Convergence Analysis and System Design for Federated Learning over Wireless Networks

Shuo Wan\textsuperscript{1,2}, Jiaxun Lu\textsuperscript{2}, Pingyi Fan*\textsuperscript{1}, \textit{Senior member, IEEE}, Yunfeng Shao\textsuperscript{2}, Chenghui Peng\textsuperscript{3}, and Khaled B. letaief\textsuperscript{4}, \textit{Fellow, IEEE}

\textsuperscript{1}Department of Electronic Engineering, Tsinghua University, Beijing, P.R. China
\textsuperscript{2}Huawei Noah’s Ark Lab
\textsuperscript{3}Huawei Wireless Technology Lab
\textsuperscript{4}Department of Electronic Engineering, Hong Kong University of Science and Technology, Hong Kong

E-mail: wan-s17@mails.tsinghua.edu.cn, *fpy@mail.tsinghua.edu.cn, {lujiaxun, shaoyunfeng, pengchenghui}@huawei.com, eckhaled@ece.ust.hk

\date{30 Apr 2021}

\begin{abstract}
Federated learning (FL) has recently emerged as an important and promising learning scheme in IoT, enabling devices to jointly learn a model without sharing their raw data sets. However, as the training data in FL is not collected and stored centrally, FL training requires frequent model exchange, which is largely affected by the wireless communication network. Therein, limited bandwidth and random package loss restrict interactions in training. Meanwhile, the insufficient message synchronization among distributed clients could also affect FL convergence. In this paper, we analyze the convergence rate of FL training considering the joint impact of communication network and training settings. Further by considering the training costs in terms of time and power, the optimal scheduling problems for communication networks are formulated. The developed theoretical results can be used to assist the system parameter selections and explain the principle of how the wireless communication system could influence the distributed training process and network scheduling.
\end{abstract}

\begin{IEEEkeywords}
Federated learning, Network scheduling, Coupling design, Convergence analysis, Edge computing
\end{IEEEkeywords}

\section{I. INTRODUCTION}

With the emergence of data sets and the rapid growth of distributed computing, distributed learning has become a promising mode for deployment of large-scale machine learning\cite{1}. Under such circumstances, due to data privacy and limited communication resources, it is very difficult to transmit all the raw data sets to a central server for learning, especially for applications with widely distributed intelligent clients. Thus, to implement machine learning with high efficiency, researchers started to focus on distributed learning schemes. In\cite{2}, the authors proposed the federated learning (FL) scheme for distributed data sets. Such a technique allows applications to collectively reap the benefits of shared models trained from the rich data while avoiding the need of central data collection.

However, implementation of FL could face constraints from insufficient communication network support, especially in a highly distributed system. Unlike conventional training with all resources on the cloud, FL training does not have direct access to all raw distributed data. Thus, it requires periodic exchange of model data over a wireless network among clients. Without sufficient message exchange, FL training could face severe degradation compared with centralized training. Therefore, in FL scheme, network scheduling would largely affect the convergence of training process.

To support FL with high efficiency, the network design must be linked with FL training to consider their coupling properties. Conventional network design mainly considers the communication efficiency without considering the computation algorithms. However, since convergence of FL can be largely affected by message exchange, how to support such a training algorithm with limited network resources would rise up as an important problem. Under such circumstances, the performance gain of FL from network scheduling would build a bridge between communication and computation. By proper analysis of such a bridge, a coupled network design can be achieved.

The problem of wireless network scheduling for FL training takes two steps to be solved. Firstly, a theoretical analysis of its convergence is required. Secondly, based on the convergence analysis, a network model should be set up for FL implementation and the settings therein will be optimized accordingly. Although convergence analysis for distributed learning has been widely studied, the analysis considering the impact of network settings has not been investigated with general and precise results. To the best of our knowledge, former works mainly consider settings of computation, such as the accuracy threshold, batch size, etc. However, as discussed above, communication will affect the model aggregation, which leads to a degraded convergence. Therefore, to support FL in network settings, the convergence analysis considering network support is an essential bridge. Then by jointly considering its tradeoff with costs in the communication network by model analysis, the optimal settings will be derived. Meanwhile, FL implementation mainly depends on the cooperation among widely distributed clients. Degradation in some participating clients also leads to a worse performance of the whole training process. Under such circumstances, the resource scheduling among clients is also required.

\subsection{A. Related works}

Recently some works studied the convergence of FL training with several bounds\cite{3,7}. Authors of\cite{4} analyzed the general
converging speed of FL. Considering the impact of training parameters, [3] proposed an analytic bound. However, the background communication was not considered systematically and the bound still requires refinement with more explicit physical meanings. The work in [5] tried to improve the classical FL scheme based on theoretical analysis. Meanwhile, [6] analyzed FL convergence considering the tradeoff between the local epoch and global epoch. Based on this work, [7] further proposed a three-layer training scheme and discussed related control algorithms. However, in the proposed bounds, the tradeoff between training and communication is based on an additionally defined resource budget, which restricts the application range.

From the point of FL implementation, some works discussed the FL scheme and the scheduling algorithm therein [6–13]. Authors of [13] proposed a multi-player game to study participants’ reactions under various incentive mechanisms in FL scenarios. [8] studied the effects of malicious clients on FL. Considering the time slot division in the TDMA protocol, a control algorithm for FL over a wireless network was proposed in [6]. Authors of [9] considered TDMA settings to jointly optimize the computation settings and time slot division in communication. However, the convergence analysis did not discuss the effects of communication network. As a result, the communication design becomes independent from training, without considering the tradeoff therein. Considering the package loss in communication, [10] analyzed the FL convergence rate and proposed a control algorithm. However, the convergence results are based on specific policies and can not provide more insights in general design. Besides, the local epoch, non-i.i.d. data set and partial participation were not considered, which restricts its generality. [11, 12] also considered the optimization of communication settings. Although their communication models are solid, the convergence analysis still needs to be improved.

B. Contributions

Previous works have implemented a set of basic analysis models and scheduling algorithms for FL implementation. Therein, the convergence analysis of FL training still needs more observations of the background wireless network. Besides, part of the inequalities should be handled more tightly to reflect the trend of some important parameters. From the network design for FL perspective, one needs to consider the coupling properties between communication and computation. To resolve such problems, our contributions are summarized as follows.

1) The convergence rate analysis of FL training will be cast into a joint optimization problem of computing (training) and communication, in a more general setting with non-i.i.d. data sets, local training epochs, partial client participation, limited bandwidth, and package loss. By taking into account training settings as well as the impacts of the communication network on model aggregation, the derived convergence bound can be clearly divided into a computation part and a communication part with explicit physical meaning. Meanwhile, the tightness of the convergence bound is also improved. Thus, the impact of the intrinsic factors can be reflected with clearer physical meanings, fitting better in experiments.

2) Considering time and power consumption as the joint training cost, the general system cost for each training epoch is defined. By taking the convergence analysis as a bridge between FL training and wireless network, the overall setting for bandwidth and local training epoch are optimized with closed-form theoretical expressions. The result could fit the common knowledge of network and distributed learning, providing more insights of network settings for efficient FL. To the best of our knowledge, this is the first explicit result considering the tradeoff between computing (training) and communication in FL with closed-form principles for selecting hyperparameters in a wireless network.

3) Considering the limitation of FL due to high level of distribution among clients, we propose network scheduling algorithms to enhance the cooperation among distributed clients. By adaptive resource scheduling, clients with varied capability and burden could achieve a better performance in synchronization. Given a specific network setting, the design could minimize the time and power cost accordingly. On the other hand, its derived costs will also affect the choice of the network hyper-parameters. Then by jointly considering the coupling factors, we set up an integrated design principle for FL implementation over wireless networks.

II. SYSTEM MODEL AND PROBLEM FORMULATION

The major focus is on a two-layer FL system composed of a central server and a set of $N$ distributed intelligent clients ($\mathbb{N} = \{1, 2, ..., N\}$). As shown in Fig. 1, the server and clients are connected through a wireless network, where a cellular-based network is used as an example. Such a system could support various IoT applications such as environmental monitoring, health-care, etc.

A. Federated learning process

In the FL system, the data set of client $j$ is $S_j = \{(x_1^{(j)}, y_1^{(j)}), (x_2^{(j)}, y_2^{(j)}), ..., (x_{D_j}^{(j)}, y_{D_j}^{(j)})\}$ with $D_j$ data sam-
The transmission rate with full bandwidth for client $j$ is given by $B_r^j = \log_2 (1 + \frac{p_j h_j}{a_j B N_0})$, where $a_j$ is the proportion of bandwidth allocated to client $j$ with $\sum_{j \in P_t} a_j \leq 1$ and $p_j = p_j^0 a_j B$. Therein, $p_j^0$ shall denote the transmission power for unit bandwidth and $h_j$ is the channel power gain of client $j$. The noise density in a wireless network is denoted as $N_0$. Then by defining $z_m$ as the required data size of transmitted model, the time cost of model uploading for client $j$ should be

$$I_{u,j} = \frac{z_m}{r_j} = \frac{z_m}{a_j \log_2 (1 + \frac{p_j h_j}{a_j B N_0})}.$$

Therefore, the power consumption should be

$$P_{u,j} = p_j I_{u,j} = p_j z_m r_j.$$  

Note that the clients in set $P_t$ are selected uniformly in each global epoch, which would change with global epochs. Therefore, considering the expectation of the cost on $P_t$, the uploading cost is defined as

$$C_u = E_{P_t} \left[ \max_{j \in P_t} \left( \frac{z_m}{a_j B r_j^0} \right) + l_0 \sum_{j \in P_t} p_j^0 \frac{z_m}{r_j} \right],$$

where $r_j^0 = \log_2 (1 + \frac{p_j h_j}{N_0})$ is the transmission rate with unit bandwidth and $l_0$ is the weight of the power cost. Due to the synchronization requirement in model aggregation, the time cost is affected by the slowest client.

