Scilight

Journal of Machine Learning and Information Security
https://www.sciltp.com/journals/jmlis

Article

Check for updates

DeFL-VP: Decentralized Federated Learning with Variance Reduction and Differential Privacy

Ruihong Xiu [1], Rui Liu [1,*], Weixu Zhang [1], Xiaokai Liao [2] and Kouichi Sakurai [3]

[1] Graduate School of Information Science and Electrical Engineering, Kyushu University, Fukuoka 819-0395, Japan
[2] Department of Computer Science, University of Rochester, New York, NY 14620, USA
[3] Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka 819-0395, Japan
* Correspondence: xiaorui121212@gmail.com

How To Cite: Xiu, R.; Liu, R.; Zhang, W.; et al. DeFL-VP: Decentralized Federated Learning with Variance Reduction and Differential Privacy. *Journal of Machine Learning and Information Security* **2026**, *2*(2), 12. https://doi.org/10.53941/jmlis.2026.100012

Received: 24 April 2026
Revised: 16 June 2026
Accepted: 24 June 2026
Published: 25 June 2026

Abstract: With the widespread application of federated learning (FL) in data security scenarios, striking a balance between communication efficiency, system robustness, and data privacy has become a key challenge. Existing FL frameworks typically rely on a central server, which introduces single points of failure and potential privacy risks. Although communication compression can improve convergence efficiency, its application in decentralized scenarios remains limited, and it lacks rigorous privacy protection mechanisms. This paper proposes a new decentralized federated learning algorithm framework, DeFL-VP, which integrates communication compression, client-side variance reduction, and differential privacy (DP) mechanisms within a graph-structured network. Specifically, we introduce local control variables into the underlying architecture of decentralized federated learning to reduce gradient variance among clients, while designing a differential privacy algorithm based on a random response mechanism to protect user privacy. Theoretically, we prove the convergence of the proposed algorithm under non-convex smoothness conditions and provide an upper bound on the impact of privacy noise on the convergence rate. Experimental results under a decentralized non-IID setting show that DeFL-VP achieves stable convergence performance, maintains model utility, and obtains a lower attack AUC under membership inference attacks compared with the baseline methods.

Keywords: federated learning; differential privacy; communication compression; inference attacks

1. Introduction

With the rapid development of mobile devices and the Internet of Things (IoT), large-scale distributed data is being generated on a massive scale. How to conduct efficient model training while protecting data privacy has become a critical issue in the field of machine learning (ML) [1–3]. By training models on local devices and exchanging only model parameters, FL effectively avoids the centralized sharing of raw data, thereby demonstrating significant advantages in privacy protection [4,5]. At the same time, FL can fully leverage the computing power of heterogeneous edge devices to achieve distributed collaborative optimization. It has been widely applied in privacy-sensitive scenarios such as smart healthcare and data analysis, providing an effective solution to the problem of data silos [6–9].

Although FL enables robust privacy protection, most existing FL methods rely on a central server for model aggregation. This centralized FL architecture has certain limitations in practical deployment. As a single point of failure, the central server is prone to becoming a bottleneck in FL systems and may even maliciously steal the privacy of FL clients, thereby compromising the system's robustness and reliability [10–14]. Furthermore, as the number of FL devices increases, frequent communication between the server and clients incurs significant communication overhead.

Existing research has attempted to reduce communication overhead through communication compression techniques or enhance privacy performance via DP mechanisms [15–20]. However, most of these methods are designed within a centralized FL framework and are difficult to directly extend to decentralized environments. Existing methods still struggle to achieve the optimal balance between robustness, communication efficiency, and privacy protection, thus necessitating a new FL framework to address these issues [21–24].

To address these issues, decentralized federated learning (DFL) replaces a central server with peer-to-peer communication between nodes, thereby reducing server-side bottlenecks, single points of failure, and the risk of privacy leaks that may arise from a central aggregator. Unlike centralized federated learning (CFL), DFL no longer relies on a unified global aggregation process; model updates are primarily propagated incrementally through information exchange between neighboring nodes. This structure enhances the system's decentralization but also complicates the interplay between local update bias, privacy perturbations, and communication constraints. However, designing a federated learning framework that balances training efficiency, privacy protection, and system stability in a decentralized environment remains a significant challenge.

(1) The first challenge is that non-independent and identically distributed (Non-IID) data is more difficult to handle in DFL. CFL can mitigate local gradient bias to some extent by aggregating updates from multiple clients via a server. In contrast, in DFL, each client can only exchange information with a limited number of neighboring nodes. Update biases caused by heterogeneous data are difficult to correct globally in a timely manner and may continue to propagate through the network over multiple communication rounds. Therefore, maintaining stable convergence under heterogeneous data distributions is a key issue in DFL.

(2) The second challenge is that the impact of privacy perturbations may be more pronounced in DFL. Due to the absence of a central aggregation process, random perturbations cannot be fully offset by large-scale global averaging. Instead, they may interact with the peer-to-peer communication process and the network topology, gradually accumulating as information propagates. Consequently, it is necessary to design a privacy protection mechanism suitable for decentralized environments that protects local information while minimizing its negative impact on the model optimization process.

To address the aforementioned challenges, this paper proposes a new decentralized federated learning framework called DeFL-VP. Based on a decentralized network architecture, this framework integrates communication compression, client-side variance reduction, and DP mechanisms. Specifically, this paper reduces gradient variance among clients by introducing local control variables and designs a differential privacy algorithm based on a stochastic response mechanism to perturb information during the model update process, thereby effectively protecting user privacy. Building on this foundation, DeFL-VP mitigates performance degradation caused by data heterogeneity and noise perturbations while ensuring privacy constraints, thereby establishing a secure and efficient federated learning framework. Furthermore, the proposed method significantly reduces the risk of privacy leakage in the face of inference attacks, further enhancing the system's security. Specifically, the main contributions of this paper are as follows.

- We propose DeFL-VP, a federated learning framework that integrates communication compression, variance reduction, and differential privacy mechanisms in a decentralized environment.
- We design a privacy-preserving algorithm based on a stochastic response mechanism that is suitable for decentralized scenarios and exhibits low utility loss. We have also conducted a theoretical analysis of its privacy-preserving performance.
- We prove the convergence of the proposed DeFL-VP and analyze its stability and reliability under communication-constrained conditions.

The remainder of this paper is structured as follows. Section 2 discusses related research. Section 3 presents the problem formulation and algorithm design. Section 4 conducts theoretical analysis and proofs. Section 5 performs experimental validation. Section 6 summarizes this paper and discusses future work.

2. Related Work

Federated Learning. As a privacy-preserving distributed learning paradigm, FL has seen extensive research progress. Existing work primarily focuses on CFL frameworks, in which a server coordinates multiple clients to perform model training and parameter aggregation. Within the CFL framework, FedAvg and its variants have become the most commonly used optimization methods and have achieved good results in various application scenarios. A significant body of research has been dedicated to improving the communication efficiency and convergence performance of CFL [25]. At the same time, some studies have focused on improving FL optimization

algorithms.Yue et al. [26] proposed an adaptive LDP method based on a noise scaler to address the model drift issue caused by adding noise to client models. This approach achieves higher accuracy while ensuring fast convergence. However, most of these methods rely on centralized server architectures, which fundamentally limit the scalability and robustness of FL systems. On the other hand, DFL enhances system robustness and scalability by replacing a central server with peer-to-peer communication among nodes. Zhao et al. [27] proposed PVD-FL, a privacy-preserving and verifiable DFL framework that employs a cryptographic matrix algorithm to ensure the confidentiality of model updates, thereby protecting privacy and guaranteeing the integrity of the training process. However, DFL frameworks still face significant challenges in terms of convergence speed and stability. Most existing research focuses on network architecture or communication protocol design, while the optimization efficiency and privacy protection remains limited [28–31]. Therefore, there is a need to further explore FL frameworks in decentralized scenarios that can simultaneously address convergence efficiency and privacy protection.

Differential Privacy. DP provides quantifiable theoretical guarantees for privacy protection in federated learning and is therefore widely used in the model training process. Most existing studies reduce the risk of sensitive information leakage by adding noise to model parameters or gradient updates, with the Gaussian mechanism being one of the most commonly used methods. However, the introduction of noise typically has a certain impact on model performance, leading to a trade-off between privacy protection and model effectiveness [32–34]. Based on the location where noise is injected, existing DP-FL methods can be broadly categorized into Centralized differential privacy (CDP) and local differential privacy (LDP). LDP involves clients perturbing model updates before participating in training, whereas CDP implements privacy protection during the server aggregation phase. Although these methods can mitigate privacy risks from inference attacks to some extent, the continuous introduction of random noise often affects model convergence speed and leads to a decline in accuracy. To resolve the conflict between privacy protection and model performance, existing research has improved DP mechanisms from various perspectives. For example, Huang et al. [35] proposed a Riemannian manifold federated learning framework based on differential privacy and analyzed its convergence properties. The Camel method proposed by Xu et al. [36] combines secret sharing with a shuffling mechanism, achieving a good balance between privacy guarantees and model utility. Although some progress has been made, most existing research is based on centralized federated learning frameworks, and there remains a lack of sufficient study on their applicability to decentralized federated learning scenarios. In DFL, model updates must propagate directly between nodes. The randomness introduced by privacy perturbations may accumulate during network communication and interact with the network topology, thereby affecting the stability and convergence performance of model training. Therefore, how to balance privacy protection and optimization efficiency in a decentralized environment remains a significant challenge in DFL research.

