applied sciences

MDPI

Article

BE-DPFL: A Blockchain-Enhanced Privacy-Preserving Federated Learning Framework for Secure Edge Network Collaboration

Wangjing Jia [1,2] and Tao Xie [1,*]

[1] College of Computer Science and Technology, National University of Defense Technology, Changsha 410073, China; 250017@sxpc.edu.cn
[2] Shanxi Police College, Taiyuan 030401, China
* Correspondence: 13393437609@163.com

Abstract

Against the deep integration of digital transformation and AI, cross-institutional collaborative modeling hinges on efficient data circulation, yet data silos and privacy regulations hinder traditional centralized training. Federated Learning (FL) keeps data local but faces issues like weak centralized trust, inadequate privacy protection, and poor robustness in edge networks. Existing improvements, including via differential privacy (DP) and blockchain, among others, still suffer from centralized budget allocation, low consensus efficiency, or single-point-of-failure addressing, failing to jointly optimize trust, performance, and privacy. The limitations are exacerbated in high-frequency, resource-constrained edge environments. To tackle these challenges, this paper proposes BE-DPFL, a blockchain-enhanced differentially private FL framework that integrates on-chain trusted supervision and off-chain efficient training. It builds a lightweight blockchain trust layer with FL-PBFT consensus and smart contracts, introduces Random Projection–ADMM optimization, and designs a multi-objective adaptive gradient clipping/noise injection strategy. Experiments on CIFAR-10 and ChestX-ray14 demonstrate that BE-DPFL outperforms mainstream methods in consensus efficiency, communication overhead, privacy-accuracy balance, and robustness. It reduces communication costs by over 97%, achieves 100% privacy compliance, and maintains stable performance even under high disturbances. Ablation studies confirm the significant contributions of core components.

Keywords: federated learning; differential privacy; blockchain; edge intelligence

Check for updates

Academic Editor: Gianluca Lax

Received: 12 January 2026
Revised: 5 February 2026
Accepted: 6 February 2026
Published: 11 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

In the context of deep integration between digital transformation and artificial intelligence, data has become an increasingly critical strategic production factor. However, the contradiction between pervasive "data silos" and stringent privacy regulations (e.g., General Data Protection Regulation (GDPR) and China's Personal Information Protection Law) significantly restricts cross-institution and cross-domain collaborative modeling. Cross-institutional collaborative modeling refers to cooperation between independent entities (e.g., hospitals, enterprises), while cross-domain collaboration involves integrating data from heterogeneous fields (e.g., medical imaging + electronic health records). Federated Learning (FL) [1] enables both scenarios by keeping data local. This approach has demonstrated strong potential in domains such as medical image diagnosis [2], financial risk control [3], and industrial IoT [4]. But trust and privacy issues are amplified in cross-institutional settings (due to conflicting interests) and cross-domain settings (due

to data heterogeneity). BE-DPFL addresses these amplified challenges via decentralized supervision and adaptive privacy strategies.

Despite its advantages, traditional FL still faces three fundamental challenges in practice. First, trust deficiency: existing FL architectures rely heavily on centralized servers for parameter aggregation and privacy budget management. If the server is compromised or malicious, risks such as gradient leakage and parameter tampering may arise. Moreover, clients lack mechanisms to verify whether the server correctly enforces privacy requirements, or whether participating clients inject noise as required or submit malicious updates. Second, insufficient privacy protection: while FL prevents raw data transmission, gradients still contain sensitive information and can be exploited by gradient inversion attacks to reconstruct private data. Fixed privacy strategies further struggle to accommodate varying data sensitivities and training demands across clients, often leading to either "over-protection" or "under-protection." Third, limited system robustness: in edge environments characterized by unstable networks, latency, and packet loss, training can be interrupted, while malicious clients may hinder convergence. Traditional FL protocols [5] generally lack effective mechanisms for anomaly detection, behavior auditing, and fault tolerance, making it difficult to ensure stable training.

To address these challenges, prior research has explored several directions. In privacy enhancement, differential privacy (DP)-based FL methods [6,7] attempt to localize privacy protection or adaptively adjust noise levels to improve model performance. However, these approaches still rely on centralized privacy budget allocation and lack transparent, verifiable records, leaving trust risks unresolved and failing to ensure clients enforce the agreed-upon privacy strategies. In trust construction, blockchain [8] has been incorporated into FL due to its decentralization and immutability, enabling auditable logging of parameter updates and client behaviors. Nevertheless, traditional consensus protocols [9] incur high communication overhead and latency, making them unsuitable for FL's high-frequency iterative nature. Directly storing high-dimensional parameters on-chain further increases bandwidth and storage costs, limiting applicability in resource-constrained edge scenarios. In robustness optimization, existing work [10] often addresses isolated problems—such as malicious client detection or network fluctuation compensation—without offering integrated solutions for multi-disturbance environments. Additionally, dynamically balancing privacy compliance and learning performance remains a critical yet unresolved challenge.

To overcome these limitations, this paper proposes BE-DPFL, a Blockchain-Enhanced Differentially Private FL framework that combines on-chain trusted supervision with off-chain efficient training to jointly optimize trust, privacy, and performance. The framework incorporates three main innovations. First, it builds a lightweight blockchain trust layer featuring the Federated Learning-adapted Practical Byzantine Fault Tolerance (FL-PBFT) consensus mechanism and modular smart contracts tailored for FL, enabling fully auditable privacy budget management, update verification, and client behavior monitoring. A phased consensus strategy—"full consensus for key information and lightweight validation for regular information"—eliminates the centralized trust bottleneck while significantly reducing latency. Second, it introduces a Random Projection–Alternating Direction Method of Multipliers (ADMM) collaborative optimization mechanism, enabling the training pipeline "high-dimensional gradient $\rightarrow$ random projection $\rightarrow$ off-chain ADMM aggregation $\rightarrow$ on-chain verification," which greatly cuts communication and storage costs and addresses performance bottlenecks in blockchain-based FL. Third, it designs a Multi-Objective Optimization (MOO)-based adaptive gradient clipping and noise injection strategy that simultaneously satisfies (ε, δ)-DP constraints and maintains model performance. By adjusting objective weights across training phases, the method achieves dynamic privacy–accuracy balancing and overcomes the poor adaptability of fixed privacy schemes. Experimental

results on CIFAR-10 and ChestX-ray14 demonstrate that the proposed BE-DPFL framework achieves outstanding performance in terms of consensus efficiency, communication overhead, and the privacy-accuracy trade-off, significantly outperforming mainstream comparative methods. It effectively mitigates the performance loss induced by privacy preservation while exhibiting strong robustness against malicious behaviors and network fluctuations—collectively verifying its effectiveness in trustworthy collaborative modeling scenarios. Ablation studies further confirm that the three core innovations (lightweight blockchain trust layer, Random Projection–ADMM optimization, and MOO-based adaptive gradient clipping) are not redundant; instead, they synergistically reinforce the framework's comprehensive performance by addressing trust, efficiency, and privacy-accuracy balance, respectively.