Wireless connection typically endures a random package loss due to the independent fading of wireless channels. Let us assume $K$ clients are uploading their model data simultaneously, and that $K_\gamma$ of them will be successfully received. For simplicity and considering the worst case, $K_\gamma \geq K(1 - \gamma)$.

**Table I: The key notations**

| Notation | Definition |
|----------|------------|
| $N$ | The total number of clients participating in FL training. |
| $P_t$ | The current client set participating in model uploading. |
| $K$ | Size of the randomly selected set $P_t$. |
| $D$ | The average size of distributed data sets among clients. |
| $E_l$ | Length of the local training epoch. |
| $G_e$ | Total global epochs to reach loss $\epsilon$. |
| $\gamma$ | The package loss rate in wireless network. |
| $B$ | Total bandwidth for model uploading. |
| $\{a_j\}$ | Ratio of bandwidth allocation among clients in $P_t$. |
| $\{\bar{w}_j\}$ | The processor frequency for local training. |
| $\{p_j^0\}$ | The model uploading power with $p_j = p_j^0 a_j B$. |
| $\psi_j$ | The transmission rate with full bandwidth for client $j$.$ |
| $C_u$ | The expected model uploading cost in one global epoch. |
| $C_{u,0}$ | $E[\frac{1}{C_u}]$: The expected uploading cost for unit $K$. |
| $C_n$ | The expected computation cost in one global epoch. |
| $l_0$ | Weight of power cost in $C_u$ and $C_n$. |
| $\lambda$ | Metric of non-i.i.d. extent in distributed data sets. |
| $T_d$ | The time cost of model download. |
where $\gamma < 1$. Note that the lost packages also take resources for transmission. Thus, the package loss does not influence $C_n$. Meanwhile, since the model aggregation here only involves $K \gamma$ local models, FL training may take more global epochs to converge.

2) Computation model: It is straightforward to see that computation latency in distributed training is proportional to the local epoch length $E_i$. The computation latency of client $j$ is

$$t_{n,j} = E_i \frac{z_{n,j}}{f_j},$$

where $f_j$ is the processor frequency and $z_{n,j}$ is the required processing cycles for one round of local training. The corresponding power consumption is

$$p_{n,j} = E_i k_j f_j^2 z_{n,j},$$

where $k_j$ is a parameter depending on the specific processor on client $j$.

In fact, more data in training usually requires more processing cycles. This means that $z_{n,j}$ is proportional to $D_j$. Therefore, it is reasonable to assume that $z_{n,j} = \alpha_0 D_j$ where $\alpha_0$ is an empirical parameter depending on training model and softwares. Meanwhile, since distributed training is an entirely local operation, the local computation cost should be the major focus in network scheduling. Thus, the average power consumption is taken instead of the summation of all training powers. Therefore, the computation cost of one global epoch is given by

$$C_n = E_p [\max_{j \in P} \{ E_i \frac{\alpha_0 D_j}{f_j} \} + l_0 (1 - \frac{1}{K} \sum_{j \in P} E_i k_j f_j^2 \alpha_0 D_j)] .$$

C. Problem formulation

We define $T_d$ as the extra time cost in the model broadcast by base station, which is rather fixed and determined by central resources. The package loss rate $\gamma$ is given as a background network parameter. Then from the definitions of $C_u$ and $C_n$ in (8) and (11), the joint optimization problem for FL implementation over a wireless network can be expressed as follows.

$$\min_{K, E_i, \{a_j\}, \{f_j\}} G_e [C_u + C_n + T_d],$$

s.t. $G_e = G_e (E_i, K, \gamma),$

$$E_i \geq 1, 0 \leq \gamma < 1,$$

$$1 \leq K \leq N, |P_t| = K,$$

$$\sum_{j \in P_t} a_j \leq 1, a_j > 0, j \in P_t,$$

$$f_j^{min} \leq f_j \leq f_j^{max}, j \in P_t.$$

In the above formulation, the parameter $G_e$ is the number of global epochs taken by FL to reach a loss $\epsilon$, which will be given by convergence analysis. Constraint (12) shows that $G_e$ is jointly affected by $K$, $\gamma$ and $E_i$. There exists an important tradeoff between $G_e$ and $C_u$. If $K$ increases with more clients uploading, $G_e$ will get smaller. However, from constraint (12c), more clients in $P_t$ leads to a smaller bandwidth for each client, which will in turn lead to a larger $C_u$. Therein, as the metric of FL training convergence, $G_e (E_i, K, \gamma)$ is actually an important bridge between AI training and the communication network. In the subsequent section, we will derive the specific closed-form expression of $G_e$. Constraints (12a), (12b), (12d) and (12e) are basic ranges in system settings. $E_i$ is the local training epoch, which stands for input from computation in training. $K$ is the size of $P_t$, representing the capability of model uploading provided by the wireless network. $\{a_j\}$ and $\{f_j\}$ are scheduling policies among clients, aimed at minimizing $(C_u + C_n)$. Note that $K$ and $E_i$ are the hyper-parameters determining the total bandwidth division and local epoch. Given such settings, $\{a_j\}$ and $\{f_j\}$ are scheduled accordingly. By minimizing the cost for an arbitrary $P_t$, the expectation of the cost on random client selection can also be minimized. Meanwhile, on the contrary, the expected cost will in turn affect selection of $K$ and $E_i$, which can be reflected by theoretical results. By solving (12), the integrated principles for the selection of the hyper-parameters and resource scheduling will be obtained.

III. FL CONVERGENCE ANALYSIS

In the following, the analysis is based on the stochastic gradient descent (SGD) update. The training process is given in Section II-A and the general non-i.i.d. data distribution is considered. In fact, the ideal i.i.d. data distribution is a special case therein. Beforehand, some common assumptions and the metric of non-i.i.d. in FL are first introduced. Then the expression of $G_e$ will be given and discussed in detail.

A. Preparations

1) Assumptions on loss function: The common assumptions of the L-smooth and $\mu$-Polyak-Lojasiewicz (PL) condition for the loss function are given as follows.

Assumption 1: (L-smooth [3, 4]) The loss function $f(.)$ in FL training satisfies

$$f(y) \leq f(x) + (y - x)^T \nabla f(x) + L/2 \| y - x \|^2 .$$

Assumption 2: ($\mu$-P-L condition [4, 14]) The loss function $f(.)$ in FL training satisfies a general extension of the $\mu$-strongly convex property, which is defined as

$$\| \nabla f(x) \|^2 \geq 2\mu [f(x) - f^*].$$

Remark 1: The P-L condition in Assumption 2 may not fit globally in classical neural networks like CNN. However, viewing $f^*$ as the local optimum, the assumption could still work. Therefore, it is reasonable to apply this in the convergence analysis. In our latter experiments with CNN, the rational of the assumption could be confirmed.

2) Non-i.i.d. data: In practice, independent observed data at distributed clients typically diverge in probability distribution. For instance, some clients may have more data of cats while others may observe dogs. To proceed with the analysis, a commonly adopted measure is introduced to quantify such a property.
Definition 1: \[ 4 \quad 15 \] Given \( N \) clients with weights \( \{ \pi_j \}(\sum_{j=1}^N \pi_j = 1) \) and \( \{ \nabla f_j(w) \} \) as their gradients, a measurement \( \lambda \) for non-i.i.d. in data set is defined as

\[
\frac{\sum_{j=1}^N \pi_j \left\| \nabla f_j(w) \right\|^2}{\left\| \sum_{j=1}^N \pi_j \nabla f_j(w) \right\|^2} = \lambda \leq 1. \tag{15}
\]

Remark 2: It is known that \( \lambda \geq 1 \) by Jensen’s inequality [\[16\]]. Therein, \( \lambda = 1 \) represents the i.i.d. condition. Parameter \( \lambda \) reflects the non-i.i.d. extent of the stochastic gradients of \( N \) clients.

B. Results of \( G_e \)

Theorem 1: Under Assumption [\[1\], \[2\]] and Definition [\[4\]] if we choose the learning rate \( \eta_1 \) as \( O(1) \), \( |P_t| = K \) and local epoch as \( E_1 \), \( G_e \) is given by

\[
G_e = \frac{4L^2\gamma^2\lambda}{\epsilon^2} \left[ \frac{\lambda - 1}{K(1-\gamma)2D} - \frac{1}{2C\phi_0} + \frac{f_0}{E_1} \right], \tag{16}
\]

where \( \gamma \) is the package loss rate in communication. Parameter \( D = \frac{1}{N} \sum_{j=1}^N D_j \) is the average size of the training data and \( G^2 \) is defined as the gradient upper-bound. \( C_1 \) and \( \sigma^2 \) are constants related to the gradient variance in SGD and \( \phi_0 - 1 \approx \frac{\sigma^2}{\lambda} \). Specific definitions of parameter \( C_1 \), \( \sigma^2 \) and \( \phi_0 \) can be found in Appendix [\[A\]] with the proof shown in Appendix [\[F\]].

