Inexact-ADMM Based Federated Meta-Learning for Fast and Continual Edge Learning

Sheng Yue
Central South University
Arizona State University
sheng.yue@csu.edu.cn

Ju Ren
Central South University
renju@csu.edu.cn

Jiang Xin
Central South University
xinjiang@csu.edu.cn

Sen Lin
Arizona State University
sen.lin.1@asu.edu

Junshan Zhang
Arizona State University
junshan.zhang@asu.edu

ABSTRACT

In order to meet the requirements for performance, safety, and latency in many IoT applications, intelligent decisions must be made right here right now at the network edge. However, the constrained resources and limited local data amount pose significant challenges to the development of edge AI. To overcome these challenges, we explore continual edge learning capable of leveraging the knowledge transfer from previous tasks. Aiming to achieve fast and continual edge learning, we propose a platform-aided federated meta-learning architecture where edge nodes collaboratively learn a meta-model, aided by the knowledge transfer from prior tasks. The edge learning problem is cast as a regularized optimization problem, where the valuable knowledge learned from previous tasks is extracted as regularization. Then, we devise an ADMM-based federated meta-learning algorithm, namely ADMM-FedMeta, where ADMM offers a natural mechanism to decompose the original problem into many subproblems which can be solved in parallel across edge nodes and the platform. Further, a variant of inexact-ADMM method is employed where the subproblems are ‘solved’ via linear approximation as well as Hessian estimation to reduce the computational cost per round to $O(n)$. We provide a comprehensive analysis of ADMM-FedMeta, in terms of the convergence properties, the rapid adaptation performance, and the forgetting effect of prior knowledge transfer, for the general non-convex case. Extensive experimental studies demonstrate the effectiveness and efficiency of ADMM-FedMeta and showcase that it substantially outperforms the existing baselines.

CCS CONCEPTS

- Theory of computation → Multi-agent learning; • Networks → Network services.

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MobiHoc ’21, July 26–29, 2021, Shanghai, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8558-9/21/07...$15.00
https://doi.org/10.1145/3466772.3467038

1 INTRODUCTION

The past few years have witnessed an explosive growth of Internet of Things (IoT) devices. In many of these IoT applications, decisions must be made in real time to meet the requirements for safety, accuracy, and performance [46]. A general consensus is that the conventional cloud-based approach would not work well in these applications, calling for edge intelligence or edge AI [28, 48]. Built on a synergy of edge computing and AI, edge intelligence is expected to push the frontier of model training and inference processes to the network edge in the physical proximity of IoT devices and data sources. Nevertheless, it is highly nontrivial for a single edge node to achieve real-time edge intelligence since AI model training usually requires extensive computing resources and a large number of data samples. To tackle these challenges, we resort to continual learning capable of leveraging the knowledge transfer from previous tasks in the cloud or by other edge nodes. Simply put, continual learning (CL) is a machine learning paradigm that is designed to sequentially learn from data samples corresponding to different tasks [27]. Rather than learning the new model from scratch, CL aims to design algorithms leveraging knowledge transfer from pre-trained models to the new learning task, assuming that the training data of previous tasks are unavailable for the newly coming task (this is the case for edge learning).

To facilitate edge learning, collaborative learning has recently been proposed to leverage the model knowledge distillation, including cloud-edge collaboration and edge-edge collaboration. More specifically, a distributionally robust optimization based edge learning framework has been introduced to build a cloud-edge synergy between the pre-trained model in the cloud and the local data samples at the edge [46]. Along a different avenue, building on the recent advances in meta-learning [9, 26] and federated learning...
[24], a significant body of work has been devoted to federated meta-learning [5, 15, 21, 47] and personalized federated learning [8, 40], under a common theme of fostering edge-edge collaboration. In particular, federated meta-learning aims to learn a good model initialization (meta-model) across source edge nodes, such that the model of the new task can be learned via fast adaptation from the meta-initialization with only a few data samples at the target edge node.

Most of the existing works on federated meta-learning focus on the learning from a given set of tasks, each with its training data, but have not addressed the well-known catastrophic forgetting issue in continual learning [11] [27]. Further, the performance of the fast adaptation depends on the similarity among tasks [8, 21]. As a result, the meta-model obtained via federated meta-learning may not work well when the tasks on the target node are dissimilar to those at the source nodes. It is also worth noting that most of the existing federated meta-learning algorithms are gradient-based, which may suffer from some limitations such as vanishing gradients and sensitivity to poor conditioning [37]. It has been observed in practice that these gradient-based algorithms often exhibit slow convergence in training the meta-model, especially on complex tasks [5, 8, 15], resulting in low communication efficiency and high computational cost.

To tackle the issues noted above, in this paper, we study continual edge learning via federated meta-learning with regularization. Notably, regularization-based methods have been widely used in continual learning [17, 31, 44] and transfer learning [32, 34]. Inspired by theoretical neuroscience models via imposing constraints on the update of the neural weights [2], regularization approaches can help to alleviate catastrophic forgetting [27]. Meanwhile, valuable knowledge learned from previous tasks can be extracted as regularization to improve the training speed and the performance of the new task (so-called “positive forward transfer” [22]). Accordingly, we cast the federated meta-learning problem as a regularized stochastic optimization problem, using Bregman divergence [4] to define the regularization. Further, to reduce the computational cost and to facilitate collaborative learning, we employ the alternative direction method of multipliers (ADMM) technique to decompose the problem into a set of subproblems that can be solved in parallel across edge nodes and the platform. In particular, by “decoupling the regularizer” from the computation at local edge nodes, it suffices to run the regularization only in the platform for global aggregation. Observe that the conventional ADMM technique requires the exact solutions in each iteration, and this is the underlying rationale of the inexact-ADMM.

The main contributions of this work are summarized as follows:

- Aiming to facilitate fast and continual edge learning, we propose a platform-aided federated-meta learning architecture where edge nodes join forces to learn a meta-model with the knowledge transfer from previous tasks. We cast the edge learning problem as a regularized optimization problem, in which the transferred knowledge is in the form of regularization using Bregman divergence. We devise an inexact-ADMM based algorithm, called ADMM-FedMeta, where the ADMM technique is employed to decompose the problem into a set of subproblems that can be solved in parallel across edge nodes and the platform, and also it suffices to run the regularization only in the platform for global aggregation. Further, a variant of the inexact-ADMM method is devised where the subproblems are ‘solved’ via linear approximation as well as Hessian estimation to reduce the computational cost of per round to $O(n)$, achieving lower computational complexity compared to most of the existing methods.
- We carry out a comprehensive analysis of the proposed algorithm for the general non-convex case, where we establish the convergence and characterize the performance of fast adaptation using local samples at the target node. We also quantify the forgetting effect of model knowledge transferred from previous tasks for a special case. Besides, we show that ADMM-FedMeta can mitigate performance degradation incurred by the dissimilarity between the source nodes and the target node.
- We evaluate the performance of the proposed algorithm on different models and benchmark datasets. Our extensive experimental results showcase that ADMM-FedMeta outperforms existing state-of-the-art approaches, in terms of convergence speed, adaptation performance, and the capability of learning without forgetting, especially with small sample sizes.

2 RELATED WORK

In this section, we briefly review the related work in the following three areas.

**Meta-Learning.** Meta-learning has emerged as a promising solution for few-shot learning. Ravi et al. [30] propose an LSTM-based meta-learning model to learn an optimization algorithm for training neural networks. Different from [30], a gradient-based Model Agnostic Meta-Learning (MAML) algorithm is proposed in [9], which
aims at learning a model initialization, based on which using a few gradient descent updates can achieve satisfactory performance on a new task. To reduce the computational complexity, Nichol et al. [26] introduce a first-order meta-learning algorithm called Reptile, which does not require the computation of the second-order derivatives. Multiple follow-up works extend MAML from different perspectives, e.g., [6, 10, 29, 33]. Along a different line, Fallah et al. [7] establish the convergence of one-step MAML for non-convex loss functions and then proposes a Hessian-free MAML to reduce the computational cost with theoretical guarantees. The convergence for multi-step MAML is studied in [13]. Wang et al. [38] further characterize the gap between the stationary point and the global optimum for MAML in a general non-convex setting.

**Federated Meta-Learning.** Very recently, the integration of federated learning and MAML has garnered much attention. Chen et al. [5] propose a federated meta-learning framework called FedMeta based on FedAvg [24] and MAML-type algorithms, which improves the performance and convergence speed of FedAvg. Jiang et al. [15] analyze the connections between FedAvg and MAML, and proposes a federated meta-learning algorithm called personalized FedAvg. Lin et al. [21] analyze the convergence properties and computational complexity of federated meta-learning for a strongly convex setting. Another recent work [8] proposes a federated meta-learning algorithm called Per-FedAvg and provides the convergence guarantee for the general non-convex setting. However, these studies focus on collaborative learning on a given set of tasks without exploring the valuable knowledge transfer from the previous tasks [27].

**ADMM.** A number of existing works [12, 23, 36, 39] analyze the convergence of ADMM for the case where the solution to each subproblem is computed exactly. Wang et al. [35] extend the ADMM method from two-block to multi-block form. Besides, there are also a few works [1, 14, 20, 25] studying the performance of ADMM in an inexact and non-convex setting, by linearizing the subproblems that are difficult to solve exactly. It is worth noting that linear approximation is insufficient for the meta-learning problem which generally requires higher-order information.

## 3 CONTINUAL EDGE LEARNING VIA FEDERATED META-LEARNING WITH REGULARIZATION

We consider a platform-aided federated meta-learning architecture for edge learning (as illustrated in Figure 1), where a set $I$ of source edge nodes joint force to learn a meta-model, aided by the valuable knowledge learned from previous tasks in the cloud. Specifically, the knowledge transfer is in the form of regularization using Bregman divergence on the prior model.

### 3.1 Problem Formulation

For ease of exposition, we consider a general supervised learning setting where each edge node $i \in I \cup \{m\}$ has a labeled dataset $D_i = \{(x_{ij}^i, y_{ij}^i)\}_{j=1}^{D_i}$ with total $D_i$ samples. Here $(x_{ij}^i, y_{ij}^i) \in \mathcal{X}_i \times \mathcal{Y}_i$ is a sample point with input $x_{ij}^i$ and true label $y_{ij}^i$, and follows an unknown underlying distribution $P_i$. For a model parameter $\phi_i \in \mathbb{R}^n$, the empirical loss function for a dataset $D_i$ is defined as

$$L_i(\phi_i, D_i) = \frac{1}{D_i} \sum_{j=1}^{D_i} l_i(\phi_i(x_{ij}^i, y_{ij}^i)), \text{ where } l_i(\cdot, \cdot) \text{ is a general differentiable non-convex loss function.}$$

Motivated by the recent success of regularization approaches in transfer learning and continual learning [27], we use regularization for extracting the valuable knowledge from the prior model to facilitate fast edge training and to alleviate catastrophic forgetting. More specially, for a model parameter $\theta \in \mathbb{R}^n$, we denote $\theta_p \in \mathbb{R}^n$ as the prior model parameter, and use the Bregman divergence $D_h(\theta, \theta_p)$ [4] as the regularization, given by:

$$D_h(\theta, \theta_p) = h(\theta) - h(\theta_p) - \langle \nabla h(\theta_p), \theta - \theta_p \rangle,$$

(1)

for some continuously-differentiable strictly convex function $h(\cdot)$. It is worth noting that Bregman divergence is a dissimilarity measure between two objects (e.g., vectors, matrices, distributions, etc.). It encompasses a rich class of divergence metrics, including squared Euclidean distance, squared Mahalanobis distance, Kullback-Leibler (KL) divergence, and Itakura-Saito (IS) distance, which are widely used in machine learning to encode the dissimilarity from different perspectives, and is particularly useful for the regularization approaches [16, 32, 42]. Note that while for ease of exposition, in this paper we consider the regularizer on the model parameters using Bregman divergence, the same methodology can be applied to generalize the regularization to be in terms of Bregman divergence between two functions of $\theta$ and $\theta_p$, respectively (see Assumption 2).

