Spatio-Temporal Federated Learning for Massive Wireless Edge Networks

Chun-Hung Liu†, Kai-Ten Feng‡, Lu Wei*, and Yu Luo†
Department of Electrical & Computer Engineering, Mississippi State University, MS, USA†
Department of Electrical & Computer Engineering, National Yang Ming Chiao Tung University, Hsinchu, Taiwan‡
Department of Computer Science, Texas Tech University, TX, USA*
e-mail: {chliu, yu.luo}@ece.msstate.edu†; ktfeng@nycu.edu.tw‡; luwei@ttu.edu*

***Abstract*—This paper presents a novel approach to conduct highly efficient federated learning (FL) over a massive wireless edge network, where an edge server and numerous mobile devices (clients) jointly learn a global model without transporting the huge amount of data collected by the mobile devices to the edge server. The proposed FL approach is referred to as spatio-temporal FL (STFL), which jointly exploits the spatial and temporal correlations between the learning updates from different mobile devices scheduled to join STFL in various training epochs. The STFL model not only represents the realistic intermittent learning behavior from the edge server to the mobile devices due to data delivery outage, but also features a mechanism of compensating loss learning updates in order to mitigate the impacts of intermittent learning. An analytical framework of STFL is proposed and employed to study the learning capability of STFL via its convergence performance. In particular, we have assessed the impact of data delivery outage, intermittent learning mitigation, and statistical heterogeneity of datasets on the convergence performance of STFL. The results provide crucial insights into the design and analysis of STFL-based wireless networks.**

I. Introduction

Conventional machine learning techniques typically work in a centralized manner, where all clients transport the data to a central server to execute learning. Although such centralized learning techniques achieve the desired learning performance while exploiting a large amount of data from clients, they may suffer risk of transporting privacy-sensitive data. For example, centralized machine learning may not be appropriate in medical data analysis since the data of patients can not be shared or transported due to privacy and security concerns. In addition, implementing centralized machine learning over wireless networks induces other practical challenges, such as large transmission latency, heavy traffic load, and large resource consumption. For example, in addition to suffering a long latency, mobile devices may consume substantial amount of power to upload their datasets to the radio access network. This is because the backbone networks to the cloud are not optimally designed for delay-sensitive services [1], [2]. Moreover, the large amount of wireless data transportation consumes considerable network resources, thereby leading to significant networking latency. To address these issues, federated learning (FL) over wireless communication becomes a promising candidate in both academia and industry.

The mechanism of FL over a wireless network is to enable all wireless clients to train a local model using their own dataset before uploading the locally trained model to a server for global model aggregation. The server then sends the aggregated model back to all the wireless clients for the next training round. As the training process between the wireless clients and the server proceeds, the locally trained models are expected to converge to a global model. The study of FL over wireless networks is still in its infancy, yet it can be categorized into three main directions: over-the-air digital and analog data aggregation, communication and computation efficiency, privacy and security [3]–[12]. References [9] and [10], for example, studied how to achieve FL through digital and analog signal combining techniques in wireless channels. The works [6], [7] focused on the computation offloading in FL, whereas the problems regarding the reduction of edge computing and communication efficiency were addressed in [13], [14].

Most existing works in the literature including the aforementioned ones have not studied how to efficiently conduct FL over a massive wireless network, such as an internet-of-thing (IoT) network, where numerous IoT (mobile) devices jointly conduct FL under the circumstance of limited radio and/or networking resources. Therefore, there is an urgent need to devise a new large-scale FL methodology to efficiently exploit the huge amount of data stored in massive wireless networks. Our main contribution in this paper is to propose a novel FL model for a massive wireless edge network with a large number of mobile devices (clients), referred to as spatio-temporal FL (STFL), which aggregates different locally trained models into a global learning model by exploiting their spatial and temporal correlations. In addition, the proposed STFL model considers a realistic intermittent learning scenario from an edge server to mobile devices due to data delivery outage, and employs a mechanism of mitigating the impact of intermittent learning. We also propose a definition of learning capability to devise a framework of analyzing the

The work of C.-H. Liu was supported in part by the National Science Foundation under Award CNS-2006453 and in part by Mississippi State University under Grant ORED 253551-060702. The work of L. Wei was supported in part by the National Science Foundation (#2150486 and #2006612). The work of K.-T. Feng was supported in part by the Ministry of Science and Technology (MoST) under Grants 110-2221-E-A49-041-MY3, 110-2224-E-A49-001, Higher Education Sprout Project of the National Yang Ming Chiao Tung University and Ministry of Education (MoE), Taiwan.

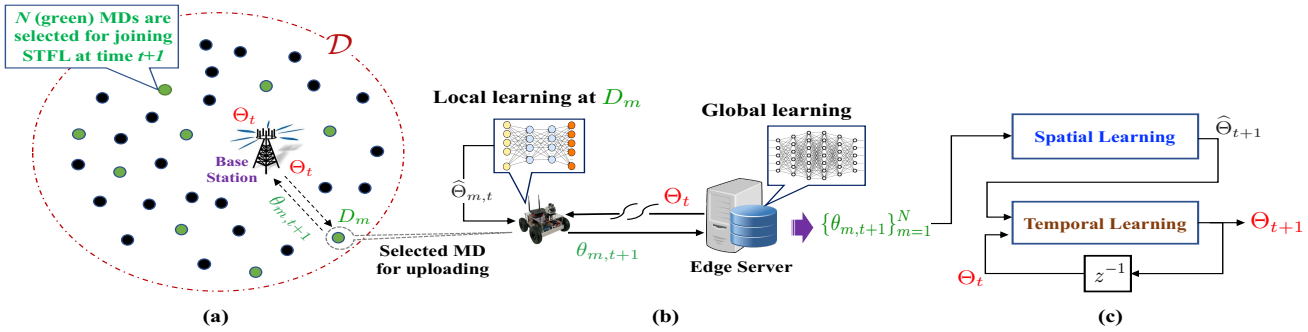

Fig. 1. (a) A massive wireless edge network in which a base station randomly selects N MDs from set $\mathcal{D}$ to join STFL in each time slot. (b) The model of STFL between the edge server and D_m selected to upload its local learning outcome $\theta_{m,t+1}$ in time slot $t+1$. (c) The block diagram of the spatio-temporal learning model at the edge server.

