FEDERATED STOCHASTIC GRADIENT DESCENT BEGETS SELF-INDUCED MOMENTUM

Howard H. Yang[§], Zuozhu Liu[§], Yaru Fu[*], Tony Q. S. Quek[†], and H. Vincent Poor[‡]

[§] *Zhejiang University/University of Illinois at Urbana-Champaign Institute, Haining 314400, China*
[*] *Hong Kong Metropolitan University, Hong Kong, 999077*
[†] *Singapore University of Technology and Design, Singapore 487372*
[‡] *Princeton University, Princeton, NJ 08544, USA*

ABSTRACT

Federated learning (FL) is an emerging machine learning method that can be applied in mobile edge systems, in which a server and a host of clients collaboratively train a statistical model utilizing the data and computation resources of the clients without directly exposing their privacy-sensitive data. We show that running stochastic gradient descent (SGD) in such a setting can be viewed as adding a momentum-like term to the global aggregation process. Based on this finding, we further analyze the convergence rate of a federated learning system by accounting for the effects of parameter staleness and communication resources. These results advance the understanding of the Federated SGD algorithm, and also forges a link between staleness analysis and federated computing systems, which can be useful for systems designers.

***Index Terms*—** Federated learning, stochastic gradient descent (SGD), momentum, convergence rate.

1. INTRODUCTION

Federated learning (FL) is a branch of machine learning models that allow a computing unit, i.e., an edge server, to train a statistical model from data stored on a swarm of end-user entities, i.e., the clients, without directly accessing the clients' local datasets [1]. Specifically, instead of aggregating all the data to the server for training, FL brings the machine learning models directly to the clients for local computing, where only the resulting parameters are uploaded to the server for global aggregation, after which an improved model is sent back to the clients for another round of local training [2]. Such a training process usually converges after sufficient rounds of parameter exchanges and computing among the server and clients, upon which all the participants can benefit from a better machine learning model [3–5]. As a result, the salient feature of on-device training mitigates many of the systemic privacy risks as well as communication overheads, hence making FL particularly relevant for next-generation mobile networks [6–8]. Nonetheless, in the setting of FL, the server usually needs to link up a massive number of clients via a resource-limited medium, e.g., the spectrum, and hence only a limited number of the clients can be selected to participate in the federated training during each round of iteration [9–12]. This, together with the fact that the time spent on transmitting the parameters can be orders of magnitude higher than that of local computations [13, 14], makes the straggler issue a serious one in FL. To that end, a simple but effective approach has been proposed [15], i.e., reusing the outdated parameters in the global aggregation stage so as to accelerate the training efficiency. The gain of this scheme has been amply demonstrated via experiments while the intrinsic rationale behind it remains unclear. In this paper, we take the stochastic gradient descent (SGD)-based FL training as an example and show that reusing the outdated parameters implicitly introduces a momentum-like term in the global updating process, and prove the subsequent convergence rate of federated computing. This result advances the understanding of FL and may be useful to guide further research in this area.

This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Grant No. LGJ22F010001. (*Corresponding Author: Zuozhu Liu.*)

2. SYSTEM MODEL

Let us consider an FL system consisting of one server and K clients, as depicted per Fig. 1, where K is usually a large number. Each client k has a local dataset $\mathcal{D}_k = \{\mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}\}_{i=1}^{n_k}$ with size $|\mathcal{D}_k| = n_k$, and we assume the local datasets are statistically independent across the clients. The goal of the server and clients is to jointly learn a statistical model over the datasets residing on all the clients without sacrificing their privacy. To be more concrete, the server aims to fit a vector $\mathbf{w} \in \mathbb{R}^d$ so as to minimize the following loss function without having explicit knowledge of $\mathcal{D} = \cup_{k=1}^K \mathcal{D}_k$:

$$\begin{aligned}\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}) &= \frac{1}{n}\sum_{i=1}^{n} \ell(\mathbf{w}; \mathbf{x}_i, y_i) \\ &= \frac{n_k}{n}\cdot\frac{1}{n_k}\sum_{j=1}^{n_k} \ell(\mathbf{w}; \mathbf{x}_j, y_j) \\ &= \sum_{k=1}^{K} p_k f_k(\mathbf{w}) \end{aligned} \tag{1}$$

