Privacy-Preserving and Fairness-Aware Federated Learning for Critical Infrastructure Protection and Resilience

Yanjun Zhang
University of Technology Sydney
Australia

Ruoxi Sun
CSIRO's Data61
Australia

Liyue Shen
The University of Queensland
Australia

Guangdong Bai
The University of Queensland
Australia

Minhui Xue
CSIRO's Data61
Australia

Mark Huasong Meng
Institute for Infocomm Research,
A*STAR
Singapore

Xue Li
The University of Queensland
Australia

Ryan Ko
The University of Queensland
Australia

Surya Nepal
CSIRO's Data61
Australia

ABSTRACT

The energy industry is undergoing significant transformations as it strives to achieve net-zero emissions and future-proof its infrastructure, where every participant in the power grid has the potential to both consume and produce energy resources. Federated learning – which enables multiple participants to collaboratively train a model without aggregating the training data – becomes a viable technology. However, the global model parameters that have to be shared for optimization are still susceptible to training data leakage. In this work, we propose Confined Gradient Descent (CGD) that enhances the privacy of federated learning by eliminating the sharing of global model parameters. CGD exploits the fact that a gradient descent optimization can start with a set of discrete points and converges to another set in the neighborhood of the global minimum of the objective function. As such, each participant can independently initiate its own private global model (referred to as the *confined model*), and collaboratively learn it towards the optimum. The updates to their own models are worked out in a secure collaborative way during the training process. In such a manner, CGD retains the ability of learning from distributed data but greatly diminishes information sharing. Such a strategy also allows the proprietary confined models to adapt to the heterogeneity in federated learning, providing inherent benefits of fairness. We theoretically and empirically demonstrate that decentralized CGD (*i*) provides a stronger differential privacy (DP) protection; (*ii*) is robust against the state-of-the-art poisoning privacy attacks; (*iii*) results in bounded fairness guarantee among participants; and (*iv*) provides high test accuracy (comparable with centralized learning) with a bounded convergence rate over four real-world datasets.

CCS CONCEPTS

• **Security and privacy**; • **Computing methodologies** → **Machine learning**;

KEYWORDS

Federated Learning, Privacy Preserving, Fairness

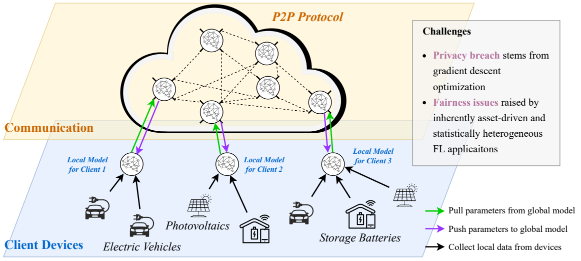
ACM Reference Format:
Yanjun Zhang, Ruoxi Sun, Liyue Shen, Guangdong Bai, Minhui Xue, Mark Huasong Meng, Xue Li, Ryan Ko, and Surya Nepal. 2024. Privacy-Preserving and Fairness-Aware Federated Learning for Critical Infrastructure Protection and Resilience. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3589334.3645545

1 INTRODUCTION

Federated learning (FL) empowers participants to jointly train a global model without the necessity of sharing their datasets [20], demonstrating the potential promise of communication efficiency and protecting proprietary user data. It has been incorporated by popular machine learning frameworks, and the burgeoning development of FL variance renders it promising to be deployed across various industries [10, 12, 34, 43], including critical infrastructure such as energy sector [10, 12, 34]. With the widespread adoption of distributed renewable energy resources such as electric cars, battery storage, wind turbines, and solar panels, managing the distributed network through leveraging FL technology has become a complex challenge. Every participant in the network now has the potential to both consume and produce energy resources, making it crucial to maintain security and privacy, equitability, and resilience in the distributed network.

Challenges. Standard FL approaches, however, have been shown vulnerable to privacy leakage which stems from its parallelization of the gradient descent optimization [29, 33]. During the training process, each participant obtains the current global model parameters, works out a local gradient based on the local data, and disseminates it for others to update the global model. This paradigm guarantees data locality, but the shared local models/gradients imply sensitive attributes of local training samples, as revealed by recent

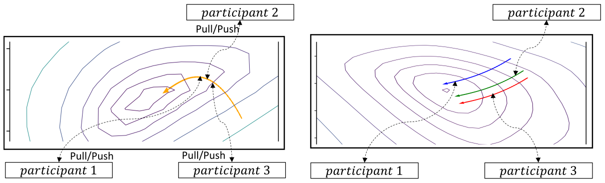
WWW '24, May 13–17, 2024, Singapore, Singapore
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0171-9/24/05.
https://doi.org/10.1145/3589334.3645545



Challenges

- Privacy breach stems from gradient descent optimization
- Fairness issues raised by inherently asset-driven and statistically heterogeneous FL applicaitons

Figure 1: An example of federated learning application in the context of critical infrastructure.



(a) (b)

(a) Gradient descent in traditional federated learning in which participants jointly work on the same global model; (b) Confined Gradient Descent. Each participant strictly confines their own global models throughout the learning process. When the optimization algorithms converge, they reach the neighbourhood of the centralized model that was to learn centrally on the gathered data.

Figure 2: The optimization process of the traditional FL and CGD on MNIST [22] (both with three participants for illustration)

studies [40]. In addition, many FL applications are inherently asset-driven and statistically heterogeneous, resulting in fairness challenges. For example, without reconciliation of privacy and fairness in the context of sustainable critical infrastructure, *i.e.,* peer-to-peer (P2P) energy network (see Figure 1), privileged individuals, groups and organizations can have advantageous access to the power grid and benefit at the cost of underserved groups, substantiated by the fact that merely implementing differential privacy techniques may lead to disproportionate impact on minority communities [36].

Solution. To ensure private and equitable participation of community members and responsible use of their data in providing secure, reliable, and sustainable energy services through FL, this work aims to enhance the privacy preservation with bounded fairness guarantee in FL by *eliminating the explicit sharing of the centralized global model* to decrease the dependency among participants. We therefore propose a new optimization algorithm named Confined Gradient Descent (CGD) that enables each participant to learn its own private "global" model. The CGD participants initiate their own global models locally, and then strictly confine these models within themselves throughout the whole training process. The updates to their own models are operated in a secure collaborative way during the training process. In such a manner, CGD retains the merit of *federation* but greatly diminishes information leakage. In the remaining of this paper, we refer to these localized and private global models as *confined models*, to distinguish them from the global model in traditional FL. In addition, CGD guarantees that each confined model, when CGD converges, is at the neighborhood of the centralized model. This property, in principle, retains desirable model accuracy. Furthermore, this strategy allows the proprietary confined models to further adapt to the heterogeneity in locality, which provides inherent benefits of fairness.

Observation and Motivation. CGD is inspired by an observation on the surface of the typical cost function. The steepness of the first derivative decreases slower when approaching the minimum of the function, due to the small values in the Hessian (*i.e.,* the second derivative) near the optimum [7]. This gives the function, when plotted, a flat valley bottom. As such, a gradient descent algorithm $\mathcal{A}$, when applied on an objective function F, could start with a set of discrete points. Iteratively descending the set using their joint gradient would lead $\mathcal{A}$ to the neighborhood of F's minimum in the "flat valley bottom". Those points would also end up with similar losses that are close to the loss of the minimum.

From a high-level view,[1] we visualize in Figure 2 a comparison between the optimization process of CGD and that of a gradient decent in traditional FL on the MNIST dataset [22]. In traditional FL (Figure 2(a), every participant updates the *same* global model using their local gradients, while in CGD (Figure 2(b)), each participant learns its own confined model and hides its confined model from each other throughout the training process. CGD allows its participants to choose their own randomization strategy to initialize their confined models. For example, they can decide their own interval range of initial weights. In every training iteration, each participant independently computes the local gradient from its current confined model and local data, and then securely works out aggregated gradients so as to update their confined models. The optimization process of CGD endows FL with the capability of *advantageous privacy-utility tradeoff* and *bounded fairness* among participants.

Contributions. We summarize our main contributions as follows.

- **Confined Gradient Descent for Privacy-enhancing Federated Learning.** We propose a new optimization algorithm CGD for privacy-preserving decentralized FL. CGD eliminates the explicit sharing of the global model and lets each participant learn a proprietary confined model. Therefore, it can easily accommodate any FL scheme regardless of their underlying machine/deep learning algorithms. It also eliminates the necessity of a central coordinating server, such that the optimization can be conducted in a fully decentralized manner.
- **Enhanced Privacy Preservation over Traditional FL.** CGD further enhances the privacy in FL. First, we prove that CGD provides a stronger privacy guarantee (*i.e.,* in CGD, less noise is required) than the existing differentially private federated learning frameworks [1, 42]. We also experimentally demonstrate the advantages of CGD. For example, given test accuracy around 95% on the MNIST dataset, CGD achieves $(0.41, 10^{-5})$-DP, which is tighter than the existing methods reaching the same accuracy, *e.g.,* $(0.95, 10^{-5})$-DP in Bayesian DP [42], and $(2.0, 10^{-5})$-DP in Abadi *et al.* [1]. In addition, we also demonstrate CGD's robustness against the state-of-the-art poisoning membership inference attacks in FL [29, 33, 46]. Such attacks typically aim to infer the training samples' membership by injecting malicious attributions against the victim samples. CGD enables each local participant to examine its confined model's loss value and test accuracy with respect to its training samples. By doing this, the participant can effectively detect suspicious privacy attack against its training

[1] For the visualization purpose, we follow the techniques proposed in [23] to plot 2D loss landscape, including using principal component analysis (PCA) to get two principle directions as projection axis, and filter normalization to mitigate the loss value drifting caused by norm deviation.

Table 1: A summary of CGD and related privacy-preserving FL methods.

Techniques	Methodology		Desired Properties					
	Local Model Heterogeneity	Bounded Loss (Local Model)	Advantageous Privacy-Utility Tradeoff	Robustness		Convergence Guarantee	Bounded Fairness	Bounded Asynchrony
				against Passive MIA	against Active MIA			
Plain FL [20]	○	○	○	○	○	●	○	○
FL via Secure Aggregation [6, 27, 31, 38, 47]	○	○	◑	○	○	●	○	○
FL via Differential Privacy [1, 11, 17, 39, 48]	○	○	◑	●	○	◑[1]	○	○
CGD (Ours)	●	●	●	●	●	●	●	●

●: The corresponding item is supported; ◑: The corresponding item is partially supported; ○: The corresponding item is not supported.