In [\[16\]], \( K(1-\gamma)2D \) and \( \frac{1}{2C\phi_0} + \frac{f_0}{E_1} \) are two key terms with clear physical meanings, separately representing the impacts resulting from communication and computation. By observing \( K(1-\gamma)2D \), we could see that \( K \) and \( 1-\gamma \) compensate with each other to speed up the convergence in the form of product. The communication term is also proportional to \( (\lambda - 1) \). Thus, the extent of non-i.i.d. will influence the effects of the communication network. This can be observed from the experiments in Section [\[V\]].

The effects of computation are reflected by \( \frac{(\lambda - 1)E_1}{2C\phi_0} + \frac{f_0}{E_1} \) in [\[16\]]. The optimal \( E_1 \) can be derived as \( E_1^* = \frac{C_1\phi_0 f_0}{2C(\lambda - 1)} \). As \( E_1 \) increases, the local training becomes more sufficient. But increasing \( E_1 \) may also cause convergence failure due to the large diversity among the distributed data sets. Thus, a proper selection of \( E_1 \) must be balanced to speed up the training. For the i.i.d. setting with \( \lambda = 1 \), \( E_1 \) can be arbitrarily large, which would be limited by the capability of local processors.

It is noted that a larger \( K \) naturally leads to a smaller allocated bandwidth. Thus \( C_u \) in [\[5\]] will definitely get larger. Combined with Theorem [\[1\]], the tradeoff between \( G_e \) and \( C_u \) can be easily observed.

IV. DESIGN PRINCIPLE FOR FL

In this section, the joint design principle for parameter selection and resource scheduling will considered.

A. Sub-problems for hyper-parameters and scheduling policy

As known, the client set \( P_t \) for model uploading is renewed by uniform selection in each global epoch. Due to different conditions of the clients, the specific system cost in each global epoch may be varied. Thus, in the definition of \( C_u \) and \( C_n \) in [\[8\]] and [\[11\]], the expectation on \( P_t \) is taken to get the expected cost for each global epoch.

As discussed in Section [\[1\]-C], \( K \) and \( E_1 \) are network hyper-parameters while \( \{ a_j \} \) and \( \{ f_j \} \) are scheduling policies based on a given setting. Thus, it is natural to see that finding solution of the hyper-parameters and scheduling policy should be decoupled. Before the decoupling, the expected costs \( C_u \) and \( C_n \) should first be transformed as a function of the hyper-parameters.

From the definition in [\[11\]], \( C_u \) is actually proportional to \( E_1 \). Therein, setting \( C_n = E_1 C_{n,0} \), the unit cost \( C_{n,0} \) can be directly defined as

\[
C_{n,0} = E_{P_t}[\max_{j \in P_t} \{ \frac{\alpha_0 D_j}{f_j} \} + \frac{l_0}{K} \sum_{j \in P_t} \kappa_j f_j^2\alpha_0 D_j]. \tag{17}
\]

Considering \( C_u \), as \( K \) increases, more clients will be uploading their models simultaneously with less bandwidth for each client. Since the clients are chosen uniformly in \( P_t \), then \( E[\frac{1}{a_j}] \approx K \) holds for an arbitrary policy to allocate the bandwidth among \( K \) clients in \( P_t \). That is, the expected allocated bandwidth for each client is inversely proportional to \( K \). Then by observing the definition of \( C_u \) in [\[8\]], the uploading cost should be proportional to \( K \) in the form of \( C_u = K C_{u,0} \).

Considering the full participation case with \( K = N \), the uploading cost for an arbitrary scheduling policy can be given as \( (\max_{j=1}^N \{ \frac{z_m}{a_j Br_j} \} + l_0 \sum_{j=1}^N \frac{p_j}{r_j^2} \) statistically equals to \( K \sum_{j=1}^N \{ \frac{z_m}{a_j Br_j} \} + l_0 \sum_{j=1}^N \frac{p_j}{r_j^2} \) for an arbitrary scheduling policy of \( \{ a_j \} \). Thus, by dividing the full participation cost with \( N \), \( C_{u,0} \) could be derived as

\[
C_{u,0} = \frac{1}{N} \sum_{j=1}^N \{ \frac{z_m}{a_j Br_j} \} + l_0 \sum_{j=1}^N \frac{p_j}{r_j^2}. \tag{18}
\]

By considering the expected cost in the form (\( K C_{u,0} + E_1 C_{n,0} + T_d \)) for a given scheduling policy, the sub-problem for the optimal hyper-parameters is defined as follows.

\[
\min_{K,E_1} G_e [K C_{u,0} + E_1 C_{n,0} + T_d], \tag{19}
\]

s.t. \[ G_e = G_e (E_1, K, \gamma) \]. \tag{19a}
\[ E_1 \geq 1, 0 \leq \gamma < 1, \tag{19b} \]
\[ 1 \leq K \leq N. \tag{19c} \]

For an arbitrary \( P_t \), \( K \) and \( E_1 \), the sub-problem for the optimal scheduling policy is defined as

\[
\min_{\{ a_j \}, \{ f_j \}} C_{u,t} + C_{n,t}, \tag{20}
\]

s.t. \[ C_{u,t} = \max_{j \in P_t} \{ \frac{z_m}{a_j Br_j} \} + l_0 \sum_{j \in P_t} \frac{p_j}{r_j^2}, \tag{20a} \]
\[ C_{n,t} = \max_{j \in P_t} \{ E_1 \frac{\alpha_0 D_j}{f_j} \} + \frac{l_0}{K} \sum_{j \in P_t} E_1 k_j f_j^2 \alpha_0 D_j. \tag{20b} \]
The coupling properties between sub-problems (19) and (20) can be explained as follows. Given $K$, $P_t$ and $E_l$, the optimal $(a_j)$ and $(f_j)$ can be derived by solving sub-problem (20). That is, by taking the hyper-parameters as the input, it would give the optimal resource scheduling and the corresponding costs. In this sense, sub-problem (20) actually represents the optimal policy to minimize $C_{u,0}$ and $C_{n,0}$. The sub-problem (19) takes $C_{u,0}$ and $C_{n,0}$ as the input to get the optimal hyper-parameters. By representing the costs $C_u$ and $C_n$ as $KC_{u,0}$ and $E_lC_{n,0}$, the hyper-parameters would be optimized by considering tradeoff between $G_e$ and $C_u + C_n$. Therein, $G_e$ is given theoretically by Theorem 1.

The joint optimal design for FL implementation is given by combining the two sub-problems. Sub-problem (20) gives the scheduling policy for arbitrarily given parameters, while the expected costs given by such policy are taken as input to optimize the hyper-parameters in sub-problem (19). Given $G_e$ as the bridge between training and communication, the hyper-parameters would consider the coupling effects from AI training and the wireless network with an optimal balance. By properly setting the hyper-parameters and scheduling policy therein, FL training with minimum cost in time and power can be achieved.

### B. Optimal hyper-parameters

By solving sub-problem (19) combined with Theorem 1 the principles for the optimal hyper-parameters can be derived.

#### 1) Bandwidth setting:

**Theorem 2:** In FL training, the system cost related to $E_l$ and $K$ is given by $(KC_{u,0} + E_lC_{n,0} + T_d)$. Under Theorem 1, the optimal $K$ in (12) for FL implementation is

$$K^* = \rho_0 \sqrt{\frac{E_l}{D(1 - \gamma)}} \frac{C_{n,0} + T_d/E_l}{C_{u,0}},$$

(21)

where $\rho_0 = \sqrt{\frac{2(1 - \gamma)C_{n,0}}{D \lambda}}$ is a multiplier related to the background training process. The proof is in Appendix E.

Viewing $\frac{E_l}{T_d}$ as the unit waiting time to begin local training, $T_d$ can be considered as part of the computation cost. Thus, $K^*$ is related to the ratio between $C_{n,0}$ and $C_{u,0}$. As $C_{u,0}$ gets larger, a higher communication cost would decrease $K^*$. Meanwhile, a larger $C_{n,0}$ is matched with larger $K^*$, so that more sufficient training will not be wasted by insufficient communication. In this case, the design achieves a balance by comparing costs in communication and computation.

The tendency of $K^*$ in Theorem 2 could fit common insights of FL training over wireless networks. By observing that $\sqrt{\frac{E_l}{D(1 - \gamma)}} K^*$ increases with $E_l$ and decreases with $1 - \gamma$. In non-i.i.d. data sets, increasing $E_l$ results in a higher level of model divergence among distributed clients. It is then natural to expect a larger $K$. By considering the term $\frac{1}{1 - \gamma}$, a larger $\gamma$ means a lower rate of successful transmission in a wireless channel. Thus, $K$ should be increased for compensation.

The properties of the data sets and training algorithm also affect the network setting, reflected by $\rho_0$. By referring to Definition [1], $\lambda$ is the metric of non-i.i.d. case. As $\lambda$ increases, the local gradients will become more diverged, which leads to a larger $K$ for compensation. Meanwhile, $C_1$, $\phi_0$ and $f_0$ are related to the gradient variance and initial training loss, which also affect the need for model aggregation. The specific definitions of these parameters can be found in Appendix A and Appendix E.

As a coupled parameter in training and communication, $K^*$ is jointly determined by the training algorithm and wireless network. The closed-formed expression in Theorem 2 reflects the specific influence of the cost ratio, $E_l$, $\gamma$, $D$ and $\lambda$. In system design, these factors should be jointly considered to adjust settings in wireless networks.