The major contributions of this work are summarized as follows:

- We design the FL-PBFT consensus and modular smart contracts to provide fully verifiable privacy budget allocation, parameter update validation, and behavior auditing. The phased consensus strategy achieves an effective balance between trustworthiness and training efficiency, addressing the long-standing challenge of simultaneously ensuring reliability and scalability in blockchain-based FL.
- By integrating random projection for dimensionality reduction with ADMM for distributed optimization, we propose a complete pipeline: local clipping → low-dimensional projection → off-chain aggregation → on-chain verification. This significantly improves throughput to 5.38–5.63 rounds per minute, making blockchain-based FL feasible in resource-limited edge settings.
- Leveraging MOO theory, we design dynamic gradient clipping and noise injection mechanisms that adjust the weighting between privacy and accuracy over different training stages. This approach effectively mitigates the limitations of fixed privacy strategies and improves the privacy–accuracy trade-off, particularly benefiting high-sensitivity domains such as healthcare.

The remainder of this paper is organized as follows. Section 2 reviews related work on DP in federated learning and blockchain-enhanced differentially private FL, and identifies the research gaps this study aims to address. Section 3 details the proposed BE-DPFL framework, including its three-layer architecture, the blockchain trust layer (FL-PBFT consensus and modular smart contracts), the Random Projection–ADMM optimization mechanism, the multi-objective adaptive gradient clipping strategy, and the overall workflow. Section 4 presents experimental evaluation and analysis from three perspectives—consensus performance, privacy–accuracy trade-offs, and robustness against malicious nodes and network fluctuations—through comprehensive comparisons with mainstream baseline methods. Section 5 concludes the study by summarizing the main findings and outlining future research directions to further enhance the applicability and performance of the framework.

2. Related Work

2.1. Differential Privacy in Federated Learning

With the growing demand for data security and privacy protection, DP, as a rigorous mathematical privacy framework, has become one of the most widely applied privacy-enhancing techniques in FL. Current research mainly focuses on global DP, local DP, personalized and adaptive DP mechanisms, and DP strategies tailored for multimodal and heterogeneous environments. Overall, these studies have advanced the balance between privacy protection and model performance. Yet, significant limitations remain in trustworthy collaboration, privacy-budget auditing, participant behavior regulation, and transparency of the update process—challenges that are particularly critical in cross-institutional, multi-party, and heterogeneous settings.

Early studies primarily focused on directly injecting DP noise into FL to mitigate the risk of gradient leakage. Wei et al. systematically analyzed the algorithmic performance and convergence characteristics of DP-FL under different noise budgets, laying the foundation for its application in FL [11]. However, such methods typically rely on a central server to allocate privacy budgets and aggregate gradients, creating a potential single point of trust. If the server is compromised or behaves maliciously, the DP guarantees may fail. To reduce this trust assumption, Truex et al. proposed LDP-Fed, shifting DP protection to the user-side local computation, thereby eliminating reliance on a trusted server [12]. Although local perturbation strengthens privacy guarantees, it introduces substantial noise that degrades model accuracy, and current methods cannot verify whether clients genuinely follow the protocol for injecting noise, thereby lacking auditability.

As applications diversify, personalized DP has gradually become a research hotspot. Hu et al. proposed allocating DP budgets adaptively based on the sensitivity of user data and individual protection requirements, enabling more fine-grained privacy control [13]. Subsequently, Wu et al. [14] and Yang et al. [15] developed adaptive DP methods that dynamically adjust the noise scale based on training states or user-specific conditions. These mechanisms improve training efficiency and model accuracy; however, the scheduling, allocation, and consumption of the privacy budget are managed by a central server, lacking transparent, verifiable records. Participants cannot verify that the server implements the privacy mechanism as specified, creating a new trust bottleneck. Furthermore, it remains impossible to verify whether clients actually apply personalized or adaptive noise injection, leaving room for malicious evasion or poisoning.

To meet evolving application needs, DP-FL has expanded further into heterogeneous environments and multimodal scenarios. Ling et al. proposed heterogeneous DP mechanisms to support clients with different sensitivity levels and data distributions [16]; Liu et al. [17] applied personalized DP to psychological-assessment modeling to achieve more fine-grained data protection; Ma et al. [18] integrated DP into multimodal FL and combined it with modality selection to reduce noise-induced performance degradation. Although these studies significantly enhance DP-FL's adaptability, they still lack robust cross-party supervision mechanisms. Privacy-budget usage is not traceable, model updates cannot be independently verified, and the system cannot track the origin of malicious or non-compliant gradient uploads—leaving accountability and traceability unaddressed.

Meanwhile, some studies investigate the capability and boundaries of DP in resisting gradient-leakage attacks. Hu et al. demonstrated through empirical analysis that DP cannot fully defend against high-precision gradient-reconstruction attacks during iterative training, especially when noise budgets are insufficient or training rounds are large [19]. In bandwidth-limited scenarios, Gad et al. [20] combined knowledge distillation and DP to reduce communication cost, yet their method still relies on trusted server-side update management. These findings indicate that although DP-FL mitigates leakage risk, it struggles to provide process-level verifiable security guarantees.