[1] Additive noise (in most cases) affects the model accuracy [11].

samples and thus determine whether to set the new iterate using the current updates.

- **Bounded Fairness.** CGD enables a more fair solution for each participant through heterogeneous confined model and bounded asynchronous optimization, *i.e.,* participants are off synchrony up to a certain bounded delay. We first provide a proof to CGD fairness-convergence theorem to demonstrate that the distance between any of the confined models and the centralized model is bounded when CGD converges. Furthermore, we empirically show that CGD decreases the variance of the test accuracy across participants compared to the state-of-the-art fairness-aware FL [24].
- **Functional Evaluations.** We implement CGD and conduct experiments on four benchmark datasets. The results demonstrate that CGD consistently achieves superior privacy, accuracy and fairness performance in all settings. The performance also remains stable even with a relatively large number of participants (*e.g.,* 112,000 participants on the MNIST dataset and 32,000 participants on the CIFAR-10 dataset).

Open Science. To the best of our knowledge, the paper presents the first practical privacy-preserving and fairness-aware federated learning against malicious clients for critical infrastructure protection and resilience. We release the source code of CGD at https://github.com/CGD-release/cgd[2] to foster further research.

2 BACKGROUND AND RELATED WORK

In this section, we introduce the background and recent works that are related to our study.

2.1 Privacy-preserving Federated Learning

In Federated Learning, each local participant $l \in \mathcal{L}$ holds a subset of the training samples privately, denoted by ξ_l. During each iteration, a random subset $\xi_{l,k} \subseteq \xi_l$ from a random participant l is selected (*e.g.,* in FedSGD [20]). The participant l then computes the gradient ∇_k^l with respect to $\xi_{l,k}$, and communicate the gradient with the aggregator (server). All participants (or the aggregator) can thus take a gradient descent step by

$$w_{k+1} \leftarrow w_k - \alpha_k \frac{1}{|\xi_{l,k}|} \nabla_k^l. \tag{1}$$

The existing privacy-preserving methods for FL mainly fall into two broad categories, *i.e., secure aggregation* and *differential privacy.*

Secure Aggregation. Secure aggregation typically employs cryptographic mechanisms (*e.g.,* homomorphic encryption (HE) [27, 38]) and/or secure multiparty computation (MPC) [6, 31, 47] to securely work out the gradient $\nabla F(w_k, \xi_{l,k})$ without revealing local gradients. Each participant obtains the $\nabla F(w_k, \xi_{l,k})$ only, rather than the local gradient of any other participant, and then all participants take a gradient descent step by Equation 1.

Learning with Differential Privacy (DP). One of the strongest privacy standards that can protect FL models from a variety of privacy attacks is differential privacy [1, 11, 17, 39, 45, 48]. Here we employ the notion of (ϵ, δ)-differential privacy which is commonly used in machine learning.

Definition 1. ***(ϵ, δ)-differential privacy.*** *A randomized mechanism $\mathcal{M} : \mathcal{D} \rightarrow \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ satisfies (ϵ, δ)-differential privacy if for any two adjacent inputs $d, d' \in \mathcal{D}$ and for any subset of outputs $S \subseteq \mathcal{R}$ it holds that*

$$Pr[\mathcal{M}(d) \in S] \leq e^{\epsilon} Pr[\mathcal{M}(d') \in S] + \delta. \tag{2}$$

The definition permits the possibility that plain ϵ-differential privacy can be broken with a preferably subpolynomially small probability δ when applying the Gaussian noise mechanism [1, 11]. The common practice of achieving differential privacy in FL is to add noise calibrated to the local updates, *i.e.,* ∇F's sensitivity $\mathcal{S}^2_{\nabla F}$. As such, a differentially private learning framework can be achieved by updating parameters with perturbed local gradients at each iteration, for example, to update parameters as

$$w_{k+1} \leftarrow w_k - \alpha_k \frac{1}{|\xi_{l,k}|} (\nabla F(w_k, \xi_{l,k}) + \mathcal{N}(0, \mathcal{S}^2_{\nabla F} \cdot \sigma^2)), \tag{3}$$

where $\mathcal{N}(0, \mathcal{S}^2_{\nabla F} \cdot \sigma^2)$ is the Gaussian distribution with mean 0 and standard deviation $\mathcal{S}_{\nabla F} \cdot \sigma$.

Recent studies introduced user-level differential privacy in federated learning [3, 28], which ensures an adversary cannot distinguish the presence/absence of *all* the data belonging to a particular participant. The guarantee relies on a particular concept of *user-adjacent datasets*, meaning that two datasets are considered user-adjacent if one can be derived by including or excluding *all* examples associated with a single user from the other [28].

2.2 Fairness in FL

In the FL scenario, many factors can affect the fairness of the client, *e.g.,* the accuracy performance in the presence of high data heterogeneity. For example, one common approach to mitigating the straggler problem (FL training progress moves at the pace of the slowest clients) in synchronous federated learning is to begin training on a larger number of clients than necessary and subsequently remove the slowest clients from the round [5]. However, this may inherently cause a fairness problem, as the data from the straggler clients is not fully involved in the training phase. Recent research [37] looks into network-level attacks against FL models, such as packet drop attacks, where an attacker controls over the network traffic of multiple clients and further impacts the accuracy of the global model. We argue that such a network-level availability attack also poses a threat to the fairness of federated learning models by preventing data communication between victim clients and the model.

To mitigate this, personalized FL is an emerging line of research that aims to improve fairness. The paradigm allows participants to adapt the local models to fit their specific data distribution. Commonly used approaches include meta-learning (*i.e.,* training a global model while fine-tuning, regularizing, and/or interpolating the local models) [24, 25] and partially local methods (*i.e.,* partitioning the

[2] https://doi.org/10.5281/zenodo.10681651

model to a shared backbone component and the adaptive personal ones) [35]. In a recent work [24], Li *et al.* introduced Ditto, a scalable federated multi-task learning framework designed to promote fairness. It fosters proximity between personalized models and the optimal global model through local training with regularization. Following this, we provide a formal definition of *fairness* in FL.

Definition 2. *γ-**Fairness.** A federated learning system satisfies γ-fairness, if $std\{F_l(w)\}_{l\in\mathcal{L}} \leq \gamma$, where $F_l(\cdot)$ denotes the test loss on participant $l \in \mathcal{L}$, and $std\{\cdot\}$ denotes the standard deviation.*

Our work fundamentally differs from these approaches by eliminating the shared global model in FL. Our strategy instead allows the proprietary confined models to further adapt to the heterogeneity in locality and push up the peer-to-peer communication layer away from locality detailed as bounded asynchrony in § 5.1, which provides inherent benefits of fairness.

3 CONFINED GRADIENT DESCENT

3.1 Problem Formulation

Consider a centralized dataset $\xi = \{(x_i, y_i)\}_{i=1}^{n}$ consisting of n training samples. The goal of machine learning is to find a model parameter w such that the overall loss, which is measured by the distance between the model prediction $h(w, x_i)$ and the label y_i for each $(x_i, y_i) \in \xi$, is minimized. This is reduced to solving the following optimization problem: $\arg\min_w \frac{1}{n}\sum_{i=1}^{n} F(w;\xi) + \beta z(w)$, where $F(w;\xi)$ stands for the loss function, and $z(w)$ is the regularizer.

In the context of FL, we have a system of m local participants, each of which holds a private dataset $\xi_l \subseteq \xi$ ($l \in [1, m]$) consisting of a part of the training dataset. The part could be a part of training samples (*i.e.,* the *horizontal partition*), a part of features that have common entries (*i.e.,* the *vertical partition*), or both.

Optimization Objective of CGD. CGD starts with a set of discrete points $\{w_1^l\}$. Assume the training takes T iterations, and let w_k^l denote the confined model of participant l at the k^{th} iteration, where $k \in [1, T]$. Let

$$g^l(w_k^l, \xi_l) = \frac{1}{|\xi_l|}\nabla F(w_k^l, \xi_l) \qquad (4)$$

represent the local gradient with respect to ξ_l. The objective of CGD is to learn a separate confined model for each participant, by synthesizing the local gradients in a secure way. When CGD converges, participant l derives its own final confined model w_*^l. We use w_* to denote the optimal solution of centralized training (*i.e.,* the *centralized model*). CGD aims to make w_*^l located at a neighborhood of w_* within a bounded gap.

3.2 CGD Architecture

As shown in Figure 3(a), in traditional FL, every participant holds its local training dataset, and all of them update the *same* global model w_k via a parameter server using their local models/gradients. The local gradients $\nabla F(w_k, \xi_{l,k})$ can be protected via either secure aggregation [6, 27, 31, 38, 47] or differential privacy mechanisms [1, 11, 17, 39, 48].

As seen in CGD (Figure 3(b)), each participant learns its own confined model (represented by different colors), *i.e.,* each w_k^l is *different* and *private.* The model updating in CGD synthesizes the aggregated information by summing up the local gradients from a subset of participants. To better position CGD, we summarize and compare existing privacy-preserving federated learning approaches with our CGD in Table 1.

Desired properties. CGD should have the following properties:

- Advantageous privacy-utility tradeoff. CGD should not degrade the utility (*e.g.,* accuracy) while preserving the privacy of participants. Concretely, CGD pursues an advantageous tradeoff between privacy and model utility via *the secrecy of the confined models* (as detailed in Section 4).
- Robustness against membership inference attacks. Considering the emerging privacy threats introduced by membership inference attacks (MIAs), as a novel and practical federated learning approach, CGD guarantees its robustness against MIAs, including passive MIA and active MIA, through *the participant-side detect-then-drop strategy* (as detailed in Section 4).
- Fairness-awareness. CGD should provide a approach to achieving a uniform utility distribution, minimizing the standard deviation of loss across participants. CGD guarantees that each confined model, when CGD converges, is at the neighborhood of the centralized model. This property retains desirable model accuracy and fairness according to Definition 2 (as proved in Section 5).
- Asynchronous FL enabled. In traditional synchronous FL, progress moves at the synchronous pace. In CGD, we enable a mechanism to allow bounded asynchronous optimization, *i.e.,* slow participants can upload data in later rounds, and such asynchrony is up to a bounded delay for the fairness guarantee.