Security in Federated Learning. In practical deployment, FL systems face various security threats, among which model poisoning attacks and Byzantine attacks are considered the primary factors affecting the reliability of FL systems. Such attacks, by tampering with local training data or uploading malicious model updates, can lead to a decline in global model performance and even hinder the normal convergence of the entire training process. To address these issues, researchers have proposed various Byzantine-robust mechanisms to enhance the stability and security of FL systems in malicious environments [37,38]. Nowroozi et al. [39] proposed a stochastic deep feature selection method, which effectively mitigates the impact of feature poisoning attacks on model performance by randomly selecting client features during training. Yang et al. [40] further proposed the MSGuard framework, which combines a cosine similarity mechanism, symbolic statistics, and spectral analysis strategies to identify and filter out anomalous model updates, thereby enhancing the model's robustness and convergence performance under Byzantine attack scenarios. Although the aforementioned methods have achieved certain results in improving FL security, most studies still rely on the CFL architecture. In a DFL environment, the absence of a unified trusted server responsible for model aggregation makes it difficult to directly transfer and apply many robust aggregation strategies that rely on central nodes. Furthermore, factors such as inter-node communication topologies and client data heterogeneity exert additional influences on the model training process, further complicating system design and optimization. Therefore, how to simultaneously balance system robustness and training efficiency in a decentralized setting remains a problem worthy of further exploration in current FL research.

3. Methodology

To address the challenge of balancing convergence efficiency, privacy protection, and system robustness in decentralized federated learning, this paper proposes a new decentralized federated learning framework, DeFL-VP. Built upon a decentralized communication architecture, this framework integrates communication compression, client-side variance reduction, and a differential privacy mechanism based on Randomized Response (RR) into a unified training process, thereby improving model training efficiency while ensuring data privacy.

The overall workflow of DeFL-VP is illustrated in Figure 1. At the beginning of each training round, clients compute stochastic gradients using local data and correct them via a variance reduction mechanism to mitigate the bias caused by data heterogeneity. Subsequently, privacy-preserving perturbations based on Randomized Response (RR) are applied to the corrected gradients to ensure local privacy protection. To reduce the information transmission overhead between nodes, the perturbed gradients are further compressed. Finally, nodes exchange model parameters and perform collaborative updates through peer-to-peer communication with neighbors, thereby enabling decentralized training.

Most existing differential privacy methods rely on continuous noise to perturb model updates, whereas DeFL-VP employs a discrete random response mechanism to ensure privacy protection. This mechanism effectively preserves gradient directional information while satisfying privacy constraints. Unlike approaches that treat privacy protection and model optimization separately, DeFL-VP jointly designs the stochastic response mechanism with a variance reduction strategy, ensuring that gradients are corrected before privacy perturbation. This mitigates the impact of data heterogeneity and stochastic perturbations on the training process. Experimental results demonstrate that this design maintains good convergence performance and training stability while ensuring privacy protection.

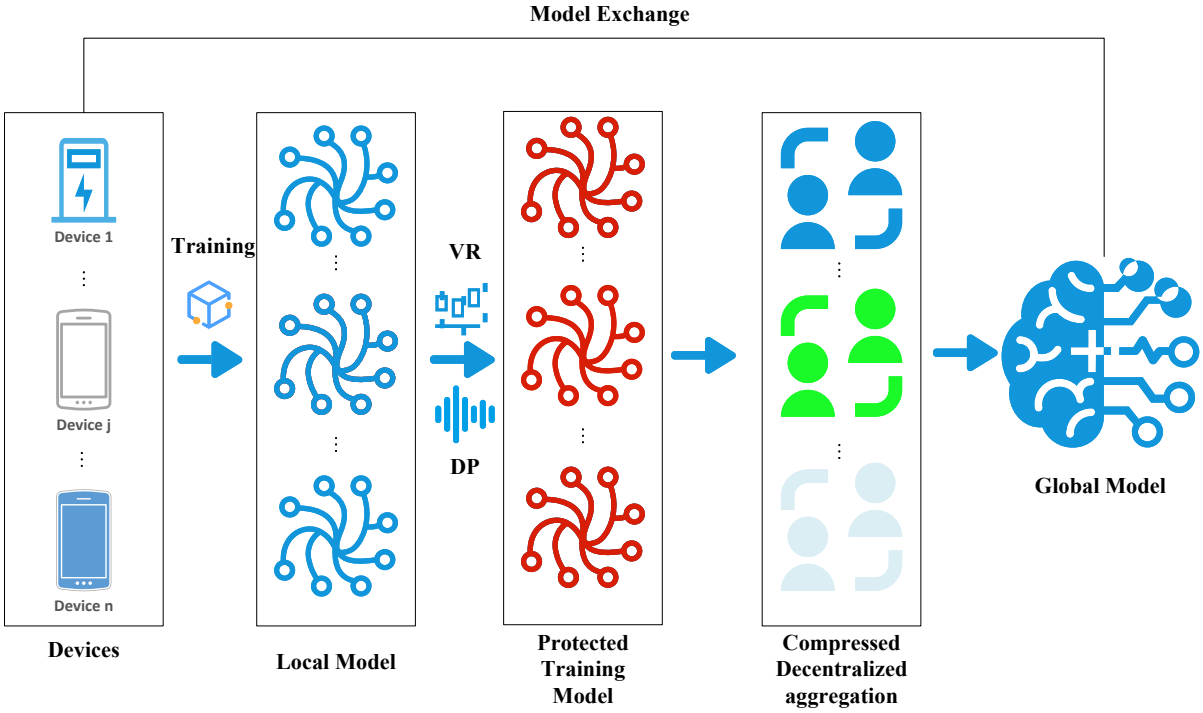


Figure 1. The Structure of DeFL-VP

As shown in Algorithm 1, during each communication round, each client first trains the model using local data and computes stochastic gradients. Given the differences in client data distributions, DeFL-VP introduces a variance reduction mechanism during the gradient update phase to correct local gradients, thereby reducing gradient discrepancies among different clients. After correction, clients do not directly upload gradient information but instead generate privatized gradient estimates using a differential privacy mechanism based on random responses. This mechanism protects privacy by randomly perturbing the gradient direction while preserving as much useful information as possible.

During the model aggregation phase, DeFL-VP employs a decentralized communication approach, where each node exchanges information and updates parameters only with its neighbors, without relying on a central server for unified aggregation. To reduce data transmission between nodes, the privatized gradients undergo compression before transmission. The combined effects of variance reduction, random response privacy protection, and communication compression during the training process enable DeFL-VP to maintain good training efficiency and convergence stability while ensuring privacy security.

Algorithm 1: Decentralized Training Procedure of DeFL-VP

Input: Initial $\{x_i^0\}$, variates $\{h_i^0\}$, stepsize η, privacy budget ϵ, compressor $\mathcal{C}$
Output: Final models $\{x_i^T\}$

for $t = 0, 1, \ldots, T-1$ **do**
  for *client* i *in parallel* **do**
    Compute g_i^t;
    Compute neighborhood average:

$$\bar{h}_i^t = \sum_{j \in \mathcal{N}(i)} w_{ij} h_j^t$$

    (Variance Reduction)

$$v_i^t = g_i^t - h_i^t + \bar{h}_i^t$$

    Update control variate:

$$h_i^{t+1} = (1-\alpha) h_i^t + \alpha g_i^t$$

    (Randomized Response DP)

$$\tilde{v}_i^t = \mathrm{RR}(v_i^t, \epsilon)$$

    (Compression)

$$q_i^t = \mathcal{C}(\tilde{v}_i^t)$$

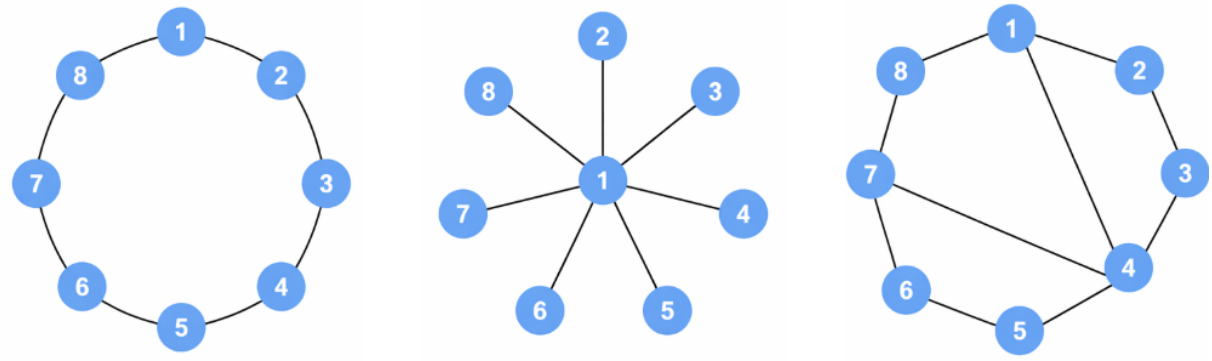
    Receive $\{q_j^t\}_{j \in \mathcal{N}(i)}$;
    Update model:

