A Localized Primal-Dual Method for Centralized/Decentralized Federated Learning Robust to Data Heterogeneity

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

***Abstract*—Generalized Edge-Consensus Learning (G-ECL) is a primal-dual method to solve loss-sum minimization problems. We propose Local Generalized Edge-Consensus Learning (Local G-ECL) as an extension of previous G-ECL, aiming to be a decentralized/centralized FL algorithm robust to heterogeneous data sets with a large number of local updates. Our contributions are as follows: (C1) success in theoretical gradient norm convergence analysis nearly independently of data heterogeneity, and (C2) equivalency proof between our primal-dual Local G-ECL and a pure primal Stochastic Controlled Averaging (SCAFFOLD) algorithm in centralized settings, where the difference in the initial local model for each round is ignored. Numerical experiments using image classification tests validated that Local G-ECL is robust to heterogeneous data with a large number of local updates.**

***Index Terms*—Data heterogeneity, federated learning, localized learning, primal-dual optimization.**

I. Introduction

FEDERATED Learning (FL) research [1], [2] has flourished for secure machine learning via message passing between user nodes without exchanging sensitive data sets. FL is gaining attention as a method for training models across multiple organizations, including hospitals, which cannot share data due to personal and confidential information. Many studies have been published for improving the accuracy and model consensus speed, e.g., [3], [4], [5]. Initially, centralized FL was mainstream, i.e., each worker updates their local model variable using a local data set, and the central server averages aggregated local model variables. Meanwhile, decentralized FL consisting of peer-to-peer (P2P) workers has also been widely used due to its flexible network topology and elimination of hierarchy among workers. In recent centralized and decentralized FL studies, a standard problem setting has been that the data sets held by each worker are statistically heterogeneous and the number of inner local updates is large ($K > 1$) for sparse communication. Hence, this study aims to construct decentralized/centralized FL algorithms robust to data heterogeneity with local updates.

Simple FL algorithms are FedAvg [6] for centralized FL and Gossip (also referred to as Local Decentralized Stochastic Gradient Descent (SGD)) [7], [8], [9], [10], [11] for decentralized FL; specifically, local model updates and averaging of aggregated model variables are alternatingly performed. However, these FL algorithms are not robust to data heterogeneity with local updates, i.e., they often do not reach the global minimum due to large gradient variance among workers. This is theoretically shown in for example [12], [13] that gradient norm convergence is guaranteed under the assumption to restrict the difference between local gradients and global gradient (an example form is given by Assumption 5: bounded gradient similarity in Section IV).

To construct FL algorithms robust to data heterogeneity with a large number of local updates, two approaches are applicable. The first approach is a pure *primal* method including stochastic variance reduction [16], [17], [18], [19] in local updates, such as Stochastic Controlled Averaging (SCAFFOLD) [12] and MIME [20] for centralized FL, and decentralized SCAFFOLD [14]. In their model update, local stochastic gradient is corrected by using control variates to reduce the variance of stochastic gradients of local workers. In centralized settings, the bounded gradient similarity assumption, often required for local algorithms without stochastic variance reduction, is not necessary for SCAFFOLD's convergence analysis, and it has been reported that SCAFFOLD is indeed robust to data heterogeneity in numerical results. On the other hand, in decentralized settings, the convergence analysis of decentralized SCAFFOLD requires the gradient similarity assumption, although the numerical results suggest it is robust to data heterogeneity. Hence, the theoretical explanation of the robustness toward data heterogeneity of decentralized SCAFFOLD is still an open problem.

The second approach is a *primal-dual* method to solve loss-sum minimization while imposing linear constraints to make local model variables identical. Representative algorithms are

Manuscript received 7 February 2023; revised 16 August 2023 and 11 October 2023; accepted 1 December 2023. Date of publication 25 December 2023; date of current version 31 January 2024. This paper was produced by the IEEE Publication Technology Group. They are in Piscataway, NJ. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Wang. *(Corresponding author: Iifan Tyou.)*

Iifan Tyou is with the NTT Social Information Laboratories, NTT Corporation, Tokyo 100-8116, Japan, and also with the The University of Tokyo, Tokyo 113-0033, Japan (e-mail: iifan.tyou@ntt.com).

Tomoya Murata is with the NTT DATA Mathematical Systems Inc., Tokyo 160-0016, Japan (e-mail: murata@msi.co.jp).

Takumi Fukami is with the NTT Social Information Laboratories, NTT Corporation, Tokyo 100-8116, Japan (e-mail: takumi.fukami.as@hco.ntt.co.jp).

Yuki Takezawa is with the Kyoto University / OIST, Kyoto 606-8501, Japan (e-mail: yuki-takezawa@ml.ist.i.kyoto-u.ac.jp).

Kenta Niwa is with the NTT, Communication Science Laboratories, NTT Corporation, Tokyo 100-8116, Japan (e-mail: kenta.niwa.bk@hco.ntt.co.jp).

Digital Object Identifier 10.1109/TSIPN.2023.3343616

TABLE I
COMPARISON OF THE COMMUNICATION COMPLEXITIES OF THE SEVERAL CENTRALIZED AND DECENTRALIZED FL ALGORITHMS

	Communication Complexity	Local Opt.	Learning Type	Assump.
FedAvg(Local SGD) [6]	$\frac{(B^2+1)L}{\varepsilon}+\frac{\sqrt{L}G}{\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Centralized	2-5
SCAFFOLD [12]	$\frac{L}{\varepsilon}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Centralized	2-4
Gossip(Local Decentralized SGD) [13]	$\frac{\sqrt{B^2+1}L}{p\varepsilon}+\frac{LG}{p\varepsilon^{3/2}}+\frac{L\sigma}{\sqrt{pK}\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Decentralized	1-5
Decentralized SCAFFOLD [14]	$\frac{(B^2+1)L}{p^2\varepsilon^2}+\frac{LG^2}{p^2\varepsilon^2}+\frac{L\sigma^2}{p^2K\varepsilon^2}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Decentralized	1-5
G-ECL [15]	$\frac{L+\sqrt{1-p}\psi}{p^2\varepsilon}+\frac{L\sigma}{p^2\varepsilon^{3/2}}+\frac{L\sigma^2}{n\varepsilon^2}$		Decentralized	1-4
Local G-ECL w/o warm-start	$\frac{L+\sqrt{1-p}\psi}{p^2\varepsilon}+\frac{L^{2/3}\zeta_{1,0}^{2/3}}{p\varepsilon}+\frac{L\sigma}{p^2\sqrt{K}\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Decentralized*	1-4
Local G-ECL w/ warm-start	$\frac{L+\sqrt{1-p}\psi}{p^2\varepsilon}+\frac{\log\frac{L\zeta_{1,0}}{p}}{p}+\frac{L\sigma}{p^2\sqrt{K}\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}$	✓	Decentralized*	1-4

* For Local G-ECL (with/without warm-start), the learning type is noted as "Decentralized" since our analysis is provided for decentralized FL Algorithm 1 in Sec. III. By selecting the fully-connected topology (p=1), its equivalent computation is performed in centralized FL Algorithm 2.
"Communication Complexity" means the order of the necessary number of communication rounds to achieve $\mathbb{E}\|\nabla f(x)\|^2 \leq \varepsilon$, where $f(x)=\frac{1}{n}\sum_{i=1}^n f_i(x)$ denotes averaged local loss functions. "Local Opt." denotes whether internal local updates ($K>1$) are considered in the algorithm analysis.
"Learning Type" indicates the assumed network type in FL. "Assump." means the required assumptions (described in Subsec. IV-A) in the theory.
p is the mixing parameter of the mixing matrix (Assumption 1). L is the smoothness parameter of f_i (Assumption 2),
σ^2 is the stochastic gradient variance (Assumption 3), and B and G are gradient similarity parameters (Assumption 5).
ε is the desired optimization accuracy, K is the number of local updates, n is the number of local workers.
$\psi := \|D-E\|_2$ is defined in Sec. IV. $\zeta_{1,0} := \frac{1}{n}\sum_{i=1}^n \|\nabla f_i(x_0)-\nabla f(x_0)\|$ is the gradient similarity at initial model variable x_0.
For simple presentations, $f(x_0)-f(x_*)$ is assumed to be $O(1)$.

the distributed Alternating Direction Method of Multiplier (distributed ADMM) [21], FedSplit [22], the Primal-Dual Method of Multipliers (PDMM) for decentralized FL [23], [24], and its extension Edge-Consensus Learning (ECL) [25], [26]. In their model update, local stochastic gradient is corrected by using dual variables originated from linear constraints. A variant of ECL called Generalized ECL (G-ECL) has recently been proposed [15]. In G-ECL, the update rules of ECL are reformulated to explicitly include the concept of mixing matrix (see Assumption 1) and it can be equivalently formulated as Gradient Tracking [27], [28]. Its convergence analysis for non-convex loss functions without using a bounded gradient similarity assumption has been rigorously proven. However, multiple local updates are not applicable to the algorithm, and local drifting due to multiple local updates is not theoretically considered.

These two approaches, namely *primal* stochastic variance reduction methods and *primal-dual* methods, resemble each other in that a local stochastic gradient is corrected by using auxiliary variables (e.g., control variates, dual variables). This correction is important to reduce gradient drift due to data heterogeneity with a large number of local updates. Their connection is first shown in ECL with Implicit Stochastic Variance Reduction (ECL-ISVR) [26], i.e., there exists an optimal point to connect a pure primal stochastic variance reduction and primal-dual formalism. However, its algorithm form is not equivalent to existing stochastic variance reduction-based FL algorithms (e.g., SCAFFOLD). Furthermore, a strong approximated assumption is used in its convergence analysis. Hence, the relationship between a primal-dual approach and a pure primal approach is still unclear.

In this paper, we propose Local G-ECL for centralized/decentralized FL robust to data heterogeneity with a large number of local updates as a variant of the existing primal-dual method (G-ECL). Our contribution is summarized as follows:

(C1) Local G-ECL and its convergence analysis: We succeeded in convergence analysis of a localized variant of G-ECL, namely Local G-ECL, where the algorithm and its analysis are shown in Subsection III-A and Section IV, respectively. The obtained analytical results are summarized in Table I. The Local G-ECL's convergence rate[1] associated with the gradient norm is nearly independent of data heterogeneity inheriting the properties of G-ECL, and it has evolved in that it takes the effect of a large number of local updates ($K>1$) into account. The robustness to data heterogeneity has not been shown in the previous analysis of decentralized SCAFFOLD, and has been only empirically justified. Compared with G-ECL, by increasing the number of local updates K in Local G-ECL, the effect of stochastic gradient drift will be relaxed.

(C2) Clarification of the connection between Local G-ECL and SCAFFOLD: We have succeeded in equivalency proof of primal-dual Local G-ECL and pure primal SCAFFOLD for centralized settings (Subsec. III-B). Local G-ECL's formulation is started from decentralized settings in Subsec. III-A. When we perform it over a fully-connected network topology, its equivalent formalism can also be performed in a centralized FL manner (details are noted in Subsec. III-B). Interestingly, we found that the update rules of the primal-dual Local G-ECL and the pure primal SCAFFOLD are *equivalent* in centralized settings, ignoring differences in the initial points of the local updates. This equivalency suggests that pure primal stochastic variance reduction-based methods may be naturally derived from the primal-dual formalism to solve constrained loss-sum minimization problems.

