FedPAGE: A Fast Local Stochastic Gradient Method for Communication-Efficient Federated Learning

Haoyu Zhao

Princeton University

haoyu@princeton.edu

Joint work with: Zhize Li and Peter Richtárik

August 11, 2021

Figure: Zhize Li

Figure: Peter Richtárik

Overview

Section 1

Introduction and Related Works

General Problem Setting

1. one central server and N clients
2. each client holds M data (for simplicity)
3. the clients can communicate with the central server but cannot connect with other clients
4. there is a global model on the server, and the clients communicates with the server to update the model

Figure: A federated learning application. The figure comes from the link.[1]

[1] https://blog.ml.cmu.edu/2019/11/12/federated-learning-challenges-methods-and-future-directions/

General Problem Setting

$$\min_{x \in \mathbb{R}^{d}} f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x)$$

parameters

devices

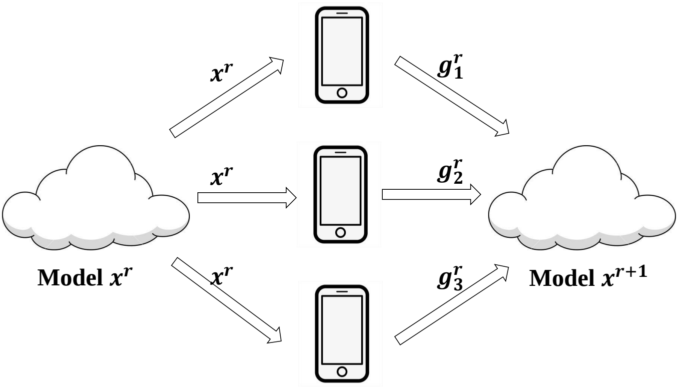
data on device i

$$f_i(x) = \frac{1}{M} \sum_{j=1}^{M} f_{i,j}(x)$$

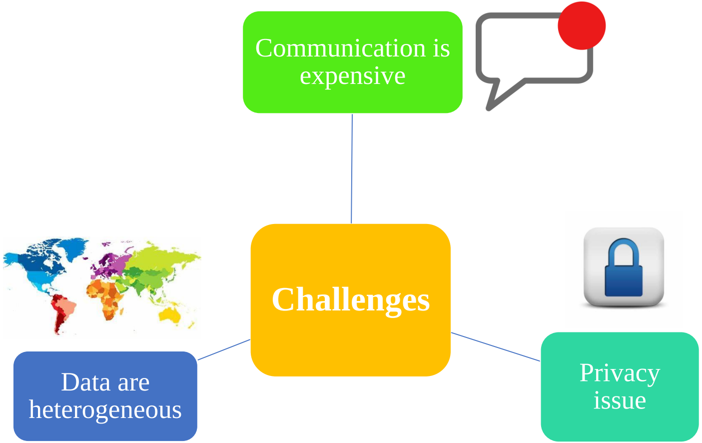
Loss function of data samples on device i

General Problem Setting

1. at each communication round r, the server broadcast the current model x^r to some clients S^r
2. the clients computes some function g_i^r and transfer back to the server
3. the server update the model according to g_i^r

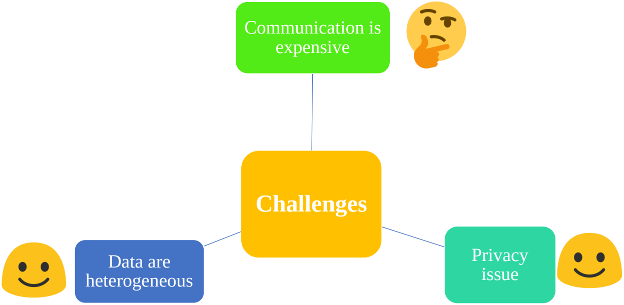


Challenges


Communication is expensive
Challenges
Data are heterogeneous
Privacy issue

Challenges

1. we do not transfer the data to the central server (privacy issue[2] ✓)
2. we do not assume $\{f_i\}$ or $\{f_{i,j}\}$ to have similarity: we view them as arbitrary functions (heterogeneity issue (non IID issue) ✓)
3. how about the expensive communication (communication issue ?)



