Accelerating Federated Learning for Edge Intelligence using Conjugated Central Acceleration with Inexact Global Line Search

Lei Zhao, *Member, IEEE,* Lin Cai*, *Fellow, IEEE,* and Wu-Sheng Lu, *Life Fellow, IEEE*

***Abstract*—Driven by the increasing demand for real-time, low-latency learning processes and the ever-growing emphasis on data privacy, Federated Learning (FL) enabled edge intelligence emerges as a promising decentralized learning paradigm at the edge of the network, which empowers collaborative model training on edge agents, allowing them to make intelligent decisions locally without relying solely on centralized cloud servers. To enhance the training efficiency of edge agents and alleviate communication burdens, we propose a novel technique called Conjugated Central Acceleration with Inexact Line Search enabled Federated Stochastic Variance Reduced Gradient (CLSFSVRG). Conjugate Central Acceleration leverages conjugate gradient technique to efficiently utilize the training information from multiple edge agents by additional updating efforts in the central server, thereby enhancing the convergence rates of the global model and reduce the local training burden. Inexact Line Search optimizes the step size for model updates, striking a balance between precision and computational efficiency. Simulation results demonstrate that the proposed scheme outperforms the state-of-the-art FL algorithms, achieving superior performance in terms of higher test accuracy and faster convergence speed. Remarkably, our approach reduces communication costs by an impressive 82.4%, while still achieving a test accuracy of 96.5%. By allowing a small portion of edge agents to participate, CLSFSVRG exhibits higher robustness without compromising the test accuracy. Moreover, the fast convergence speed achieved with a limited number of participating edge agents contributes to significant reductions in edge computing cost during the training procedure.**

***Index Terms*—Edge Intelligence, Federated Learning, Conjugate Central Acceleration, Global Inexact Line Search**

I. Introduction

In real-time machine learning applications such as for connected robots or intelligent vehicles, ensuring high reliability and low latency decision making are crucial. However, transporting large volumes of time-sensitive data to the cloud for analysis and learning can be too costly and even impractical due to long and unpredictable WAN delays [1] [2]. To address this challenge, the paradigm of edge intelligence has emerged by processing and analyzing data at the network edge. This approach significantly enhances efficiency and flexibility in real-time machine learning applications.

While edge intelligence offers a promising avenue for enhancing real-time machine learning applications, there are critical challenges inherent in its implementation. One major concern is that the limited local data may be insufficient for achieving high-quality model training, which results in the need for augmenting local training. Given users' privacy concern, safeguarding sensitive data during distributed processing becomes a paramount consideration. Furthermore, the resource-intensive nature of model training poses energy challenges for edge agents, demanding innovative solutions for sustainable operation [3] [4] [5] [6].

Federated learning (FL) enabled edge intelligence provides a desirable platform as it tackles the challenges facing large-scale learning while ensuring data privacy [7] [8] [9] [10]. It enables real-time and privacy-preserving AI capabilities at the edge of the network with applications in IoT, healthcare, autonomous vehicles, and more.

There are many challenges for FL when dealing with edge intelligence applications. These include keeping the communications between the edge agents and the central server (often reside in cloud) to a low volume, and at the same time protecting the security and privacy of the data at the local agents [11]. We need to address specific challenges inherent in federated learning, such as communication overhead, convergence speed, and robustness to heterogeneity and dynamic participation among edge agents. Traditional FL algorithms often suffer from slow convergence and instability, especially in dynamic edge intelligence environments, where limited edge agents participation leads to high variances in local model updates and deteriorates the convergence in federated training [12].

Our main contributions lie in ensuring rapid convergence and highly accurate learning across diverse local datasets with a novel approach called Conjugate Central Acceleration with Inexact Line Search integrating Federated Stochastic Variance Reduced Gradient (CLSFSVRG). Based on the updating information from multiple edge agents, the central server estimates the curvature information of the global objective function and applies conjugated directions to achieve more efficient and stable updates, improving the convergence rate and reducing the computational burden on edge agents. Furthermore, we design the inexact line search method for the central model updating to enhance the robustness of our algorithm, particularly in scenarios with noisy updates and varying data distributions, by dynamically adjusting the step size to maintain stability. Moreover, integration with the global anchor gradient in local updates enhances the stability and speed of convergence, particularly beneficial for heterogeneous local datasets. The proposed CLSFSVRG method achieves higher accuracy in fewer training rounds, even with a limited number of partici-

L. Zhao, L. Cai, and W-S. Lu are with Dept. of Electrical & Computer Engineering, University of Victoria, 3800 Finnerty Road, Victoria, BC, V8P 5C2, Canada. Corresponding author: Lin Cai (E-mail: cai@ece.uvic.ca). This work was supported in part by the Natural Sciences and Engineering Research Council of Canada and Compute Canada.

pating edge agents in each round, resulting in greater flexibility and robustness and substantially lower communication and edge computing costs. Simulation results demonstrate that CLSFSVRG outperforms state-of-the-art methods, achieving 98% test accuracy with 50% fewer iterations.

The rest of this paper is organized as follows. The related works are summarized in Section II. Section III formulates the federated objective among edge agents and provides basic properties of the local and global objective functions. The proposed CLSFSVRG is explained in Section IV. Simulation results are presented in Section V followed by the conclusions and further research issues in Section VI.

II. Related Work

FL-enabled edge intelligence is particularly well-suited for applications where data privacy, low latency, and real-time decision-making are critical, such as in IoT, healthcare, autonomous vehicles, and intelligent manufacturing [13] [14]. By combining the benefits of FL and edge intelligence, this approach empowers edge devices to become intelligent, adaptive, and privacy-preserving entities, ushering in a new era of decentralized AI and data processing [15]. Instead of aggregating data to a central server for training, FL enables the training of models directly on edge devices [16]. This decentralized approach ensures data privacy and reduces the need for data transmission, as raw data remains on the edge devices [17].

There are many challenges for the cooperation among multiple edge agents by FL. First, the communication among the edge agents and the cloud server is very heavy to exchange information in each training iteration. To reduce the communication cost, edge agents can perform multiple local model updates before communicating with the cloud server [18] [10]. Second, statistical heterogeneity of the local data sets is common as each edge agent can differ from its peers in multiple aspects [7] [19]. The FL with proximal term, namely FedProx, can improve the performance of FL with data heterogeneity by adding a proximal term to local objective functions [19]. SCAFFOLD [20] uses control variates to correct local updates, mitigating client drift and aligning local updates with the global objective in non-i.i.d. settings. MimeSVRG [21] combines the SVRG framework with periodic averaging and momentum techniques to enhance stability and convergence rates in environments with high data variability. LoSAC [22] employs adaptive control variates and sparse communication to simultaneously reduce variance and communication overhead. However, these FL solutions still have challenges to satisfy the stringent requirements of edge intelligence in terms of robust training performance with dynamic edge agents participation, restricted and heterogeneous local resources and low communication cost and fast convergence speed.

Adaptive learning rate and step length are crucial components in enhancing federated learning. Work [23] adjusts learning rates to accommodate the heterogeneous local data distributions. FedUR [24] leverages adaptive centralized learning optimizers to improve convergence rates. SAFA [25] is a semiasynchronous protocol that allows a subset of edge agents to send updates to the server asynchronously to reduce waiting times. HiFlash [26] employs a hierarchical federated learning framework with adaptive staleness control. In addition to asynchronous updates, agent selection strategies can further enhance communication efficiency [27] [28], considering channel conditions, computational capabilities, and data quality for resource-constrained edge computing environments.

TABLE I
Comparison of Federated Learning Algorithms

Algorithm	Conver. Speed	Comm. Efficiency	Var. Reduc.	Adapt. to Limit Part.
FedAvg [10]	Moderate	Low	No	Low
FedProx [19]	Moderate	Medium	No	Medium
SCAFFOLD [20]	Fast	High	Yes	Medium
MimeSVRG [21]	Fast	High	Yes	Medium
LoSAC [22]	Fast	High	Yes	High
CLSFSVRG	**Very Fast**	**High**	**Yes**	**High**

Different from the above, we incorporate a novel variance reduction technique by scaling the auxiliary local gradient based on the number of non-zero features of the samples. This scaling ensures that the updates consider the sparsity patterns in the local datasets, leading to more balanced and accurate global model updates. Our method includes a conjugate gradient-based global model updating mechanism, which aims to design a finite set of conjugate updating directions. This technique boosts the global model updating process by ensuring Hessian based orthogonality in the search directions, thereby enhancing convergence efficiency. This aspect is a significant departure from the adaptive optimizers discussed in the referenced works. We employ an inexact line search strategy to fine-tune the global learning rate during model aggregation. This approach allows for more flexible and adaptive adjustments compared to the fixed or predefined schedules typically used in traditional adaptive optimization methods. By integrating these novel components, our approach not only adapts the learning rates and step lengths but also ensures a more robust and efficient federated learning process. The combination of variance reduction through gradient scaling, conjugate gradient-based updating, and inexact line search positions our method as a significant advancement in the field of federated optimization.

III. System Model and Federated Objective

As illustrated in Fig. 1, massive data are generated from numerous smart sensors. The sensing data will be delivered to edge agents which have stronger processing units and larger memories and storage space. The edge agents can guarantee low-latency local services for data analysis supporting intelligent applications in various fields, including smart cities, industrial IoT, autonomous vehicles, healthcare, and more. It enables the development of intelligent systems that can process and respond to data in real-time, bringing increased efficiency, responsiveness, and autonomy to a wide range of applications [4].

We use E to denote the set of all edge agents and the number of edge agents is denoted by $|E|$. Each edge agent collects the information sensed by the devices in vicinity

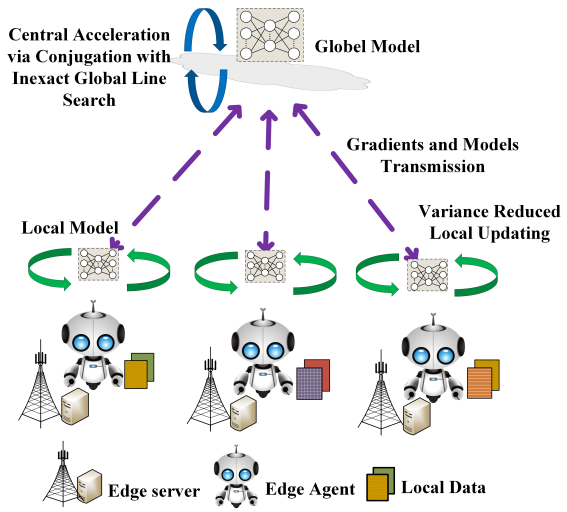


Fig. 1. Federated Learning enabled Edge Intelligence Architecture.

which forms the local data set. Edge agent i possesses n_i local training samples and we use P_i to denote the local data set where $i = 1, 2, \cdots, |E|$ and $P_i \cap P_j = \emptyset$ whenever $i \neq j$. The entire training data samples can be represented by $\{\boldsymbol{x_k}, y_k\}_{k=1}^{n}$ where $n = \sum_{i=1}^{|E|} n_i$. The local training objective of edge agent i can be represented by

$$f_i(\boldsymbol{w}) = \frac{1}{n_i} \sum_{k \in P_i} F_k(\boldsymbol{w}) \quad i = 1, \cdots, |E|, \tag{1}$$

where $\{F_k(\boldsymbol{w})\}_{k \in P_i}$ denotes the empirical loss over the local data set $\{\boldsymbol{x_k}, y_k\}_{k=1}^{P_i}$. The federated optimization objective among the edge agents in E can be formulated as

$$\underset{\boldsymbol{w} \in \mathbb{R}^q}{\text{minimize}} \quad f(\boldsymbol{w}) = \sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\boldsymbol{w}). \tag{2}$$

IV. Conjugate Central Acceleration with Inexact Line Search integrating Federated Stochastic Variance Reduced Gradient

In this section, we design the Conjugate Central Acceleration with Inexact Line Search integrating Federated Stochastic Variance Reduced Gradient (CLSFSVRG). The conjugate federated updating selects the successive global model updating directions as a conjugate version of the successive anchor gradients obtained in the federated optimization progresses. The global model updating directions are determined sequentially at each federation round. In each round, the central server evaluates the current negative anchor gradient and adds to it a linear combination of the previous conjugate directions to obtain a new conjugate direction along which to update the current global model. The learning rate in the central acceleration is obtained by inexact line search.