II. RELATED WORKS

In this section, we first summarize some representative symbols and notations used throughout this paper. Then, the most

[1] In Table I, convergence rates for Local G-ECL with two initialization conditions (with/without warm-start) are introduced (See Corollary 5 and 6 in Section IV). With warm-start initialization, the effect of data heterogeneity can be mostly ignored.

related previous methods are illustrated to clarify their connection to our proposed method described in Section III. In Section II-A, as a pure primal method, SCAFFOLD is explained. In Section II-B, as a primal-dual method, G-ECL is illustrated. In Section II-C, the remaining issues of the previous methods are described. In the next section, we will propose Local G-ECL, which is a localized variant of G-ECL, and a connection between pure primal SCAFFOLD and primal-dual Local G-ECL is discussed.

Symbols and Notation

- n *:* the number of workers.
- $\mathcal{N}_i$*:* the index set of the neighborhood of worker i.
- $\mathcal{N}_i^+$*:* $\mathcal{N}_i \cup \{i\}$.
- $\mathcal{N}$*:* the index set of all the workers.
- W*:* mixing matrix of the decentralized topology.
- f_i*:* local objective function of worker i.
- f*:* objective function, which is the mean of $\{f_i\}_{i=1}^n$.
- $\mathcal{D}_i$*:* the local data distribution of worker i.
- x_i*:* local model parameter of worker i.
- $\widetilde{x}_i$*:* aggregated model parameter using the model parameters of $\mathcal{N}_i^+$.
- c_i, $\nu_{i|\cdot}$, $\check{\nu}_{i|\cdot}$, λ_i*:* gradient correction variables or dual variables of worker i.
- $\nabla F_i(\cdot, \xi_i)$*:* stochastic gradient using local sample ξ_i of worker i.
- η*:* step size.
- K*:* the number of inner local updates.
- R*:* the number of communication rounds.

A. Stochastic Controlled Averaging (SCAFFOLD)

As a pure primal method, SCAFFOLD for centralized settings is briefly explained. Let us consider the following n local worker's loss-sum minimization problem with respect to the primal model variable x as

$$\min_x \frac{1}{n}\sum_{i=1}^n f_i(x), \text{ where } f_i(x) := \mathbb{E}_{\xi_i \sim \mathcal{D}_i}[F_i(x;\xi_i)], \tag{1}$$

f_i is the risk function on the data set $\mathcal{D}_i$ held by the i-th local worker, and $\xi_i \sim \mathcal{D}_i$ denotes the stochastic data sampling in each local worker. Each local worker is allowed to have statistically heterogeneous data sets, i.e., $\mathcal{D}_i \neq \mathcal{D}_j$ when $i \neq j$.

In SCAFFOLD, the problem (1) is solved in a centralized FL manner. Specifically, (i) the average of model variables aggregated on a central server and (ii) stochastic variance reduction-based local updates of x_i for $K(\geq 1)$ times using local data set $\mathcal{D}_i$ are alternately repeated. In principle, the gradient drift in each local worker due to data heterogeneity with local updates can be corrected by using approximated local full-gradient $c_i \approx \nabla f_i(x_i)$ and global full-gradient $\widetilde{c} = \frac{1}{n}\sum_{j=1}^n c_j$. When using SCAFFOLD (option II [12]), update rules are given by

[Central server]

$$\widetilde{x}^{(r+1)} = \frac{1}{n}\sum_{j=1}^n x_j^{(r+1)}, \quad \widetilde{c}^{(r+1)} = \frac{1}{n}\sum_{j=1}^n c_j^{(r+1)},$$

[Local worker]i

$$\begin{aligned} x_i^{(r),k+1} &= x_i^{(r),k} - \eta(\nabla F_i(x_i^{(r),k};\xi_i^{(r),k}) - c_i^{(r)} + \widetilde{c}^{(r)}),\\ c_i^{(r+1)} &= c_i^{(r)} - \widetilde{c}^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r)} - x_i^{(r+1)}), \end{aligned} \tag{2}$$

where $r \in \{1,\dots,R\}$ and $k \in \{1,\dots,K\}$ denote the index of outer communication rounds and that of inner local updates respectively, $\eta(>0)$ is the learning rate, and $\nabla F_i(x_i^{(r)};\xi_i)$ denotes stochastic gradient using mini-batch samples $\xi_i \sim \mathcal{D}_i$. In (2), initial points of the central server and local workers are given by $x_i^{(r+1)} = x_i^{(r),K}$ and $x_i^{(r),0} = \widetilde{x}^{(r)}$, respectively.

Recently, SCAFFOLD is extended to be performed in decentralized settings (decentralized SCAFFOLD [14]). Since n workers are then connected by P2P relationship, each local worker is allowed to update the local model variable x_i without using computation on the central server.

B. Generalized Edge-Consensus Learning (G-ECL)

Another approach to solving the problem (1) for both centralized and decentralized FL is *primal-dual* methods. The original problem (1) can be rewritten by a loss-sum minimization with constraints regarding the local model variables $\{x_i\}_{i=1}^n$:

$$\min_{\{x_i\}_{i=1}^n} \frac{1}{n}\sum_{i=1}^n f_i(x_i) \quad \text{s.t. } x_i = x_j \ (\forall i \in \mathcal{N}, j \in \mathcal{N}_i), \tag{3}$$

where $\mathcal{N} := \{1,\dots,n\}$, $\mathcal{N}_i$ denotes the index set of the i-th local worker's connection ($\mathcal{N}_i$ is regarded as $\mathcal{N}$ for centralized cases). To solve (3) by alternately repeating (i) *synchronous* message passing between connected workers and (ii) inner local updates for K times, many primal-dual algorithms have been studied. Since linear constraints to make model variables identical are imposed in (3), it is expected to be robust to gradient drift due to data heterogeneity.

Particularly for decentralized FL settings, when the *lifted* dual variable is introduced such that it satisfies $\nu_{i|j} = \nu_{j|i}$ $(i \in \mathcal{N}, j \in \mathcal{N}_i)$ and its stacked notation is given by $\nu_i := \{\nu_{i|1},\dots,\nu_{i|i-1},\nu_{i|i+1},\dots,\nu_{i|n}\}$, the primal-dual problem of (3) is formulated as

$$\begin{aligned} &\min_{\{x_i\}_{i=1}^n} \max_{\{\nu_i\}_{i=1}^n} \frac{1}{n}\sum_{i=1}^n f_i(x_i) + \frac{1}{n}\sum_{i=1}^n \sum_{j\in\mathcal{N}_i} \langle -A_{i|j}\nu_{j|i}, x_i\rangle\\ &\text{s.t. } \nu_{i|j} = \nu_{j|i} \ (\forall i \in \mathcal{N}, j \in \mathcal{N}_i), \end{aligned} \tag{4}$$

where $A_{i|j} := I$ for $i > j$ and $A_{i|j} := -I$ for $i < j$. Assuming that f_i is restricted to be convex, the min-max problem (4) is generally solved by a primal-dual algorithm. Concretely, after introducing an auxiliary dual variable $\check{\nu}_{i|j}$ through some parameter transformation of $\nu_{i|j}$, the primal variable x_i and new dual variable $\check{\nu}_{i|j}$ are alternately updated, and the dual variables are exchanged between connected workers $\check{\nu}_{i|j} \rightleftharpoons \check{\nu}_{j|i}$ in PDMM [23], [24]:

$$x_i^{(r+1)} = \arg\min_{x_i}\Big(f_i(x_i) + \sum_{j\in\mathcal{N}_i}\frac{\gamma_{i|j}}{2}\|A_{i|j}x_i - \check{\nu}_{j|i}^{(r)}\|^2\Big),$$

$$\check{\nu}_{i|j}^{(r+1)} = \check{\nu}_{i|j}^{(r)} - 2A_{i|j}x_i^{(r+1)}, \qquad (j \in \mathcal{N}_i) \tag{5}$$

where $\gamma_{i|j}(>0)$ denotes some weight coefficient. In ECL [25], [26], $f_i(x_i)$ in (5) is replaced by local quadratic function $f_i(x_i^{(r)}) + \langle \nabla f_i(x_i^{(r)}), x_i - x_i^{(r)} \rangle + 1/(2\eta)\|x_i - x_i^{(r)}\|^2$ to allow f_i to be non-convex. Then, x_i-update rule in (5) can be rewritten in a closed form. Additionally, stochastic gradient is used to reduce the computational cost.

Furthermore in G-ECL [15], the update rules in (5) are reformulated to explicitly include general model mixing, which is commonly used in many decentralized FL algorithms, such as the Gossip method. Then, the update rules are given by

$$\begin{aligned} v_i^{(r)} &= \nabla F_i(x_i^{(r)}; \xi_i^{(r)}) - \lambda_i^{(r)}, \\ x_i^{(r+1)} &= \widetilde{x}_i^{(r)} - \eta v_i^{(r)} \text{ and } \widetilde{x}_i^{(r+1)} = \sum_{j \in \mathcal{N}_i^+} W_{ij} x_j^{(r+1)}, \\ \lambda_i^{(r+1)} &= \lambda_i^{(r)} - \sum_{j \in \mathcal{N}_i^+} W_{ij} v_j^{(r)} + v_i^{(r)} + \sum_{j \in \mathcal{N}_i} \frac{\alpha_{i|j}}{2}(\widetilde{x}_j^{(r)} - \widetilde{x}_i^{(r)}), \end{aligned} \tag{6}$$

where $\mathcal{N}_i^+ := \mathcal{N}_i \cup \{i\}$ denotes the augmented index set including both connected workers with own worker, W_{ij} represents the mixing weight of the model variable of the worker $j \in \mathcal{N}_i^+$ in the i-th worker's local model update and $\alpha_{i|j}(\geq 0)$ is another weight coefficient. From x_i-update rule in (6), we can see that the stochastic gradient is corrected by using new dual variable λ_i. This idea to correct stochastic gradients using auxiliary variables resembles the x_i-update rule in pure primal SCAFFOLD (2). By using analysis techniques in [13] associated with Gossip's convergence analysis, G-ECL's convergence rate is rigorously proven without using the bounded gradient similarity assumption (see Assumption 5 in Section IV).

C. Remaining Issues in Related Works

There are several issues in both previous primal and primal-dual methods for decentralized settings as below.

(P1) The convergence analysis of decentralized SCAFFOLD requires the bounded gradient similarity assumption and the theoretical justification of the robustness to data heterogeneity is not achieved. On the other side, G-ECL cannot handle multiple local updates ($K > 1$), and its effect is not considered in its convergence analysis, although robustness to data heterogeneity is proven in the case $K = 1$ for decentralized settings.

(P2) The connection between pure primal SCAFFOLD and primal-dual G-ECL is still unclear, even though they resemble an idea to correct stochastic gradient using auxiliary variables.