convergence performance of STFL. We derive a sufficient condition that ensures STFL to achieve its learning capability, which further leads to the corresponding transient analysis of STFL. Our analytical findings quantify the fundamental relationships between data delivery outage, effect of mitigating intermittent learning, and statistical heterogeneity of data in different datasets. The results provide profound insights into the improvement of the convergence rate of STFL.

II. Model of Spatio-Temporal Federated Learning

We consider a massive wireless edge network consisting of a base station (BS) and a large set of mobile devices (MDs) denoted by $\mathcal{D}$. The BS is connected to an edge server with a learning facility and helps the information exchange between the edge server and the MDs in set $\mathcal{D}$. Each MD possesses a dataset to perform a local learning algorithm. The edge server and all the MDs in the edge network aim to learn a global model (vector) by utilizing the data stored in all the datasets of the MDs. Due to privacy and other practical concerns, none of the datasets of the MDs can be transported to or accessed by any other devices in the edge network. Since set $\mathcal{D}$ is fairly large and the radio resource is limited, current prevailing FL models in the literature may not be directly adopted in such a scenario. In order to efficiently exploit the large amount of data stored in $\mathcal{D}$, we propose a new model of FL between the edge server and set $\mathcal{D}$ as follows. In each training epoch of the learning process, the BS randomly selects N MDs from $\mathcal{D}$ to join the learning process, which is merely a small portion of the MDs in $\mathcal{D}$. Each of the selected MDs utilizes its own dataset to train the local learning model before uploading its training outcome to the edge server through the BS. The edge server then aggregates all the received local models into a global model and sends it back to *all* the MDs to complete a training epoch.

To ensure the learning process will efficiently exploit the data from the N selected datasets, we propose a novel FL model as follows. Suppose MD m is selected to upload its local learning model at time t and its dataset can be expressed as

$$\mathcal{S}_m \triangleq \{S_{m,i} \in \mathbb{R}^d \times \mathbb{R} : S_{m,i} = (x_{m,i}, y_{m,i}), \\ x_{m,i} \in \mathbb{R}^d, y_{m,i} \in \mathbb{R}\}, \tag{1}$$

where $S_{m,i}$ denotes data point i in dataset $\mathcal{S}_m$, $x_{m,i}$ is a d-dimensional (d-D) data vector with d feature elements, and $y_{m,i}$ is the labeled (scalar) output corresponding to $x_{m,i}$. At (training epoch) time t, MD D_m receives the global model $\Theta_t \in \mathbb{R}^d$ sent by the BS and uses this Θ_t to update its local learning model $\theta_{m,t+1}$ via the following algorithm:

$$\begin{cases} \Theta_{m,t} & = \gamma_{m,t}\Theta_t + (1-\gamma_{m,t})\widehat{\Theta}_{m,t} \\ \mathbf{g}_m(\Theta) & = \frac{1}{|\mathcal{S}_m|}\sum_{S_{m,i}\in\mathcal{S}_m} \nabla_\Theta \ell(S_{m,i},\Theta)\,. \\ \theta_{m,t+1} & = \Theta_{m,t} - \alpha_m \mathbf{g}_m\left(\Theta_{m,t}\right) \end{cases} \tag{2}$$

Here, $\gamma_{m,t} \in \{0,1\}$ is a Bernoulli random variable that characterizes the data delivery outage from the edge server to D_m[1], $\widehat{\Theta}_{m,t}$ is the estimate of Θ_t once Θ_t is lost, $\alpha_m > 0$ denotes the *spatial* learning rate for MD m, $|\mathcal{S}_m|$ denotes the number of the data points in $\mathcal{S}_m$, $\ell(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}_+$ is a differentiable *loss* function[2], and $\nabla_\Theta \ell(\cdot,\Theta)$ represents the gradient of $\ell(\cdot,\Theta)$ with respect to argument Θ. The function $\widehat{\Theta}_{m,t}$ is to compensate Θ_t when MD D_m does not receive it at time t and we will demonstrate how to determine $\widehat{\Theta}_{m,t}$ in Section III. The learning model between the edge server and MD D_m in a massive wireless edge network is illustrated in Fig. 1(a) and (b).

At time $t+1$, the N MDs selected by the BS upload their locally trained models to the edge server through the BS. The edge server then adopts the following algorithm to find Θ_{t+1},

$$\begin{cases} \widehat{\Theta}_{t+1} & = \sum_{m=1}^N \frac{|\mathcal{S}_{m,t}|}{|\mathcal{S}_t|}\theta_{m,t+1} \\ \Theta_{t+1} & = \Theta_t - \beta_t\left(\Theta_t - \widehat{\Theta}_{t+1}\right) \end{cases}, \tag{3}$$

where $\beta_t \in (0,1)$ is known as the *temporal* learning rate, $\mathcal{S}_{m,t}$ denotes the dataset of an MD D_m selected at time t, and $|\mathcal{S}_t| \triangleq \sum_{m=1}^N |\mathcal{S}_{m,t}|$. The algorithm for $\widehat{\Theta}_{t+1}$ is the spatial learning over the N datasets $\{\mathcal{S}_{m,t}\}_{m=1}^N$, whereas the recursive algorithm for Θ_{t+1} conducts the temporal learning over all $\overline{\Theta}_t$ before time $t+1$. The algorithm in (3) is motivated by the idea of the moving average among the t global learning

[1]The data delivery outage from the server to an MD could be caused, among others, by wireless channel fading, network congestion, and queuing delay.

