DP-Norm: Differential Privacy Primal-Dual Algorithm for Decentralized Federated Learning

Takumi Fukami, Tomoya Murata, Kenta Niwa, *Senior Member, IEEE*, and Iifan Tyou, *Member, IEEE*

***Abstract*— A novel algorithm is proposed for highly privacy-preserving decentralized federated learning (FL). Several studies have reported security risks in decentralized FL by reconstructing data even from model update differences. A common approach to overcome this issue is to use the diffusion process following differential privacy (DP), i.e., message passing between nodes is hidden by noise. However, this often makes the learning process unstable, leading to degraded results compared to without using DP diffusion process. In this paper, we propose a primal-dual DP algorithm with denoising normalization (DP-Norm) for less sensitivity to noise/interference, such as DP diffusion and heterogeneous data allocation. For DP-Norm, privacy analysis to determine minimal noise level and convergence analysis are conducted. Through image classification benchmark tests, we confirmed that DP-Norm performed close to the single-node reference score, even when statistically heterogeneous data was allocated on six nodes.**

***Index Terms*— Federated learning, differential privacy, data heterogeneity, primal-dual optimization.**

I. Introduction

FEDERATED learning (FL) is an emerging distributed learning for model training without data aggregation. It is in high demand for privacy-preserving applications, such as medical data analysis in hospitals, anomaly prediction in factories, and speech mining in call centers [1], [2], [3].

Recent trends in FL include (a) decentralized flexible network (NW) architectures (peer-to-peer nodes) instead of centralized ones (clients and a server) e.g., [4], [5], [6], and [7], (b) statistical data bias settings, i.e., each local node holds heterogeneous (non-independent and identically distributed: non-IID) data subsets, e.g., [6] and [8], and (c) information privacy in FL message passing among local nodes. Several studies [9], [10], [11], [12] have reported that updated differences in model variables inherit the statistical properties of data sets, i.e., the node's confidential data itself can be inferred under several assumptions. The goal of this work is to formulate a new FL algorithm that runs on (a) a decentralized NW with (b) a non-IID data allocation and (c) high information privacy, aiming to reach the performance of a reference model trained on a single node by using all aggregated data. Since several FL algorithms such as Edge Consensus Learning (ECL) [8], [13] have already achieved stable model learning on (a) a distributed NWs with (b) non-IID data allocation, we aim for a high privacy new FL algorithm based on the ECL algorithm.

Several studies have introduced differential privacy (DP) in decentralized FL to overcome this privacy issue, i.e., scaled Gaussian noise is added to message passing information to reduce the risks of local samples being leaked. Hereafter, this noise addition procedure is called a *DP diffusion process*. In Differentially Private Stochastic Gradient Descent (DP-SGD) [14], Gaussian noise with a particular variance is added to a local model to hide it. Although DP-SGD is originally used for privacy-preserving single-node model training, it can be easily applied to decentralized FL. Similarly, a DP diffusion process is introduced into the local model variables exchanged between nodes in the update rule of the alternating direction method of multipliers (ADMM) [15], which is referred to as DP-ADMM [16]. However, introducing the DP diffusion process will degrade learning results compared to without using it. This learning issue is caused by noise/interference due to the DP diffusion process and non-IID data allocation.

To overcome this problem, we propose a primal-dual DP algorithm with denoising normalization, referred to as DP-Norm. First, we introduce a DP diffusion process into ECL as linear constraints regarding model variables that make it robust to non-IID data allocation. Then, message passing will be performed to exchange diffused dual variables to satisfy constraints. In our pre-testing, the effect of noise/interference due to the DP diffusion process and non-IID data allocation was mitigated; however, the learning stagnation issue remained due to an explosive norm increase through message passing of dual variables among local nodes. To reduce this explosive norm increase, *denoising process* is introduced in our problem formulation. Note that even when a denoising process is applied, privacy-preserving message passing following DP is guaranteed since statistically independent diffusion noise is added on each local node.

Our contributions are summarized as follows:

A. (C1) DP-Norm Algorithm Formulation (Sec. IV)

A cost function used in ECL is reformulated to introduce (i) a diffusion process for diffused message passing following

Manuscript received 7 February 2023; revised 16 October 2023 and 18 March 2024; accepted 5 April 2024. Date of publication 18 April 2024; date of current version 29 May 2024. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Albert Levi. *(Corresponding author: Takumi Fukami.)*

Takumi Fukami and Iifan Tyou are with NTT Social Information Laboratories, NTT Corporation, Yokosuka 239-0847, Japan (e-mail: takumi.fukami.as@hco.ntt.co.jp).

Tomoya Murata is with NTT DATA Mathematical Systems Inc., Tokyo 160-0016, Japan.

Kenta Niwa is with NTT Communication Science Laboratories, NTT Corporation, Yokosuka 239-0847, Japan.

This article has supplementary downloadable material available at https://doi.org/10.1109/TIFS.2024.3390993, provided by the authors.

Digital Object Identifier 10.1109/TIFS.2024.3390993

DP and (ii) a denoising process to reduce the explosive norm increase in model variables due to DP diffusion. By applying Peaceman-Rachford splitting [17], we formulate an algorithm to alternatingly repeat (i) diffused message passing of dual variables and (ii) local model update with denoising.

B. (C2) Theoretical Analyses of DP-Norm (Sec. V)

Two analyses associated with DP-Norm are conducted: privacy analysis and convergence analysis. In the privacy analysis for convex and non-convex cost functions, we derive the minimum noise level for DP-Norm's message passing to exchange diffused dual variables given a privacy level. Meanwhile, only the convergence analysis is performed by restricting the cost functions that are convex. Then, our method is proven to achieve the best-known utility bound of decentralized primal-dual algorithms obtained by [16] and that primal model and even dual variables reach their optimal solutions without any of the boundedness assumptions used in e.g., DP-ADMM [16], [18]. This indicates that the norm increase issue regarding both primal model and dual variables is resolved by our DP-Norm including the denoising process.

C. (C3) Experimental Validations (Sec. VI)

We experimentally evaluated our DP-Norm in image classification benchmark tests. The explosive norm increase in message passing dual variables was observed to be reduced. Due to this, resulting scores with DP-Norm approached those of single-node reference under realistic noisy/interference situations (high privacy level, non-IID data allocation), whereas previous DP-SGD and DP-ADMM degraded in their learning processes.

II. Preliminary

A. Problem Settings

Symbols and notations used throughout this paper are briefly summarized in Table I. Decentralized NW is drawn by a graph $G(\mathcal{N}, \mathcal{E})$, where $N := |\mathcal{N}|$ nodes and $E := |\mathcal{E}|$ edges exist and the i-th node is connected to neighboring of $E_i := |\mathcal{E}_i|$ nodes. Assuming that the computing power of N nodes is approximately identical, each local node updates the local model variable $\mathbf{w}_i$ for K inner-loop iterations in an outer-loop round $r \in \{0, \ldots, R-1\}$. Message passing with neighboring nodes is possibly asynchronously performed once per outer-loop round at random timing. Each node is allowed to access different data subsets $\mathbf{x}_i$ consisting of d_i data samples (non-IID data allocation is assumed). For simple notation, N node stacked models and data subsets are denoted by $\mathbf{w} := \{\mathbf{w}_i\}_{i\in\mathcal{N}}$ and $\mathbf{x} := \{\mathbf{x}_i\}_{i\in\mathcal{N}}$. Our goal is to search for a set of model variables such that minimizes $f(\mathbf{w}, \mathbf{x}) = \frac{1}{N}\sum_{i\in\mathcal{N}} f_i(\mathbf{w}_i, \mathbf{x}_i)$ while satisfying $\mathbf{w}_1 =, \ldots, = \mathbf{w}_N$ (consensus model). Although f_i is assumed to be non-convex cost functions (e.g., DNN), only convergence analysis in Subsec. V-C is performed by restricting f_i to be convex.[1] In model updating, we compute stochastic gradient $\nabla f_i(\mathbf{w}_i, \boldsymbol{\xi}_i)$ instead of full gradient $\nabla f_i(\mathbf{w}_i, \mathbf{x}_i)$, where B mini-batch samples are randomly chosen from a local data subset as $\boldsymbol{\xi}_i \sim \mathbf{x}_i$. An effective strategy to obtain a consensus model is to impose *linear constraints* to model variables. We denote the linear constraints following the ECL [8], [13] as $\mathbf{A}_{i|j}\mathbf{w}_i + \mathbf{A}_{j|i}\mathbf{w}_j = \mathbf{0}$, where $\{\mathbf{A}_{i|j}, \mathbf{A}_{j|i}\} = \{\mathbf{I}, -\mathbf{I}\}$. For simple notation, the constraint parameters in Table I are stacked by a diagonal matrix as $\mathbf{A} = \mathrm{diag}[\mathbf{A}_1, \ldots, \mathbf{A}_N]$. To solve a constrained cost minimization problem in Sec. IV, dual variables are introduced as $\boldsymbol{\lambda}_{i|j}$ $(j \in \mathcal{E}_i)$ where it is *lifted* so each local node can update it asynchronously. The lifted dual variables in Table I can be stacked by $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1^\mathsf{T}, \ldots, \boldsymbol{\lambda}_N^\mathsf{T}]^\mathsf{T}$.

[1] Experimental results for the convex cost functions and those for the non-convex cost functions are summarized in Sec. VI and the supplementary material, respectively.

B. (ε, δ)-Differential Privacy (DP) [19]

There are potential risks of local samples being leaked from the message passing information or local sample memberships being inferred. DP is a concept that quantitatively evaluates these risks, and the DP mechanism is commonly used to guarantee privacy. For an algorithm $\mathcal{M}$ and two adjacent data subsets $\{\mathbf{x}_i, \mathbf{x}_i'\}$ where one data sample is different from the other, (ε, δ)-DP can be guaranteed when $\mathcal{M}$ satisfies

$$\Pr[\mathcal{M}(\mathbf{x}_i)] \leq e^{\varepsilon}\Pr[\mathcal{M}(\mathbf{x}_i')] + \delta, \tag{1}$$

where $\varepsilon > 0$ denotes the distinguishable bound of all outputs on two adjacent data subsets and $\delta \in (0, 1)$ represents the probability of information leakage. When the algorithm $\mathcal{M}$ can choose a small value for (ε, δ), it is a highly-secured algorithm with less risk of local data leakage. To guarantee a given privacy level (ε, δ), variance-scaled Gaussian noise $\mathbf{n}_i \sim \mathrm{Norm}(\mathbf{0}, \sigma_i^2\mathbf{I})$ is generally added to an existing algorithm F as $\mathcal{M}(\mathbf{x}_i) := F(\mathbf{x}_i) + \mathbf{n}_i$. We call this noise addition procedure *DP diffusion process*. Note that the minimal noise level σ_i^2 given privacy level (ε, δ) is dependent for each algorithm $\mathcal{M}$ since message passing information is different for each algorithm. Hence, our problem setting includes both (i) a formulation of a decentralized FL algorithm $\mathcal{M}$ including a DP diffusion process (see Sec. IV) and (ii) analysis to estimate the minimal noise level σ_i^2 (See Subsec. V-B).

III. Prior Works

As for decentralized FL including the DP diffusion process, DP-SGD (applied for decentralized FL) and DP-ADMM are explained in Subsec. III-A. In Subsec. III-B, ECL, to which DP has not applied before, is introduced as a preliminary step for our proposed method formulation in Sec. IV.