III. Local Generalized ECL (Local G-ECL) and Its Connection to SCAFFOLD

In Section III-A, a localized variant of previous G-ECL, referred to as Local G-ECL, is proposed, where its convergence analysis appears in Section IV. In Section III-B, the relationship between the centralized version of the Local G-ECL and SCAFFOLD is revealed.

Algorithm 1: Local G-ECL for Decentralized FL.

1: **function** WORKER PROCESS ($x_i^{(0)}$, $\lambda_i^{(0)}$, η, W_{ij}, $\{\alpha_{i|j}\}_{j\in\mathcal{N}_i}$, K, R)
2: $\widetilde{x}_i^{(0)} \leftarrow x_i^{(0)}$.
3: **for each** $r = 0, \ldots, R-1$ **do**
4: $\widetilde{x}_i^{(r),0} \leftarrow \widetilde{x}_i^{(r)}$, $x_i^{(r),0} \leftarrow x_i^{(r)}$.
5: **for each** $k = 0, \ldots, K-1$ **do**
6: $v_i^{(r),k+1} \leftarrow \nabla F_i(x_i^{(r),k}; \xi_i^{(r),k}) - \lambda_i^{(r)}$.
7: $x_i^{(r),k+1} \leftarrow \widetilde{x}_i^{(r),k} - \eta v_i^{(r),k+1}$.
8: $\widetilde{x}_i^{(r),k+1} \leftarrow x_i^{(r),k+1}$.
9: $x_i^{(r+1)} \leftarrow x_i^{(r),K}$.
10: $\bar{v}_i^{(r+1)} \leftarrow \frac{1}{K}\sum_{k=0}^{K-1} v_i^{(r),k+1}$.
11: **Send**: $(x_i^{(r+1)}, \bar{v}_i^{(r+1)})$ to worker $j \in \mathcal{N}_i$.
12: **Receive**: $\{(x_j^{(r+1)}, \bar{v}_j^{(r+1)})\}_{j\in\mathcal{N}_i}$.
13: $\widetilde{x}_i^{(r+1)} \leftarrow \sum_{j\in\mathcal{N}_i^+} W_{ij} x_j^{(r+1)}$.
14: $\lambda_i^{(r+1)} \leftarrow \lambda_i^{(r)} - \sum_{j\in\mathcal{N}_i^+} W_{ij}\bar{v}_j^{(r+1)} + \bar{v}_i^{(r+1)} + \sum_{j\in\mathcal{N}_i} \frac{\alpha_{i|j}}{2}(\widetilde{x}_j^{(r)} - \widetilde{x}_i^{(r)})$.
15: **if** $\{\alpha_{i|j}\}_{j\in\mathcal{N}_i} \neq \{0\}_{j\in\mathcal{N}_i}$ **then**
16: **Send**: $\widetilde{x}_i^{(r+1)}$ to worker $j \in \mathcal{N}_i$.
17: **Receive**: $\{\widetilde{x}_j^{(r+1)}\}_{j\in\mathcal{N}_i}$.
18: **end if**
19: **end function**

A. Local G-ECL for Decentralized FL

The procedures of the proposed Local G-ECL for decentralized settings are illustrated in Algorithm 1). Although our algorithm is based on the update rules of G-ECL in (6), the critical difference from G-ECL is the usage of multiple local updates. For each communication round r, local worker i updates their own local model variable x_i for $K(>1)$ times using stochastic local data sampling $\xi_i \sim \mathcal{D}_i$ (line 5–8). In line 7, gradient drift caused by data heterogeneity with local updates will be corrected by subtracting dual variable $\lambda_i^{(r)}$ from the stochastic gradient $\nabla F_i(x_i^{(r),k}; \xi_i^{(r),k})$. Note that for the initial point of local updates ($k = 0$), the stochastic gradient is computed at the local model $x_i^{(r),0} = x_i^{(r)}$ rather than the aggregated model $\widetilde{x}_i^{(r),0} = \widetilde{x}_i^{(r)}$. In line 12, worker i sends $(x_i^{(r+1)}, \bar{v}_i^{(r+1)})$ to the connected workers $j \in \mathcal{N}_i$ and receives $\{(x_j^{(r+1)}, \bar{v}_j^{(r+1)})\}_{j\in\mathcal{N}_i}$. In line 13, the local models are aggregated. In line 14, dual variable $\lambda_i^{(r)}$ is updated.

B. Local G-ECL for Centralized FL and Its Connection to SCAFFOLD

In this subsection, we start from the reformulation of Local G-ECL for decentralized settings (Algorithm 1) to be a centralized manner (Algorithm 2). This is mainly aimed at associating the primal-dual Local G-ECL with pure primal SCAFFOLD, explained in Section II-A.

a) Equivalent reformulation of Local G-ECL from decentralized to centralized settings: Let us consider performing Local G-ECL in Algorithm 1 over a fully-connected decentralized

Algorithm 2: Local G-ECL for Centralized FL.

function SERVER PROCESS ($\widetilde{x}^{(0)}$, R)
  Send: $\widetilde{x}^{(0)}$.
  for each $r = 0, 1, \ldots, R-1$ **do**
    Receive: $\{x_j^{(r+1)}\}_{j\in\mathcal{N}}$.
    $\widetilde{x}^{(r+1)} \leftarrow \frac{1}{n}\sum_{j=1}^n x_j^{(r+1)}$.
    Send: $\widetilde{x}^{(r+1)}$ to worker $j \in \mathcal{N}$.
end function

function WORKER PROCESS ($\lambda_i^{(0)}$, η, K, R)
  Receive: $\widetilde{x}^{(0)}$ and $x_i^{(0)} \leftarrow \widetilde{x}^{(0)}$.
  for each $r = 0, 1, \ldots, R-1$ **do**
    $x_i^{(r),0} \leftarrow x_i^{(r)}$, $\widetilde{x}_i^{(r),0} \leftarrow \widetilde{x}^{(r)}$.
    for each $k = 0, 1, \ldots, K-1$ **do**
      $v_i^{(r),k+1} \leftarrow \nabla F_i(x_i^{(r),k}; \xi_i^{(r),k}) - \lambda_i^{(r)}$.
      $x_i^{(r),k+1} \leftarrow \widetilde{x}_i^{(r),k} - \eta v_i^{(r),k+1}$.
      $\widetilde{x}_i^{(r),k+1} \leftarrow x_i^{(r),k+1}$.
    $x_i^{(r+1)} \leftarrow x_i^{(r),K}$.
    Send: $x_i^{(r+1)}$ to the server.
    Receive: $\widetilde{x}^{(r+1)}$ from the server.
    $\lambda_i^{(r+1)} \leftarrow \lambda_i^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r+1)} - x_i^{(r+1)})$.
end function

Algorithm 3: SCAFFOLD (Option II) Without Client Sampling [12] for Centralized FL.

function SERVER PROCESS ($\widetilde{x}^{(0)}$, $\widetilde{c}^{(0)}$, R)
  Send: $(\widetilde{x}^{(0)}, \widetilde{c}^{(0)})$.
  for each $r = 0, \ldots, R-1$ **do**
    Receive: $\{(x_j^{(r+1)}, c_j^{(r+1)})\}_{j\in\mathcal{N}}$.
    $\widetilde{x}^{(r+1)} \leftarrow \frac{1}{n}\sum_{j=1}^n x_j^{(r+1)}$.
    $\widetilde{c}^{(r+1)} \leftarrow \frac{1}{n}\sum_{j=1}^n c_j^{(r+1)}$.
    Send: $(\widetilde{x}^{(r+1)}, \widetilde{c}^{(r+1)})$ to worker $j \in \mathcal{N}$.
end function

function WORKER PROCESS ($c_i^{(0)}$, η, K, R)
  Receive: $(\widetilde{x}^{(0)}, \widetilde{c}^{(0)})$ and $x_i^{(0)} \leftarrow \widetilde{x}^{(0)}$.
  for each $r = 0, 1, \ldots, R-1$ **do**
    $x_i^{(r),0} \leftarrow \widetilde{x}^{(r)}$.
    for each $k = 0, 1, \ldots, K-1$ **do**
      $v_i^{(r),k+1} \leftarrow \nabla F_i(x_i^{(r),k}; \xi_i^{(r),k}) - c_i^{(r)} + \widetilde{c}^{(r)}$.
      $x_i^{(r),k+1} \leftarrow x_i^{(r),k} - \eta v_i^{(r),k+1}$.
    $x_i^{(r+1)} \leftarrow x_i^{(r),K}$.
    Send: $(x_i^{(r+1)}, c_i^{(r+1)})$ to the server.
    Receive: $(\widetilde{x}^{(r+1)}, \widetilde{c}^{(r+1)})$ from the server.
    $c_i^{(r+1)} \leftarrow c_i^{(r)} - \widetilde{c}^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r)} - x_i^{(r+1)})$.
end function

network and when the mixing matrix is set to the Metropolis-Hastings weights, i.e., $W_{ij} = 1/n$ for every i, j. Although the equivalency proof is noted later, one can construct a simplified algorithm of Local G-ECL for centralized settings, which is summarized in Algorithm 2. The communication cost per round of Algorithm 2 is two times smaller than that of Algorithm 1 because communicating $\bar{v}_i^{(r+1)}$ is not necessary. The simplification comes from the fact that the aggregated model $\widetilde{x}_i^{(r+1)}$ does not depend on worker index i in centralized cases and thus $\lambda_i^{(r+1)}$ can be computed only through $\{x_j^{(r+1)}\}_{j\in\mathcal{N}}$. The equivalency of Algorithm 2 to Algorithm 1 is proven as follows:

Proposition 1: Suppose that $W_{ij}=1/n$ for every i, j in Algorithm 1. Then, Local G-ECL for decentralized FL (Algorithm 1) is equivalent to centralized formalism (Algorithm 2).

Proof: First observe that $\widetilde{x}_i^{(r+1)} = \frac{1}{n}\sum_{j=1}^n x_j^{(r+1)} =: \widetilde{x}^{(r+1)}$ and $\bar{v}_i^{(r+1)} = \frac{1}{K\eta}(x_i^{(r+1)} - \widetilde{x}^{(r)})$ for $i \in \mathcal{N}$ in Algorithm 1. To show the equivalency of the two algorithms, we only need to show the equivalency of the update rules of $\lambda_i^{(r+1)}$. In Algorithm 1, line 14 holds that $-\sum_{j\in\mathcal{N}_i^+} W_{ij}\bar{v}_j^{(r+1)} + \bar{v}_i^{(r+1)} = \frac{1}{n}\sum_{j=1}^n \bar{v}_j^{(r+1)} - \bar{v}_i^{(r+1)} = \frac{1}{K\eta}(\widetilde{x}^{(r+1)} - \widetilde{x}^{(r)}) - \frac{1}{K\eta}(x_i^{(r+1)} - \widetilde{x}^{(r)}) = \frac{1}{K\eta}(\widetilde{x}^{(r+1)} - x_i^{(r+1)})$. Finally, since $\widetilde{x}_j^{(r)} - \widetilde{x}_i^{(r)} = 0$, it is concluded that $\lambda_i^{(r+1)} = \lambda_i^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r+1)} - x_i^{(r+1)})$. ■

b) Equivalency to SCAFFOLD: The concrete procedures of SCAFFOLD are given by Algorithm 3. The important observation is that the dual variable $\lambda_i^{(r)}$ in Algorithm 2 can be regard as the centered control variate $c_i^{(r)} - \widetilde{c}^{(r)}$ in Algorithm 3. The following proposition reveals a surprising connection between a primal-dual formulation and a pure primal formulation with a stochastic variance reduction technique; Local G-ECL (Algorithm 2) originated from a primal-dual formulation is equivalent to the famous SCAFFOLD with option II (Algorithm 3) that is a pure primal method, where the differences in the initial local model for each round are ignored.