A. Global Anchor and Local Auxiliary Gradients

Due to the dynamic participation of edge agents, at the beginning of the r-th round, only a subset of edge agents $\boldsymbol{S}^r \subseteq [E]$ with size $|\boldsymbol{S}^r| = S$ contributes to the global model update at the central server. Edge agent i randomly selects a batch of training samples to calculate $\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)$ in each local updating, where we define $\hat{\boldsymbol{w}}_{i,k}^r$ as the i-th local model in the r-th global round and the k-th local iteration. The local gradient $\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)$ is an unbiased estimation of $\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)$. According to Jensen's inequality, we obtain the upper bound of the stochastic local gradient as

$$E[||\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)||^2] \leq ||\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)||^2. \tag{3}$$

We use $\boldsymbol{w}_i^*$ to denote the optimal local model for client i where $||\nabla f_i(\boldsymbol{w}_i^*)||^2 = 0$. Then, we obtain the upper bound of the variance of the local gradient as

$$E[||\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)||^2] \leq ||\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) - \nabla f_i(\boldsymbol{w}_i^*)||^2. \tag{4}$$

According to (38) in the Appendix, we have

$$||\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) - \nabla f_i(\boldsymbol{w}^*)||^2 \leq L_i^2 ||\hat{\boldsymbol{w}}_{i,k-1}^r - \boldsymbol{w}_i^*||^2, \tag{5}$$

where L_i refers to the upper bound of the eigenvalues of the Hessian of the i-th local objective function, used to quantify the local curvature information.

However, in practical FL with heterogeneous local data sets, all local optimization procedures will converge to different solutions. To ease the biased local gradients and make efficient progress in the local training procedure, we first evaluate the global anchor gradient. The current global model $\boldsymbol{w^{r-1}}$ is distributed to the current participated edge agents in $\boldsymbol{S}^r$. The available edge agents in $\boldsymbol{S}^r$ evaluate their full local gradients and transmit $\{\nabla f_i(\boldsymbol{w}^{r-1})\}_{i \in \boldsymbol{S}^r}$ to the central server to aggregate the current anchor gradient as

$$\boldsymbol{g}(\boldsymbol{w}^{r-1}) = \sum_{i \in \boldsymbol{S}^r} \frac{n_i}{n^r} \nabla f_i(\boldsymbol{w}^{r-1}), \tag{6}$$

TABLE II
Description of Notations

NOTATION	DESCRIPTION
$f(\boldsymbol{w})$	The global objective function
$f_i(\boldsymbol{w})$	The i-th local objective function
$\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)$	The local gradient from individual training sample or a batch of the training samples
$\boldsymbol{w}^{r-1}$	The current global model in the r-th round
$\hat{\boldsymbol{w}}^{r-1}$	The global model after central acceleration in the r-th round
$\hat{\boldsymbol{w}}_{i,k}^r$	The i-th local model in the r-th global round and the k-th local iteration
E	The set of total edge agents
$\boldsymbol{S}^r$	The selected edge agent subset in the r-th round
α_l^i	The local learning rate of edge agent i
$\boldsymbol{d}_{i,k}^r$	The local model updating direction of edge agent i in r-th round and k-th local iteration
n_i	The number of local samples in edge agent i
n^r	The total number of samples to calculate the anchor gradient in the r-th round
$\boldsymbol{g}(\boldsymbol{w}^{r-1})$	The anchor gradient in the r-th round
n^j	The number of samples with nonzero j-th feature
n_i^j	The number of samples in the local data set of edge agent i with nonzero j-th feature
q	The number of model parameters
η_r	The step size of r-th conjugate federated updating
$\boldsymbol{d}_r$	The conjugated directions in the r-th training round

where n^r denotes the number of samples from all edge agents in subset $\boldsymbol{S}^r$. Given the dynamic edge agents participation, the anchor gradient $\boldsymbol{g}(\boldsymbol{w}^{r-1})$ obtained by an entire data pass of all the accessible local data sets is an unbiased estimation of the full global gradient $\nabla f(\boldsymbol{w}^{r-1})$, i.e., $E[\boldsymbol{g}(\boldsymbol{w}^{r-1})] = \nabla f(\boldsymbol{w}^{r-1})$.

The current anchor gradient $\boldsymbol{g}(\boldsymbol{w}^{r-1})$ distributed to all the edge agents in subset $\boldsymbol{S}^r$ can be applied to force the local gradient to be unbiased for the local training procedure, where the stochastic local gradient $\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1})$ is replaced by the auxiliary local gradient as

$$\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1}) + \boldsymbol{g}(\boldsymbol{w}^{r-1}), \tag{7}$$

where by using $\boldsymbol{g}_i(\boldsymbol{w}^{r-1})$, it adjusts the local gradient $\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1})$ to align more closely with the global gradient direction. This ensures that the updates made during local training are consistent with the overall objective of minimizing the global loss function. Based on the auxiliary local gradient, the stochastic update in edge agent i yields an unbiased estimate of the global gradient $\nabla f(\boldsymbol{w})$ as

$$\nabla f(\hat{\boldsymbol{w}}^{r-1}) \approx E[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1}) + E[\boldsymbol{g}(\boldsymbol{w}^{r-1})]]. \tag{8}$$

The auxiliary local gradient (7) is crafted to address the challenge of high variance in local gradient estimates due to non-i.i.d data distributions across multiple edge agents. By incorporating both the local gradient difference and the current global gradient, the auxiliary local gradient ensures a balanced update that mitigates local variance and aligns with global optimization objectives. The primary advantage of defining the auxiliary local gradient in this manner is its ability to significantly reduce the variance of local gradient estimates. By combining the local gradient difference with the global anchor gradient, our method ensures that updates from different edge agents are more consistent and less noisy, which leads to more stable and faster convergence of the global model.

Since $E[\boldsymbol{g}(\boldsymbol{w}^{r-1})]$ is constant during the local updating, we obtain

$$\begin{aligned} &\text{Var}(\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1}) + E[\boldsymbol{g}(\boldsymbol{w}^{r-1})]) \\ &= \text{Var}(\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})), \end{aligned} \tag{9}$$

which leads to

$$\begin{aligned} \text{Var}(\nabla f(\hat{\boldsymbol{w}}^{r-1})) &\approx \text{Var}(\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})) \\ &\le E[||\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})||^2], \end{aligned} \tag{10}$$

where $||E[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})]||^2 > 0$. According to Jensen's inequality, we obtain

$$\begin{aligned} &E[||\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})||^2] \\ &\le ||E[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1})] - E[\boldsymbol{g}_i(\boldsymbol{w}^{r-1})]||^2. \end{aligned} \tag{11}$$

Since the stochastic local gradients are unbiased to the full local gradients, we obtain

$$E[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})] = \nabla f_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \nabla f_i(\boldsymbol{w}^{r-1}), \tag{12}$$

which leads to

$$\begin{aligned} \text{Var}(\nabla f(\hat{\boldsymbol{w}}^{r-1})) &\le ||\nabla f_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \nabla f_i(\boldsymbol{w}^{r-1})||^2 \\ &\le L_i^2 ||\hat{\boldsymbol{w}}^r_{i,k-1} - \boldsymbol{w}^{r-1}||^2. \end{aligned} \tag{13}$$

Over a sequence of the global model updating, the variance would go to zero over time since the global model estimation becomes closer to the converged point.

B. Local Updating with Variance Reduction

The data available locally may exhibit a specific pattern. The auxiliary local gradient and the aggregation step need to be carefully tuned due to the large variance among local data sets. To enforce the auxiliary local gradient to be of the correct magnitude, it is scaled carefully by the number of non-zero features of the samples. This scaling is crucial because the local data sets often have varying sparsity levels, and features that appear less frequently need to be weighted appropriately to ensure balanced updates. By scaling the gradients based on the number of non-zero features, we account for the differences in feature representation across edge agents, leading to more accurate and stable global model updates.

The number of samples in the local data set of edge agent i with nonzero j-th feature is denoted by n_i^j. After going through their local data sets, edge agents send the number of local nonzero j-th feature $\{n_i^j\}_{i\in E}$ to the central server, the central server can obtain the number of samples with nonzero j-th feature over all local data sets as

$$n^j = \sum_{i\in E} n_i^j. \tag{14}$$

The variance between the gradient w.r.t. the current local model $\hat{\boldsymbol{w}}^r_{i,k-1}$ and global model $\hat{\boldsymbol{w}}^{r-1}$ is scaled by diagonal matrix $\boldsymbol{\Lambda}_i$ as

$$\boldsymbol{\Lambda}_i[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\hat{\boldsymbol{w}}^{r-1})], \tag{15}$$

where

$$\boldsymbol{\Lambda}_i = \text{diag}\left(\left\{\frac{n^j \cdot n_i}{n \cdot n_i^j}\right\}_{j=1,\cdots,q}\right). \tag{16}$$

The scaled updating direction in the k-th local step of edge agent i can be obtained as

$$\boldsymbol{d}^r_{i,k} = -(\boldsymbol{\Lambda}_i[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})] + \boldsymbol{g}(\boldsymbol{w}^{r-1})), \tag{17}$$

and the local model update of the edge agent i in the k-th local step can be formulated as

$$\hat{\boldsymbol{w}}^r_{i,k} = \hat{\boldsymbol{w}}^r_{i,k-1} + \alpha_l^i \boldsymbol{d}^r_{i,k}. \tag{18}$$

The local training procedure guided by these unbiased gradients can be formulated as

$$\hat{\boldsymbol{w}}^r_{i,k} = \hat{\boldsymbol{w}}^{r-1} + \sum_{k=1}^{K} \alpha_l^i \boldsymbol{d}^r_{i,k}. \tag{19}$$