$$x_i^{t+1} = \sum_j w_{ij} x_j^t - \eta \sum_j w_{ij} q_j^t$$

  end
end

3.1. DeFL-VP with Communication Compression

In DeFL-VP, clients exchange model updates with neighboring nodes according to a predefined communication graph and perform local model optimization based on neighborhood information. The communication topology influences how model parameters propagate through the network, thereby affecting training efficiency, system stability, and the model convergence process. Common communication structures include ring topologies, star topologies, and sparse point-to-point topologies, as shown in Figure 2. Considering that star topologies rely on a central node and ring topologies have relatively slow information propagation speeds, this paper selects a sparse point-to-point topology as the communication structure to balance communication efficiency and system reliability.



(a) Ring Topology **(b)** Star Topology **(c)** Peer-to-Peer Sparse Topology

Figure 2. Typical communication topologies in DFL.

During the transmission of model updates, DeFL-VP introduces a compression mechanism to quantize or sparsify gradients or model updates, as detailed in Algorithm 2. By reducing the amount of data to be transmitted, this strategy effectively reduces the communication burden between nodes. However, the compression operation introduces some information loss and may affect the accuracy of model updates.

Algorithm 2: Unbiased Compression

Input: Vector $\tilde{v}_i^t$, compressor $\mathcal{C}$
Output: Compressed vector q_i^t

$$q_i^t = \mathcal{C}(\tilde{v}_i^t)$$

return q_i^t

When the stochastic response privacy mechanism and communication compression coexist, the errors generated by both may accumulate during training, thereby affecting model convergence. To mitigate this impact, DeFL-VP introduces a variance reduction mechanism during the gradient update phase to correct local gradients. The corrected gradients better reflect the optimization direction while reducing gradient drift and noise accumulation among clients, thereby enhancing the stability of the training process in a decentralized environment.

3.2. DeFL-VP with Randomized Response

Although FL avoids the centralized storage of raw data, the model parameters and gradient information exchanged during training may still compromise user privacy. Existing research has shown that attackers can exploit model update information to carry out membership inference attacks to determine whether a specific sample was used in training; they can also recover the statistical characteristics of training data through model inversion attacks, or even reconstruct some of the original inputs. In DFL scenarios, model updates must be transmitted across multiple nodes, further expanding the scope of information propagation and thereby exacerbating the risk of privacy leakage.

To address these issues, this paper introduces a local differential privacy mechanism based on random responses on the client side, as detailed in Algorithm 3. Most existing differential privacy methods employ continuous noise perturbation techniques, such as Gaussian mechanisms, whereas this paper adopts a discrete randomization strategy to process gradient information. The client first applies Randomized Response perturbation to the gradient information and then uses the perturbed results to participate in subsequent training and communication processes. This mechanism preserves partial gradient structural information while meeting privacy protection requirements, thereby supporting subsequent optimization analysis.

Algorithm 3: Randomized Response Mechanism (RR)

Input: Vector v_i^t, privacy budget ϵ
Output: Privatized vector $\tilde{v}_i^t$

Normalize:

$$s_i^t = \operatorname{sign}(v_i^t)$$

Set probability:

$$p = \frac{e^\epsilon}{e^\epsilon + 1}$$

for *Each dimension* k **do**

Sample:

$$\tilde{s}_{i,k}^t = \begin{cases} s_{i,k}^t, & \text{w.p. } p \\ -s_{i,k}^t, & \text{w.p. } 1-p \end{cases}$$

end

Rescale:

$$\tilde{v}_i^t = \frac{1}{2p-1}\tilde{s}_i^t$$

return $\tilde{v}_i^t$

Let g_i^t denote the local stochastic gradient computed by client i in training round t. To implement stochastic response perturbation, we first map the gradient to a binary vector:

$$s_i^t = \operatorname{sign}(g_i^t),$$

where each element of the vector belongs to the set $-1, +1$. On this basis, we apply a stochastic response mechanism

independently to each dimension

$$\tilde{s}_{i,j}^t = \begin{cases} s_{i,j}^t, & \text{with probability } p, \\ -s_{i,j}^t, & \text{with probability } 1-p, \end{cases}$$

where the flipping probability is determined by the privacy budget ϵ as

$$p = \frac{e^\epsilon}{e^\epsilon + 1}.$$

To ensure that the privatized gradients still reflect information about the true gradient direction, the results obtained after applying the random response are reconstructed as follows.

$$\tilde{g}_i^t = \frac{1}{2p-1}\tilde{s}_i^t.$$

The reconstructed gradient estimates satisfy the ϵ-local differential privacy constraint, meaning that the probability of any output result does not change significantly due to the variation of a single data sample. Compared to continuous noise perturbation methods such as the Gaussian mechanism, stochastic response introduces randomness through bit flipping, without the need to directly add continuous noise to the gradients. This discrete processing approach is better suited for DFL scenarios, as model updates primarily rely on local communication and collaborative aggregation among nodes.

The privacy protection provided by random response typically comes with some performance trade-offs. In DFL, data distributions often vary across clients, and local gradients inherently exhibit significant shifts. When random response perturbations are further applied to gradient updates, the differences in updates between clients become more pronounced, thereby affecting the model's convergence speed and final accuracy.

On the other hand, the impact of privacy perturbations is also constrained by the network structure. Since clients can only exchange information with neighboring nodes, random perturbations cannot be fully offset during large-scale aggregation as in CFL; instead, they may accumulate as information propagates, causing additional effects on model training.

Based on these considerations, DeFL-VP is designed to combine differential privacy mechanisms with variance reduction strategies, rather than simply embedding privacy protection as an independent module within the training process. By introducing local control variables to correct gradients, the bias caused by data heterogeneity and random response noise can be mitigated, resulting in more stable model updates. Experimental results demonstrate that this design maintains good convergence performance while ensuring privacy protection, making it more suitable for DFL environments.

3.3. Variance Reduction under Heterogeneous Clients

In a DFL environment, clients typically originate from different users or devices, and their local data varies significantly in terms of sample size, class distribution, and statistical characteristics. As a result, the gradients computed by different clients often point in different directions, making it difficult for the model to converge quickly. The more pronounced the data heterogeneity, the more pronounced the optimization drift between clients tends to be.

In a CFL environment, the server can aggregate model updates from a large number of clients, so some gradient discrepancies are mitigated during the aggregation process. In contrast, DFL lacks a unified coordination node; each client can only exchange information with neighboring nodes, making it difficult to correct discrepancies arising from local updates in a short period of time. As the number of training iterations increases, these discrepancies may continue to propagate throughout the network, further affecting the stability of model training.

The introduction of differential privacy mechanisms further complicates these issues. Random response mechanisms protect privacy by randomly flipping the signs of gradients, but this also introduces additional random noise. When client data exhibits strong heterogeneity, this noise may further amplify the differences in updates across clients, making client drift more pronounced and affecting the model's convergence speed.

To address this, DeFL-VP introduces a variance reduction mechanism on the client side to correct gradients, as detailed in Algorithm 1. Each client maintains a control variable to record historical gradient information and uses this, combined with information from neighboring nodes, to adjust the current gradient update direction. After correction, local updates align more closely with the global optimization direction, thereby reducing gradient discrepancies among clients.

In DeFL-VP, the variance reduction step is executed prior to privacy perturbation. The purpose of this is to

correct gradients before introducing the stochastic response mechanism, ensuring that effective information in the gradients is better preserved and reducing the impact of stochastic perturbations on the subsequent optimization process. Experimental results indicate that this design can mitigate, to some extent, the performance degradation caused by the combined effects of data heterogeneity and privacy perturbation.

On the other hand, the variance reduction mechanism also complements communication compression. Communication compression introduces additional estimation errors, while the stochastic response mechanism increases the randomness in the update process; when combined, these factors may lead to greater instability in the training process. By correcting gradients through controlled variables, DeFL-VP mitigates the effects of compression errors and privacy noise, thereby maintaining a relatively stable optimization process.