[2]The loss function $\ell(\cdot,\cdot)$ can be generally designed to represent various learning models adopting a stochastic gradient decent method, e.g., support vector machines, neural networks, and linear regression.

models found prior to time $t+1$ because Θ_t in (3) reduces to the moving average of the set $\{\Theta_1, \ldots, \Theta_t\}$ if $\beta_t = 1/(t+1)$. For the simplicity of modeling the learning behavior at the edge server, we assume that the data delivery outage from an MD to the edge server in (3) does not impact the learning process of Θ_t[3]. *The global learning algorithm in* (3) *can exploit the spatial (local) learning outcomes across different times and datasets distributed over the edge network* since the global models found at different times are likely based on different local datasets. As a result, the local learning algorithm in (2) and the global learning algorithm in (3) are referred to as STFL in this paper. Thus, the proposed STFL model is able to achieve the spatial average over the set $\mathcal{D}$ as $t \to \infty$ since $\mathcal{D}$ is large according to the mean-field theory [15]. Note that the proposed STFL model reduces to the general FL model in the literature whenever $\gamma_{m,t} = \beta_t = 1$ for all $t \in \mathbb{N}$. This manifests the fact that a general FL model cannot handle data delivery outage in the presence of a huge amount of data widely distributed over a massive edge network. The block diagram of the STFL algorithm is shown in Fig. 1(c). In the following section, we will study the convergence performance of STFL.

III. Convergence Analysis of STFL

In this section, we investigate the convergent performance of the proposed STFL so as to understand how data delivery outages impact federated learning. Let $\Theta_* \in \mathbb{R}^d$ be the *target* global model that all the MDs would like to learn, that is, all the local models in (2) are expected to converge to Θ_* as time goes to infinity. We will first analyze the steady-state behavior of STFL to understand when the proposed STFL algorithm is able to accurately learn the global model. We will then focus on the transient analysis of STFL in understanding the transient behavior and convergence rate.

A. Analysis of the Learning Capability of STFL

To evaluate if STFL is able to learn in a long-time sense, we propose the notion of "learning capability" for the proposed STFL, as defined in the following.

Definition 1 (Learning capability of STFL)**.** *Let* $\epsilon_{m,t} \triangleq \mathbb{E}\left[\|\theta_{m,t} - \Theta_*\|^2\right]$ *be the learning error of the local model* $\theta_{m,t}$ *of MD* $\mathcal{D}_m$ *at time* t*. The proposed STFL is said to possess the learning capability if* $\epsilon_{m,t}$ *converges to zero as* t *goes to infinity for all* $D_m \in \mathcal{D}$*.*

Definition 1 manifests the fact that the proposed STFL possesses a learning capability if the learning error of all the local models is sufficiently small after a long training time. According to this definition, we can find the condition that enables STFL to possess a learning capability, as shown in the following theorem.

Theorem 1. *If each MD* $D_m \in \mathcal{D}$ *is able to estimate its missed global model such that* $\mathbb{E}[\|\Theta_t - \widehat{\Theta}_{m,t}\|^2] \leq \delta_m \epsilon_{m,t}$ *holds for a small* $\delta_m > 0$ *and all* $t \in \mathbb{N}$*, then the STFL algorithm proposed in* (2) *and* (3) *possesses the learning capability if the following inequality holds*

$$\max_{m:D_m \in \mathcal{D}} \left\{ \sqrt{(1+q_m\delta_m)}\sigma_m(\Theta) \right\} < 1, \tag{4}$$

where $q_m = \mathbb{P}[\gamma_{m,t} = 0]$ *is the outage probability of the data delivery from the edge server to* D_m *and* $\sigma_m(\Theta)$ *is the spectral radius of matrix* $\mathbf{I}_d - \alpha_m \nabla_\Theta \mathbf{g}_m(\Theta)$*, and* $\mathbf{I}_d$ *denotes a* $d \times d$ *identity matrix.*

Proof: See Appendix A. ∎

The inequality in Theorem 1 reveals some important implications, which are worth pointing out in the following. First of all, the result characterizes how the data delivery outage and the accuracy of estimating the missed global model at each MD impact the learning capability of STFL. The data delivery outage brings a negative impact on the learning capability because it increases the right hand side of the inequality in (4), which makes (4) more difficult to hold. Nonetheless, such a negative impact can be significantly mitigated if all δ_m's are very small, which is accomplished whenever each MD is able to accurately estimate and compensate its lost global model. We will propose a stochastic approximation algorithm for each MD to estimate the global model and numerically evaluate its estimation performance in the following section. In addition to accurately compensating the lost global model, another effective method that helps STFL equip with the learning capability is to find an optimal α_m that minimizes $\sqrt{(1+q_m\delta_m)}\sigma_m(\Theta)$. If $\lambda_{m,i} > 0$ denotes one of the eigenvalues of $\nabla_\Theta \mathbf{g}_m(\Theta)$, then $\sigma_m(\Theta) = \max_i\{|1-\alpha_m\lambda_{m,i}|\}$ and the following optimization problem for α_m can be formulated,

$$\min_{\alpha_m} \left\{ \max_{i\in\{1,\ldots,d\}} \{|1-\alpha_m\lambda_{m,i}|\} \right\} \tag{5}$$