Proposition 2: Suppose that $\lambda_i^{(0)}$ in Algorithm 2 is equal to $c_i^{(0)} - \widetilde{c}^{(0)}$ in Algorithm 3 and $\frac{1}{n}\sum_{i=1}^n c_i^{(0)} = \widetilde{c}^{(0)}$. Then, the centralized version of Local G-ECL (Algorithm 2) is equivalent to SCAFFOLD (Algorithm 3) except that Algorithm 2 computes the stochastic gradient at $x_i^{(r)}$ rather than $\widetilde{x}^{(r)}$ at the initial iteration $k = 0$ for each round r.

Proof: We will show that if the initial points of the local updates $x_i^{(r),0} = x_i^{(r)}$ are replaced by aggregated model $\widetilde{x}^{(r)}$ in Algorithm 2, then the obtained algorithm is exactly equivalent to Algorithm 3. To show this, it is necessary to show that $\lambda_i^{(r)}$ in Algorithm 2 matches $c_i^{(r)} - \widetilde{c}^{(r)}$ in Algorithm 3 using mathematical induction. Suppose that $\lambda_i^{(r)} = c_i^{(r)} - \widetilde{c}^{(r)}$ for some $r \geq 0$. Then, observe that $c_i^{(r+1)} - \widetilde{c}^{(r+1)} = c_i^{(r)} - \widetilde{c}^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r)} - x_i^{(r+1)}) - \widetilde{c}^{(r+1)} = \lambda_i^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r)} - x_i^{(r+1)}) - \frac{1}{n}\sum_{i=1}^n((c_i^{(r)} - \widetilde{c}^{(r)}) + \frac{1}{K\eta}(\widetilde{x}^{(r)} - \widetilde{x}^{(r+1)})) = \lambda_i^{(r)} + \frac{1}{K\eta}(\widetilde{x}^{(r+1)} - x_i^{(r+1)}) - \frac{1}{n}\sum_{i=1}^n \lambda_i^{(r)}$. Finally, observing that $\frac{1}{n}\sum_{i=1}^n c_i^{(r)} = \widetilde{c}^{(r)}$ and thus $\frac{1}{n}\sum_{i=1}^n \lambda_i^{(r)} = 0$ gives the desired claim. ■

IV. Analysis of Local G-ECL

Here, a theoretical analysis of Local G-ECL (Algorithm 1) is provided. First, we provide the notation used in this section.

Notations: For a vector $a \in \mathbb{R}^m$, $\|\cdot\|$ means the Euclidean norm, i.e., $\|a\| := \sqrt{\sum_{j=1}^m a_j^2}$. For a matrix $A \in \mathbb{R}^{m_1 \times m_2}$, $\|\cdot\|$ denotes the Frobenius norm, i.e., $\|A\| := \sqrt{\sum_{i,j} a_{ij}^2}$ for matrix $A \in \mathbb{R}^{m_1 \times m_2}$. Also, for a matrix $A \in \mathbb{R}^{m_1 \times m_2}$ $\|\cdot\|_2$ denotes the operator norm, i.e., $\|A\|_2 := \sup_{x \in \mathbb{R}^{m_2} \setminus \{0\}} \frac{\|Ax\|}{\|x\|}$. $D \in \mathbb{R}^{n \times n}$ is defined by $D_{ij} := \alpha_{i|j}$ if $(i,j) \in \mathcal{E}$ and $D_{ij} = 0$ otherwise. $E \in \mathbb{R}^{n \times n}$ denotes $\mathrm{diag}(\sum_{k \in \mathcal{N}_1} \alpha_{k|1}, \dots, \sum_{k \in \mathcal{N}_n} \alpha_{k|n})$. $\mathbf{1}$ means $(1, \dots, 1)^\top$. For $a, b \in \mathbb{R}$, $a \wedge b$ denotes $\min\{a, b\}$. $f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x)$.

A. Theoretical Assumptions

In this subsection, theoretical assumptions used in our analysis are introduced.

Assumption 1 (Mixing matrix [13]): $\|XW - \bar{X}\|_F^2 \leq (1-p)\|X - \bar{X}\|_F^2$ for every $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$ for some $p \in (0, 1]$, where $\bar{X} := (1/n) X \mathbf{1}\mathbf{1}^\top$ and W is symmetric and doubly stochastic. W is called *mixing matrix* with mixing parameter p.

Note that it holds that $p = 1$ when the network topology is fully-connected.

Assumption 2 (Smoothness): For every $i \in [n]$, f_i is L-smooth for some $L > 0$, i.e., $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$ for every $x, y \in \mathbb{R}^d$.

Assumption 3 (Bounded stochastic gradient variance): $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\nabla F_i(x_i; \xi_i)] = \nabla f_i(x_i)$ and $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}\|\nabla F_i(x_i; \xi_i) - \nabla f_i(x_i)\|^2 \leq \sigma^2$ for every $x_i \in \mathbb{R}^d$ and $i \in [n]$ for some $\sigma^2 \geq 0$.

Assumption 4 (Existence of a global minima): f has a global minima $x_* \in \mathbb{R}^d$.

Assumption 5 (Bounded gradient similarity): $\frac{1}{n} \sum_{i=1}^n \|\nabla f_i(x)\|^2 \leq G^2 + B^2 \|\nabla f(x)\|^2$ for every $x \in \mathbb{R}^d$ for some $G, B \geq 0$.

Assumption 5 is *not necessary* in our theory, although Assumption 5 is typically used to analyze local or decentralized settings to guarantee the convergence.[2]

B. Convergence Rate and Communication Complexity

In this subsection, the convergence rate and communication complexity of Algorithm 1 are derived. All the proofs are included in the supplementary material, a proof sketch is provided.[3]

Proof sketch: We first derive a decent lemma, which is typically used in first-order optimization theory.

Lemma 1 (Descent lemma): Suppose that Assumptions 1, 2 and 3 hold. Assume that $\alpha_{i|j} = \alpha_{j|i} \geq 0$ for every $(i,j) \in \mathcal{E}$ and $\frac{1}{n} \sum_{i=1}^n \lambda_i^{(0)} = 0$. Then, under $\eta \leq 1/(4L)$, it holds that

$$\mathbb{E}[f(\bar{x}^{(r+1)})] \leq \mathbb{E}[f(\bar{x}^{(r)})] + O\left(\frac{K\eta^2 L \sigma^2}{n} + \eta L^2 \sum_{k=0}^{K-1} \Xi^{(r),k}\right) - \Omega(\eta \sum_{k=0}^{K-1} \left(\mathbb{E}\|\nabla f(\bar{x}^{(r),k})\|^2 + \mathbb{E}\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(x_i^{(r),k})\|^2\right)$$

for $r \in \{0, \dots, R-1\}$. Here, $\Xi^{(r),k} := \frac{1}{n} \sum_{i=1}^n \mathbb{E}\|x_i^{(r),k} - \bar{x}^{(r),k}\|^2$ and $\bar{x}^{(r),k} := (1/n) \sum_{i=1}^n x_i^{(r),k}$.

The error $\Xi^{(r),k}$, which is the averaged deviation of each client model from the global model, arises in the upper bound in Lemma 1 and can be further bounded using the recursive inequality in the following lemma.

Lemma 2 (Recursion for $\Xi^{(r),k}$): Suppose that Assumptions 1, 2 and 3 hold. Under $\eta \leq p/(48KL)$,

$$\Xi^{(r),k} \leq (1 - \Omega(p))\Xi^{(r)} + O(K\eta^2\sigma^2) + O\left(\frac{K^2\eta^2}{p}\frac{1}{K}\sum_{\kappa=0}^{K-1}(\mathcal{E}^{(r),\kappa} + \mathbb{E}\|\nabla f(\bar{x}^{(r),\kappa})\|^2)\right)$$

for $k \in \{1, \dots, K\}$ and $r \in \{0, \dots, R-1\}$. Here, $\mathcal{E}^{(r),\kappa} := \frac{1}{n}\mathbb{E}\|\nabla f(\bar{X}^{(r),\kappa}) - \Lambda^{(r)} - \frac{1}{n}\nabla f(\bar{X}^{(r),\kappa})\mathbf{1}\mathbf{1}^\top\|^2$ and $\Xi^{(r)} := \Xi^{(r),0}$, where $\Lambda^{(r)} := [\lambda_1^{(r)}, \dots, \lambda_n^{(r)}]$.

The error $\mathcal{E}^{(r),k}$ represents the averaged deviation of local gradients $\{\nabla f_i(\bar{x}^{(r),k})\}_i$ corrected by dual variable $\Lambda^{(r)}$ from the global gradient $\nabla f(\bar{x}^{(r),k})$. The main difficulty of our analysis is to simultaneously evaluate the client drift caused by the decentralized nature and even caused by the multiple local updates. Since the dual variables $\Lambda^{(r)}$ are only periodically updated, we need to carefully investigate the effect of the drift correction by the dual variables containing only old information. This situation is very different from [15], where multiple updates are not applied in their algorithm and the dual variables always possess fresh information.

Lemma 3 (Recursion for $\mathcal{E}^{(r),k}$): Suppose that Assumptions 1, 2 and 3 hold. Assume that $\alpha_{i|j} = \alpha_{j|i} \geq 0$ for every $(i,j) \in \mathcal{E}$. Then, it holds that

$$\mathcal{E}^{(r),k} \leq (1 - \Omega(p))\mathcal{E}^{(r)} + O\left(\frac{\sigma^2}{pK} + \frac{L^2\Delta^{(r),K}}{p}\right) + O\left(\frac{L^2}{pK}\sum_{\kappa=0}^{K-1}(\Xi^{(r),\kappa} + \Delta^{(r),\kappa}) + \frac{(1-p)\|D-E\|_2^2}{p}\Xi^{(r)}\right)$$

for every $k \in \{0, \dots, K-1\}$ and $r \in \{0, \dots, R-1\}$. Here, $\mathcal{E}^{(r)} := \mathcal{E}^{(r),0}$ and $\Delta^{(r),\kappa} := \frac{1}{n}\mathbb{E}\|\bar{X}^{(r),\kappa} - \bar{X}^{(r)}\|^2 = O(K\eta^2 \sum_{\kappa=0}^{K-1} \mathbb{E}\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(x_i^{(r),k})\|^2 + K\eta^2\sigma^2/n)$.