In summary, while existing DP-FL research has achieved substantial progress in privacy-mechanism design, performance optimization, and scenario expansion, most methods remain dependent on centralized architectures and lack trustworthy auditing and traceability for privacy budgets, noise injection, and model updates. They also lack mechanisms for regulating participant behavior, making DP-FL still face trust deficiencies in high-sensitivity or cross-institutional collaboration scenarios. Blockchain technology, with its decentralization, immutability, traceability, and programmable smart-contract execution, offers inherent advantages for enabling trustworthy privacy-budget management, verifiable noise-injection behavior, update provenance tracking, and automated governance

of abnormal behaviors. Therefore, integrating blockchain into differential privacy FL is a crucial direction for building more trustworthy, auditable, and secure collaborative systems.

2.2. Blockchain-Driven Differentially Private Federated Learning

As the demand for large-scale deployment of FL in industrial IoT, healthcare, and IoT scenarios continues to grow, blockchain—owing to its decentralization, traceability, and tamper-resistance—has been widely regarded as a feasible solution to issues in traditional FL, such as "untrusted central servers, easily tampered parameters, and difficulty in auditing participant behaviors." Consequently, an increasing number of studies have begun exploring the integration of blockchain with differentially private FL (DP-FL) to construct a trustworthy distributed training framework.

Lu et al. [21] were among the first to propose using blockchain to enhance the trustworthiness of data sharing in industrial IoT scenarios, enabling auditable participant behavior and transparent parameter exchange by recording model updates on-chain. However, this method does not incorporate DP mechanisms and therefore cannot defend against gradient inference attacks or deep reconstruction attacks. To strengthen privacy protection, Jia et al. [22] further combined DP with homomorphic encryption and utilized blockchain to verify and aggregate model updates, addressing the problem of untrusted multi-party collaboration in industrial Internet settings. Although this approach offers strong security guarantees, it relies on consortium-chain consensus mechanisms (e.g., PBFT), requiring multiple rounds of node communication for each parameter update, which significantly increases training latency.

In data-sharing and healthcare applications involving highly sensitive data, Chen et al. [23] and Javed et al. [24] proposed using blockchain to ensure the immutability of data flow while incorporating DP during FL to defend against gradient inversion attacks. These approaches typically rely on smart contracts to manage model uploads and incentive rules, offering transparency, trustworthiness, and regulatory capabilities. However, unavoidable factors such as block generation time and node synchronization limit on-chain write frequency, making it difficult for such systems to support the high-frequency model updates required in FL, which may involve hundreds or even thousands of iterative rounds.

In more complex scenarios such as UAV networks, disaster response, and edge IoT, Pauu et al. [25] and Hussain et al. [26] explored blockchain-enabled local DP FL mechanisms. Although blockchain improves collaboration trustworthiness and reduces server-side attack surfaces, the latency introduced by consensus becomes the core bottleneck limiting system scalability. If public-chain consensus such as proof of work (PoW) is used, the TPS level is insufficient to support frequent uploads of gradients or model updates; even in consortium chains using PBFT, communication complexity increases exponentially with the number of nodes, further extending end-to-end training time and preventing the model from converging as expected.

More recent research, such as that by De Chaudhury et al. [27], introduces incentive mechanisms with local DP, leveraging blockchain to ensure transparent incentives and trustworthy model updates, thereby enhancing system usability. However, similar to other studies, this line of work still depends on frequent on-chain interactions, and when on-chain throughput becomes insufficient, training efficiency drops significantly. Moreover, after incorporating DP noise, gradient signals are further weakened; if blockchain-induced latency forces the system to reduce the number of iterations or shrink the learning rate, the resulting accuracy loss compounds over time, creating a dual degradation effect of "blockchain latency $\times$ DP noise".

In summary, existing blockchain-based DP-FL solutions effectively address issues such as parameter tampering, untrusted participants, and non-traceable behaviors, but they still face two critical challenges: (1) training performance bottlenecks caused by consensus mechanisms, which make it difficult to meet FL's requirement for high-frequency model iteration; and (2) accuracy degradation introduced by DP noise, which becomes further amplified under high-latency blockchain architectures. Therefore, reducing on-chain interaction frequency, designing lightweight consensus mechanisms, developing hybrid on-chain/off-chain update frameworks, and creating low-noise, high-efficiency DP strategies will be key research directions for making blockchain-enabled DP-FL practically deployable in the future.

3. Methodology

3.1. Overall Framework Design

This paper proposes a Blockchain-Enhanced Differentially Private Federated Learning framework (BE-DPFL), which leverages blockchain as the trusted collaborative infrastructure and organically integrates the random projection–ADMM optimization and multi-objective adaptive gradient clipping mechanism to construct a hybrid architecture of on-chain trusted supervision + off-chain efficient training. It aims to simultaneously address the key issues existing in traditional federated learning, including insufficient collaborative trustworthiness, performance bottlenecks caused by blockchain consensus, and accuracy degradation induced by DP. The overall framework adopts a hierarchical collaborative design comprising the Blockchain Trust Layer, the Federated Learning Core Layer, and the Privacy Enhancement Layer. Through cross-layer collaboration, it achieves trusted, efficient, and high-precision distributed model training, as illustrated in Figure 1.