$$\text{s.t. } \frac{\sqrt{(1+q_m\delta_m)}-1}{\lambda_m^{\max}\sqrt{(1+q_m\delta_m)}} < \alpha_m < \frac{\sqrt{(1+q_m\delta_m)}+1}{\lambda_m^{\max}\sqrt{(1+q_m\delta_m)}}, \tag{6}$$

where (6) is derived from the constraint $\sigma_m(\Theta) = \max_i\{|1-\alpha_m\lambda_{m,i}|\} < 1$ and $\lambda_m^{\max} \triangleq \max\{\lambda_{m,1}, \ldots, \lambda_{m,d}\}$. The solution to this optimization problem can be found as

$$\alpha_m^* = \frac{2}{\lambda_m^{\max} + \lambda_m^{\min}} \tag{7}$$

if the following inequality holds

$$1 < \frac{\lambda_m^{\max}}{\lambda_m^{\min}} < \frac{\sqrt{(1+q_m\delta_m)}+1}{\sqrt{1+q_m\delta_m}-1},$$

where $\lambda_m^{\min} = \min\{\lambda_{m,1}, \ldots, \lambda_{m,d}\}$ and $\frac{\lambda_m^{\max}}{\lambda_m^{\min}}$ is known as the condition number of $\nabla_\Theta \mathbf{g}_m(\Theta)$. Apparently, the condition number of $\nabla_\Theta \mathbf{g}_m(\Theta)$ has a profound impact on the learning capability of STFL. For example, if $\frac{\lambda_m^{\max}}{\lambda_m^{\min}}$ is small and close to one (i.e., $\lambda_{m,1} \approx \lambda_{m,i} \approx \lambda_{m,d} \approx \lambda_m$), $\alpha_m^* \approx \frac{1}{\lambda_m}$, thereby $1 - \alpha_m^*\lambda_{m,i} \approx 0$. In such a scenario, STFL will readily possess the learning capability by using the optimal learning rate α_m^* even if it may seriously suffer from the data delivery outage. A large condition number usually leads to small α_m

[3]This is a reasonable assumption as it is very unlikely that many of the N MDs fail to transmit their data to the server at the same time, where the temporal learning also mitigates the effects of data delivery outage.

in order to enable the learning capability, thereby reducing the convergence rate of STFL. Another important implication that can be learned from (4) is how the different distributions of the non-i.i.d. datasets in the edge network affect the learning capability. This is because whether or not (4) holds hings on the eigenvalues of $\nabla \mathbf{g}_m(\Theta)$ that are dependent on the distribution of the data point in dataset $\mathcal{S}_m$. In general, the term $\sqrt{(1+q_m\delta_m)}\sigma_m(\Theta)$ in (4) increases and α_m^* in (7) accordingly decreases as the number of the non-i.i.d. datasets in the edge network increases. Hence, Theorem 1 provides an analytical foundation on how non-i.i.d. datasets influences the learning performance. Nonetheless, we are unable to infer the transient behavior of STFL from Theorem 1 alone, the study of which is presented in the following subsection.

B. Analysis of the Transient Performance of STFL

The transient behavior of STFL can be characterized by how fast the local learning models converge within a fixed number of training epochs. Accordingly, we propose the concept of the learning time constant of STFL specified in the following definition.

Definition 2 (Learning Time Constant of STFL)**.** *Suppose STFL possesses the learning capability. The local time constant of MD D_m, denoted by τ_m, is defined as*

$$\tau_m \triangleq \inf\{T : \epsilon_{m,t+T} \leq e^{-1}\epsilon_{m,t}, t \in \mathbb{N}\}. \tag{8}$$

The learning time constant of STFL is $\tau \triangleq \max_{m:D_m\in\mathcal{D}}\{\tau_m\}$.

The physical meaning of the learning time constant τ_m is the minimum time period needed to make the learning error $\epsilon_{m,t+\tau_m}$ reduce to 36.79% of $\epsilon_{m,t}$, and it is similar to the meaning of the time constant of a linear dynamic system. The learning time constant of STFL is thus defined as the maximum among all the local time constants. Therefore, the key to improving the convergence rate of STF is to reduce the learning time constant. The time constant defined in (8) can be equivalently expressed as

$$\tau_m = \inf\left\{T : \ln\left(\frac{\epsilon_{m,t}}{\epsilon_{m,t+T}}\right) \geq 1, t \in \mathbb{N}\right\}, \tag{9}$$

which can be explicitly found as shown in the following theorem.

Theorem 2. *If the STFL possesses the learning capability, the local time constant of MD D_m can be found as*

$$\tau_m = \frac{-1}{2\ln\left(\sqrt{(1+q_m\delta_m)}\sigma_m^*(\Theta)\right)}, \tag{10}$$

where $\sigma_m^*(\Theta) \triangleq \max_{i\in\{1,\ldots,d\}}\{|1-\alpha_m^*\lambda_{m,i}|\}$ and α_m^* is given in (7)*. The learning time constant of STFL is thus obtained as*

$$\tau = -\frac{1}{2}\left\{\max_{m:D_m\in\mathcal{D}} \ln\left(\sqrt{(1+q_m\delta_m)}\sigma_m^*(\Theta)\right)\right\}^{-1}. \tag{11}$$