Combining Lemma 1 with Lemmas 2 and 3 gives the following theorem.

Theorem 4 (Convergence rate): Suppose that Assumptions 1, 2, 3, and 4 hold. Assume that $\alpha_{i|j} = \alpha_{j|i} \geq 0$ for every $(i,j) \in \mathcal{E}$, $\frac{1}{n}\sum_{i=1}^n \lambda_i^{(0)} = 0$ and $x_i^{(0)} = x_j^{(0)}$ $(i \neq j)$. Then, Local G-ECL (Alg. 1) with appropriate $\eta = \Theta((p^2/(KL)) \wedge$

[2]For simple convergence analysis, client sampling is not considered in this paper.

[3]Due to the 13-page maximum limit for the first submission, a part of our proofs will appear in the Appendix of the revised paper.

$(p^2/(\sqrt{1-p}K\|D-E\|_2K)))$ satisfies

$$\frac{1}{KR}\sum_{r=0}^{R-1}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla f(\bar{x}^{(r),k})\|^2 \leq O\left(\frac{f_{0,*}}{\eta KR}+\left(\frac{\eta L}{n}+\frac{K\eta^2L^2}{p^4}\right)\sigma^2+\frac{K^2\eta^2L^2}{p^3R}\mathcal{E}^{(0)}\right),$$

where $\mathcal{E}^{(0)} := \frac{1}{n}\sum_{i=1}^n \mathbb{E}\|\nabla f_i(\bar{x}^{(0)}) - c_i^{(0)} - \nabla f(\bar{x}^{(0)})\|^2$ and $f_{0,*} := f(\bar{x}^{(0)}) - f(x_*)$.

Importantly, the heterogeneity $\mathcal{E}^{(0)}$ introduced in the convergence rate only depends on the initial point $\bar{x}^{(0)}$ and thus the uniform boundedness of the gradient similarity is not necessary. Theorem 4 immediately yields the following corollary.

Corollary 5 (Communication complexity): Assume that $f(\bar{x}^{(0)}) - f(x_*) = O(1)$. Let $\Lambda^{(0)} = [\lambda_1^{(0)},\dots,\lambda_n^{(0)}] := [0,\dots,0]$. Then, under the same conditions as in Theorem 4, if we appropriately choose learning rate η, the necessary number of communication rounds R of Local G-ECL becomes

$$O\left(\frac{L+\sqrt{1-p}\psi}{p^2\varepsilon}+\frac{L^{2/3}\zeta_{1,0}^{2/3}}{p\varepsilon}+\frac{L\sigma}{p^2\sqrt{K}\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}\right)$$

to satisfy $(1/(KR))\sum_{r=0}^{R-1}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla f(\bar{x}^{(r),k})\|^2 \leq \varepsilon$. Here, $\zeta_{1,0} := (1/n)\sum_{i=1}^n \|\nabla f_i(\bar{x}^{(0),0}) - \nabla f(\bar{x}^{(0),0})\|$ and $\psi := \|D-E\|_2$.

a) Improvement by warm-start strategy: To reduce the impact of $\mathcal{E}^{(0)}$, one can adopt a *warm-start* strategy for $\Lambda^{(0)} := [\lambda_1^{(0)},\dots,\lambda_n^{(0)}]$ rather than simply initializing $\lambda_i^{(0)} = 0$ at the expense of additional communication and computation cost. [4] For centralized cases, it is natural to use $\lambda_i^{(0)} = -\nabla f(\bar{x}^{(0)}) + \nabla f_i(\bar{x}^{(0)})$. For decentralized cases, to approximate $\nabla f(\bar{x}^{(0)})$, $\{\nabla f_i(\bar{x}^{(0)})\}_{i=1}^n$ is mixed $\widetilde{r}$ times.

Corollary 6 (Communication complexity with warm-start): Suppose that $f(\bar{x}^{(0)}) - f(x_*) = \Theta(1)$. Let $x_i^{(0)} = x_j^{(0)}$ and $\Lambda^{(0)} = [\lambda_1^{(0)},\dots,\lambda_n^{(0)}] := -\nabla f(X^{(0)})W^{\widetilde{r}} + \nabla f(X^{(0)})$, where $\nabla f(X^{(0)}) := [\nabla f_1(x_1^{(0)}),\dots,\nabla f_n(x_n^{(0)})]$. Then, under the same conditions as in Theorem 4, if we appropriately choose learning rate η, the necessary number of communication rounds $\widetilde{r}+R$ of Local G-ECL with warm-start becomes

$$O\left(\frac{L+\sqrt{1-p}\psi}{p^2\varepsilon}+\frac{\log\frac{L\zeta_{1,0}}{p}}{p}+\frac{L\sigma}{p^2\sqrt{K}\varepsilon^{3/2}}+\frac{L\sigma^2}{Kn\varepsilon^2}\right)$$

to satisfy $(1/(KR))\sum_{r=0}^{R-1}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla f(\bar{x}^{(r),k})\|^2 \leq \varepsilon$, where ψ and $\zeta_{1,0}$ are defined in Corollary 5.

We can see that the second term has a much better dependence on $\zeta_{1,0}$ and ε than the one in Corollary 5 even taking into account the additional communication rounds.

Remark 1: The communication complexity of Local G-ECL for centralized settings (Alg. 2) can be immediately obtained by substituting $p \leftarrow 1$ in Corollary 5 or 6 since the Alg. 2 is equivalent to Alg. 1 with a fully-connected network topology.

[4]The warm-start strategy *only requires the local full gradients* instead of the global full gradient. Although the strategy requires $\widetilde{O}(1/p)$ times communication rounds to mix the local gradients, the additional communication cost is reflected in the rate shown in Corollary 6. Also, the cost of the local full gradient computations is negligible when $RKb \gg n$ for empirical risk minimization with local data set size n.

C. Comparison With the Previous Results

Here, the obtained theoretical communication complexities are compared with the previous results.

a) Decentralized settings: As shown in Table I, the authors of the decentralized SCAFFOLD derived the convergence analysis under Assumption 5, i.e., bounded gradient similarity, which depends on B, G. Since the update rule of the decentralized SCAFFOLD and the proposed method are similar, by devising a convergence analysis of the decentralized SCAFFOLD, a convergence rate that does not depend on gradient similarity may be possible to obtain.

The communication complexities of the previous decentralized local algorithms (Local Decentralized SGD [13] and Decentralized SCAFFOLD [14]) heavily depend on heterogeneity parameters G and B in Assumption 5 due to the nature of decentralized network topology and the usage of local updates. In contrast, our theory does not rely on Assumption 5 and the obtained communication complexities only depend on the gradient heterogeneity *at initial point*. In particular, the use of the warm-start strategy will mitigate the impact of the gradient heterogeneity and its dependence is only logarithmic order. This shows that Local G-ECL is highly robust to heterogeneous data, which is typically desirable in FL. Also, compared with G-ECL, the usage of the multiple local updates reduces the third and fourth terms of the communication complexity bounds in Corollaries 5 and 6.

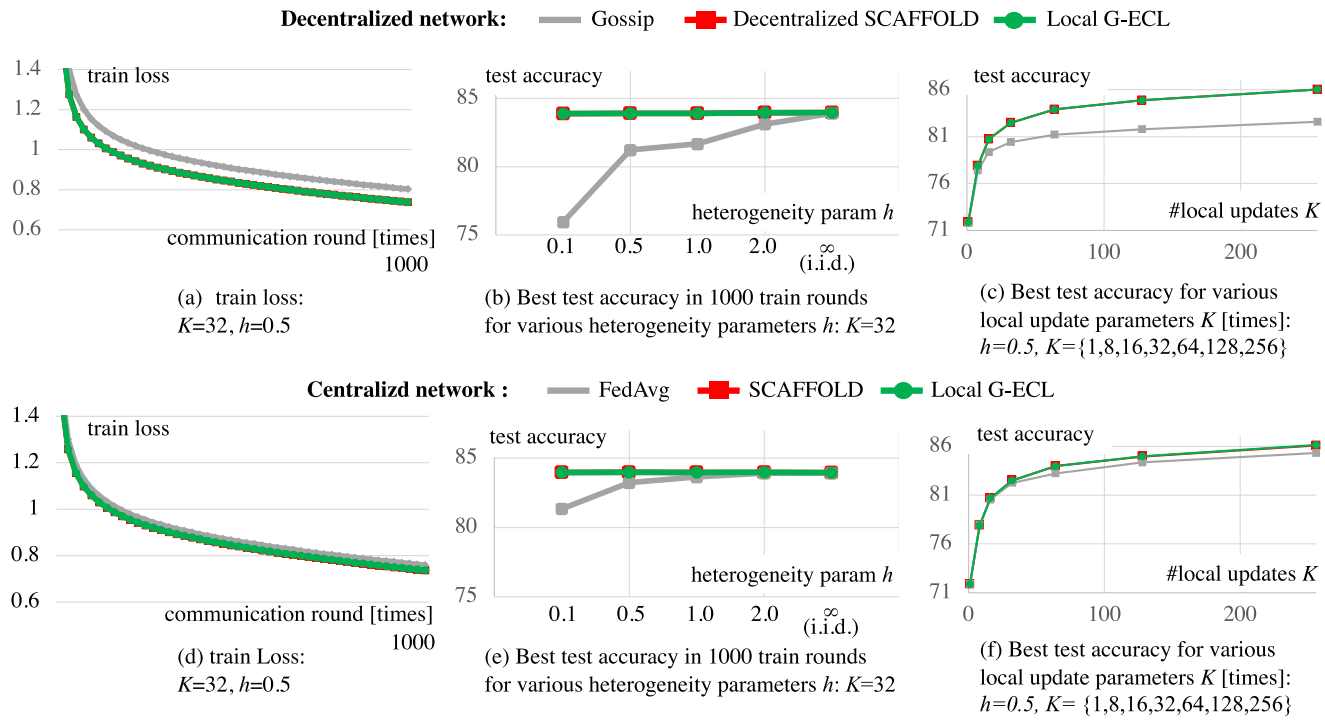
b) Centralized settings: In centralized cases, the communication complexity of Local G-ECL is almost equal to that of SCAFFOLD [12]. This is natural because Proposition 2 shows that our algorithm is nearly equivalent to SCAFFOLD in centralized cases. In particular, Local G-ECL inherits the robustness to heterogeneous data of SCAFFOLD and achieves better communication complexity than FedAvg [12].

V. Numerical Experiments

Numerical experiments using image classification benchmark tests are performed to validate our theoretical results. We prepared Local G-ECL for decentralized FL in Alg. 1 and that for centralized FL in Alg. 2, respectively. Through experiments on decentralized and centralized settings, we will show (i) robustness to data heterogeneity with local updates by comparing Local G-ECL vs. average consensus methods (Gossip and FedAvg), and (ii) convergence curve trend equivalence between primal-dual Local G-ECL and SCAFFOLD in centralized settings.