4. Theoretical Analysis

In this section, we provide the theoretical analysis of the proposed DeFL-VP framework from two aspects. First, we show that the local privacy-preserving mechanism used in DeFL-VP satisfies local differential privacy. Second, under standard smooth nonconvex assumptions, we establish the convergence guarantee of DeFL-VP and characterize the influence of randomized response perturbation, client heterogeneity, compression error, and decentralized topology on the final convergence behavior.

4.1. Preliminaries and Assumptions

Consider a decentralized system with M clients connected over an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Let $P = [p_{ij}] \in \mathbb{R}^{M\times M}$ denote the mixing matrix associated with the communication graph. We assume that P is symmetric and doubly stochastic, i.e.,

$$P\mathbf{1} = \mathbf{1}, \qquad \mathbf{1}^\top P = \mathbf{1}^\top, \tag{1}$$

and that

$$\rho\left(P - \frac{1}{M}\mathbf{1}\mathbf{1}^\top\right) < 1. \tag{2}$$

The global optimization objective is defined as

$$F(x) = \frac{1}{M}\sum_{i=1}^{M} F_i(x), \tag{3}$$

where $F_i(x)$ denotes the local objective function of client i.

For the convenience of analysis, we impose the following assumptions.

Assumption 1. *For any client $i \in \{1, \ldots, M\}$, the local objective function F_i is L-smooth, i.e., for any $x, y \in \mathbb{R}^d$,*

$$\|\nabla F_i(x) - \nabla F_i(y)\| \le L\|x - y\|. \tag{4}$$

Assumption 2. *For any client i and any model parameter x, the local stochastic gradient satisfies*

$$\mathbb{E}[g_i(x, \xi_i)] = \nabla F_i(x), \tag{5}$$

and

$$\mathbb{E}\|g_i(x, \xi_i) - \nabla F_i(x)\|^2 \le \sigma^2. \tag{6}$$

Assumption 3. *The client heterogeneity is bounded in the sense that*

$$\frac{1}{M}\sum_{i=1}^{M}\|\nabla F_i(x) - \nabla F(x)\|^2 \le \delta^2, \qquad \forall x \in \mathbb{R}^d. \tag{7}$$

Assumption 4. *The compression operator $\mathcal{Q}(\cdot)$ is unbiased and satisfies*

$$\mathbb{E}[\mathcal{Q}(u)] = u, \tag{8}$$

and

$$\mathbb{E}\|\mathcal{Q}(u) - u\|^2 \le \beta\|u\|^2, \qquad \forall u \in \mathbb{R}^d, \tag{9}$$

where $\beta \ge 0$ is the compression variance parameter.

In DeFL-VP, each client maintains a local control variate c_i^t for variance reduction. For client i, the neighborhood-averaged control variate is defined as

$$\bar{c}_i^t = \sum_{j \in \mathcal{N}(i)} p_{ij} c_j^t. \tag{10}$$

Given the local stochastic gradient g_i^t, each client first performs variance reduction to construct the corrected gradient

$$u_i^t = g_i^t - c_i^t + \bar{c}_i^t. \tag{11}$$

To ensure privacy protection, the client then applies a randomized response mechanism to u_i^t. Specifically, let

$$z_i^t = \operatorname{sign}(u_i^t), \tag{12}$$

where each coordinate of z_i^t takes values in $\{-1, +1\}$. For each dimension k, the privatized sign is generated as

$$\hat{z}_{i,k}^t = \begin{cases} z_{i,k}^t, & \text{with probability } q, \\ -z_{i,k}^t, & \text{with probability } 1-q, \end{cases} \tag{13}$$

where

$$q = \frac{e^\epsilon}{e^\epsilon + 1}, \tag{14}$$

and $\epsilon > 0$ is the privacy budget.

To preserve unbiasedness with respect to the sign estimator, the privatized update is reconstructed as

$$\hat{u}_i^t = \frac{1}{2q-1} \hat{z}_i^t. \tag{15}$$

After privacy perturbation, the client compresses the privatized update:

$$y_i^t = \mathcal{Q}(\hat{u}_i^t). \tag{16}$$

The decentralized update rule of client i is

$$x_i^{t+1} = \sum_{j \in \mathcal{N}(i)} p_{ij} x_j^t - \eta \sum_{j \in \mathcal{N}(i)} p_{ij} y_j^t, \tag{17}$$

where $\eta > 0$ is the stepsize.

Define the network average model as

$$\bar{x}^t = \frac{1}{M} \sum_{i=1}^M x_i^t. \tag{18}$$

Since P is doubly stochastic, the average iterate satisfies

$$\bar{x}^{t+1} = \bar{x}^t - \eta \cdot \frac{1}{M} \sum_{i=1}^M y_i^t. \tag{19}$$

4.2. Differential Privacy Guarantee

In DeFL-VP, privacy protection is achieved through a client-side randomized response mechanism. Since the released message is constructed from the sign vector of the corrected local update, the corresponding privacy analysis is conducted under the framework of local differential privacy rather than Gaussian differential privacy.

Definition 1 (Adjacent Data)**.** *For client i at communication round t, let*

$$z_i^t = \operatorname{sign}(u_i^t) \in \{-1, +1\}^d,$$

where u_i^t denotes the variance-reduced local update and the sign operator is applied coordinate-wise. Two inputs $z, z' \in \{-1, +1\}^d$ are called adjacent if they differ in at least one coordinate and correspond to two possible local sign vectors before privatization. For simplicity, we assume that $\Pr(u_{i,k}^t = 0) = 0$ for all coordinates k, or equivalently, ties are broken deterministically.

Definition 2 (Local Differential Privacy)**.** *A randomized mechanism $\mathcal{R}$ satisfies ε-local differential privacy (ε-LDP) if for any two adjacent inputs a, a' and any measurable set $\mathcal{S}$, it holds that*

$$\Pr[\mathcal{R}(a) \in \mathcal{S}] \leq e^{\varepsilon} \Pr[\mathcal{R}(a') \in \mathcal{S}]. \tag{20}$$

For each coordinate $k \in \{1, \ldots, d\}$, DeFL-VP applies randomized response independently:

$$\hat{z}_{i,k}^t = \begin{cases} z_{i,k}^t, & \text{with probability } r, \\ -z_{i,k}^t, & \text{with probability } 1 - r, \end{cases} \tag{21}$$

where

$$r = \frac{e^{\varepsilon_0}}{e^{\varepsilon_0} + 1}, \tag{22}$$

and $\varepsilon_0 > 0$ denotes the privacy budget allocated to each coordinate.

The full privatized sign vector is given by

$$\hat{z}_i^t = \big(\hat{z}_{i,1}^t, \ldots, \hat{z}_{i,d}^t\big). \tag{23}$$

We first establish the privacy guarantee at the single-coordinate level.

Lemma 1. *For any coordinate k, the mechanism*

$$\mathcal{R}_k : z_{i,k}^t \mapsto \hat{z}_{i,k}^t$$

satisfies ε_0-local differential privacy.

Proof. Since $z_{i,k}^t \in \{-1, +1\}$, it is sufficient to consider the two possible inputs $+1$ and -1. For any output $y \in \{-1, +1\}$, we have

$$\Pr[\hat{z}_{i,k}^t = y \mid z_{i,k}^t = y] = r, \qquad \Pr[\hat{z}_{i,k}^t = y \mid z_{i,k}^t = -y] = 1 - r.$$

Hence,

$$\frac{\Pr[\hat{z}_{i,k}^t = y \mid z_{i,k}^t = y]}{\Pr[\hat{z}_{i,k}^t = y \mid z_{i,k}^t = -y]} = \frac{r}{1 - r}. \tag{24}$$

Substituting

$$r = \frac{e^{\varepsilon_0}}{e^{\varepsilon_0} + 1},$$

we obtain

$$\frac{r}{1 - r} = e^{\varepsilon_0}. \tag{25}$$

Therefore, for any measurable output event $\mathcal{S}$,

$$\Pr[\mathcal{R}_k(a) \in \mathcal{S}] \leq e^{\varepsilon_0} \Pr[\mathcal{R}_k(a') \in \mathcal{S}], \tag{26}$$

which proves that $\mathcal{R}_k$ satisfies ε_0-LDP. □

We next extend the result to the full sign vector.

Theorem 1. *Suppose randomized response is applied independently to all d coordinates, each with privacy budget ε_0. Then the released sign vector $\hat{z}_i^t$ satisfies*

$$d\varepsilon_0\text{-LDP}. \tag{27}$$

Equivalently, if the target privacy budget for the full vector is ε, one can choose

$$\varepsilon_0 = \frac{\varepsilon}{d}, \tag{28}$$

so that $\hat{z}_i^t$ satisfies ε-LDP.