3.3 Threat Model of Privacy and Fairness Attacks

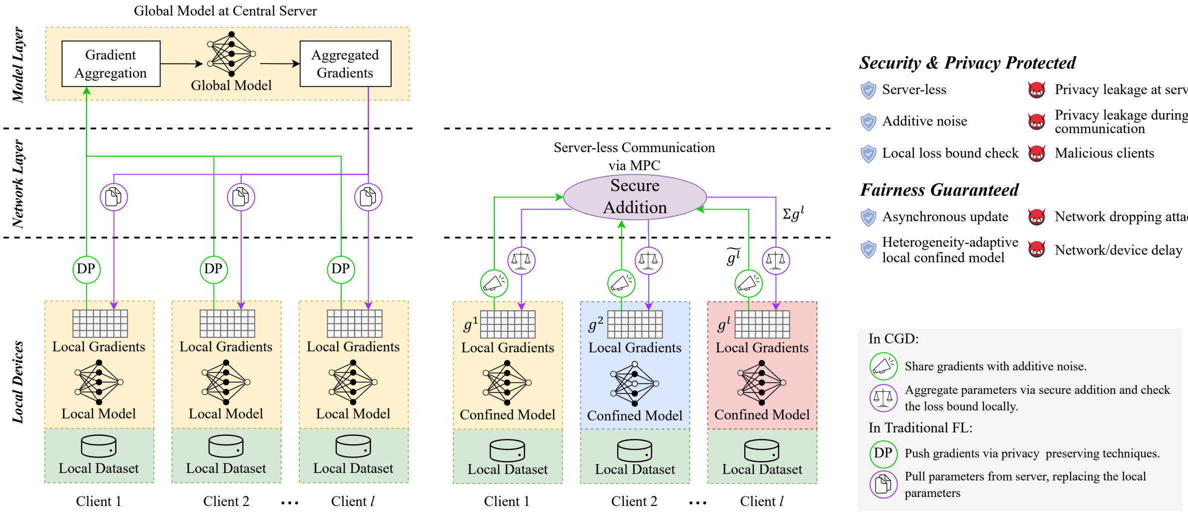
Threat model of privacy attacks. We consider a membership inference adversary who aims to infer whether a given data record was used to train the model. We include two representative modes of privacy attacks in CGD, *i.e.,* passive mode and active mode. In the passive mode, the adversary makes observations of the genuine computations during the federated learning; while in the active mode, the adversary can further actively modify its local updates with respect to the victim samples for the inference attack [29, 33, 46]. In both modes, the adversary has white-box access to the local model, following the existing privacy-preserving federated learning frameworks [1, 29, 33, 39]. The attack participants can collude with each other, but we assume that there are at most $m - 2$ out of m participants controlled by the attacker.

Threat model of fairness attacks. Regarding model fairness, we focus on network-level packet drop threats. They can arise from various methods, which have been studied in the context of network-level adversaries who carry out physical-layer attacks or transport-layer attacks [2, 18]. Adapted from Severi *et al.* [37], in the network-level packet drop threat, the attacker aims to decrease the accuracy of the FL model on a particular group of clients.

In this research, we particularly focus on the privacy and fairness attacks. The general poisoning attacks that aim to corrupt the overall model prediction are beyond the scope of this study.

3.4 CGD Algorithm

Algorithm 1 outlines the optimization process.


Global Model at Central Server
Model Layer
Gradient Aggregation
Global Model
Aggregated Gradients
Network Layer
Local Devices
DP
Local Gradients
Local Model
Local Dataset
Client 1
Client 2
Client l
Server-less Communication via MPC
Secure Addition
Σg^l
$\widetilde{g^l}$
g^1
g^2
g^l
Confined Model
Security & Privacy Protected
Server-less
Additive noise
Local loss bound check
Privacy leakage at server
Privacy leakage during communication
Malicious clients
Fairness Guaranteed
Asynchronous update
Heterogeneity-adaptive local confined model
Network dropping attack
Network/device delay
In CGD:
Share gradients with additive noise.
Aggregate parameters via secure addition and check the loss bound locally.
In Traditional FL:
Push gradients via privacy preserving techniques.
Pull parameters from server, replacing the local parameters

(a) Traditional Federated Learning (b) Confined Gradient Descent (CGD)

Figure 3: Architectural comparison of CGD and the traditional federated learning.

- **Initialization** (Line 3). Each participant l randomizes its own w_1^l with an initialization function $init()$ which takes three arguments: w_{anchor} denotes an anchor matrix that is shared among participants (*e.g.,* a zero matrix); $Rand^l$ denotes the randomization distribution (a typical one is a Gaussian distribution); and λ^l is to control the distance of the initial points to the anchor, and CGD allows each participant to choose their own λ^l. On the other hand, $init()$ also represents the tradeoff of privacy and accuracy. Intuitively, higher degree of randomness leads to differing confined models, but may undermine the accuracy performance of the confined models. In the following, we define a procedure called *balanced initialization* which balance out the randomness.

  DEFINITION 3. *(Balanced initialization). Given an initialization scheme Rand, each CGD participant uses an initialization parameter λ^l to initialize its confined model w^l in the way of*

  $$w_1^l = w_{anchor} + \lambda^l \cdot Rand^l \quad \text{for all } l \in (1, m). \tag{5}$$

  The key idea of the balanced initialization is to let each participant determines its own $Rand^l$ and λ^l independently to avoid leaking the average distance among the confined models.
- **Step 1: Compute local gradients** (Line 6). Every participant computes the local gradient g_k^l with respect to its current confined model w_k^l and local dataset ξ_l using Equation 4. We highlight that in traditional FL, each participant's local gradient is computed based on the same global model at the current iteration. In contrast, the local gradients in CGD are computed from private confined models which are different among participants.
- **Step 2: Add noise** (Line 7). In this step, CGD allows each participant to add noise to its local gradients with an $addNoise(g_k^l, \lambda^l)$ function. The perturbed local gradient is denoted as $\widetilde{g_k^l}$. This step aims to facilitate the comparison of privacy preservation between CGD and the traditional DP mechanism. We show that less noise is required for the same level of privacy guarantee, resulting in better privacy-accuracy tradeoff (detailed in Section 4).

Algorithm 1 Confined Gradient Descent Optimization

Input: A set of participants $l \in L$, the local training data ξ_l, fraction of selected participants C, the number of training iterations T, and the loss threshold LRR.
Output: Confined model for each participant w_*^l.

```
1:  $k \leftarrow 1$
2:  for each $l$ do
3:      $w_1^l \leftarrow init(w_{anchor}, Rand^l, \lambda^l)$
4:  while $k \leq T$ do
5:      for participants $l \in L$ do
6:          $g_k^l \leftarrow F(w_k^l, \xi_l)$
7:          $\widetilde{g_k^l} \leftarrow addNoise(g_k^l, \lambda^l)$
8:          $S \leftarrow randomSelect(C, L)$
9:          $\widetilde{g_k} \leftarrow \sum_{s \in S} \widetilde{g_k^s}$
10:         $\alpha_k = chooseStepSize()$
11:         $w \leftarrow w_k^l - \alpha_k \widetilde{g_k}$
12:         $LRR_k \leftarrow F(w, \xi_l) - g_k^l$
13:         if $LRR_k \leq LRR$ then
14:             $w_{k+1}^l \leftarrow w$
15:         else
16:             $w_{k+1}^l \leftarrow w_k^l$
17:     $k \leftarrow k + 1$
```

- **Step 3: Secure addition of gradients** (Line 9). A subset of participants S (randomly selected with the fraction parameter C) jointly compute the aggregated gradient $\widetilde{g_k}$, which is later used for loss bound checking. In order to preserve the secrecy of the confined models, we employ the additive secret sharing [4] for securely evaluating the sum of a set of secret values to avoid releasing the addends in plain text. In Appendix F, we prove that the secrecy of the confined models is guaranteed.
- **Step 4: Confined model pre-step** (Lines 10 to 11). Every participant in the current iteration computes its own w with learning rate α_k as a pre-step, which will be applied to the update of the confined model (or dropped) after the local loss bound check.
- **Step 5: Local loss bound checking** (Lines 12 to 16). Every participant compute loss LRR_k in this round if w is applied, *i.e.,* the change of model's prediction performance in terms of the loss

function value, and takes a descent step on w if the performance degradation is less than the local loss threshold LRR.

3.5 Implementation and Benchmarks

Datasets. We implement and evaluate CGD's performance on four benchmark datasets, *i.e.*,, MNIST [22], CIFAR-10 [21], Location30 [44] and Hourly Energy Consumption (HEC) [32]. We give the details of the datasets and models in Appendix A.

Benchmarks. We compare CGD's performance in terms of privacy and fairness with the benchmarks in the following.

i) Differentially private benchmarks. We compare CGD with FL with DP mechanisms. Specifically, we take the following two state-of-the-art differentially private FL schemes in our experiments.

- **SDP** [1]. The standard Gaussian mechanism is based on the moment accountant proposed by Abadi *et al.* [1].
- **BDP** [42]. It takes into account that training samples in ML are typically drawn from the same distribution, and calibrates noise to the distribution and thus provides tight privacy guarantees.

We further compare the performance of CGD with **Centralized training** (*i.e.,* training on aggregated data), **Local training** (*i.e.,* training on each participant's local data only), and **FedAvg** [20].

ii) Preserving privacy against poisoning MIA in the active mode. We implement the state-of-the-art poisoning membership inference attack against FL [26, 29, 33, 46]. In such attack, the adversary manipulates updates to incline towards ascending the gradients of the global model.

iii) Fairness benchmark. We compare with Ditto [24], which is the state-of-the-art towards fairness in FL. It is a personalization add-on that encourages the personalized models to be close to the optimal global model through local training with regularization.

Default FL setting. CGD is trained using Algorithm 1 on the dataset which is partitioned and distributed to m participants, where $m = (m^h \times m^v)$. By default, the initialization parameter λ is set to 0.1, and m is set to (4×4).