Proof: See Appendix B. ∎

The outcomes shown in Theorem 2 clearly indicate how the convergence rate of STFL is quantitatively affected by the data delivery outage, the effect of compensating the lost global model on the MD side, and the statistical property of the datasets in the edge network. To decrease τ_m, we need to keep $\sqrt{(1+q_m\delta_m)}\sigma_m^*(\Theta)$ as small as possible. As such, both the transient behavior and the learning capability of STFL are dominated by the value of $\sqrt{(1+q_m\delta_m)}\sigma_m^*(\Theta)$. The distributions of the non-i.i.d. datasets have a much more critical impact on the transient behavior of STFL than the data delivery outage and the compensating effect. The reason is that $(1+q_m\delta_m)$ can be reduced to one, whereas $\ln(\sigma_m^*(\Theta))$ can be reduced to a value much smaller than one. To demonstrate this, consider a scenario of no data delivery outage (or the lost global learning model is perfectly compensated), the convergence performance of STFL is dominated by the maximum of $\sigma_m^*(\Theta)$,

$$\max_m\{\sigma_m^*(\Theta)\} = \max_m \max_i \left\{\left|1 - \frac{2\lambda_{m,i}}{\lambda_m^{\max}+\lambda_m^{\min}}\right|\right\}, \tag{12}$$

which approaches to zero as $\lambda_m^{\min}$ increases to $\lambda_m^{\max}$, and thereby $-\ln(\sigma_m^*(\Theta))$ can be quite large. Namely, τ can be considerably reduced as the condition number of $\nabla_\Theta \mathbf{g}_m(\Theta)$ that depends on $\mathcal{S}_m$ is close to unity. This suggests that carefully removing some features of the non-crucial input data points helps to reduce the condition number and accordingly improves the convergence rate of STFL. We will illustrate this observation in the following section.

IV. Numerical Results

In this section, some numerical results are provided to illustrate our analytical findings in the previous section. We consider the wireless edge network in Fig. 1, where 10000 MDs are uniformly distributed in set $\mathcal{D}$ and the BS randomly schedules 100 MDs to join the learning process of STFL in each training epoch. For simplicity, a linear regression problem is tackled in the wireless edge network by using STFL. As a result, the loss function becomes $\ell(S_{m,i},\Theta) = \frac{1}{2}(y_{m,i} - \Theta^T x_{m,i})^2$ so that we obtain $\nabla_\Theta \ell(S_{m,i},\Theta) = (\Theta^T x_{m,i} - y_{m,i})x_{m,i}$ and $\nabla_\Theta \mathbf{g}_m(\Theta) = \frac{1}{|\mathcal{S}_m|}\sum_{S_{m,i}\in\mathcal{S}_m} x_{m,i}x_{m,i}^T \approx \Sigma_m + \mathbb{E}[x_m]\mathbb{E}[x_m]^T$, where Σ_m is the covariance matrix of all $x_{m,i}$'s in $\mathcal{S}_m$. Each MD adopts the following algorithm to find $\widehat{\Theta}_{m,t}$ in (2),

$$\widehat{\Theta}_{m,t} = (1-\omega)\sum_{i=1}^{t}\omega^{t-i}\left[(1-\gamma_{m,i})\theta_{m,i-1} + \gamma_{m,i}\Theta_i\right], \tag{13}$$

where $\omega \in (0,1)$. Note that $\omega = 0$ corresponds to the case that each MD does not compensate its lost global model. Since it is clear that $\lim_{t\to\infty}\widehat{\Theta}_{m,t} = \Theta_*$ once $\Theta_t \to \Theta_*$ as $t \to \infty$, there must exist a fairly small δ_m such that $\|\widehat{\Theta}_{m,t} - \Theta_t\| \leq \delta_m\|\theta_{m,t} - \Theta_*\| = \delta_m\epsilon_{m,t}$. In the following simulation, $\omega = 0.25$ will be used in (13) if MD D_m needs to compensate its lost global model and consider $\delta_m \leq 1$. Moreover, all the datasets in the network are of the same size with 100 data points, i.e., $|\mathcal{S}_m| = 100$ and $x_{m,i} \in \mathbb{R}^3$ for $m = 1, 2, \ldots, 10000$ and $i = 1, 2, \ldots, 100$. The datasets are

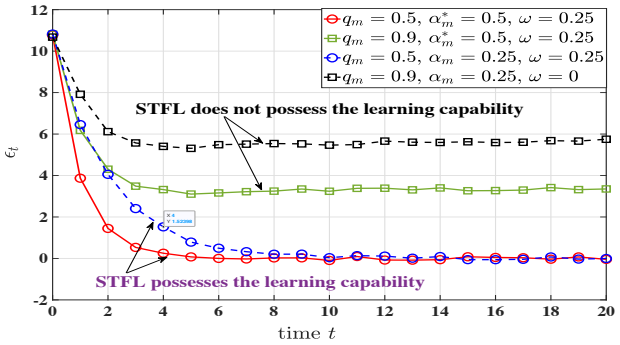

Fig. 2. The simulation results of the averaged learning error ϵ_t.

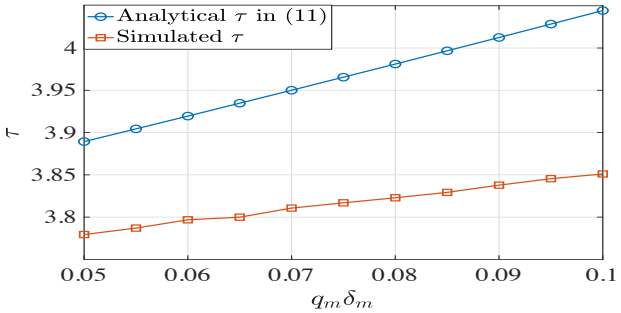

Fig. 3. The simulation results of the learning time constant τ.

non-i.i.d. and equally likely share two different 2-D Gaussian distributions with two distinct means

$$\mathbb{E}[x_{m,i}] \in \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2.2 \\ 1.8 \end{bmatrix} \right\}, \tag{14}$$

and covariance matrices

$$\Sigma_m \in \left\{ \begin{bmatrix} 1 & 1.25 \\ 1.25 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 1.75 \\ 1.75 & 2 \end{bmatrix} \right\}. \tag{15}$$

The STFL algorithm aims to learn a 2-D global model $\Theta_* = \frac{1}{\sqrt{2}}[1\ -1]^T$. Accordingly, all the input data points $x_{m,i}$ in $\mathcal{S}_m$ are generated by using (14) and (15), and all the output data points $y_{m,i}$ in $\mathcal{S}_m$ are generated by $y_{m,i} = \Theta_*^T x_{m,i}$.