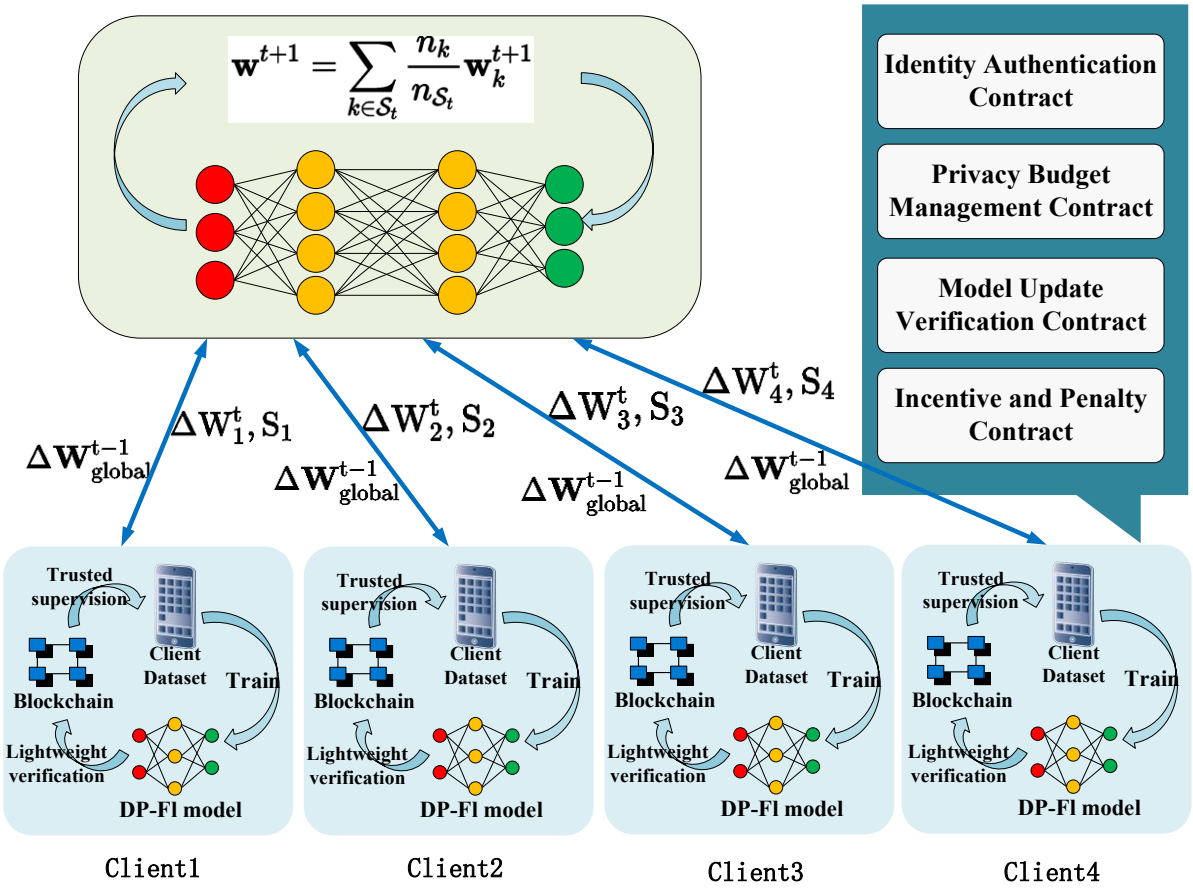

Figure 1. Architecture Diagram of BE-DPFL Framework.

As the foundation of the system, the Blockchain Trust Layer adopts a lightweight consortium blockchain architecture [28], providing decentralized identity management, privacy budget allocation, behavior-based training auditing, and result authentication capabilities for the federated learning process. Based on the customized FL-PBFT consensus mechanism, this layer ensures strong consistency of key training information while reducing consensus latency and communication overhead, thereby adapting to the high-frequency, iterative nature of federated learning. The supporting smart contracts for identity management, privacy budgeting, parameter verification, and incentive-punishment mechanisms enable the whole training process to be traceable, auditable, and accountable, eliminating the reliance on centralized trusted third parties.

Focusing on alleviating the performance bottlenecks introduced by blockchain, the Federated Learning Core Layer proposes a Random-Projection-ADMM distributed optimization mechanism. This mechanism reduces the dimensionality of locally updated parameters via random projection and transmits only low-dimensional projections, thereby reducing communication and storage overhead at the source. Subsequently, it uses ADMM to optimize distributed consistency, ensuring global convergence while reducing reliance on synchronous interaction. The random projection matrix is generated from random seeds synchronized via blockchain, and the hash values of key parameters, as well as projection information, are stored on-chain for verification, ensuring the verifiability and consistency of the training process. Theoretical and experimental analyses demonstrate that this mechanism significantly improves overall training efficiency while preserving ADMM's convergence properties.

Targeting the accuracy loss caused by DP [29], the Privacy Enhancement Layer designs an MOO-based adaptive gradient clipping mechanism. Under the constraint of (ε, δ)-DP, this mechanism takes "privacy compliance" and "maximization of model accuracy" as joint optimization objectives to dynamically adjust the gradient clipping norm and noise injection intensity. By prioritizing model convergence in the early stage of training and gradually strengthening privacy protection in the later stage, it realizes a refined trade-off between privacy intensity and model performance. The clipping parameters and noise configurations generated in each training round are synchronously recorded on the blockchain for privacy budget compliance verification, forming a closed loop with the Blockchain Trust Layer and the Federated Learning Core Layer.

Driven by the collaboration of the above three-layer mechanisms, BE-DPFL constructs a complete closed-loop process covering initialization, privacy-enhanced local training, low-dimensional parameter uploading, compliance verification, and global aggregation, as well as model updating and convergence authentication. It ultimately realizes the unified optimization of blockchain trustworthiness, federated learning efficiency, and DP-preserving accuracy.

3.2. Technical Details of the Blockchain Trust Layer

As the "trust cornerstone" of the BE-DPFL framework, the Blockchain Trust Layer not only needs to adapt to the high-frequency iteration and low-latency requirements of FL but also provides core functions, such as identity authentication, behavior supervision, and data certification. To this end, this section designs a consortium chain architecture optimized for FL scenarios, which includes a lightweight consensus mechanism and a modular smart contract system to meet the dual requirements of trustworthiness and efficiency.

3.2.1. Consortium Chain Network Topology

The consortium chain network adopts a three-level network structure of "validation nodes-participant nodes-aggregation nodes", where different types of nodes perform their

respective duties to collaboratively ensure the stable operation of the network. Validation nodes are composed of industry authorities and regulators, for example, the health commission in medical scenarios and the regulatory bureau in financial scenarios. They are mainly responsible for reaching consensus, verifying transactions, and holding malicious nodes accountable. To control consensus overhead, the number of validation nodes is usually limited to 5–10. Participant nodes are FL clients, such as hospitals, enterprises, and edge devices, which are mainly responsible for local model training and privacy-enhancing processing (including gradient clipping and random projection). During the entire process, only key information (e.g., parameter hashes, privacy budget consumption) is uploaded to the blockchain to reduce unnecessary data transmission. Aggregation nodes are served by trusted third parties or rotating validation nodes, which are responsible for receiving low-dimensional projection parameters uploaded by participant nodes, executing ADMM global update operations [30], and synchronizing the hashes of aggregation results and training logs to the blockchain, ensuring the continuity and traceability of the global training process.