4 PRIVACY ANALYSIS

In this section, we first analyze CGD's privacy preservation within the framework of DP in Sections 4.1. We then conduct experimental evaluation to demonstrate privacy-utility performance of CGD as well as its capacity in defending MIA (Section 4.2).

4.1 Privacy Bound in CGD

In our analysis, we let the privacy cost be: $\mathcal{P} = \log\left(\frac{Pr(\mathcal{M}(d)=S)}{Pr(\mathcal{M}(d')=S)}\right)$. We use Chernoff bound, which is widely used in various privacy accounting techniques [1, 30, 42], to convert Inequality 2 as

$$Pr(\mathcal{P} \geq \epsilon) \leq \frac{\mathbb{E}[e^{c\mathcal{P}}]}{e^{c\epsilon}}. \tag{6}$$

The inequality holds for any c, and it determines the tightness of the bound (*i.e.,* minimizes δ and/or ϵ). We then define

$$\mathcal{L}(\mathcal{M})_c = \mathbb{E}\left[e^{c\mathcal{P}}\right] = \mathbb{E}\left[\left(\frac{Pr(\mathcal{M}(d) = S)}{Pr(\mathcal{M}(d') = S)}\right)^c\right]. \tag{7}$$

Inspired by the moments accountant [1], we aim to bound $\mathcal{L}(\mathcal{M})_c$, and we can then use the Chernoff inequality to transform the bound to (ϵ, δ)-differential privacy. To bound $\mathcal{L}(\mathcal{M})_c$, we first analyze the relationship of the randomness in CGD and the traditional noise in DP framework based on Gaussian mechanism [11].

Table 2: Comparison of privacy bounds between CGD and DP-FL.

	MNIST		CIFAR-10		Location30		Energy HEC	
	Test Acc	(ϵ, δ)-DP	Test Acc	(ϵ, δ)-DP	Test Acc	(ϵ, δ)-DP	MSE	(ϵ, δ)-DP
CGD	95.00%	(**0.41**, 10^{-5})	73.00%	(**0.35**, 10^{-5})	75.80%	(**0.75**, 10^{-5})	0.0062	(**0.80**, 10^{-5}) [5]
BDP [42]	95.00%	(0.95, 10^{-5})	73.00%	(0.76, 10^{-5})	56.04%	(329.12, 10^{-5})	0.0082	(1.46, 10^{-5})
SDP [1]	95.00%	(2.00, 10^{-5})	70.00%	(4.00, 10^{-5})	56.04%	(329.29, 10^{-5})	0.0082	(1.46, 10^{-5})

Existing DP framework adds noise to the local gradients which are computed from the same global model. The noise, which we refer to as the *standard noise*, denoted as $\sigma_{standard}$ is required to be added in each single training iteration [1, 11]. In contrast, the randomness in the confined models, which is protected by additive secret sharing scheme, is incorporated to the local gradient's calculation[3]. This fact can be utilized to save the amount of additive noise to reach the same ϵ. Intuitively, such randomness can be regarded as inherent noise which allows the standard additive noise to be reduced.

In CGD, each participant conducts computation based on different models with the randomness of $\mathcal{N}(0, \lambda^l)$. Thus, each w^l satisfies (ϵ', δ)-differential privacy, where $\epsilon' \geq \sqrt{2 \ln \frac{1.25}{\delta\lambda^l}}$ by standard arguments (Theorem 3.22 in [11]). We then have g^l satisfying (ϵ', δ)-differential privacy based on the post-processing property of DP. We thus can construct g^l (clipped) as the sum of a g^l_{anchor} and $\mathcal{N}(0, \lambda^l)$, and rewrite Line 7 in Algorithm 1 as

$$\begin{aligned} \widetilde{g^l} &= g^l/\max(1, \|g^l_1\|_2) + \mathcal{N}(0, \sigma^2) \\ &= g^l_{anchor} + \mathcal{N}(0, \lambda^l) + \mathcal{N}(0, \sigma^2) = g^l_{anchor} + \mathcal{N}(0, \lambda^l + \sigma^2) \end{aligned} \tag{8}$$

Therefore, we have

$$\widetilde{g} = \sum_{l=1}^{m} \widetilde{g^l} = \sum_{l=1}^{m} g^l_{anchor} + \sum_{l=1}^{m} \mathcal{N}(0, \lambda^l + \sigma^2) = \sum_{l=1}^{m} g^l_{anchor} + \mathcal{N}(0, \sum_{l=1}^{m} \lambda^l + m\sigma^2) \tag{9}$$

Lemma 1. *For any positive integer $c \geq 1$, CGD with an addNoise function defined as $\widetilde{g^l_k} = g^l_k/\max(1, \|g^l_k\|_2) + \mathcal{N}(0, \sigma^2)$ satisfies*

$$\log \mathcal{L}(\mathcal{M})_c \leq \frac{c(c+1)}{\sum_{l=1}^{m} \lambda^l + m\sigma^2} + O\left(\frac{c^3}{(\sum_{l=1}^{m} \lambda^l + m\sigma^2)^{3/2}}\right) \tag{10}$$

We give the proof in Appendix B.

Theorem 1. *Given the number of training iteration T, there exists a constant a such that CGD is (ϵ, δ)-differentially private for any $\delta > 0$ if we choose $\sigma \geq 2\sqrt{\frac{Ta\log\frac{1}{\delta}}{m\epsilon^2} - \mathbb{E}[\lambda^l]}$.*

We give the proof in Appendix C.

We remark that the result of Theorem 1 does not consider the privacy amplification via sub-sampling because the selection of the training samples at each iteration can be non-IID in the federated learning. As such, the result is the worst case in which the batch gradient descent is conducted. In Appendix D, we also provide a tighter bound of CGD's privacy preservation with IID data.

4.2 Empirical Results on Privacy

Privacy Utility *vs.* Accuracy Performance. Table 2 summarizes the privacy bounds in CGD, BDP [42] and SDP [1] respectively,

[3] In Appendix F, we provide a security analysis to demonstrate that the confidentiality of the confined models is guaranteed against an honest-but-curious white-box adversary who may control t out of m participants, where $t \leq m - 2$

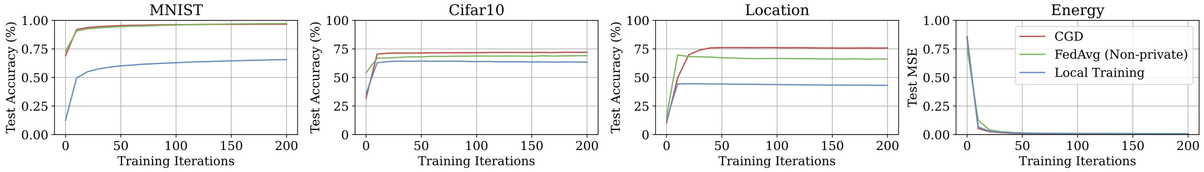


Figure 4: CGD's test accuracy compared to FedAvg and local training.

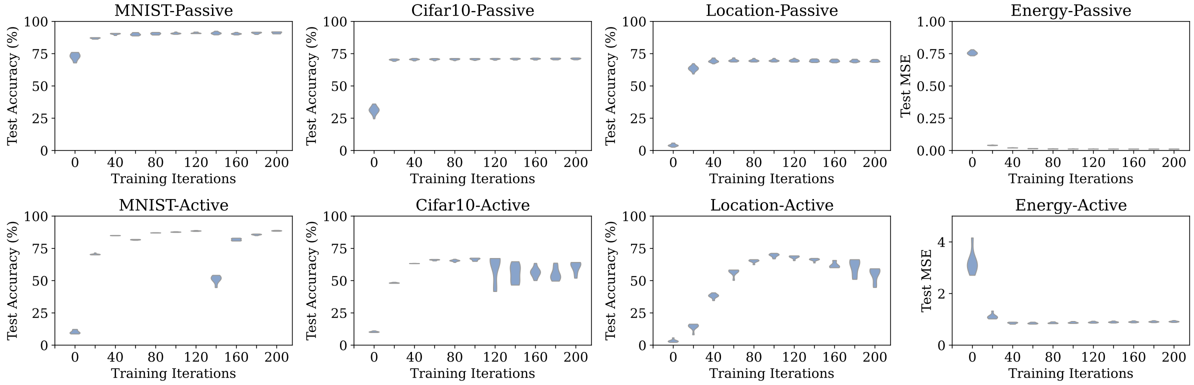


Figure 5: The variation of test accuracy among participants in CGD.

Table 3: CGD's performance against active MIA.

		MNIST	CIFAR	Location30	Energy HEC
Active MIA Success Rate	CGD	**51.30%**	**50.00%**	**55.10%**	**50.10%**
	FedAvg	54.30%	60.30%	88.30%	72.87%
Model Performance	CGD	**92.00%**	**66.90%**	**68.82%**	**0.94**
	FedAvg	86.80%	45.60%	33.33%	1.14

when they reach the same accuracy level. CGD retains a lower ϵ when reaching the same accuracy level as others. It achieves $(0.41, 10^{-5})$-DP on MNIST, $(0.35, 10^{-5})$ on CIFAR-10, $(0.75, 10^{-5})$ on Location30, and $(0.80, 10^{-5})$ on Energy HEC, all stronger than BDP and SDP. Figure 4 illustrates the performance of CGD on the test accuracy compared to the FedAvg and local training. For MNIST, with the privacy bounds at $(0.54, 10^{-5})$, CGD achieves 96.9% in test accuracy, which is close to 97.54% of FedAvg and outperforms of 71.77% of the local training. For CIFAR-10, FedAvg reaches the test accuracy of 75.75%, which is taken as our baseline.[4] In line with the performance on MNIST, with the privacy bounds at $(0.36, 10^{-5})$, CGD achieves 74.30% in test accuracy, while the local training only reaches 53.67%. For Location30 and Energy HEC, CGD achieves 75.8% of test accuracy with $(0.75, 10^{-5})$-DP and 0.0062% of MSE with $(0.80, 10^{-5})$, respectively, outperforming the test accuracy of 73.27%, 0.0072 in FedAvg, and 42.80%, 0.007 in the local training.