Following the same line as in MAML [9], we divide the dataset $D_i$ for the source edge node $i \in I$ into two disjoint sets, i.e., the support set $D^s_i$ and the query set $D^q_i$. Based on the prior model, we can formulate the federated meta-learning with knowledge transfer among the source edge nodes as the following regularized optimization problem:

$$\min_{\theta} \sum_{i \in I} w_i L_i(\phi_i(\theta), D^q_i) + \lambda D_h(\theta, \theta_p), \quad \text{s.t.} \quad \phi_i(\theta) = \theta - \alpha \nabla L_i(\theta, D^q_i), \quad i \in I$$

(2)

(3)

where $w_i \leq D_i/\sum_{i \in I} D_i$, $\alpha$ is the learning rate, and $\lambda$ is a penalty parameter that can be used to balance the trade-off between the loss and the regularization. In this formulation, we aim to find a good meta-model such that slight updating, i.e., one-step gradient descent, results in substantial performance improvement for any task across the edge nodes. That is, the source edge nodes collaboratively learn how to learn fast with a few data samples.
Further, by penalizing changes in the model via regularization, the learned model from (2)-(3) is confined to stay ‘close’ to the prior model for enabling collaborative edge learning without forgetting prior knowledge, thus the learned meta-model can widely adapt to different types of tasks.

In the fast adaptation stage, the platform transfers the learned meta-model \( \theta \) to the target node (denoted by \( m \)) after solving the regularized federated meta-learning problem (2)-(3). Based on \( \theta \), the target node \( m \) can use its local data set \( D_m^n \) to quickly compute a new model \( \phi_m \) by performing one-step stochastic gradient descent, i.e.,

\[
\phi_m = \theta - \alpha \nabla L_m(\theta, D_m^n).
\]

(4)

Note that the target node also can execute a few steps of stochastic gradient descent updates for better performance when needed.

### 3.2 An Inexact-ADMM Based Algorithm for Regularized Federated Meta-Learning

As alluded to earlier, general gradient-based federated meta-learning approaches cannot handle the regularized optimization problem (2)-(3) well. To address this problem, we propose an inexact-ADMM based federated meta-learning algorithm (ADMM-FedMeta) to solve (2)-(3).

Observe that the federated meta-learning problem (2)-(3) is equivalent to the following constrained optimization problem:

\[
\begin{align*}
\min_{\{\theta_i, \theta\}} & \sum_{i \in I} w_i L_i(\phi_i(\theta_i), D_i^T) + \lambda D_h(\theta, \theta_p) \\
\text{s.t.} & \quad \theta_i - \theta = 0, \ i \in I
\end{align*}
\]

where \( \phi_i(\theta_i) \) is given by (3). To solve (5), we form the augmented Lagrangian function as follows:

\[
L(\{\theta_i, y_i\}, \theta) = \sum_{i \in I} \left( w_i L_i(\phi_i(\theta_i), D_i^T) + \langle y_i, \theta_i - \theta \rangle + \frac{\rho_i}{2} \|\theta_i - \theta\|^2 \right) + \lambda D_h(\theta, \theta_p),
\]

\[
\tag{6}
\text{s.t.} \quad \theta_i - \theta = 0, \ i \in I
\]

where \( y_i \in \mathbb{R}^n \) is a dual variable and \( \rho_i > 0 \) is a penalty parameter for each \( i \in I \).

When the classical ADMM method is applied [3], the variables \( \theta_i, \theta \) and \( y_i \) are updated alternatively in solving (5) as follows:

\[
\begin{align*}
\theta_i^{t+1} &= \arg \min_{\theta_i} L_i(\theta_i, y_i^t, \theta_{-i}^{t+1}), \\
y_i^{t+1} &= y_i^t + \rho_i (\theta_i^{t+1} - \theta_i^{t+1}), \\
\theta_{-i}^{t+1} &= \arg \min_{\theta_{-i}} L_i(\theta_{-i}, y_i^t, \theta_{-i}^{t+1}),
\end{align*}
\]

\[
\tag{7}
\text{s.t.} \quad \theta_i - \theta = 0, \ i \in I
\]

where \( L_i(\theta_i, y_i, \theta_{-i}) \) is a subproblem of the combined computation power of the local edge nodes and the platform, we provide the following alternating updating strategy: 1) updating \( \theta_i \) at the platform and 2) updating \( \{\theta_i, y_i\} \) at the source edge nodes in a distributed manner. Particularly, in this way, the computation corresponding to the regularizer can be decoupled from the edge nodes to the platform. However, attaining the exact solution to each subproblem is computationally costly, especially with a complex DNN model. To tackle this challenge, we devise the inexact-ADMM based federated meta-learning (ADMM-FedMeta) below.

Specifically, in communication round \( t = 0 \), the platform initializes \( \theta^0 \) and sends it to all edge nodes. Each node \( i \in I \) initializes \( y_i^{-1} \) locally.

- **Local update of \( \{\theta_i, y_i\} \).** After receiving \( \theta^t \) from the platform at communication round \( t \), each edge node \( i \in I \) would do the following updates:
  1. **Update node-specific model \( \phi_i \).** Based on the dataset \( D_i^t \), \( \phi_i^t \) is updated as:

\[
\phi_i^t = \theta^t - \alpha \nabla L_i(\theta^t, D_i^T).
\]

\[
\tag{8}
\]

2. **Update local parameter \( \theta_i \).** Based on (7), given the meta-model \( \theta^t \) and local dual variable \( y_i^{-1} \) from last communication round, the local parameter \( \theta_i \) should be updated as:

\[
\theta_i^t = \arg \min_{\theta_i} \left\{ w_i L_i(\phi_i(\theta_i), D_i^T) + \langle y_i^{-1}, \theta_i - \theta^t \rangle + \frac{\rho_i}{2} \|\theta_i - \theta^t\|^2 \right\}.
\]

\[
\tag{9}
\]

To simplify the computation, we use linear approximation (i.e., first-order Taylor expansion) around \( \theta^t \) to relax this subproblem, i.e.,

\[
\theta_i^t = \arg \min_{\theta_i} \left\{ w_i L_i(\phi_i^t(\theta_i), D_i^T) + \langle y_i^{-1}, \theta_i - \theta^t \rangle + \frac{\rho_i}{2} \|\theta_i - \theta^t\|^2 \right\},
\]

\[
\tag{10}
\]

where \( \phi_i^t \) is from (8). Nevertheless, (10) is still insufficient since the computational complexity of the Hessian-gradient product \( \nabla^2 L_i(\theta^t, D_i^T) \nabla L_i(\phi_i^t, D_i^T) \) is \( O(n^2) \). To further reduce the computational cost, as in [7, 13], we replace the Hessian-gradient product by a first-order estimator, i.e.,

\[
y_i^t = \nabla L_i(\phi_i^t, D_i^T),
\]

\[
\tag{11}
\text{where} \quad r_i^t \approx \nabla L_i(\theta_i^t, D_i^T) \quad \text{and} \quad \delta_{i,t} > 0 \quad \text{is the degree of freedom capturing the estimation accuracy. In a nutshell, the local parameter} \quad \theta_i^t \quad \text{is updated as follows:}
\]

\[
\theta_i^t = \theta^t - \frac{y_i^{-1} + w_i(\nabla L_i(\phi_i^t, D_i^T) - \alpha g_i^t)}{\rho_i},
\]

\[
\tag{12}
\]

where (12) is derived by the optimality of (10) after replacing \( \nabla^2 L_i(\theta^t, D_i^T) \nabla L_i(\phi_i^t, D_i^T) \) with \( g_i^t \).

3. **Update local dual variable \( y_i \).** Based on \( \theta^t \) and the updated local parameter \( \theta_i^t \), the auxiliary dual variable \( y_i^t \) is next updated according to:

\[
y_i^t = y_i^{-1} + \rho_i (\theta_i^t - \theta^t).
\]

\[
\tag{13}
\]

- **Global Aggregation towards Meta-Model \( \theta \).** Each edge node \( i \in I \) sends the updated local parameters \( \theta_i^t \) and \( y_i^t \) to the platform. With the prior model \( \theta_p \) transferred from the
Algorithm 1: Inexact-ADMM Based Meta-Learning Algorithm (ADMM-FedMeta)

| Input: $\theta_p$, $\alpha$, $\lambda$, $\rho_i$, $D_i^s$, $D_i^f$ for $i \in I$ |
| Output: Final meta-model $\theta$ |

1. Each edge node $i \in I$ initializes $y_i^{-1}$.
2. Platform initializes $\theta_0$ and sends it to all edge nodes.
3. For $t = 0$ to $T$ do
   4. For $i = 1$ to $I$ do
      5. Compute $\phi_i^t = \theta^t - \alpha \nabla L_i(\theta^t, D_i^s)$;
      6. Compute $\theta_0^t$ by (12);
      7. Compute $y_i^{t+1} = y_i^{t-1} + \rho_i (\theta_0^t - \theta^t)$;
      8. Send $\phi_i^t$ and $y_i^{t+1}$ back to the platform;
   end
4. Platform updates $\theta^{t+1}$ by (14) and sends it to all edge nodes $i \in I$;
5. Platform transfers $\theta^T$ to target node for fast adaptation;

The details of ADMM-FedMeta are summarized in Algorithm 1. Note that due to linearizing all decomposed subproblems and estimating them by its first-order estimation, we enable the computation complexity of ADMM-FedMeta to be $O(n)$ per round, which maintains the lowest among all existing federated meta-learning approaches.

4 PERFORMANCE ANALYSIS

In this section, we analyze the performance of ADMM-FedMeta. First, we study the convergence properties and characterize the communication complexity for ADMM-FedMeta. Then, we quantify the forgetting effect to previous tasks of the meta-model and analyze the fast adaptation performance at the target edge node.

4.1 Convergence Analysis

For convenience, we denote the objective function of (2) as $F(\theta)$:

$$ F(\theta) \equiv \sum_{i \in I} \omega_i L_i(\phi_i(\theta), D_i^f) + \lambda D_h(\theta, \theta_p), $$

where $\phi_i(\theta) = \theta - \alpha \nabla L_i(\theta, D_i^s)$. Next, we characterize the convergence and communication complexity of the proposed algorithm for finding a first-order stationary point of function $F(\theta)$. Formally, the definition of an $\epsilon$-approximate first-order stationary point is given as follows.

**Definition 1** ($\epsilon$-FOSP). A solution $\theta \in \mathbb{R}^n$ is called an $\epsilon$-approximate first-order stationary point ($\epsilon$-FOSP) of (2), if $\|\nabla F(\theta)\| \leq \epsilon$, for $\epsilon > 0$.

The above definition implies that if a solution $\theta$ obtained by an algorithm is a $\epsilon$-FOSP, then the gradient norm of the objective function is bounded above by $\epsilon$.

Note that the first-order estimator of Hessian introduced in the subproblem (12) inevitably complicates the convergence analysis of ADMM-FedMeta, making the existing analysis methods of ADMM [1] not suitable here. To establish the convergence of ADMM-FedMeta, we impose the following standard assumptions in the literature.

**Assumption 1.** $F(\theta)$ is lower-bounded, i.e., $F(\theta) > -\infty$, for all $\theta \in \mathbb{R}^n$.

**Assumption 2** (Smoothness and Bounded Gradient). For each $i \in I \cup \{m\}$, any $D_i^s$, and $\theta_p \in \mathbb{R}^m$, both $L_i(\cdot, D_i^s)$ and $D_h(\cdot, \theta_p)$ are twice continuously differentiable and smooth, i.e., for any $x, y \in \mathbb{R}^n$, there exist constants $\mu_i > 0$ and $\mu_p > 0$ such that

$$ \|\nabla L_i(x, D_i^s) - \nabla L_i(y, D_i^s)\| \leq \mu_i \|x - y\|, $$

$$ \|\nabla_x D_h(x, \theta_p) - \nabla_y D_h(y, \theta_p)\| \leq \mu_p \|x - y\|. $$

Besides, the gradient norms of $L_i(\cdot, D_i^s)$ is bounded by a positive constant $\beta_i > 0$, i.e., for any $x \in \mathbb{R}^n$, the following holds:

$$ \|\nabla L_i(x, D_i^s)\| \leq \beta_i. $$

**Assumption 3** (Lipschitz Continuous Hessian). For any $i \in I$ and any $D_i^s$, the Hessian of $L_i(\cdot, D_i^s)$ is $\zeta_i$-Lipschitz continuous, i.e., for any $x, y \in \mathbb{R}^n$, we have:

$$ \|\nabla^2 L_i(x, D_i^s) - \nabla^2 L_i(y, D_i^s)\| \leq \zeta_i \|x - y\|. $$

**Assumption 4** (Bounded Variance). For any $i \in I \cup \{m\}$ and $\theta \in \mathbb{R}^n$, the stochastic gradient $\nabla_l(\theta, (x, y))$ and $\nabla^2 l(\theta, (x, y))$ with respect to data point $(x, y) \in X_i \times Y_i$ have bounded variances, i.e.,

$$ \mathbb{E}_{(x,y)\sim P} \|\nabla l_i(\theta, (x, y)) - \nabla L_i(\theta)\|^2 \leq \sigma_l^2, $$

$$ \mathbb{E}_{(x,y)\sim P} \|\nabla^2 l_i(\theta, (x, y)) - \nabla^2 L_i(\theta)\|^2 \leq \sigma_h^2, $$

for some positive constants $\sigma_l^2 > 0$ and $\sigma_h^2 > 0$.