3.2.2. Lightweight Consensus Mechanism: FL-PBFT, a Practical Byzantine Fault Tolerance (PBFT) Variant Adapted for Federated Learning

To address the problems of high latency and large communication overhead in the traditional PBFT consensus mechanism [31], this study designs a PBFT variant for FL (FL-PBFT) that is better suited to FL requirements through multi-faceted optimizations. First, a phased consensus strategy is adopted, dividing the consensus process into two categories: "key information consensus" and "ordinary information certification". For key information, such as privacy budget allocation, clipping norm thresholds, and global model hashes, the complete PBFT consensus process, i.e., the three phases of proposal-prepare-commit, is adopted to ensure the tamper-proof nature of this core information. For ordinary information, such as local parameter hashes and training round logs, a simplified consensus process is adopted that requires confirmation by more than 50% of the validation nodes before being uploaded to the chain, effectively reducing consensus latency. Secondly, batch processing optimization is implemented, where ordinary information generated by multiple rounds of FL iterations is packaged and uploaded to the chain in batches. Typically, a block is generated every 10–20 iterations, reducing the number of blocks and consensus rounds and further improving consensus efficiency. Finally, a dynamic view change mechanism is introduced. When an aggregation node fails, validation nodes can quickly elect a new aggregation node according to predefined rules, and the view change time is kept within 100 ms to ensure the continuity of the training process.

The consensus latency of FL-PBFT can be calculated by the following formula:

$$T_{\text{consensus}} = T_{\text{prop}} + T_{\text{prepare}} + T_{\text{commit}} \quad (1)$$

where T_{prop} is the proposal propagation time with $T_{\text{prop}} \leq 20$ ms; T_{prepare} is the confirmation time of the preparation phase with $T_{\text{prepare}} \leq 30$ ms; T_{commit} is the confirmation time of the commit phase with $T_{\text{commit}} \leq 30$ ms. Through calculation, the total latency of single-round key information consensus is $\leq$80 ms, which can fully meet the high-frequency iteration requirements of FL.

The communication complexity of FL-PBFT is $O(N)$ for ordinary information and $O(N^2)$ for key information (same as standard PBFT), but the phased consensus reduces the number of $O(N^2)$ operations by 90% (only triggered for 5–10% of key data). Computational complexity is reduced by 40% due to batch processing of ordinary information. Compared to standard PBFT, FL-PBFT achieves a 63.7% reduction in consensus latency (verified in ablation studies) while maintaining the same fault tolerance (up to 1/3 malicious nodes).

3.2.3. Modular Smart Contract Design

To realize the automation and trustworthiness of the entire FL process, four core smart contracts are designed based on Solidity 0.8.23 and deployed on the Hyperledger Fabric 2.5 consortium chain platform, each with clear functions and mutual coordination, forming a complete smart contract system. The Identity Contract is mainly responsible for the registration, qualification verification, and public–private key management of participant nodes. Its core logic is as follows: participant nodes submit corresponding qualification certificates, such as medical institution licenses and enterprise business licenses; validation nodes conduct voting audits on the qualification certificates; after passing the audit, a unique on-chain identity identifier (NodeID) is assigned to the participant node and bound to its public key for subsequent encrypted parameter transmission, ensuring the legitimacy and uniqueness of node identities.

The Privacy Budget Contract functions include initializing the global privacy budget ϵ_{total} , dynamically allocating the local budget ϵ_i for clients, and recording budget consumption logs. In privacy budget allocation, the distribution is determined based on the data scale and sensitivity of participant nodes, with the core formula as follows:

$$\epsilon_i = \epsilon_{\text{total}} \cdot \frac{\sqrt{n_i} \cdot s_i}{\sum_{j=1}^{K} \sqrt{n_j} \cdot s_j} \tag{2}$$

where n_i represents the local data volume of client i, s_i is the data sensitivity coefficient, ranging from 0.1 to 1.0, which is determined by the audit of validation nodes, and K is the total number of participant nodes. This formula enables reasonable and fair allocation of privacy budgets.

The Update Verify Contract is mainly used to verify the integrity of parameter hashes uploaded by participant nodes and the compliance of privacy-enhancing operations, such as judging whether gradient clipping meets the threshold requirements. Its core logic is as follows: participant nodes upload the parameter hash $H(z_i^t)$, clipping norm C_i^t, and projection matrix seed, $seed_i^t$. The contract performs verification through the following equation:

$$H(z_i^t) \equiv H(A_i^t \cdot \text{clip}(g_i^t, C_i^t)) \tag{3}$$

where z_i^T is the projected low-dimensional parameter, A_i^T is the random projection matrix, and (g_i^T, C_i^T) is the gradient clipping result. After passing the verification, relevant information is recorded on the blockchain to ensure the compliance and trustworthiness of the model update process.

The Incentive Penalty Contract serves to allocate incentives based on node contributions and hold malicious nodes accountable (e.g., nodes uploading false parameters or circumventing privacy mechanisms). The calculation of node contribution is realized through the following formula:

$$\text{Contribution}_i = \alpha \cdot \text{Accuracy}_i + \beta \cdot \text{Privacy Compliance}_i - \gamma \cdot \text{Delay}_i \tag{4}$$

where Accuracy_i is the local model accuracy contribution, $\text{Privacy Compliance}_i$ is the privacy mechanism compliance score, Delay_i is the parameter upload delay, and α, β, γ are weight coefficients determined through negotiation by validation nodes. This formula enables an objective and reasonable evaluation of node contributions, thereby realizing effective incentives and accountability.

3.3. Random-Projection-ADMM Optimization Mechanism

To address the training performance bottleneck caused by blockchain consensus latency, this section organically combines random projection technology with the ADMM optimization framework and designs a distributed training mechanism adapted to the BE-DPFL architecture. It significantly reduces communication overhead while effectively ensuring model consensus efficiency, providing technical support for the efficient operation of the framework.

3.3.1. Core Design Concept

In traditional FL, clients need to upload complete high-dimensional model parameters [32]. Taking the VGG16 model as an example, its parameter scale can reach 138 M, which undoubtedly leads to huge communication overhead and high on-chain storage pressure. In contrast, this mechanism adopts a three-step process of "local gradient clipping-random projection dimensionality reduction-ADMM distributed consensus", transmitting only the reduced-dimension low-dimensional projection parameters, thus greatly reducing data transmission volume. Meanwhile, it takes full advantage of the parallel optimization characteristics of the ADMM algorithm to reduce the number of interactions between clients and aggregation nodes, making it better adapted to the low-latency requirements of blockchain and effectively improving the overall performance of distributed training.