#### 2) $E_l$ setting:

The setting of $E_l$ is considered to minimize $G_e$, with $E_l^* = \sqrt{\frac{C_{n,0}K}{2(1 - \gamma)}}$ as the optimum value. In FL training, the local epoch $E_l$ is originally introduced to lower down the frequency of communication. This is due to the fact that computation resources are typically more sufficient. Meanwhile, local training can be organized locally without complexity in interaction. At local processors, some more rounds of training may not increase much cost compared with the whole process of model uploading and downloading. Thus, in FL training, it is reasonable to set $E_l^*$ to minimize $G_e$.

Though $\phi_0$, $C_1$ and $f_0$ makes it hard to directly compute $E_l^*$, experiments that shall be provided later will show that $E_l^*$ does exist. Thus, assisted by the theory and estimation in experiments, $E_l^*$ can be obtained as one empirical parameter.

By jointly considering the results of $K^*$ and $E_l^*$, the background loss rate $\gamma$, the integrated principles for choosing hyper-parameters can be obtained. Such design could ensure a faster training convergence with less power cost, which is meaningful for FL implementation.

### C. Scheduling policy

The principles for optimizing the hyper-parameters have been discussed theoretically in Section IV-B. Given $K$, $E_l$ and a randomly selected client set $P_t$, the solution of sub-problem (20) for optimal $(a_j)$ and $(f_j)$ will be discussed in this subsection.

From (20), (20a) and (20b), the optimization objective of sub-problem (20) can be written as $[\max_{j \in P_t} (\frac{1}{a_j B r_j} E_l) + \max_{j \in P_t} (\frac{E_l \alpha_0 \gamma D_j}{f_j}) + \sum_{j \in P_t} (\frac{C_{n,0}}{a_j B r_j} + \frac{E_l \alpha_0 \gamma D_j}{f_j})]$. Note that that in practice, once the local training is completed, a client may immediately upload the model without waiting for others. Thus, the computation time and uploading time can be combined together with the transformed objective as $[\max_{j \in P_t} (\frac{1}{a_j B r_j} + \frac{E_l \alpha_0 \gamma D_j}{f_j}) + \sum_{j \in P_t} (\frac{C_{n,0}}{a_j B r_j} + \frac{E_l \alpha_0 \gamma D_j}{f_j})]$. Then by defining parameter $H$ with $\frac{C_{n,0}}{a_j B r_j} + \frac{E_l \alpha_0 \gamma D_j}{f_j} \leq H$ for arbitrary $j$ in set $P_t$, the sub-problem (20) can be transformed as follows.
\[
\min_{\{a_j\}, \{f_j\}} \sum_{j \in P_j} \left( \frac{z_m p_j^0}{r_j} + \frac{1}{K} E_i (a_0 \kappa_j D_j f_j^2) + H \right), \quad (22)
\]
\[
s.t. \quad H \geq \frac{z_m}{a_j B_{r_j}} + \frac{E_i a_j D_j}{f_j}, \quad (22a)
\]
\[
\sum_{j \in P_j} a_j \leq 1, a_j > 0, j \in P_j, \quad (22b)
\]
\[
f_j^{min} \leq f_j \leq f_j^{max}, j \in P_t. \quad (22c)
\]

It is straightforward to observe that (22) is a convex optimization problem. The corresponding Lagrange function can be defined as

\[
L = l_0 \sum_{j \in P_j} \left( \frac{z_m p_j^0}{r_j} + \frac{1}{K} E_i (a_0 \kappa_j D_j f_j^2) + H \right) + R \left( \sum_{j \in P_j} a_j - 1 \right) + \sum_{j \in P_j} \beta_j \left( \frac{z_m}{a_j B_{r_j}} + \frac{E_i a_j D_j}{f_j} - H \right), \quad (23)
\]

where \( R \) and \( \{\beta_j\} \) are dual variables. By KKT conditions on (23), the closed-form expression for \( a_j \) can be obtained by the following theorem.

**Theorem 3:** Given the transmission power \( p_j^0 \), channel gain \( h_j \) and background noise density \( N_0 \), the bandwidth allocation is given by

\[
a_j = \frac{\sqrt{\kappa_j f_j}}{\sum_{j \in P_j} \sqrt{\kappa_j f_j}}. \quad (24)
\]

The proof is provided in Appendix G.

Based on Theorem 3, the scheduling policy in (20) is proposed as follows.

1) **Optimal centralized solution:** From Theorem 3, \( a_j \) in (20) can be substituted by \( f_j \) and 20 would be a convex problem for \( f_j \). Then the optimal \( \{f_j\} \) can be derived at the central server by applying convex optimization solvers and \( \{a_j\} \) can be solved further.

Such a process could give an optimal solution by central scheduling. Therein, \( \{a_j\} \) is the central bandwidth allocation, which is fitted for central scheduling. However, in the FL training mode, as the local processor frequency, it would be better for \( f_j \) to be determined by clients locally. That is, client \( j \) should directly schedule its own \( f_j \), without counting on control from a central server. Besides, since set \( P_t \) is renewed in each global epoch, frequently using optimization solvers to solve such a problem may cause much higher costs. Therefore, we propose a substitute method to solve for near-optimal \( \{f_j\} \) in a distributed manner, which is rather simple and direct for FL engineering scenarios.

2) **Distributed scheduling policy:** Before optimization of \( f_j \), the average processor frequency \( \bar{f} = \frac{1}{N} \sum_{j=1}^N f_j \) is first considered as the basis. From definition of \( C_n \) in (11), suppose that all clients take unified parameters \( f_j = \bar{f} \), \( D_j = D = \frac{1}{N} \sum_{j=1}^N D_j \) and \( \kappa_j = \kappa = \frac{1}{N} \sum_{j=1}^N \kappa_j \). \( \bar{f} \) is chosen to minimize the averaged computation cost \( (E_i a_0 D + l_0 E_i \kappa^2 a_0 D) \). By simple calculation of its stationary point, the optimal \( \bar{f} \) is

\[
\bar{f} = \frac{1}{2l_0 \kappa}. \quad (25)
\]

Then due to the local data size \( D_j \), \( f_j \) can be given by

\[
f_j = \frac{D_j}{D} \bar{f} = \frac{D_j}{D} \frac{1}{2l_0 \kappa}. \quad (26)
\]

Considering constraint \( f_j^{min} \leq f_j \leq f_j^{max} \), the final \( f_j \) can be derived by taking the intersection between (26) and \([f_j^{min}, f_j^{max}]\). In this process, the employed \( \bar{f} \) in (25) considers the tradeoff between time and power in computation. With \( \bar{f} \) as the basic reference, clients will schedule \( f_j \) locally to achieve a balance of the distributed training time. Note that \( \bar{f} \) is a rather fixed parameter, which can be optimized beforehand and reported to clients for reference. Thus, \( f_j \) can be directly determined by clients in each round without counting on central control. Then, together with Theorem 3, the optimal \( a_j \) can be obtained.

**D. Integrated scheduling process**

Based on the principles for selecting the hyper-parameters in Section IV-B and the optimal scheduling policy in Section IV-C2, the proposed integrated scheduling process is shown in Fig. 2.

**Fig. 2:** The scheduling process for FL in wireless networks.

Before the learning process, the hyper-parameters \( E_i \) and \( K \) are first selected. \( E_i \) is adjusted due to the extent of non-i.i.d. in data sets by referring to Theorem 1 and discussions in Section IV-B2. From the scheduling policy in Section IV-C2 the unit computation cost \( C_{n,0} \) in (17) for Theorem 3 can be given as \( C_{n,0} = \frac{mD}{l_0} + l_0 \bar{f}^2 a_0 D \). By combining Section IV-C2 and Theorem 3, the bandwidth allocation policy can be taken in (19) to obtain \( C_{n,0} \). Thus, by combining the cost from scheduling policy and the principles from the theoretical results, \( K \) can be adjusted due to the cost ratio, package loss rate \( \gamma \), average data size \( D \) and \( E_i \) by referring to Theorem 2.

FL begins with initialization of the training model, data sets and channel state of distributed clients. Note that some clients may endure a very bad channel or process very little data. To achieve an efficient scheduling, such clients will be first removed from the candidate client set. That is, the client set \( P_t \) will exclude those in extremely bad conditions. Meanwhile, \( \bar{f} \) in (25) will be initialized and broadcast for local scheduling of \( f_j \). Then the random set \( P_t \) will be selected before the training process begins. As the local training is conducted at distributed
clients, the server will collect distributed states and allocate the bandwidth at the same time. Once the local training ends, the models will be uploaded and aggregated. Then $P_t$ will be renewed and the next global begins. Note that $\{a_t\}$ and $\{f_j\}$ are renewed for set $P_t$ in each global epoch. As shown in Fig. 2, the scheduling policy can be performed in parallel with the training process without much complexity.

V. EXPERIMENTS

A. Basic test settings

The database for training are selected as the classical MNIST and CIFAR10. The FL setting is the same as [22] and the models are based on CNN. To simulate the distributed data sets in FL, all samples in the training data set are distributed among $N$ clients. As both MNIST and CIFAR10 are for multi-classification, the non-i.i.d. data sets are realized by distributing the training data unevenly among the class labels. Meanwhile, the parameters for the communication channel are set as $B = 20MHz$ and $N_0 = 5 \times 10^{-20}$. The random channel power gain is generated following an exponential distribution, where $h_j = g_0(\frac{d}{\lambda})^\theta exp(1)$ with $g_0 = 10^{-3}$, $\theta = 4$, $d_0 = 1$, and $d = 200$.