A. Decentralized FL Algorithms With DP

Two methods using (ε, δ)-DP for decentralized FL are explained. The first method, DP-SGD [14], was originally proposed for privacy-preserving single-node model training. Since it can be used for decentralized FL by combining a DP diffusion process with DSGD [5], we call this simply DP-SGD. The second method, DP-ADMM [16], applies a DP diffusion process to the ADMM based decentralized update rule. Although the update rules of these methods are summarized in the supplemental materials, their privacy-preserving message passing consists of diffused primal model

TABLE I
SYMBOL DEFINITION OF LATENT VARIABLES AND PARAMETERS

Symbol	Description	Element and dimension
$G(\mathcal{N},\mathcal{E})$	Topology of a decentralized network	$\{1,\ldots,N\}\in\mathcal{N}, \{j\in\mathcal{N}\|(i,j)\in\mathcal{E}\}\in\mathcal{E}_i$
$\mathbf{x}_i$	Data subset on a local node	$\mathbf{x}_i\in\mathbb{R}^{q\times d_i}$
$\mathbf{w}_i$	Primal model variables on a local node	$\mathbf{w}_i\in\mathbb{R}^p$
$\boldsymbol{\lambda}_i$	Dual variables on a local node	$\boldsymbol{\lambda}_i=[\boldsymbol{\lambda}_{i\|1}^\mathsf{T},\ldots,\boldsymbol{\lambda}_{i\|i-1}^\mathsf{T},\boldsymbol{\lambda}_{i\|i+1}^\mathsf{T},\ldots,\boldsymbol{\lambda}_{i\|N}^\mathsf{T}]^\mathsf{T}\in\mathbb{R}^{pE_i}$
$\mathbf{A}_i$	Constraint parameters on a local node	$\mathbf{A}_i=[\mathbf{A}_{i\|1}^\mathsf{T},\ldots,\mathbf{A}_{i\|i-1}^\mathsf{T},\mathbf{A}_{i\|i+1}^\mathsf{T},\ldots,\mathbf{A}_{i\|N}^\mathsf{T}]^\mathsf{T}\in\mathbb{R}^{pE_i\times p}$
$\mathbb{E}_{\mathbf{x}_i}[f_i(\mathbf{w}_i,\mathbf{x}_i)]$	Cost function on a local node	$\mathbb{E}_{\mathbf{x}_i}[f_i(\mathbf{w}_i,\mathbf{x}_i)]:\mathbb{R}^{p+q\times d}\mapsto\mathbb{R}$

variable transmission. In the local node update, a DP diffusion process to add noise $\mathbf{n}_i$ to $\mathbf{w}_i$ is performed for each inner iteration (K times per round), where noise level σ_i to guarantee (ε,δ)-DP is summarized in Sec. V.

However, it was experimentally shown that these methods result in degraded learning results in our assumed situations, i.e., non-IID data allocation and high privacy level approaching (ε,δ) to zero (experimental results are summarized in Sec. VI and the supplementary material). This will be caused by noise/interference due to a DP diffusion process and non-IID data allocation.

B. ECL for Decentralized FL (without DP)

To reduce the negative impact of non-IID data allocation, using ECL [8], [13] may be effective since robustness to it has been experimentally shown. As a preliminary step for our proposed method formulation in Sec. IV, we briefly explain ECL to which (ε,δ)-DP has not been applied previously.

In ECL, we will search for a set of primal model variables $\{\mathbf{w}_1,\ldots,\mathbf{w}_N\}$ that minimize the cost function $f_i(\mathbf{w}_i,\mathbf{x}_i)$ while linearly imposing consensus constraints to be $\mathbf{w}_1=,\ldots,=\mathbf{w}_N$ as

$$\inf_{\{\mathbf{w}_1,\ldots,\mathbf{w}_N\}}\frac{1}{N}\sum_{i\in\mathcal{N}}f_i(\mathbf{w}_i,\mathbf{x}_i),\quad \text{s.t. }\mathbf{A}_{i|j}\mathbf{w}_i+\mathbf{A}_{j|i}\mathbf{w}_j=\mathbf{0}\quad(i\in\mathcal{N},\,j\in\mathcal{E}_i).\tag{2}$$

To use primal-dual formalism for solving linearly constrained minimization problems, f_i is assumed to be convex in this section. When f_i is non-convex, it can be approximated by a quadratic function around the current variable as $f_i(\mathbf{w}_i,\mathbf{x}_i)\approx f_i(\mathbf{w}_i^r,\mathbf{x}_i)+\langle\nabla f_i(\mathbf{w}_i^r,\mathbf{x}_i),\mathbf{w}_i-\mathbf{w}_i^r\rangle+\frac{1}{2\mu}\|\mathbf{w}_i-\mathbf{w}_i^r\|^2$, where $\mu(>0)$ is assumed to be sufficiently small.

To solve (2), the Lagrange function is formulated. To make an update rule work in a decentralized communication manner, Lagrange multipliers (dual variables) are lifted as $\{\boldsymbol{\lambda}_{i|j}|i\in\mathcal{V},j\in\mathcal{E}_i\}\in\boldsymbol{\lambda}$, as performed in primal-dual method of multipliers (PDMM) [20], [21] and ECL. The cost function in them can be reformulated by the sum of two functions, namely the convex conjugate function $f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda},\mathbf{x})=\sup_{\mathbf{w}}\langle\boldsymbol{\lambda},\mathbf{A}\mathbf{w}\rangle-f(\mathbf{w},\mathbf{x})$ and the indicator function $\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda})$, to make it consistent with the original Lagrange function even when using dual variables $\boldsymbol{\lambda}$, as

$$\inf_{\boldsymbol{\lambda}}f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda},\mathbf{x})+\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda}),\quad \text{where}\quad \iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda})=\begin{cases}0 & (\boldsymbol{\lambda}=\mathbf{P}_G\boldsymbol{\lambda})\\ +\infty & (\text{otherwise}),\end{cases}\tag{3}$$

and $\mathbf{P}_G$ denotes the permutation matrix to exchange dual variables over a graph $G(\mathcal{N},\mathcal{E})$ as $\boldsymbol{\lambda}_{i|j}\rightleftharpoons\boldsymbol{\lambda}_{j|i}$.

Summarizing this subsection, a constrained minimization problem (2) is reformulated as a minimization problem (3) using dual variables. Since consensus constraints are imposed in (2), gradient drift due to non-IID data allocation is expected to be reduced. Then, message passing consists of dual variable exchange over a graph as $\boldsymbol{\lambda}_{i|j}\rightleftharpoons\boldsymbol{\lambda}_{j|i}$. Message passing using dual variables $\{\boldsymbol{\lambda}_{i|j}\}$ is much more confidentially safer than using the primal model itself $\{\mathbf{w}_i\}$. However, the risk of information leaking remains since the statistical nature of the data subsets is reflected. In the next section, we will start by introducing a DP diffusion process to ECL, which is expected to be robust to non-IID data allocation.

IV. PROPOSED ALGORITHM

We propose DP-Norm for highly privacy-preserving FL with less sensitivity to interference due to DP diffusion process and non-IID data allocation. We start by introducing DP to ECL in Subsec. IV-A. To make the proposed algorithm (DP-Norm) robust to the DP diffusion process, a denoising process is added. The update rule for DP-Norm is given in Subsec. IV-B.

A. Cost Formulation for DP-Norm

To reduce information leakage from message passing in ECL, a DP diffusion process is introduced. Motivated by adding Gaussian noise $\mathbf{n}:=\{\mathbf{n}_i\}_{i\in\mathcal{N}}$ where $\mathbf{n}_i\sim\text{Norm}(\mathbf{0},\sigma_i^2\mathbf{I})$ to message passing dual variable $\boldsymbol{\lambda}_{i|j}$, a natural cost formalism is to add an expectation term $\mathbb{E}_{\mathbf{n}}[\langle\mathbf{A}^\mathsf{T}\boldsymbol{\lambda},\mathbf{n}\rangle]$ to (3) as

$$\inf_{\boldsymbol{\lambda}}f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda},\mathbf{x})+\mathbb{E}_{\mathbf{n}}[\langle\mathbf{A}^\mathsf{T}\boldsymbol{\lambda},\mathbf{n}\rangle]+\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda}).\tag{4}$$

Although the stationary point in (4) will not be affected by introducing a second diffusion term since $\mathbb{E}_{\mathbf{n}}[\mathbf{n}]=\mathbf{0}$, we can naturally formulate a DP primal-dual algorithm with respect to both data sample and Gaussian noise (the update rule is illustrated in Subsec. IV-B). However, in our pre-testing to investigate the performance of the derived algorithm to include Gaussian noise sampling, the learning stagnation issue was not completely resolved. Although experimental results are shown in Sec. VI, evaluation scores of learned models were then obviously degraded from that without using a DP diffusion process ($\sigma_i=0$). This degradation can be attributed to the explosive increase in the norm of the dual variables $\boldsymbol{\lambda}$ due to recursive diffusion noise addition (see Sec. VI).

Algorithm 1 DP-Norm for Decentralized FL. It Is Reduced to Previous ECL When Setting $\alpha=0$, $\sigma_i=0$.

1: ▷ Set $\mathbf{w}_i^{0,0}=\mathbf{w}_j^{0,0}(\sim$ Norm), $\boldsymbol{\lambda}_{i|j}^{0,0}=\mathbf{z}_{i|j}^{0}=\mathbf{0}$, μ, $\eta_i=1/(\mu E_i K)$, $\gamma_i=1+\alpha\eta_i$
2: ▷ Set privacy level (ε,δ)
3: ▷ Set noise level σ_i to follow Theorem 2.
4: **for each** $r\in\{0,\ldots,R-1\}$ (Outer-loop round) **do**
5: **for each** $i\in\mathcal{N}$ **do**
6: $\mathbf{n}_i\sim\text{Norm}(\mathbf{0},\sigma_i^2\mathbf{I})$ (Noise sampling).
7: **for each** $k\in\{0,\ldots,K-1\}$ (Inner-loop iteration) **do**
8: ▷ Update local primal model and dual variables
9: $\boldsymbol{\xi}_i\sim\mathbf{x}_i$ (Mini-batch data sampling).
10: $g_i(\mathbf{w}_i^{r,k})\leftarrow\nabla f_i(\mathbf{w}_i^{r,k},\boldsymbol{\xi}_i^{r,k})$.
11: $\mathbf{w}_i^{r,k+1}\leftarrow\frac{\gamma_i}{\gamma_i+\eta_i\mu E_i}\Big(\mathbf{w}_i^{r,k}-\mu g_i(\mathbf{w}_i^{r,k})+\frac{\mu\eta_i}{\gamma_i}\sum_{j\in\mathcal{E}_i}\mathbf{A}_{i|j}^{\top}\mathbf{z}_{i|j}^{r}\Big)$.
12: **for each** $j\in\mathcal{E}_i$ **do**
13: $\boldsymbol{\lambda}_{i|j}^{r,k+1}\leftarrow\frac{\eta_i}{\gamma_i}\{(\mathbf{z}_{i|j}^{r}-\mathbf{A}_{i|j}(\mathbf{w}_i^{r,k+1}+\mathbf{n}_i)\}$.
14: $\mathbf{y}_{i|j}^{r,k+1}\leftarrow\frac{2}{\eta_i}\boldsymbol{\lambda}_{i|j}^{r,k+1}-\mathbf{z}_{i|j}^{r}$.
15: ▷ Message passing with j-th node
16: **for each** $j\in\mathcal{E}_i$ (at random time) **do**
17: **Transmit**$_{j\to i}(\mathbf{y}_{j|i}^{r,k+1})$.
18: $\mathbf{z}_{i|j}^{r}\leftarrow\mathbf{y}_{j|i}^{r,k+1}$.
19: $\mathbf{w}_i^{r+1,0}\leftarrow\mathbf{w}_i^{r,K}$, $\mathbf{z}_{i|j}^{r+1}\leftarrow\mathbf{z}_{i|j}^{r}$.

To reduce this issue, we introduce a denoising normalization term $\rho(\boldsymbol{\lambda})$ to (4). Then, the cost function formulation is given by

[DP-Norm = ECL (3) + DP diffusion process + Denoising process]

$$\inf_{\boldsymbol{\lambda}}\underbrace{f^{\star}(\mathbf{A}^{\top}\boldsymbol{\lambda},\mathbf{x})+\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda})}_{\text{ECL}}+\underbrace{\mathbb{E}_{\mathbf{n}}[\langle\mathbf{A}^{\top}\boldsymbol{\lambda},\mathbf{n}\rangle]}_{\text{DP diffussion process}}+\underbrace{\rho(\boldsymbol{\lambda})}_{\text{Denoising process}},\tag{5}$$

where $\rho(\boldsymbol{\lambda})$ has a role in relaxing the explosive increase in the norm of $\boldsymbol{\lambda}$. To implement this normalization term, we set $\rho(\boldsymbol{\lambda})=\frac{\alpha}{2}\|\boldsymbol{\lambda}\|^2$ with experimentally selected α ($\geq$ 0). By adding this normalization term, the stationary point in (5) will be moved from the original problem (3). Even with that disadvantage, the reduction of interference due to a DP diffusion process by introducing a denoising process is more beneficial, which will be illustrated through experiments in Sec. VI.