[2] Here transferring the gradient may also leak the personal information, but we do not consider this 'advanced' privacy in this project. Please refer to more differential privacy works for more information.

Related Works

There are two lines of work to overcome the communication problem: compression operators and local methods.

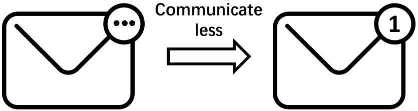


Figure: Compression operators: communicate less during one communication round

Figure: Local methods: work more during one communication round

Related Works — Compression Operators

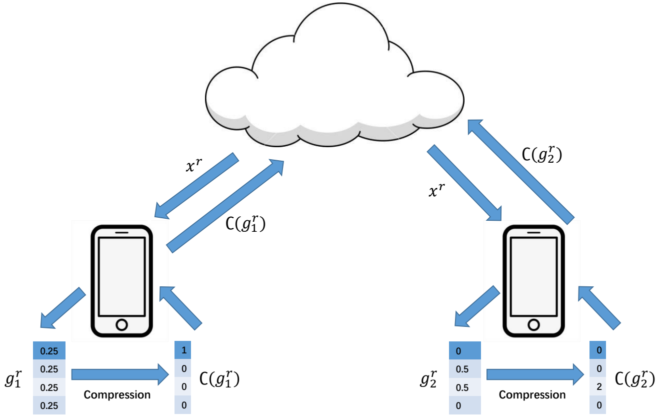


Figure: Compression operators: each device send the compressed information to the central server.

Related Works — Compression Operators

1. QSGD [Alistarh et al., 2017]: compressed version of SGD
2. SignSGD [Bernstein et al., 2018]: compressed version of SGD
3. DIANA [Mishchenko et al., 2019]: compressed version of SVRG
4. ADIANA [Li et al., 2020]: accelerated version of DIANA

Related Works — Local Methods

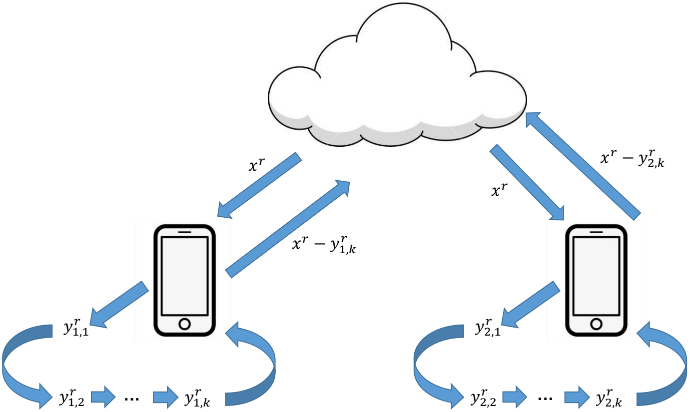


Figure: Local methods: each device perform multiple local updates before communicating with the central server.

Related Works — Local Methods

1. FedAvg [McMahan et al., 2017]: Stochastic gradient descent(SGD) with local steps
2. Local-SVRG [Gorbunov et al., 2020]: SVRG[Johnson and Zhang, 2013] with local steps
3. SCAFFOLD [Karimireddy et al., 2020]: SAGA[Defazio et al., 2014] with local steps
4. FedPAGE (this paper): PAGE[Li et al., 2021] with local steps

Section 2

Problem Setting and Assumptions

Problem Setting

$$\min_{x \in \mathbb{R}^{d}} f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x)$$

parameters

devices

data on device i

$$f_i(x) = \frac{1}{M} \sum_{j=1}^{M} f_{i,j}(x)$$

Loss function of data samples on device i

Problem Setting

Nonconvex Setting:

1. All functions can be nonconvex.
2. We want to find x such that $\mathbb{E}\|\nabla f(x)\| \leq \epsilon$.

Convex Setting:

1. We assume $f(x)$ is convex, and $f_i(x)$ may be nonconvex.
2. We want to find x such that $\mathbb{E} f(x) - f^* \leq \epsilon$.