Proof. For any input vectors $z, z' \in \{-1, +1\}^d$ and any output vector $y \in \{-1, +1\}^d$, the conditional probability

factorizes as

$$\Pr[\hat{z}_i^t = y \mid z_i^t = z] = \prod_{k=1}^{d} \Pr[\hat{z}_{i,k}^t = y_k \mid z_{i,k}^t = z_k]. \tag{29}$$

Hence,

$$\frac{\Pr[\hat{z}_i^t = y \mid z_i^t = z]}{\Pr[\hat{z}_i^t = y \mid z_i^t = z']} = \prod_{k=1}^{d} \frac{\Pr[\hat{z}_{i,k}^t = y_k \mid z_{i,k}^t = z_k]}{\Pr[\hat{z}_{i,k}^t = y_k \mid z_{i,k}^t = z_k']}. \tag{30}$$

By Lemma 1, each factor is bounded by e^{ε_0}, and therefore

$$\frac{\Pr[\hat{z}_i^t = y \mid z_i^t = z]}{\Pr[\hat{z}_i^t = y \mid z_i^t = z']} \leq e^{d\varepsilon_0}. \tag{31}$$

This proves that the full vector release satisfies $d\varepsilon_0$-LDP. □

To obtain an unbiased signed estimator, DeFL-VP reconstructs the privatized update as

$$\hat{u}_i^t = \frac{1}{2r-1}\hat{z}_i^t. \tag{32}$$

Since this reconstruction is a deterministic transformation of $\hat{z}_i^t$, it does not alter the privacy guarantee due to the post-processing property of differential privacy.

Corollary 1. *The reconstructed update $\hat{u}_i^t$ satisfies the same privacy guarantee as $\hat{z}_i^t$. In particular, if $\varepsilon_0 = \varepsilon/d$, then $\hat{u}_i^t$ satisfies ε-LDP.*

Proof. The mapping

$$\hat{z}_i^t \mapsto \hat{u}_i^t = \frac{1}{2r-1}\hat{z}_i^t$$

is deterministic. By the post-processing property of local differential privacy, any deterministic transformation of an ε-LDP output remains ε-LDP. Therefore, $\hat{u}_i^t$ inherits the same privacy guarantee as $\hat{z}_i^t$. □

When the training process proceeds over multiple communication rounds, the overall privacy budget accumulates according to the sequential composition property of local differential privacy.

Corollary 2. *Suppose each communication round of DeFL-VP satisfies ε-LDP. Then after T rounds, the overall mechanism satisfies*

$$T\varepsilon\text{-LDP}. \tag{33}$$

Proof. The result follows directly from the sequential composition theorem of local differential privacy. Since each round releases one ε-LDP message, the total privacy loss after T rounds is additive and bounded by $T\varepsilon$. □

The above lemma, theorem, and corollaries show that the privacy-preserving mechanism in DeFL-VP provides rigorous local differential privacy guarantees without relying on a centralized server, making it naturally compatible with decentralized federated learning.

4.3. Convergence Analysis

We next analyze the convergence behavior of DeFL-VP under smooth nonconvex objectives. Different from Gaussian perturbation-based methods, DeFL-VP applies randomized response to the sign vector of the variance-reduced local update. Therefore, the transmitted message is a debiased stochastic sign estimator rather than an additive noisy gradient. Consequently, the convergence analysis is conducted under a sign-based stochastic descent framework.

For notational consistency, we redefine the key variables. Let M denote the number of clients, and let $P = [p_{ij}]$ be the mixing matrix. Define the variance-reduced local update

$$u_i^t = g_i^t - c_i^t + \bar{c}_i^t, \qquad \bar{c}_i^t = \sum_{j \in \mathcal{N}(i)} p_{ij} c_j^t. \tag{34}$$

Let $z_i^t = \operatorname{sign}(u_i^t)$ denote the sign vector. The randomized response mechanism produces $\tilde{z}_i^t$, and we define the debiased estimator

$$\hat{z}_i^t = \frac{1}{2r-1}\tilde{z}_i^t, \tag{35}$$

where $r = \frac{e^{\varepsilon_0}}{e^{\varepsilon_0}+1}$.

The final transmitted message is given by

$$y_i^t = \mathcal{Q}(\hat{z}_i^t). \tag{36}$$

We impose the following additional assumptions.

Assumption 5. *There exists a constant $\kappa > 0$ such that*

$$\mathbb{E}\big[\langle \operatorname{sign}(g_i^t), \nabla F_i(x_i^t)\rangle \big| \mathcal{F}_t\big] \geq \kappa \|\nabla F_i(x_i^t)\|_1. \tag{37}$$

Assumption 6. *There exists $\mu > 0$ such that*

$$\left\langle \nabla F(\bar{x}^t), \frac{1}{M}\sum_{i=1}^{M}\Big(\operatorname{sign}(\nabla F_i(x_i^t)) - c_i^t + \bar{c}_i^t\Big)\right\rangle \geq \mu \|\nabla F(\bar{x}^t)\|_1 - \Delta_{\text{net}}, \tag{38}$$

where $\Delta_{\text{net}} \geq 0$ characterizes the bias induced by decentralized inconsistency.

Assumption 7. *There exists $C > 0$ such that*

$$\frac{1}{M}\sum_{i=1}^{M}\|c_i^t\|^2 \leq C^2, \qquad \frac{1}{M}\sum_{i=1}^{M}\|\bar{c}_i^t\|^2 \leq C^2. \tag{39}$$

We first characterize the statistical properties of the debiased randomized response estimator.

Lemma 2. *For any coordinate k, the estimator satisfies*

$$\mathbb{E}[\hat{z}_{i,k}^t \mid z_{i,k}^t] = z_{i,k}^t. \tag{40}$$

Consequently,

$$\mathbb{E}[\hat{z}_i^t \mid \mathcal{F}_t] = \operatorname{sign}(\nabla F_i(x_i^t)). \tag{41}$$

Proof. The proof follows directly from the definition of randomized response and linearity of expectation. □

Lemma 3. *The second moment satisfies*

$$\mathbb{E}\|\hat{z}_i^t\|^2 = \frac{d}{(2r-1)^2}. \tag{42}$$

Proof. Since each coordinate takes value $\pm(2r-1)^{-1}$, summing over d coordinates yields the result. □

We next bound the corrected estimator after variance reduction and compression.

Lemma 4. *Under unbiased compression,*

$$\mathbb{E}[y_i^t \mid \mathcal{F}_t] = \operatorname{sign}(\nabla F_i(x_i^t)) - c_i^t + \bar{c}_i^t. \tag{43}$$

Proof. The result follows from the unbiasedness of both randomized response and the compressor. □

Lemma 5. *Define*

$$B^2 = \frac{3d}{(2r-1)^2} + 6C^2. \tag{44}$$

Then

$$\frac{1}{M}\sum_{i=1}^{M}\mathbb{E}\|u_i^t\|^2 \leq B^2, \tag{45}$$

and

$$\frac{1}{M}\sum_{i=1}^{M}\mathbb{E}\|y_i^t\|^2 \leq (1+\beta)B^2. \tag{46}$$

Proof. The result follows from standard norm inequalities and bounded control variates. □

We now establish the main convergence result.

Theorem 2 (Convergence of DeFL-VP)**.** *Under Assumptions 1–7, with stepsize $\eta \le 1/L$, the iterates satisfy*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\bar{x}^t)\|_1 \le \frac{F(\bar{x}^0)-F^\star}{\mu\eta T} + \frac{\Delta_{\text{net}}}{\mu} + \frac{L\eta(1+\beta)B^2}{2\mu}. \tag{47}$$

Proof. By L-smoothness,

$$F(\bar{x}^{t+1}) \le F(\bar{x}^t) + \langle \nabla F(\bar{x}^t), \bar{x}^{t+1}-\bar{x}^t\rangle + \frac{L}{2}\|\bar{x}^{t+1}-\bar{x}^t\|^2. \tag{48}$$

Using the update rule

$$\bar{x}^{t+1} = \bar{x}^t - \eta\cdot\frac{1}{M}\sum_{i=1}^{M} y_i^t, \tag{49}$$

and taking expectation, we obtain

$$\begin{aligned}\mathbb{E}[F(\bar{x}^{t+1})] \le \mathbb{E}[F(\bar{x}^t)] - \eta\mu\mathbb{E}\|\nabla F(\bar{x}^t)\|_1 \\ + \eta\Delta_{\text{net}} + \frac{L\eta^2}{2}(1+\beta)B^2.\end{aligned} \tag{50}$$

Summing over t yields the result. □

Corollary 3. *If $\eta = O(T^{-1/2})$, then*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\bar{x}^t)\|_2 = O\left(\frac{1}{\sqrt{T}}\right) + O(\Delta_{\text{net}}) + O\left(\frac{d}{(2r-1)^2\sqrt{T}}\right). \tag{51}$$

5. Experiments

This section conducts experiments from multiple perspectives to evaluate the performance of DeFL-VP. All experiments were conducted on four A100 GPUs.