Robustness against Active MIA. We set $m = (8 \times 4)$, which corresponds to 32 participants in the experiments. Specifically, 20% of these participants are manipulated by the poisoning adversary, aligning with the typical configuration in the literature [9, 15]. Table 3 summarizes CGD's performance against the active membership inference attack. CGD significantly mitigates the attack, reducing the attack success rate to around 50% (*i.e.,* random guess of member and non-member) across all datasets. It also retains the prediction performance: 92.00%, 66.90%, 68.82% of test accuracy, and 0.94 of MSE on MNIST, CIFAR-10, Location30, and Energy HEC, respectively, whereas the performance in FedAvg (without the protection of CGD) reduces significantly due to the poisoning.

Table 4: CGD's performance of fairness (Definition 2): Standard deviation of test accuracy on MNIST, CIFAR, Location30, and standard deviation of MSE on Energy HEC among participants.

		MNIST	CIFAR	Location30	Energy HEC
Clean (honest-but-curious)	CGD	**0.0013**	**0.0043**	**0.0023**	**0.0001**
	Ditto	0.0082	0.1223	0.0698	0.0051
Active MIA	CGD	**0.0043**	**0.1370**	**0.0523**	**0.0005**
	Ditto	0.1974	0.1720	0.1657	0.0103

5 FAIRNESS ANALYSIS

In this section, we first conduct a formal analysis on the bound of the distance between an arbitrary confined model w^l_* learned by CGD and the centralized model w_*, demonstrating the fairness-convergence guarantee (Sections 5.1- 5.2). We then conduct experiments to demonstrate CGD's fairness on real-world datasets, comparing with the state-of-the-art fair FL method [24] (Section 5.3).

5.1 Bounded Asynchrony in CGD

In CGD, a participant does not take the descent step when detecting any suspicious updates against it. This results in the asynchronous optimization, indicating that any two participants may be at different iteration rounds. For any participant l at iteration k, the confined model w^l_k is updated as

$$w^l_{k+1} = w^l_k - \alpha_k \sum_{j=1}^{S_t} \widetilde{g^j}(w^j_{k-\tau^j_k}, \xi_j), \tag{11}$$

[4] We note that by making the network deeper or using other advanced techniques, better accuracy can be obtained, given the state-of-the-art being 99.37% [19].

where τ_k^j denotes the *staleness*, *i.e.,* how many iterations of a delay between the participants j and l at k^{th} iteration. We show that, when $C \leq 1 - P_{\mathcal{A}}$ (recall that C is a fraction parameter of m participants, and $P_{\mathcal{A}}$ denotes the malicious proportion), the staleness in CGD is bounded (*i.e.,* $\tau_k^j \leq \tau$), with a preferably subpolynomially small probability being broken (see Appendix G).

Thus, the bounded staleness of τ can also be utilized to make CGD resistant to network-level attacks against FL models, such as a packet drop attack, *i.e.,* CGD is tolerant to the drop up to τ.

5.2 Fairness-Convergence Theorem

We now give an upper bound of the distance between any participant's w_*^l and w_*, demonstrating the convergence rate and the fairness guarantee in CGD. Our analysis investigates a *regret function* $\mathcal{R}$, which represents the difference between the CGD's loss and the loss of the centralized model. $\mathcal{R}$ is defined as

$$\mathcal{R} = \frac{1}{T}\sum_{k=1}^{T}(F(w_k^l) - F(w_*)). \tag{12}$$

We first present the assumption on a bounded solution space which is widely used in related studies [8, 17].

Assumption 1. *The set of* $\{w_k^l\}|_{k\in[1,T]}$ *is contained in an open set over which* F *is bounded below a scalar* F_{inf}, *such that* (i) *there exists a* $D > 0$, *s.t.,* $\|w_k^l - w_*\|_2^2 \leq D^2$ *for all* k, *and* (ii) *there exists a* $G > 0$, *s.t.,* $\|g^l(w_k^l, \xi_l)\|_2^2 \leq G^2$ *for all* $w_k^l \subset \mathbb{R}^d$.

We also follow prior works [17, 41] in which Lipschitz continuity and convexity are applied.

Theorem 2. *For any cost function that is L-lipschitz and satisfies convexity and Assumption 1, assuming that the learning rate meets* $\alpha_k = \frac{\alpha}{(k+\mu T)^2}$ $(0 < \mu < 1)$, *and a bounded staleness of* τ, *the CGD optimization gives the regret* $\mathcal{R}$

$$\mathcal{R} = O(\gamma + \frac{1}{\mu T + 1} + \frac{\ln|\mu T + 1|}{T} + \frac{1}{T(1 - \tau + T)}), \tag{13}$$

where $\gamma = m\| \mathop{\mathbb{E}}_{j\in m}[w_1^l - w_1^j]\|$.

Below we discuss the implications of the theorem, and leave its proof to Appendix E.

Implication #1: Convergence rate. $\frac{1}{\mu T+1}$, $\frac{\ln|\mu T+1|}{T}$ and $\frac{\ln|\mu T+1|}{T} + \frac{1}{T(1-\tau+T)}$ approach 0 as T increases, implying that CGD will converge toward the optimum. The convergence rate is affected by the staleness bound τ, *i.e.,* a larger τ will lead to a slower rate. It can also be adjusted by the parameter μ, named *diminishment factor*.

Implication #2: Bounded optimality gap. The gap between the confined models and the centralized model is bounded by γ, implying the fairness guarantee. The gap is affected by the variance of w_1^j. We show that, CGD keeps robust (in terms of test accuracy) even when the participants' λ^ls differ by two orders of magnitude.

5.3 Empirical Results on Fairness

Figure 5 shows the fairness performance of CGD during the training iterations. When CGD converges, the standard deviation (SD) of test accuracy across participants achieves 0.0013, 0.0043, 0.0023, and 0.0001 on MNIST, CIFAR-10, Location30, and Energy HEC in the honest-but-curious scenario, and 0.0043, 0.1370, 0.0523, and 0.0005 on these datasets against active membership inference.

In Table 4, we compare the performance of CGD with Ditto [24]. In the passive scenario (which is in line with the "clean data" setting in Ditto), CGD reduces the standard deviation of test accuracy across participants by 0.0069, 0.1180, 0.0675, and 0.0050 (MSE) on MNIST, CIFAR-10, Location30, and Energy HEC, respectively. In the active scenario (which is in line with the "data poisoning attack" setting in Ditto), CGD reduces the standard deviation of test accuracy by 0.1931, 0.0350, 0.1134, and 0.0098 on these datasets.

6 ABLATION STUDIES OF PERFORMANCE

Scalability. In this section, we investigate how the number of participants affects CGD's privacy and accuracy performance. CGD's performance remains relatively stable as the participant number grows, even with a large number of participants. With 112,000 participants (*i.e.,* $m = (1000 \times 112)$) for MNIST, CGD still achieves 95.78% in test accuracy, close to 97.54% of the centralized baseline. With 32,000 participants (*i.e.,* $m = (1000 \times 32)$) for CIFAR, CGD achieves 67.67% in test accuracy, which is also close to the centralized baseline (75.72%). We give more experimental results in Appendix H.1.

Effect of the Initialization. We conduct experiments to investigate the effect of initialization parameter λ. We let each participant randomly select its own λ based on the uniform distribution within the range from 0.001 to 0.1. The performance of CGD is still comparable with the centralized mode. For example, with $m = (1000\times112)$ participants on MNIST and $m = (1000\times32)$ on CIFAR, CGD achieves 97.07% and 71.01% in the test accuracy respectively. This suggests that CGD keeps robust when the λs of its participants differ by two orders of magnitude. This provides sufficient randomness in the balanced initialization strategy. More results of the investigate on the parameter λ is given in Appendix H.2.

7 CONCLUSION

We have presented CGD, a novel optimization algorithm for learning confined models to enhance privacy and fairness for federated learning. It achieves tighter differential privacy bounds than the state-of-the-art DP-FL frameworks. CGD is also effective in defending against poisoning MIA in the active mode, reducing the attack accuracy to 50%. We also formally prove the convergence-fairness of CGD optimization. Since we focus on the privacy and fairness of malicious clients or local models, security attacks such as intentionally injecting poisoned data and uploading backdoored local models on CGD provide avenues for our future research. We hope that CGD developed in this paper could be steadily leaving its infancy for building protection and resilience for critical infrastructure.

ACKNOWLEDGMENTS

This work is supported in part by Australian Research Council (ARC) DP240103068 and DP230101753, the UQ Cyber Strategic Research Funding, and the UQ-Data61 Cooperative Research Program funding. Minhui Xue and Surya Nepal are supported by CSIRO – National Science Foundation (US) AI Research Collaboration Program.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 308–318.

[2] Y Ozan Basciftci, Fangzhou Chen, Joshua Weston, Ron Burton, and C Emre Koksal. 2015. How vulnerable is vehicular communication to physical layer jamming attacks?. In *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*. 1–5.

[3] Alberto Bietti, Chen-Yu Wei, Miroslav Dudik, John Langford, and Steven Wu. 2022. Personalization improves privacy-accuracy tradeoffs in federated learning. In *International Conference on Machine Learning*. 1945–1962.

[4] Dan Bogdanov, Sven Laur, and Jan Willemson. 2008. Sharemind: A framework for fast privacy-preserving computations. In *European Symposium on Research in Computer Security*. 192–206.

[5] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečný, Stefano Mazzocchi, Brendan McMahan, et al. 2019. Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems* 1 (2019), 374–388.

[6] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1175–1191.

[7] Léon Bottou. 2012. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*. Springer, 421–436.

[8] Léon Bottou, Frank E Curtis, and Jorge Nocedal. 2018. Optimization methods for large-scale machine learning. *Siam Review* 60, 2 (2018), 223–311.

[9] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. 2021. FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping. In *the Network and Distributed System Security Symposium (NDSS)*.

[10] Xu Cheng, Fan Shi, Yongping Liu, Xiufeng Liu, Lizhen Huang, et al. 2022. Wind turbine blade icing detection: A federated learning approach. *Energy* 254, PC (2022).

[11] Cynthia Dwork and Aaron Roth. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[12] Mohammad Navid Fekri, Katarina Grolinger, and Syed Mir. 2022. Distributed load forecasting using smart meter data: Federated learning with Recurrent Neural Networks. *International Journal of Electrical Power & Energy Systems* 137 (2022), 107669.

[13] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 249–256.