According to (7) and (15), we can find $\alpha_m^* \approx 0.5$ for all $D_m \in \mathcal{D}$. We adopt $\beta_t = 1/(t+1)$ in (3), assume the same q_m for all $D_m \in \mathcal{D}$, and define the averaged learning error as $\epsilon_t = \frac{1}{N}\sum_{m=1}^N \epsilon_{m,t}$. The simulation results of ϵ_t is shown in Fig. 2 for four different combinations of q_m and α_m. As shown in the figure, all the curves eventually converge, yet two of them (i.e., the curves with squares) do not converge to zero. Namely, the two combinations, $q_m = 0.9, \alpha_m = 0.25$ and $q_m = 0.9, \alpha_m^* = 0.5$, do not make STFL possess the learning capability because they do not satisfy (4). Furthermore, the dash-square curve is the case of no global model compensation so that its ϵ_t apparently converges to a much higher value than the other three. The other two combinations satisfy (4) and indeed their curves converge to zero. Also, the circle-solid (red) curve with optimal $\alpha_m^* = 0.5$ converges much faster than the circle-dash (blue) one without α_m^*, as expected. Fig. 3 shows the simulation results of how the learning time constant τ varies with $q_m\delta_m$. As can be seen from the figure, the learning time constant found in (11) is slightly larger than the simulated one. This is because we use a (tight) bound when deriving τ. Nonetheless, τ in (11) is still very accurate so that it provides a useful evaluation on how the convergence rate of STFL is affected under different values of $q_m\delta_m$.

V. Conclusion

To effectively learn from the data stored in a massive edge network with a limited radio resource, we devised a novel STFL technique, which is able to spatially and temporally conduct the learning process among a large number of MDs. The salient features of STFL lie in two facets of modeling the realistic conditions incurred by massive wireless networking of unreliable communication and limited radio resource. The proposed STFL algorithm employs a mechanism of compensating lost global models at each MD as well as scheduling MDs at the edge server. An analytical framework for characterizing the learning capability of STFL was proposed and employed to find a sufficient condition for STFL convergence in a long-term sense. The learning time constant of STFL was defined and derived in order to evaluate the transient performance of STFL. Our analytical findings shed new light on the fundamental interplay between data delivery outage, compensation of lost data, and distribution of non-i.i.d. datasets.

Appendix

A. Proof of Theorem 1

First of all, we adopt the notation $\Delta\mathbf{z} \triangleq \mathbf{z} - \Theta_*$ for the difference between any vector $\mathbf{z}$ and the vector Θ_*. Such a notation will be frequently used in the following derivations. As a result, let us define $\Delta\bar{\theta}_{t+1} \triangleq \sum_{m=1}^N \frac{|\mathcal{S}_{m,t}|}{|\mathcal{S}_t|}\Delta\theta_{m,t+1}$ and thereby we can rewrite (3) as

$$\Delta\Theta_{t+1} = (1-\beta_t)\Delta\Theta_t + \beta_t\Delta\bar{\theta}_{t+1}.$$

This gives rise to the following inequality

$$\|\Delta\Theta_{t+1}\| \le (1-\beta_t)\|\Delta\Theta_t\| + \beta_t\|\Delta\bar{\theta}_{t+1}\|$$

due to the triangle inequality, where $\|\mathbf{z}\|$ denotes the (any) norm of vector $\mathbf{z}$. Thus, we can further obtain

$$\mathbb{E}\left[\|\Delta\Theta_{t+1}\|^2\right] \le (1-\beta_t)^2\mathbb{E}\left[\|\Delta\Theta_t\|^2\right] + 2\beta_t(1-\beta_t) \times \mathbb{E}\left[\|\Delta\Theta_t\|\|\Delta\bar{\theta}_{t+1}\|\right] + \beta_t^2\mathbb{E}\left[\|\Delta\bar{\theta}_{t+1}\|^2\right].$$

This means that $\mathbb{E}\left[\|\Delta\Theta_t\|^2\right]$ eventually converges to zero as long as we ensure $\lim_{t\to\infty}\mathbb{E}\left[\|\Delta\bar{\theta}_t\|^2\right] = 0$ and $\beta_t < 1$ in that $\lim_{t\to\infty}\mathbb{E}\left[\|\Delta\bar{\theta}_t\|^2\right] = 0$ makes $\|\Delta\bar{\theta}_t\|$ converge to zero and $\beta_t < 1$ makes $\|\Delta\Theta_t\|$ reduce to zero almost surely as time goes to infinity. In order to ensure $\mathbb{E}\left[\|\Delta\bar{\theta}_t\|^2\right]$ eventually converge to zero, we need to have $\lim_{t\to\infty}\mathbb{E}[\|\Delta\theta_{m,t}\|^2] = \lim_{t\to\infty}\epsilon_{m,t} = 0$ for all m because it guarantees $\|\Delta\theta_{m,t}\|$ reduce to zero, which leads to $\lim_{t\to\infty}\mathbb{E}\left[\|\Delta\bar{\theta}_t\|^2\right] = 0$ when considering $\mathbb{E}\left[\|\Delta\bar{\theta}_t\|^2\right] \le (\sum_{m=1}^N \frac{|\mathcal{S}_{m,t}|}{|\mathcal{S}_t|}\|\Delta\theta_{m,t}\|)^2$.