5.1. Experimental Setup

We conducted experiments on the Fashion-MNIST dataset. To simulate data heterogeneity, we employed a Dirichlet partition. To achieve stronger data heterogeneity, we set the centralization parameter $\alpha = 0.3$. We configured the decentralized system under DeFL-VP with $N = 10$ clients connected via a peer-to-peer topology. It should be noted that this experimental setup is primarily adopted to analyze the impact of variance reduction, random response-based privacy protection mechanisms, and communication compression modules on model training under uniform decentralized non-IID conditions. Through this controlled setup, the roles of different modules can be more clearly compared under identical conditions.

In addition, each client executes one local epoch per round, with a total of 200 communication rounds. We use the stochastic gradient descent (SGD) algorithm with a learning rate of 0.01 and employ a momentum term to accelerate training, with the momentum coefficient set to 0.9. The privacy level of the experiments depends on the size of the privacy budget. Therefore, we conducted experiments with different privacy budgets, where $\epsilon \in \{0.5, 1, 2, 4, 8\}$. The model performance of FL is evaluated using accuracy, and we also compare the number of training rounds required to achieve the target accuracy. Meanwhile, the security against inference attacks is evaluated using AUC.

5.2. Performance under Decentralized Non-IID Environment

In a decentralized heterogeneous environment, we conducted performance comparison experiments between DeFL-VP and existing methods to validate the effectiveness of DeFL-VP. The results are shown in Figure 3.

As shown in Figure 3 , in a decentralized environment with client heterogeneity, D-PSGD requires more communication rounds than the proposed DeFL-VP to reach the target accuracy of 85%, indicating a slower convergence rate. Furthermore, when D-PSGD incorporates the random response privacy protection mechanism, its convergence rate decreases further, and the final accuracy also declines. Meanwhile, the training results of DFL+Comp. show that relying solely on communication compression techniques is insufficient to effectively improve the model's convergence speed. The results of DFL+VR demonstrate that introducing a variance reduction

mechanism can significantly accelerate the convergence process and mitigate, to some extent, the negative impacts caused by data heterogeneity and random response noise.

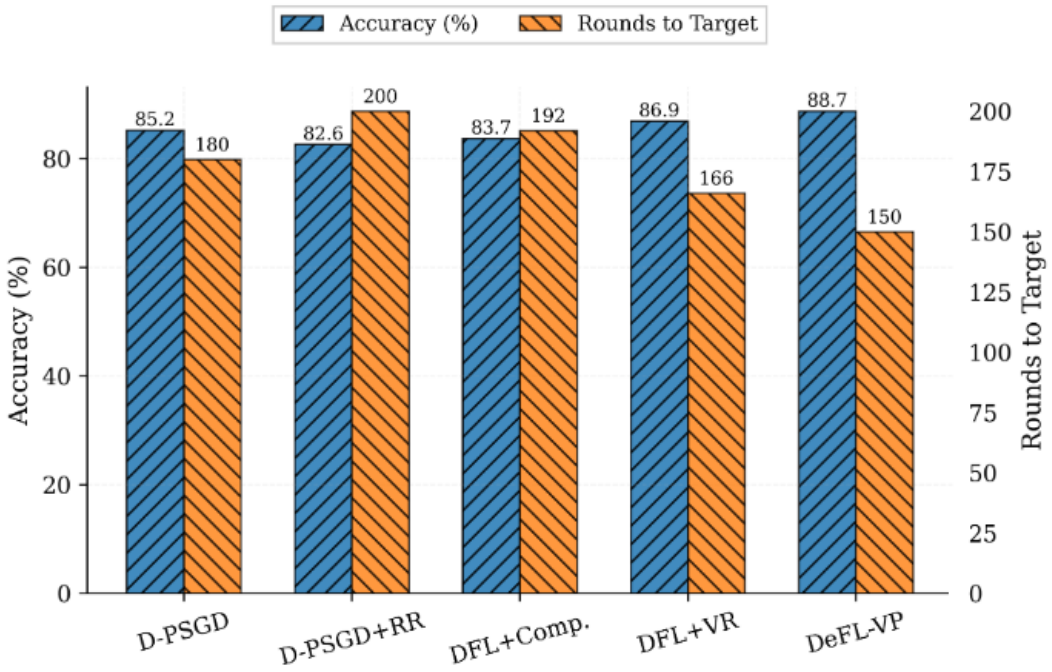


Figure 3. Performance comparison under non-IID decentralized environment.

Table 1 shows the quantitative training results. It can be seen that DeFL-VP achieves faster convergence and higher accuracy, confirming its ability to perform efficient and stable model training in a decentralized heterogeneous environment.

Table 1. Performance comparison under non-IID decentralized environment.

Method	Accuracy (%)	Round to 85%
D-PSGD	85.2	180
D-PSGD+RR	82.6	200
DFL+Compression	83.7	192
DFL+VR	86.9	166
DeFL-VP	88.7	150

5.3. Privacy Protection Performance of DeFL-VP

To evaluate DeFL-VP's privacy protection performance against attacks, we experimentally compared the performance of DeFL-VP, D-PSGD, and D-PSGD+RR in a Member Inference Attack (MIA) scenario. Given that privacy protection mechanisms typically impact data utility, thereby reducing model training accuracy, we comprehensively evaluate the security and utility of different FL frameworks using the attack's ROC curve, as well as the model's accuracy and AUC metrics. The ROC curve results for MIA are shown in Figure 4.

As shown in the Figure 4, compared to D-PSGD, the ROC curves of both D-PSGD+RR and DeFL-VP are closer to the diagonal, indicating that both can effectively reduce the attacker's discriminative power, thereby mitigating the impact of MIA attacks. Moreover, the ROC curve of DeFL-VP lies overall below that of D-PSGD+RR, indicating superior performance in privacy protection. This suggests that introducing a variance reduction mechanism helps mitigate the instability caused by privacy protection mechanisms and reduces the model's overfitting to the training data, thereby further enhancing its resistance to attacks. Furthermore, the quantification results of different methods in the MIA scenario are shown in Table 2.

As shown in the Table 2, DeFL-VP achieves the lowest AUC value while maintaining a high model accuracy. Compared to existing methods, DeFL-VP achieves a lower attack AUC while maintaining high model accuracy, indicating that this method can effectively reduce the risk of privacy leakage without causing significant performance degradation.

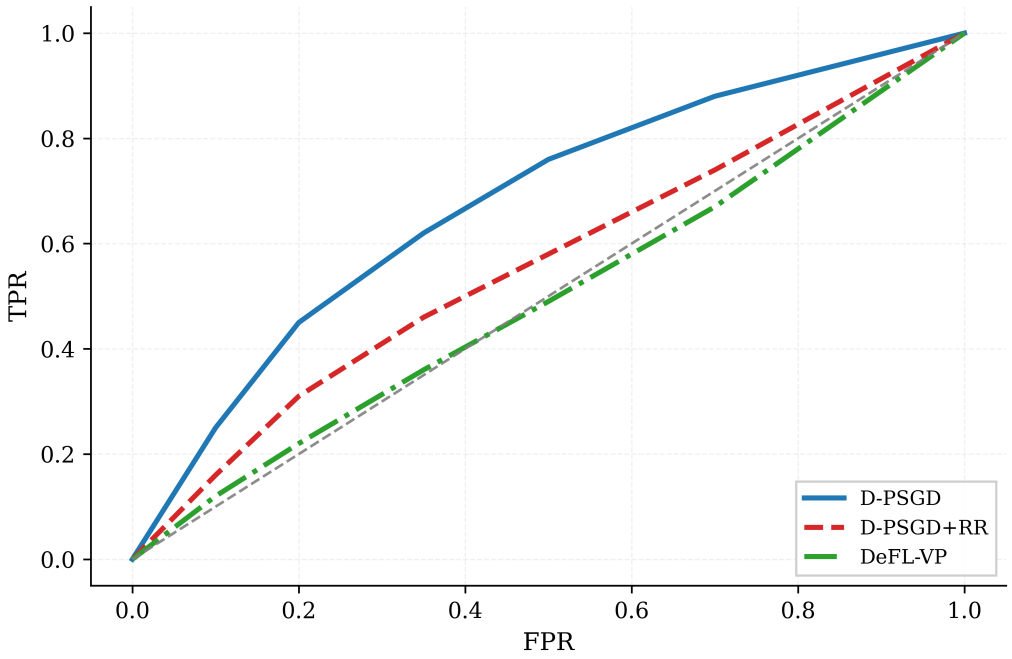


Figure 4. ROC curves of MIA.

Table 2. Performance of Different FL Frameworks Under MIA.

Method	Accuracy (%)	AUC
D-PSGD	85.2	0.677
D-PSGD+RR	82.6	0.560
DeFL-VP	88.7	0.497

5.4. Performance Analysis of DeFL-VP under Different Privacy Budgets

To analyze the impact of the privacy budget ϵ on DeFL-VP performance, we conducted experiments under different privacy budget settings, with the results shown in Figure 5 and Table 3.