[14] Oded Goldreich, Silvio Micali, and Avi Wigderson. 2019. How to play any mental game, or a completeness theorem for protocols with honest majority. In *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*. 307–328.

[15] Zirui Gong, Liyue Shen, Yanjun Zhang, Leo Yu Zhang, Jingwei Wang, Guangdong Bai, and Yong Xiang. 2023. AGRAMPLIFIER: Defending Federated Learning Against Poisoning Attacks Through Local Update Amplification. *IEEE Transactions on Information Forensics and Security* 19 (2023), 1241–1250.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[17] Yaochen Hu, Di Niu, Jianming Yang, and Shengping Zhou. 2019. FDML: A collaborative machine learning framework for distributed features. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2232–2240.

[18] Samuel Jero, Md Endadul Hoque, David R Choffnes, Alan Mislove, and Cristina Nita-Rotaru. 2018. Automated Attack Discovery in TCP Congestion Control Using a Model-guided Approach. In *the Network and Distributed System Security Symposium (NDSS)*.

[19] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. 2020. Big transfer (bit): General visual representation learning. In *16th European Computer Vision Conference (ECCV)*. 491–507.

[20] Jakub Konečný, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. 2016. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492* (2016).

[21] Alex Krizhevsky and Geoffrey Hinton. 2009. Learning multiple layers of features from tiny images. (2009).

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* (1998), 2278–2324.

[23] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. 2018. Visualizing the loss landscape of neural nets. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. 6391–6401.

[24] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. 2021. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*. 6357–6368.

[25] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B Allen, Randy P Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2020. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523* (2020).

[26] Mengyao Ma, Yanjun Zhang, Pathum Chamikara Mahawaga Arachchige, Leo Yu Zhang, Mohan Baruwal Chhetri, and Guangdong Bai. 2023. Loden: Making every client in federated learning a defender against the poisoning membership inference attacks. In *Proceedings of the 2023 ACM Asia Conference on Computer and Communications Security*. 122–135.

[27] Tilen Marc, Miha Stopar, Jan Hartman, Manca Bizjak, and Jolanda Modic. 2019. Privacy-Enhanced Machine Learning with Functional Encryption. In *European Symposium on Research in Computer Security*. 3–21.

[28] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning differentially private recurrent language models. *The International Conference on Learning Representations* (2018).

[29] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. 691–706.

[30] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. 263–275.

[31] Payman Mohassel and Yupeng Zhang. 2017. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy (SP)*. 19–38.

[32] Rob Mulla. 2019. *Hourly Energy Consumption*. https://www.kaggle.com/datasets/robikscube/hourly-energy-consumption

[33] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. 739–753.

[34] PHOENIX Project. 2020. Federated Learning for enhancing the security of Critical Energy Infrastructures. .

[35] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. 2022. Federated learning with partial model personalization. In *International Conference on Machine Learning*. 17716–17758.

[36] Alexis R Santos-Lozada, Jeffrey T Howard, and Ashton M Verdery. 2020. How differential privacy will affect our understanding of health disparities in the United States. *Proceedings of the National Academy of Sciences* 117, 24 (2020), 13405–13412.

[37] Giorgio Severi, Matthew Jagielski, Gökberk Yar, Yuxuan Wang, Alina Oprea, and Cristina Nita-Rotaru. 2022. Network-Level Adversaries in Federated Learning. In *2022 IEEE Conference on Communications and Network Security (CNS)*. 19–27.

[38] Sagar Sharma and Keke Chen. 2019. Confidential boosting with random linear classifiers for outsourced user-generated data. In *European Symposium on Research in Computer Security*. 41–65.

[39] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. 1310–1321.

[40] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*. 3–18.

[41] Anvith Thudi, Hengrui Jia, Ilia Shumailov, and Nicolas Papernot. 2022. On the necessity of auditable algorithmic definitions for machine unlearning. In *31st USENIX Security Symposium (USENIX Security 22)*. 4007–4022.

[42] Aleksei Triastcyn and Boi Faltings. 2020. Bayesian differential privacy for machine learning. In *International Conference on Machine Learning*. 9583–9592.

[43] Minhui Xue, Surya Nepal, Ling Liu, Subbu Sethuvenkatraman, Xingliang Yuan, Carsten Rudolph, Ruoxi Sun, and Greg Eisenhauer. 2024. RAI4IoE: Responsible AI for Enabling the Internet of Energy. In *The Fifth IEEE International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (IEEE TPS-ISA)*.

[44] Dingqi Yang, Daqing Zhang, and Bingqing Qu. 2016. Participatory cultural mapping based on collective behavior data in location-based social networks. *ACM Transactions on Intelligent Systems and Technology (TIST)* 7, 3 (2016), 1–23.

[45] Kai Zhang, Yanjun Zhang, Ruoxi Sun, Pei-Wei Tsai, Muneeb Ul Hassan, Xin Yuan, Minhui Xue, and Jinjun Chen. 2024. Bounded and Unbiased Composite Differential Privacy. In *45th IEEE Symposium on Security and Privacy (IEEE S&P)*.

[46] Yanjun Zhang, Guangdong Bai, Mahawaga Arachchige Pathum Chamikara, Mengyao Ma, Liyue Shen, Jingwei Wang, Surya Nepal, Minhui Xue, Long Wang, and Joseph Liu. 2023. AgrEvader: Poisoning Membership Inference against Byzantine-robust Federated Learning. In *Proceedings of the ACM Web Conference 2023*. 2371–2382.

[47] Yanjun Zhang, Guangdong Bai, Xue Li, Caitlin Curtis, Chen Chen, and Ryan KL Ko. 2020. PrivColl: Practical Privacy-Preserving Collaborative Machine Learning. In *European Symposium on Research in Computer Security*. 399–418.

[48] Yanjun Zhang, Guangdong Bai, Mingyang Zhong, Xue Li, and Ryan Ko. 2020. Differentially private collaborative coupling learning for recommender systems. *IEEE Intelligent Systems* (2020).

APPENDIX

A DATASETS AND MODELS

- **MNIST** [22]. It has 60, 000 training samples and 10, 000 test samples, each with 784 features representing 28 × 28 pixels in the image. We use a CNN with ReLU of 256 units.
- **CIFAR-10** [21]. It consists of 60,000 32 × 32 colour images in 10 classes (*e.g.,* airplane, bird, and cat). We follow the settings of Abadi *et al.* [1]. We pretrain CIFAR-100 with the ResNet-56 [16] and retrain only the last FC layer on CIFAR-10.
- **Location30** [44]. It contains mobile users' "check-in" data in the Foursquare social network. We use the processed dataset [40] which contains 30 classes and 5,010 user profiles, each with 446 binary features. The model is a FCN with layer sizes {446, 512, 128, 30}.
- **Hourly Energy Consumption (HEC)** [32]. It contains 121,275 historical hourly energy consumption records (in MegaWatts) in the United States over 10 years. It is used for time-series forecasting tasks. The model architecture is LSTM, where we use MSE to measure the accuracy of regression prediction.

B PROOF OF LEMMA 1

Proof. Let v_0 denote the pdf of $\mathcal{N}(0,\sigma'^2)$ and let v_1 denote the pdf of $\mathcal{N}(1,\sigma'^2)$, where $\sigma'^2 = \sum_{l=1}^{m} \lambda^l + m\sigma^2$. Without loss of generality, we have: $\mathcal{M}(d) \sim v_0, \mathcal{M}(d') \sim v_1$.

We aim to demonstrate the bound of

$$\max\left(\mathbb{E}_{x\sim v_1}\left[(v_1(x)/v_0(x))^c\right], \mathbb{E}_{x\sim v_0}\left[(v_0(x)/v_1(x))^c\right]\right).$$

We will prove one case and the method of proving the other case is similar.

$$\begin{aligned}
&\mathbb{E}_{x\sim v_1}\left[(v_1(x)/v_0(x))^c\right] = \mathbb{E}_{x\sim v_0}\left[(v_1(x)/v_0(x))^{c+1}\right] \\
&= \mathbb{E}_{x\sim v_0}\left[\left(1+\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{c+1}\right] = \sum_{i=0}^{c+1}\binom{c+1}{i}\mathbb{E}_{x\sim v_0}\left[\left(\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{i}\right] \\
&= \binom{c+1}{0}\mathbb{E}_{x\sim v_0}\left[\left(\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{0}\right] + \binom{c+1}{1}\mathbb{E}_{x\sim v_0}\left[\left(\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{1}\right] \\
&\quad + \binom{c+1}{2}\mathbb{E}_{x\sim v_0}\left[\left(\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{2}\right] + \ldots
\end{aligned} \tag{14}$$

The first term of the binomial expansion is 1, and the second term is

$$\begin{aligned}
\mathbb{E}_{x\sim v_1}\left[\left(\frac{v_0(x)-v_1(x)}{v_1(x)}\right)^{1}\right] &= \int_{-\infty}^{\infty} v_1(x)\frac{v_0(x)-v_1(x)}{v_1(x)}\,\mathrm{d}x \\
&= \int_{-\infty}^{\infty} v_0(x)-v_1(x)\,\mathrm{d}x = 1-1=0
\end{aligned}$$

The third term is

$$\begin{aligned}
&\mathbb{E}_{x\sim v_0}\left[\left(\frac{v_1(x)-v_0(x)}{v_0(x)}\right)^{2}\right] = \mathbb{E}_{x\sim v_0}\left[\left(\frac{\frac{1}{\sigma'\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-1}{\sigma'}\right)^2}}{\frac{1}{\sigma'\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x}{\sigma'}\right)^2}}-1\right)^{2}\right] \\
&= \mathbb{E}_{x\sim v_0}\left[\exp\left(\frac{4x-2}{2\sigma'^2}\right)\right] - 2\mathbb{E}_{x\sim v_0}\left[\exp\left(\frac{2x-1}{2\sigma'^2}\right)\right] + 1 \\
&= \exp\left(\frac{-2}{2\sigma'^2}\right)\int_{-\infty}^{\infty}\frac{1}{\sigma'\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x}{\sigma'}\right)^2}e^{\left(\frac{4x}{2\sigma'^2}\right)}\,\mathrm{d}x \\
&\quad - 2\exp\left(\frac{-1}{2\sigma'^2}\right)\int_{-\infty}^{\infty}\frac{1}{\sigma'\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x}{\sigma'}\right)^2}e^{\left(\frac{2x}{2\sigma'^2}\right)}\,\mathrm{d}x + 1 = \exp\left(\frac{1}{\sigma'^2}\right) - 1
\end{aligned}$$