C. Conjugated Global Model Updating with Inexact Line Search

1) Global Model Updating: The idea behind the conjugate federated central updating is to design a finite set of conjugate updating directions $\{\boldsymbol{d}_i\}_{i=0}^{r-1}$ in the central server to boost the global model updating where $\boldsymbol{d}_i^T \nabla^2 f(\boldsymbol{w}) \boldsymbol{d}_j = 0$ for all $i \neq j$ and $\boldsymbol{d_0} = -\boldsymbol{g}(\boldsymbol{w^0})$. After the local training procedure, the edge agents send $\{\hat{\boldsymbol{w}}^r_{i,k}\}_{i\in \boldsymbol{S}^r}$ to the central server. Then, the

drift from the current global model is scaled based on whether the specific feature appears in one local data set or not. The intuitive idea is that if a feature appears less frequently in local data sets, we want to enhance the update amount to the gradient related to this feature more. The number of edge agents that contain data samples with nonzero j-th feature is obtained by

$$\omega^j = \sum_{i \in E} 1_{n_i^j \neq 0}. \tag{20}$$

The scaling diagonal matrix for model aggregation can be defined as

$$\boldsymbol{A} = \text{diag}\left(\left\{ \frac{|S|}{\omega^j} \right\}_{j=1,\cdots,q} \right). \tag{21}$$

The global model update with global learning rate α_g as

$$\boldsymbol{w}^r = \hat{\boldsymbol{w}}^{r-1} + \frac{\alpha_g}{S} \boldsymbol{A} \sum_{i \in \boldsymbol{S}^r} \frac{n_i}{n^r} \sum_{k=1}^{K} \alpha_l^i \boldsymbol{d}_{i,k}^r. \tag{22}$$

After the global model $\boldsymbol{w}^r$ is aggregated, the central server randomly selects the next participation edge agent set $\boldsymbol{S}^{r+1}$ and distribute $\boldsymbol{w}^r$ to these edge agents to evaluate the anchor gradient

$$\boldsymbol{g}(\boldsymbol{w}^r) = \sum_{i \in \boldsymbol{S}^{r+1}} \frac{n_i}{n^{r+1}} \nabla f_i(\boldsymbol{w}^r), \tag{23}$$

Then, we design

$$\beta_r = \frac{\boldsymbol{g}(\boldsymbol{w}^r)^T \boldsymbol{g}(\boldsymbol{w}^r)}{\boldsymbol{g}(\boldsymbol{w}^{r-1})^T \boldsymbol{g}(\boldsymbol{w}^{r-1})}. \tag{24}$$

The conjugate direction in the federation is designed as

$$\boldsymbol{d}_r = -\boldsymbol{g}(\boldsymbol{w}^r) + \beta_r \boldsymbol{d}_{r-1}. \tag{25}$$

2) Global Inexact Line Search: We introduce the global inexact line search method to explore the inexact step length along the central acceleration updating direction, which not only is computing efficient but also can achieve better performance when the condition number of $\nabla^2 f(\boldsymbol{w})$ is large. The central acceleration transformed to the problem of determining the step length parameter η_r after the search direction $\boldsymbol{d}_r$ is determined. In the central acceleration, the target is to update the current global model $\boldsymbol{w}^r$ to the accelerated global model $\hat{\boldsymbol{w}}^r$ along the search direction to minimize the objective function value $f(\boldsymbol{w})$.

Our design is based on the Taylor expansion of the global objective function where we design a perfect symmetric quadratic function associated with the accelerated global model $\hat{\boldsymbol{w}}^r$, i.e., the condition number of this designed quadratic function's Hessian is 1. We use η_r to control the curvature in every dimension, which is designed as

$$\begin{aligned} \psi_{\eta_r}(\hat{\boldsymbol{w}}^r) = &\frac{1}{2\eta_r} \left\| \hat{\boldsymbol{w}}^r - \boldsymbol{w}^r \right\|^2 \\ &+ \boldsymbol{g}(\boldsymbol{w}^r)^T (\hat{\boldsymbol{w}}^r - \boldsymbol{w}^r) + f(\boldsymbol{w}^r). \end{aligned} \tag{26}$$

Because the condition number of $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ is 1, the contour of $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ is perfect circles, and we can easily find the optimal solution $\hat{\boldsymbol{w}}^r$ to $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ by

$$\nabla \psi_{\eta_r}(\hat{\boldsymbol{w}}^r) = \frac{1}{\eta_r}(\hat{\boldsymbol{w}}^r - \boldsymbol{w}^r) + \boldsymbol{g}(\boldsymbol{w}^r) = 0. \tag{27}$$

This function parameterized by η_r gives us a point $\hat{\boldsymbol{w}}^r$ which is the optimal solution for $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$, and this point is paired with the parameter η_r as

$$\psi_{\eta_r}(\hat{\boldsymbol{w}}^r) = -\frac{\eta_r}{2} \boldsymbol{g}(\boldsymbol{w}^r)^T \boldsymbol{g}(\boldsymbol{w}^r) + f(\boldsymbol{w}^r). \tag{28}$$

We compare the optimal value of the designed quadratic function $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ and the objective function $f(\boldsymbol{w})$ at $\hat{\boldsymbol{w}}^r$ by tuning η_r to increase the curvature of $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ to a point that the lowest value of $\psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$ is above $f(\hat{\boldsymbol{w}}^r)$.

Instead of communicating with all edge agents to obtain the global objective function and compare $f(\hat{\boldsymbol{w}}^r) \geq \psi_{\eta_r}(\hat{\boldsymbol{w}}^r)$, we build a validation data set in the central server which can be obtained from held-out edge agents [29]. For the inexact line search in the central server, we define a dampening factor $\gamma \in (0,1)$ and initialize $\hat{\eta} = 1$ at the beginning of each round. The central server update $\hat{\eta} = \gamma \hat{\eta}$ and set $\eta_r = \hat{\eta}$ until the condition

$$\frac{\eta_r}{2} \boldsymbol{g}(\boldsymbol{w}^r)^T \boldsymbol{g}(\boldsymbol{w}^r) \geq f_v(\boldsymbol{w}^r) - f_v(\hat{\boldsymbol{w}}^r). \tag{29}$$

is satisfied, where $f_v(\boldsymbol{w})$ is the global validation function to replace $f(\boldsymbol{w})$ in order to make fast hyper-parameter tuning in the central acceleration without causing severe communication burden. After (29) is satisfied, the global model $\boldsymbol{w}^r$ is updated to $\hat{\boldsymbol{w}}^r$ by

$$\hat{\boldsymbol{w}}^r = \boldsymbol{w}^r + \eta_r \boldsymbol{d}_r, \tag{30}$$

which is shared to the selected edge agents as initialized local models $\{\hat{\boldsymbol{w}}_{i,0}^{r+1} = \hat{\boldsymbol{w}}^r\}_{i \in \boldsymbol{S}^{r+1}}$ for the next round training. This method is suitable to the situation where there are many local minimums and maximums for the global objective function. The details of CLSFSVRG is illustrated in **Algorithm 1**.

Algorithm 1 Conjugated central acceleration with inexact Line Search enabled FSVRG (CLSFSVRG).

Initial global model $\boldsymbol{w}^0$, $\beta_0 = 0$ and $\eta_0 = 1$
for $r \leftarrow 1$ to T **do**
  Randomly select a subset $\boldsymbol{S}^r$ out of E
  Compute and transmit $\{\nabla f_i(\boldsymbol{w}^{r-1})\}_{i \in \boldsymbol{S}^r}$ to the central server
  $\boldsymbol{g}(\boldsymbol{w}^{r-1}) = \sum_{i \in \boldsymbol{S}^r} \frac{n_i}{n^r} \nabla f_i(\boldsymbol{w}^{r-1})$
  $\rho = \frac{\eta_{r-1}}{2} \boldsymbol{g}(\boldsymbol{w}^{r-1})^T \boldsymbol{g}(\boldsymbol{w}^{r-1})$
  $\hat{\boldsymbol{w}}^{r-1} = \boldsymbol{w}^{r-1} + \eta_{r-1} \boldsymbol{d}_{r-1}$
  while $\rho \geq f_v(\boldsymbol{w}^{r-1}) - f_v(\hat{\boldsymbol{w}}^{r-1})$ **do**
    $\eta_{r-1} = \gamma \eta_{r-1}$
    $\hat{\boldsymbol{w}}^{r-1} = \boldsymbol{w}^{r-1} + \eta_{r-1} \boldsymbol{d}_{r-1}$
  end while
  Distribute $\boldsymbol{g}(\boldsymbol{w}^{r-1})$ and $\hat{\boldsymbol{w}}^{r-1}$ to edge agents in $\boldsymbol{S}^r$
  for $i \in \boldsymbol{S}^r$ **in parallel do**
    $\triangle \hat{\boldsymbol{w}}_i^r \leftarrow$ **Procedure 1** ($\boldsymbol{g}(\boldsymbol{w}^{r-1})$, $\hat{\boldsymbol{w}}^{r-1}$, i)
  end for
  $\boldsymbol{w}^r = \hat{\boldsymbol{w}}^{r-1} + \frac{\eta_{r-1}}{S} \boldsymbol{A} \sum_{i \in \boldsymbol{S}^r} \frac{n_i}{n^r} \triangle \hat{\boldsymbol{w}}_i^r$
  Randomly select a subset $\boldsymbol{S}^{r+1}$ out of E
  $\boldsymbol{g}(\boldsymbol{w}^r) = \sum_{i \in \boldsymbol{S}^{r+1}} \frac{n_i}{n^{r+1}} \nabla f_i(\boldsymbol{w}^r)$
  $\beta_r = \frac{\boldsymbol{g}(\boldsymbol{w}^r)^T \boldsymbol{g}(\boldsymbol{w}^r)}{\boldsymbol{g}(\boldsymbol{w}^{r-1})^T \boldsymbol{g}(\boldsymbol{w}^{r-1})}$
  $\boldsymbol{d}_r = -\boldsymbol{g}(\boldsymbol{w}^r) + \beta_r \boldsymbol{d}_{r-1}$
end for

Procedure 1 Local-Updating ($\boldsymbol{g}(\boldsymbol{w}^{r-1})$, $\hat{\boldsymbol{w}}^{r-1}$, i)

$\hat{\boldsymbol{w}}^r_{i,0} = \hat{\boldsymbol{w}}^{r-1}$
Normalize the local learning rate $\alpha^i_l = \frac{\alpha_l}{n_i}$
for $k \leftarrow 0$ to K **do**
  Compute $\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1})$ and $\boldsymbol{g}_i(\boldsymbol{w}^{r-1})$
  $\boldsymbol{d}^r_{i,k} = -(\boldsymbol{\Lambda}_i[\boldsymbol{g}_i(\hat{\boldsymbol{w}}^r_{i,k-1}) - \boldsymbol{g}_i(\boldsymbol{w}^{r-1})] + \boldsymbol{g}(\boldsymbol{w}^{r-1}))$
  $\hat{\boldsymbol{w}}^r_{i,k} = \hat{\boldsymbol{w}}^r_{i,k-1} + \alpha^i_l \boldsymbol{d}^r_{i,k}$
end for
Return $\triangle\hat{\boldsymbol{w}}^r_i = \hat{\boldsymbol{w}}^r_{i,k} - \hat{\boldsymbol{w}}^{r-1}$

CLSFSVRG offers several key advantages to address challenges in FL. Firstly, it employs conjugate federated central acceleration, which enables more efficient exploration of the optimization landscape by incorporating conjugate search directions globally. This approach accelerates convergence rates and enhances optimization performance. Additionally, the adoption of inexact line search within CLSFSVRG facilitates adaptive adjustment of step lengths along the central acceleration updating direction. This mechanism allows the algorithm to dynamically tune step lengths, improving its ability to navigate complex optimization surfaces and converge to optimal solutions more effectively. Furthermore, CLSFSVRG mitigates the impact of noisy gradients and reduces overall variance during model updates through the use of anchor gradients in the central server and variance reduction in edge agents. Moreover, the algorithm's robustness to heterogeneity is another key factor contributing to its superior performance. CLSFSVRG accommodates variations in local data distributions and computational resources across edge agents through techniques such as global anchor gradient evaluation and scaled local updates.