B. Update Rule Derivation of DP-Norm

An update rule of DP-Norm is derived to solve (5), where f_i is approximated by a quadratic function as explained in Sec. III-B. Since the cost function in (5) consists of the sum of the first three differentiable/smooth convex terms and the last non-differentiable/non-smooth indicator function. As a preliminary for update rule derivation, we define two operators, namely $T_1(\boldsymbol{\lambda})=\mathbf{A}\nabla f^{\star}(\mathbf{A}^{\top}\boldsymbol{\lambda},\mathbf{x})+\mathbb{E}_{\mathbf{n}}[\mathbf{An}]+\alpha\boldsymbol{\lambda}$ and $T_2(\boldsymbol{\lambda})=\partial\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda})$, where ∇ and ∂ denote the differential operator and the subdifferential operator for non-smooth functions, respectively, and $\mathbf{n}$ in T_1 is randomly sampled. The stationary point of (5) satisfies $\mathbf{0}\in T_1(\boldsymbol{\lambda})+T_2(\boldsymbol{\lambda})$, where we use the symbol $\in$ instead of $=$ because the subdifferential of the non-smooth function will include the set of values at discontinuous points.

When T_1 and T_2 are ill-matched, it is common to use operator splitting (e.g., [22], [23]) to derive the variable update rule. We use Peaceman-Rachford splitting [17]. Although detailed derivations are noted in the supplementary material, the update rule to solve (5) using the Peaceman-Rachford Splitting is summarized by Alg. 1, which alternatingly repeats (i) the local node procedure to update primal model variable $\mathbf{w}_i$ and associated dual variables $\{\mathbf{y}_{i|j}|j\in\mathcal{E}_i\}$ and (ii) the privacy-preserving message passing of dual variables over a graph (received dual variables are noted in $\{\mathbf{z}_{i|j}|j\in\mathcal{E}_i\}$). The denoising process is included in lines 9, 10, and 11 of Alg. 1. In addition, $\gamma_i=1+\alpha\eta_i$ is used for simple notation and $\eta_i=\frac{1}{\mu E_i K}$ is selected to make $\mathbf{w}_i$-update to follow stochastic variance reduction formalism [8].

Fig. 1 shows an illustration image of model update trajectory differences with (a) DP-SGD for decentralized FL, (b) DP-Norm without normalization ($\alpha=0$), and (c) DP-Norm ($\alpha>0$). In all methods, stochastic gradient (orange dot line) and diffusion noise (black line) are used in their update rule. In (a) DP-SGD for decentralized FL, non-IID data and DP diffusion at each node cause the update to face different directions and fail to approach the stationary point. In (b) DP-Norm without normalization ($\alpha=0$), gradient modification is added that uses dual variables (red line) originating from consensus constraints in (2). Then, gradient drift due to non-IID data allocation is expected to be reduced. Furthermore, with (c) DP-Norm ($\alpha>0$), denoising normalization (blue line) is added to avoid an explosive norm increase. Although this will move the stationary point from the original problem (3), the effect of DP diffusion process can be reduced and mitigated. It will result in making all node model variables approach the stationary point.

In the next section, we perform two analyses for DP-Norm: (i) privacy analysis to derive minimal noise level σ_i to guarantee (ε,δ)-DP and (ii) convergence analysis.

V. Algorithm Analyses

In this section, we analyze DP-Norm in Alg. 1 under several assumptions summarized in Subsec. V-A. Note that our analyses for DP-Norm are rigorously performed in synchronous communication settings (dual variables are exchanged at the beginning of each round). First, in Subsec. V-B, we specify the minimal noise level σ_i to guarantee (ε,δ)-DP. Second, in Subsec. V-C, convergence analysis is conducted to investigate the effect of DP diffusion/denoising processes. All proofs are summarized in the supplementary material. In Subsec. V-D, analysis results are briefly discussed.

A. Assumptions

Assumptions used throughout this section are summarized as follows:

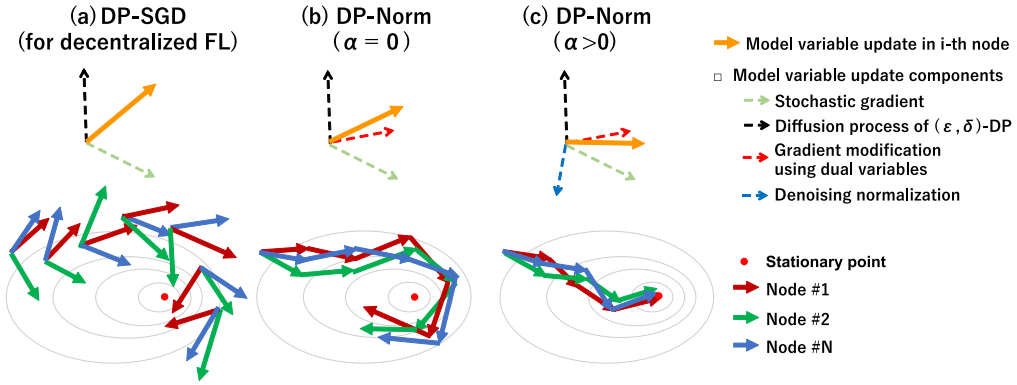


Fig. 1. Illustration image of model update trajectory with (a) DP-SGD for decentralized FL, (b) DP-Norm without normalization ($\alpha = 0$) and (c) DP-Norm ($\alpha > 0$).

TABLE II

SUMMARY OF PRIVACY AND CONVERGENCE ANALYSES. B: MINI-BATCH SIZE, d_i: DATA SUBSET SIZE, p: DIMENSION OF PRIMAL MODEL VARIABLE, R: OUTER-LOOP ROUNDS, K: INNER-LOOP ITERATIONS, c AND $J_{R,\varepsilon,\delta}$ ARE DEFINED IN THEOREM 1. ALL THE METHODS USE ASSUMPTIONS (A1) AND (A2) FOR PRIVACY ANALYSIS. MEANWHILE, ASSUMPTIONS (A1)-(A3) ARE USED FOR CONVERGENCE RATE ANALYSIS FOR CONVEX COST FUNCTIONS. SAMPLING WITHOUT REPLACEMENT MEANS THAT EACH LOCAL NODE DOES NOT USE A DATA SAMPLE OF THE LOCAL SUBSET MORE THAN ONCE IN THE MINI-BATCH. RANDOM PERMUTATION MEANS THAT EACH LOCAL NODE CYCLES THROUGH ITS OWN LOCAL DATA SUBSETS IN THE ORDER DEFINED BY A PERMUTATION THAT IS RANDOMLY GENERATED AT THE BEGINNING OF EACH ROUND r. $\widetilde{\mathcal{O}}$ IS USED TO ABBREVIATE THE LOGARITHMIC FACTOR FOR SIMPLE PRESENTATION

Methods	Privacy analysis		
	DP Diffusion style	Data sampling	Minimal noise level to guarantee (ε, δ)-DP
DP-SGD[2] [14]	Add $\mathbf{n}_i$ to update $\mathbf{w}_i$ for each inner-loop iteration (=K times per round)	Sampling without replacement	$\sigma_i \geq \frac{\sqrt{KR\log(1/\delta)}\mu}{\varepsilon}\frac{1}{d_i}$
DP-ADMM [16]	Add $\mathbf{n}_i$ to update $\mathbf{w}_i$ for each inner-loop iteration (=K times per round)	Sampling without replacement	$\sigma_i \geq \frac{2G}{(2\rho E_i+\eta_i^{k,r})}\frac{\sqrt{KR\cdot 2\log(1.25/\delta)}}{\varepsilon}\frac{1}{d_i}$ $(\eta_i^{k,r} = \frac{\sqrt{2kr}}{D}\sqrt{\frac{G^2}{N^2}+\frac{pG^2KR8\log(1.25/\delta)}{d_i^2\varepsilon^2}})$
DP-Norm	Add $\mathbf{n}_i$ to update $\boldsymbol{\lambda}_{i\|j}$ for each outer-loop round	Random permutation	$\sigma_i \geq J_{R,\varepsilon,\delta} \times 2c\mu G(\frac{K}{d_i}+\frac{1}{B})$
Methods	Convergence rate analysis (for convex cost functions)		
	Additional assumption to A1-A3		Convergence rate
DP-SGD [14]	(Not analyzed for decentralized FL)		
DP-ADMM [16]	Boundedness of primal and dual variables		$\widetilde{\mathcal{O}}(\frac{\beta DE}{NR}+\frac{D}{N\sqrt{KR}}+\frac{\sqrt{p}D}{d_{\min}\varepsilon})$ under $\|\mathbf{w}_i^r\| \leq D$ and $\|\boldsymbol{\lambda}_{i\|j}^r\| \leq \beta$
DP-Norm	No unique assumptions		$\widetilde{\mathcal{O}}(\frac{E}{NR}+\frac{1}{\sqrt{BKR}}+\frac{\sqrt{p}}{d_{\min}\varepsilon}+\frac{\sqrt{p}}{KB\varepsilon})$

(A1: G-Lipschitzness) $f_i(\cdot, \xi_i)$ is G-Lipschitz function, i.e., $\|f_i(\mathbf{a}, \boldsymbol{\xi}_i) - f_i(\mathbf{b}, \boldsymbol{\xi}_i)\| \leq G\|\mathbf{a} - \mathbf{b}\|$ for any i, $\mathbf{a}$, $\mathbf{b}$ and single data sample $\boldsymbol{\xi}_i \sim \mathbf{x}_i$.

(A2: L-smoothness) $\nabla f_i(\cdot, \boldsymbol{\xi}_i)$ is L-smooth, i.e., $\|\nabla f_i(\mathbf{a}, \boldsymbol{\xi}_i) - \nabla f_i(\mathbf{b}, \boldsymbol{\xi}_i)\| \leq L\|\mathbf{a} - \mathbf{b}\|$ for any i, $\mathbf{a}$, $\mathbf{b}$ and mini-batch data sample $\boldsymbol{\xi}_i \sim \mathbf{x}_i$.

(A3: Stochastic gradient bound) The variance of mini-batch stochastic gradient for each local node is bounded by ς_i^2, i.e., $\mathbb{E}_{\boldsymbol{\xi}_i \sim \mathbf{x}_i}[\|\nabla f_i(\mathbf{a}, \boldsymbol{\xi}_i) - \nabla f_i(\mathbf{a}, \mathbf{x}_i)\|^2] \leq \varsigma_i^2$ for any i and $\mathbf{a}$.

B. Privacy Analysis

The aim of this subsection is to derive the minimal noise level σ_i that guarantees (ε, δ)-DP. In privacy analysis, $f_i(\cdot)(:= f_i(\cdot, x_i))$ is allowed for both convex and non-convex. Let us denote that $\mathbf{w}_i(\mathbf{x}_i)$ is a function to generate primal model variables $\mathbf{w}_i$ from a local data subset $\mathbf{x}_i$ through a set of procedures in Alg. 1 without the DP diffusion process. First, we investigate L_2 sensitivity of primal model variables $\mathbf{w}_i(\mathbf{x}_i)$, i.e., upper bound of changes in primal model variables $\mathbf{w}_i(\mathbf{x}_i)$ for each adjacent data subset $\{\mathbf{x}_i, \mathbf{x}_i'\}$ is investigated. In the following lemma, mini-batch sampling is assumed to be without replacement for simple analysis.

Lemma 1 (L2 sensitivity of DP-Norm's message passing): Suppose that assumptions (A1), (A2) and random permutation hold, and $\mu \leq 1/(c_i K L)$, where $c := 1 + 2(\gamma + 1)$ and we set $\gamma := \gamma_i = \gamma_j$ for simplicity of notation. Conditioned on the previous outputs of $\mathbf{y}_{i\|j}(j \in \mathcal{E}_i)$, the randomness of the

mini-batch sampling of node i and all the randomness of the other nodes, L2 sensitivity of DP-Norm's primal model variables w_i^r at round r for adjacent data subsets $\{\mathbf{x}_i, \mathbf{x}_i'\}$ is bounded by

$$\Delta_2 := \max_{\{\mathbf{x}_i, \mathbf{x}_i'\}} \|\mathbf{w}_i^r(\mathbf{x}_i) - \mathbf{w}_i^r(\mathbf{x}_i')\| \leq 2c\mu \left(\frac{K}{d_i} + \frac{1}{B}\right) G.$$

Next, we investigate the minimal noise level σ_i that guarantees (ε, δ)-DP for overall R rounds using the moments account method [14], [24]. The result is summarized in the following theorem:

Theorem 1 (Minimal noise level to guarantee (ε, δ)-DP): Suppose that $\mu \leq 1/(cKL)$, where c is defined in the statement of Lemma 1. Under assumptions (A1), (A2), and random permutation. For any privacy level (ε, δ), noise level σ_i that guarantees (ε, δ)-DP in the DP-Norm is given by

$$\sigma_i \geq \max\left\{\frac{\sqrt{2hR\log\left(e + \frac{\varepsilon}{\sqrt{2h}\delta}\right)}}{\varepsilon}, \sqrt{\frac{R}{2\varepsilon}}\right\}\Delta_2 := J_{R,\varepsilon,\delta}\Delta_2,$$

where, e denotes Napier's constant, $h := (1 + \sqrt{1+\varepsilon/l(\varepsilon,\delta)})^2/4 < (1+\sqrt{1+\varepsilon})^2/4$ and $l(\varepsilon,\delta) := \log(e + (\sqrt{2}\varepsilon)/((1+\sqrt{1+\varepsilon})\delta))$.