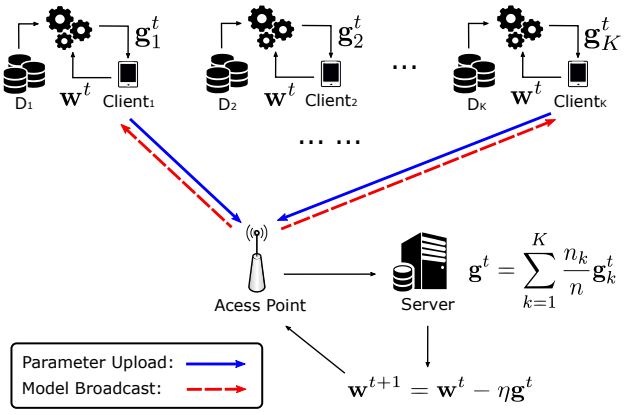

Fig. 1. An illustration of Federated SGD training: (A) clients leverage their local datasets to evaluate the gradient term, (B) the server aggregates the received updates to produce a new global model, (C) the new model is sent back to the clients, and the process is repeated.

where $n = \sum_{k=1}^{K} n_k$, $p_k = n_k/n$, $\ell(\cdot)$ is the loss function assigned on each data point, and $f_k(\mathbf{w}) = \sum_{j=1}^{n_k} \ell(\mathbf{w}; \mathbf{x}_j, y_j)/n_k$ is the local empirical loss function of client k.

Because the server has no direct access to the individual datasets, the model training needs to be carried out by the clients in a federated fashion. In this paper, we adopt Federated SGD, a widely used mechanism, for this task. The details are summarized in Algorithm 1 [2]. Specifically, at iteration t, the server needs to send the global model $\mathbf{w}^t$ to a subset of clients S_t, where in general $N = |S_t| \ll K$ because the limited communication resources cannot support simultaneous transmissions from a vast number of clients [9], for on-device model training. Upon receiving $\mathbf{w}^t$, the selected clients will leverage it to evaluate the gradient of the local empirical loss – by means of an H-step estimation – and upload the estimated gradients $\mathbf{g}_k^t$, $k \in S_t$. In essence, this comprises computing the stochastic gradient with a batch size of H data points. Finally, the server aggregates the collected parameters to produce a new output per (4). Such an orchestration amongst the server and clients repeats for a sufficient number of communication rounds until the learning process converges.

It is worth noting that the gradient aggregation step (3) in Algorithm 1 utilizes not only the fresh updates collected from the selected clients but also the outdated gradients from the unselected ones. As will be shown later, this procedure, in essence, induces an implicit momentum into the learning process.

3. ANALYSIS

This section comprises the main technical part of this paper, in which we analytically characterize the updating process of global parameters and derive the convergence rate of the Federated SGD algorithm.

Algorithm 1 Federated SGD Algorithm

Parameters: H = number of local steps per communication round, η = step size for stochastic gradient descent

Initialize: $\mathbf{w}^0 \in \mathbb{R}^d$

for $t = 0, 1, 2, ..., T-1$ **do**

  The server randomly selects a set S_t of N clients and broadcasts the global parameter $\mathbf{w}^t$ to them

  for each client $k \in S_t$ in parallel **do**

    Initialize $\mathbf{g}_k^{t,0} = 0$

    for $s = 0$ to $H-1$ **do**

      Sample $i \in \mathcal{D}_k$ uniformly at random, and update the local estimation of the gradient, $\mathbf{g}_k^{t,s}$, as follows:

$$\mathbf{g}_k^{t,s+1} = \mathbf{g}_k^{t,s} + \nabla \ell(\mathbf{w}^t; \mathbf{x}_i, y_i) \tag{2}$$

    Set $\mathbf{g}_k^t = \mathbf{g}_k^{t,H}/H$ and send the parameter back to the server

  The server collects all the updates of $\{\mathbf{g}_i^t\}_{i \in S_t}$ and assigns $\mathbf{g}_j^t = \mathbf{g}_j^{t-1}$ for all $j \neq S_t$. Then, the server updates both the estimation of gradient $\mathbf{g}^t$ and parameter $\mathbf{w}^{t+1}$ as follows:

$$\mathbf{g}^t = \sum_{k=1}^{K} \frac{n_k}{n} \mathbf{g}_k^t, \tag{3}$$

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \eta \mathbf{g}^t \tag{4}$$

Output: $\mathbf{w}^T$

3.1. Update Process of Global Parameters

Due to limited communication resources, the server can only select a subset of the clients to conduct local computing and update their gradients in every round of global iteration. As a result, the gradients of the unselected clients become stale. In accordance with (3) and (4), after the t-th communication round, the update of global parameters at the server side can be rewritten as follows:

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \eta \sum_{k=1}^{K} p_k \, \mathbf{g}_k^{t-\tau_k} \tag{5}$$

in which τ_k is the staleness of the parameters corresponding to the k-th client. Because the clients to participate in the FL are selected uniformly at random in each communication round, the staleness of parameters, $\{\tau_k\}_{k=1}^K$, can be abstracted as independently and identically distributed (i.i.d.) random variables with each following a geometric distribution:

$$\mathbb{P}(\tau_k = l) = \beta^l (1-\beta), \quad l = 0, 1, 2, ... \tag{6}$$

where $\beta = 1 - N/K$.

These considerations bring us to our first result.

Lemma 1. *Under the depicted FL framework, the parameter updating process constitutes the following relationship:*

$$\mathbb{E}\big[\mathbf{w}^{t+1}-\mathbf{w}^{t}\big]=\beta\,\mathbb{E}\big[\mathbf{w}^{t}-\mathbf{w}^{t-1}\big]-(1-\beta)\eta\,\mathbb{E}\big[\mathbf{g}^{t}\big]. \quad (7)$$

Proof. Using (5), we can subtract $\mathbf{w}^t$ from $\mathbf{w}^{t+1}$ and obtain the following:

$$\mathbf{w}^{t+1}-\mathbf{w}^{t}=\mathbf{w}^{t}-\mathbf{w}^{t-1}-\eta\sum_{k=1}^{K}p_k\big(\mathbf{g}_k^{t-\tau_k}-\mathbf{g}_k^{t-\tau_k-1}\big). \quad (8)$$

By taking an expectation with respect to the staleness τ_k, $k\in\{1,\cdots,K\}$ on both sides of the above equation, the following holds:

$$\mathbb{E}[\mathbf{w}^{t+1}-\mathbf{w}^{t}]=\mathbb{E}[\mathbf{w}^{t}-\mathbf{w}^{t-1}]-\eta\sum_{k=1}^{K}p_k\underbrace{\mathbb{E}\big[\,\mathbf{g}_k^{t-\tau_k}-\mathbf{g}_k^{t-\tau_k-1}\,\big]}_{Q_1}. \quad (9)$$

Since $\tau_k\sim Geo(1-\beta)$, we can calculate Q_1 as

$$\begin{aligned}Q_1&=\big(1-\beta\big)\mathbb{E}\big[\mathbf{g}_k^{t}\big]+\sum_{l=1}^{\infty}(1-\beta)\beta^{l}\mathbb{E}\big[\mathbf{g}_k^{t-l-1}\big]-\sum_{l=0}^{\infty}(1-\beta)\beta^{l}\mathbb{E}\big[\mathbf{g}_k^{t-l-1}\big]\\&=\big(1-\beta\big)\mathbb{E}\big[\mathbf{g}_k^{t}\big]-\sum_{l=0}^{\infty}(1-\beta)^{2}\beta^{l}\mathbb{E}\big[\mathbf{g}_k^{t-l-1}\big]. \quad (10)\end{aligned}$$

Furthermore, by noticing that for the stochastic gradient of each client k, the following result holds:

$$\mathbb{E}\big[\mathbf{g}_k^{t-\tau_k-1}\big]=\sum_{l=0}^{\infty}(1-\beta)\beta^{l}\mathbb{E}\big[\mathbf{g}_k^{t-l-1}\big], \quad (11)$$

we have

$$\begin{aligned}&\eta\sum_{k=1}^{K}p_k\,\mathbb{E}\big[\,\mathbf{g}_k^{t-\tau_k}-\mathbf{g}_k^{t-\tau_k-1}\,\big]\\&=\big(1-\beta\big)\eta\sum_{k=1}^{K}p_k\mathbb{E}\big[\mathbf{g}_k^{t}\big]-\big(1-\beta\big)\eta\sum_{k=1}^{K}p_k\sum_{l=0}^{\infty}(1-\beta)\beta^{l}\mathbb{E}\big[\mathbf{g}_k^{t-l-1}\big]\\&=\big(1-\beta\big)\eta\,\mathbb{E}\big[\mathbf{g}^{t}\big]-\big(1-\beta\big)\eta\sum_{k=1}^{K}p_k\mathbb{E}\big[\mathbf{g}_k^{t-\tau_k-1}\big]\\&\overset{(a)}{=}\big(1-\beta\big)\eta\,\mathbb{E}\big[\mathbf{g}^{t}\big]+\big(1-\beta\big)\mathbb{E}\big[\mathbf{w}^{t}-\mathbf{w}^{t-1}\big], \quad (12)\end{aligned}$$

where (a) follows from (5). Finally, by substituting (12) into (9), we complete the proof. □