V. Experiments

In our experiments, we first employed the MNIST dataset, featuring 10 classification categories, to assess the efficacy of our proposed algorithm. Subsequently, we applied the CIFAR-10 dataset, providing supplementary validation for the performance of our proposed method. To generate local training datasets for 400 edge agents, we introduce a non-i.i.d distribution to simulate heterogeneity. The non-i.i.d local datasets are constructed using the symmetric Dirichlet distribution with a parameter $\xi = 0.5$, where a smaller ξ intensifies the degree of heterogeneity in the data distribution [30].

We conduct a comparative analysis between our proposed CLSFSVRG algorithm and FedAvg [10], FedProx [19], SCAFFOLD [20], MimeSVRG [21], and LoSAC [22]. Furthermore, we introduce Conjugated FSVRG (CFSVRG) as a baseline, exclusively applying conjugated acceleration in the central server. This inclusion aims to highlight the advancements achieved by incorporating inexact line search in the central acceleration. These algorithms represent a diverse range of approaches in federated learning, allowing for a comprehensive evaluation of our proposed method. The analysis focuses on accuracy, communication overhead, convergence speed, and robustness to heterogeneity in datasets of edge agents. Our proposed method exhibits superior performance compared to the baseline algorithms, achieving higher accuracy with lower communication overhead and faster convergence.

A. Convergence Performance

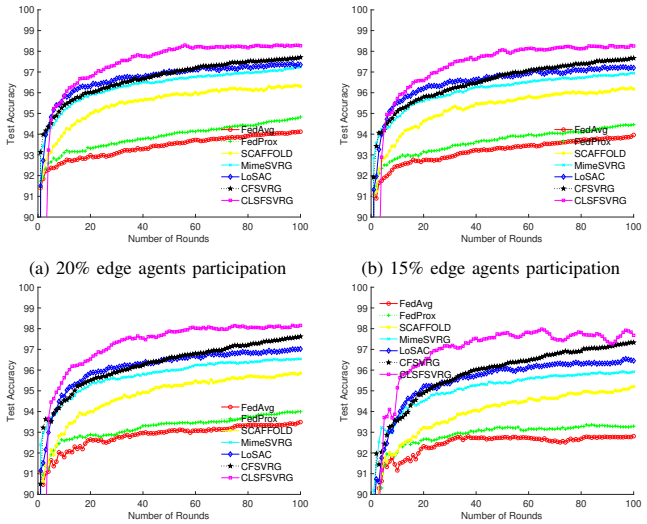
We first investigate the performance of the proposed CLSFSVRG with varying participation ratios of edge agents. As shown in Fig. 2, we compare it against six baseline algorithms: FedAvg [10], FedProx [19], SCAFFOLD [20], MimeSVRG [21], LoSAC [22], and CFSVRG. The participation ratios range from 20% to 5% out of 400 total edge agents, with the local learning rate set to $\alpha_l = 0.3$.

The results demonstrate that CLSFSVRG achieves higher test accuracy significantly faster than the other baselines. Specifically, CLSFSVRG reaches a test accuracy of 98% within 48, 50, 59, and 65 rounds for participation ratios of 20%, 15%, 10%, and 5%, respectively. In contrast, none of the other baselines achieve the same level of test accuracy even after 100 rounds. Furthermore, CLSFSVRG outperforms CFSVRG, which lacks the inexact linear search in the central acceleration. The benefits of the conjugated central acceleration in CLSFSVRG become more obvious as the participation ratio decreases. As shown in Fig. 2(c) and Fig. 2(d), other baseline FL algorithms exhibit a clear decline in test accuracy performance with lower participation ratios. Both CLSFSVRG and CFSVRG maintain comparable performance to scenarios with a 20% participation ratio, with CLSFSVRG achieving higher effectiveness thanks to its inexact line search in the central acceleration. CLSFSVRG surpasses all algorithms and achieves the highest test accuracy in fewer than 10 federated training rounds.

Moreover, CLSFSVRG exhibits higher stability across different participation ratios compared to other approaches. This resilience is particularly evident when fewer edge agents participate, as shown in Fig. 2(d) with only 5% participation. In such scenarios, the performance of baseline methods significantly deteriorates, while CLSFSVRG remains robust. For instance, with 5% edge agent participation, MimeSVRG achieves a test accuracy of only 95.93% at 100 rounds, CFSVRG reaches 96.02% at 40 rounds, and CLSFSVRG achieves 96.14% within just 16 training rounds. In comparison, LoSAC achieves 96.13% accuracy in 61 rounds. These results clearly demonstrate the faster convergence and improved test accuracy achieved by CLSFSVRG, leading to significant savings in overall edge resources and communication costs.

The advantage of our proposed algorithm is clearly demonstrated by its performance in reducing cross-entropy loss during the training process. The results in Fig. 3 illustrate the cross-entropy loss achieved by CLSFSVRG and the baseline algorithms as the participation ratio of edge agents decreases from 20% to 5% out of 400 agents. By analyzing the cross-entropy loss, we can evaluate the convergence speed w.r.t the training procedure. Although LoSAC converges faster than the other baselines, both CLSFSVRG and CFSVRG outperform the baselines by achieving lower cross-entropy loss across all participation ratios. This is attributed to the conjugated central acceleration employed by CLSFSVRG and CFSVRG. As the participation ratio of edge agents decreases, the advantage

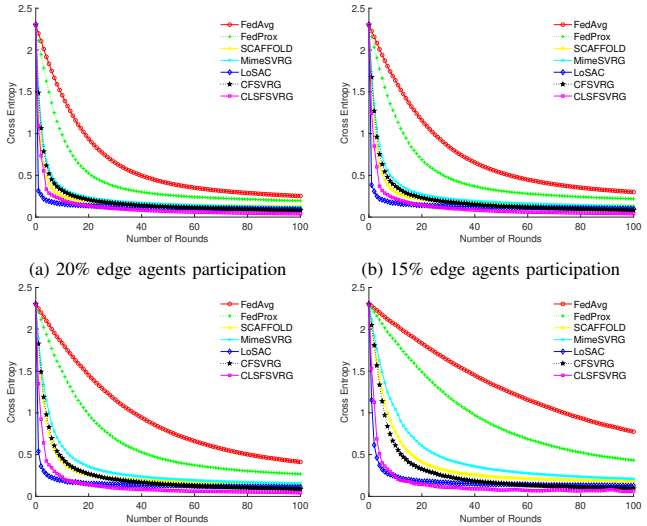
This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2024.3454273



(a) 20% edge agents participation (b) 15% edge agents participation

(c) 10% edge agents participation (d) 5% edge agents participation

Fig. 2. Test accuracy comparison with decreasing ratio of participated edge agents.



(a) 20% edge agents participation (b) 15% edge agents participation

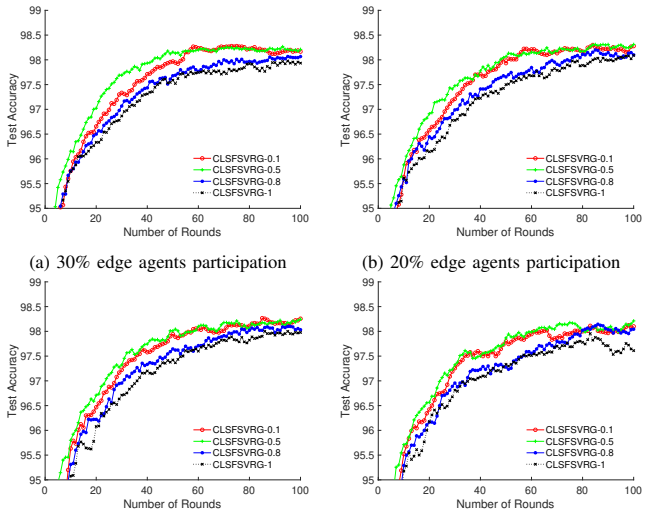
(c) 10% edge agents participation (d) 5% edge agents participation

Fig. 3. Training loss comparison with decreasing ratio of participated edge agents.

of conjugated central acceleration in reducing cross-entropy loss becomes more evident, as shown in Fig. 3. Moreover, CLSFSVRG's convergence speed approaches that of LoSAC when the number of participating edge agents is very limited, as depicted in Fig. 3(d). This robustness of CLSFSVRG is due to the inexact line search and global conjugated direction updating used in the central acceleration. Consequently, CLSFSVRG consistently reduces cross-entropy loss across varying participation ratios of edge agents.

B. Impact of the local training settings

Next, we focus on the performance of the proposed CLSFSVRG under different local training settings. As shown in Fig. 4, we compare the test performance of CLSFSVRG with local learning rates set to $\{0.1, 0.5, 0.8, 1.0\}$, named as CLSFSVRG-0.1, CLSFSVRG-0.5, CLSFSVRG-0.8, and



(a) 30% edge agents participation (b) 20% edge agents participation

(c) 15% edge agents participation (d) 10% edge agents participation

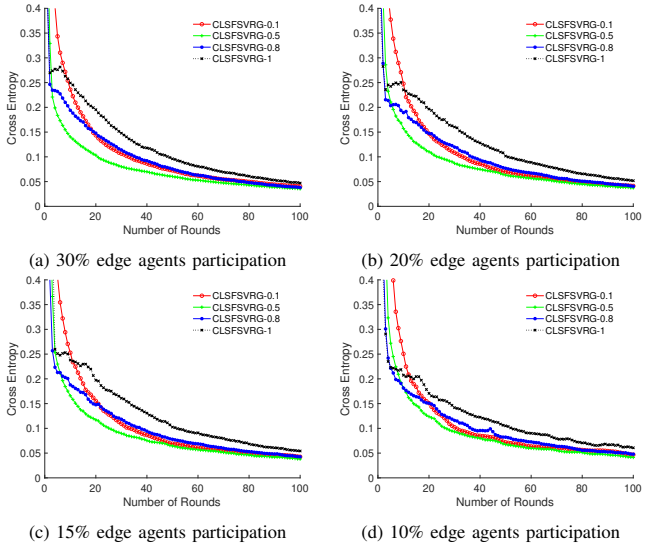
Fig. 4. Test accuracy comparison with different local learning rates.