B. FL convergence

Fig. 3(a), Fig. 3(b) and Fig. 3(c) show the trend of $G_\epsilon$ with respect to the parameters $K$ and $E_t$ in a non-i.i.d. setting. The target loss $\epsilon$ in the non-i.i.d. data sets is set as 1.88 for CIFAR and 0.81 for MNIST. For the i.i.d. case, it is set as 1.68 for CIFAR and 0.2 for MNIST. Total number of clients are set as 50 and 100. The size of the data sets in each client with $N = 50$ is twice as much as that with $N = 100$. Each point in the figure is obtained by running 20 times with Monte Carlo simulation. The stripe around the curves represents the confidence interval and the maximum global training epoch is set as 150 in each experiment.

As shown in Fig. 3(a), $G_\epsilon$ is approximately inversely proportional to $K$, consistent with the results in Theorem 1. For $N = 50$, $G_\epsilon$ gets smaller compared with $G_\epsilon$ for $N = 100$. By observing the result in Theorem 1, $N = 50$ corresponds to a larger $D$, which is consistent with such a trend.

In the i.i.d. setting, Fig. 3(b) shows that $K$ has little effect on the convergence speed. By observing Theorem 1 as $\lambda = 1$, there is $\frac{1}{\sqrt{(1-\gamma)\frac{D}{2\pi}}}$, which vanishes as $K$ increases, which is consistent with the results listed in Fig. 3(b). Thus, in the i.i.d. setting, we could set a smaller $K$ to save bandwidth.

Fig. 3(c) shows that $G_\epsilon$ has a minimum value with respect to the local epoch $E_t$. Such a tendency is consistent with the result in Theorem 1. As MNIST is a rather simple data set for CNN, it is not sensitive to $E_t$. For the results in Fig. 3(a) and Fig. 3(b) $E_t$ is set as 20 to get a relatively lower frequency of the model uploading.

C. Network scheduling

Based on the training loss for various $K$, we change the unit cost $C_{u,0}$ and $C_{u,0}$ to see the tendency of $K^\star$ with the cost ratio $E_tC_{u,0}$ ($E_t = 20$). In Fig. 5(a) and Fig. 5(b) $K^\star$ increases with the cost ratio. Meanwhile, both figures show that the curves of $K^\star$ with $N = 50$ are just below the curves with $N = 100$. This is because each client in $N = 50$ get more training data compared with that in $N = 100$. Then by observing Theorem 2, $K^\star$ will be reduced. Therefore, for the FL implementation
over wireless networks, the system need to jointly consider the cost ratio and size of local data sets in order to have an efficient usage of limited bandwidth.

By observing the curves with respect to $K$ and $E_l$ in Fig. 6(a) and Fig. 6(b), the former settings $C_u = K_{C_{u,0}}$ and $C_n = E_l C_{n,0}$ can be directly validated. The evenly allocation policy in Fig. 6(c) refers to $a_j = \frac{1}{K}$ and $f_j = \bar{f}$. Compared with such a policy, the proposed network scheduling algorithm achieves a lower cost with less $C_{u,0}$ and $C_{n,0}$. Besides, by comparing the curves with $Var = 0.1$ and $Var = 0.5$, it is straightforward to see that the variation among clients can be handled well by the proposed policy without significant degradation in costs.

VI. CONCLUSION

In this paper, we investigated the implementation of FL training over wireless networks. Considering a general FL training process on non-i.i.d. data sets, we jointly considered the coupling problem between communication and computing (training) in FL and presented some theoretical results in principle on the FL design. The inequalities were handled more elegantly to derive bounds with clearer physical meaning. Viewing latency and power consumption as a joint system cost, we discovered the tradeoff between FL convergence and system cost. The integrated design principles for FL implementation were also proposed in closed-form. Simulations on MNIST and CIFAR10 have demonstrated that the proposed convergence analysis and parameter selection principles have a good fit with the real recorded training loss. In particular, the work clearly pointed out how the wireless network and training settings would jointly influence FL convergence, which is meaningful for FL implementation in a wide range.

REFERENCES

[1] L. Bottou, F. E. Curtis, and J. Nocedal, “Optimization methods for large-scale machine learning,” Siam Review, vol. 60, no. 2, pp. 223–311, 2018.
[2] 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, pp. 1273–1282, 2017.
[3] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, “On the convergence of fedavg on non-iid data,” in International Conference on Learning Representations, 2019.
[4] F. Haddadpour and M. Mahdavi, “On the convergence of local descent methods in federated learning,” arXiv preprint arXiv:1910.14425, 2019.
[5] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, “On the convergence of federated optimization in heterogeneous networks,” arXiv preprint arXiv:1812.06127, vol. 3, 2018.
[6] 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 Journal on Selected Areas in Communications, vol. 37, no. 6, pp. 1205–1221, 2019.
[7] L. Liu, J. Zhang, S. Song, and K. B. Letaief, “Client-edge-cloud hierarchical federated learning,” in ICC 2020-2020 IEEE International Conference on Communications (ICC), pp. 1–6, 2020.
[8] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, “Analyzing federated learning through an adversarial lens,” in International Conference on Machine Learning, pp. 634–643, 2019.
[9] C. T. Dinh, N. H. Tran, M. N. Nguyen, C. S. Hong, W. Bao, A. Y. Zomaya, and V. Gramoli, “Federated learning over wireless networks: convergence analysis and resource allocation,” IEEE/ACM Transactions on Networking, pp. 1–12, 2020.
[10] 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, pp. 269–283, 2020.
[11] S. Luo, X. Chen, Q. Wu, Z. Zhou, and S. Yu, “Hfel: Joint edge association and resource allocation for cost-efficient hierarchical federated edge learning,” arXiv preprint arXiv:2002.11343, 2020.
Global aggregated model weight:
\[ \overline{w}_t = \frac{1}{K_t} \sum_{j \in P_t} w^j_t. \]  

Global aggregated SGD gradient:
\[ \overline{g}_t = \frac{1}{K_t} \sum_{j \in P_t} \overline{g}^j_t. \]  

Expectation of global aggregated gradient:
\[ \bar{g}_t = \frac{1}{K_t} \sum_{j \in P_t} g^j_t. \]  

Update of aggregated global model:
\[ \bar{w}_{t+1} = \bar{w}_t - \eta \bar{g}_t. \]  

Note that \( t \) in (28) (29) (30) can be an arbitrary time slot, not necessarily to be \( t = t_c = \frac{1}{\tau_e} t_l \). The aggregation terms are derived from virtually aggregated values without affecting the local training, while the real aggregation with feedback only occurs at \( t_c \).

A. Shorthand notations

For simplicity, some shorthand notations are defined as follows.
\[ w^j_t: \text{The local model weight in client } j \text{ at time } t. \]
\[ g^j_t: \text{The local stochastic gradient of client } j \text{ at time } t \text{ due to random data } \xi. \]
\[ D: \text{Average local data size among clients.} \]
\[ g^j_t: \text{The expected gradient of client } j \text{ at time } t. \text{ Note that } E[\xi](g^j_t) = g^j_t. \]
\[ P_{t, \gamma}: \text{Set of clients successfully uploading models without package loss in set } P_t, \text{ Its size is denoted as } |P_{t, \gamma}| = K_\gamma. \]
\[ E[.] \] is the general expectation involving \( E[\xi] \) and \( E_{P_{t, \gamma}} \). In the following proof, without specific explanation, \( E[.] \) will be taken to denote expectations for short.

Based on these notations, some related terms of the global model and gradients are defined as follows.

A. Assumptions of bounds on training gradients

Some assumptions on the SGD gradient are given by referring to [3, 4].

Assumption 3: The stochastic gradient on the local data set suffers a variance upper-bounded by
\[ E[||\overline{g}^j_t - g^j_t||^2] \leq \frac{C_1||\overline{g}^j_t||^2 + \sigma^2}{D}, \]  
where \( C_1 \) is the stochastic coefficient of the variation of gradients and \( \sigma^2 \) is the variance of noise in sampling. The gradient variance is inversely proportional to the local data size \( D_j = \frac{D}{P_j} \), where \( D \) is the average data size and \( l_j \) is the specific ratio for client \( j \). Taking client weight \( q_j \) as \( q_j \propto D_j \), there is
\[ \sum_{j=1}^{n} q_j l_j = 1. \]  

Assumption 4: Given \( \{\pi_j \} \) as general client weights with \( \sum_{j=1}^{n} \pi_j = 1 \) and \( \lambda \) in Definition 3, the weighted gradients are supposed to be upper-bounded by
\[ C_1 \sum_{j=1}^{N} \pi_j g_{kj}^l ||^2 + \frac{\sigma^2}{\lambda} \leq G^2, \]  
where \( \sum_{j=1}^{N} \pi_j g_{kj}^l \) is a weighted aggregation of the local model. Under Assumption 3, \[ C_1 \sum_{j=1}^{N} \pi_j g_{kj}^l ||^2 \] is upper-bounded by
\[ C_1 \sum_{j=1}^{N} \pi_j g_{kj}^l ||^2 \leq \frac{G^2}{\phi_0}, \]  
where \( \phi_0 \) is a parameter related to the ratio between \( G^2 \) and the weighted gradients, which is affected by \( \sigma^2 \).
C. A lemma for summation in random set $P_{t,γ}$

**Lemma 1:** Given $P_t$ with $|P_t| = K$ as set of clients for model aggregation, $P_{t,γ}$ with $|P_{t,γ}| = K_γ$ is the set of the actually received model by central server. Suppose that the package loss rate is i.i.d. in a wireless network with the uniform worst case $K_γ ≥ K(1 - γ)$, then the summation in random set $P_{t,γ}$ satisfies the following equations:

\[
E_{P_{t,γ}}[\frac{1}{K_γ} \sum_{j \in P_{t,γ}} x_j] = \sum_{j=1}^{N} q_j x_j,
\]

\[
E_{P_{t,γ}}[\frac{1}{K_γ^2} \sum_{j \in P_{t,γ}} x_j] ≤ \frac{1}{K(1-γ)} \sum_{j=1}^{N} q_j x_j,
\]

where $q_j$ is the probability of choosing client $j$ in $P_t$ and $x_j$ denotes an arbitrary random variable from client $j$.