Assumptions 1-4 are standard in the state-of-the-art studies on the analysis of federated learning algorithms [8, 21, 45]. In particular, (18) is critical for analyzing the convergence as it enables characterizing the estimation error of the Hessian. Assumption 3 implies the high-order smoothness of $L_i(\cdot, D_i^s)$ for dealing with the second-order information in the update steps of Algorithm 1. Furthermore, Assumption 4 provides the upper bounds of the variances of the gradient and Hessian estimations.

To quantify the convergence behavior of ADMM-FedMeta, we first study the properties of the objective function $F(\theta)$. Denote $F_l(\theta) \triangleq L_i(\phi_i(\theta), D_i^f)$. Based on Assumptions 2 and 3, we have the following result about the smoothness of $F_l$ as in the standard analysis of federated meta-learning approaches.
Lemma 1. Given Assumptions 2 and 3, for each $i \in I \cup \{m\}$, $F_i$ is proper and $v_i$-smooth, i.e.,
\[ \|\nabla F_i(x) - \nabla F_i(y)\| \leq v_i \|x - y\|, \forall x, y \in \mathbb{R}^n, \]
where $v_i$ is defined as follows:
\[ v_i \doteq (1 + \alpha \mu_i)(1 + \rho_i)\mu_i + \alpha \rho_i \gamma_i. \quad (23) \]
Proof. The proof is standard. The detailed proof is provided in Appendix A of the technical report [43].

Next, we impose the assumptions on the hyper-parameters.

Assumption 5. For all $i \in I$, $\rho_i$ is large enough such that:
\[ \frac{\rho_i}{2} - 4w_i v_i > 0, \quad (24) \]
\[ \frac{\rho_i}{2} - 2w_i^2 v_i^2 \left(4w_i v_i + \frac{1}{\rho_i}\right) - \frac{\lambda \rho_i}{2} > 0, \quad (25) \]
\[ \rho_i - 3v_i > 0, \quad (26) \]
where $v_i$ is a smooth scalar defined in (23). Besides, for all $i \in I$, the additional degree of freedom parameter $\delta_{ij}$ for the approximation of Hessian-gradient products is chosen to be a monotonically non-increasing positive sequence and satisfies $\sum_{i=1}^{\infty} \delta_{ij} < \infty$.

We impose Assumption 5 on the penalty parameter $\rho_i$ and the degree of freedom parameter $\delta_{ij}$. Intuitively, (24)-(26) imply that a large $\rho_i$ is required to balance the error caused by the linear approximation and Hessian estimation in (12).

Based on Lemma 1, we are ready to establish the convergence and characterize the communication complexity for Algorithm 1.

Theorem 1 (Convergence and Communication Complexity). Under Assumptions 1-5, we have the following results based on Algorithm 1:

(i) $\{\theta^t\}$ has at least one limit point and each limit point $\theta^*$ is a stationary solution of (2), i.e., $\|\nabla F(\theta^*)\| = 0$.

(ii) Algorithm 1 finds an $\epsilon$-FOSP of Problem (2) (3) after at most $O(1/\epsilon^2)$ communication rounds.

Proof. For part (i), we first characterize the successive difference of the augmented Lagrangian function. Based on that, we next show $\lim_{t \to \infty} \|\theta_t^{*+1} - \theta_t^*\| = 0$ and $\lim_{t \to \infty} \|\theta_t^{*+1} - \theta_t^*\| = 0$. Lastly, we bound $\|\nabla F(\theta^*)\|$ via $\|\theta_t^{*+1} - \theta_t^*\|$ and $\|\theta_t^{*+1} - \theta_t^*\|$. We prove (ii) by dividing the sum of the Lagrangian successive difference into two finite parts. The detailed proof is presented in Appendix F of the technical report [43].

Corollary 1. Given Assumptions 1-5, the $\epsilon$-FOSP solution $\theta_\epsilon$ found by Algorithm 1 satisfies that:
\[ E\left\{ \sum_{i \in I} w_i L_i(\theta_\epsilon - \alpha \nabla L_i(\theta_\epsilon)) + \lambda D_k(\theta_\epsilon, \theta_p) \right\} \]
\[ \leq \epsilon + \sum_{i \in I} w_i w_i^g \left( \frac{\alpha \mu_i}{\sqrt{D_{i}^g}} + \frac{1}{\sqrt{D_{i}^q}} \right), \quad (27) \]
where $w_i = 1/I$ and $L_i(\cdot)$ is the expected loss denoted by:
\[ L_i(\theta) \doteq E_{(x,y) \sim P_i} [l_i(\theta, (x,y))]. \quad (28) \]
Proof. The detailed proof is presented in Appendix G of the technical report [43].

Corollary 1 implies that despite $\epsilon$-FOSP of the deterministic loss (2) can be obtained within $O(1/\epsilon^2)$ rounds, performance degradation may happen due to small sample size.

4.2 Performance of Rapid Adaptation at Target Node

While the task similarity assumption is not required to guarantee the convergence of Algorithm 1, we impose such an assumption to study the fast adaptation performance at the target node $m$.

Assumption 6 (Task Similarity). There exist positive constants $\psi^g > 0$ and $\psi^h > 0$ such that for any $i \in I$ and $\theta \in \mathbb{R}^n$, the following holds:
\[ \|\nabla L_m(\theta) - \nabla L_i(\theta)\| \leq \psi^g, \quad (29) \]
\[ \|\nabla^2 L_m(\theta) - \nabla^2 L_i(\theta)\| \leq \psi^h, \quad (30) \]
where $L_i(\cdot)$ is defined in (28), and the same applies to $L_m(\cdot)$.

Assumption 6 indicates that the variations of the gradients between the loss of source edge nodes and the target edge node are bounded by some constants, which capture the similarity of the tasks corresponding to non-IID data and holds for many practical loss functions [45], such as logistic regression and hyperbolic tangent functions. In particular, $\psi^g$ and $\psi^h$ can be roughly seen as a distance between data distributions $P_m$ and $P_i$ [7].

Next, we present the following result about the performance of rapid adaptation.

Theorem 2 (Fast Adaptation Performance). Suppose that Assumptions 1-6 hold. For any $\epsilon > 0$, the $\epsilon$-FOSP solution $\theta_\epsilon$ obtained by Algorithm 1 satisfies that:
\[ E\{\|\nabla F_m(\theta_\epsilon)\|\} \leq \epsilon + \alpha \beta m \sum_{i \in I} w_i \psi^g + (\alpha \mu + 1)^2 \sum_{i \in I} w_i \psi^g \]
\[ + \alpha \mu (\alpha \mu + 1) \sum_{i \in I} w_i \psi^g \left( \frac{1}{\sqrt{D_{i}^g}} + \frac{1}{\sqrt{D_{i}^q}} \right) \]
\[ + (\alpha \mu + 1) \sum_{i \in I} w_i \psi^h \left( \frac{1}{\sqrt{D_{i}^g}} + \frac{1}{\sqrt{D_{i}^q}} \right) \]
\[ + \alpha \beta \psi^h \sum_{i \in I} w_i \psi^h \left( \frac{1}{\sqrt{D_{i}^g}} + \frac{1}{\sqrt{D_{i}^q}} \right). \quad (31) \]
where $P_m(\theta) \triangleq \mathbb{E}_m(\theta - \alpha \nabla L_m(\theta, D_m^m), D_m^m) + \lambda D_h(\theta, \emptyset_p)$ for any $D_m$ and $D_m^m$, with respect to distribution $P_m$, and $\mu = \max_{i \in I} \{\mu_i\}$.

Proof. The detailed proof is presented in Appendix H of the technical report [43].

Theorem 2 sheds light on the performance of fast adaptation with the previous knowledge, which depends on the size of datasets, the variance of stochastic gradient and Hessian, and the similarity between the target node and source nodes. In particular, if $D_i^s = O(e^{-2})$ and $D_i^q = O(e^{-2})$ for $i \in \mathcal{I} \cup \{m\}$, then an $O(e + \sum_{i \in \mathcal{I}} w_i(\psi_i^h + \psi_i^q))$-FOSP can be obtained at the target node. However, it is clear that the larger the dataset of source nodes dissimilar to the target node is, the worse the rapid adaptation performs. In the next subsection, we will show these issues can be alleviated via regularization with a good prior model.

4.3 Forgetting effect on Prior Knowledge

In this section, we quantify the forgetting effect of the previous task of Algorithm 1 in a special case, where the regularizer is squared Euclidean distance, i.e., $D_h(\theta, \emptyset_p) = \|\theta - \emptyset_p\|^2$. To do so, we first derive an upper bound of $D_h(\theta, \emptyset_p)$ via the following lemma.

**Lemma 2.** Given Assumptions 1-5, for any $e > 0$, the e-FOSP solution $\theta_e$ obtained by Algorithm 1 satisfies that:

$$\mathbb{E}\{D_h(\theta_e, \emptyset_p)\} \leq \frac{1}{\lambda} \left( e + \sum_{i \in \mathcal{I}} w_i \left( \beta_i + \frac{\alpha \mu_i \sigma_i^q}{\sqrt{D_i^s}} + \frac{\sigma_i^q}{\sqrt{D_i^q}} \right) \right) \|\theta_e - \emptyset_p\|. \quad (32)$$

Particularly, suppose $D_h(\theta, \emptyset_p)$ is strongly convex with respect to $\theta$, i.e., there exists $M > 0$ such that for $x, y \in \mathbb{R}^n$, the following holds:

$$\langle \nabla D_h(x, \emptyset_p) - \nabla D_h(y, \emptyset_p), x - y \rangle \geq M \|x - y\|^2. \quad (33)$$

Then (32) can be written as:

$$\mathbb{E}\{D_h(\theta_e, \emptyset_p)\} \leq \frac{1}{M\lambda^2} \left( e + \sum_{i \in \mathcal{I}} w_i \left( \beta_i + \frac{\alpha \mu_i \sigma_i^q}{\sqrt{D_i^s}} + \frac{\sigma_i^q}{\sqrt{D_i^q}} \right) \right)^2. \quad (34)$$

Proof. The detailed proof is presented in Appendix I of the technical report [43].

For a current model parameter $\theta$, we define the forgetting cost of $\theta$ on the previous task $\rho$ as $L_{\rho}(\theta)$ [18], where $L_{\rho}(\cdot)$ is the expected loss over the data distribution of task $\rho$ (defined by (28)). Based on Lemma 2, we next characterize the forgetting cost of the e-FOSP solution.