Proof sketch of Theorem 1

Although detailed proofs are shown in the supplementary material, its proof sketch is shown here. In DP-Norm, an auxiliary variable $\mathbf{y}_{i|j}$, which is injected with Gaussian noise through a dual variable $\lambda_{i|j}$, is transmitted between connected nodes for each communication round (see line 17 in Alg. 1) to guarantee DP. In our algorithm formulation, we can see that the requirement of (ε, δ)-DP with respect to client i can be formulated as follows:

$$\Pr(\mathbf{y}_{i|\mathcal{E}_i}^{1:R} \in S | x_i, I_i^{1:R}, \mathrm{Rand}_{\mathcal{E}\setminus i}^{1:R}) - e^{\varepsilon}\Pr(\mathbf{y}_{i|\mathcal{E}_i}^{1:R} \in S | x_i', I_i^{1:R}, \mathrm{Rand}_{\mathcal{E}\setminus i}^{1:R}) \leq \delta, \quad (6)$$

for every S and adjacent datasets x_i and x_i', where $I_i^{1:r,1:K}$ denotes the mini-batch sampled data indices for $k \in [K]$ and $r \in [R]$, and $\mathrm{Rand}_{\mathcal{E}\setminus i}^{1:r-1}$ denotes all the randomness (of sampling indexes and noise) of nodes $\mathcal{E}_{\setminus i} := \mathcal{E} \setminus \{i\}$ at round r. We first give a DP guarantee for a single global update of DP-Norm based on the L2 sensitivity analysis of the message passing through the multiple local updates given in Lemma 1. Then, to find the smallest possible DP noise size σ_i satisfying (6), we adopt the tight analysis of the advanced composition theorem given in [24]. Following the proof strategy in Appendix B in [24], (6) can be reformulated as

$$\varepsilon \geq R\Delta_2^2/(2\sigma_i^2) + \sqrt{(2R\Delta_2^2/\sigma_i^2)\log(e + (1/\delta)\sqrt{R\Delta_2^2/\sigma_i^2})}, \quad (7)$$

where Δ_2 is the L2 sensitivity of the message passing through the multiple local updates given in Lemma 1.

Furthermore, for obtaining a tighter bound for σ_i than [24], we introduce a parameter $h \in (0, (1+\sqrt{1+\varepsilon})^2/4)$ and describe σ_i^2 as a function of h:

$$\sigma_i^2 := (2hR\Delta_2^2/\varepsilon^2)\log(e + \varepsilon/(\sqrt{2h}\delta)). \quad (8)$$

Finally, we optimize h to minimize σ_i satisfying the constraint (7) and obtain the expression of h and σ_i described in Theorem 2.

C. Convergence Rate Analysis

For only convergence rate analysis, the cost function is assumed to be restricted to be convex.[3] Our analysis strategy basically follows [25]. To measure the difference between current primal model variables $\mathbf{w}_i^r$ and its stationary point $\mathbf{w}_i^*$, Bregman divergence is defined as $D_{f_i}(\mathbf{w}_i^r, \mathbf{w}_i^*) := f_i(\mathbf{w}_i^r) - f_i(\mathbf{w}_i^*) - \langle \nabla f_i(\mathbf{w}_i^*), \mathbf{w}_i^r - \mathbf{w}_i^* \rangle$, where $f_i(\cdot) := f_i(\cdot, x_i)$. When f_i is strongly convex, $D_{f_i} = 0$ when only $\mathbf{w}_i^r = \mathbf{w}_i^*$. In this subsection, $E_i = E_j$ is assumed for simple presentation. The following theorem shows that the primal model and dual variables reach their stationary point $(\mathbf{w}_i^*, \boldsymbol{\lambda}_{i|j}^*)$ of (5) by investigating addition of $\sum_{i\in\mathcal{N}} D_{f_i}(\mathbf{w}_i^r, \mathbf{w}_i^*)$ and $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{E}_i} \|\boldsymbol{\lambda}_{i|j}^r - \boldsymbol{\lambda}_{i|j}^*\|^2$ with scaling. $\widetilde{\mathcal{O}}$ is used to abbreviate the logarithmic factor for simple presentation.

Theorem 2 (Convergence rate): Suppose that assumptions (A1)-(A3) in addition to f_i being convex hold. Let $\boldsymbol{\lambda}_{i|j}^0 = 0$ and assume $\|\mathbf{w}_i^0 - \mathbf{w}_i^*\|^2$, $\|\boldsymbol{\lambda}_{i|j}^0 - \boldsymbol{\lambda}_{i|j}^*\|^2$, G and L be $\mathcal{O}(1)$ and $\varsigma_i^2 = \mathcal{O}(1/B)$, and $\sqrt{\varepsilon} = \mathcal{O}(\varepsilon)$. Then, when $\alpha \leq \mathcal{O}(E/(NL))$, with an appropriate choice of μ and $\eta_i := 1/(\mu E_i K)$, it holds that

$$\frac{1}{N}\sum_{i\in\mathcal{N}}\mathbb{E}[D_{f_i}(\mathbf{w}_i^{\mathrm{out}}, \mathbf{w}_i^*)] + \frac{\alpha}{N}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{E}_i}\mathbb{E}\|\boldsymbol{\lambda}_{i|j}^{\mathrm{out}} - \boldsymbol{\lambda}_{i|j}^*\|^2 \leq \widetilde{\mathcal{O}}\left(\frac{1}{KR} + \frac{E}{NR} + \frac{1}{\sqrt{BKR}} + \frac{\sqrt{p}}{d_{\min}\varepsilon} + \frac{\sqrt{p}}{KB\varepsilon}\right). \quad (9)$$

where p represents the dimension of $\mathbf{w}_i$, $\mathbf{w}_i^{\mathrm{out}} := \frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=0}^{K-1}\mathbf{w}_i^{r,k}$, $\boldsymbol{\lambda}_{i|j}^{\mathrm{out}} := \frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\boldsymbol{\lambda}_{i|j}^{r,k}$ and $d_{\min} := \min_{i\in\mathcal{N}} d_i$[4].

D. Discussion

Table II summarizes the minimal noise levels for $(\varepsilon, \delta)-$DP guarantee and the convergence rates for three methods (DP-SGD, DP-ADMM, and DP-Norm).

Associated with Theorem 1, we found that DP-Norm can guarantee DP with less noise in total since the DP diffusion process to add noise on a dual variable is performed once an outer round as in Alg. 1, while DP-SGD and DP-ADMM perform it for every inner iteration (K times for a round). If $(\varepsilon, \delta)-$DP can be guaranteed even with a relatively low noise level, that is an advantage of our formulation starting from ECL. In that situation, the denoising process in DP-Norm will work well to further reduce the number of iterations of the DP diffusion process.

[2]Since outer-loop updates are not considered in the original DP-SGD since it is proposed for single-node model training, we changed its update rules to include outer-loop updates for use in decentralized FL, as shown in the supplementary material. Then, the noise level shown in [14] is changed from K to $K \times R$, and it results in the minimal noise level shown in Table II.

[3]Although the theoretical convergence analysis for non-convex functions is a future work, experimental verification to confirm convergence curves are shown in the supplemental material.

[4]Factor $1/d_{\min}$ can be improved to $((1/N)\sum_{i\in\mathcal{N}} 1/d_i^2)^{1/2}$, but we do not discuss this point here due to the space limitation. Details are noted in the supplementary material.

In Theorem 2, no boundedness assumptions for primal and dual variables are assumed. This is a critical difference from DP-ADMM [16] where the researchers essentially used the boundedness to derive the convergence rate. As mentioned in Subsec. IV-A and empirically observed in Subsec. VI-B, diffusion noise addition generally causes the explosive increase in the norm of the dual variables and the boundedness assumption does not hold. Our proposed method enjoys the standard utility $\widetilde{\mathcal{O}}(\sqrt{p}/(d_{\min}\varepsilon))$ (for $KB \geq d_{\min}$) without assuming the boundedness and guarantees the convergence of both primal model and even dual variables (only when $\alpha > 0$) thanks to DP-Norm formulation including a denoising normalization term.

Note that the stationary points by using DP-Norm are generally different from other methods (DP-SGD, DP-ADMM) since a denoising normalization term is added as in (5). Hence, the generalization ability of the optimized models should also be considered. Empirically, we found that the generalization ability of the optimized model of DP-Norm is always better than the one of DP-ADMM and DP-Norm ($\alpha = 0$). This point is experimentally investigated in Sec. VI.

VI. Numerical Experiments

Numerical experiments to compare our DP-Norm and prior works (DP-SGD, DP-ADMM) by using image classification benchmark tests with the convex logistic regression model.

A. Experimental Setup

1) Dataset and Accessibility/Model: We used Fashion MNIST [26] (28×28 pixels, 10 classes) as an image classification benchmark test. Then, each node has access to a non-IID data subset. In our implementation, each subset consists of $d_i = 4,000$ data with 6 classes randomly selected from 10 classes. Each data sample is normalized so that its L_2 norm is equal to 1. We prepared the convex logistic regression with squared L_2 regularizer with empirical weight and the non-convex ResNet-10 model [27]. For the convex model, $(G, L) = (1, 0.5)$ is theoretically selected following [28]. Also, $(G, L) = (1, 0.5)$ is empirically selected for the non-convex model. Since it is theoretically difficult to select (G, L) for a neural network model such as non-convex ResNet-10, which is commonly used in prior experiments with convex logistic regression model, we considered the most important point to use a common (G, L) among competing learning methods.

2) Network Graph/Communication: A ring topology network with $N = 6$ nodes is used ($E_i = 2$). Assuming that the computational and communication performances of all nodes are approximately equal, we allowed each node to asynchronously communicate with connected nodes once per outer-loop round at random timing. The pair of the number of outer and inner loops is empirically set for each model, i.e., $(R, K) = (2000, 10)$ is used for the convex model and $(R, K) = (1000, 10)$ is used for the non-convex model.

3) Comparing Methods: We evaluated five methods: (M1) DP-SGD, (M2) DP-ADMM, (M3) DP-Norm without normalization ($\alpha = 0$), i.e., ECL with DP diffusion process, (M4) DP-Norm ($\alpha > 0$), and (M5) single-node reference. The goal of decentralized algorithms (M1)-(M4) is to obtain primal model variables that lead to an accuracy approaching that of (M5) trained with all datasets and vanilla SGD without DP diffusion process. Our proposed method consists of (M3) and (M4). The learning rate μ is selected for each model such that it satisfies $\mu < \frac{1}{cKL}$ as defined in Lemma 1, namely $\mu = 0.03$ for the convex model and $\mu = 0.001$ for the non-convex model. In addition, α in (M4) is experimentally selected as $\alpha = 0.2$ for the convex model and $\alpha = 0.001$ for the non-convex model.

4) Privacy Level/Noise Level: Three privacy levels are prepared: non-private $\varepsilon = \infty$ and two privacy levels $\varepsilon = \{1, 0.5\}$, $\delta = 0.001$. Following Table II, the smallest noise level is selected.

5) Mini-Batch Size: In DP-SGD, DP-ADMM, and DP-Norm, stochastic fluctuation due to both stochastic gradient using mini-batch selection and DP diffusion process are included. Stochastic fluctuation that is too large will not reach an optimal solution due to unstable learning process. An implementation technique to relax this issue is to increase mini-batch size B, which will reduce the ς_i^2-stochastic gradient bound assumption in Subsec. V-A. We experimentally selected $B = 2,000$ for the convex model and $B = 500$ for the non-convex model.

B. Experimental Results

Fig. 2 shows node-averaged learning curves using test accuracy for the convex logistic regression with L_2 regularizer model and the non-convex ResNet-10 model given three privacy levels ($\varepsilon = \{\infty, 1, 0.5\}, \delta = 0.001$). Our (M4) DP-Norm ($\alpha > 0$) performed closest to the single-node reference scores in all settings. Although it could not reach single-node reference scores when $\varepsilon = \{1.0, 0.5\}$. When increasing the privacy level as $\varepsilon \to 0$, noise level σ_i is increased; thus, applying a denoising process with appropriate weight selection was effective, even when the stationary point was moved from the original problem. As discussed in Subsec. V-D, DP-Norm can guarantee (ε, δ)-DP with less noise σ_i in total than that in other methods (DP-SGD and DP-ADMM). In addition to this, applying the denoising process is effective in reducing DP diffusion noise.