**Proof A.1:** Setting $f_0(K_γ)$ to be $\frac{1}{K_γ}$ or $\frac{1}{K_γ^2}$, then the summation can be deduced as follows.

\[
E_{P_{t,γ}}[f_0(K_γ) \sum_{j \in P_{t,γ}} x_j] = E_{P_t}E_γ[f_0(K_γ) \sum_{j \in P_t} x_j]
\]

\[= E_γE_{P_t}[f_0(K_γ) \sum_{j \in P_t} x_j]
\]

\[= E_γ[f_0(K_γ)E_{P_t}[\sum_{j \in P_t} x_j]
\]

\[= E_γ[f_0(K_γ) \sum_{j \in P_t} E_{P_t}[x_j]]
\]

\[= E_γ[f_0(K_γ) \sum_{j \in P_t} q_j x_j]
\]

\[= \sum_{j=1}^{N} q_j x_j E_γ[f_0(K_γ)K_γ],
\]

where $\sum_{j=1}^{N} q_j x_j$ is due to the assumption that $γ$ is i.i.d. across the clients. By focusing on the worst case $K_γ ≥ K(1 - γ)$, such an assumption is reasonable. $\sum_{j=1}^{N} q_j x_j$ is due to the assumption that client $j$ is selected by probability $q_j$. By taking in $f_0(K_γ) = \frac{1}{K_γ}$ or $f_0(K_γ) = \frac{1}{K_γ^2}$, the proof is completed.

**APPENDIX B**

**Lemma 2:** Suppose that clients are selected in $P_t$ by probability \{q_j\}, under Assumption 3, Definition 1, (37), (29) and (30), then $E[||\bar{g}_t||^2]$ is upper-bounded by

\[
E[||\bar{g}_t||^2] ≤ \frac{\lambda C_1}{DK(1-γ)} \sum_{j=1}^{N} ||q_j l_j g_j^t||^2 + \frac{1}{K(1-γ)} \sum_{j=1}^{N} ||q_j g_j^t||^2 + \frac{1}{K_γ} \sum_{j=1}^{N} ||q_j g_j^t||^2.
\]

**Proof:**

It is natural to see that $\bar{g}_t = E[\tilde{g}_t]$. Due to the basic properties of expectation, $E[||\bar{g}_t||^2]$ can be transformed as

\[
E[||\bar{g}_t||^2] = E[||\tilde{g}_t - E[\tilde{g}_t]||^2] + ||E[\tilde{g}_t]||^2
\]

\[= E[||\tilde{g}_t - \tilde{g}_t||^2] + ||\tilde{g}_t||^2.
\]

(40)

$||\bar{g}_t||^2$ in (40) can be upper-bounded as follows due to Jensen’s inequality.

\[
||\bar{g}_t||^2 = \frac{1}{K_γ} \sum_{j \in P_{t,γ}} ||g_j^t - \tilde{g}_t||^2 + \sum_{i \neq j} <\tilde{g}_t^j - g_i^t, \tilde{g}_t - g_i^t>
\]

\[= E_γ \frac{1}{K_γ} \sum_{j \in P_{t,γ}} ||\tilde{g}_t - g_i^t||^2
\]

\[= \frac{1}{K_γ^2} \sum_{j \in P_{t,γ}} ||\tilde{g}_t - g_i^t||^2,
\]

where $\sum_{j \in P_{t,γ}} ||\tilde{g}_t - g_i^t||^2$ is due to the assumption that the local data sampling is independent among clients. That is, the error $\tilde{g}_t^j - g_i^j$ resulting from local sampling in client $i$ is independent from $\tilde{g}_t^j - g_i^j$ in client $j$. Note that the gradients themselves are not independent among clients, though their errors are in fact independent. Then from (33) and Lemma 1, there is

\[
E[||\tilde{g}_t||^2] ≤ \frac{E_{P_{t,γ}}[\frac{1}{K_γ^2} \sum_{j \in P_{t,γ}} q_j l_j g_j^t ||^2 + \frac{1}{K_γ} \sum_{j \in P_{t,γ}} ||q_j g_j^t||^2 + \frac{1}{K_γ} \sum_{j \in P_{t,γ}} ||q_j g_j^t||^2] ≤ \frac{C_1}{DK(1-γ)} \sum_{j=1}^{N} q_j l_j ||g_j^t||^2 + \frac{1}{K(1-γ)} \sum_{j=1}^{N} ||q_j g_j^t||^2
\]

\[\leq \frac{C_1}{D} \sum_{j=1}^{N} q_j l_j ||g_j^t||^2 + \frac{1}{K(1-γ)} \sum_{j=1}^{N} ||q_j g_j^t||^2.
\]

(43)

Taking in metric $\lambda$ from Definition 1, the proof is completed.

**APPENDIX C**

**Lemma 3:** Under Assumption 3, Definition 1, (39), (29), (28) and (31), suppose clients are selected associated with weight \{q_j\} and $\eta_t$ diminishing by $O(\frac{1}{t})$, then $-\eta_t E[<\nabla f(\tilde{w}_t), \tilde{g}_t>]$ is upper-bounded as follows.

\[
-\eta_t E[<\nabla f(\tilde{w}_t), \tilde{g}_t>] ≤ \frac{1}{2} ||\nabla f(\tilde{w}_t)||^2 - \frac{1}{2} \sum_{j=1}^{N} q_j g_j^t ||^2
\]

\[+ \eta_t^2 L^2 \alpha \frac{1}{2D} \sum_{k \in \mathcal{C}_t} ||q_j l_j g_j^t ||^2
\]

\[+ \frac{E_{P_t} \eta_t^2 L^2 \sigma^2}{2D} \frac{\lambda - 1}{K(1-γ)} \sum_{j=1}^{N} ||q_j l_j g_j^t ||^2
\]

\[= E_{P_t} \frac{\eta_t^2 L^2 \sigma^2}{2D} \frac{\lambda - 1}{K(1-γ)} \sum_{j=1}^{N} ||q_j l_j g_j^t ||^2.
\]

(44)
From the basic definitions (30), (29), (28) and (31), \(-E[\nabla f(\mathbf{w}_t), \mathbf{g}_t] \geq c\) can be transformed as follows.

\[
-E[\nabla f(\mathbf{w}_t), \mathbf{g}_t] = -\frac{1}{2} \|\mathbf{g}_t\|^2 + \frac{1}{2} \| \mathbf{w}_t + \mathbf{g}_t \|^2
\]

where (1) is due to \(E[|x - E(x)|^2] = E[|x|^2 - |E(x)|^2]\). The super index \(l\) is applied to differ from \(j\), both represent mark for a client.

In (50), \(\sum_{k=1}^{N_{c+1}} \eta_{c_k + k} \overline{g}_{t+c+k}^i\) is the accumulated gradient, which is also a form of the local gradient \(\nabla f_j(w)\).

From (36) in Lemma 1 \(E_{P_{c,y}}[\frac{1}{K_y} \sum_{j \in P_{c,y}} x_j] = \sum_{j=1}^{N_{c+1}} q_j x_j\). Then \(E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2] \leq \sum_{j=1}^{N_{c+1}} q_j \| x_j \|^2\).

Then from \(E[|x|^2] = E[|x - E(x)|^2] + |E(x)|^2\) and \(E[\overline{g}_t^i] = g_f^i\), (52) further leads to

\[
E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \overline{w}_t - w_t^f \|^2] \leq (\lambda - 1) E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \sum_{k=1}^{r} \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2]
\]

where (1) holds because \(E_c(\overline{g}_{t+c+k}^i - g_{t+c+k}^i) = 0\) and the gradient errors are independent among clients.

Then from \(\| \frac{1}{m} \sum_{j=1}^{m} |x_i| \|^2 \leq \frac{1}{m} \sum_{j=1}^{m} |x_i|^2\) (Jensen’s inequality), Eq. (53) can be bounded as

\[
(\lambda - 1) E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \sum_{k=1}^{r} \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2]
\]

where (1) is due to \(E[|x - E(x)|^2] = E[|x|^2 - |E(x)|^2]\). The super index \(l\) is applied to differ from \(j\), both represent mark for a client.

In (50), \(\sum_{k=1}^{N_{c+1}} \eta_{c_k + k} \overline{g}_{t+c+k}^i\) is the accumulated gradient, which is also a form of the local gradient \(\nabla f_j(w)\).

From (36) in Lemma 1 \(E_{P_{c,y}}[\frac{1}{K_y} \sum_{j \in P_{c,y}} x_j] = \sum_{j=1}^{N_{c+1}} q_j x_j\). Then \(E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2] \leq \sum_{j=1}^{N_{c+1}} q_j \| x_j \|^2\).