Assumptions

Assumption (L-smoothness)

All functions $f_{i,j} : \mathbb{R}^d \to \mathbb{R}$ *for all* $i \in [N], j \in [M]$ *are L-smooth. That is, there exists* $L \geq 0$ *such that for all* $x_1, x_2 \in \mathbb{R}^d$ *and all* $i \in [N], j \in [M]$,

$$\|\nabla f_{i,j}(x_1) - \nabla f_{i,j}(x_2)\| \leq L\|x_1 - x_2\|.$$

Can be generalized to different functions are $L_{i,j}$ smooth.

Assumption (Bounded Variance)

There exists $\sigma \geq 0$ *such that for any client* $i \in [N]$ *and* $x \in \mathbb{R}^d$,

$$\frac{1}{M}\sum_{j=1}^{M} \|\nabla f_{i,j}(x) - \nabla f_i(x)\|_2^2 \leq \sigma^2.$$

Section 3

FedPAGE Algorithm

PAGE in the federated learning setting

Algorithm 1 PAGE in the federated learning setting

for $r = 1, 2, \ldots, R$ **do**
  sample $q \sim \text{Bernoulli}(p_r)$
  if $q = 1$ **then**
    clients $S^r = [N]$, communicate x^r to all $i \in S^r$
    clients $i \in S^r$ compute $g_i^r \leftarrow \nabla f_i(x^r)$
    $g^r \leftarrow \frac{1}{|S^r|} \sum_{i \in S^r} g_i^r$
  else
    clients $S^r \subseteq [N]$ with size S, send (x^r, x^{r-1}, g^{r-1}) to all $i \in S^r$
    clients $i \in S^r$ compute $g_i^r \leftarrow \nabla f_i(x^r) - \nabla f_i(x^{r-1}) + g^{r-1}$
    $g^r \leftarrow \frac{1}{|S^r|} \sum_{i \in S^r} g_i^r$
  end if
  $x^{r+1} \leftarrow x^r - \eta_g g^r$
end for

Add local steps to PAGE: FedPAGE-Full

Algorithm 2 LocalSteps-Full

procedure $\textsc{LocalSteps-Full}(i, x^r, x^{r-1}, g^{r-1})$
  $y_{i,0}^r \leftarrow x^r$
  $g_{i,0}^r \leftarrow \nabla f_i(x^r) - \nabla f_i(x^{r-1}) + g^{r-1}$
  $y_{i,1}^r \leftarrow y_{i,0}^r - \eta_l g_{i,0}^r$
  for $k = 1, 2, \ldots, K-1$ **do**
    $g_{i,k}^r \leftarrow \nabla f_i(y_{i,k}^r) - \nabla f_i(y_{i,k-1}^r) + g_{i,k-1}^r$
    $y_{i,k+1}^r \leftarrow y_{i,k}^r - \eta_l g_{i,k}^r$
  end for
  $\Delta y_i^r \leftarrow x^r - y_{i,K}^r$
  return Δy_i^r
end procedure

1. We add local steps to PAGE when the server does not communicate with all clients ($q = 0$)

Algorithm 3 FedPAGE-Full

```
for r = 1, 2, ..., R do
    sample q ~ Bernoulli(p_r)
    if q = 1 then
        clients S^r = [N], communicate x^r to all i ∈ S^r
        clients i ∈ S^r compute g_i^r ← ∇f_i(x^r) and send to the server
        g^r ← (1/|S^r|) Σ_{i∈S^r} g_i^r
    else
        clients S^r ⊆ [N] with size S, send (x^r, x^{r-1}, g^{r-1}) to all i ∈ S^r
        Δy_i^r ← LocalSteps-Full(i, x^r, x^{r-1}, g^{r-1})
        g^r ← (1/(K η_l |S^r|)) Σ_{i∈S^r} Δy_i^r
    end if
    x^{r+1} ← x^r − η_g g^r
end for
```

Optimize the local steps: FedPAGE

1. FedPAGE-Full computes K local full gradients at each client when performing local steps, which is time consuming (procedure LocalSteps-Full)
2. when M is very large, computing the local full gradient may not be affordable