Next, $\Delta\theta_{m,t}$ in (2) can be explicitly written as

$$\begin{aligned}\Delta\theta_{m,t+1} &= \Delta\Theta_{m,t} - \alpha_m\mathbf{g}_m\left(\Delta\Theta_{m,t} + \Theta_*\right) \\ &\overset{(a)}{\approx} \Delta\Theta_{m,t} - \alpha_m\left[\mathbf{g}_m(\Theta_*) + \nabla\mathbf{g}_m(\Theta_*)\Delta\Theta_{m,t}\right] \\ &\overset{(b)}{=} \left[\mathbf{I}_d - \alpha_m\nabla\mathbf{g}_m(\Theta_*)\right]\Delta\Theta_{m,t},\end{aligned}$$

where (a) is obtained by using the first-order Taylor's expansion of $\mathbf{g}_m(\Delta\Theta_{m,t}+\Theta_*)$ and (b) is due to considering $\mathbf{g}_m(\Theta_*)=\mathbf{0}$ and a $d\times d$ identity matrix $\mathbf{I}_d$. Hence, the covariance matrix of $\Delta\theta_{m,t}$, denoted by $\mathbf{P}_{m,t}\triangleq\mathbb{E}[\Delta\theta_{m,t}\Delta\theta_{m,t}^T]$, can be expressed as

$$\mathbf{P}_{m,t+1}=\mathbf{Q}_m\mathbb{E}\left[\Delta\Theta_{m,t}\Delta\Theta_{m,t}^T\right]\mathbf{Q}_m^T, \tag{A.1}$$

where $\mathbf{Q}_m\triangleq\mathbf{I}_d-\alpha_m\nabla\mathbf{g}(\Theta_*)$ and T denotes the transpose notation of vectors and matrices. Moreover, we know

$$\begin{aligned}\mathbb{E}\left[\Delta\Theta_{m,t}\Delta\Theta_{m,t}^T\right]=&(1-q_m)\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]\\&+q_m\mathbb{E}[\Delta\widehat{\Theta}_{m,t}\Delta\widehat{\Theta}_{m,t}^T]\end{aligned}$$

because $q_m=\mathbb{P}[\gamma_{m,t}=0]$. In addition, we know $\Delta\widehat{\Theta}_{m,t}=\widehat{\Theta}_{m,t}-\Theta_t+\Theta_t-\Theta_*=\Delta\Theta_t-(\Delta\Theta_t-\Delta\widehat{\Theta}_{m,t})$ whereas $\Delta\Theta_t$ and $(\Delta\Theta_t-\Delta\widehat{\Theta}_{m,t})$ are orthogonal. We thus have $\mathbb{E}[\Delta\widehat{\Theta}_{m,t}\Delta\widehat{\Theta}_{m,t}^T]=\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]+\mathbb{E}[(\Delta\Theta_t-\Delta\widehat{\Theta}_{m,t})(\Delta\Theta_t-\Delta\widehat{\Theta}_{m,t})^T]$ so that $\mathbb{E}[\Delta\Theta_{m,t}\Delta\Theta_{m,t}^T]$ can be expressed as

$$\begin{aligned}&\mathbb{E}[\Delta\Theta_{m,t}\Delta\Theta_{m,t}^T]=\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]+q_m\\&\mathbb{E}[(\Theta_t-\widehat{\Theta}_{m,t})(\Theta_t-\widehat{\Theta}_{m,t})^T].\end{aligned}$$

Since $\Theta_t-\Theta_*$ is "less random" than $\theta_{m,t}-\Theta_*$, we know $\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]\leq\mathbb{E}[\Delta\theta_{m,t}\Delta\theta_{m,t}^T]=\mathbf{P}_{m,t}$; i.e., $\mathbf{P}_{m,t}-\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]$ is positive semi-definite. Now consider the mean square estimation error of Θ_t at D_m, i.e., $\mathbb{E}[\|\Theta_t-\widehat{\Theta}_{m,t}\|^2]$ can be bounded by $\delta_m(\mathbb{E}[\|\Delta\Theta_t\|^2])$ for all t such that $\delta_m\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]\geq\mathbb{E}[(\Theta_t-\widehat{\Theta}_{m,t})(\Theta_t-\widehat{\Theta}_{m,t})^T]$. Therefore, we have

$$\begin{aligned}\mathbb{E}[\Delta\Theta_{m,t}\Delta\Theta_{m,t}^T]&\leq\mathbf{P}_{m,t}+q_m\delta_m\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]\\&\leq(1+q_m\delta_m)\mathbf{P}_{m,t}\end{aligned} \tag{A.2}$$

due to $\mathbb{E}[\Delta\Theta_t\Delta\Theta_t^T]\leq\mathbf{P}_{m,t}$. According to (A.1) and (A.2), we can conclude the following:

$$\mathbf{P}_{m,t+1}\leq(1+q_m\delta_m)\mathbf{Q}_m\mathbf{P}_{m,t+1}\mathbf{Q}_m, \tag{A.3}$$

which leads to

$$\|\mathbf{P}_{m,t+1}\|\leq\left(\sqrt{(1+q_m\delta_m)}\|\mathbf{Q}_m\|\right)^2\|\mathbf{P}_{m,t}\|, \tag{A.4}$$

where $\|\cdot\|$ is any matrix norm. When $\sqrt{(1+q_m\delta_m)}\|\mathbf{Q}_m\|$ is smaller than unity, $\|\mathbf{P}_{m,t}\|$ will converge to zero as t goes to infinity. Namely, $\epsilon_{m,t}\to0$ as $\sqrt{(1+q_m\delta_m)}\|\mathbf{Q}_m\|<1$ for all $D_m\in\mathcal{D}$. Since all the matrix norms are equivalent, we choose $\|\mathbf{Q}_m\|_2=\sigma_m(\Theta)$, i.e., the spectral norm of $\mathbf{Q}_m$. As a result, STFL possesses the learning capability as long as the inequality in (4) holds.