**Theorem 3.** Suppose that $L_{\rho}(\cdot)$ is $\mu_{\rho}$-smooth, $\|\nabla L_{\rho}(\theta_p)\| \leq \epsilon_p$ for some $\epsilon_p > 0$, and $D_h(\theta, \emptyset_p) = \|\theta - \emptyset_p\|^2$. Under Assumptions 1-5, for the e-FOSP solution $\theta_e$, we have the following result:

$$\|\nabla L_{\rho}(\theta_e)\| \leq \epsilon_p + \frac{V_p}{2\lambda} \left( e + \sum_{i \in \mathcal{I}} w_i \left( \beta_i + \frac{\alpha \mu_i \sigma_i^q}{\sqrt{D_i^s}} + \frac{\sigma_i^q}{\sqrt{D_i^q}} \right) \right) \quad (35)$$

Proof. The result can be directly obtained by Lemma 2. □

Based on Theorem 3 and corollary 1, it is clear that by selecting a suitable $\lambda$, the regularizer enables the meta-model to learn on the current task while maintaining good performance on the previous task. On the other hand, combined with Theorem 2, Theorem 3 also implies that due to the independence of (35) and similarity conditions, a good prior reference model (e.g., with a relatively small $\|\nabla L_{\rho}(\emptyset_p)\|$) can effectively alleviate the significant performance degradation caused by the dissimilarity between the source nodes and the target nodes.

5 EXPERIMENTAL RESULTS

In this section, we evaluate the experimental performance of ADMM-FedMeta on different datasets and models. In particular, our experimental studies are designed to evaluate the performance of the proposed ADMM-FedMeta algorithm in challenging edge learning settings where edge nodes have limited data samples. Specifically, we assume that each source node has only tens of data samples during the training stage and that in the testing phase, each target node has only 10-20 data samples. Clearly, edge learning in these settings is highly nontrivial, particularly for sophisticated datasets (e.g., CIFAR-100).

**Datasets and models.** We evaluate the performance of ADMM-FedMeta on three widely-used benchmarks, including Fashion-MNIST [41], CIFAR-10 [19], and CIFAR-100 [19]. Specifically, the data is distributed among $I$ edge nodes as follows: 1) Each node has samples from only two random classes [21]; 2) the number of samples per node follows a discrete uniform distribution, i.e., $D_i \sim U(a, b)$ for $i \in \mathcal{I}$. Here we set $a = 20$, $b = 40$, $I = 50$ for Fashion-MNIST and CIFAR-10, and $I = 100$ for CIFAR-100. We randomly select $80\%$ and $20\%$ nodes as the source nodes and the target nodes, respectively. For each node, we divide the local dataset into a support set and a query set (i.e., $D_i^s$ and $D_i^q$), each with $50\%$ of the local data. We set the meta-step size as $\alpha = 0.01$, the penalty parameters $\rho = 0.3$ for Fashion-MNIST, and $\rho = 0.7$ for CIFAR-10 and CIFAR-100, where $\rho_i = \rho$ for all $i \in \mathcal{I}$. We set the regularizer as squared $\ell_2$-norm, and the degree of freedom parameter $\delta_i = 1/(10^t + 100)$ with $t = 1, 2, \ldots, 100$ for $i \in \mathcal{I}$. For Fashion-MNIST, we use a convolutional neural network (CNN) with max-pooling operation and Exponential Linear Unit (ELU) activation function, which contains two convolutional layers with sizes 32 and 64 followed by a fully connected layer and softmax. The strides are set as 1 for convolution operation and 2 for pooling operation. For CIFAR-10 and CIFAT-100, we use a CNN containing three convolutional layers with sizes 32, 64, and 128, and a CNN containing four convolutional layers with sizes 32, 64, 128, and 256, respectively, while keeping all the other setups the same as that in Fashion-MNIST.

**Implementation.** We implement the code in TensorFlow Version 1.14 on a server with two Intel® Xeon® Golden 5120 CPUs and one Nvidia® Tesla-V100 32G GPU. Please refer to https://github.com/XinJiang1994/HFMaml for full details.

**Baselines.** We consider two existing baseline algorithms, i.e., FedAvg [24] and Per-FedAvg [8] with one or multiple local update steps. For the sake of fair comparison, we test different hyperparameters of Per-FedAvg from $\{0.001, 0.005, 0.01, 0.05, 0.1\}$ (i.e., $\beta$
We have the following observations. (1) ADMM-FedMeta substantially outperforms Per-FedAvg and FedAvg on CIFAR-10 and CIFAR-100, and use squared Euclidean distance as the regularizer. As illustrated in Table 2, the existing methods for Fashion-MNIST, 5% of AdMM-FedMeta and the two base-lines on CIFAR-10 are smaller when the number of local updates increases. The underlying rationale is that with more local update steps, the number of the overall iterations in the two baseline algorithms actually increases, thus resulting in a better model. However, a large number of local update steps would lead to high computational cost and may cause failure to convergence [8, Theorem 4.5].

Convergence speed and hyper-parameter. As shown in Figure 2, ADMM-FedMeta converges significantly faster than the existing methods, often requiring tens of rounds to obtain a high-quality meta-model, which indicates ADMM-FedMeta can achieve a great communication efficiency. Besides, Figure 2 also suggests that despite the sample size is small, edge nodes can obtain a satisfactory model via federated meta-learning with only one-step stochastic gradient descent. Further, we investigate the impact of the hyper-parameter $\rho$ on the convergence of ADMM-FedMeta (we let $\rho_i$ be the same across different nodes). It can be seen from Figure 3 that ADMM-FedMeta has a relatively faster convergence speed with a smaller $\rho$ in terms of the training loss. In particular, a small change of $\rho$ does not greatly affect the convergence properties of the algorithm, which implies that ADMM-FedMeta is robust to the hyper-parameters.

Forgetting effect to the previous task. To demonstrate the forgetting effect to the prior task on different algorithms, we pre-train a model $\theta_p$ with satisfactory performance on the data of the first five classes as a prior task, and then use $\theta_p$ as the initialization to train the meta-models by different algorithms on the data of the last five classes as a new task. Similarly, over CIFAR-100, we use the first fifty classes and the last fifty classes as the prior and the last five classes as a new task. Similarly, over CIFAR-100, we use the prior model via federated meta-learning with only one-step stochastic gradient descent. Further, we investigate the impact of the hyper-parameter $\rho$ on the convergence of ADMM-FedMeta (we let $\rho_i$ be the same across different nodes). It can be seen from Figure 3 that ADMM-FedMeta has a relatively faster convergence speed with a smaller $\rho$ in terms of the training loss. In particular, a small change of $\rho$ does not greatly affect the convergence properties of the algorithm, which implies that ADMM-FedMeta is robust to the hyper-parameters.

Performance and computational efficiency. To be fair, we set $\lambda = 0$ to remove the benefit of using the regularization in ADMM-FedMeta. We repeat the experiments 10 times, then show the comparison of the accuracy along with 95% confidence intervals in Table 1. We have the following observations. (1) ADMM-FedMeta substantially outperforms Per-FedAvg and FedAvg, especially on sophisticated datasets. Specifically, ADMM-FedMeta achieves 7.7% over FedAvg and 5.2% over Per-FedAvg on Fashion-MNIST, 31.87% over FedAvg and 10.65% over Per-FedAvg on CIFAR-10, and 27.12% over FedAvg and 28.25% over Per-FedAvg on CIFAR-100. Note that the computation costs of each local update are $O(n), O(n^2)$, and $O(n)$ for FedAvg, Per-FedAvg, and ADMM-FedMeta, respectively. This performance improvement clearly indicates that ADMM-FedMeta is more computationally efficient on non-convex loss and heterogeneous datasets with a small number of data samples. (2) It should be noted that the gaps between ADMM-FedMeta and the two baselines on CIFAR-10 are smaller when the number of local updates increases. The underlying rationale is that with more local update steps, the number of the overall iterations in the two baseline algorithms actually increases, thus resulting in a better model. However, in [8, Algorithm 1]), and select the best for the experiments, i.e., 0.005 for Fashion-MNIST and CIFAR-10, and 0.001 for CIFAR-100.

To demonstrate the impact of the knowledge transfer and the inexact-ADMM based methods respectively, we first remove the regularization term (i.e., letting $\lambda = 0$) and compare the convergence speed and adaptation performance between ADMM-FedMeta and the baselines. Then, we conduct the experiment using a prior model for regularization and show the performance improvement in terms of convergence, adaptation, and forgetting effect.

| Dataset | # local updates | FedAvg | Per-FedAvg | ADMM-FedMeta |
|---------|-----------------|--------|------------|--------------|
| Fashion-MNIST | 1 | 83.99% ± 3.90% | 87.55% ± 2.42% | 95.69% ± 0.37% |
|          | 5 | 88.86% ± 1.57% | 89.65% ± 3.26% | N/A          |
|          | 10 | 85.29% ± 1.93% | 90.95% ± 2.71% | N/A          |
| CIFAR-10 | 1 | 41.97% ± 1.33% | 60.53% ± 1.12% | 74.61% ± 2.19% |
|          | 5 | 56.58% ± 2.27% | 65.93% ± 9.97% | N/A          |
|          | 10 | 56.58% ± 1.15% | 67.43% ± 0.99% | N/A          |
| CIFAR-100 | 1 | 42.35% ± 1.55% | 48.19% ± 2.18% | 63.56% ± 0.87% |
|          | 5 | 50.00% ± 1.09% | 49.56% ± 1.09% | N/A          |
|          | 10 | 49.97% ± 1.04% | 48.73% ± 1.23% | N/A          |

Table 1: Comparison of the accuracy on target nodes of different algorithms.
Table 2: Comparison of the accuracy on prior and current tasks.

| Dataset | Task | Prior Model | FedAvg | Per-FedAvg | ADMM-FedMeta |
|---------|------|-------------|--------|------------|--------------|
| Fashion-MNIST | Prior | 95.63% | 41.27% | 49.60% | 92.86% |
| | New | 49.21% | 94.05% | 94.84% | 94.04% |
| CIFAR-10 | Prior | 75.74% | 41.08% | 42.08% | 62.38% |
| | New | 17.33% | 55.45% | 47.03% | 71.29% |
| CIFAR-100 | Prior | 66.27% | 35.32% | 37.62% | 59.52% |
| | New | 45.63% | 40.48% | 57.92% | 63.10% |

Figure 4: Impact of \( \theta_p \) and \( \lambda \).

6 CONCLUSION

In this paper, we presented an inexact-ADMM based federated meta-learning approach for fast and continual edge learning. More specifically, we first proposed a platform-aided federated meta-learning architecture enabling edge nodes to collaboratively learn a meta-model with the knowledge transfer of previous tasks. We cast the federated meta-learning problem as a regularized optimization problem, where the previous knowledge is extracted as regularization. Then, we devised an ADMM based algorithm, namely ADMM-FedMeta, in which the original problem is decomposed into many subproblems which can be solved in parallel across edge nodes and the platform. Further, we developed a variant of the inexact-ADMM method to reduce the computational cost per round to \( O(n) \) via employing linear approximation as well as Hessian estimation. We provided a comprehensive analysis and empirical results to demonstrate the effectiveness and efficiency of ADMM-FedMeta. The advantages of the proposed algorithm are summarized as follows: First, it can decouple the regularizer from edge nodes to the platform, which helps to alleviate the local computational cost while exploiting the resources between local devices and the server effectively. Besides, by inexact-ADMM technique, we further reduce the computational complexity during local update and global aggregation. We show that ADMM-FedMeta can converge under mild conditions, particularly, with weak task similarity assumptions. Lastly, empirical results show that ADMM-FedMeta converges faster than existing benchmark algorithms.

There are a number of interesting questions and directions for future work. First, it is of interest to incorporate the experience replay method and parameter isolation approaches into the ADMM-FedMeta to further mitigate the catastrophic forgetting. Secondly, despite ADMM-FedMeta can be directly applied to reinforcement learning with policy gradient, it may lead to poor sample efficiency. It remains largely open to develop efficient collaborative reinforcement learning for edge learning. Moreover, our experimental results indicate that even without the regularization term, in practice ADMM-FedMeta can still converge faster than the existing gradient-based methods, especially on small sample sizes. It is intriguing to get a more deep understanding of this phenomenon.

7 ACKNOWLEDGMENTS

This research was supported in part by NSF under Grants CNS-2003081 and CPS-1739344, National Key R&D Program of China.
under Grant No. 2019YFA0706403, National Natural Science Foundation of China under Grant Nos. 62072472, 61702562 and U19A2067, Natural Science Foundation of Hunan Province, China under Grant No. 2020JJ2050, 111 Project under Grant No. B18059, the Young Elite Scientists Sponsorship Program by CAST under Grant No. 2018QNRC001, the Young Talents Plan of Hunan Province of China under Grant No. 2019RS2001, and also financially supported by China Scholarship Council (CSC).