CLSFSVRG-1, respectively, where participation ratios of edge agents set to $\{30\%, 20\%, 15\%, 10\%\}$.

The simulation results in Fig. 4(a) show that with 30% of edge agents participating in each federated training round, a local learning rate of 0.5 yields better performance faster than other settings. Specifically, local learning rate of 0.5 achieves 98% test accuracy after 40 rounds, while local learning rate of 0.1 reaches the same accuracy level around 60 rounds. In contrast, learning rates of 0.8 and 1.0 fail to achieve 98% test accuracy even after 100 rounds. As shown in Fig. 4(b), the performance with local learning rate of 0.5 slightly deteriorates when the participation of edge agents decreases to 20%. However, even at this participation level, it converges faster within the first 50 rounds. After 50 rounds, the performance with local learning rate of 0.5 becomes comparable to that with local learning rate of 0.1, with both achieving faster convergence and higher test accuracy compared to local learning rates of 0.8 and 1.0. When the edge agent participation ratio further decreases to 15%, as shown in Fig. 4(c), the performance across all local learning settings becomes more variable. Nonetheless, the best results are still achieved with learning rates of 0.5 and 0.1. As depicted in Fig. 4(d), both 0.1 and 0.5 local learning rates surpass the 0.8 setting within 80 federated training rounds. After 80 rounds, the performance with local learning rate of 0.8 aligns more closely with that of 0.1 and 0.5, indicating that a relatively larger local learning rate can still lead to promising results. However, using a rate of 1.0 significantly diverges from the others, illustrating that an excessively large local learning rate can negatively impact performance. While a larger local learning rate may yield faster initial progress, it also introduces greater oscillations during updates, adversely affecting the final converged results.

The comparison of training cross-entropy loss across different local training settings is shown in Fig. 5. Smaller learning rates reduce oscillation during updates but suffer from slower convergence. However, with the global inexact line search scheme, even smaller learning rates can achieve faster convergence and higher test accuracy. When edge agent

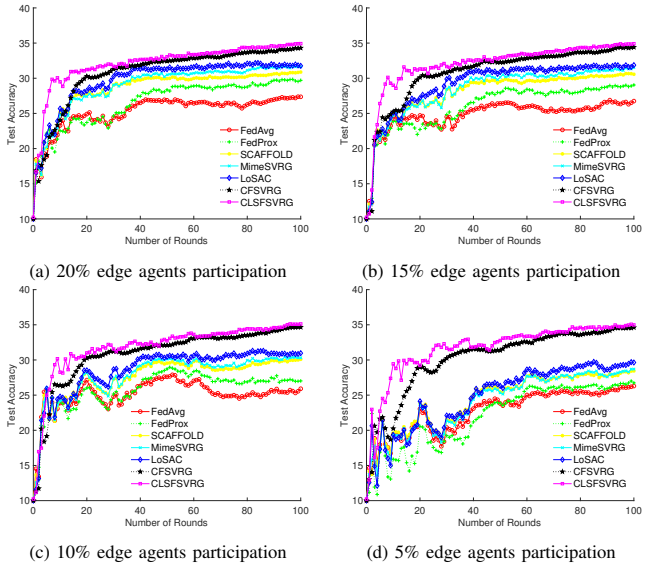
This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2024.3454273



(a) 30% edge agents participation (b) 20% edge agents participation

(c) 15% edge agents participation (d) 10% edge agents participation

Fig. 5. Training loss comparison with different local learning rates.



(a) 20% edge agents participation (b) 15% edge agents participation

(c) 10% edge agents participation (d) 5% edge agents participation

Fig. 6. Test accuracy comparison with decreasing ratio of participated edge agents from CIFAR-10.

participation ratio is reduced from 30% to 10%, as shown in Fig. 5, the initial performance gap between local learning rates of 1.0 and 0.8 compared to 0.1 becomes more pronounced. Initially, the higher local learning rates show faster progress, but they fail to converge to a low loss and lose their advantage after a few rounds due to oscillation caused by the large step sizes. In contrast, local learning rates of 0.1 and 0.5 exhibit smoother performance and their advantages become evident early in the training process.

We extend our investigation to the CIFAR-10 dataset, which consists of 50000 training images and 10000 test images, to further validate the effectiveness of our proposed method. The performance comparisons, illustrated in Fig. 6, highlight the impact of decreasing participation ratios of edge agents. Our proposed CLSFSVRG consistently outperforms the baselines in both convergence speed and test accuracy.

The robustness of CLSFSVRG is particularly evident when compared to the baselines, whose performance degrades significantly as the participation ratio decreases. Additionally, the benefits of the inexact line search in central acceleration are highlighted through comparisons between CLSFSVRG and CFSVRG. This analysis reveals not only faster convergence but also superior performance in converged test accuracy, especially in scenarios with limited edge agent participation during training. Even as the variance in performance across all methods increases at a participation ratio as low as 5%, CLSFSVRG distinguishes itself by achieving significantly greater stability than the others. This underscores the reliability of CLSFSVRG, particularly in federated learning scenarios with highly dynamic participation.

C. Computation and Communication Evaluation

We then focus on two key aspects of performance evaluation: communication and computation costs. In our evaluation, we focus on comparing our proposed method with LoSAC, as LoSAC has demonstrated clear advantages over the other baselines. Figs. 7 and 8 compare the communication and computation cost required to achieve test accuracy of 95.5%, 96%, and 96.5%, respectively. We set the scenario with only 5% of edge agents participating in federated training. Fig. 7 shows the quantitative assessment of the data volume that needs to be transmitted to achieve the specified test accuracy. The results in Figs. 7 and 8 demonstrate that our design substantially reduces the overall communication burden, as well as the overall local computing workload to achieve the same test accuracy at the expense of slightly increased computation cost at the central server.

Our approach reduces communication cost through two main factors. First, the proposed central acceleration approach minimizes the number of required rounds. Second, within each round, the ratio of participating edge agents can be reduced. To quantify the communication cost, we define the unit communication cost as the number of values included in the transmitted gradients and local model drifts. In our experiments, each gradient and local model drift consists of 2.25 KB of data. The total computational workload for a given FL task is evaluated by 5% edge agents participation to achieve the required test accuracy levels.

The results in Fig. 8 further illustrate the advantages of our proposed CLSFSVRG in terms of the overall computation time measured with the experimental platform features an 8-core CPU, a 14-core GPU, and 16GB of RAM. The results demonstrate by slightly increasing the computation time of the central server, we can substantially reduce the local agent computation time, and the overall computation cost can be reduced by up to 65%.

VI. Conclusions and Discussions

In this work, we proposed an accelerated FL method with global updating based on conjugate directions and inexact line search, named CLSFSVRG. Our approach not only demonstrates a faster convergence rate but also achieves higher test accuracy compared to the state-of-the-art. Simulation results

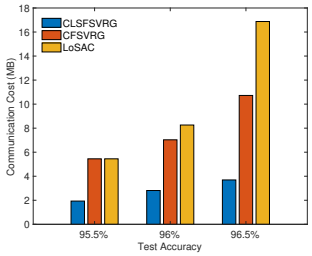
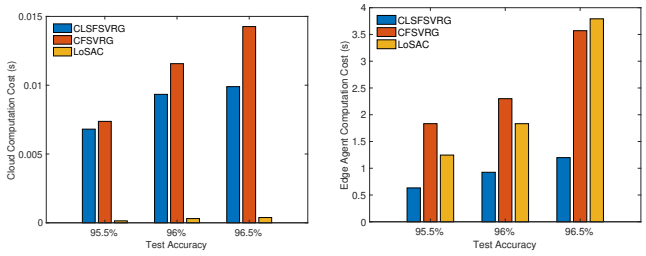


Fig. 7. System Communication Cost Comparison.



(a) Central Server Computation Time (b) Edge Agent Computation Time

Fig. 8. Overall Computation Time Comparison.

indicate that our global conjugate direction updating and inexact line search scheme significantly accelerate federated optimization convergence and are more robust with limited edge agent participation, offering a substantial advantage in dynamic FL environments. In our future work, we will evaluate the impact of different edge agent selection strategies on communication costs, model performance, and resource utilization in various network conditions. While our proposed CLSFSVRG method primarily focuses on synchronous updates, it is essential to discuss asynchronous strategies, which can mitigate the straggler effect and enable more efficient use of resources. However, asynchronous training introduces several challenges, such as how to ensure consistent global model states and stable convergence, handling client failures and ensuring fault tolerance, which beckon further investigation.

APPENDIX

A. Properties of Objective Functions

We define an auxiliary variable $\boldsymbol{z}_k^r$ as

$$\boldsymbol{z}_k^r = \hat{\boldsymbol{w}}_{i,k-1}^r + \tau(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r), \tag{31}$$

where $0 \leq \tau \leq 1$, and notice that

$$\int_0^1 df_i(\boldsymbol{z}_k^r) = \int_0^1 \nabla f_i(\boldsymbol{z}_k^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)d\tau. \tag{32}$$

Since

$$\begin{aligned} &f_i(\hat{\boldsymbol{w}}_{i,k}^r) - f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) \\ &= \int_0^1 f_i(\hat{\boldsymbol{w}}_{i,k-1}^r + \tau(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r))d\tau. \end{aligned} \tag{33}$$

combining (33) and (32), we obtain

$$\begin{aligned} f_i(\hat{\boldsymbol{w}}_{i,k}^r) &= f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) + \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r) \\ &+ \int_0^1 (\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)d\tau. \end{aligned} \tag{34}$$

Since

$$\begin{aligned} &\left|\int_0^1 (\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)d\tau\right| \\ &\leq \int_0^1 |(\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)|d\tau, \end{aligned} \tag{35}$$

and

$$\begin{aligned} &|(\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)| \\ &\leq ||(\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))||_2 \cdot ||(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)||_2, \end{aligned} \tag{36}$$

and by the Lipschitz continuous gradient assumption, we have

$$\begin{aligned} &||(\nabla f_i(\boldsymbol{z}_k^r) - \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r))||_2 \leq L_i||\boldsymbol{z}_k^r - \hat{\boldsymbol{w}}_{i,k-1}^r||_2 \\ &= L_i||\tau(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r)||_2. \end{aligned} \tag{37}$$

Combining (34)-(37), we can upper bound $f_i(\hat{\boldsymbol{w}}_{i,k}^r)$ by

$$\begin{aligned} f_i(\hat{\boldsymbol{w}}_{i,k}^r) \leq\ & f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) + \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r) \\ &+ \frac{L_i}{2}||\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r||^2. \end{aligned} \tag{38}$$

Similarly, we can lower bound $f_i(\hat{\boldsymbol{w}}_{i,k}^r)$ by

$$\begin{aligned} f_i(\hat{\boldsymbol{w}}_{i,k}^r) \geq\ & f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) + \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r) \\ &+ \frac{\mu_i}{2}||\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r||^2, \end{aligned} \tag{39}$$

where μ_i refers to the lower bound of the eigenvalues of the Hessian of the local objective f_i.