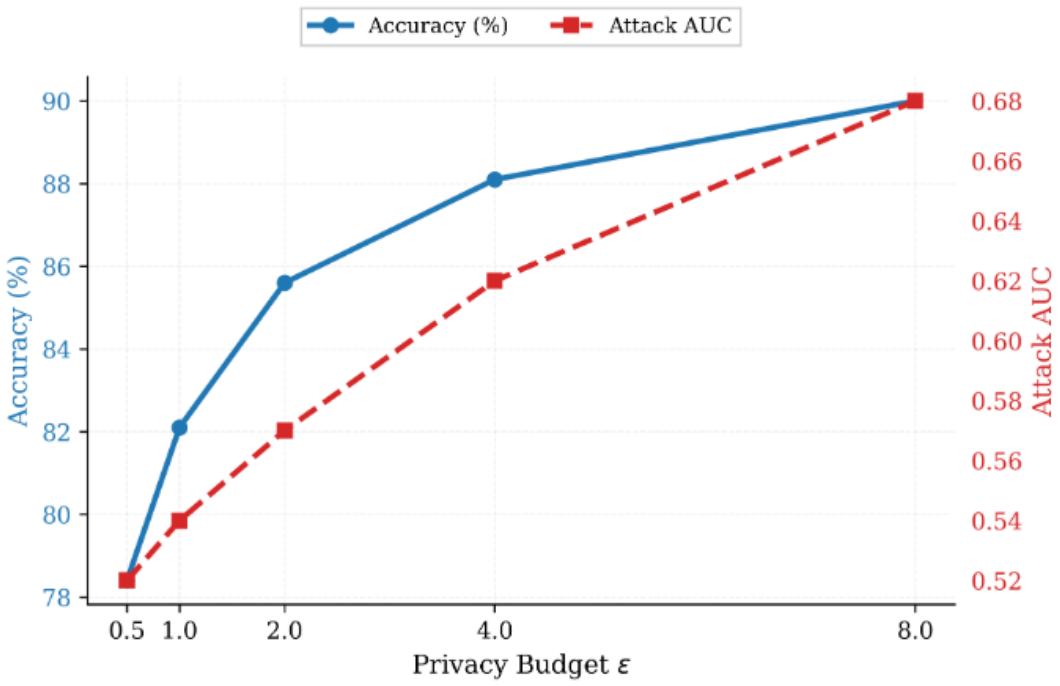


Figure 5. Data Utility Under Different Privacy Budgets.

Table 3. The Effect of the Privacy Budget ϵ on DeFL-VP.

ϵ	Accuracy (%)	Attack AUC	Round to 85%
0.5	78.4	0.52	–
1	82.1	0.54	–
2	85.6	0.57	194
4	88.1	0.62	162
8	90.0	0.68	141

It can be observed that as ϵ increases, the model accuracy of DeFL-VP gradually improves, but the attack AUC also rises accordingly. This indicates that a larger ϵ is beneficial for improving model performance but entails a greater risk of privacy leakage.

It is worth noting that even under small values of ϵ, DeFL-VP still maintains a relatively high accuracy. This indicates that the introduced variance reduction mechanism compensates to some extent for the randomness introduced by privacy perturbations, thereby enabling DeFL-VP to retain good data utility even under strict privacy constraints.

5.5. Performance Analysis of DeFL-VP

To more intuitively evaluate the roles of each key mechanism in DeFL-VP, we conducted ablation experiments. Specifically, by sequentially removing the privacy protection mechanism (DP), the variance reduction mechanism (VR), and the communication compression mechanism (Comp.), we analyzed the impact of each module on overall performance. The experimental results under the MIA scenario are shown in Figure 6 and Table 4.

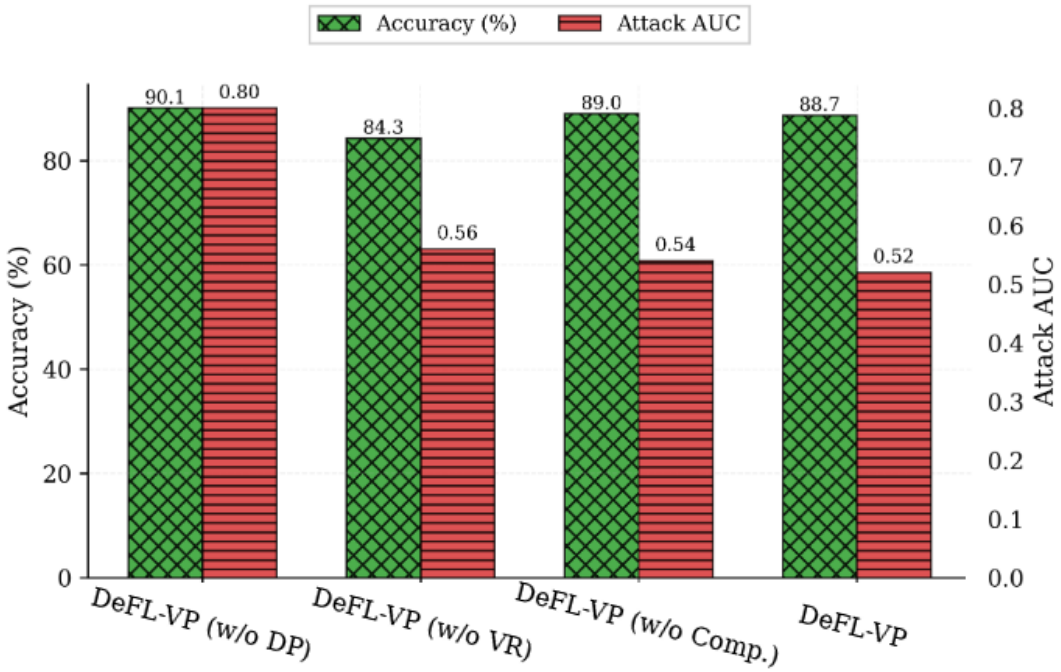


Figure 6. Performance of DeFL-VP under Different Conditions.

Table 4. Performance of DeFL-VP under Different Conditions.

Variant	Accuracy (%)	Attack AUC
DeFL-VP (w/o DP)	90.1	0.80
DeFL-VP (w/o VR)	84.3	0.56
DeFL-VP (w/o Comp.)	89.0	0.54
DeFL-VP	88.7	0.52

As shown in the results of Table 4, the DP mechanism has a significant impact on the attack AUC. When the DP module is removed, the attack AUC increases markedly, indicating that the model is more susceptible to

inference attacks, and thus its privacy protection capability is reduced. In contrast, introducing the DP mechanism effectively reduces the attack success rate, making it a crucial component in ensuring the model's privacy security. The VR mechanism has a more pronounced effect on model accuracy. Removing VR results in a decrease in model accuracy, indicating that client-side biases caused by data heterogeneity can impact the training process. By correcting local gradients, the variance reduction mechanism mitigates update discrepancies across different clients, making the training process more stable.

As for the communication compression module, its impact on the final model accuracy is relatively limited. In the experiments, removing the compression mechanism resulted in a slight improvement in model accuracy, primarily because compression errors no longer existed. However, during actual deployment, frequent exchange of model updates between nodes often imposes a significant communication burden. Although communication compression introduces some error, it reduces the amount of data transmitted.

In addition, we conducted a further analysis of the impact of the communication compression mechanism on performance and communication efficiency. Using total communication time as the evaluation metric, the experimental results are shown in Table 5. It can be observed that, with only a slight decrease in accuracy, the introduction of communication compression significantly reduces the overall communication overhead. Therefore, DeFL-VP demonstrates superior overall performance compared to existing FL systems.

Table 5. Communication efficiency under compression in decentralized FL.

Method	Accuracy (%)	Comm. Time (s)
DFL (No Compression)	86.9	128.0
DFL + Compression	83.7	57.6
DeFL-VP (No Comp.)	89.0	113.6
DeFL-VP	88.7	45.0

6. Conclusions

This paper proposes the DeFL-VP framework to address the trade off between communication efficiency, privacy protection, and convergence performance in decentralized federated learning. In a graph-structured network without a central server, we jointly design communication compression, client variance reduction, and a differential privacy mechanism based on stochastic responses. By introducing local control variables, we effectively mitigate client drift and optimization instability caused by data heterogeneity and privacy protection algorithms. In terms of theoretical analysis, this paper presents convergence results for DeFL-VP under non-convex smoothness conditions and examines the impact of privacy parameters on the algorithm's convergence behavior. It elucidates the relationship between the stochastic response privacy mechanism and the optimization process from a theoretical perspective. Experimental results validate the effectiveness of the proposed method. Compared to baseline methods, DeFL-VP maintains superior convergence performance with lower communication overhead while achieving higher model accuracy. Furthermore, in the MIA scenario, DeFL-VP achieves a lower attack AUC, indicating its ability to reduce the risk of privacy leakage while maintaining model performance. Future research will focus on more flexible privacy protection mechanisms and communication topology designs suitable for large-scale decentralized environments to further enhance the algorithm's applicability and training efficiency in complex scenarios.