A. Experimental Setups

a) Network, data set, and its heterogeneous allocation: We prepared both decentralized and centralized settings obeying $n = 10$ workers. For decentralized settings, we chose a bidirectional ring architecture (i.e., the number of nodes connected to each node is $|\mathcal{N}_i| = 2$) and the mixing matrix parameter in Assumption 1 is then $p < 1.0$, while $p = 1.0$ for centralized



(a) train loss: K=32, h=0.5

(b) Best test accuracy in 1000 train rounds for various heterogeneity parameters h: K=32

(c) Best test accuracy for various local update parameters K [times]: h=0.5, K={1,8,16,32,64,128,256}

(d) train Loss: K=32, h=0.5

(e) Best test accuracy in 1000 train rounds for various heterogeneity parameters h: K=32

(f) Best test accuracy for various local update parameters K [times]: h=0.5, K= {1,8,16,32,64,128,256}

Fig. 1. Experimental results for FashionMNIST classification tests using a two-layer perceptron.

settings. We solved image classification problems using the FashionMNIST data set [29] by fitting a non-convex two-layer perceptron model.[5] Then, the cross entropy loss was used. For heterogeneous data allocation, we used latent Dirichlet allocation to partition 60,000 training images for $n = 10$ workers, e.g., [30], [31], where the number of local data subsets was allowed to be different for each worker. By discretely varying the concentration parameter as $h = \{0.1, 0.5, 1.0, 2.0, \infty\}$ in the Dirichlet distribution, data heterogeneity was controlled. Although h cannot be directly associated with $\{G, B\}$ in Assumption 5 and $\zeta_{1,0}$ in Corollary 5 and 6, data heterogeneity can be increased by approaching h to 0. In total, $R = 400$ communication rounds were performed. For each communication round r, all workers are synchronously communicating with each other, where K is discretely varied as $K = \{1, 8, 16, 32, 64, 128, 256\}$ to investigate the robustness to local updates. Note that Local G-ECL when $K = 1$ shrinks to the previous G-ECL for decentralized settings.

b) Algorithms and hyper-parameters: For each network setting, the following methods were tested. For decentralized settings, we tested three methods, namely Gossip vs. decentralized SCAFFOLD v.s. Local G-ECL (Alg. 1). For Local G-ECL, we chose $\lambda_i^{(0)} = 0$ and $\alpha_{i|j} = 0$ to expect fast convergence by setting $\psi = 0$ in Corollary 5.[6] Meanwhile, for centralized settings, we tested three methods, namely FedAvg v.s. SCAFFOLD (option II) vs. Local G-ECL (Alg. 2) with $\lambda_i^{(r)} = 0$, where client sampling was not applied.

The learning rate was determined by testing with several learning rates $\eta = \{0.2/4^i : i \in \{0, \ldots, 5\}\}$, and we chose it to give the highest accuracy for each method. The mini-batch size was set to 128.

B. Experimental Results

a) Convergence speed: Convergence curves for training loss for each network setting are shown in Fig. 1(a) and (d), respectively. From Fig. 1(a) and (d), compared with average consensus methods (Gossip, FedAvg), the training losses of Local G-ECL were rapidly reduced for each network setting. Meanwhile, there was little difference in the trend of the convergence curves between (decentralized) SCAFFOLD and Local G-ECL. Comparing Fig. 1(a) and (d), more stable and faster convergence was obtained on the centralized network. This is because the centralized algorithms are equivalent to the decentralized ones with fully-connected topology and Metropolis-Hastings weights (i.e., $W_{ij} = 1/n$), and the mixing parameter p in Assumption 1 is then maximized to 1. Also, from Fig. 1(d), the performances of Local G-ECL and SCAFFOLD were almost equivalent. This is because centralized Local G-ECL (Alg. 2) is nearly equivalent to SCAFFOLD (Alg. 3) as shown in Proposition 2.

b) Effect of data heterogeneity: In Fig. 1(b) and (e), robustness to data heterogeneity h are investigated. The dots in SCAFFOLD that are not visible are due to the fact that their results were almost identical with Local G-ECL. When data heterogeneity was increased by approaching h to 0, high test accuracy was maintained with Local G-ECL and (decentralized) SCAFFOLD. Meanwhile, test accuracy with average consensus methods (Gossip and FedAvg) decreased as data heterogeneity

[5]The details of the architecture are found in Appendix A. We also tested another model to validate the effectiveness of the proposed algorithms.

[6]Experiments investigating the effectiveness of warm-start setting of $\lambda_i^{(0)}$ in Corollary 6 are summarized in Appendix A.

TABLE II
COMPUTING ENVIRONMENTS

Worker process number	10
Machine spec	CPU=Xeon, GPU=RTX3080*2
Centralized network configuration	1 central server for aggregation/averaging + 10 local workers
Decentralized network configuration	10 local workers with 1 server for communication synchronization
Frameworks	Python (3.8.8), PyTorch(1.7.1), gRPC(1.36.1)[7],CUDA(11.4)
Mini-batch size	128
Loss function	Cross entropy loss with L_2 weight decay regularization (coefficient: 0.005)

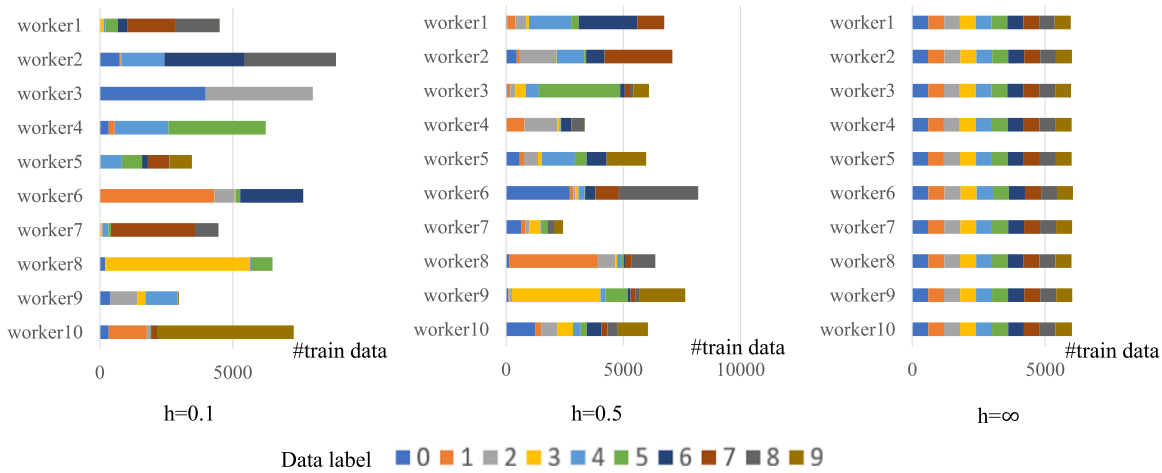


Fig. 2. Heterogeneous data allocations when $h = \{0.1, 0.5, \infty\}$.

increased. These results support that their convergence rates require Assumption 5, as noted in Table I.

c) Effect of the number of local updates: In Fig. 1(c) and (f), robustness to the client drift due to the local updates was investigated. When a large number of local updates was chosen ($K \geq 64$), high test accuracy was maintained with Local G-ECL and (decentralized) SCAFFOLD, while test accuracy was decreased when $K = 1$ (then Local G-ECL shrinks to G-ECL). The differences in test accuracy due to local updates K are explainable using convergence rate analysis in Table I. Since the third and fourth terms in the convergence rate of Local G-ECL depend on K and its effect can be reduced by increasing K, the learned model would be closer to its optimal point.

In summary, our theoretical findings are well-supported by the numerical experimental results.

VI. CONCLUSION

Local G-ECL was proposed for decentralized/centralized FL robust to data heterogeneity with a large number of local updates. First, our theoretical analysis showed that the convergence rates of Local G-ECL are nearly independent of data heterogeneity (see Table I). Second, we found that a pure primal SCAFFOLD (Alg. 3) and a primal-dual Local G-ECL for centralized setting (Alg. 2) are equivalent, ignoring differences in the initial points of the local updates. Through numerical experiments using image classification tests, Local G-ECL was found to be robust to data heterogeneity with a large number of local updates, as with (decentralized) SCAFFOLD for both decentralized and centralized settings.

APPENDIX
ADDITIONAL NUMERICAL EXPERIMENTS

A. Experimental Setups

To support experimental reproducibility, we provide a part of the latest source code as supplementary material. The computing environment is summarized as Table II:

As explained in Subsec.V-A, 60,000 Fashion-MNIST image samples were heterogeneously allocated over $n = 10$ local workers. Examples of heterogeneous data allocations when $h = \{0.1, 0.5, \infty\}$ are shown in Fig. 2. When $h = \infty$ is selected, data is homogeneously allocated; specifically, each local worker has 600 image samples for each class. Segregated from training data sets, test data sets with the homogeneous allocation (1,000 image samples for each class) were prepared. All the local workers (in decentralized settings) and the central server (in centralized settings) accessed these test data sets, and evaluation scores using accuracy and loss were calculated for every 10 communication rounds.

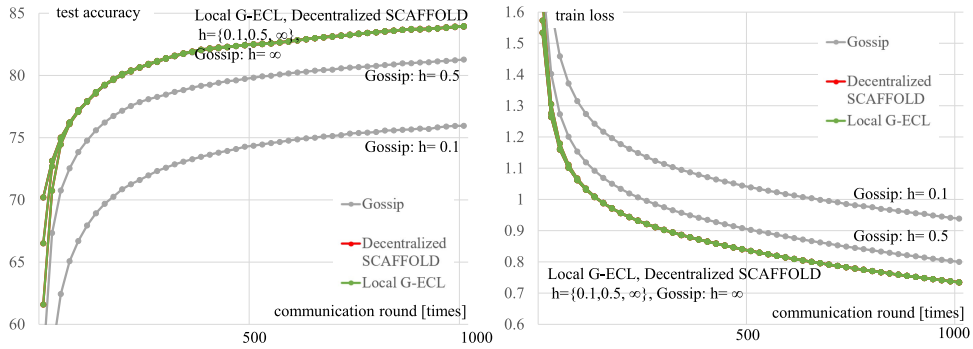
B. Details of Numerical Experiments in Section V

In Section V, a two-layer perceptron was used to solve the image classification problem. Its architecture consists of 1 hidden layer with 500 neurons and ReLU activation. Table III shows the best test accuracy in 1,000 rounds of each learning rate

TABLE III
PRE-EXPERIMENTAL RESULTS FOR TUNING LEARNING RATE η USING BEST TEST ACCURACY FOR EACH METHOD WHILE HYPER-PARAMETERS WERE SET AS $(h, K, R) = (0.1, 64, 1000)$ AND A TWO-LAYER PERCEPTION MODEL WAS USED

Learning rate η	0.2	0.05	0.0125	0.00312	0.000781	0.000195
Local G-ECL	73.44	76.65	80.24	83.20	**83.95**	76.58
Decentralized SCAFFOLD	73.30	76.75	79.58	83.25	**83.95**	76.58
Gossip	66.37	70.83	73.65	74.02	**75.95**	71.72

The maximum accuracy is bolded.