B. Proof of Theorem 2

Suppose the eigenvalue decomposition of $\mathbf{Q}_m$ is $\mathbf{V}_m\mathbf{\Lambda}_m\mathbf{V}_m^T$, where $\mathbf{V}_m$ is a unitary matrix of $\nabla_\Theta\mathbf{g}_m(\Theta)$ and $\mathbf{\Lambda}_m=\text{diag}\{(1-\alpha_m\lambda_{m,1})\cdots(1-\alpha_m\lambda_d)\}$. According to (A.3) and $\text{tr}(\mathbf{P}_{m,t})=\epsilon_{m,t}$, we can obtain

$$\begin{aligned}\epsilon_{m,t+1}&\leq(1+q_m\delta_m)\text{tr}\left(\mathbf{V}_m\mathbf{\Lambda}_m\mathbf{V}_m^T\mathbf{P}_{m,t}\mathbf{V}_m\mathbf{\Lambda}_m\mathbf{V}_m^T\right)\\&\overset{(a)}{\leq}(1+q_m\delta_m)\text{tr}\left(\mathbf{\Lambda}_m^2\mathbf{V}_m\mathbf{P}_{m,t}\mathbf{V}_m^T\right)\\&\overset{(b)}{\leq}(1+q_m\delta_m)\sigma_m^2(\Theta)\text{tr}(\mathbf{P}_{m,t}),\end{aligned}$$

where (a) is due to $\text{tr}(\mathbf{AB})=\text{tr}(\mathbf{BA})$ for two matrices $\mathbf{A}$ and $\mathbf{B}$ with the same dimension, and (b) is because $\mathbf{V}_m$ is a unitary matrix such that $\text{tr}(\mathbf{V}_m\mathbf{P}_{m,t}\mathbf{V}_m^T)=\text{tr}(\mathbf{P}_{m,t})$ and $\text{tr}(\mathbf{\Lambda}_m^2\mathbf{P}_{m,t})\leq\sigma_m^2(\Theta)\text{tr}(\mathbf{P}_{m,t})=\sigma_m^2(\Theta)\epsilon_{m,t}$. Thus, we obtain

$$\ln\left(\frac{\epsilon_{m,t}}{\epsilon_{m,t+T}}\right)\geq-T\ln\left((1+q_m\delta_m)\sigma_m^2(\Theta)\right),$$

and then let the above lower bound be greater than or equal to one yields

$$T\geq\frac{-1}{\ln(1+q_m\delta_m)+\ln\left(\sigma_m^2(\Theta)\right)}.$$

The lower bound on T can be minimized by substituting α_m^* in (7) into $\sigma_m(\Theta)$ so as to ensure $\sigma_m(\Theta)$ achieve its maximum value $\sigma_m^*(\Theta)$. Therefore, τ_m in (10) is found according to (9) and τ in (11) is readily obtained based on the definition.

REFERENCES

[1] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel, A. Puschmann, A. Mitschele-Thiel, M. Muller, T. Elste, and M. Windisch, "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 70–78, Feb. 2017.

[2] S.-Y. Lien, S.-C. Hung, K.-C. Chen, and Y.-C. Liang, "Ultra-low latency ubiquitous connections in heterogeneous cloud radio access networks," *IEEE Wireless Commun.*, vol. 22, no. 3, pp. 22–31, Jun. 2015.

[3] Y. Qian, L. Hu, J. Chen, X. Guan, M. M. Hassan, and A. Alelaiwi, "Privacy-aware service placement for mobile edge computing via federated learning," *Information Sciences*, vol. 505, pp. 562–570, 2019.

[4] J. Ren, H. Wang, T. Hou, S. Zheng, and C. Tang, "Federated learning-based computation offloading optimization in edge computing supported internet of things," *IEEE Access*, no. 7, pp. 69 194–69 201, 2019.

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

[6] J. Ren, H. Wang, T. Hou, S. Zheng, and C. Tang, "Federated learning-based computation offloading optimization in edge computing-supported internet of things," *IEEE Access*, vol. 7, pp. 69 194–69 201, 2019.

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

[8] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.

[9] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.

[10] ——, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2166, Mar. 2020.

[11] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 2120–2135, Mar. 2021.

[12] C. Li, G. Li, and P. K. Varshney, "Communication-efficient federated learning based on compressed sensing," *IEEE Internet Things J.*, vol. 8, no. 20, pp. 15 531–15 541, Apr. 2021.

[13] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Arcas, "Communication-efficient learning of deep networks from decentralized data," *arXiv preprint: arXiv:1602.05629*, 2016.

[14] J. Konecný, H. B. McMahan, F. X. Yu *et al.*, "Federated learning: Strategies for improving communication efficiency," in *the 29th Conference on Neural Information Processing Systems (NIPS)*, 2016, pp. 1–6.

[15] A. Bensoussan, J. Frehse, and P. Yam, *Mean Field Games and Mean Field Type Control Theory*, 1st ed. Springer-Verlag, 2013.