Optimize the local steps: FedPAGE

Update rule in **LocalSteps-Full**

1

$$g_{i,0}^r = \nabla f_i(x^r) - \nabla f_i(x^{r-1}) + g^{r-1}$$
$$g_{i,k+1}^r = g_{i,k}^r + \nabla f_i\left(y_{i,k+1}^r\right) - \nabla f_i\left(y_{i,k}^r\right)$$

2

$$g_{i,0}^r = \nabla f_i(x^r) - \nabla f_i(x^{r-1}) + g^{r-1}$$
$$g_{i,k+1}^r = g_{i,k}^r + \nabla f_{i,j}\left(y_{i,k+1}^r\right) - \nabla f_{i,j}\left(y_{i,k}^r\right)$$

Use a large batch

3

$$g_{i,0}^r = \nabla_{\mathcal{J}_2} f_i(x^r) - \nabla_{\mathcal{J}_2} f_i(x^{r-1}) + g^{r-1}$$
$$g_{i,k+1}^r = g_{i,k}^r + \nabla f_{i,j}\left(y_{i,k+1}^r\right) - \nabla f_{i,j}\left(y_{i,k}^r\right)$$

Optimize the local steps: FedPAGE

When M is very large and we cannot compute the local full gradient, we use large minibatch to estimate the local full gradient.

Algorithm 4 LocalSteps

procedure $\textsc{LocalSteps}(i, x^r, x^{r-1}, g^{r-1})$
    $y_{i,0}^r \leftarrow x^r$
    $g_{i,0}^r \leftarrow \nabla_{\mathcal{I}_2} f_i(x^r) - \nabla_{\mathcal{I}_2} f_i(x^{r-1}) + g^{r-1}$
    $y_{i,1}^r \leftarrow y_{i,0}^r - \eta_l g_{i,0}^r$
    for $k = 1, 2, \ldots, K-1$ **do**
        sample $j \in [M]$, $g_{i,k}^r \leftarrow \nabla f_{i,j}(y_{i,k}^r) - \nabla f_{i,j}(y_{i,k-1}^r) + g_{i,k-1}^r$
        $y_{i,k+1}^r \leftarrow y_{i,k}^r - \eta_l g_{i,k}^r$
    end for
    $\Delta y_i^r \leftarrow x^r - y_{i,K}^r$
    return Δy_i^r
end procedure

Optimize the local steps: FedPAGE

Algorithm 5 FedPAGE

for $r = 1, 2, \ldots, R$ **do**
  sample $q \sim \text{Bernoulli}(p_r)$
  if $q = 1$ **then**
    clients $S^r = [N]$, communicate x^r to all $i \in S^r$
    clients $i \in S^r$ compute $g_i^r \leftarrow \nabla_{\mathcal{I}_1} f_i(x^r)$ and send to the server
    $g^r \leftarrow \frac{1}{|S^r|} \sum_{i \in S^r} g_i^r$
  else
    clients $S^r \subseteq [N]$ with size S, send (x^r, x^{r-1}, g^{r-1}) to all $i \in S^r$
    $\Delta y_i^r \leftarrow \textsc{LocalSteps}(i, x^r, x^{r-1}, g^{r-1})$
    $g^r \leftarrow \frac{1}{K\eta_l |S^r|} \sum_{i \in S^r} \Delta y_i^r$
  end if
  $x^{r+1} \leftarrow x^r - \eta_g g^r$
end for

Section 4

Convergence Results

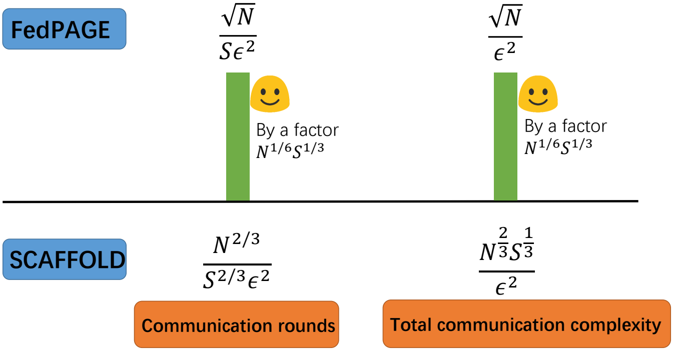
Convergence in the Nonconvex Setting

Theorem (Convergence of FedPAGE in nonconvex setting)