Then from \(E[|x|^2] = E[|x - E(x)|^2] + |E(x)|^2\) and \(E[\overline{g}_t^i] = g_f^i\), (52) further leads to

\[
E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \overline{w}_t - w_t^f \|^2] \leq (\lambda - 1) E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \sum_{k=1}^{r} \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2].
\]

Note again that \(\overline{w}_t\) is the virtually aggregated model weight for an arbitrary \(t\). The real model aggregation and model broadcast only occurs at \(t_c\). Then due to \(47\), \(48\) and \(49\), there is

\[
E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \overline{w}_t - w_t^f \|^2]
\]

where (1) holds because \(E_c(\overline{g}_{t+c+k}^i - g_{t+c+k}^i) = 0\) and the gradient errors are independent among clients.

Then from \(\| \frac{1}{m} \sum_{j=1}^{m} |x_i| \|^2 \leq \frac{1}{m} \sum_{j=1}^{m} |x_i|^2\) (Jensen’s inequality), Eq. (53) can be bounded as

\[
(\lambda - 1) E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \sum_{k=1}^{r} \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2]
\]

where (1) is due to \(E[|x - E(x)|^2] = E[|x|^2 - |E(x)|^2]\). The super index \(l\) is applied to differ from \(j\), both represent mark for a client.

In (50), \(\sum_{k=1}^{N_{c+1}} \eta_{c_k + k} \overline{g}_{t+c+k}^i\) is the accumulated gradient, which is also a form of the local gradient \(\nabla f_j(w)\).

From (36) in Lemma 1 \(E_{P_{c,y}}[\frac{1}{K_y} \sum_{j \in P_{c,y}} x_j] = \sum_{j=1}^{N_{c+1}} q_j x_j\). Then \(E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2] \leq \sum_{j=1}^{N_{c+1}} q_j \| x_j \|^2\).

Then from \(E[|x|^2] = E[|x - E(x)|^2] + |E(x)|^2\) and \(E[\overline{g}_t^i] = g_f^i\), (52) further leads to

\[
E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \overline{w}_t - w_t^f \|^2] \leq (\lambda - 1) E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \sum_{k=1}^{r} \eta_{c_k + k} \overline{g}_{t+c+k}^i \|^2].
\]

Note again that \(\overline{w}_t\) is the virtually aggregated model weight for an arbitrary \(t\). The real model aggregation and model broadcast only occurs at \(t_c\). Then due to \(47\), \(48\) and \(49\), there is

\[
E[\frac{1}{K_y} \sum_{j \in P_{c,y}} \| \overline{w}_t - w_t^f \|^2]
\]
Then under Assumption 5, considering expectation $E_{\xi} [\cdot]$, \textbf{(54)} can be bounded as follows.

$$
E_{\xi} \left[ \frac{1}{K} \sum_{j \in P_{r,\gamma}} ||\hat{w}_t - w_j^t||^2 \right] \\
\leq (\lambda - 1) \left( \frac{1}{K} \sum_{j \in P_{r,\gamma}} \sum_{k=1}^{r} \eta_{t+k}^j l_j \left( \frac{C_1}{D} ||g_{t+k}^j||^2 + \frac{\sigma^2}{D} \right) \right) \\
+ \frac{r}{K} \sum_{j \in P_{r,\gamma}} \sum_{k=1}^{r} \eta_{t+k}^j ||g_{t+k}^j||^2.
$$

(55)

Then taking expectation $E_{P_{r,\gamma}} [\cdot]$ on both sides of \textbf{(55)} and applying Lemma \textbf{1} we have

$$
\sum_{j=1}^{N} q_j ||\hat{w}_t - w_j^t||^2 \leq \\
(\lambda - 1) \left( \frac{1}{K} \sum_{j \in P_{r,\gamma}} \sum_{k=1}^{r} \eta_{t+k}^j l_j \left( \frac{C_1}{D} ||g_{t+k}^j||^2 + \frac{\sigma^2}{D} \right) \right) \\
+ r \sum_{j=1}^{N} q_j \sum_{k=1}^{r} \eta_{t+k}^j ||g_{t+k}^j||^2 \\
= (\lambda - 1) \frac{C_1}{D} \sum_{j \in P_{r,\gamma}} \sum_{k=1}^{r} \eta_{t+k}^j l_j \left( \frac{C_1}{D} ||g_{t+k}^j||^2 + \frac{\sigma^2}{D} \right) \\
+ \frac{1}{K} \sum_{j \in P_{r,\gamma}} \sum_{k=1}^{r} \eta_{t+k}^j ||g_{t+k}^j||^2 + r \sum_{j=1}^{N} q_j \sum_{k=1}^{r} \eta_{t+k}^j ||g_{t+k}^j||^2 \\
\leq \frac{\lambda - 1}{K} \sum_{k=1}^{r} \eta_{t+k}^j + (\lambda - 1) E_l \sum_{j=1}^{N} q_j \sum_{k=1}^{r} \eta_{t+k}^j ||g_{t+k}^j||^2, \\
$$

(56)

where \textbf{(53)} comes from the fact that $r \leq E_l$ in FL training. Then under \textbf{(53), (56)} is further transformed as

$$
\sum_{j=1}^{N} q_j ||\hat{w}_t - w_j^t||^2 \leq \\
\lambda - 1 \frac{C_1}{D} \sum_{k=1}^{r} \eta_{t+k}^j + (\lambda - 1) E_l \sum_{k=1}^{r} \eta_{t+k}^j \sum_{j=1}^{N} q_j ||g_{t+k}^j||^2 \\
= \lambda - 1 \frac{C_1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2 + \\
\sigma^2 \frac{\lambda - 1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2,
$$

(57)

where \{q_j\} and \{q_j l_j\} can both be viewed as set of weights with $\sum_{j=1}^{N} q_j = 1$ and $\sum_{j=1}^{N} q_j l_j = 1$. Then consider metrics of non-i.i.d. data set in Definition \textbf{1} \textbf{(57)} leads to

$$
\sum_{j=1}^{N} q_j ||\hat{w}_t - w_j^t||^2 \leq \\
\lambda - 1 \frac{C_1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2 + \\
\sigma^2 \frac{\lambda - 1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2.
$$

(58)

Considering the limited diminishing speed of $\eta_t$, it is reasonable to have $\eta_k^2 \leq \eta_t (t_c + 1 \leq k \leq t)$. Besides, according to the definition of $t_c = \frac{t}{E_l(\gamma)}$, the summation of $k$ can be upper-bounded with a range from $t_c + 1$ to $t_c + E_l$. Then, \textbf{(58)} can be transformed as follows.

$$
\sum_{j=1}^{N} q_j ||\hat{w}_t - w_j^t||^2 \leq \\
\frac{\lambda - 1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2 + \\
E_l \eta_t \sigma^2 \frac{\lambda - 1}{D} \sum_{k=t+c+1}^{r} \eta_{k}^j \sum_{j=1}^{N} q_j ||g_{k}^j||^2.
$$

(59)

By combining \textbf{(59)} with \textbf{(43)}, the proof is completed.

**APPENDIX D**

**LEMMA 4**

**Lemma 4:** If $E[f(\hat{w}_t + 1)] - f^* \leq \eta_t^2 M + (1 - \mu \eta_t) [E[f(\hat{w}_t)] - f^*]$ holds and $\eta_t \propto O(\frac{1}{t})$, then the training loss converges as

$$
E[f(\hat{w}_t)] - f^* = \frac{1}{t} \max \{ \frac{4}{\mu^2} M, 2L \lambda_1 \},
$$

(60)

where $\lambda_1$ is the initial loss. $L$ is from Assumption \textbf{1} and $\lambda$ is the non-i.i.d. metric in Definition \textbf{1}.

**Proof:**

Assume that the training loss function of FL satisfies the following form.

$$
E[f(\hat{w}_t + 1)] - f^* \leq \eta_t^2 M + (1 - \mu \eta_t) [E[f(\hat{w}_t)] - f^*], \quad (61)
$$

where $M$ is short for a closed-form math expression. We denote $\Delta_t = E[f(\hat{w}_t)] - f^*$, then \textbf{(61)} can be simplified as follows.

$$
\Delta_{t+1} \leq (1 - \mu \eta_t) \Delta_t + \eta_t^2 M. \quad (62)
$$

Generally, the learning rate can be represented as $\eta_t = \frac{\nu}{(t + \beta)^{\alpha}}$, where $\nu$ and $\beta$ are parameters from the initial $\eta_0$ and $\alpha$ represents its diminishing speed. If $\alpha \leq 1$, $\Delta_t$ is proved to be bounded in the form as follows.

$$
\Delta_t \leq \frac{X}{(t + \beta)^{\alpha}}, \quad (63)
$$

where $X$ is short for a closed-form expression which will be defined in the following proof.