REFERENCES

[1] Bina Foygel Barber and Emily T Sidky. 2020. Convergence for nonconvex ADMM, with applications to CT imaging. arXiv preprint arXiv:2006.07278 (2020).
[2] Pablo Barros, German I Parisi, Di Fu, Xun Liu, and Stefan Wermter. 2018. Expectation learning for adaptive crossmodal stimuli association. arXiv preprint arXiv:1801.07654 (2018).
[3] Stephen Boyd, Neal Parikh, and Eric Chu. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Now Publishers Inc.
[4] Lev M Bregman. 1967. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR computational mathematics and mathematical physics 7, 3 (1967), 200–217.
[5] Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. 2018. Federated meta-learning with fast convergence and efficient communication. arXiv preprint arXiv:1802.07876 (2018).
[6] Liam Collins, Aryan Mokhtari, and Sanjay Shakkottai. 2020. Distribution-Agnostic Model-Agnostic Meta-Learning. arXiv preprint arXiv:2002.04766 (2020).
[7] Alireza Fallah, Aryan Mokhtari, and Asauman Ozdaglar. 2020. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In International Conference on Artificial Intelligence and Statistics. 1082–1092.
[8] Alireza Fallah, Aryan Mokhtari, and Asauman Ozdaglar. 2020. Personalized federated learning: A meta-learning approach. arXiv preprint arXiv:2002.07948 (2020).
[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. arXiv preprint arXiv:1703.03400 (2017).
[10] Chelsea Finn, Kelvin Xu, and Sergey Levine. 2018. Probabilistic model-agnostic meta-learning. In Advances in Neural Information Processing Systems. 9516–9527.
[11] Robert M French. 1999. Catastrophic forgetting in connectionist networks. In Connectionist models of cognition and perception. 128–135.
[12] Mengyi Hong, Zhi-Quan Luo, and Meisam Razaviyayn. 2016. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. SIAM Journal on Optimization 26, 1 (2016), 337–364.
[13] Kaiyi Ji, Junjie Yang, and Yingbin Liang. 2020. Multi-Step Model-Agnostic Meta-Learning: Convergence and Improved Algorithms. arXiv preprint arXiv:2002.07836 (2020).
[14] Bo Jiang, Tianyi Lin, Shiqian Ma, and Shuzhong Zhang. 2019. Structured nonconvex and non-smooth optimization: algorithms and iteration complexity analysis. Computational Optimization and Applications 72, 1 (2019), 115–157.
[15] Yihan Jiang, Jakub Kononen, Keith Rush, and Sreeram Kannan. 2019. Improving federated learning personalization via model agnostic meta learning. arXiv preprint arXiv:1905.12488 (2019).
[16] Heechul Jung, Jeongwooo Ju, Minju Jung, and Junmo Kim. 2016. Less-forgetting learning in deep neural networks. arXiv preprint arXiv:1607.00122 (2016).
[17] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quinn, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences 114, 13 (2017), 3521–3526.
[18] R Krishna and Prasanna Balaprakash. 2020. Meta Continual Learning via Dynamic Programming. arXiv preprint arXiv:2008.02219 (2020).
[19] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features with a convolutional neural network. Advances in neural information processing systems. 1081–1088.
[20] Alessandro Lanza, Serena Morigi, Ivan Selesnick, and Fiorella Sgallari. 2017. Nonconvex nonsmooth optimization via convex–nonconvex majorization–minimization. Numer. Math. 136, 2 (2017), 343–381.
[21] Sen Lin, Guang Yang, and Junshan Zhang. 2020. A Collaborative Learning Framework via Federated Meta-Learning. arXiv preprint arXiv:2003.03229 (2020).
[22] David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. In Advances in neural information processing systems. 6457–6467.
[23] Sandri Magninsson, Pradeep Chathuranga Weeraddana, Michael G Rabbat, and Carlo Fischione. 2015. On the convergence of alternating direction lagrangian methods for nonconvex structured optimization problems. IEEE Transactions on Control of Network Systems 3, 3 (2015), 296–309.
[24] Brendan McMahen, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In Artificial Intelligence and Statistics. 1273–1282.
[25] Madeh Chandra Mukkamala, Peter Ochs, Thomas Pock, and Shokam Sahsah. 2020. Convex-Concave Backtracking for Inertial Bregman Proximal Gradient Algorithms in Nonconvex Optimization. SIAM Journal on Mathematics of Data Science 2, 3 (2020), 658–682.
[26] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On first-order meta-learning algorithms. arXiv preprint arXiv:1803.02999 (2018).
[27] German I Parisi, Ronald Remker, Jose L Part, Christopher Kanan, and Stefan Wermter. 2019. Continual lifelong learning with neural networks: A review. Neural Networks 113 (2019), 54–71.
[28] Jihong Park, Sunumda Samarakoon, Mehdi Benni, and Mérioane Debah. 2019. Wireless network intelligence at the edge. Proc. IEEE 107, 11 (2019), 2204–2239.
[29] Aniruddha Raghu, Maithra Raghu, Sunny Berger, and Orson Vinyals. 2019. Rapid learning or feature reuse? towards understanding the effectiveness of mamf. arXiv preprint arXiv:1909.09157 (2019).
[30] Sachin Ravi and Hugo Larochelle. 2016. Optimization as a model for few-shot learning. (2016).
[31] Jonathan Schwartz, Julena Luketina, Wojciech M Czarnecki, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. 2018. Progress & compress: A scalable framework for continual learning. arXiv preprint arXiv:1805.06370 (2018).
[32] Si Si, Dacheng Tao, and Bo Geng. 2019. Bregman divergence-based regularization for transfer subspace learning. IEEE Transactions on Knowledge and Data Engineering 22, 7 (2009), 929–942.
[33] Xingyou Song, Wenbo Gao, Xuyang Yang, Krzysztof Choromanski, Aldo Pacchiano, and Yunhao Tang. 2019. Es-maml: Simple hessian-free meta learning. arXiv preprint arXiv:1910.01215 (2019).
[34] Masashi Takada and Hiroson Fujisawa. 2020. Transfer Learning via $f_\alpha$ Regularization. arXiv preprint arXiv:1910.01215 (2019).
[35] Qiong Wu, Kaiwen He, and Xu-Kun Xu. 2014. Convergence of Bregman alternating direction method with multipliers for nonconvex composite problems. Science China Information Sciences 61, 12 (2018), 122101.
[36] Fenghui Wang, Wenhao Cao, and Zongben Xu. 2018. Convergence of multi-block Bregman ADMM for nonconvex composite problems. Science China Information Sciences 61, 12 (2018), 122101.
[37] Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide. 2013. KL-divergence minimization with applications to CT imaging. arXiv preprint arXiv:1303.6361 (2013).
[38] Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. 2020. A Collaborative Learning Framework via Federated Meta-Learning. arXiv preprint arXiv:2002.07948 (2020).
APPENDIX

A PROOF OF LEMMA 1

The proof is standard. For simplicity, we denote $F_i(\theta) \triangleq L_i(\theta - \alpha \nabla L_i(\theta, D_i^t), D_i^t)$. Recall that $\nabla F_i(x) = (I_n - \alpha \nabla^2 L_i(x, D_i^t)) \nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t)$, and we have

$$
\|\nabla F_i(x) - \nabla F_i(y)\| \leq \|\nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t) - \nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\| \\
+ \alpha \|\nabla^2 L_i(x, D_i^t)\| \|\nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t) - \nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\|,
$$

and

$$
-\mu_i l_i \leq \nabla^2 L_i(x, D_i^t) \leq \mu_i l_i, \forall x \in \mathbb{R}^n.
$$

To prove (22), we need to bound (36) and (37). For (36), based on Assumption 2, we have

$$
\|\nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t) - \nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\| \\
\leq \mu_i \|x - y + \alpha \nabla L_i(x, D_i^t) - \nabla L_i(y, D_i^t)\| \\
\leq \mu_i (\|x - y\| + \alpha \|\nabla L_i(x, D_i^t) - \nabla L_i(y, D_i^t)\|) \\
\leq (1 + \alpha \mu_i) \|x - y\|.
$$

To bound (37), it can be shown that

$$
\|\nabla^2 L_i(x, D_i^t)\| \|\nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t) - \nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\| \\
\leq \|\nabla^2 L_i(x, D_i^t)\| \cdot \|\nabla L_i(x - \alpha \nabla L_i(x, D_i^t), D_i^t) - \nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\| \\
+ \|\nabla^2 L_i(x, D_i^t)\| \cdot \|\nabla L_i(y - \alpha \nabla L_i(y, D_i^t), D_i^t)\| \\
\leq (1 + \alpha \mu_i) \|x - y\|.
$$

where (40) follows from (18), (38), (39) and Assumption 3. Combining (39) and (40) yields the result.

B PROOF OF LEMMA 3

Based on Lemma 1, we next prove the Lemmas 3-6 for the convergence analysis. In the following lemma, we first bound the variations of $y_i^{t+1}$ via the variations of $\theta^t$. For simplicity, denote

$$
f_i(\theta) \triangleq L_i(\theta, D_i^t), \ f_i^t(\phi_i) \triangleq L_i(\phi_i, D_i^t), \ F_i(\theta) \triangleq L_i(\theta - \alpha \nabla L_i(\theta, D_i^t), D_i^t).
$$

**Lemma 3.** Suppose Assumption 1-3 are satisfied. Then, the following holds true

$$
\|y_i^{t+1} - y_i^t\| \leq w_i\|\theta^{t+1} - \theta^t\| + (\delta_i t + \tilde{\delta}_i t + 1)\alpha w_i \eta_i \beta_i^2.
$$

**Proof.** First, define

$$
\tilde{\nabla} F_i(\theta^{t+1}) \triangleq \nabla f_i^t(\phi_i^{t+1}) - \alpha g_i^{t+1},
$$

where $\phi_i^{t+1} = \theta^{t+1} - \alpha \nabla f_i(\theta^{t+1})$. We have the following observation from (12)

$$
w_i \tilde{\nabla} F_i(\theta^{t+1}) + y_i^t + \rho_i(\theta_i^{t+1} - \theta^t) = 0.
$$

Using (13), we conclude that (44) is equivalent to

$$
-y_i^{t+1} = w_i \tilde{\nabla} F_i(\theta^{t+1}).
$$

Thus, for all $t \in \mathbb{N}$, the following is true

$$
\|y_i^{t+1} - y_i^t\| = w_i \|\tilde{\nabla} F_i(\theta^{t+1}) - \tilde{\nabla} F_i(\theta^t)\|.
$$

(46)
Using Lemma 1, we derive the upper bound of (a) as follows
\[ (a) = \| \nabla F_i(\theta^{t+1}) - \nabla F_i(\theta^t) \| \]
\[ \leq \| \nabla F_i(\theta^{t+1}) - \nabla F_i(\theta^t) \| + \| \nabla F_i(\theta^{t+1}) - \nabla F_i(\theta^t) \| \]
\[ \leq \| \nabla F_i(\theta^{t+1}) - \nabla F_i(\theta^t) \| + \| \nabla F_i(\theta^{t+1}) - \nabla F_i(\theta^t) \| + \| \nabla F_i(\theta^t) - \nabla F_i(\theta^t) \| \]
\[ \leq \omega_i \| \theta^{t+1} - \theta^t \| + (\delta_{t,i} + \delta_{t,i+1}) \alpha \xi_i \beta_i^2, \]  
(47)

where the last equality uses the following result in [7]
\[ \| \nabla^2 f_i(\theta^{t+1}) \nabla^T q(\phi_i^{t+1}) - g_i^{t+1} \| \leq \delta_{t,i+1} \xi_i \beta_i^2. \]  
(48)

Plugging (47) into (46), we have
\[ \| y_i^{t+1} - y_i^t \| \leq \omega_i \| \theta^{t+1} - \theta^t \| + (\delta_{t,i} + \delta_{t,i+1}) \alpha \omega_i \xi_i \beta_i^2, \]  
(49)

which completes the proof.

\[ \square \]

C PROOF OF LEMMA 4

To bound the successive difference of the augmented Lagrangian function \( \mathcal{L}(\theta^t, \theta^{t+1}, y^t_i) \) defined in (6), we first bound the successive difference of \( \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) \), which is defined as follows
\[ \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) = \mathcal{L}(\theta^t, \theta^{t+1}, y^t_i). \]  
(50)

Then, we have the following lemma.