Meanwhile, DP-SGD and DP-ADMM resulted in a degraded learning process when $\varepsilon = \{1.0, 0.5\}$. Although DP-ADMM and DP-Norm can be derived from a common primal-dual formalism, DP-ADMM resulted in degraded learning results compared to DP-Norm. This would be because DP-ADMM requires adding a diffusion process to the primal model variable K times, whereas DP-Norm requires adding a diffusion process to the dual variable only once, and the convergence of DP-ADMM depends on the norm of message passing variables, i.e., main and dual variables, as summarized in Table II. Although additional experimental results are shown in the supplementary material, the norm of message passing variables explosively increased. Due to non-IID data allocation, a part of the scores cannot reach those with DP-Norm even in non-private settings ($\varepsilon = \infty$).

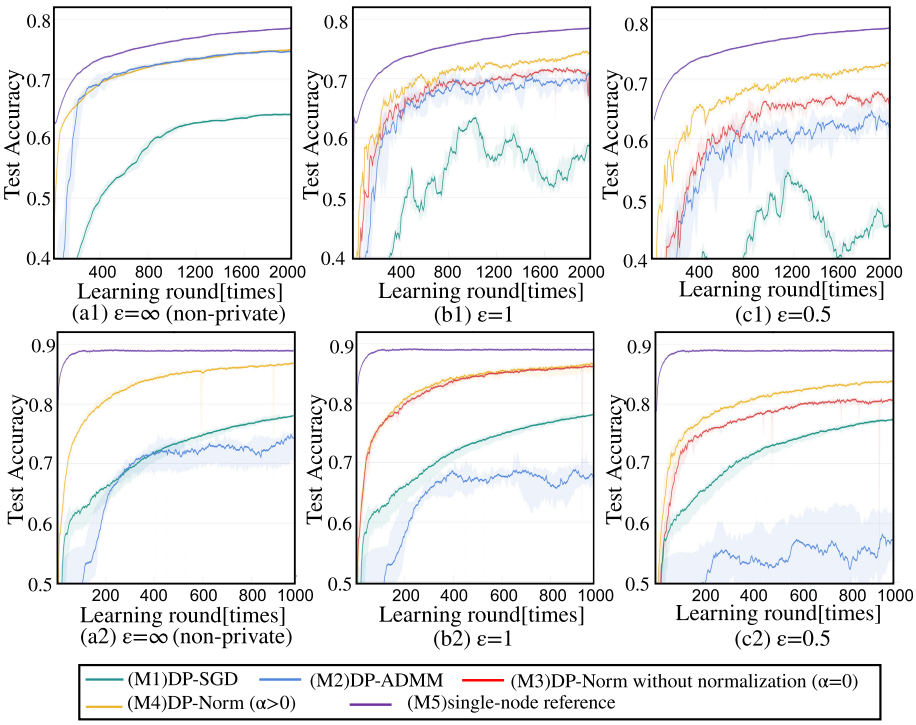


Fig. 2. The upper figure shows the learning curve for the convex logistic regression with L_2norm model, and the lower figure shows the learning curve for the non-convex ResNet-10 model, both using test accuracy given three privacy level $\varepsilon = \{\infty, 1, 0.5\}$, $\delta = 0.001$.

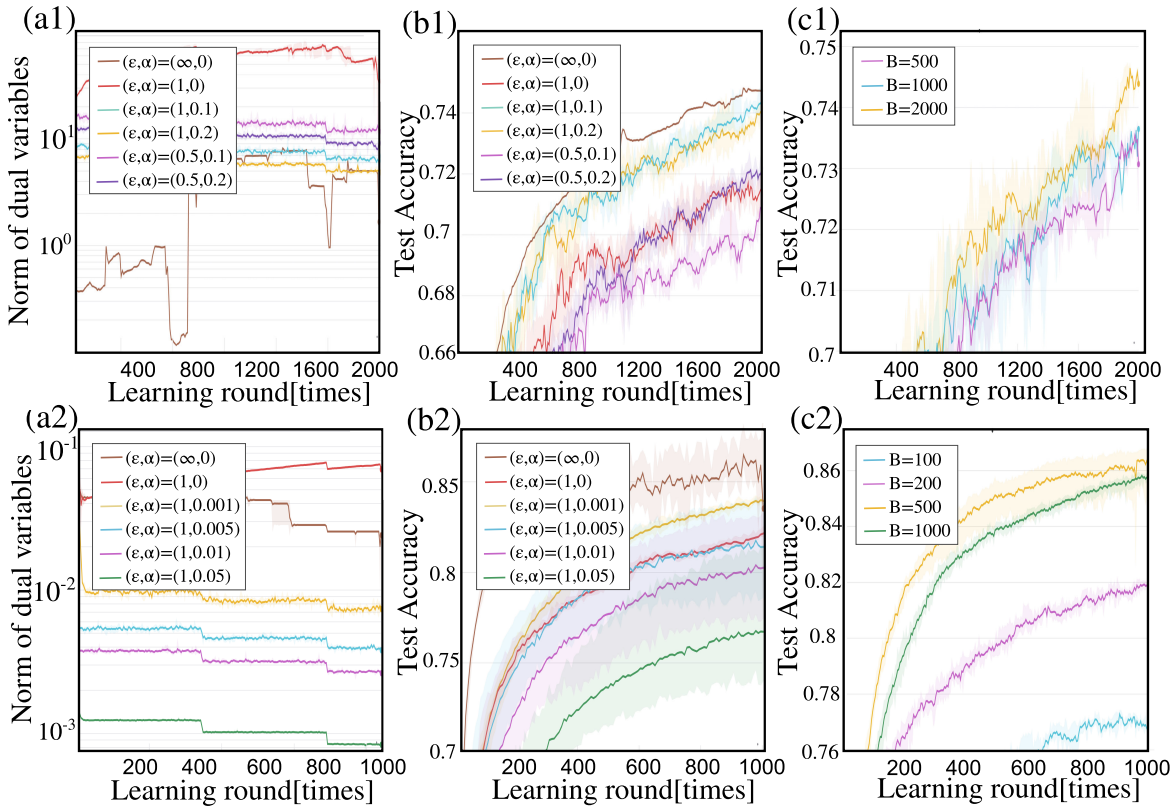


Fig. 3. Experiments to select hyperparameters in DP-Norm, normalization weight α and mini-batch size B, using (upper) the convex logistic regression with L_2 norm model and (lower) using the non-convex ResNet-10 model. (a) Relationship between α and averaged norm of dual variables, (b) relationship between α and test accuracy, and (c) relationship between B and test accuracy when $\varepsilon = 1$.

As shown in Fig. 3 (a), we experimentally investigated the relationship between α and the node-averaged norm of dual variables $\|\boldsymbol{\lambda}_{i|j}\|$. When increasing α, an explosive increase in the norm of dual variables was mitigated; however, too

large α degraded the test accuracy, as shown in Fig. 3 (b). This would be because information to make consensus among nodes is missing in the transmitted dual variable due to denoising normalization that is too large. In addition to this, we also investigated the relationship between mini-batch size B and the test accuracy. From experimental results in Fig. 3 (c), we selected $B = 2,000$ to make algorithms insensitive to the mini-batch data selection. Through experimental hyperparameter selection, results in Fig. 2 are summarized.

VII. Conclusion

We proposed a DP-Norm algorithm for privacy-preserving decentralized federated learning (FL) to guarantee (ε, δ)-DP. In DP-Norm, an alternating update rule consists of (i) diffused message passing and (ii) local model update using denoising normalization. Through theoretical analyses for the DP-Norm method, we derived the minimal noise level given the privacy level and convergence rate. (For only convergence analysis, the cost function is assumed to be convex.) Through numerical experiments using image classification benchmark tests, DP-Norm stably approaches single-node reference scores under non-independent and identically (non-IID) data allocation.

Appendix A
Additional Experimental Results

The code execution environment is noted in Subsec. A-A. In Subsec. A-B, the relationship between the norm of primal/dual variables and learning rounds is investigated. In Subsec. A-C-A-D, additional experiments associated with prior works, namely DP-SGD and DP-ADMM, are summarized. In Subsec.A-E, additional experiments with IID data allocation. In Subsec.A-F, additional experiments to investigate the effect of the number of nodes.

A. Code Execution Environment

1) Hardware Setting: Our experiments are performed on six servers (CPU: Intel Xeon Gold 5218 2.10GHz, GPU: NVIDIA GeForce 3080), which are connected with 100 Gb Ethernet.

2) Software Setting: We used PyTorch v1.6.0 + CUDA v10.1 and Gloo[5] for communication. For decentralized FL algorithm coding, we started from ECL's source code.[6]

B. Relationship Between the Norm of Primal/Dual Variables and Learning Rounds

Fig. 4 shows the node-averaged norm of the primal model variables $\|\mathbf{w}_i\|$ in the upper row (M1a)-(M4a), that of the dual variables $\|\boldsymbol{\lambda}_{i|j}\|$ in the middle row (M2b)-(M4b). In addition in the bottom row figures (M2c)-(M4c), the node-averaged normalized norm of dual variable, which is a linear conversion of middle row figures (M2b)-(M4b) to clearly show the norm increase compared with the case without DP diffusion process. For computing the normalized node-averaged norm of dual variable, the norm of the dual variables in the middle row (M2b)-(M4b) was normalized by the final learning round value without using DP diffusion process for each method to clearly show the norm increase compared with the case without DP diffusion process.

From Fig. 4, we experimentally found that the norm of both primal model variables and dual variables explosively increased by increasing privacy level ($\varepsilon \to 0$) for three methods, namely (M1) DP-SGD, (M2) DP-ADMM, and (M3) DP-Norm without normalization ($\alpha = 0$). In contrast, as shown in Fig. 4(M4a)-(M4b) for DP-Norm ($\alpha > 0$), the denoising process in the DP-Norm mitigated explosive norm increase compared with DP-Norm without normalization. It was observed that the normalized norm of the dual variables in DP-Norm was the smallest among these methods. This indicates that DP-Norm resolves the explosive norm increase issue observed in DP-SGD, DP-ADMM, and DP-Norm without normalization ($\alpha = 0$) due to DP diffusion process and validates the effectiveness of our denoising normalization process.

C. Additional Experiments to Search Learning Rate in DP-SGD Using Convex Model

Although we used a common learning rate among comparing methods in Subsec. VI-B, we performed additional experiments to investigate an appropriate learning rate for DP-SGD. Since the minimum noise level σ_i in DP-SGD depends on the learning rate μ as summarized in Table II, the relationship between μ and test accuracy needs to be investigated for each privacy level $\varepsilon = \{1, 0.5\}$ for the *convex* logistic regression with L_2 weight decay regularization model in DP-SGD.

Experimental results are summarized in Fig. 5. Fig. 2 compares the test accuracy of DP-SGD when $\mu = 0.03$ and other methods for $\varepsilon = \{\infty, 1, 0.5\}$, whereas DP-SGD has the highest test accuracy score when $\mu = 0.1$ for any privacy levels. However, the test accuracy of our DP-Norm in Fig. 2 exceeds 72% for any privacy levels, and the test accuracy of DP-SGD is lower than that of DP-Norm in all cases.

D. Additional Experiments to Search Hyperparameters in DP-ADMM Using Convex Model

We investigated the relationship between hyperparameters (ρ, μ) in DP-ADMM and test accuracy for the convex logistic regression with L_2 weight decay regularization model. Note that the minimal noise level σ_i in DP-ADMM depends on ρ and the learning rate μ as noted in Table II. Although we used a common learning rate among comparing methods in Subsec. VI-B, we performed additional experiments to investigate appropriate hyperparameters (ρ, μ) for DP-ADMM.