(a) test accuracy: K=64, ring network, h={0.1, 0.5, ∞ }

(b) train loss: K=64, ring network, h={0.1, 0.5, ∞ }

Fig. 3. Convergence curves using (a) test accuracy and (b) train loss when data heterogeneity parameter is changed as $h = \{0.1, 0.5, \infty\}$ while hyper-parameters are set as $(K, R) = (64, 1000)$. With Local G-ECL and decentralized SCAFFOLD methods, convergence curves overlap even when h is changed. With the Gossip method, resulting scores (test accuracy, train loss) were degraded when data heterogeneity was increased (by approaching h to zero).

η at $h = 0.1$ for each method. For all methods, we selected $\eta = 0.000781$ for this two-layer perceptron model. For centralized settings summarized in Section V, we commonly used $\eta = 0.000781$.

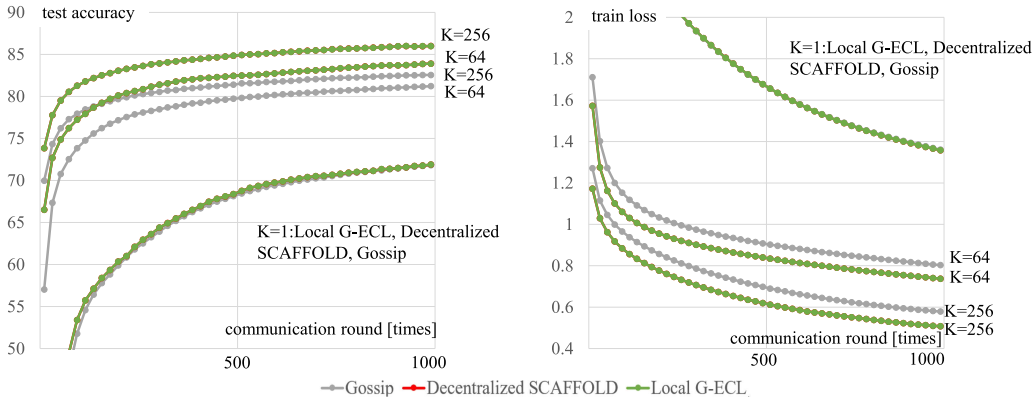
1) Investigation of Convergence Curve Trends by Changing Data Heterogeneity Parameter h: Fig. 3 shows the convergence curves using (a) test accuracy and (b) train loss for decentralized settings over the ring topology, where heterogeneity parameters were restricted as $h = \{0.1, 0.5, \infty\}$ for simply showing convergence curve trends. The number of local updates was fixed by $K = 64$ and the number of total communication rounds was fixed by $R = 1,000$. In Fig. 3, convergence curves (both training loss and test accuracy) when using Local G-ECL and decentralized SCAFFOLD overlap even when the data heterogeneity parameter is changed as $h = \{0.1, 0.5, \infty\}$. Fig. 1(b) shows that resulting scores (test accuracy) at the last communication round $R = 1,000$ were nearly independent of data heterogeneity parameter h. Since convergence curves (test accuracy, train loss) for each communication round overlap even when heterogeneity parameters are changed as $h = \{0.1, 0.5, \infty\}$ in Fig. 3, experimental validation showing independence of convergence trends toward to h in Fig. 1(b) is not coincidental. Although theoretical convergence analysis in decentralized SCAFFOLD depends on data heterogeneity as shown in Table I, it is experimentally shown that convergence curves with decentralized SCAFFOLD were nearly independent of data heterogeneity.

Meanwhile, the convergence curves with the Gossip method were relatively sensitive to data heterogeneity. By making h approach to zero to increase data heterogeneity, resulting scores (test accuracy, train loss) were degraded. This supports the theoretical analysis in Table I that the convergence rate of the Gossip method needs Assumption 5 to bound gradient similarity.

2) Investigation of Convergence Curve Trends by Changing the Number of Local Updates K: Fig. 4 shows the convergence curves using test accuracy and train loss over a decentralized setting (ring topology) for each method (Local G-ECL, decentralized SCAFFOLD, Gossip) by changing the number of local updates as $K = \{1, 64, 256\}$. Although only the best test accuracy is used to show the dependency toward K for each method in Fig. 1, Fig. 4 shows overall convergence curve trends. The trends shown in Fig. 1 are found to be maintained for each communication round.

By increasing K with all methods (Local G-ECL, decentralized SCAFFOLD, Gossip), improvements in the resulting scores (test accuracy, train loss) for each communication round were increased. This was caused by the number of model update iterations (RK) being different. Since resulting score improvements are not significantly increased when increasing $K \geq 64$, we fixed $K = 64$ for other hyper-parameter turnings.

3) Investigation of Convergence Curve Trends by Changing Network Topologies: Fig. 6 shows convergence curves using test accuracy and train loss for each method by changing network topology, where a data heterogeneity parameter was fixed by $h = 0.5$. For decentralized settings, three topologies were prepared, composed of ring ($|\mathcal{N}_i| = 2$), double ring ($|\mathcal{N}_i| = 4$), and fully-connected ($|\mathcal{N}_i| = n - 1 = 9$) as in Fig. 5. In addition, we prepared the centralized setting composed of a central server and $n = 10$ local workers. The mixing parameter p in Assumption 1 depends on network topology; specifically, a denser network makes p approach to 1.0. Since many of the



(a) test accuracy: h=0.5, ring network, K={1, 64, 256}

(b) train loss: h=0.5, ring network, K={1, 64, 256}

Fig. 4. Convergence curves using test accuracy (a) and train loss (b) when the number of local updates is changed as $K = \{1, 64, 256\}$ while hyper-parameters are set as $(h, R) = (0.5, 1000)$.

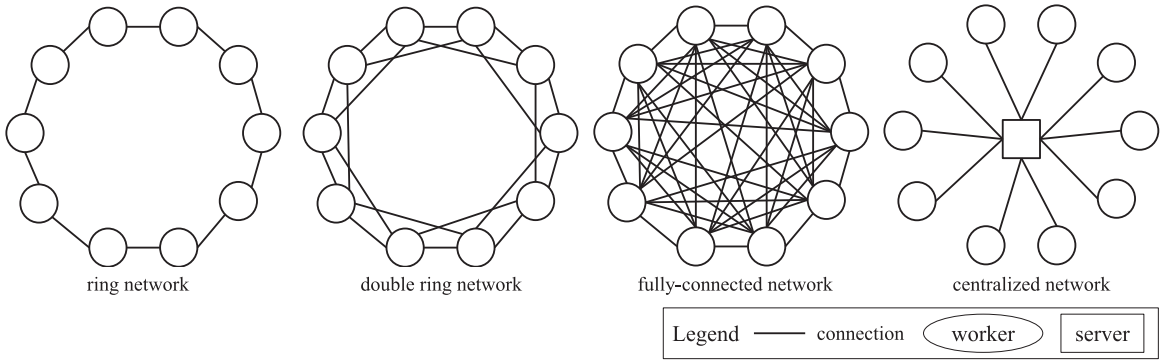
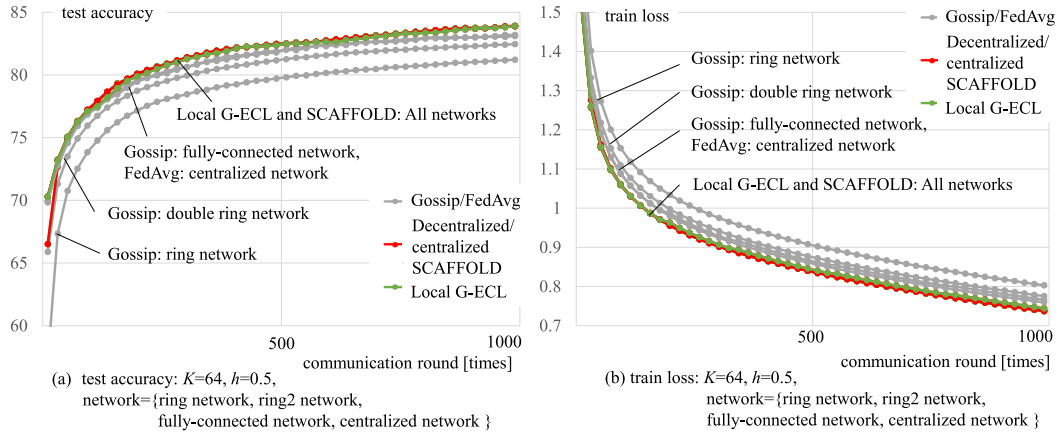


Fig. 5. Network topologies of ring ($|\mathcal{N}_i| = 2$), double ring ($|\mathcal{N}_i| = 4$), fully-connected ($|\mathcal{N}_i| = 9$), and the centralized network.



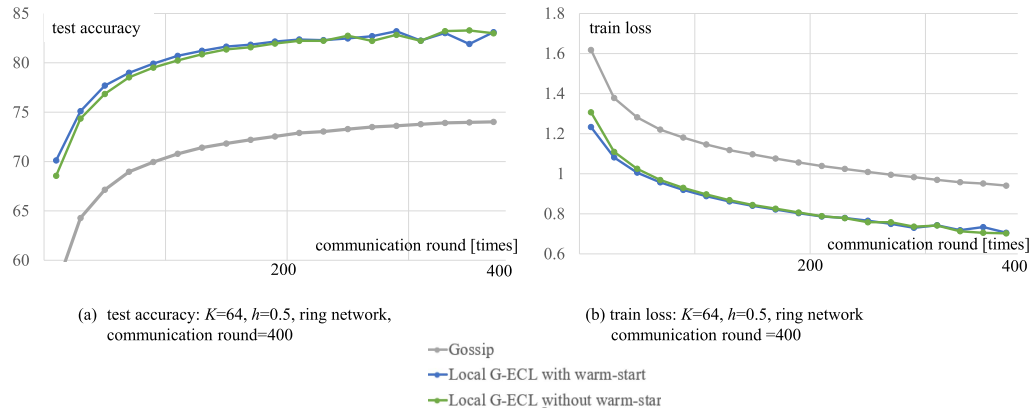
(a) test accuracy: K=64, h=0.5, network={ring network, ring2 network, fully-connected network, centralized network }

(b) train loss: K=64, h=0.5, network={ring network, ring2 network, fully-connected network, centralized network }

Fig. 6. Convergence curves using (a) test accuracy and (b) train loss for each network topology; i.e., three decentralized networks composed of ring ($|\mathcal{N}_i| = 2$), double ring ($|\mathcal{N}_i| = 4$), and fully-connected ($|\mathcal{N}_i| = 9$) and the centralized network. Associated hyper-parameters were set as $(h, K, R) = (0.5, 64, 1000)$. With Local G-ECL and decentralized SCAFFOLD, convergence curves are overlapped for all network topologies. Although the effect of network topology did not appear, it may be due to the small size of networks ($n = 10$).

convergence analyses summarized in Table I including our Local G-ECL are dependent on p, its effect on convergence curves is experimentally investigated.