Lemma 4. Suppose that Assumption 1-3 are satisfied. The following holds true
\[ \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) - \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) \leq -\rho_i \frac{1}{2} \| \theta_i - \theta^{t+1} \|^2 + \frac{1}{2} \| \theta_i - \theta^{t+1} \|^2 \]
\[ + \frac{2 \omega_i \xi_i \beta_i^2 \delta_{t,i+1}}{\rho_i} \| y_i^{t+1} - y_i^t \| + \alpha \omega_i \xi_i \beta_i^2 \frac{1}{\rho_i} \| \theta_i - \theta^{t+1} \|. \]

Proof. First, we define \( \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \) and \( \tilde{L}_i(\theta_i, \theta^{t+1}, y^t_i) \) as follows
\[ \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) = \omega_i \left( \left( 1 + \alpha \nabla^2 f_i(\theta^{t+1}) \right) \nabla^T q(\phi_i^{t+1}), \theta_i - \theta^{t+1} \right) + \omega_i \bar{F}_i(\theta^{t+1}) + \langle y_i^t, \theta_i - \theta^{t+1} \rangle \]
\[ + \frac{\rho_i}{2} \| \theta_i - \theta^{t+1} \|^2, \]  
(51)
\[ \tilde{L}_i(\theta_i, \theta^{t+1}, y^t_i) = \omega_i \left( \nabla^T q(\phi_i^{t+1}) - \alpha g_i^{t+1}, \theta_i - \theta^{t+1} \right) + \omega_i \bar{F}_i(\theta^{t+1}) + \langle y_i^t, \theta_i - \theta^{t+1} \rangle \]
\[ + \frac{\rho_i}{2} \| \theta_i - \theta^{t+1} \|^2, \]  
(52)

where \( g_i^{t+1} \) and \( \phi_i^{t+1} \) are defined in (11) and (8), respectively. For each \( i \in I \), using Lemma 1 yields
\[ \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) \leq \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \]
\[ + \frac{\omega_i \xi_i \beta_i^2}{2} \| \theta_i - \theta^{t+1} \|. \]  
(53)

Recall that
\[ \| \nabla^2 f_i(\theta^{t+1}) \nabla^T q(\phi_i^{t+1}) - g_i^{t+1} \| \leq \xi_i \beta_i^2 \delta_{t,i+1}. \]  
(54)

Thus, using the Cauchy-Schwarz inequality, we can write
\[ \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \leq \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) + \alpha \omega_i \xi_i \beta_i^2 \delta_{t,i+1} \| \theta_i - \theta^{t+1} \|. \]  
(55)

Combining (55) and (57) yields that
\[ \mathcal{L}_i(\theta_i, \theta^{t+1}, y^t_i) \leq \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) + \frac{\omega_i \xi_i \beta_i^2}{2} \| \theta_i - \theta^{t+1} \|^2 + \alpha \omega_i \xi_i \beta_i^2 \delta_{t,i+1} \| \theta_i - \theta^{t+1} \|. \]  
(56)

Based on (44) and the strong convexity of \( \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \) with modulus \( \rho_i \), we can show that for each \( i \in I \)
\[ \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) - \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \leq -\frac{\rho_i}{2} \| \theta_i^{t+1} - \theta_i^t \|^2. \]  
(57)

It follows that
\[ \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) - \hat{L}_i(\theta_i, \theta^{t+1}, y^t_i) \]
\[ = \omega_i \left( \nabla^T q(\phi_i^{t+1}) - \alpha g_i^{t+1}, \theta_i - \theta^{t+1} \right) + \omega_i \bar{F}_i(\theta^{t+1}) - \omega_i \bar{F}_i(\theta_i) \]
\[ = \omega_i \bar{F}_i(\theta^{t+1}) - \omega_i \bar{F}_i(\theta_i) + \omega_i \left( \nabla^T q(\phi_i^{t+1}) - \alpha g_i^{t+1}, \theta_i - \theta^{t+1} \right) - \frac{\omega_i \xi_i \beta_i^2}{2} \| \theta_i - \theta^{t+1} \|^2 + \frac{\omega_i \xi_i \beta_i^2}{2} \| \theta_i - \theta^{t+1} \|^2. \]
where \((c)\) is derived from \((13)\). This completes the proof.

where \((a)\) is derived from Lemma 1 and \((b)\) is based on the following fact

\[
\begin{align*}
\text{D PROOF OF LEMMA 5} & \\
\text{that due to the error induced by linear approximation and first-order Hessian estimation, the} & \\
\text{inexact-ADMM based federated meta-learning for fast and continual edge learning} & \\
\text{MobiHoc ‘21, July 26–29, 2021, Shanghai, China} & \\
\end{align*}
\]

where \((a)\) is derived from Lemma 1 and \((b)\) is based on the following fact

\[
\|x + y\|^2 \leq 2\|x\|^2 + 2\|y\|^2, \quad x, y \in \mathbb{R}^n.
\]

Combining (58)-(60), we conclude that

\[
\begin{align*}
\mathcal{L}(\theta_{t+1}^i, \theta_{t+1}^{i'}, y_{t+1}^i) - & \mathcal{L}_i(\theta_{t+1}^i, \theta_{t+1}^{i'}, y_{t+1}^i) \\
\leq & \sum_{d=1}^D \mathcal{L}_d(\theta_{t+1}^i, \theta_{t+1}^{i'}, y_{t+1}^i) + \mathcal{L}_i(\theta_{t+1}^i, \theta_{t+1}^{i'}, y_{t+1}^i) - \mathcal{L}_i(\theta_{t+1}^i, \theta_{t+1}^{i'}, y_{t+1}^i) \\
+ & \frac{\gamma_i}{2} \|\theta_{t+1}^i - \theta_{t+1}^{i'}\|^2 + \alpha \frac{\gamma_i}{2} \|\theta_{t+1}^i - \theta_{t+1}^{i'}\|^2
\end{align*}
\]

where \((c)\) is derived from \((13)\). This completes the proof.

\[
\begin{align*}
\text{D PROOF OF LEMMA 5} & \\
\text{Based on Lemma 4, we derive the successive difference of the augmented Lagrangian function} & \\
\text{\(\mathcal{L}(\{\theta_t^i, y_t^i\}, \theta^t)\) in the following lemma. Note} & \\
\text{that due to the error induced by linear approximation and first-order Hessian estimation, the} & \\
\text{sufficient descent does not hold below.} & \\
\end{align*}
\]

\textbf{Lemma 5.} Suppose that Assumption 1-3 hold. Then the following holds

\[
\mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta_{t+1}^i) - \mathcal{L}(\{\theta_t^i, y_t^i\}, \theta^t) \leq \sum_{i \in I} \left( a_{i,e} \|\theta_{t+1}^i - \theta_t^i\|^2 + a_{i,p} \|\theta_{t+1}^i - \theta^t\|^2 \\
- b_{t+1}^i \|\theta_{t+1}^i - \theta_t^i\|^2 - b_{t+1}^{i,p} \|\theta_{t+1}^i - \theta^t\|^2 - c_{t+1}^i \right),
\]

where \(a_{i,e} \) and \(a_{i,p} \) are defined in \((24)\) and \((25)\), respectively; \(b_{t+1}^i \), \(b_{t+1}^{i,p} \), and \(c_{t+1}^i \) are defined as follows

\[
\begin{align*}
b_{t+1}^i & \triangleq \alpha \gamma_i \beta_i^2 \delta_{i,t+1}, \\
b_{t+1}^{i,p} & \triangleq \frac{2 \alpha \gamma_i \beta_i^2 \delta_{i,t+1}^2}{\rho_i}, \\
c_{t+1}^i & \triangleq 2(\delta_{i,t} + \delta_{i,t+1})^2 \left( \frac{4 \gamma_i}{\rho_i^2} + \frac{1}{\rho_i} \right) + \frac{2 \alpha \gamma_i \beta_i^2 \delta_{i,t+1}^2}{\rho_i} (\delta_{i,t} + \delta_{i,t+1}).
\end{align*}
\]

\textbf{Proof.} Based on the update \((13)\), we first obtain

\[
\begin{align*}
\mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta_{t+1}^i) - \mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta^t) & \\
= & \sum_{i \in I} \left( y_{t+1}^i - y_t^i, \theta_{t+1}^i - \theta^t \right) \\
= & \sum_{i \in I} \frac{1}{\rho_i} \|y_{t+1}^i - y_t^i\|^2.
\end{align*}
\]

Using Lemma 4 and Assumption 2, we have

\[
\begin{align*}
\mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta_{t+1}^i) - & \mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta^t) \\
= & \mathcal{L}(\{\theta_{t+1}^i, y_{t+1}^i\}, \theta_{t+1}^i) - \mathcal{L}(\{\theta_{t+1}^i, y_t^i\}, \theta_{t+1}^i) + \mathcal{L}(\{\theta_{t+1}^i, y_t^i\}, \theta_{t+1}^i) - \mathcal{L}(\{\theta_{t+1}^i, y_t^i\}, \theta^t)
\end{align*}
\]
Given Assumption 1-5, the augmented Lagrangian function defined in (6) is lower bounded by Algorithm 1. Moreover, due to Lemma 1, we can write

$$-y_i^{t+1} = w_i \nabla F_i(\theta^{t+1})$$

(70)

where $\nabla F_i(\theta^{t+1}) = \nabla f_i(\theta_i^{t+1}) - \alpha g_i^{t+1}$. Besides, due to Lemma 1, we can write

$$F_i(\theta^{t+1}) \leq F_i(\theta_i^{t+1}) + (\nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + \frac{w_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2$$

and

$$= F_i(\theta_i^{t+1}) + (\nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + (\nabla F_i(\theta_i^{t+1}) - \nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + \frac{w_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2$$

$$\leq F_i(\theta_i^{t+1}) + F_i(\theta_i^{t+1}) + 3\frac{w_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2.$$  

(71)

Based on the definition of the augmented Lagrangian function (6) and (71), we can show the following observation

$$\mathcal{L}(\theta_i^{t+1}, y_i^{t+1}) = \lambda D_h(\theta^{t+1}, \theta_p) + \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + (y_i^{t+1}, \theta^{t+1} - \theta_i^{t+1}) + \frac{p_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2 \right)$$

$$= \lambda D_h(\theta^{t+1}, \theta_p) + \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + (w_i \nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + \frac{p_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2 \right)$$

$$= \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + (w_i \nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + \frac{p_i}{2} \|\theta^{t+1} - \theta_i^{t+1}\|^2 \right) + \lambda D_h(\theta^{t+1}, \theta_p)$$

$$= \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + (w_i \nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) + w_i(\nabla F_i(\theta_i^{t+1}) - \nabla F_i(\theta_i^{t+1}), \theta^{t+1} - \theta_i^{t+1}) \right)$$

E  PROOF OF LEMMA 6

In the next lemma, we show that the augmented Lagrangian function $\mathcal{L}(\theta_i^t, y_i^t)$ is lower bounded for any $t \in \mathbb{N}$.