The properties of the federated optimization objective function depends on the properties of the local objective functions of each edge agent. We assume all local objective functions $\{f_i(\boldsymbol{w})\}_{i=1,\cdots,|E|}$ satisfy

$$f_i(\alpha\boldsymbol{w}^r + (1-\alpha)\boldsymbol{w}^{r-1}) \leq \alpha f_i(\boldsymbol{w}^r) + (1-\alpha)f_i(\boldsymbol{w}^{r-1}), \tag{40}$$

for $i = 1, \cdots, |E|$, where $0 < \alpha < 1$ and $\boldsymbol{w}^r$ denotes the global model after the r-th round. Then, we can obtain

$$\begin{aligned} &f(\boldsymbol{\alpha w^r + (1-\alpha) w^{r-1}}) = \sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\alpha\boldsymbol{w}^r + (1-\alpha)\boldsymbol{w}^{r-1}) \\ &\leq \alpha \sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\boldsymbol{w}^r) + (1-\alpha)\sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\boldsymbol{w}^{r-1}) \\ &= \alpha f(\boldsymbol{w}^r) + (1-\alpha)f(\boldsymbol{w}^{r-1}), \end{aligned} \tag{41}$$

which means the federated optimization objective function also satisfies the same property.

We remark that an upper bound of the eigenvalues of the Hessian $\boldsymbol{\nabla^2 f}$ is given by

$$L = \sum_{i=1}^{|E|} \frac{n_i}{n} L_i. \tag{42}$$

Based on Eq. (39), the lower bound of the eigenvalues of $\boldsymbol{\nabla^2 f}$ is obtained by

$$\mu = \sum_{i=1}^{|E|} \frac{n_i}{n} \mu_i. \tag{43}$$

Consequently, the global objective function is upper bounded by

$$f(\hat{\boldsymbol{w}}^r) \leq \sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) + \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r) + \frac{L}{2}||\hat{\boldsymbol{w}}^r - \hat{\boldsymbol{w}}^{r-1}||^2, \tag{44}$$

and is lower bounded by

$$f(\hat{\boldsymbol{w}}^r) \geq \sum_{i=1}^{|E|} \frac{n_i}{n} f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) + \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)^T(\hat{\boldsymbol{w}}_{i,k}^r - \hat{\boldsymbol{w}}_{i,k-1}^r) + \frac{\mu}{2}||\hat{\boldsymbol{w}}^r - \hat{\boldsymbol{w}}^{r-1}||^2. \tag{45}$$

B. Convergence of Local Updating

The target to analyze the local convergence is to prove that in the local training procedure of the i-th edge agent, we can upper bound the distance from the current local model $\hat{\boldsymbol{w}}_{i,k}^r$ and the optimal global model $\boldsymbol{w}^*$ as

$$E[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}|\hat{\boldsymbol{w}}_{\boldsymbol{i,0}}^{\boldsymbol{r}}] \leq \text{Var}||\hat{\boldsymbol{w}}_{\boldsymbol{i,0}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}, \tag{46}$$

which is sufficient to support the linear convergence rate in the local updating. The sequence $\{\hat{\boldsymbol{w}}_{i,k}^r\}_{k=0}^{\infty}$ can converge to $\boldsymbol{w}^*$ with linear rate. The convergence ratio ρ is defined as

$$\rho = \lim_{k\to\infty} \frac{||\hat{\boldsymbol{w}}_{\boldsymbol{i,k+1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||}{||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{p}}}, \tag{47}$$

which is one indicator to get the order of the convergence speed. The convergence speed is quantified by the largest nonnegative integer p, namely, the order of convergence, which is a measure of how good the worst part of the tail is. Larger values of the convergence order p imply faster convergence, since the distance from the optimal global $\boldsymbol{w}^*$ is reduced by the p-th power in a single step. From the definition of the convergence ratio, asymptotically we have

$$||\hat{\boldsymbol{w}}_{\boldsymbol{i,k+1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*|| = \rho||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{p}}. \tag{48}$$

The speed of convergence is increased if p is increased and ρ is reduced. If we can prove $p = 1$ and $\rho < 1$, it will have linear convergence.

To prove the above conclusion, we begin from the one step local iteration. After adding $\boldsymbol{w}^*$, the k-th local updating of edge agent i in r-th round can be formulated as the conditional expectation $E_{k-1}[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^2]$ which can be rewritten as

$$E_{k-1}[||\hat{\boldsymbol{w}}_{i,k-1}^r + \alpha_l \boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^2]. \tag{49}$$

Then, we decompose Eq. (49) into 3 components as

$$\begin{aligned} C_1 &= ||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}, \\ C_2 &= 2\alpha_l(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^T E_{k-1}[\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}], \\ C_3 &= \alpha_l^2 E_{k-1}[||\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}||^2], \end{aligned} \tag{50}$$

where $E_{k-1}[\cdot]$ refers to the conditional expectation over the $\{\hat{\boldsymbol{w}}_{\boldsymbol{i,j}}^{\boldsymbol{r}}\}_{j=0,\cdots,k-1}$ local step updating.

Based on the unbiased local gradient estimation and $\nabla f(\boldsymbol{w}^{r-1})$ is independent with $\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}$, we obtain that

$$\begin{aligned} &(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^T E_{k-1}[\boldsymbol{g}_i(\hat{\boldsymbol{w}}_{i,k-1}^r)] \\ &= (\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^T \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r), \end{aligned} \tag{51}$$

and

$$(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^T(\nabla f_i(\boldsymbol{w}^{\boldsymbol{r-1}}) - \nabla f_i(\boldsymbol{w}^{r-1})) = 0. \tag{52}$$

Thus, we can rewrite C_2 as

$$C_2 = -2\alpha_l(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^T \nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r). \tag{53}$$

By strong convexity as shown in Lemma 3, C_2 can be upper bounded by

$$C_2 \leq -2\alpha_l\mu_i||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}. \tag{54}$$

Then, we transform the variance of the local search direction $\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}$ as

$$\text{Var}_{\boldsymbol{k-1}}(\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) = \text{Var}_{\boldsymbol{k-1}}(\boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}}) - \boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}})), \tag{55}$$

which leads to

$$\text{Var}_{\boldsymbol{k-1}}(\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) \leq E_{k-1}[||\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) - \boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}})||^{\boldsymbol{2}}]. \tag{56}$$

We rewrite the conditional expectation of the local updating direction $E_{k-1}[||\boldsymbol{d}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}||^2]$ as

$$E_{k-1}[||\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) - \boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}})||^{\boldsymbol{2}}] + ||\nabla \boldsymbol{f_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}})||^{\boldsymbol{2}}, \tag{57}$$

which leads to upper bound C_3 as to upper bound

$$E_{k-1}[||\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) - \boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}})||^{\boldsymbol{2}}]. \tag{58}$$

To achieve this goal, first, according to Jensen's inequality, we have

$$\begin{aligned} &E_{k-1}[||\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) - \boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}})||^{\boldsymbol{2}}] \\ &\leq ||E_{k-1}[\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}})] - E_{k-1}[g_i(w^{r-1})]||^2. \end{aligned} \tag{59}$$

And according to (38), we have

$$||\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r) - \nabla f_i(\boldsymbol{w}^{r-1})||^2 \leq L_i^2||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}^{\boldsymbol{r-1}}||^2. \tag{60}$$

Since we can rewrite

$$||\hat{\boldsymbol{w}}_{i,k-1}^r - \boldsymbol{w}^{r-1}||^2 = ||(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*) - (\boldsymbol{w}^{\boldsymbol{r-1}} - \boldsymbol{w}_{\boldsymbol{i}}^*)||^{\boldsymbol{2}}. \tag{61}$$

According to the triangle inequality, we have

$$\begin{aligned} &\frac{\theta}{2}||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}} + \frac{1}{2\theta}||\boldsymbol{w}^{\boldsymbol{r-1}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}} \\ &\geq (\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*)^{\boldsymbol{T}}(\boldsymbol{w}^{\boldsymbol{r-1}} - \boldsymbol{w}_{\boldsymbol{i}}^*), \end{aligned} \tag{62}$$

where $\forall\ \theta > 0$. By setting $\theta = 1$, we obtain

$$\begin{aligned} &||\hat{\boldsymbol{w}}_{i,k-1}^r - \boldsymbol{w}^{r-1}||^2 \leq 2||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}} \\ &+ 2||\boldsymbol{w}^{\boldsymbol{r-1}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}. \end{aligned} \tag{63}$$

Then, we can achieve the conclusion that

$$\begin{aligned} &E_{k-1}[||\boldsymbol{g_i}(\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}}) - \boldsymbol{g_i}(\boldsymbol{w}^{\boldsymbol{r-1}})||^{\boldsymbol{2}}] \\ &\leq 2L_i^2||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}} + 2L_i^2||\boldsymbol{w}^{\boldsymbol{r-1}} - \boldsymbol{w}_{\boldsymbol{i}}^*||^{\boldsymbol{2}}. \end{aligned} \tag{64}$$

Furthermore, since $\nabla \boldsymbol{f_i}(\boldsymbol{w_i^*}) = \mathbf{0}$ and by (38) we get

$$||\nabla f_i(\hat{\boldsymbol{w}}_{i,k-1}^r)||^2 \leq L_i^2 ||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^{\mathbf{2}}. \tag{65}$$

Combining the conclusions together, finally we obtain that

$$\begin{aligned}
&E_{k-1}\left[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2\right] \\
&\leq E_{k-1}||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2 - 2\alpha_l\mu_i E_{k-1}||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2 \\
&\quad + \alpha_l^2 L_i^2 E_{k-1}||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2 \\
&\quad + \alpha_l^2(2L_i^2 E_{k-1}||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2 + 2L_i^2||\boldsymbol{w^{r-1}} - \boldsymbol{w_i^*}||^{\mathbf{2}}) \\
&= (1 - 2\alpha_l\mu_i + 3\alpha_l^2 L_i^2)E_{k-1}[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k-1}}^{\boldsymbol{r}} - \boldsymbol{w_i^*}||^2] \\
&\quad + 2\alpha_l^2 L_i^2||\boldsymbol{w^{r-1}} - \boldsymbol{w_i^*}||^{\mathbf{2}}.
\end{aligned} \tag{66}$$

C. Convergence of Conjugated Global Updating

There are three primary advantages of the conjugate federated learning. First, unless the optimal global model is attained in less than q federation rounds, the anchor gradient is always nonzero and linearly independent of all previous global updating directions. Second, the conjugate federated updating only slightly more complicated than the traditional global model aggregation strategy, because the conjugated global model updating directions are also based on the anchor gradients, the process makes good uniform progress toward the optimal global model at every federation round.