3.3.2. Complete Implementation Process

Figure 2 illustrates the core pipeline of the Random-Projection-ADMM mechanism: each client updates the local model first, then compresses the parameter via random projection, and finally submits the compressed vector to the aggregation node for global consensus, repeating until convergence. In the step of local model training and gradient clipping, client i conducts model training based on the local dataset D_i and calculates the gradient $g_i^t = \nabla f_i(w_i^t)$. Then, the gradient is processed through the MOO adaptive gradient clipping mechanism introduced in Section 3.4 to obtain $\mathrm{clip}(g_i^t, C_i^t)$, thereby ensuring that the gradient satisfies DP constraints and laying the foundation for subsequent privacy protection.

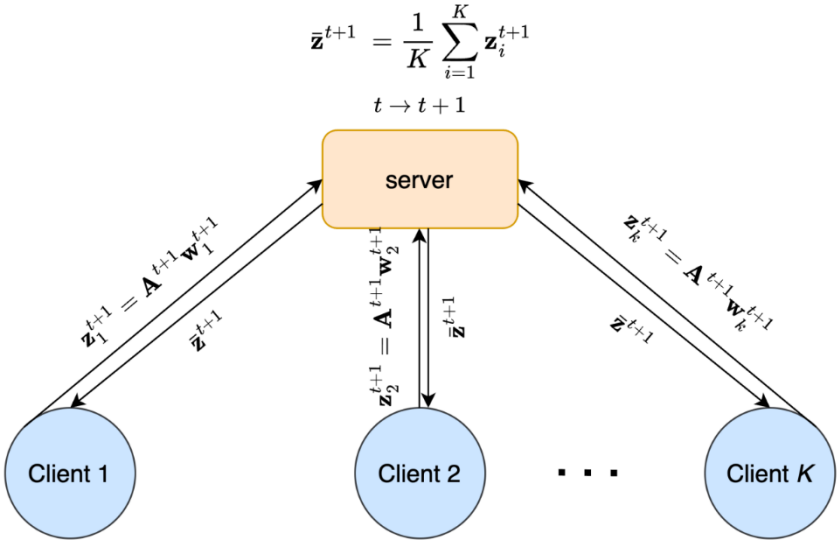

Figure 2. Random-Projection-ADMM Workflow.

The process of random projection matrix generation and dimensionality reduction consists of two key steps, with the matrix computation implemented via NumPy 1.26.0 and statistical sampling via SciPy 1.11.3. First, in projection matrix generation, clients generate

a random projection matrix $A_i^t \in \mathbb{R}^{m \times n}$ based on the seed $seed^t$ synchronized by blockchain smart contracts (the seed is updated in each training round). The elements of the matrix follow a normal distribution $A_{ij}^t \sim \mathcal{N}(0, 1/n)$. This distribution characteristic ensures that the pairwise distance of projected data is approximately preserved, which complies with the requirements of the Johnson–Lindenstrauss lemma. Second, in low-dimensional parameter calculation, the clipped gradient is projected into a low-dimensional space, with the calculation formula as follows:

$$z_i^t = A_i^t \cdot \text{clip}(g_i^t, C_i^t) \tag{5}$$

where m is the dimension after projection, which can be dynamically adjusted according to the actual communication bandwidth. For example, when $m = 100$, the parameter transmission volume of the VGG16 model is drastically reduced from 512 MB to 400 Bytes, a reduction of 1.28×10^6 times, which greatly reduces communication overhead.

In the experiments on CIFAR-10 (ResNet-18) and ChestX-ray14 (DenseNet-121), the random projection dimension m is set to 80 and 100, respectively. The selection of m follows two criteria: (1) maintaining the pairwise distance preservation property of the Johnson–Lindenstrauss lemma ($m \geq 4log(K)/\varepsilon^2$, where K is the number of clients and $\varepsilon = 0.1$); (2) adapting to edge network bandwidth constraints (ensuring single-round parameter transmission $\leq$20 KB).

ADMM distributed consensus optimization, implemented on PyTorch 2.1.0 with CUDA 12.2 for parallel acceleration, achieves distributed consensus by alternately updating local variables, global variables, and dual variables, which is well adapted to the requirements of off-chain parallel training and on-chain lightweight certification. In local variable update (client side), based on the global parameter $\bar{z}^{t-1}$ and dual variable y_i^{t-1} of the previous round, clients update the local variable w_i^t by minimizing the local objective function, with the objective function as follows:

$$w_i^t = \arg\min_{w_i} \left[f_i(w_i) + (y_i^{t-1})^T (A_i^t \cdot \text{clip}(\nabla f_i(w_i), C_i^t) - \bar{z}^{t-1}) + \frac{\rho}{2} \parallel A_i^t \cdot \text{clip}(\nabla f_i(w_i), C_i^t) - \bar{z}^{t-1} \parallel_2^2 \right] \tag{6}$$

where $f_i(w_i) = \frac{1}{|D_i|} \sum_{(x,y) \in D_i} \ell(y, f(x; w_i))$ is the local loss function. This formula ensures the rationality and effectiveness of local variable updates.

In the global variable update (aggregation node side), the aggregation node collects low-dimensional parameters z_i^t uploaded by all clients and updates the global parameter $\bar{z}^t$ by weighted averaging, with the calculation formula as follows:

$$\bar{z}^t = \frac{1}{K} \sum_{i=1}^{K} \left(z_i^t + \frac{1}{\rho} y_i^{t-1} \right) \tag{7}$$

If some clients fail to upload parameters on time, to ensure the continuity of global updates, historical parameter interpolation is adopted for filling, and the calculation formula is adjusted as follows:

$$\bar{z}^t = \frac{1}{K} \left(\sum_{i \in S_t} z_i^t + \frac{1}{\rho} y_i^{t-1} + \sum_{i \notin S_t} \bar{z}^{t-1} \right) \tag{8}$$

where S_t is the set of clients that uploaded parameters normally in the t-th round. This processing method ensures the stability of the global training process.