Fig. 6 shows the relationship between (ρ, μ) and the test accuracy in DP-ADMM for each privacy level $\varepsilon = 1, 0.5$. In Fig. 6 (a1) and (b1), ρ is varied by fixing $(\varepsilon, \mu) = (1, 0.01)$ in (a1) and $(\varepsilon, \mu) = (0.5, 0.01)$ in (b1), respectively. Meanwhile, μ is varied by fixing $(\varepsilon, \rho) = (1, 0.0001)$ in (a2) and $(\varepsilon, \rho) = (0.5, 0.00005)$ in (b2), respectively. To compare the learning curves of DP-ADMM and our DP-Norm, the

[5]https://pytorch.org/docs/stable/distributed.html
[6]https://github.com/nttcslab/ecl-isvr

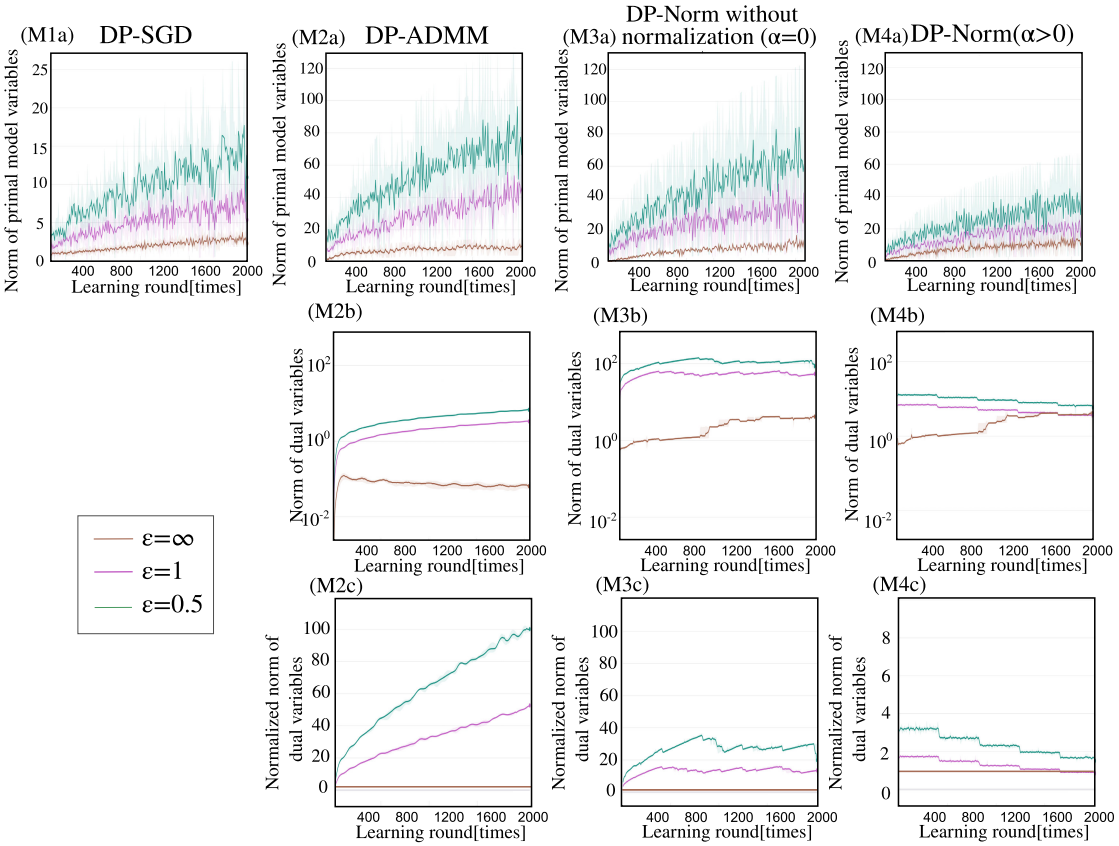


Fig. 4. The relationship between ε and norm of primal model/dual variables for the convex logistic regression with L_2 weight decay normalization model in (M1) DP-SGD, (M2) DP-ADMM, (M3) DP-Norm without normalization ($\alpha = 0$), and (M4) DP-Norm ($\alpha > 0$). The upper row figures (M1a)-(M4a) and middle row figures (M2b)-(M4b) compare the norm of the primal variables and the norm of the dual variables. In addition in the bottom row figures (M2c)-(M4c), the normalized norm of dual variable, which is a conversion of middle row figures (M2b)-(M4b) to clearly show the norm increase compared with the case without DP diffusion process. Then, the norm of the dual variables in the middle row was normalized by the final learning round value without using a DP diffusion process for each method.

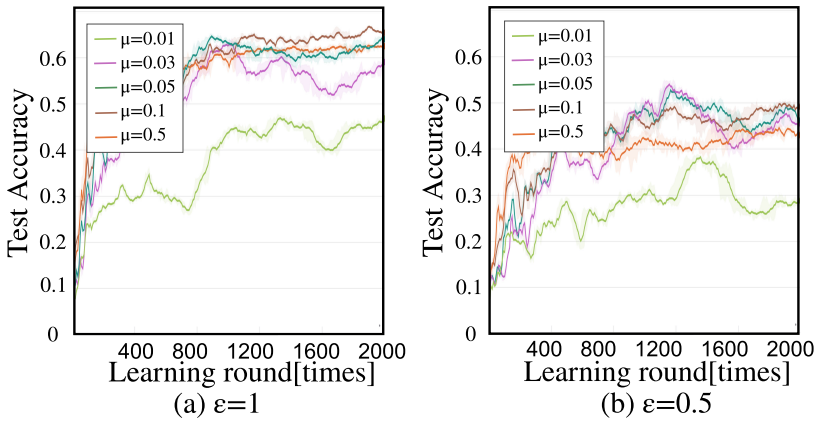


Fig. 5. The relationship between μ and test accuracy when (a) $\varepsilon = 1$ and (b) $\varepsilon = 0.5$ for the convex logistic regression with L_2 weight decay regularization model in DP-SGD.

learning curves of DP-Norm are shown in Fig. 2 also add to Fig. 6.

In Fig. 2, we compare the test accuracy of DP-ADMM when $(\rho, \mu) = (0.0001, 0.03)$ and other methods for $\varepsilon = \{\infty, 1, 0.5\}$, whereas DP-ADMM has the highest test accuracy score when $(\rho, \mu) = (0.0001, 0.01)$ for $\varepsilon = 1$ and $(\rho, \mu) = (0.00005, 0.01)$ for $\varepsilon = 0.5$. However, the test accuracy of DP-ADMM is lower than that of DP-Norm in all cases.

E. Additional Experiments With IID Data Allocation

We experimentally evaluated the performance of each method when each node has access to an IID data subset. In our implementation, each subset consists of $d_i = 4,000$ data, with each node having 10 classes of data uniformly. Fig. 7 shows node-averaged learning curves using test accuracy for the convex logistic regression with L_2 regularizer model and the non-convex ResNet-10 model

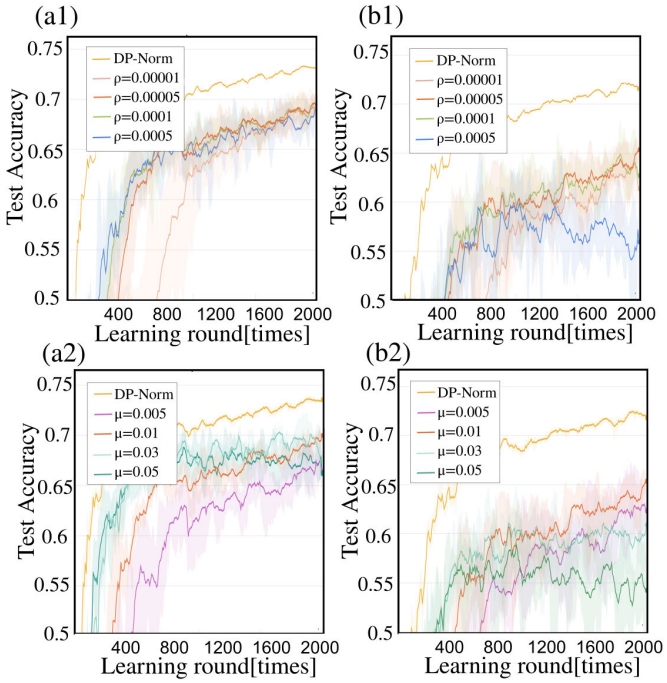


Fig. 6. (a) The relationship between ρ and test accuracy, and (b) the relationship between μ and test accuracy when (1) $\varepsilon = 1$ and (2) $\varepsilon = 0.5$ for the convex logistic regression with L_2 weight decay regularization norm model in DP-ADMM.

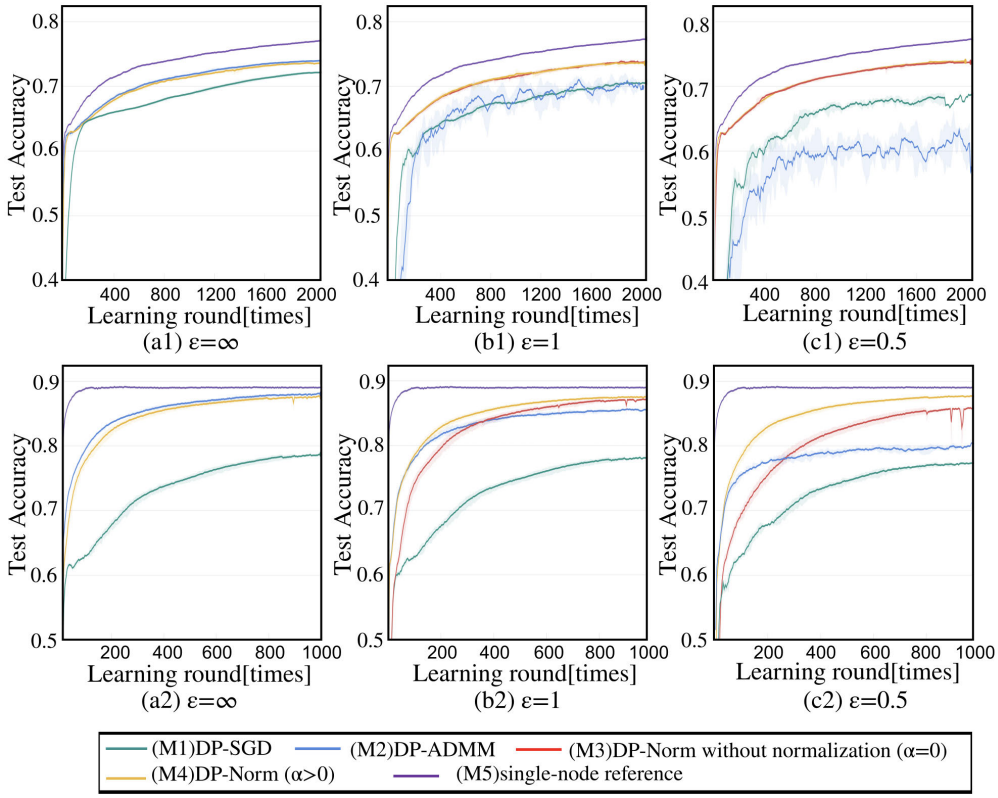


Fig. 7. The upper figure shows the learning curve for the convex logistic regression with L_2norm model, and the lower figure shows the learning curve for the non-convex ResNet-10 model, both using test accuracy given three privacy level $\varepsilon = \{\infty, 1, 0.5\}$, $\delta = 0.001$ with IID data allocation.

given three privacy levels ($\varepsilon = \infty, 1, 0.5, \delta = 0.001$). As with the non-IID data allocation noted in Subsec. VI-B, the performance of the proposed DP-Norm was closest to that of the reference with the IID data allocation. Compared with the non-IID allocation, the performance of the methods besides the DP-Norm was closer to that of the reference. On the other hand, the DP-SGD performed well only with the IID data allocation. This would be a natural result since the DP-SGD