**Lemma 6.** Given Assumption 1-5, the augmented Lagrangian function defined in (6) is lower bounded by Algorithm 1.

**Proof.** Recall that in (45)

$$-y_i^{t+1} = w_i \nabla F_i(\theta^{t+1})$$

(70)

thereby completing the proof. □
\[ \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + \langle w_i \nabla F_i(\theta_i^{t+1}), \theta_i^{t+1} - \theta_i^t \rangle - w_i \| \nabla F_i(\theta_i^{t+1}) - \nabla F_i(\theta_i^t) \| \cdot \| \theta_i^{t+1} - \theta_i^t \| \right) \\
+ \frac{\rho_i}{2} \| \theta_i^{t+1} - \theta_i^t \|^2 \right) + \lambda D_h(\theta_i^{t+1}, \theta_p) \\
\geq \sum_{i \in I} \left( w_i F_i(\theta_i^{t+1}) + \frac{\rho_i - 3\nu_i}{2} \| \theta_i^{t+1} - \theta_i^t \|^2 - \alpha w_i \xi_i^2 \delta_i, t+1 \| \theta_i^{t+1} - \theta_i^t \|^2 \right) + \lambda D_h(\theta_i^{t+1}, \theta_p), \]

where the last inequality is derived from \( \nu_i \)-smoothness of \( F_i \) and first-order Taylor expansion. Due to Assumption 1, \( \lambda D_h(\theta_i^{t+1}, \theta_p) + \sum_{i \in I} w_i F_i(\theta_i^{t+1}) \) is lower bounded. According to Assumption 5, it is easy to show that

\[ \sum_{i \in I} \left( \frac{\rho_i - 3\nu_i}{2} \| \theta_i^{t+1} - \theta_i^t \|^2 - \alpha w_i \xi_i^2 \delta_i, t+1 \| \theta_i^{t+1} - \theta_i^t \|^2 \right) > -\infty, \]

thereby completing the proof.

\[ \square \]

### F PROOF OF THEOREM 1

First, we prove part (i). Note that the RHS of (63) is the sum of some independent quadratic functions of \( \| \theta_i^{t+1} - \theta_i^t \| \) and \( \| \theta_i^{t+1} - \theta_i^t \| \). Due to Assumption 5 and Lemma 5, for each \( i \in I \), based on the form of roots of quadratic function, it is easy to see that there exist \( \sigma_i^{t+1} \) and \( y_i^{t+1} \) such that

\[ \lim_{t \to \infty} \sigma_i^{t+1} = 0, \]
\[ \lim_{t \to \infty} y_i^{t+1} = 0. \]

When \( \| \theta_i^{t+1} - \theta_i^t \| > \sigma_i^{t+1} \),
\[ a_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 - c_i^{t+1} > 0; \]
and when \( \| \theta_i^{t+1} - \theta_i^t \| > y_i^{t+1} \),
\[ a_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 > 0. \]

Next, we show \( \lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0 \) and \( \lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0 \) by two steps.

1) Suppose that there exists \( T \geq 0 \) such that for all \( t \geq T \), the following is true

\[ \sum_{i \in I} \left( a_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 + a_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - c_i^{t+1} \right) > 0. \]

It follows that under Assumption 5, using Lemma 5-6, \( L \{ (\theta_i^{t+1}, y_i^{t+1}), \theta_i^{t+1} \} \) will monotonically decrease and converges. Thus, we obtain

\[ \lim_{t \to \infty} \sum_{i \in I} \left( a_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 + a_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,e} \| \theta_i^{t+1} - \theta_i^t \|^2 + a_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - b_{i,p} \| \theta_i^{t+1} - \theta_i^t \|^2 - c_i^{t+1} \right) = 0, \]

which implies that \( \| \theta_i^{t+1} - \theta_i^t \| \) and \( \| \theta_i^{t+1} - \theta_i^t \| \) converge to the positive roots of corresponding quadratic functions, i.e., LHS of (74) and (75), otherwise the limitation will not be 0. Due to (73), the positive roots of the above quadratic function converge to 0, which implies

\[ \lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0, \forall i \in I, \]
\[ \lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0. \]

By Lemma 3 and (13), we can show that

\[ \lim_{t \to \infty} \| y_i^{t+1} - y_i^t \| = 0, \forall i \in I, \]
\[ \lim_{t \to \infty} \| y_i^{t+1} - y_i^t \| = 0, \forall i \in I. \]

2) Suppose that there exists a sequence \( \{ t_j \mid j \in \mathbb{N} \} \) such that

\[ \sum_{i \in I} \left( a_{i,e} \| \theta_i^{t_j+1} - \theta_i^t \|^2 + a_{i,p} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - b_{i,e} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - b_{i,p} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - c_i^{t_j+1} \right) \leq 0. \]

Due to Assumption 5, the minimum value of the above quadratic function converges to 0, which implies

\[ \lim_{t \to \infty} \sum_{i \in I} \left( a_{i,e} \| \theta_i^{t_j+1} - \theta_i^t \|^2 + a_{i,p} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - b_{i,e} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - b_{i,p} \| \theta_i^{t_j+1} - \theta_i^t \|^2 - c_i^{t_j+1} \right) = 0. \]
Similar to (78) and (79), we have
\[
\lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0, \forall i \in \mathcal{I},
\]
\[
\lim_{t \to \infty} \| \theta_t^{t+1} - \theta_t^t \| = 0.
\]

We also define a nontrivial sequence \( \{\eta_q \mid q \in \mathbb{N}\} \supset \mathbb{N} - \{t_j \mid j \in \mathbb{N}\} \). Note that
\[
\sum_{i \in \mathcal{I}} \left( a_i, e \| \theta_i^{t+1} - \theta_i^t \|^2 + a_i, p \| \theta_t^{t+1} - \theta_t^t \|^2 - b_i, p \| \theta_i^{t+1} - \theta_i^t \|^2 - b_i, p \| \theta_t^{t+1} - \theta_t^t \|^2 - c_i, p \| \theta_t^{t+1} - \theta_t^t \|^2 - \right) > 0.
\]

Similar to 1), we have
\[
\lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0, \forall i \in \mathcal{I},
\]
\[
\lim_{t \to \infty} \| \theta_t^{t+1} - \theta_t^t \| = 0.
\]

Based on the above observations, for any \( \eta > 0 \), there exists \( k \geq 2 \) such that when \( j > k \) and \( q > k \), the following holds true
\[
\| \theta_i^{t+1} - \theta_i^t \| \leq \eta, \forall i \in \mathcal{I},
\]
\[
\| \theta_t^{t+1} - \theta_t^t \| \leq \eta,
\]
\[
\| \theta_i^{t+1} - \theta_i^t \| \leq \eta, \forall i \in \mathcal{I},
\]
\[
\| \theta_t^{t+1} - \theta_t^t \| \leq \eta.
\]

Thus, for any \( t > t_k \), we can write
\[
\| \theta_i^{t+1} - \theta_i^t \| \leq \eta, \forall i \in \mathcal{I},
\]
\[
\| \theta_t^{t+1} - \theta_t^t \| \leq \eta,
\]
which implies that
\[
\lim_{t \to \infty} \| \theta_i^{t+1} - \theta_i^t \| = 0, \forall i \in \mathcal{I},
\]
\[
\lim_{t \to \infty} \| \theta_t^{t+1} - \theta_t^t \| = 0,
\]

thereby (78)-(81) hold.

Using the optimality condition of (12) leads to
\[
\omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) + \gamma_i^{t+1} = 0,
\]
where \( \nabla F_i \left( \theta_i^{t+1} \right) \equiv \nabla f_i \left( \theta_i^{t+1} \right) - \alpha g_i^{t+1} \). For each \( i \in \mathcal{I} \), we derive an upper bound of \( \| \nabla \theta_i^{t+1} \mathcal{L} \left( \{ \theta_i^{t+1}, y_i^{t+1} \}, \theta^{t+1} \} \| \) as
\[
\| \nabla \theta_i^{t+1} \mathcal{L} \left( \{ \theta_i^{t+1}, y_i^{t+1} \}, \theta^{t+1} \} \|
\leq \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) + \gamma_i^{t+1} + \rho_i \| \theta_i^{t+1} - \theta^t \| \| \|
\leq \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) + \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) + \gamma_i^{t+1} \| \|
\leq \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) - \omega_{i} \nabla F_i \left( \theta_i^{t} \right) \| \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) \| + \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) \| \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) \| \|
\leq \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) \| \| \omega_{i} \nabla F_i \left( \theta_i^{t+1} \right) \| + \rho_i \| \theta_i^{t+1} - \theta^t \| \| \gamma_i^{t+1} \| \|
\leq \alpha \omega_{i} \gamma_i^{t+1} \delta_i^{t, t+1}.
\]

Taking limitation of \( t \to \infty \) on both sides of (98), and using Assumption 5 and (81) yields
\[
\| \nabla \theta_i \mathcal{L} \left( \{ \theta_i, y_i \}, \theta^* \} \| = 0, \forall i \in \mathcal{I}.
\]

Note that
\[
\| \nabla \theta_i \mathcal{L} \left( \{ \theta_i, y_i \}, \theta^* \} \| \leq \sum_{i \in \mathcal{I}} \left( \rho_i \| \theta_i^{t+1} - \theta_i^t \| + \omega_i v_i \| \theta_i^{t+1} - \theta_i^t \| + \alpha \omega_{i} \gamma_i^{t+1} \delta_i^{t, t+1} \right).
\]

Similarly, we obtain
\[
\nabla \theta \mathcal{L} \left( \{ \theta, y \}, \theta^* \} = 0.
\]

Finally, we bound \( \| \sum_{i \in \mathcal{I}} \omega_i \nabla F_i \left( \theta_i^{t+1} \right) \| \lambda \mathcal{D}_h \left( \theta_i^{t+1}, \theta_p^t \right) \| \) by
\[
\| \sum_{i \in \mathcal{I}} \omega_i \nabla F_i \left( \theta_i^{t+1} \right) \| \lambda \mathcal{D}_h \left( \theta_i^{t+1}, \theta_p^t \right) \|.
where $a_2$ and $a_1$ corresponding to $\rho_i$ such that

$$
\sum_{i=0}^{\infty} z^t \leq L((\theta_i^0, y_i^0), (\theta^0)) - L((\theta_i^0, y_i^0), \theta^0) < \infty,
$$

where $z^t$ is denoted by

$$
z^t \doteq a_2 \sum_{i \in I} \left( \|\theta_i^{t+1} - \theta_i^t\|^2 + \|\theta_i^{t+1} - \theta^t\|^2 \right) - a_1 \sum_{i \in I} \left( \delta_{i,t+1} (\|\theta_i^{t+1} - \theta_i^t\|^2 + \|\theta_i^{t+1} - \theta^t\|) + \delta_{i,t}^2 \right) \quad (104)
$$

$$
z^2_t - z^1_t \quad (105)
$$

$z^2_t, z^1_t$ are denoted as the first and second sum terms, respectively. Due to Assumption 5 and Theorem 1, it is easy to see that there exists some positive constant $a_3$ such that the following holds true

$$
\sum_{i=0}^{\infty} z^0_i = a_1 \sum_{i=0}^{\infty} \sum_{i \in I} \left( \delta_{i,t+1} (\|\theta_i^{t+1} - \theta_i^t\|^2 + \|\theta_i^{t+1} - \theta^t\|) + \delta_{i,t}^2 \right) \quad (106)
$$

$$
n \sum_{i=0}^{\infty} z^0_i \leq b < \infty, \text{ for some constant } b > 0. \quad (109)
$$

Due to $\sum_{i=0}^{\infty} z^0_i < \infty$, it is easy to see that the augmented Lagrangian function is upper bounded and $\theta^t$ is finite, which shows $\{\theta^t\}$ has at least one limit point. Denoting $T^2(\epsilon) \doteq \min \{ t \mid \|\theta^t - \theta^0\|^2 \leq \epsilon, t \geq 0 \}$ and $T^2(\epsilon) \doteq \min \{ t \mid \|\theta^t - \theta^0\|^2 \leq \epsilon, t \geq 0 \}$, then we can write

$$
a_2 T^2(\epsilon) \leq \sum_{i=1}^{\infty} z^2_i \leq b, \quad (110)
$$

$$
a_2 T^2(\epsilon) \leq \sum_{i=1}^{\infty} z^1_i \leq b. \quad (111)
$$

That is, $T^2(\epsilon) = O(1/\epsilon)$ and $T^2(\epsilon) = O(1/\epsilon)$ hold. Further, we denote $T(\epsilon) \doteq \min \{ t \mid \|\theta^t - \theta^0\|^2 \leq \epsilon, t \geq 0 \}$ and $T(\epsilon) \doteq \min \{ t \mid \|\theta^t - \theta^0\|^2 \leq \epsilon, t \geq 0 \}$. Based on (110) and (111), we have $T(\epsilon) = O(1/\epsilon^2)$ and $T(\epsilon) = O(1/\epsilon^2)$. Due to Assumption 5, combining (42) and (98) yields

$$
\|\nabla_{\theta_i} \mathcal{L}((\theta_i^{t+1}, y_i^{t+1}), \theta^t)\| \leq \frac{(w_i^1 + \rho_i)w_i^1}{\rho_i} \|\theta^{t+1} - \theta^t\|^2 + \frac{(2w_i^1 + 3\rho_i)\sigma w_i^1 \beta^2}{\rho_i} \cdot \delta_{i,t}. \quad (112)
$$

Similarly, it is easy to see that the convergence rate of $\delta_{i,t}$ is $O(1/\epsilon)$. Therefore, for any $\epsilon > 0$, Algorithm 1 finds a point $((\theta_i), \theta, y)$ with $\|\nabla_{\theta_i} \mathcal{L}((\theta_i, y_i), \theta)\| \leq \epsilon$, after at most $O(1/\epsilon^2)$. In the same way, it can be shown that

$$
\|\nabla_{\theta_i} \mathcal{L}((\theta_i^{t+1}, y_i^{t+1}), \theta^t)\| \leq \sum_{i \in I} \left( \rho_i \|\theta_i^{t+1} - \theta_i^t\|^2 + w_i^1 \|\theta^{t+1} - \theta^t\|^2 + 2\sigma w_i^1 \beta^2 \delta_{i,t} \right). \quad (113)
$$
which implies that $\|\nabla_{\theta^{t+1}} \mathcal{L}(\theta^{t+1}, y^{t+1})\|$ has the same communication complexity as $\|\nabla_{\theta^{t+1}} \mathcal{L}(\theta^{t+1}, y^{t+1})\|$. Moreover, it is easy to show that $\|\nabla_{y_i} \mathcal{L}(\theta, y_i, \theta)\|$ also the same complexity. Combining with (102), it completes the proof.