The proof is supported by mathematical induction method. If $\Delta_t \leq \frac{X}{(t + \beta)^{\alpha}}$ holds, we will prove that $\Delta_{t+1} \leq \frac{X}{(t + \beta)^{\alpha}}$ holds.
only for $\alpha \leq 1$. By taking $\eta_t = \frac{v}{(t+\beta)^\alpha}$ and $\Delta_t \leq \frac{X}{(t+\beta)^\alpha}$ into (62), we get
\[
\Delta_{t+1} \leq \frac{1 - \frac{\mu v}{(t+\beta)^\alpha} \Delta_t + \frac{v}{(t+\beta)^\alpha}^2 M}{(t+\beta)^{2\alpha} X} \leq \frac{\frac{\mu v}{(t+\beta)^\alpha} \Delta_t + \frac{v}{(t+\beta)^\alpha}^2 M}{(t+\beta)^{2\alpha} X} = (t+\beta)^\alpha - \frac{\mu v}{(t+\beta)^\alpha} \Delta_t + \frac{v^2}{(t+\beta)^{2\alpha} X}.
\]
If parameter $X$ satisfies $X \geq \frac{\mu^2 M}{(t+\beta)^{2\alpha}}$, where $\mu v \geq 1$, there is
\[
\frac{v^2 M}{(t+\beta)^{2\alpha}} - \frac{\mu^2 M}{(t+\beta)^{2\alpha}} X \leq 0.
\]
Then, the following bound holds.
\[
\Delta_{t+1} \leq \frac{(t+\beta)^\alpha - 1}{(t+\beta)^{2\alpha} X}.
\]
Therefore, the problem reduces to proving $\frac{(t+\beta)^\alpha - 1}{(t+\beta)^{2\alpha} X} \leq \frac{1}{(t+\beta)^{2\alpha} X}$. Denote $(t+\beta) = y$, there is $y \geq 1$. Then
\[
\frac{(t+\beta)^\alpha - 1}{(t+\beta)^{2\alpha} X} \leq \frac{1}{(t+\beta)^{2\alpha} X} \iff y^{2\alpha} \geq (y^{\alpha} - 1)(1+y)^{\alpha} \iff 1 \geq (1 - \frac{1}{y^{\alpha}})(1 + \frac{1}{y^{\alpha}}) = \frac{y^{2\alpha} - 1}{y^{2\alpha}}.
\]
Define function $h(m) = (1-m^\alpha)(1+m)^\alpha$, where $0 \leq m \leq 1$. Proof of (66) corresponds to proving $h(m) \leq 1$. The derivative of $h(m)$ is
\[
h’(m) = \alpha (1+m)^{\alpha-1}(1-2m^\alpha - m^{\alpha-1}).
\]
From definition of $y$, $m = \frac{1}{t+\beta}$. Now let us observe (67). Firstly, $h(0) = 1$ holds. If $\alpha > 1$, $h’(m)$ is definitely positive as $m$ approaches 0, which means $h(m) \leq 1$ does not hold for $0 \leq m \leq 1$. If $\alpha < 1$, $h’(m)$ remains negative for $m \in [0, 1]$, which ensures that bound (66) holds for $t \geq 1$. Therefore, it leads to the conclusion that $\Delta_t \leq \Delta_{t+1} \leq \frac{X}{(t+\beta)^{2\alpha} X}$ for $\alpha \leq 1$. Considering initial loss, let $\Delta_t \leq \frac{X}{(t+\beta)^\alpha}$, parameter $X$ should be
\[
X = \max \left\{ \frac{v^2 M}{\mu v - 1}, (\beta + 1)^\alpha \Delta_1 \right\}.
\]
Set $v = \frac{2}{\mu}$, $\beta + 1 = 2L\alpha$ and $\alpha = 1$, it gets to
\[
E[f(\vec{w}_t)] - f^* \leq \frac{1}{t+\beta} \max \left\{ \frac{4}{\mu^2} M, 2L\alpha \Delta_1 \right\}.
\]
Omitting the unimportant bias $\beta$, the proof is completed with results in Lemma 4.

**APPENDIX E**

**PROOF OF THEOREM 1**

Under (31) and Assumption 1, the training loss function satisfies
\[
E[f(\vec{w}_{t+1}) - f(\vec{w}_t)] \leq -\eta_t E[\nabla f(\vec{w}_t), \vec{g}_t; \vec{g}_t] + \frac{\eta_t^2 L}{2} E[\|\vec{g}_t\|^2].
\]
From Lemma 2 in Appendix B, $E[\|\vec{g}_t\|^2]$ is bounded as
\[
\frac{\Lambda C_1}{DK(1-\gamma)} \sum_{j=1}^N ||q_j l_j g_j^t||^2 + \frac{1}{K(1-\gamma)} \frac{\sigma^2}{D} + \Lambda \sum_{j=1}^N ||q_j g_j^t||^2.
\]
From Lemma 3 in Appendix C, $-\eta_t E[\nabla f(\vec{w}_t), \vec{g}_t; \vec{g}_t]$ is upper-bounded as follows.
\[
-\eta_t E[\nabla f(\vec{w}_t), \vec{g}_t; \vec{g}_t] \leq \frac{1}{2} ||\nabla f(\vec{w}_t)||^2 - \frac{1}{2} \sum_{j=1}^N ||q_j g_j^t||^2 + \eta_t^2 L^2 \frac{\lambda - 1}{K(1-\gamma)} \frac{C_1}{2D} \sum_{k=\eta+t}^N ||q_j l_j g_k^t||^2 + \eta_t^2 L^2 \frac{\lambda - 1}{K(1-\gamma)} \frac{C_1}{2D} \sum_{k=\eta+t}^N ||q_j l_j g_k^t||^2.
\]
By properties of $\mu$-$\nu$-L condition in Assumption 2, term $-\frac{1}{2} \sum_{j=1}^N ||\nabla f(\vec{w}_t)||^2$ in Lemma 3 can be substituted by $-\mu \int f(x) - f^*$. Then applying Lemma 2 and 3 to (70), it gets to
\[
E[f(\vec{w}_{t+1})] - f^* \leq (1 - \mu \eta_t) [E[f(\vec{w}_t)] - f^*] + \eta_t^2 L^2 \frac{\lambda - 1}{K(1-\gamma)} \frac{C_1}{2D} \sum_{k=\eta+t}^N ||q_j l_j g_k^t||^2 + \eta_t^2 L^2 \frac{\lambda - 1}{K(1-\gamma)} \frac{C_1}{2D} \sum_{k=\eta+t}^N ||q_j l_j g_k^t||^2.
\]
Set the learning rate as $\eta_t \leq \frac{1}{\Lambda L \lambda}$, it leads to $\eta_t^2 L^2 \lambda \sum_{j=1}^N ||q_j l_j g_k^t||^2 - \frac{\eta_t^2 L}{K(1-\gamma)} \sum_{j=1}^N ||q_j g_j^t||^2 \leq 0$. Then this term in the above upper-bound can be removed. Therefore, by simple transformation, (73) is equivalent to
\[
E[f(\vec{w}_{t+1})] - f^* \leq (1 - \mu \eta_t) [E[f(\vec{w}_t)] - f^*] + \eta_t^2 L^2 \frac{\lambda - 1}{K(1-\gamma)} \frac{C_1}{2D} \sum_{k=\eta+t}^N ||q_j l_j g_k^t||^2 + \frac{\sigma^2}{\lambda} + \frac{\eta_t^2 L}{K(1-\gamma)} \sum_{j=1}^N ||q_j l_j g_k^t||^2.
\]
Under (33) and Assumption 4, term $C_1 ||q_j l_j g_k^t||^2 + \frac{\sigma^2}{\lambda}$ is upper-bounded by $G^2$. By definitions in (33), it has $C_1 ||q_j l_j g_k^t||^2 \leq C_1 G^2$. Then it leads to
\[
E[f(\vec{w}_{t+1})] - f^* \leq (1 - \mu \eta_t) [E[f(\vec{w}_t)] - f^*] + \eta_t^2 L^2 \frac{\lambda - 1}{DK(1-\gamma)} \frac{C_1}{2LKD} \frac{1}{\gamma} E_t + \frac{\lambda - 1}{K(1-\gamma)} \frac{1}{2D}.
\]
\[ E[f(\tilde{w}_{t+1})] - f^* \leq \frac{1}{t} \max \left\{ \frac{(\lambda - 1)E_l}{2C_1\phi_0} + \frac{1}{2 KlK_D(1-\gamma)} E_l + \frac{1}{4E_l} + \frac{\lambda - 1}{K(1-\gamma)} \frac{1}{2D} \right\}. \] 

(79)

Note that \( \frac{1}{KlK_D(1-\gamma)} E_l \) is relatively small compared with \( \frac{1}{4E_l} \). Therefore, it can just be omitted for simplicity, which will not cause large effects to the tendency of training parameters. Then we have

\[ E[f(\tilde{w}_{t+1})] - f^* \leq \frac{1}{t} \max \left\{ \frac{4E_lL^2G^2\lambda}{\mu^2} \left\{ \frac{(\lambda - 1)E_l}{2C_1\phi_0} + \frac{1}{2 KlK_D(1-\gamma)} E_l + \frac{1}{4E_l} + \frac{\lambda - 1}{K(1-\gamma)} \frac{1}{2D} \right\} \right\}. \]

(80)

Define \( t_\epsilon \) as the time slot where \( E[f(\tilde{w}_{t+1})] - f^* \) reaches \( \epsilon \), the corresponding global training epoch \( G_\epsilon = \frac{t_\epsilon}{E_l} \). From (80), it is straightforward to get

\[ G_\epsilon = \frac{4E_lL^2G^2\lambda}{\mu^2} \left\{ \frac{(\lambda - 1)E_l}{2C_1\phi_0} + \frac{1}{2 KlK_D(1-\gamma)} E_l + \frac{1}{4E_l} + \frac{\lambda - 1}{K(1-\gamma)} \frac{1}{2D} \right\}. \]

(81)

Here the proof is completed.