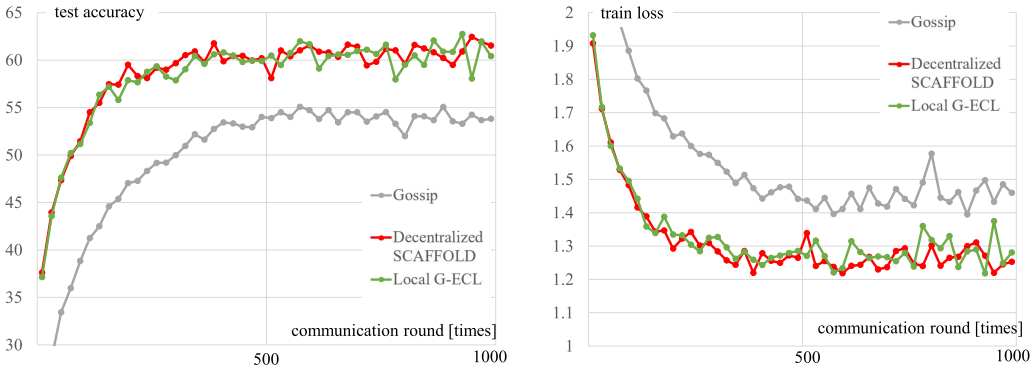
Fig. 6 shows that convergence curves with Local G-ECL and decentralized SCAFFOLD overlap even though network topologies are changed. We consider that this weak effect of p would be presumably due to the small size of networks using 10 local workers. In addition, experimental results using Local G-ECL support that the convergence curve with a fully-connected decentralized network ($p = 1.0$) and that with the centralized network ($p = 1.0$) are equivalent (Since stochastic local data sampling is difficult to control in these two network settings,



(a) test accuracy: K=64, h=0.5, ring network, communication round=400

(b) train loss: K=64, h=0.5, ring network communication round =400

Fig. 7. Convergence curves using (a) test accuracy and (b) train loss using Local G-ECL with/without warm-start, and Gossip as a reference. Associated hyper-parameters are set as $(h, K, R) = (0.5, 64, 400)$.



(a) test accuracy of LeNet: K=64, h=0.5, ring network

(b) train loss of LeNet: K=64, h=0.5, ring network

Fig. 8. Convergence curves using (a) test accuracy and (b) train loss of LeNet, where hyper-parameters were set as $(h, K, R) = (0.5, 64, 1000)$, and the ring topology with $n = 10$ workers was used.

slight differences existed in convergence curves. However, these differences do not significantly impact our conclusion; that is, experimental results match our theory in Section III-B.)

4) Investigation of Convergence Curve Trends With Warm-Start Initialization Setting: The effect of warm-start initialization discussed in Section IV-B is investigated. Let us recall that the warm-start initialization is proposed to reduce the effect of data heterogeneity at the initial point $\zeta_{1,0}$. For this investigation, $\eta = 0.00312$ was selected. Experimental results using Local G-ECL with/without warm-start setting, $h = 0.5$, and $K = 64$ are summarized in Fig. 7. Although slight differences appeared in their convergence curves, especially in early communication rounds, overall curve trends were maintained regardless of the warm-start initialization setting. The resulting scores using Local G-ECL with/without warm-start were much higher than those of Gossip.

C. Experimental Validation Using Other Models

To validate the effectiveness of the proposed algorithms, we conducted experiments using non-convex LeNet [32]. LeNet [32] is a neural network that consists of two convolution layers, three hidden layers, and ReLU activation. Fig. 8 shows experimental results for CIFAR10 [33] classification tests using LeNet, which shows that Local G-ECL has advantages over the Gossip method (higher test accuracy, lower loss) and is almost as robust as SCAFFOLD against data heterogeneity.

Experiments have shown that the proposed Local G-ECL is robust to data heterogeneity and effective independent of neural models, whether in two-layer perceptron or logistic regression or in networks containing convolutional layers such as LeNet.

D. Evaluation of Computational Complexity

The computational complexity of Local G-ECL is shown in Table IV, consisting of training time and traffic volume when training two-layer MLP for CIFAR10 recognition for each method.

Table IV shows that the traffic volume when using SCAFFOLD requires twice as much as other methods under centralized conditions since it sends $c, \widetilde{c}$. On the other hand, Centralized Local G-ECL reduces the traffic volume since it does not send $\bar{\nu}$ or λ and is almost equivalent to FedAvg. However, in a decentralized setting, Local G-ECL cannot omit to send $\bar{\nu}$; thus, the traffic volume is almost the same between Decentralized SCAFFOLD and Decentralized Local G-ECL.

TABLE IV
THE COMPUTATIONAL COMPLEXITY WHEN USING TWO-LAYER PERCEPTRON USING CIFAR10, 10 WORKERS OVER RING NETWORK, AND ASSOCIATED HYPER-PARAMETERS ARE SET BY $(h, K, R) = (0.5, 128, 400)$

	Training time [sec]	Traffic volume [GiB]
FedAvg	3,410	95
SCAFFOLD	3,592	189
Centralized Local G-ECL	3,624	98
Gossip	3,402	96
Decentralized SCAFFOLD	3,456	188
Decentralized Local G-ECL	3,512	188

In addition, Table IV shows that training time for each method is almost equivalent. This may be caused by the fact that we conducted the experiments using GPUs mounted on one server; specifically, experiments were not affected by network latency and volume.

REFERENCES

[1] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.

[2] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, and H. Yu, "Federated learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 13, no. 3, pp. 1–207, 2019.

[3] N. Rieke et al., "The future of digital health with federated learning," *NPJ Digit. Med.*, vol. 3, no. 1, pp. 1–7, 2020.

[4] C. Zheng, S. Liu, Y. Huang, W. Zhang, and L. Yang, "Unsupervised recurrent federated learning for edge popularity prediction in privacy-preserving mobile-edge computing networks," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 24328–24345, Dec. 2022.

[5] C. Xu, S. Liu, Z. Yang, Y. Huang, and K.-K. Wong, "Learning rate optimization for federated learning exploiting over-the-air computation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3742–3756, Dec. 2021.

[6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," *Artif. Intell. Statist.*, PMLR, 2017.

[7] P. H. Jin, Q. Yuan, F. Iandola, and K. Keutzer, "How to scale distributed deep learning?," 2016, *arXiv:1611.04581*.

[8] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[9] X. Lian, W. Zhang, C. Zhang, and J. Liu, "Asynchronous decentralized parallel stochastic gradient descent," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3043–3052.

[10] H. Xing, O. Simeone, and S. Bi, "Decentralized federated learning via SGD over wireless D2D networks," in *Proc. IEEE Workshop Signal Process. Adv. Wireless Commun.*, 2020, pp. 1–5.

[11] B. Ying, K. Yuan, Y. Chen, H. Hu, P. Pan, and W. Yin, "Exponential graph is provably efficient for decentralized deep training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 13975–13987.

[12] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5132–5143.

[13] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized SGD with changing topology and local updates," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 5381–5393.

[14] Y. Liu, S. U. Stich, T. Lin Roger Wattenhofer, and M. Jaggi, "Variance reduction in decentralized training over heterogeneous data," M.S. thesis, Comput. Eng. Netw. Lab., ETH Züürich, Züürich, Switzerland, 2021. [Online]. Available: https://pub.tik.ee.ethz.ch/students/2020-HS/MA-2020-32.pdf

[15] Y. Takezawa, K. Niwa, and M. Yamada, "Theoretical analysis of primal-dual algorithm for non-convex stochastic decentralized optimization," 2022, *arXiv:2205.11979v2*.

[16] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013.

[17] L. Xiao and T. Zhang, "A proximal stochastic gradient method with progressive variance reduction," *SIAM J. Optim.*, vol. 24, no. 4, pp. 2057–2075, 2014.

[18] A. Defazio, F. Bach, and S. Lacoste-Julien, "SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014.

[19] A. Cutkosky and F. Orabona, "Momentum-based variance reduction in non-convex SGD," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019.

[20] S. P. Karimireddy et al., "Mime: Mimicking centralized stochastic algorithms in federated learning," *CoRR*, 2020, *arXiv:200803606*.

[21] S. Boyd et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[22] R. Pathak and M. J. Wainwright, "FedSplit: An algorithmic framework for fast federated optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 7057–7066.

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

[24] 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.

[25] 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. Discov. Data Mining*, 2020, pp. 668–678.

[26] K. Niwa, G. Zhang, W. B. Kleijn, N. Harada, H. Sawada, and A. Fujino, "Asynchronous decentralized optimization with implicit stochastic variance reduction," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 8195–8204. [Online]. Available: https://proceedings.mlr.press/v139/niwa21a.html

[27] P. D. Lorenzo and G. Scutari, "NEXT: In-network nonconvex optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 120–136, Jun. 2016.

[28] A. Koloskova, T. Lin, and S. U. Stich, "An improved analysis of gradient tracking for decentralized machine learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 11422–11435.

[29] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

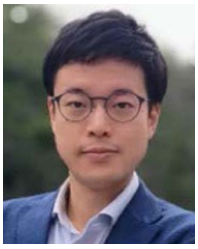
[30] C. He et al., "FEDCV: A federated learning framework for diverse computer vision tasks," 2021, *arXiv:2111.11066*.

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

[32] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[33] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (Canadian institute for advanced research)," 2009. [Online]. Available: http://www.cs.toronto.edu/kriz/cifar.html

Iifan Tyou (Member, IEEE) received the B.E. degree in information engineering and M.E. degree in engineering from the Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba, Japan, in 2008 and 2010. Since 2023, he is working toward the Ph.D. degree in information science and engineering with the Graduate School of Information Science and Technology, the University of Tokyo, Tokyo, Japan. He is currently with NTT Social Informatics Laboratories. Since joining NTT in 2010, he has been engaged in the research and development of traceability systems and Internet of Things gateway security systems. His research interests include decentralized platforms such as blockchain, cryptography, data distribution platform, and federated learning.



Tomoya Murata received the B.S. degree in science from the Tokyo University of Science, Tokyo, Japan, in 2015, the M.S. degree in science from the Tokyo Institute of Technology, Tokyo, in 2017, and the Ph.D. degree in information science and engineering from the University of Tokyo, Tokyo, in 2023. Since joining NTT Data Mathematical Systems Inc. in 2017, he has been engaged in the research and development of various machine learning techniques. His research interests include statistical learning theory, stochastic optimization, and federated learning.

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

Yuki Takezawa received the B.E. degree in 2021 in engineering from the Undergraduate School of Informatics and Mathematical Science, the M.E. degree in informatics in 2023 from the Graduate School of Informatics from Kyoto University, Kyoto, Japan, where he currently working toward the Ph.D. degree in informatics with the Graduate School of Informatics. From 2021 to 2023, he was with RIKEN AIP as a Research Assistant. He also a Visiting Research Student with the Okinawa Institute of Science and Technology, Okinawa, Japan. His research interests include machine learning, distributed optimization, and optimal transport theory.

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