In dual variable update (client side), clients receive the global parameter $\bar{z}^t$ sent by the aggregation node and update the dual variable according to the deviation between the

global parameter and the local parameter to reduce the difference between local and global variables, with the update formula as follows:

$$y_i^t = y_i^{t-1} + \rho\left(z_i^t - \bar{z}^t\right) \tag{9}$$

The update of dual variables further improves the accuracy and consistency of distributed consensus. In the step of on-chain certification and synchronization, clients upload the hash value of z_i^t, projection matrix seed $seed^t$, and clipping norm C_i^t to the blockchain. The Update Verify Contract verifies the compliance of this information and records it on the blockchain after passing the verification. Meanwhile, the aggregation node synchronizes the hash value of $\bar{z}^t$) and training round t to the blockchain, ensuring the traceability of global parameters and providing a basis for subsequent audits and verifications.

3.3.3. Convergence Analysis

Theorem 1. *In the BE-DPFL framework, the global parameter $\bar{z}^t$ of the Random-Projection-ADMM mechanism satisfies the following convergence property:*

$$\| \bar{z}^t - \bar{z}^* \|_2^2 \leq \frac{1}{t}\left(\frac{\| y_i^0 \|_2^2}{\rho} + \frac{2L}{\rho^2}\sum_{i=1}^{K} \mathbb{E} \| z_i^0 - \bar{z}^0 \|_2^2\right) \tag{10}$$

where $\bar{z}^*$ is the globally optimal parameter, L is the Lipschitz constant of the loss function f_i, and y_i^0 is the initial value of the dual variable.

The proof of this theorem is based on the nonconvex convergence theory of the ADMM algorithm [33]. In the BE-DPFL framework, the random projection matrix A_i^t satisfies the Johnson–Lindenstrauss lemma, which means that the pairwise distance of projected parameters can be approximately preserved. Therefore, the original consensus optimization characteristics of the ADMM algorithm are not affected. Meanwhile, the use of low-dimensional parameters reduces the computational complexity during update iterations, further improving the convergence speed of the algorithm, thus ensuring the effectiveness and stability of the Random-Projection-ADMM optimization mechanism in the BE-DPFL framework.

3.4. MOO Adaptive Gradient Clipping Mechanism

To mitigate the model accuracy loss caused by the introduction of DP noise, this section designs an adaptive gradient clipping mechanism based on MOO. Under the premise of strictly ensuring (ε, δ)-DP, this mechanism can dynamically adjust the clipping norm to maximize model training utility, achieving collaborative optimization of privacy protection and model performance.

3.4.1. Core Design Concept

Traditional gradient clipping methods [34] usually adopt a fixed norm (e.g., C = 1.0), which is difficult to adapt to the differences in gradient distribution at different training stages and the heterogeneity of data. It is prone to problems such as over-strong privacy protection (i.e., excessive clipping leading to the loss of a large amount of effective gradient information) or over-weak privacy protection (i.e., insufficient clipping failing to meet privacy protection requirements). In contrast, this mechanism takes "satisfaction of privacy constraints" and "maximization of model accuracy" as two core objectives, and dynamically optimizes the clipping norm C_i^t through the MOO method. In each training round, the gradient clipping operation of clients can not only strictly meet the DP requirements, but

also retain effective gradient information to the greatest extent, providing a guarantee for the improvement of model performance.

3.4.2. Differential Privacy Constraint Modeling

This mechanism is based on the local differential privacy (LDP) framework. After clients perform clipping processing on gradients, Gaussian noise is injected to satisfy the (ε, δ)-DP requirements. Let the clipped gradient be $\hat{g}_i^t = \text{clip}\left(g_i^t, C_i^t\right)$, its L2 sensitivity can be expressed as follows:

$$\Delta = \max_{i,t} \ \| \hat{g}_i^t \|_2 \leq C_i^t \tag{11}$$

The injected Gaussian noise needs to satisfy $\eta_i^t \sim \mathcal{N}\left(0, \sigma^2 I\right)$, where the value of noise variance σ must meet the following condition:

$$\sigma \geq \frac{\Delta\sqrt{2\ln(1.25/\delta)}}{\epsilon_i} \tag{12}$$

Combined with the privacy budget allocation method introduced in Section 3.2.3, the variance of Gaussian noise injected by client i in the t-th training round can be calculated by the following formula:

$$\sigma_i^t = \frac{C_i^t\sqrt{2\ln(1.25/\delta)}}{\epsilon_i} \tag{13}$$

The gradient after noise injection is as follows:

$$\tilde{g}_i^t = \hat{g}_i^t + \eta_i^t \tag{14}$$

The above modeling process ensures that the gradient processing process strictly satisfies DP constraints, providing reliable protection for data privacy.

3.4.3. Construction of MOO Model

To maximize model accuracy under the premise of satisfying DP constraints, a bi-objective optimization problem is constructed, with its mathematical expression as follows:

$$\begin{cases} \max\limits_{\{C_i^t\}} \quad U = \sum_{i=1}^{K} \frac{1}{|D_i|}\ell(y, f(x; w_i^t)) \\ s.t. \ \ \epsilon_i \leq \epsilon_{\text{alloc}}^t (\text{Privacy Budget Constraint}) \\ \quad\quad C_i^t \geq \underline{C} (\text{Minimum Clipping Norm Constraint}) \\ \quad\quad C_i^t \leq \overline{C} (\text{Maximum Clipping Norm Constraint}) \end{cases} \tag{15}$$

In this optimization model, U represents the global model utility (i.e., model accuracy), $\ell(\cdot)$ is the prediction loss function used to measure the difference between the model prediction results and the actual labels; $\epsilon_{\text{alloc}}^t$ is the allocated budget of client i in the t-th training round, which is dynamically updated by the Privacy Budget Contract according to actual conditions; $\underline{C} = 0.1$ and $\overline{C} = 5.0$ are the lower and upper bounds of the clipping norm, respectively. This range is a standard and empirically validated interval for privacy-preserving model training, which can achieve a good balance between privacy protection and model performance.