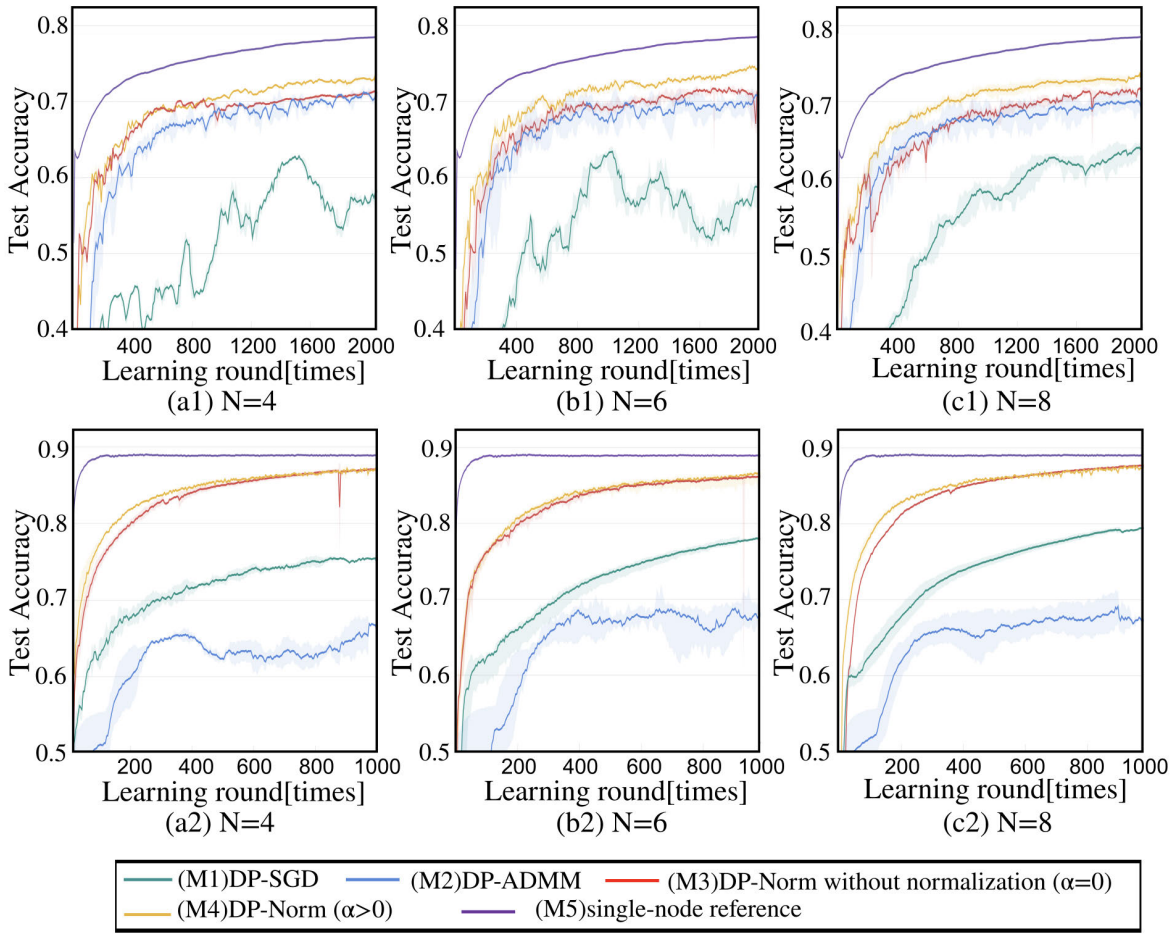


Fig. 8. The upper figure shows the learning curve for the convex logistic regression with L2norm model, and the lower figure shows the learning curve for the non-convex ResNet-10 model, both using the test accuracy given the number of nodes $N = \{4, 6, 8\}$ and the privacy level $(\varepsilon, \delta) = (1, 0.001)$ on the ring topology network.

does not have a term to modify gradient drift due to non-IID data allocations.

F. Additional Experiments to Investigate the Effect of the Number of Nodes

We experimentally investigated the effect of the number of nodes N. Firstly, we performed experiments in Sec. VI by changing the number of nodes was changed as $N = \{4, 6, 8\}$ over the ring topology while fixing the privacy level as $(\varepsilon, \delta) = (1, 0.001)$ and using the convex logistic regression with L_2 norm model and the non-convex ResNet-10 model. Fig. 8 shows the learning curves for five methods used in Sec. VI. When using (M4) DP-SGD ($\alpha > 0$), smoother learning curves were obtained by increasing the number of nodes, while the learning curves with other methods did not significantly change depending on the number of nodes for both convex and non-convex models. Our (M4) DP-Norm ($\alpha > 0$) performed closest to the single node reference score for $N = \{4, 6, 8\}$.

Secondly, we investigated the differences among learning curves with (M4) DP-Norm ($\alpha > 0$) for $N = \{3, 4, 6, 8\}$ over the ring topology. From Fig. 9, there was little difference in the convergence curves when the number of nodes was greater than 3 ($N = \{4, 6, 8\}$), but when $N = 3$, the convergence rate was a bit slower than the other cases. This can be theoretically explained from DP-Norm's convergence analysis (at least for the convex model). In fact, the second term of (9) in Theorem 2 is affected by $E/N = (N+1)/N$ since we used ring topology. Thus, the convergence speed would change depending on N and is expected to be slow for small N (e.g., $N = 3$). The empirical observations from Fig. 9 were consistent with this fact.

APPENDIX B
DP-NORM UPDATE RULE DERIVATION

DP-Norm update rule summarized in Alg. 1 is derived. Let us recall that the stationary point of (5) satisfies

$$\mathbf{0} \in T_1(\boldsymbol{\lambda}) + T_2(\boldsymbol{\lambda}), \tag{10}$$

where two operators are defined as

$$\begin{cases} T_1(\boldsymbol{\lambda}) = \mathbf{A}\nabla f^{\star}(\mathbf{A}^{\mathsf{T}}\boldsymbol{\lambda}, \mathbf{x}) + \mathbb{E}_{\mathbf{n}}[\mathbf{An}] + \alpha\boldsymbol{\lambda}, \\ T_2(\boldsymbol{\lambda}) = \partial\iota_{\ker(\mathbf{I}-\mathbf{P}_G)}(\boldsymbol{\lambda}). \end{cases} \tag{11}$$

Before reformulating (10) to be recurrent update rule, several operators associated with $\{T_1, T_2\}$ are introduced. The resolvent operators $\{R_1, R_2\}$ and the Cayley operators $\{C_1, C_2\}$

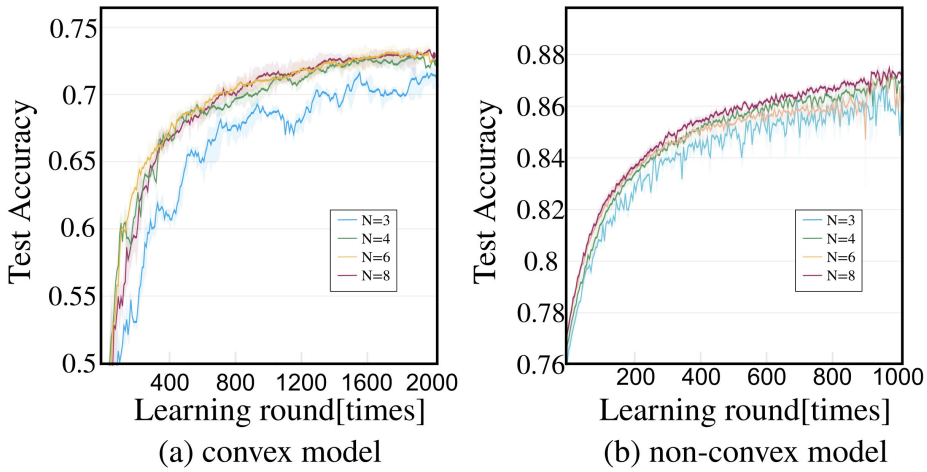


Fig. 9. (M4) DP-Norm ($\alpha > 0$)'s learning curve for (a) the convex logistic regression with L_2 norm model and (b) the non-convex ResNet-10 model using the test accuracy given the number of nodes $N = \{3, 4, 6, 8\}$ and the privacy level $(\varepsilon, \delta) = (1, 0.001)$ on the ring topology network.

are defined by

$$R_1 = (Id + \eta T_1)^{-1},$$
$$R_2 = (Id + \eta T_2)^{-1},$$
$$C_1 = (Id - \eta T_1)(Id + \eta T_1)^{-1} = 2R_1 - Id,$$
$$C_2 = (Id - \eta T_2)(Id + \eta T_2)^{-1} = 2R_2 - Id,$$

where Id is the identity operator, $^{-1}$ is the inverse operator (e.g., [23]), and η is the step-size for dual variable update.

A reformulation of (10) results in

$$\mathbf{0} \in (Id + \eta T_2)(\boldsymbol{\lambda}) - (Id - \eta T_1)(\boldsymbol{\lambda}). \tag{12}$$

Let an auxiliary variable $\mathbf{z}$ be associated with the lifted dual variable $\boldsymbol{\lambda}$ through the relationship $\boldsymbol{\lambda} \in R_{T_1}(\mathbf{z})$. Then, (12) can be written as

$$\mathbf{0} \in (Id + \eta T_2)R_1(\mathbf{z}) - C_1(\mathbf{z}),$$
$$\mathbf{0} \in R_1(\mathbf{z}) - R_2C_1(\mathbf{z}),$$
$$\mathbf{0} \in \tfrac{1}{2}(C_1 + Id)(\mathbf{z}) - \tfrac{1}{2}(C_2 + Id)C_1(\mathbf{z}),$$

which implies that the stationary point condition can be written as

$$\mathbf{z} \in C_2C_1(\mathbf{z}), \quad \boldsymbol{\lambda} \in R_1(\mathbf{z}), \text{ (Peaceman-Rachford Splitting)}. \tag{13}$$

This indicates that the dual variables are recursively updated through two different Cayley operators C_1 and C_2. (13) can be decomposed into the following procedure sets:

$$\boldsymbol{\lambda}^{r+1} = R_1(\mathbf{z}^r) = (Id + \eta T_1)^{-1}(\mathbf{z}^r), \tag{14}$$
$$\mathbf{y}^{r+1} = C_1(\mathbf{z}^r) = (2R_1 - Id)(\mathbf{z}^r) = 2\boldsymbol{\lambda}^{r+1} - \mathbf{z}^r, \tag{15}$$
$$\boldsymbol{\zeta}^{r+1} = R_2(\mathbf{y}^{r+1}) = (Id + \eta T_2)^{-1}(\mathbf{y}^{r+1}), \tag{16}$$
$$\mathbf{z}^{r+1} = C_2(\boldsymbol{\zeta}^{r+1}) = (2R_2 - Id)(\mathbf{y}^{r+1}) = 2\boldsymbol{\lambda}^{r+1} - \boldsymbol{\zeta}^{r+1}. \tag{17}$$

First, (14) is specified. (14) can be reformulated as

$$(Id + \eta T_1)(\boldsymbol{\lambda}) = \mathbf{z},$$
$$\mathbf{0} = \eta \mathbf{A} \nabla f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda}, \mathbf{x}) + \eta(\mathbb{E}_\mathbf{n}[\mathbf{An}] + \alpha\boldsymbol{\lambda}) + \boldsymbol{\lambda} - \mathbf{z}. \tag{18}$$

Since $\nabla f^\star = (\nabla f)^{-1}$ when f is restricted by convex, $\nabla f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda}, \mathbf{x})$ in (18) can be associated with primal model variable $\mathbf{w}$ as

$$\mathbf{w} = \nabla f^\star(\mathbf{A}^\mathsf{T}\boldsymbol{\lambda}, \mathbf{x}), \tag{19}$$
$$\nabla f(\mathbf{w}, \mathbf{x}) = \mathbf{A}^\mathsf{T}\boldsymbol{\lambda}. \tag{20}$$

Substituting (19) into (18) results in

$$\mathbf{0} = \eta \mathbf{Aw} + \eta(\mathbb{E}_\mathbf{n}[\mathbf{An}] + \alpha\boldsymbol{\lambda}) + \boldsymbol{\lambda} - \mathbf{z},$$
$$\mathbf{0} = \eta \mathbf{Aw} + (1 + \alpha\eta)\boldsymbol{\lambda} + \eta\mathbb{E}_\mathbf{n}[\mathbf{An}] - \mathbf{z},$$
$$\boldsymbol{\lambda} = \frac{\mathbf{z} - \eta\mathbf{A}(\mathbf{w} + \mathbb{E}_\mathbf{n}[\mathbf{n}])}{1 + \alpha\eta}. \tag{21}$$

To simplify notation, we introduce the scaled dual variables as $\hat{\mathbf{z}} = \frac{1}{\eta}\mathbf{z}$. Then, (21) is rewritten by

$$\boldsymbol{\lambda} = \frac{\eta(\hat{\mathbf{z}} - \mathbf{A}(\mathbf{w} + \mathbb{E}_\mathbf{n}[\mathbf{n}]))}{1 + \alpha\eta}. \tag{22}$$

By substituting (22) into (20) results in

$$\mathbf{0} = \nabla f(\mathbf{w}, \mathbf{x}) + \frac{\eta}{1 + \alpha\eta}\{\mathbf{A}^\mathsf{T}\mathbf{A}(\mathbf{w} + \mathbb{E}_n[\mathbf{n}]) - \mathbf{A}^\mathsf{T}\hat{\mathbf{z}}\}. \tag{23}$$