Therefore, the sum of the first three terms is

$$1+\binom{c+1}{2}\left(\exp\left(\frac{1}{\sigma'^2}\right)-1\right) < \exp\left(\frac{c(c+1)}{\sum_{l=1}^{m}\lambda^l+m\sigma^2}\right) \quad \textit{(with } c \geq 1\textit{)}$$

The remaining terms is dominated by the $i = 3$, which is $O(\frac{c^3}{(\sum_{l=1}^{m}\lambda^l+m\sigma^2)^{3/2}})$. We omit the derivation as it can be easily obtained by the standard calculus. □

C PROOF OF PRIVACY THEOREM (THEOREM 1)

Proof. The accumulated privacy cost with T iterations is

$$\log\mathcal{L}(\mathcal{M})_c^T = T\log\mathcal{L}(\mathcal{M})_c < T\frac{ac(c+1)}{\sum_{l=1}^{m}\lambda^l+m\sigma^2} \tag{15}$$

Equation 15 follows the Composability (*i.e.,* Theorem 2.1 in [1]).
To make CGD (ϵ,δ)-differentially private, according to Chernoff inequality (Theorem 2.2 in [1]), it suffices that

$$\begin{gathered}
T\frac{ac(c+1)}{\sum_{l=1}^{m}\lambda^l+m\sigma^2} \leq \frac{c\epsilon}{2}, \\
\exp\left(\frac{-c\epsilon}{2}\right) \leq \delta
\end{gathered}$$

Therefore,

$$\sum_{l=1}^{m}\lambda^l+m\sigma^2 \geq \frac{2Ta(c+1)}{\epsilon} \geq 4Ta\frac{\log\frac{1}{\delta}}{\epsilon^2} \tag{16}$$

Inequation 16 is from $c \geq 2\log\frac{1}{\delta}/\epsilon$. Therefore, we have

$$\sigma \geq 2\sqrt{\frac{Ta\log\frac{1}{\delta}}{m\epsilon^2}-\mathbb{E}[\lambda^l]} \tag{17}$$

□

D PRIVACY BOUND WITH IID DATA

In this section, we give CGD's privacy bound with IID data.

Corollary 1. *Given the number of training iteration T, the fraction of selected participants C there exists a constant a so that CGD is (ϵ,δ)-differentially private for any $\delta > 0$ if we choose*

$$\sigma \geq 2\sqrt{\frac{C^2Ta\log\frac{1}{\delta}}{m\epsilon^2}-\mathbb{E}[\lambda^l]} \tag{18}$$

We omit the proof as it can be easily obtained using similar methods in Lemma 1 and Theorem 1.

E PROOF OF FAIRNESS-CONVERGENCE THEOREM (THEOREM 2)

Our proof aims to identify an upper bound of $\mathcal{R}$. To this end, we consider the trend of the distance from w_k^l to w_*, which shrinks as k increases, if there exists a bound. Since w_k^l is updated using $\sum_{j=1}^{S_t}\widetilde{g^j}(w_{k-\tau_k^j}^j,\xi_j)$ (*i.e.,* the descent in CGD), and w_* is obtained by ∇F (*i.e.,* the descent in the centralized training), the trend of the distance therefore should be related to the deviation between these two. Exploring this leads to the following lemma which describes this relationship.

Lemma 2. *(Optimization trajectory). Let $S_{k+1} = \frac{1}{2}\|w_{k+1}^l - w_*\|_2^2$ and $S_k = \frac{1}{2}\|w_k^l - w_*\|_2^2$. Let $\nabla F(\cdot) = \frac{1}{n}\sum_{i=1}^{n}\nabla F(\cdot,\xi_i)$. We have*

$$\begin{aligned}
S_{k+1}-S_k &= \frac{1}{2}\|\alpha_k\sum_{j=1}^{S_t}\widetilde{g^j}(w_{k-\tau_k^j}^j,\xi_j)\|_2^2 \\
&\quad - \alpha_k\langle w_k^l - w_*, \nabla F(w_k^l)\rangle - \alpha_k\langle w_k^l - w_*, \sum_{j=1}^{S_t}\widetilde{g^j}(w_{k-\tau_k^j}^j,\xi_j) - \nabla F(w_k^l)\rangle.
\end{aligned}$$

We omit the proof of the Lemma as it can be easily obtained by substituting w_{k+1}^l according to Equation 11.
Proof of Theorem 2.

Proof. By the definition of the regret function (Equation 12) and convexity, we have

$$\mathcal{R} = \frac{1}{T}\sum_{k=1}^{T}(F(w_k^l) - F(w_*)) \leq \frac{1}{T}\sum_{k=1}^{T}\langle w_k^l - w_*, \nabla F(w_k^l)\rangle \tag{19}$$

Applying Lemma 2 to Inequation 19 and multiplying it by T, we have

$$\begin{aligned} T \cdot R \leq & \sum_{k=1}^{T}\frac{1}{2}\alpha_k \| \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j)\|_2^2 - \sum_{k=1}^{T}\frac{1}{\alpha_k}(S_{k+1} - S_k) \\ & - \sum_{k=1}^{T}\langle w_k^l - w_*, \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j) - \nabla F(w_k^l)\rangle \end{aligned} \tag{20}$$

Inequation 20 can be decomposed into three terms, *i.e.*, $\sum_{k=1}^T \frac{1}{2}\alpha_k \| \sum_{j=1}^{S_t} \widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j)\|_2^2$ and $-\sum_{k=1}^T \frac{1}{\alpha_k}(S_{k+1} - S_k)$ are *overall updates*, and $-\sum_{k=1}^T\langle w_k^l - w_*, \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j) - \nabla F(w_k^l)\rangle$ is *gradients margin*. Next, we explore the boundedness of each term. For the *overall updates*, we have

$$\sum_{k=1}^{T}\frac{1}{2}\alpha_k \| \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j)\|_2^2 \leq \sum_{k=1}^{T}\frac{1}{2}\frac{\alpha}{(k + \mu T)^2}C^2 m^2 G^2 \tag{21}$$

$$= \frac{\alpha C^2 m^2 G^2}{2}\sum_{k=1}^{T}\frac{1}{(k+\mu T)^2} < \alpha C^2 m^2 G^2 \ . \tag{22}$$

Inequation 21 is based on Assumption 11, and Inequation 22 is based on the solution to the Basel problem that $\sum_{x=1}^{\infty}\frac{1}{x^2} < 2$.

$$-\sum_{k=1}^{T}\frac{1}{\alpha_k}(S_{k+1} - S_k) = \sum_{k=1}^{T}\frac{1}{\alpha_k}\left(\frac{1}{2}\| w_k^l - w_*\|_2^2 - \frac{1}{2}\| w_{k+1}^l - w_*\|_2^2\right) \tag{23}$$

$$\leq \sum_{k=1}^{T}\frac{1}{2\alpha_k}\|(w_k^l - w_*) - (w_{k+1}^l - w_*)\|_2^2 \tag{24}$$

$$= \sum_{k=1}^{T}\frac{1}{2}\alpha_k \| \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j)\|_2^2 < \alpha C^2 m^2 G^2 \tag{25}$$

Inequation 24 follows reverse triangle inequality, and Inequation 25 reuses the result of the first term (Equation 22).

The third term gradients margin:

$$-\sum_{k=1}^{T}\langle w_k^l - w_*, \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j) - \nabla F(w_k^l)\rangle \tag{26}$$

$$\leq \| w_k^l - w_*\| \cdot \| \sum_{k=1}^{T}\left(\nabla F(w_k^l) - \sum_{j=1}^{S_t}\widetilde{g^j}(w^j_{k-\tau_k^j}, \xi_j)\right)\| \tag{27}$$

$$\leq \| w_k^l - w_*\| \cdot L\| \sum_{k=1}^{T}\sum_{j=1}^{S_t}(w_k^l - w^j_{k-\tau_k^j})\| \tag{28}$$

$$\leq DL\| \sum_{j=1}^{S_t}(w_1^l - w^j_{1-\tau_1^j}) + \ldots + \sum_{j=1}^{S_t}(w_T^l - w^j_{T-\tau_T^j})\| \tag{29}$$

$$= DL\| \sum_{j=1}^{S_t}(w_1^l - w_1^j) + \ldots + \sum_{j=1}^{S_t}(w_T^l - w_T^j) + \sum_{k=1}^{T}\sum_{j=1}^{S_t}\sum_{q=k-\tau_k^j}^{q=k-1}(w_{q+1}^j - w_q^j)\| \tag{30}$$

$$\begin{aligned} = DL\| T\sum_{j=1}^{S_t}(w_1^l - w_1^j) - \sum_{k=1}^{T-1}\sum_{j=1}^{S_t}\alpha_{(T-k)}k\big(\nabla F(w_{(T-k)}^l) - \sum_{j=1}^{S_t}g^j(w^j_{(T-k)}, \xi_j)\big) \\ + \sum_{k=1}^{T}\sum_{j=1}^{S_t}\sum_{q=k-\tau_k^j}^{q=k-1}(w_{q+1}^j - w_q^j)\| \end{aligned} \tag{31}$$

$$\begin{aligned} \leq DL\Bigg(\| T\sum_{j=1}^{S_t}(w_1^l - w_1^j)\| + \| \sum_{k=1}^{T-1}\sum_{j=1}^{S_t}\alpha_{(T-k)}k\big(\nabla F(w_{(T-k)}^l) - \sum_{j=1}^{S_t}g^j(w^j_{(T-k)}, \xi_j)\big)\| \\ + \| \sum_{k=1}^{T}\sum_{j=1}^{S_t}\sum_{q=k-\tau_k^j}^{q=k-1}(w_{q+1}^j - w_q^j)\|\Bigg) \end{aligned} \tag{32}$$

$$\leq DL\| T\sum_{j=1}^{S_t}(w_1^l - w_1^j)\| + DL \cdot 2C^2 m^2 G\sum_{k=1}^{T-1}\alpha_{(T-k)}k + DLCmG\sum_{k=1}^{T}\sum_{q=k-\tau_k^j}^{q=k-1}\alpha_q \tag{33}$$

$$\begin{aligned} \leq DL\| T\sum_{j=1}^{S_t}(w_1^l - w_1^j)\| + 2C^2 m^2 GDL\sum_{k=1}^{T-1}\frac{\alpha k}{((1+\mu)T - k)^2} \\ + DLCmG\sum_{k=1}^{T}\frac{\alpha\tau}{(k - \tau + T)^2} \end{aligned} \tag{34}$$

$$\begin{aligned} < DL\| T\sum_{j=1}^{S_t}(w_1^l - w_1^j)\| + 2\alpha C^2 m^2 GDL(\frac{T}{\mu T + 1} + \ln|\mu T + 1|) \\ + DLCmG\alpha\tau(\frac{1}{1 - \tau + T}) \end{aligned} \tag{35}$$

Inequations 27 and 32 are from triangle inequality. Inequation 28 is from the fact $\nabla F(w_k^l) = \frac{1}{n}\sum_{i=1}^{n}\nabla F(w_k^l, \xi_i) = \sum_{j=1}^{m}g^j(w_k^l, \xi_j)$ and blockwise Lipschitz-continuity. Inequation 29 is from Assumption 1.1 and represents $\sum_{k=1}^{T}(\cdot)$ by a summand sequence.