From Lemma 1, we can identify a momentum-like term, namely $\beta\mathbb{E}[\mathbf{w}^t-\mathbf{w}^{t-1}]$, when the global parameter is updated from $\mathbf{w}^t$ to $\mathbf{w}^{t+1}$. This can mainly be attributed to the reuse of gradients, which introduces memory during the global aggregation step and makes the parameter vector $\mathbf{w}^{t+1}$ stay close to the current server model $\mathbf{w}^t$. Notably, such a phenomenon is also observable in the context of completely asynchronized SGD algorithms owing to similar reasons [16]. As a result, Lemma 1 can serve as a useful reference to adjust the controling factor if one intends to accelerate the Federated SGD algorithm by running it in conjunction with an *explicit momentum* term [16–18]. Besides, if the delayed gradient averaging such as [19] is employed, the design of *gradient correction* shall take into account the effect of such an implicit momentum as well.

In the sequel, we quantify the effect of this implicit momentum on the convergence performance of the FL system.

3.2. Convergence Analysis

To facilitate the analysis of the FL convergence rate, we make the following assumption on the structure of the global empirical loss function.

Assumption 1. *The gradient of each f_k is Lipschitz continuous with a constant $L>0$, i.e., for any $\mathbf{w},\mathbf{v}\in\mathbb{R}^d$ the following is satisfied:*

$$\|\nabla f_k(\mathbf{w})-\nabla f_k(\mathbf{v})\|\le L\|\mathbf{w}-\mathbf{v}\|. \quad (13)$$

This assumption is standard in the machine learning literature and is satisfied by a wide range of machine learning models, such as SVM, logistic regression, and neural networks. Besides, no assumption regarding the convexity of the objective function is made. We further leverage a notion, termed gradient coherence, to track the variant of the gradient during the training process, defined as follows [20].

Definition 1. *The gradient coherence at communication round t is defined as*

$$\mu_t=\min_{0\le s\le t}\frac{\langle\,\nabla f(\mathbf{w}^s),\nabla f(\mathbf{w}^t)\,\rangle}{\|\nabla f(\mathbf{w}^s)\|^2}. \quad (14)$$

The gradient coherence characterizes the largest deviation of directions between the current gradient and the gradients along the past iterations. As such, if μ_t is positive, then the direction of the current gradient is well aligned to those of the previous ones, and hence reusing the trained parameters can push forward the global parameter vector toward the optimal point.

Theorem 1. *Suppose the gradient coherence μ_t is lower bounded by some $\mu>0$ for all t and the variance of the stochastic gradient is upper bounded by $\sigma^2>0$. If we choose the step size to be $\eta=1/\sqrt{LT}$, then after T rounds of*

communication, the Alg. 1 converges as follows:

$$\min_{0\le t\le T-1}\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]\le\frac{2\sqrt{L}\big[f(\mathbf{w}^0)-f(\mathbf{w}^*)+\sigma^2\big]}{\big[1-(1-\mu)\beta\big]\sqrt{T}} \tag{15}$$

in which $\mathbf{w}^*=\arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$.

Proof. Following Assumption 1, we know that the empirical loss function $f(\cdot)$ is L-smooth, and hence after the t-th round of global parameter update the following holds:

$$\mathbb{E}\big[f(\mathbf{w}^{t+1})\big]\le\mathbb{E}\big[f(\mathbf{w}^t)\big]+\underbrace{\mathbb{E}\big[\langle\,\mathbf{w}^{t+1}-\mathbf{w}^t,\nabla f(\mathbf{w}^t)\,\rangle\big]}_{Q_1}+\frac{L}{2}\underbrace{\mathbb{E}\big[\|\mathbf{w}^{t+1}-\mathbf{w}^t\|^2\big]}_{Q_2}. \tag{16}$$