The integral of (23) results in $\mathbf{w}$-update formula as

$$\mathbf{w}^{r+1} = \arg\min_\mathbf{u}\Bigg(f(\mathbf{u}, \mathbf{x}) + \frac{\eta}{2(1 + \alpha\eta)}\|(\mathbf{A}(\mathbf{u} + \mathbb{E}_\mathbf{n}[\mathbf{n}]) - \hat{\mathbf{z}}^r)\|^2\Bigg). \tag{24}$$

This can be performed through (i) K inner loop iteration for around $(k = 0, \ldots, K-1)$ and (ii) mini-batch data sampling $\boldsymbol{\xi}_i^{r+1,k} \sim \mathbf{x}_i$ and noise sampling $\mathbf{n}_i^{r+1,k} \sim \text{Norm}(\mathbf{0}, \sigma^2\mathbf{I})$ as

$$\mathbf{w}^{r+1,k+1} = \arg\min_\mathbf{u}\Bigg(f(\mathbf{u}, \boldsymbol{\xi}^{r+1,k}) + \frac{\eta}{2(1 + \alpha\eta)}\|(\mathbf{A}(\mathbf{u} + \mathbf{n}^{r+1,k}) - \hat{\mathbf{z}}^r)\|^2\Bigg),$$

where $\mathbf{w}^{r+1,0} = \mathbf{w}^{r,K}$ and $\mathbf{w}^{r+1} = \mathbf{w}^{r+1,K}$. Similar to this, $\boldsymbol{\lambda}$-update rule is given by

$$\boldsymbol{\lambda}^{r+1} = \frac{\eta(\hat{\mathbf{z}}^r - \mathbf{A}(\mathbf{w}^{r+1} + \mathbf{n}^{r+1}))}{1 + \alpha\eta}. \tag{25}$$

Combining (15) and (25) gives

$$\hat{\mathbf{y}}^{r+1} = \frac{2}{\eta}\boldsymbol{\lambda}^{r+1} - \hat{\mathbf{z}}^{r} = \frac{1}{1+\alpha\eta}\left((1-\alpha\eta)\hat{\mathbf{z}}^{r} - 2\mathbf{A}(\mathbf{w}^{r+1}+\mathbf{n}^{r+1})\right). \quad (26)$$

We omit specification of (16)-(17) since it is identical to PDMM [20], [21] and ECL [8], [13]. When the subdifferential of the indicator function is used in T_2 given in (11), (16)-(17) is summarized by

$$\hat{\mathbf{z}}^{r+1} = \mathbf{P}_G\hat{\mathbf{y}}^{r+1},$$

where this indicates dual variables are exchanged/swapped between connected nodes using permutation matrix $\mathbf{P}_G$.

Summarizing from here, the alternatingly recurrent update rule using Peaceman-Rachford splitting (14)-(17) results in (24), (26). To simplify the notation, we replaced symbols as $\mathbf{y} \leftarrow \hat{\mathbf{y}}$, $\mathbf{z} \leftarrow \hat{\mathbf{z}}$. Then, the update rule is summarized by

$$\mathbf{w}^{r+1,k+1} = \arg\min_{\mathbf{u}}\left(f(\mathbf{u}, \boldsymbol{\xi}^{r+1,k}) + \frac{\eta}{2(1+\alpha\eta)}\|\mathbf{A}(\mathbf{u}+\mathbf{n}^{r+1,k}) - \mathbf{z}^{r}\|^2\right), \quad (27)$$

$$\mathbf{y}^{r+1} = \frac{1}{1+\alpha\eta}\left((1-\alpha\eta)\mathbf{z}^{r} - 2\mathbf{A}(\mathbf{w}^{r+1}+\mathbf{n}^{r+1})\right),$$
$$\mathbf{z}^{r+1} = \mathbf{P}_G\mathbf{y}^{r+1},$$
$$\mathbf{w}^{r+1} = \mathbf{w}^{r+1,K}, (k = 0, \ldots, K-1).$$

Let us recall that, in Subsec. IV-A, the convex approximated cost function f using the order Taylor expansion is introduced.

$$f(\mathbf{w}, \mathbf{x}) \approx f(\mathbf{w}^{r+1,k}, \mathbf{x}^{r+1,k}) + \left\langle \nabla f(\mathbf{w}_i^{r+1,k}, \mathbf{x}^{r+1,k}), \mathbf{w} - \mathbf{w}_i^{r+1,k}\right\rangle + \frac{1}{2\mu}\|\mathbf{w} - \mathbf{w}^{r+1,k}\|^2. \quad (28)$$

Substituting (28) into (27) results in

$$\mathbf{0} = \nabla f(\mathbf{w}^{r+1,k}, \boldsymbol{\xi}^{r+1,k}) + \frac{1}{\mu}(\mathbf{w} - \mathbf{w}^{r+1,k}) + \frac{\eta}{1+\alpha\eta}\{\mathbf{A}^{\mathsf{T}}\mathbf{A}(\mathbf{w}+\mathbf{n}^{r+1,k}) - \mathbf{A}^{\mathsf{T}}\mathbf{z}^{r}\}$$
$$\mathbf{0} = \mu(1+\alpha\eta)\nabla f(\mathbf{w}^{r+1,k}, \boldsymbol{\xi}^{r+1,k}) + (1+\alpha\eta)(\mathbf{w} - \mathbf{w}^{r+1,k}) + \mu\eta\{\mathbf{A}^{\mathsf{T}}\mathbf{A}(\mathbf{w}+\mathbf{n}^{r+1,k}) - \mathbf{A}^{\mathsf{T}}\mathbf{z}^{r}\}$$

As defined in Subsec. IV-A, $\{\mathbf{A}_{i|j}, \mathbf{A}_{j|i}\} = \{\mathbf{I}, -\mathbf{I}\}$. Then, $\mathbf{A}^{\mathsf{T}}\mathbf{A} = \mathrm{diag}[E_1\mathbf{I}, \ldots, E_N\mathbf{I}]$. Then, (27) can be replaced by

$$\mathbf{w}_i^{r+1,k+1} \leftarrow \frac{1}{1+\alpha\eta+\eta\mu E_i}\left(\{1+\alpha\eta\}\mathbf{w}_i^{r+1,k} - \mu\{1+\alpha\eta\}\nabla f_i(\mathbf{w}_i^{r+1,k}, \boldsymbol{\xi}_i^{r+1,k}) + \mu\eta(\sum \mathbf{A}_{i|j}^{\mathsf{T}}\mathbf{z}_{i|j}^{r} - E_i\mathbf{n}_i^{r+1,k})\right).$$

REFERENCES

[1] M. Joshi, A. Pal, and M. Sankarasubbu, "Federated learning for healthcare domain–pipeline, applications and challenges," *ACM Trans. Comput. Healthcare*, vol. 3, no. 4, pp. 1–36, Nov. 2022, doi: 10.1145/3533708.

[2] M. Vucovich et al., "Anomaly detection via federated learning," 2022, *arXiv:2210.06614*.

[3] D. Guliani, F. Beaufays, and G. Motta, "Training speech recognition models with federated learning: A quality/cost framework," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2021, pp. 3080–3084.

[4] M. Blot, D. Picard, M. Cord, and N. Thome, "Gossip training for deep learning," 2016, *arXiv:1611.09726*.

[5] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[6] R. Xin, U. A. Khan, and S. Kar, "Variance-reduced decentralized stochastic optimization with accelerated convergence," *IEEE Trans. Signal Process.*, vol. 68, pp. 6255–6271, 2020.

[7] G. Zhang and R. Heusdens, "Distributed optimization using the primal-dual method of multipliers," *IEEE Trans. Signal Inf. Process. over Netw.*, vol. 4, no. 1, pp. 173–187, Mar. 2018.

[8] K. Niwa et al., "Asynchronous decentralized optimization with implicit stochastic variance reduction," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8195–8204.

[9] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2019, pp. 2512–2520.

[10] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 1–11. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/60a6c4002cc7b29142def8871531281a-Paper.pdf

[11] W. Wei et al., "A framework for evaluating gradient leakage attacks in federated learning," 2020, *arXiv:2004.10397*.

[12] X. Luo, Y. Wu, X. Xiao, and B. C. Ooi, "Feature inference attack on model predictions in vertical federated learning," in *Proc. IEEE 37th Int. Conf. Data Eng. (ICDE)*, Apr. 2021, pp. 181–192.

[13] K. Niwa, N. Harada, G. Zhang, and W. B. Kleijn, "Edge-consensus learning: Deep learning on P2P networks with nonhomogeneous data," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020.

[14] M. Abadi et al., "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318, doi: 10.1145/2976749.2978318.

[15] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Comput. Math. Appl.*, vol. 2, no. 1, pp. 17–40, 1976.

[16] Z. Huang and Y. Gong, "Differentially private ADMM for convex distributed learning: Improved accuracy via multi-step approximation," 2020, *arXiv:2005.07890*.

[17] D. W. Peaceman and H. H. Rachford Jr., "The numerical solution of parabolic and elliptic differential equations," *J. Soc. Ind. Appl. Math.*, vol. 3, no. 1, pp. 28–41, 1955.

[18] Z. Huang, R. Hu, Y. Guo, E. Chan-Tin, and Y. Gong, "DP-ADMM: ADMM-based distributed learning with differential privacy," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1002–1012, 2020, doi: 10.1109/TIFS.2019.2931068.

[19] C. Dwork, "Differential privacy," in *Automata, Languages and Programming. ICALP*, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds. Berlin, Germany: Springer, 2006.

[20] G. Zhang and R. Heusdens, "On simplifying the primal-dual method of multipliers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 4826–4830.

[21] T. W. Sherson, R. Heusdens, and W. B. Kleijn, "Derivation and analysis of the primal-dual method of multipliers based on monotone operator theory," *IEEE Trans. Signal Inf. Process. over Netw.*, vol. 5, no. 2, pp. 334–347, Jun. 2019.

[22] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, 2017.

[23] E. K. Ryu and S. Boyd, "Primer on monotone operator methods," *Appl. Comput. Math*, vol. 15, no. 1, pp. 3–43, 2016.

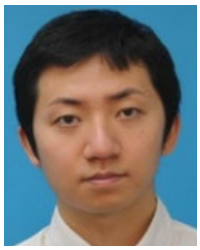
[24] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," *IEEE Trans. Inf. Theory*, vol. 63, no. 6, pp. 4037–4049, Jun. 2017.

[25] K. Rajawat and C. Kumar, "A primal-dual framework for decentralized stochastic optimization," 2020, *arXiv:2012.04402*.
[26] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.
[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.
[28] D. Böhning, "Multinomial logistic regression algorithm," *Ann. Inst. Stat. Math.*, vol. 44, no. 1, pp. 197–200, Mar. 1992, doi: 10.1007/BF00048682.

Takumi Fukami received the B.E. and M.E. degrees from the School of Advanced Science and Engineering, Waseda University, in 2018 and 2020, respectively. Since joining the Nippon Telegraph and Telephone Corporation (NTT) in 2020, he has been engaged in research on secure computation, data privacy, and distributed learning. He is currently with NTT Social Informatics Laboratories.

Tomoya Murata received the B.S. degree from Tokyo University of Science in 2015 and the M.S. degree from Tokyo Institute of Technology in 2017. He is currently pursuing the Ph.D. degree from The University of Tokyo. After joining NTT DATA Mathematical Systems Inc., in 2017, he has been engaged in research and development of various machine learning techniques. His research interests include statistical learning theory, stochastic optimization, and federated learning.

Kenta Niwa (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in information science from Nagoya University in 2006, 2008, and 2014, respectively. Since joining Nippon Telegraph and Telephone Corporation (NTT) in 2008, he has been engaged in research on microphone array signal processing. From 2017 to 2018, he was a Visiting Researcher of Victoria University of Wellington and involved with research on distributed machine learning and mathematical optimization. He is currently with NTT Communication Science Laboratories. He received the Awaya Prize by the Acoustical Society of Japan in 2010.



Iifan Tyou (Member, IEEE) received the B.E. and M.E. degrees from the Graduate School of Systems and Information Engineering, University of Tsukuba, in 2008 and 2010, respectively. Since joining the Nippon Telegraph and Telephone Corporation (NTT) in 2010, he has been engaged in research and development of traceability systems and the Internet of the Thing (IoT) gateway security systems. He is currently with NTT Social Informatics Laboratories. His research interests include decentralized platform, such as blockchain, cryptography, data distribution platform, and federated learning.