 1, \forall t, i,$ $(\frac{\eta\eta_L(M-m')^2L^2\tau^2}{M^2} + \frac{L}{2})\eta\eta_L \leq \frac{1}{4}$, *and* $\frac{30L^2\eta_L^2\tau}{M}(\sum_{i\in[M]} K_{t,i}^2) \leq \frac{1}{4}$.
- *Then, under same assumptions in Thm 2, output sequence $\{\mathbf{x}_t\}$ generated by AFA-CS under general worker information arrival processes satisfies:*

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(\mathbf{x}_t)\|^2 \leq \frac{4f(\mathbf{x}_0) - f(\mathbf{x}_T)}{\eta\eta_L T} + \alpha_L\sigma_L^2 + \alpha_G\sigma_G^2,$$

where the constants α_L and α_G are problem-dependent constants.

Convergence Performance of AFA-CS

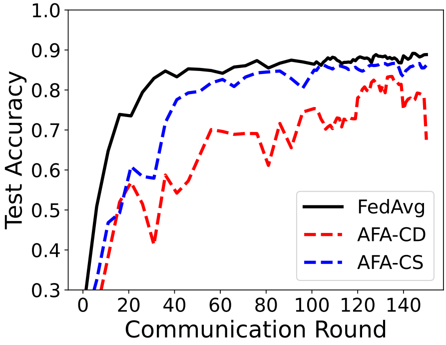
Corollary 5 (Linear Speedup)

Suppose a constant local step K, and let $\eta_L = \frac{1}{\sqrt{T}}$, and $\eta = \sqrt{MK}$, the convergence rate of the AFA-CS algorithm under general worker information arrival processes is:

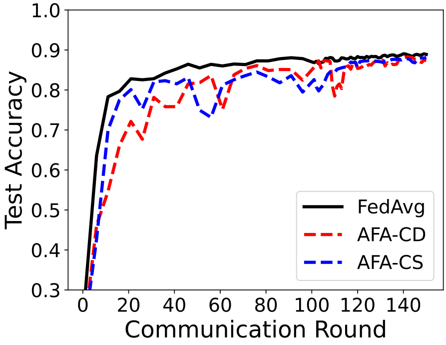
$$\mathcal{O}\left(\frac{1}{\sqrt{MKT}}\right) + \mathcal{O}\left(\frac{K^2}{MT}\right) + \mathcal{O}\left(\frac{\tau^2 (M - m')^2}{TM^2}\right).$$

Numerical Results

- Logistic regression test accuracy: 10-worker, non-i.i.d. MNIST dataset



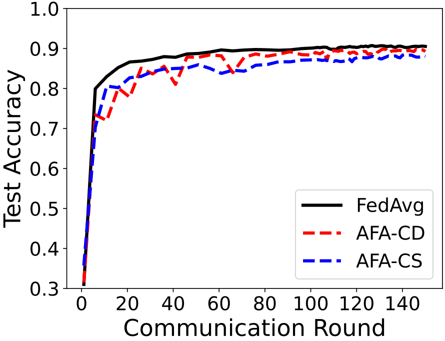
Non-i.i.d. index $p = 1$



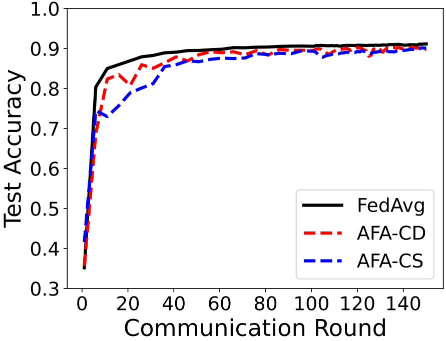
Non-i.i.d. index $p = 2$

Numerical Results

- Test accuarcy for logistic regression on non-i.i.d. MNIST dataset



Non-i.i.d. index $p = 5$



Non-i.i.d. index $p = 10$

Conclusion

- Proposed a new federated learning paradigm – Anarchic Federated Learning (AFL)
  - From server-centric to worker-spontaneous
  - Loose server-worker coupling
  - The workers can learn anytime in anyway they want

- Provided basic understandings on convergence conditions under AFL

- Showed that the highly desirable linear speedup effect remains achievable under AFL

Thank You!

Discussions: Poster Session 3, Thu 7/21 6 p.m. – 8 p.m. EDT, Hall E #711