Using Lemma 1, we can expand the terms in Q_1 and obtain the following:

$$\begin{aligned}
Q_1&\overset{(8)}{=}\mathbb{E}\big[\langle\,\beta(\mathbf{w}^t-\mathbf{w}^{t-1})-\eta(1-\beta)\mathbf{g}^t,\nabla f(\mathbf{w}^t)\,\rangle\big]\\
&\overset{(a)}{=}\beta\mathbb{E}\big[\langle\,\mathbf{w}^t-\mathbf{w}^{t-1},\nabla f(\mathbf{w}^t)\,\rangle\big]-\eta(1-\beta)\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]\\
&\overset{(b)}{=}-\eta\beta\mathbb{E}\big[\langle\,\nabla f(\mathbf{w}^{t-\tau-1}),\nabla f(\mathbf{w}^t)\,\rangle\big]-\eta(1-\beta)\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]\\
&\le-\eta\beta\mu\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]-\eta(1-\beta)\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]\\
&=-\eta\big[1-(1-\mu)\beta\big]\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]
\end{aligned} \tag{17}$$

where (a) follows by noticing that $\mathbb{E}[\mathbf{g}^t]=\nabla f(\mathbf{w}^t)$ and in (b) we notice that all the random variables $\{\tau_k\}_{k=1}^K$ possess the same distribution and hence unify them by introducing a random variable τ that satisfies $\tau=\tau_k$ in distribution. On the other hand, as the stochastic gradient has a bounded variance, Q_2 can be evaluated as

$$\begin{aligned}
Q_2&=\mathbb{E}\Big[\big\|\eta\sum_{k=1}^K p_k\mathbf{g}_k^{t-\tau_k}\big\|^2\Big]\\
&=\eta^2\mathbb{E}\Big[\big\|\sum_{k=1}^K p_k\mathbf{g}_k^{t-\tau_k}-\nabla f(\mathbf{w}^t)+\nabla f(\mathbf{w}^t)\big\|^2\Big]\\
&\le 2\eta^2\big(\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]+\sigma^2\big).
\end{aligned} \tag{18}$$

By taking (17) and (18) back into (16) and telescoping t from 0 to $T-1$, we have

$$\begin{aligned}
\mathbb{E}[f(\mathbf{w}^{T-1})]-\mathbb{E}[f(\mathbf{w}^0)]&\le L\sum_{t=0}^{T-1}\eta^2\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]\\
&+\sum_{t=0}^{T-1}L\eta^2\sigma^2-\big(1-(1-\mu)\beta\big)\sum_{t=0}^{T-1}\eta\,\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big].
\end{aligned} \tag{19}$$

Further rearranging the terms of the above inequality yields

$$\begin{aligned}
&\sum_{t=0}^{T-1}\big[\big(1-(1-\mu)\beta\big)\eta-L\eta^2\big]\mathbb{E}\big[\|f(\mathbf{w}^t)\|^2\big]\\
&\le f(\mathbf{w}_0)-\mathbb{E}\big[f(\mathbf{w}^{T-1})\big]+L\sigma^2\sum_{t=1}^{T}\eta^2.
\end{aligned} \tag{20}$$

Note that $\mathbb{E}[f(\mathbf{w}^T)]\ge f(\mathbf{w}^*)$ and $\eta=1/\sqrt{LT}$, and so we have

$$\begin{aligned}
\min_{0\le t\le T-1}\mathbb{E}\big[\|\nabla f(\mathbf{w}^t)\|^2\big]&\le\frac{f(\mathbf{w}_0)-f(\mathbf{w}^*)+L\sigma^2\sum_{t=1}^{T-1}\eta^2}{\sum_{t=0}^{T-1}\big[\big(1-(1-\mu)\beta\big)\eta-L\eta^2\big]}\\
&=\frac{f(\mathbf{w}_0)-f(\mathbf{w}^*)+\sigma^2}{\big(1-(1-\mu)\beta\big)\sqrt{T/L}-1}.
\end{aligned} \tag{21}$$

Finally, when T is taken to be sufficiently large, we have

$$\big(1-(1-\mu)\beta\big)\sqrt{T/L}-1\ge\frac{\big(1-(1-\mu)\beta\big)\sqrt{T}}{2\sqrt{L}} \tag{22}$$

and the result follows. □

Following Theorem 1, several observations can be made: (i) For non-convex objective functions, Federated SGD converges to stationary points on the order of $1/\sqrt{T}$; (ii) the staleness of parameters impacts the convergence rate via the multiplicative constant, which unveils that when the communication resources are abundant, i.e., the server can select many clients for parameter updates in each iteration, that leads to an increase in N which in turns reduces β and results in a faster convergence rate, and vice versa; and (iii) this result also provides further evidence to the claim that having more clients participate in each round of FL training is instrumental in speeding up the model convergence [21, 22].[1]