The intuitive design guidance is that the optimal global model $\boldsymbol{w}^* \in R^q$ can be represented by q independent directions, i.e.,

$$\boldsymbol{w}^* = \sum_{i=0}^{q-1} \eta_i^* \boldsymbol{d_i^*}. \tag{67}$$

We can regard this representation as the result of an iterative process of q steps where at the i-th step, $\eta_i^* \boldsymbol{d_i^*}$ is added. The conjugate federated learning allows for an arbitrary initial global model $\boldsymbol{w^0} \in \boldsymbol{R^q}$ where the sequence $\{\hat{\boldsymbol{w}}^{\boldsymbol{r}}\}$ generated in central server as

$$\boldsymbol{w^{r+1}} = \boldsymbol{w^r} + \eta_r \boldsymbol{d}_r. \tag{68}$$

The idea is to update the initialized global model $\boldsymbol{w^0}$ to the optimal global model $\boldsymbol{w}^*$ as

$$\boldsymbol{w^*} - \boldsymbol{w^0} = \sum_{i=0}^{q-1} \eta_i \boldsymbol{d_i}, \tag{69}$$

by designing the conjugate directions and the parameters, i.e., $\{\boldsymbol{d_i}, \eta_i\}_{i=0}^{q-1}$.

First, we can prove that the conjugated direction set $\{\boldsymbol{d}_i\}_{i=0}^{r-1}$ can span a r-dimensional space $\boldsymbol{B_r}$ since they are r independent vectors. To prove the linear independence of the conjugated global model updating directions, we premultiply $\boldsymbol{d_j^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})$ to $\sum_{i=0}^{r-1} \eta_i \boldsymbol{d_i}$ and apply the conjugate property, we can have

$$\sum_{i=0}^{r-1} \eta_j \boldsymbol{d_j^T}\nabla^{\mathbf{2}} f(\boldsymbol{w}) \boldsymbol{d_i} = \eta_j \boldsymbol{d_j^T}\nabla^{\mathbf{2}} f(\boldsymbol{w}) \boldsymbol{d_j}. \tag{70}$$

As long as the eigenvalues of $\nabla^{\mathbf{2}} f(\boldsymbol{w})$ are non-zero, we can guarantee $\boldsymbol{d_j^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d_j} \neq 0$. Then, there is only one way to achieve

$$\sum_{i=0}^{r-1} \eta_i \boldsymbol{d_i} = 0, \tag{71}$$

namely, $\eta_j = 0$ for $j = 0, \cdots, r-1$, which guarantees the conjugated directions in $\{\boldsymbol{d_i}\}_{\boldsymbol{i=0}}^{\boldsymbol{r-1}}$ are independent vectors.

To show the idea behind the conjugate federated updating step length design, we premultiply $\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})$ to $(\boldsymbol{w^* - w^0})$ and according to Eq. (69) to obtain

$$\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^* - w^0}) = \sum_{i=0}^{q-1} \eta_i \boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d_i}. \tag{72}$$

Then, in order to generate the conjugate property which leads to

$$\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^* - w^0}) = \eta_r \boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d}_r, \tag{73}$$

the step length parameter in conjugate federated updating should be designed as

$$\eta_r = \frac{\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^* - w^0})}{\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d}_r}. \tag{74}$$

In the conjugate federated learning, the trajectory by updating initial global model $\boldsymbol{w^0}$ to $\boldsymbol{w}^r$ can be represented by

$$\boldsymbol{w}^r = \boldsymbol{w^0} + \sum_{i=0}^{r-1} \eta_i \boldsymbol{d_i}. \tag{75}$$

Similarly as shown in Eq. (72) and Eq. (73), we premultiply $\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})$ to $(\boldsymbol{w^r - w^0})$ and according to Eq. (75) we have

$$\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^r - w^0}) = \sum_{i=0}^{r-1} \eta_i \boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d_i} = 0, \tag{76}$$

which leads to the conclusion that

$$\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{w^r} = \boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{w^0}. \tag{77}$$

Then, we apply the conclusion in Eq. (77) into Eq. (74), we obtain

$$\eta_r = \frac{\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^* - w^r})}{\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d}_r}. \tag{78}$$

Furthermore, since the Hessian is the changing rate of the gradient w.r.t. the parameter variable $\boldsymbol{w}$, the change caused in gradient can be approximated by

$$E[\boldsymbol{g}(\boldsymbol{w^r}) - \boldsymbol{g}(\boldsymbol{w^*})] \approx \nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^r - w^*}), \tag{79}$$

then, according to that $\nabla f(\boldsymbol{w^*}) = \mathbf{0}$, we obtain

$$\nabla^{\mathbf{2}} f(\boldsymbol{w})(\boldsymbol{w^r - w^*}) \approx E[\boldsymbol{g}(\boldsymbol{w^r})]. \tag{80}$$

Based on Eq. (78), we design the conjugate federated updating step length as

$$\eta_r = -\frac{\boldsymbol{d_r^T}\boldsymbol{g}(\boldsymbol{w^r})}{\boldsymbol{d_r^T}\nabla^{\mathbf{2}} f(\boldsymbol{w})\boldsymbol{d}_r}, \tag{81}$$

to guarantee the conjugate property of the generated directions.

The conjugate federated updating can substantially improve the FL training procedure, because the sequential conjugated directions are generated by solving the problem optimally dimension by dimension, namely, the conjugate federated

updating not only optimizes the updating in each direction, i.e., $\hat{\boldsymbol{w}}^{\boldsymbol{r}}$ minimizes $f(\boldsymbol{w})$ on the line $\boldsymbol{w} = \boldsymbol{w_{r-1}} + \eta \boldsymbol{d}_{r-1}$, where η is the variable, but also minimizes $f(\boldsymbol{w})$ on the subspace $\boldsymbol{B_r}$ spanned by $\{\boldsymbol{d_i}\}_{\boldsymbol{i=0}}^{\boldsymbol{r-1}}$, i.e., on the linear variety $\boldsymbol{w^0} + \boldsymbol{B_r}$. This conclusion will hold by showing that $\boldsymbol{g(w^r)} \perp \boldsymbol{B_r}$, i.e., the gradient information in the current spanned subspace $\boldsymbol{B_r}$ is 0 leading to optimal solution projected on $\boldsymbol{B_r}$.

To provide proof of $\boldsymbol{g(w^r)} \perp \boldsymbol{B_r}$, first, we set $\boldsymbol{B_0}$ as null space, thus, for $r = 0$, $\boldsymbol{g(w^0)} \perp \boldsymbol{B_0}$. Then, we assume $\boldsymbol{g(w^r)} \perp \boldsymbol{B_r}$. Based on

$$\boldsymbol{g(w^{r+1})} - \boldsymbol{g(w^r)} = \boldsymbol{\nabla^2} f(\boldsymbol{w}^r)(\boldsymbol{w^{r+1}} - \boldsymbol{w^r}), \tag{82}$$

we have

$$\boldsymbol{g(w^{r+1})} = \boldsymbol{g(w^r)} + \eta_r \boldsymbol{\nabla^2} f(\boldsymbol{w^r}) \boldsymbol{d_r}, \tag{83}$$

which leads to

$$\boldsymbol{g(w^{r+1})^T d_i} = \boldsymbol{g(w^r)^T d_i} + \eta_r \boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w^r}) \boldsymbol{d_i}, \tag{84}$$

by inner product with $\boldsymbol{d_i}$. When $i < r$, $\boldsymbol{g(w^r)^T d_i}$ vanishes because of the induction hypothesis, and $\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w^r}) \boldsymbol{d_i} = 0$ by the $\boldsymbol{\nabla^2} f(\boldsymbol{w}^r)$-orthogonality, then we can get the following conclusion

$$\boldsymbol{g(w^{r+1})^T d_i} = \boldsymbol{g(w^r)^T d_i} + \eta_r \boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0. \tag{85}$$

Furthermore, based on the conjugate federated updating step length designed in Eq. (81), we can obtain

$$\boldsymbol{g(w^{r+1})^T d_r} = \boldsymbol{g(w^r)^T d_r} + \eta_r \boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w^r}) \boldsymbol{d_r} = 0, \tag{86}$$

when $i = r$. Then, combining the conclusions in Eq. (85) and Eq. (86) we obtain $\boldsymbol{g(w^{r+1})} \perp \boldsymbol{B_{r+1}}$.

Then, we introduce the idea behind the design of β_r which controls the momentum of the previous global updating directions. To generate direction set $\{\boldsymbol{d_i}\}_{i=0}^{q-1}$ to be a conjugate set w.r.t. $\boldsymbol{\nabla^2} f(\boldsymbol{w})$, which satisfies

$$\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0, \quad \text{for } 0 \le i < r \text{ and } 1 \le r \le q. \tag{87}$$

First, we assume that

$$\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0, \quad \text{for } 0 \le i < r. \tag{88}$$

Since $\boldsymbol{\nabla^2 f(w) d_i}$ is a vector in the subspace spanned by $\{\boldsymbol{d_i}\}_{\boldsymbol{i=0}}^{\boldsymbol{r}}$, which means when $i < r$, it can be represented by

$$\boldsymbol{\nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = \sum_{i=0}^{r} a_i \boldsymbol{d_i}. \tag{89}$$

By post-multiplying $\boldsymbol{\nabla^2} f(w) \boldsymbol{d_i}$ to $\boldsymbol{d_{r+1}^T}$ and according to Eq. (25) and Eq. (89), we can obtain

$$\boldsymbol{d_{r+1}^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = -\sum_{i=0}^{r} a_i \boldsymbol{g(w^{r+1})^T d_i} + \beta_r \boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i}. \tag{90}$$

Since $\boldsymbol{g(w^{r+1})}$ orthogonal to the subspace spanned by $\{\boldsymbol{d_i}\}_{\boldsymbol{i=0}}^{\boldsymbol{r}}$, and we have assumed that $\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0$. Thus, we can get

$$\boldsymbol{d_{r+1}^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0, \tag{91}$$

for $0 \le i < r$. Then, we design

$$\beta_r = \frac{\boldsymbol{g(w^{r+1})^T \nabla^2 f(w) d_r}}{\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r}}, \tag{92}$$

to guarantee

$$\begin{aligned} &\boldsymbol{d_{r+1}^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r} = -\boldsymbol{g(w^{r+1})^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r} \\ &+ \beta_r \boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r} = 0. \end{aligned} \tag{93}$$

Therefore, we have proved that

$$\boldsymbol{d_{r+1}^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_i} = 0, \quad \text{for } 0 \le i < r+1. \tag{94}$$

Combining the results from Eq. (88) and Eq. (94), we can guarantee the generated direction set is a conjugate set.