Author Contributions

R.X.: conceptualization, methodology, software, validation, formal analysis, visualization, writing—original draft preparation. R.L.: supervision, project administration, methodology, writing—reviewing and editing. W.Z.: data curation, investigation, validation, visualization. X.L.: resources, formal analysis, methodology, writing—reviewing and editing. K.S.: supervision, project administration, resources, writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This work is supported in part by the China Scholarship Council program (Project ID: 202506050068).

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Not applicable.

Conflicts of Interest

The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

During the preparation of this work, the authors used ChatGPT to improve the grammar of the manuscript. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

1. Hallaji, E.; Razavi-Far, R.; Saif, M.; et al. Decentralized federated learning: A survey on security and privacy. *IEEE Trans. Big Data* **2024**, *10*, 194–213.
2. Zhang, Z.; Hu, R. Byzantine-robust federated learning with variance reduction and differential privacy. In Proceedings of the 2023 IEEE Conference on Communications and Network Security (CNS), Orlando, FL, USA, 2–5 October 2023; pp. 1–9.
3. Fu, J.; Hong, Y.; Ling, X.; et al. Differentially private federated learning: A systematic review. *ACM Comput. Surv.* **2026**, *58*, 271.
4. Zhang, C.; Weng, J.; Weng, J.; et al. Robust and secure federated learning with verifiable differential privacy. *IEEE Trans. Dependable Secur. Comput.* **2025**, *22*, 5713–5729.
5. Mashiko, S.; Kawamata, Y.; Nakayama, T.; et al. Anomaly detection in double-entry bookkeeping data by federated learning system with non-model sharing approach. *Sci. Rep.* **2025**, *15*, 42208.
6. Kanamori, S.; Abe, T.; Ito, T.; et al. Privacy-preserving federated learning for detecting fraudulent financial transactions in japanese banks. *J. Inf. Process.* **2022**, *30*, 789–795.
7. Fukami, T.; Murata, T.; Niwa, K.; et al. Dp-norm: Differential privacy primal-dual algorithm for decentralized federated learning. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 5783–5797.
8. Chen, S.; Yu, D.; Zou, Y.; et al. Decentralized wireless federated learning with differential privacy. *IEEE Trans. Ind. Inform.* **2022**, *18*, 6273–6282.
9. Yan, Z.; Wicaksana, J.; Wang, Z.; et al. Variation-aware federated learning with multi-source decentralized medical image data. *IEEE J. Biomed. Health Inform.* **2020**, *25*, 2615–2628.
10. Lu, Y.; Yu, Z.; Suri, N. Privacy-preserving decentralized federated learning over time-varying communication graph. *ACM Trans. Priv. Secur.* **2023**, *26*, 33.
11. Sun, T.; Li, D.; Wang, B. Decentralized federated averaging. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 4289–4301.
12. Xin, R.; Kar, S.; Khan, U.A. Decentralized stochastic optimization and machine learning: A unified variance-reduction framework for robust performance and fast convergence. *IEEE Signal Process. Mag.* **2020**, *37*, 102–113.
13. Zhang, A.; Zhao, P.; Lu, W.; et al. Personalized Decentralized Federated Learning: A Privacy-Enhanced and Byzantine-Resilient Approach. *IEEE Trans. Comput. Soc. Syst.* **2025**, *12*, 3206–3217.
14. Wu, J.; Li, C.; Wu, W.; et al. Client Selection in Federated Learning With Differential Privacy-Based Data Stream. *IEEE Trans. Netw.* **2025**, *34*, 1128–1144.
15. Li, Q.; Wen, Z.; Wu, Z.; et al. A survey on federated learning systems: Vision, hype and reality for data privacy and protection. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 3347–3366.
16. Truex, S.; Baracaldo, N.; Anwar, A.; et al. A hybrid approach to privacy-preserving federated learning. In Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security, London, UK, 15 November 2019; pp. 1–11.
17. Wei, K.; Li, J.; Ding, M.; et al. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3454–3469.
18. Zhao, Y.; Zhao, J.; Yang, M.; et al. Local differential privacy-based federated learning for internet of things. *IEEE Internet Things J.* **2020**, *8*, 8836–8853.
19. Chen, J.; Yan, H.; Liu, Z.; et al. When federated learning meets privacy-preserving computation. *ACM Comput. Surv.* **2024**, *56*, 319.
20. Hidayat, M.A.; Nakamura, Y.; Arakawa, Y. Privacy-preserving federated learning with resource-adaptive compression for edge devices. *IEEE Internet Things J.* **2023**, *11*, 13180–13198.

21. Dinh, C.T.; Tran, N.H.; Nguyen, T.D.; et al. Federated learning with proximal stochastic variance reduced gradient algorithms. In Proceedings of the 49th International Conference on Parallel Processing, Edmonton, AB, Canada, 17–20 August 2020; pp. 1–11.
22. Zhang, H.; Li, C.; Dai, W.; et al. Federated learning based on model discrepancy and variance reduction. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 10407–10421.
23. Wang, B.; Fang, J.; Li, H.; et al. Communication-efficient federated learning: A variance-reduced stochastic approach with adaptive sparsification. *IEEE Trans. Signal Process.* **2023**, *71*, 3562–3576.
24. Guo, Z.; Yu, K.; Lv, Z.; et al. Deep federated learning enhanced secure POI microservices for cyber-physical systems. *IEEE Wirel. Commun.* **2022**, *29*, 22–29.
25. Taya, A.; Nishio, T.; Morikura, M.; et al. Decentralized and model-free federated learning: Consensus-based distillation in function space. *IEEE Trans. Signal Inf. Process. Over Netw.* **2022**, *8*, 799–814.
26. Yue, G.; Yan, L.; Kang, L.; et al. Adapldp-fl: An adaptive local differential privacy for federated learning. *IEEE Trans. Mob. Comput.* **2025**, *24*, 5569–5583.
27. Zhao, J.; Zhu, H.; Wang, F.; et al. PVD-FL: A privacy-preserving and verifiable decentralized federated learning framework. *IEEE Trans. Inf. Forensics Secur.* **2022**, *17*, 2059–2073.
28. Xiu, R.; Le, J.; Zhang, D.; et al. Neuron Membership Inference Attack in Federated Learning for bit Scenarios. *IEEE Trans. Emerg. Top. Comput. Intell.* **2026**, *10*, 1849–1862.
29. Lian, Z.; Yang, Q.; Wang, W.; et al. DEEP-FEL: Decentralized, efficient and privacy-enhanced federated edge learning for healthcare cyber physical systems. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3558–3569.
30. Li, H.; Lü, Q.; Chen, G.; et al. Distributed Constrained Optimization Over Unbalanced Directed Networks Using Asynchronous Broadcast-Based Algorithm. *IEEE Trans. Autom. Control* **2021**, *66*, 1102–1115.
31. Li, Z.; Zeng, X.; Xiao, Y.; et al. Pattern-sensitive local differential privacy for finite-range time-series data in mobile crowdsensing. *IEEE Trans. Mob. Comput.* **2024**, *24*, 1–14.
32. Lian, Z.; Zeng, Q.; Wang, W.; et al. Blockchain-based two-stage federated learning with non-IID data in IoMT system. *IEEE Trans. Comput. Soc. Syst.* **2022**, *10*, 1701–1710.
33. Li, B.; Schmidt, M.N.; Alstrøm, T.S.; et al. On the effectiveness of partial variance reduction in federated learning with heterogeneous data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 3964–3973.
34. Malinovsky, G.; Yi, K.; Richtárik, P. Variance reduced proxskip: Algorithm, theory and application to federated learning. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 15176–15189.
35. Huang, Z.; Huang, W.; Jawanpuria, P.; et al. Federated learning on Riemannian manifolds with differential privacy. *Mach. Learn.* **2026**, *115*, 84.
36. Xu, S.; Hua, Z.; Zheng, Y. Federated Learning in the Shuffle Model of Differential Privacy: A Communication-Efficient and Maliciously Secure Realization. *IEEE Trans. Dependable Secur. Comput.* **2026**, *23*, 6311–6329.
37. Asad, M.; Shaukat, S.; Javanmardi, E.; et al. Secure and efficient blockchain-based federated learning approach for VANETs. *IEEE Internet Things J.* **2023**, *11*, 9047–9055.
38. Lian, Z.; Wang, W.; Han, Z.; et al. Blockchain-based personalized federated learning for internet of medical things. *IEEE Trans. Sustain. Comput.* **2023**, *8*, 694–702.
39. Nowroozi, E.; Haider, I.; Taheri, R.; et al. Federated Learning Under Attack: Exposing Vulnerabilities through Data Poisoning Attacks in Computer Networks. *IEEE Trans. Netw. Serv. Manag.* **2025**, *22*, 822–831.
40. Yang, L.; Miao, Y.; Liu, Z.; et al. Enhanced model poisoning attack and multi-strategy defense in federated learning. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 3877–3892.