\section*{G \hspace{1em} PROOF OF COROLLARY 1}

First, we prove the following result.

\begin{align}
\mathbb{E}\left[ \|\nabla L_i(\theta - \alpha \nabla L_i(\theta)) - \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^q))\| \right] & \leq \frac{\alpha \mu_i \sigma_i^2}{\sqrt{D_i^q}} + \frac{\sigma_i^2}{\sqrt{D_i^q}}. \quad (114)
\end{align}

We can write

\begin{align}
\mathbb{E}\left[ \|\nabla L_i(\theta - \alpha \nabla L_i(\theta)) - \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^q))\| \right] \\
\leq \mathbb{E}\left[ \|\nabla L_i(\theta - \alpha \nabla L_i(\theta)) - \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^q))\| \right] + \mathbb{E}\left[ \|\nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^q)) - \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^q))\| \right]. \quad (115)
\end{align}

For (b), due to Assumption 4, we have:

\begin{align}
\left( c \right) & \leq \sqrt{\mathbb{E}\left[ \|\nabla L_i(\phi_i) - \frac{1}{D_i^q} \sum_{j=1}^{D_i^q} \nabla l(\phi_i, (x_i^j, y_i^j))\|^2 \right]} \\
& = \frac{1}{(D_i^q)^2} \mathbb{E}\left[ \sum_{j=1}^{D_i^q} \|\nabla L_i(\phi_i) - \nabla l(\phi_i, (x_i^j, y_i^j))\|^2 \right] \\
& \leq \frac{\sigma_i^2}{\sqrt{D_i^q}}. \quad (116)
\end{align}

where $\phi_i = \theta - \alpha \nabla L_i(\theta, D_i^q)$. For (a), based on Assumption 2, it can be bounded by:

\begin{align}
\left( b \right) & \leq \mathbb{E}\left[ \alpha \mu_i \|\nabla L_i(\theta, D_i^q)\| - \nabla L_i(\theta)\right] \leq \frac{\alpha \mu_i \sigma_i^2}{\sqrt{D_i^q}}. \quad (117)
\end{align}

Plugging (116) and (117) into (115), (114) holds. Based on (114), the following holds

\begin{align}
\mathbb{E}\left[ \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p)) \right] \\
\leq \sum_{i \in I} \mathbb{E}\left[ \|\nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e) - \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e, D_i^q))\| + \mathbb{E}\left[ \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e, D_i^q)) + \lambda \nabla D_h(\theta_e, \theta_p)\right]\right] \\
\leq \varepsilon + \sum_{i \in I} w_i \sigma_i^2 \left( \frac{\alpha \mu_i}{\sqrt{D_i^q}} + \frac{1}{\sqrt{D_i^q}} \right), \quad (118)
\end{align}

thereby completing the proof.

\section*{H \hspace{1em} PROOF OF THEOREM 2}

Let $\theta_e$ denote the $\epsilon$-FOSP obtained by Algorithm 1, which satisfies that

\begin{align}
\|\sum_{i \in I} w_i \nabla F_i(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p)\| \leq \varepsilon, \quad (119)
\end{align}

for some $\varepsilon > 0$. Then, for the learned model parameter $\theta_e$, $\mathbb{E}\left[ \|\nabla F_m(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p)\| \right]$ can be upper bounded by

\begin{align}
\mathbb{E}\left[ \|\nabla F_m(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p)\| \right] & = \mathbb{E}\left[ \|\sum_{i \in I} w_i \nabla F_i(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p) + \sum_{i \in I} w_i (\nabla F_m(\theta_e) - \nabla F_i(\theta_e))\| \right] \\
& \leq \varepsilon + \mathbb{E}\left[ \|\nabla F_m(\theta_e) - \sum_{i \in I} w_i \nabla F_i(\theta_e)\| \right]. \quad (120)
\end{align}
Due to Assumption 6, for $i \in I \cup \{m\}$ and $D_i$ with respect to $P_i$, we can write
\[
\mathbb{E}\left\{ \| \nabla L_i(\theta, D_i) - \nabla L_i(\theta) \| \right\} \leq \frac{\sigma^g_i}{\sqrt{D_i}}
\] (121)

Based on (121), observe that
\[
\mathbb{E}\left\{ \| \nabla L_m(\theta, D_m) - \nabla L_i(\theta, D_i) \| \right\} \\
\leq \mathbb{E}\left\{ \| \nabla L_m(\theta, D_m) - \nabla L_m(\theta) \| \right\} + \mathbb{E}\left\{ \| \nabla L_m(\theta) - \nabla L_i(\theta) \| \right\} + \mathbb{E}\left\{ \| \nabla L_i(\theta, D_i) - \nabla L_i(\theta) \| \right\} \\
\leq \psi^g_i + \frac{\sigma^g_i}{\sqrt{D_m}} + \frac{\sigma^g_i}{\sqrt{D_i}}.
\] (122)

Similarly, we can show that
\[
\mathbb{E}\left\{ \| \nabla^2 L_m(\theta, D_m) - \nabla^2 L_i(\theta, D_i) \| \right\} \leq \psi^h_i + \frac{\sigma^h_i}{\sqrt{D_m}} + \frac{\sigma^h_i}{\sqrt{D_i}}.
\] (123)

Thus, for (a), we obtain
\[
\left\| \nabla F_m(\theta) - \sum_{i \in I} w_i \nabla F_i(\theta) \right\| \\
= \left\| \nabla L_m(\theta - \alpha \nabla L_m(\theta, D_m^i), D_m^i) - \sum_{i \in I} w_i \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i) \right\| \\
\leq \left\| \nabla L_m(\theta - \alpha \nabla L_m(\theta, D_m^i), D_m^i) - \sum_{i \in I} w_i \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i) \right\| + \alpha \left\| \nabla^2 L_m(\theta, D_m^i) \nabla L_m(\theta - \alpha \nabla L_m(\theta, D_m^i), D_m^i) \right\| - \sum_{i \in I} w_i \nabla^2 L_i(\theta, D_i^i) \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i)
\] (b)

Based on Assumption 2 and 6, we have
\[
\mathbb{E}(b) \leq \mathbb{E}\left\{ \| \nabla L_m(\theta - \alpha \nabla L_m(\theta, D_m^i), D_m^i) - \sum_{i \in I} w_i \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i) \right\| \\
+ \left\| \sum_{i \in I} w_i \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i) - \sum_{i \in I} w_i \nabla L_i(\theta - \alpha \nabla L_i(\theta, D_i^i), D_i^i) \right\| \\
\leq \sum_{i \in I} w_i \left( \psi^g_i + \frac{\sigma^g_i}{\sqrt{D_m^i}} + \frac{\sigma^g_i}{\sqrt{D_i^i}} \right) + \alpha \mu \sum_{i \in I} w_i \left( \psi^h_i + \frac{\sigma^h_i}{\sqrt{D_m^i}} + \frac{\sigma^h_i}{\sqrt{D_i^i}} \right).
\] (125)

(d)

Similarly, for (c), it follows that
\[
\mathbb{E}(c) \leq \alpha \beta_m \sum_{i \in I} w_i \left( \psi^h_i + \frac{\sigma^h_i}{\sqrt{D_m^i}} + \frac{\sigma^h_i}{\sqrt{D_i^i}} \right) + \alpha \mu \cdot (d).
\] (126)
Plugging (125) and (126) in (124) yields
\[
\mathbb{E}\{d\} \leq \alpha \beta m \sum_{i \in I} w_i \left( \psi_i^h + \frac{\sigma_i^h}{\sqrt{D_i^m}} + \frac{\sigma_i^h}{\sqrt{D_i^q}} \right) + (\alpha \mu + 1)(\alpha \mu) \sum_{i \in I} w_i \left( \psi_i^g + \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
+ (\alpha \mu + 1) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right) + (\alpha \mu + 1) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
\]
\[
= \alpha \beta m \sum_{i \in I} w_i \psi_i^h + (\alpha \mu + 1)^2 \sum_{i \in I} w_i \psi_i^g + \alpha \beta m \sum_{i \in I} w_i \left( \frac{\sigma_i^h}{\sqrt{D_i^m}} + \frac{\sigma_i^h}{\sqrt{D_i^q}} \right) + (\alpha \mu + 1) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
+ (\alpha \mu + 1)(\alpha \mu) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right) + (\alpha \mu + 1) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
\]
\]
Therefore, plugging (127) in (120), we obtain
\[
\mathbb{E}\{\|F_m(\theta_e) + \lambda \nabla D_h(\theta_e, \theta_p)\|\} \leq \epsilon + \alpha \beta m \sum_{i \in I} w_i \psi_i^h + (\alpha \mu + 1)^2 \sum_{i \in I} w_i \psi_i^g + \alpha \beta m \sum_{i \in I} w_i \left( \frac{\sigma_i^h}{\sqrt{D_i^m}} + \frac{\sigma_i^h}{\sqrt{D_i^q}} \right)
+ (\alpha \mu + 1)(\alpha \mu) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right) + (\alpha \mu + 1) \sum_{i \in I} w_i \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
\]
thereby completing the proof.

I PROOF OF LEMMA 2

From Corollary 1, the following holds
\[
\mathbb{E}\{\left\| \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e)) + \lambda \nabla D_h(\theta_e, \theta_p) \right\} \leq \epsilon + \sum_{i \in I} w_i \sigma_i^g \left( \frac{\sigma_i^g}{\sqrt{D_i^m}} + \frac{1}{\sqrt{D_i^q}} \right).
\]

Thus, based on Assumption 2, we can obtain
\[
\lambda \|\nabla D_h(\theta_e, \theta_p)\| = \left\| \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e)) + \lambda \nabla D_h(\theta_e, \theta_p) - \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e)) \right\|
\leq \left\| \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e)) + \lambda \nabla D_h(\theta_e, \theta_p) \right\| + \left\| \sum_{i \in I} w_i \nabla L_i(\theta_e - \alpha \nabla L_i(\theta_e)) \right\|
\leq \epsilon + \sum_{i \in I} w_i \left( \beta_i + \frac{\alpha \mu \sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right)
\]
\[
(129)
\]
Due to the convexity of $D_h(\cdot, \theta_p)$ and $D_h(\theta_p, \theta_p) = 0$, we have
\[
\mathbb{E}\{D_h(\theta, \theta_p)\} \leq \frac{1}{\alpha} \left( \epsilon + \sum_{i \in I} w_i \left( \beta_i + \frac{\alpha \mu \sigma_i^g}{\sqrt{D_i^m}} + \frac{\sigma_i^g}{\sqrt{D_i^q}} \right) \right) \|\theta_e - \theta_p\|.
\]
\[
(130)
\]
Equation (34) can be directly derived via (33).