Furthermore, we can redesign the formulation of β_r to get rid of $\boldsymbol{\nabla^2} f(\boldsymbol{w})$. By pre-multiplying $\boldsymbol{-g(w^r)^T}$ to $\boldsymbol{d}_r$, we have

$$\boldsymbol{-g(w^r)^T d_r} = \boldsymbol{g(w^r)^T g(w^r)} - \beta_{r-1} \boldsymbol{g(w^r)^T d_{r-1}}. \tag{95}$$

Since $\boldsymbol{g(w^r)} \perp \boldsymbol{B_r}$, thus we can get $\boldsymbol{g(w^r)^T d_{r-1}} = 0$ where $\boldsymbol{d}_{r-1}$ is inside the subspace $\boldsymbol{B_r}$. Therefore, we obtain that

$$\boldsymbol{-g(w^r)^T d_r} = \boldsymbol{g(w^r)^T g(w^r)}, \tag{96}$$

which allows us to reformulate η_r as

$$\eta_r = -\frac{\boldsymbol{g(w^r)^T d_r}}{\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r}} = \frac{\boldsymbol{g(w^r)^T g(w^r)}}{\boldsymbol{d_r^T \nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r}}. \tag{97}$$

On the other hand, we premultiply $\boldsymbol{g(w^{r+1})}$ to

$$\boldsymbol{\nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r} = \frac{1}{\eta_r} (\boldsymbol{g(w^{r+1})} - \boldsymbol{g(w^r)}), \tag{98}$$

and we obtain

$$\begin{aligned} &\boldsymbol{g(w^{r+1})^T \nabla^2 f(w) d_r} \\ &= \frac{1}{\eta_r} (\boldsymbol{g(w^{r+1})^T g(w^{r+1})} - \boldsymbol{g(w^{r+1})^T g(w^r)}). \end{aligned} \tag{99}$$

We can get the conclusion that the subspace spanned by the gradients is the same subspace as spanned by the search directions according to

$$\boldsymbol{g(w^{r+1})} = \boldsymbol{g(w^r)} + \eta_r \boldsymbol{\nabla^2} f(\boldsymbol{w}) \boldsymbol{d_r}, \tag{100}$$

which means $\boldsymbol{g(w^r)}$ is also inside the subspace $\boldsymbol{B_{r+1}}$. Then, we can get the conclusion that

$$\boldsymbol{g(w^{r+1})^T g(w^r)} = 0. \tag{101}$$

Thus, β_r can be reformulated as

$$\beta_r = \frac{\boldsymbol{g(w^{r+1})^T \nabla^2 f(w) d_r}}{\boldsymbol{d_r^T \nabla^2 f(w) d_r}} = \frac{\boldsymbol{g(w^{r+1})^T g(w^{r+1})}}{\boldsymbol{g(w^r)^T g(w^r)}}. \tag{102}$$

In the overall updating, and combining with (66) we obtain that

$$\begin{aligned} &\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}^*||^2] \\ &\le (1 - 2\alpha_l \mu_i + 5\alpha_l^2 L_i^2) \boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r-1}} - \boldsymbol{w}^*||^2]. \end{aligned} \tag{103}$$

To contract the upper bound of the distance from the current local model $\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}}$ to the optimal global solution $\boldsymbol{w}^*$, we define a factor of e, and as long as we can guarantee that

$$e \boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}_{\boldsymbol{i,k}}^{\boldsymbol{r}} - \boldsymbol{w}^*||^2] \ge \boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}_{\boldsymbol{i,0}}^{\boldsymbol{r}} - \boldsymbol{w}^*||^2], \tag{104}$$

we have

$$\boldsymbol{E_{r-1}}||\boldsymbol{w^{r-1}}-\boldsymbol{w^*}||^2 \leq e\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k-1}}-\boldsymbol{w^*}||^2], \tag{105}$$

since $\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,0}} = \hat{\boldsymbol{w}}^{r-1}$. Combining (103)-(105), we have

$$\begin{aligned}&\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k}}-\boldsymbol{w^*}||^2]\\ &\leq (1-2\alpha_l\mu_i+5\alpha_l^2L_i^2e)\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k-1}}-\boldsymbol{w^*}||^2].\end{aligned} \tag{106}$$

If we set $\alpha_l\mu_i = 5\alpha_l^2L_i^2e$ where the local learning rate $\alpha_l = \frac{\mu_i}{5L_i^2e}$ we can get

$$\begin{aligned}&\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k}}-\boldsymbol{w^*}||^2]\\ &\leq (1-\frac{\mu_i^2}{5L_i^2e})\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k-1}}-\boldsymbol{w^*}||^2].\end{aligned} \tag{107}$$

By applying the updating recursively, we obtain

$$\begin{aligned}&\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k}}-\boldsymbol{w^*}||^2]\\ &\leq (1-\frac{\mu_i^2}{5L_i^2e})^k\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r-1}}-\boldsymbol{w^*}||^2],\end{aligned} \tag{108}$$

which can be rewritten as

$$\begin{aligned}&\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}_{\boldsymbol{i,k}}-\boldsymbol{w^*}||^2]\\ &\leq \exp(-\frac{\mu_i^2k}{5L_i^2e})^k\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r-1}}-\boldsymbol{w^*}||^2].\end{aligned} \tag{109}$$

We define $\hat{L} = \max\{L_i\}_{i\in E}$ and $\hat{\mu} = \min\{\mu_i\}_{i\in E}$, and it leads to the conclusion that

$$\boldsymbol{E_{r-1}}[||\hat{\boldsymbol{w}}^{\boldsymbol{r}}-\boldsymbol{w^*}||^2] \leq e^{-\frac{5r\hat{L}^2e}{\hat{\mu}^2}}||\hat{\boldsymbol{w}}^{\boldsymbol{0}}-\boldsymbol{w^*}||^2, \tag{110}$$

with recursively updating, where it should satisfy

$$r \geq \frac{\hat{\mu}^2}{5\hat{L}^2e}\log(\frac{||\hat{\boldsymbol{w}}^{\boldsymbol{0}}-\boldsymbol{w^*}||^2}{\epsilon}), \tag{111}$$

to guarantee error tolerance ϵ.

REFERENCES

[1] Q. Pu, G. Ananthanarayanan, P. Bodik, S. Kandula, A. Akella, P. Bahl, and I. Stoica, "Low latency geo-distributed data analytics," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 4, pp. 421–434, 2015.

[2] F. Qian, Y. Jin, S. J. Qin, and K. Sundmacher, "Guest editorial special issue on deep integration of artificial intelligence and data science for process manufacturing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3294–3295, 2021.

[3] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: Intelligentizing mobile edge computing, caching and communication by federated learning," *Ieee Network*, vol. 33, no. 5, pp. 156–165, 2019.

[4] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.

[5] E. Puik, D. Telgen, L. van Moergestel, and D. Ceglarek, "Assessment of reconfiguration schemes for reconfigurable manufacturing systems based on resources and lead time," *Robotics and Computer-Integrated Manufacturing*, vol. 43, pp. 30–38, 2017.

[6] L. Ren, Z. Meng, X. Wang, R. Lu, and L. T. Yang, "A wide-deep-sequence model-based quality prediction method in industrial process analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3721–3731, 2020.

[7] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[8] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[9] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 4615–4625.

[10] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[11] A. Z. Tan, H. Yu, L. Cui, and Q. Yang, "Towards personalized federated learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[12] H. Chen, H. Wang, Q. Long, D. Jin, and Y. Li, "Advancements in federated learning: Models, methods, and privacy," *ACM Computing Surveys*, 2023.

[13] Y. Qu, C. Dong, J. Zheng, H. Dai, F. Wu, S. Guo, and A. Anpalagan, "Empowering edge intelligence by air-ground integrated federated learning," *IEEE Network*, vol. 35, no. 5, pp. 34–41, 2021.

[14] L. Li, D. Shi, R. Hou, H. Li, M. Pan, and Z. Han, "To talk or to work: Flexible communication compression for energy efficient federated learning over heterogeneous mobile edge devices," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[15] W. Y. B. Lim, J. S. Ng, Z. Xiong, J. Jin, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 536–550, 2021.

[16] B. Luo, X. Li, S. Wang, J. Huang, and L. Tassiulas, "Cost-effective federated learning design," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[17] Y. Deng, F. Lyu, J. Ren, Y.-C. Chen, P. Yang, Y. Zhou, and Y. Zhang, "Fair: Quality-aware federated learning with precise user incentive and model aggregation," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[18] D. Basu, D. Data, C. Karakus, and S. Diggavi, "Qsparse-local-sgd: Distributed sgd with quantization, sparsification and local computations," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[20] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5132–5143.

[21] S. P. Karimireddy, M. Jaggi, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, "Mime: Mimicking centralized stochastic algorithms in federated learning," *arXiv preprint arXiv:2008.03606*, 2020.

[22] H. Chen, H. Wang, Q. Yao, Y. Li, D. Jin, and Q. Yang, "Losac: An efficient local stochastic average control method for federated optimization," *ACM Transactions on Knowledge Discovery from Data*, vol. 17, no. 4, pp. 1–28, 2023.

[23] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečnỳ, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," *arXiv preprint arXiv:2003.00295*, 2020.

[24] H. Zhang, K. Zeng, and S. Lin, "Fedur: Federated learning optimization through adaptive centralized learning optimizers," *IEEE Transactions on Signal Processing*, 2023.

[25] W. Wu, L. He, W. Lin, R. Mao, C. Maple, and S. Jarvis, "Safa: A semi-asynchronous protocol for fast federated learning with low overhead," *IEEE Transactions on Computers*, vol. 70, no. 5, pp. 655–668, 2020.

[26] Q. Wu, X. Chen, T. Ouyang, Z. Zhou, X. Zhang, S. Yang, and J. Zhang, "Hiflash: Communication-efficient hierarchical federated learning with adaptive staleness control and heterogeneity-aware client-edge association," *IEEE Transactions on Parallel and Distributed Systems*, vol. 34, no. 5, pp. 1560–1579, 2023.

[27] 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 Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2020.

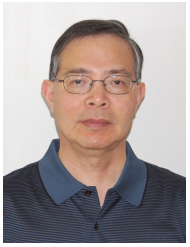
[28] J. Liu, H. Xu, L. Wang, Y. Xu, C. Qian, J. Huang, and H. Huang, "Adaptive asynchronous federated learning in resource-constrained edge computing," *IEEE Transactions on Mobile Computing*, vol. 22, no. 2, pp. 674–690, 2021.

[29] J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data *et al.*, "A field guide to federated optimization," *arXiv preprint arXiv:2107.06917*, 2021.

[30] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.

Lei Zhao (S'17) received the B.S. and M.A.Sc. degrees in computer science and technology from Xidian University, Xi'an, China, in 2015 and 2018, respectively, and earned his Ph.D. in Electrical and Computer Engineering from the University of Victoria in 2023. He is currently a Post-Doctoral Fellow in the E&CE Department at the University of Victoria. His research focuses on federated learning and optimization with applications in finance.

Lin Cai (S'00-M'06-SM'10-F'20) is a Professor with the E&CE Department at the University of Victoria. She is an NSERC Steacie Memorial Fellow, a CAE fellow, a EIC Fellow, an IEEE Fellow, an RSC College member, and a 2020 "Star in Computer Networking and Communications" by N2Women. Her research interests span several areas in communications and networking, with a focus on network protocol and architecture design supporting ubiquitous intelligence.



Wu-Sheng Lu (F'99-LF'12) received the B.Sc. degree in Mathematics from Fudan University, Shanghai, China, in 1964, the M.S. degree in electrical engineering, and the Ph.D. degree in control science from the University of Minnesota, Minneapolis, USA, in 1983 and 1984, respectively. Since 1987, he has been with the University of Victoria, Victoria, B.C., Canada, and is now Professor Emeritus. He is the co-author with A. Antoniou of Two-Dimensional Digital Filters (Marcel Dekker, 1992) and Practical Optimization: Algorithms and Engineering Applications (2nd ed., Springer, 2021), and with E. K. P. Chong and S. H. Zak of An Introduction to Optimization (5th ed., Wiley, 2023).