Combining Inequations 20, 22, 25 and 35 and dividing by T we obtain

$$\begin{aligned} \mathcal{R} < \frac{2\alpha m^2 G^2}{T} + DL\| \sum_{j=1}^{m}(w_1^l - w_1^j)\| + 2\alpha C^2 m^2 GDL(\frac{1}{\mu T + 1} + \frac{\ln|\mu T + 1|}{T}) \\ + DLCmG\alpha\tau(\frac{1}{T(1 - \tau + T)}) = O(\gamma + \frac{1}{\mu T + 1} + \frac{\ln|\mu T + 1|}{T} + \frac{1}{T(1 - \tau + T)}) \end{aligned}$$

□

F THE SECRECY OF $\widetilde{g_k^l}$

We demonstrate that, in CGD, given only the sum of the local gradients computed on different confined models, the adversary can learn no information other than their own inputs and the sum of the local gradients from other honest parties, *i.e.*, $\sum\widetilde{g_k^l}(w_k^l, \xi_l)\mid_{l\in\mathcal{L}\setminus c}$. Our analysis is based on the simulation paradigm [14]. It compares what an adversary can do in a real protocol execution to what it can do in an ideal scenario, which is secure by definition. An indistinguishability between adversary's view in real and ideal scenarios guarantees that it can learn nothing more than their own inputs and the information required for the simulation.

Denote $g_k^{\mathcal{L}'} = \{\widetilde{g_k^l}(w_k^l, \xi_l)\}_{l\in\mathcal{L}'}$ as the local gradients of any subset of participants $\mathcal{L}' \subseteq \mathcal{L}$ at k^{th} iteration. Let $VIEW_{real}(g_k^{\mathcal{L}}, t, \mathcal{P}, c)$ denote their combined views from the execution of a real protocol $\mathcal{P}$. Let $VIEW_{ideal}(g_k^c, z, t, \mathcal{F}_p, c)$ denote the views of c from an ideal execution that securely computes a function $\mathcal{F}_p$, where z is the information required by the simulator $\mathcal{S}$ in the ideal execution for simulation. The following theorem shows that when executing CGD with the threshold t, the joint view of the participants in c can

Table 5: CGD against active MIA with various number of participants.

Datasets	# of Participants	Attack Accuracy
MNIST	10 × 7	50.76%
	100 × 49	51.46%
	1000 × 112	50.57%
CIFAR	10 × 2	50.00%
	100 × 16	50.18%
	1000 × 32	50.06%

be simulated by their own inputs and the sum of the local gradients from the remaining honest nodes, *i.e.,* $\sum \widetilde{g_k^l} \mid_{l\in\mathcal{L}\setminus c}$. Therefore, $\sum \widetilde{g_k^l} \mid_{l\in\mathcal{L}\setminus c}$ is the only information that the adversary can learn.

Theorem 3. *When executing CGD with the threshold t, there exists a simulator $\mathcal{S}$ such that for $\mathcal{L}$ and c, with $c \subseteq \mathcal{L}$ and $|c| \le t$, the output of $\mathcal{S}$ from $VIEW_{ideal}$ is perfectly indistinguishable from the output of $VIEW_{real}$, namely*

$$VIEW_{real}(g_k^{\mathcal{L}}, t, \mathcal{P}, c) \equiv VIEW_{ideal}(g_k^c, z, t, \mathcal{F}_p, c)$$

where

$$z = \sum \widetilde{g_k^l}(w_k^l, \xi_l) \mid_{l\in\mathcal{L}\setminus c} .$$

We omit the proof since it can be easily obtained by inductively defining $\mathcal{S}$ through each training iteration.

G BOUNDED ASYNCHRONY

Let S_t denote a random set of $C \cdot m$ participants, and $\mathcal{A}$ denote the set of malicious participants, and $P_{\mathcal{A}}$ denote the malicious proportion, *i.e.,* $P_{\mathcal{A}} = |\mathcal{A}|/m$. Given a participant l at k^{th} iteration, the probability of any participant j being at t^{th} iteration where $t \le k - \tau$ is

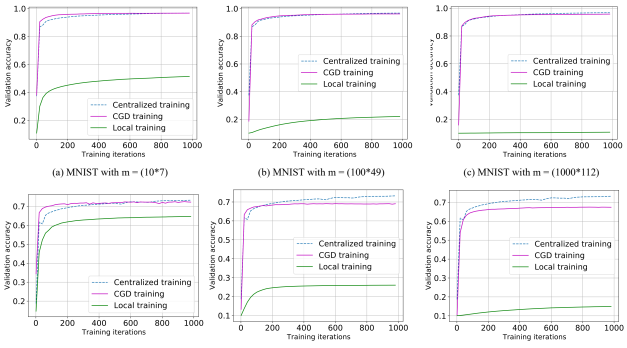
$$\begin{aligned} Pr(t \le k-\tau) &\le Pr^{\tau}(S_t \cap \mathcal{A} \neq \emptyset) = (1 - Pr(S_t \cap \mathcal{A} = \emptyset))^{\tau} \\ &= (1 - \binom{(1-P_{\mathcal{A}})m}{Cm} / \binom{m}{Cm})^{\tau} \quad (\text{with } C < 1 - P_{\mathcal{A}}) \end{aligned} \tag{36}$$

For example, with $P_{\mathcal{A}} = 0.2$ and $m = 30$, which is the commonly used setting in the existing FL literature [9], with $C = 0.2$, the probability of the staleness exceeding $\tau = 30$ is 1.33×10^{-5}.

H DETAILS OF ABLATION STUDIES

H.1 Scalability

Here, we investigate how the number of participants affects CGD's privacy and accuracy performance. Figure 6 summarizes its performance with varying participant number. In general, CGD stays close to the centralized baseline in both validation loss and accuracy in all settings. In all settings, the privacy bound (ϵ, δ) of CGD remains lower than $(0.36, 10^{-5})$, and no significant difference is observed with varying number of participants. Table 5 shows the performance of CGD against active MIA with various number of participants. The results demonstrate that CGD effectively mitigates the threat, suppressing the attack accuracy around 50% in all settings. Table 6 shows the fairness performance of CGD on CIFAR with various number of participants. The standard deviation of test accuracy slightly increases with the growing number of participants, while retaining the superior performance. For example, CGD achieves 7.46% even with a large number of participants (*i.e.,* 32,000 participants), outperforming Ditto [24] which achieves 12.23% with 16 participants (see Table 4).



(a) MNIST with m = (10*7) (b) MNIST with m = (100*49) (c) MNIST with m = (1000*112)

(d) CIFAR with m = (10*2) (e) CIFAR with m = (100*16) (f) CIFAR with m = (1000*32)

Figure 6: Results on the test accuracy for different number of participants in the honest-but-curious setting with privacy bounds (ϵ, δ) at $(0.359, 10^{-5})$ on MNIST and $(0.350, 10^{-5})$ on CIFAR.

Table 6: Fairness performance of CGD on CIFAR.

# of Participants	Standard Deviation of Test Accuracy
10 × 2	4.97%
100 × 16	5.18%
1000 × 32	7.46%

Table 7: Performance on validation accuracy with different initialization parameter λ.

Datasets	# of Participants	$\lambda = 0.1$	$\lambda = 0.06$	$\lambda = 0.01$	$\lambda = 0.001$	Random
MNIST	1000 × 112	95.78%	97.04%	97.74%	97.76%	97.07%
CIFAR	1000 × 32	67.59%	70.17%	73.99%	69.43%	71.01%

H.2 Effect of the Initialization

We conduct two experiments to investigate the effect of λ. We vary λ while keeping other settings unchanging (*i.e.,* $\mu = 0$ and $T = 2000$). In the first experiment, we let all participants use the same λ from 0.1, 0.06, 0.01, to 0.001, as they are around $\frac{1}{\sqrt{n}}$ (where n is the sample size). In the other experiment, we let each participant randomly select its own λ based on the uniform distribution within the range from 0.001 to 0.1.

The results of our first experiment are shown in the first five columns in Table 7. In general, as λ decreases, CGD achieves better performance. This confirms our expectation: decreasing λ would reduce the value of $\| \mathop{\mathbb{E}}_{j\in m^h}(w_1^l - w_1^j)\|$, such that the confined models are closer to the centralized optimum. We have not observed significant difference from $\lambda = 0.1$, 0.06 and 0.01. However, λ cannot be set too small, in order to maintain the numerical stability in neural network [13]. For example, when we lower λ to 0.001, the performance on CIFAR starts decreasing.

The results of our second experiment are shown in the last column in Table 7. When the participants uniformly randomize their λs from 0.001 to 0.1, the performance of CGD is still comparable with the centralized mode. For example, with $m = (1000 \times 112)$ participants on MNIST and $m = (1000 \times 32)$ on CIFAR, CGD achieves 97.07% and 71.01% in the test accuracy respectively. This suggests that CGD keeps robust when the λs of its participants differ by two orders of magnitude.