4. CONCLUSION

In this paper, we have carried out an analytical study toward a deeper understanding of the FL system. For the Federated SGD algorithm that uses both fresh and outdated gradients in the aggregation stage, we have shown that this implicitly introduces a momentum-like term during the update of global parameters. We have also analyzed the convergence rate of such an algorithm by taking into account the parameter staleness and communication resources. Our results have confirmed that increasing the number of selected clients in each communication round can accelerate the convergence of the FL algorithm through a reduction in the staleness of parameters. The analysis does not assume convexity of the objective function and hence is applicable to even the setting of deep learning systems. The developed framework reveals a link between staleness analysis and FL convergence rate, and may be useful for further research in this area.

[1] A few simulation examples that corroborate these observations are available in: https://person.zju.edu.cn/person/attachments/2022-01/01-1641711371-850767.pdf

5. REFERENCES

[1] D. Ramage S. Hampson H. B. McMahan, E. Moore and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Stat. (AISTATS)*, Fort Lauderdale, FL, USA, Apr. 2017, pp. 1273–1282.

[2] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surv. & Tut.*, vol. 22, no. 3, pp. 2031–2063, Third Quarter, 2020.

[3] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," Available as ArXiv:1610.05492, 2016.

[4] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019.

[5] A. Khaled, K. Mishchenko, and P. Richtárik, "Tighter theory for local sgd on identical and heterogeneous data," in *Proc. Int. Conf. Artif. Intell. Stat. (AISTATS)*, 2020, pp. 4519–4529.

[6] Z. Zhao, C. Feng, H. H. Yang, and X. Luo, "Federated learning-enabled intelligent fog-radio access networks: Fundamental theory, key techniques, and future trends," *IEEE Wireless Commun. Mag.*, vol. 27, no. 2, pp. 22–28, Apr. 2020.

[7] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[8] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y. J. A. Zhang, "The roadmap to 6G–AI empowered wireless networks," *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, Aug. 2019.

[9] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 317–333, Jan. 2020.

[10] H. H. Yang, A. Arafa, T. Q. S. Quek, and H. V. Poor, "Age-based scheduling policy for federated learning in mobile edge networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Barcelona, Spain, 2020, pp. 8743–8747.

[11] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[12] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

[13] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Math. Program.*, pp. 1–48, Dec. 2018.

[14] Y. Arjevani, O. Shamir, and N. Srebro, "A tight convergence analysis for stochastic gradient descent with delayed updates," in *Algorithmic Learning Theory*, 2020, pp. 111–132.

[15] T. Chen, G. B. Giannakis, T. Sun, and W. Yin, "LAG: Lazily aggregated gradient for communication-efficient distributed learning," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, CANADA, Dec. 2018.

[16] I. Mitliagkas, C. Zhang, S. Hadjis, and C. Ré, "Asynchrony begets momentum, with an application to deep learning," in *Proc. 54th Annu. Allerton Conf. Commun., Control, and Comput. (Allerton)*, Monticello, IL, Sept. 2016, pp. 997–1004.

[17] W. Liu, L. Chen, Y. Chen, and W. Zhang, "Accelerating federated learning via momentum gradient descent," *IEEE Trans. Parallel and Distrib. Syst.*, vol. 31, no. 8, pp. 1754–1766, Aug. 2020.

[18] Z. Huo, Q. Yang, B. Gu, L. Carin, and H. Huang, "Faster on-device training using new federated momentum algorithm," Available as ArXiv:2002.02090, 2020.

[19] L. Zhu, H. Lin, Y. Lu, Y. Lin, and S. Han, "Delayed gradient averaging: Tolerate the communication latency for federated learning," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2021.

[20] W. Dai, Y. Zhou, N. Dong, H. Zhang, and E. P. Xing, "Toward understanding the impact of staleness in distributed machine learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, New Orleans, Louisiana, May 2019, pp. 1–6.

[21] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.

[22] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun.*, Shanghai, China, May 2019, pp. 1–7.