Under standard assumptions, if we choose the parameters properly, FedPAGE *will find a point x such that $\mathbb{E}\|\nabla f(x)\|_2 \leq \epsilon$ within the following number of communication rounds:*

$$R = O\left(\frac{L(\sqrt{N} + S)}{S\epsilon^2}\right).$$

1. The number of communication round is $O(\sqrt{N}/(S\epsilon^2))$ when $S \leq \sqrt{N}$, which matches the convergence rate of PAGE
2. The total communication complexity is $O(N + \sqrt{N}/\epsilon^2)$, because we communicate with all the clients in the first round



1. SCAFFOLD[Karimireddy et al., 2020]: State-of-the-art, ICML 2020
2. FedPAGE is more suitable when N is very large, e.g. federated learning applications related to mobile phones or PCs.

Convergence in the Convex Setting

Theorem (Convergence of FedPAGE in convex setting)

Under standard assumptions, if we choose the parameters properly, FedPAGE *will find a point* x *such that* $\mathbb{E}f(x) - f^* \leq \epsilon$ *with the number of communication rounds bounded by*

$$R = O\left(\frac{N^{3/4}L}{S\epsilon}\right).$$

1. The number of communication round is $O(N^{3/4}/(S\epsilon))$ when $S \leq \sqrt{N}$
2. The total communication complexity is $O(N + N^{3/4}/\epsilon)$, because we communicate with all the clients in the first round

Convergence in the Convex Setting

	Communication rounds	Total communication complexity
FedPAGE	$\frac{N^{3/4}}{S\epsilon}$	$\frac{N^{3/4}}{\epsilon}$
	By a factor $N^{1/4}$	By a factor $N^{1/4}$
SCAFFOLD	$\frac{N}{S\epsilon}$	$\frac{N}{\epsilon}$

Section 5

Proof Sketch

Proof Idea in the Nonconvex Setting

1. The main part is to bound the 'variance term'. We bound $\mathbb{E}\|g^r - \nabla f(x^r)\|^2$
2. It is easy to bound $\mathbb{E}\|\frac{1}{S}\sum_{i\in S^r} g_{i,0}^r - \nabla f(x^r)\|^2$ (main lemma in PAGE)
3. Bound $\mathbb{E}\|\frac{1}{S}\sum_{i\in S^r} g_{i,0}^r - g^r\|^2$
4. Bound $\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}_r\|g_{i,k}^r - g_{i,0}^r\|^2$, for any i, k, r
5. If η_l is small, then $y_{i,k}^r \approx x^r$, $g_{i,k}^r \approx g_{i,0}^r$, and the above equation will be small

Proof Idea in the Convex Setting

1. PAGE/FedPAGE uses biased gradient estimator, need to bound the following inner product term.
2. We can bound the inner product $\sum_{r=1}^{t} \mathbb{E}\langle \nabla f(x^r) - g^r, x^r - x^* \rangle$
3. For FedPAGE, we still need to consider the 'local error' generated from the local steps.
4. Use the bounds on $\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}_r \|g_{i,k}^r - g_{i,0}^r\|^2$, for any i, k, r

Section 6

Numerical Experiments

Experiments Setup

1. We set the objectives to be robust linear regression and logistic regression with nonconvex regularizer
2. The objective function for robust linear regression is $f(x) = \frac{1}{n}\sum_{i=1}^{n} \ell(x^T a_i - b_i)$, where $\ell(t) = \log(1 + \frac{t^2}{2})$
3. The objective function for logistic regression with nonconvex regularizer is $f(x) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + \exp(-b_i x^T a_i)\right) + \alpha \sum_{j=1}^{d} \frac{x_j^2}{1+x_j^2}$
4. We perform two experiments: the first shows the effectiveness of local steps, and the second compares FedPAGE with other methods.

Effectiveness of Local Steps

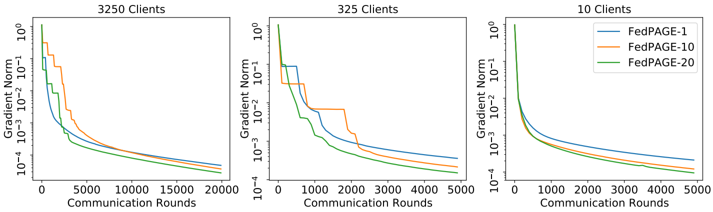


Figure: Robust linear regression on a9a dataset

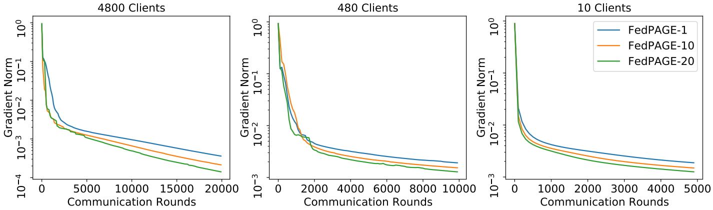


Figure: Robust linear regression on w8a dataset

Superiority over Other Methods

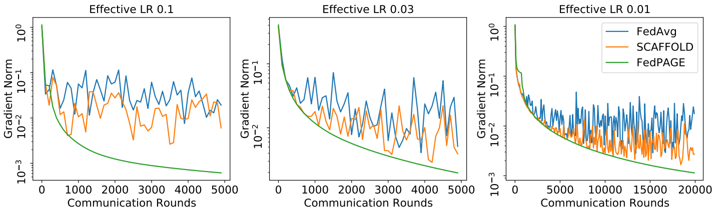


Figure: Robust linear regression on a9a with 3250 clients (each with 10 sample)

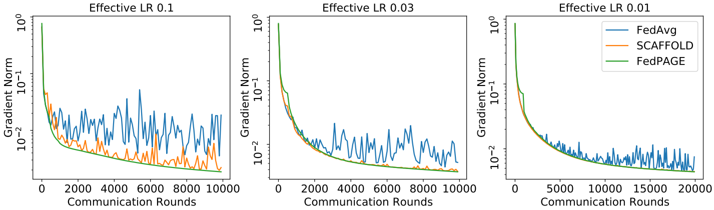


Figure: Robust linear regression on w8a with 4800 clients (each with 10 sample)

Superiority over Other Methods

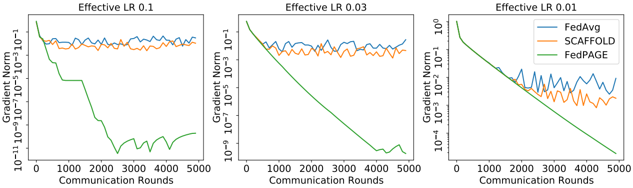


Figure: Logistic regression with nonconvex regularizer on a9a with 3250 clients (each with 10 sample)

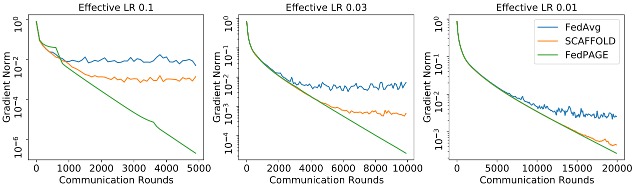


Figure: Logistic regression with nonconvex regularizer on w8a with 4800 clients (each with 10 sample)

Conclusion

1. We design FedPAGE algorithm, which is a communication-efficient local method for federated learning.
2. From theory, we improve the communication rounds and communication complexity of the state-of-the-art SCAFFOLD.
3. From experiments, we show the effectiveness of local steps and superiority of FedPAGE over other existed methods.

Thanks

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. *Advances in Neural Information Processing Systems*, 30: 1709–1720, 2017.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pages 560–569. PMLR, 2018.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654, 2014.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. *arXiv preprint arXiv:2011.02828*, 2020.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. In *International Conference on Machine Learning*, pages 5895–5904. PMLR, *arXiv:2002.11364*, 2020.

Zhize Li, Hongyan Bao, Xiangliang Zhang, and Peter Richtárik. PAGE: A simple and optimal probabilistic gradient estimator for nonconvex optimization. In *International Conference on Machine Learning*, pages 6286–6295. PMLR, *arXiv:2008.10898*, 2021.

H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.