To solve the above MOO problem, the weighted sum method is adopted to convert the bi-objective problem into a single-objective function, with the converted single-objective function as follows:

$$\max_{\{C_i^t\}} J = \alpha \cdot U - \beta \cdot \text{DPViolation} \tag{16}$$

where α and β are weight coefficients satisfying $\alpha + \beta = 1$. In the early stage of training, to prioritize ensuring that the model can quickly learn effective features, $\alpha = 0.6$ and $\beta = 0.4$ are set; as the training process progresses, the weight coefficients are gradually adjusted to $\alpha = 0.4$ and $\beta = 0.6$ to strengthen the emphasis on privacy protection, realizing the dynamic balance between model accuracy and privacy protection. DPViolation is used to measure the degree of violation of privacy constraints, which is defined as follows:

$$\text{DPViolation} = max\left(0, \frac{\epsilon_i - \epsilon_{\text{alloc}}^t}{\epsilon_{\text{alloc}}^t}\right) \tag{17}$$

This definition can clearly judge whether the current privacy protection measures meet the constraint requirements, providing a basis for the optimization of the single-objective function.

3.4.4. Solution of Adaptive Clipping Norm

The gradient descent method [35] is used to solve the single-objective optimization problem (Equation (16)) to realize the dynamic update of the clipping norm C_i^t, with the specific steps as follows. First, in the initialization phase, in the 1st training round, the initial value of the clipping norm C_i^1 is determined by calculating the statistical information of local gradients, with the calculation formula as $C_i^1 = \frac{1}{|D_i|}\sum_{(x,y)\in D_i} \parallel \nabla f(x; w_i^0) \parallel_2$. This initial value can be well adapted to the gradient distribution characteristics of local data.

Second, in the gradient calculation step, the gradient of the objective function J with respect to C_i^t is calculated. This gradient reflects the degree of influence of the change of the clipping norm on the value of the objective function, with the calculation formula as follows:

$$\frac{\partial J}{\partial C_i^t} = \alpha \cdot \frac{\partial U}{\partial C_i^t} - \beta \cdot \frac{\partial \text{DPViolation}}{\partial C_i^t} \tag{18}$$

where $\frac{\partial J}{\partial C_i^t}$ is calculated through the local validation set, reflecting the influence of the change of clipping norm on model accuracy; $\frac{\partial \text{DPViolation}}{\partial C_i^t}$ reflects the influence of the change of clipping norm on the satisfaction of privacy constraints. Through the careful consideration of these two parts, the optimization direction of the clipping norm can be determined.

Finally, in the iterative update phase, the clipping norm is iteratively updated according to the calculated gradient information using the following formula:

$$C_i^{t+1} = \text{clip}\left(C_i^t + \gamma \cdot \frac{\partial J}{\partial C_i^t}, \underline{C}, \overline{C}\right) \tag{19}$$

where $\gamma = 0.01$ is the learning rate used to control the update step size of the clipping norm, avoiding unstable convergence caused by excessively large step sizes or slow convergence caused by excessively small step sizes; the $\text{clip}(\cdot)$ function is used to ensure that the updated clipping norm is always kept between $\underline{C}$ and $\overline{C}$, complying with the preset constraint requirements and ensuring the rationality and effectiveness of the gradient clipping operation.

The weight coefficients α/β and learning rate γ are determined via grid search on the validation set: $\alpha = 0.6/\beta = 0.4$ (early training, rounds 1–40), $\alpha = 0.5/\beta = 0.5$ (middle training, rounds 41–70), $\alpha = 0.4/\beta = 0.6$ (late training, rounds 71–100); $\gamma = 0.01$ is fixed after verifying stability across $\gamma \in \{0.005, 0.01, 0.02\}$. A sensitivity analysis confirms that model accuracy and privacy compliance remain within $\pm 1\%$ when α/β varies by ± 0.1, and γ varies by ± 0.005.

3.4.5. Adaptation to the Overall Architecture

After each training round, clients upload the clipping norm C_i^t and noise variance σ_i^t to the blockchain. The Privacy Budget Contract verifies whether these parameters meet the privacy budget constraints through the following equation:

$$\epsilon_i = \frac{C_i^t \sqrt{2\ln(1.25/\delta)}}{\sigma_i^t} \leq \epsilon_{\text{alloc}}^t \tag{20}$$

If the verification is passed, the clipped gradient is reduced in dimension through random projection technology and then uploaded to the aggregation node to participate in global model updates; if the verification fails, the clipping norm and noise variance need to be adjusted again until the privacy budget constraints are satisfied. This verification mechanism ensures the compliance and auditability of privacy-enhancing operations, making the entire training process proceed in an orderly manner under the premise of protecting privacy and realizing the seamless adaptation of the MOO adaptive gradient clipping mechanism and the overall BE-DPFL architecture.

4. Experimental Validation and Result Analysis

To comprehensively verify the performance of the Blockchain-Enhanced Differentially Private FL framework (BE-DPFL), this chapter designs targeted experiments covering three core dimensions: the framework's problem-solving capability, the effectiveness of key mechanisms, and its adaptability to real-world scenarios. The FL environment is built using the PyTorch 2.0 deep-learning framework, the blockchain layer is implemented with Hyperledger Fabric 2.5 [36] as a consortium chain, smart contracts are deployed on peer nodes with endorsement policies requiring $\geq 2/3$ validation node approval. The privacy enhancement layer employs the TensorFlow Privacy toolkit for DP noise injection.

4.1. Experimental Setup

The hardware environment includes two categories of nodes. Participant nodes (clients) are equipped with Intel Core i7-12700H processors (Intel Corporation, Santa Clara, CA, USA), 32 GB DDR5 memory, and NVIDIA RTX 4070 GPUs (NVIDIA Corporation, Santa Clara, CA, USA). Validation and aggregation nodes are configured with Intel Xeon Gold 6330 processors (Intel Corporation, Santa Clara, CA, USA), 128 GB DDR4 memory, and NVIDIA A100 GPUs (NVIDIA Corporation, Santa Clara, CA, USA) to ensure sufficient high-performance computing capability. All nodes are connected via a 1000 Mbps Ethernet network, with latency maintained between 5 and 15 ms, simulating a typical edge–cloud collaborative setting. In all experiments, the number of participant nodes (clients) $K = 20$, validation nodes = 7, and aggregation nodes = 2 (rotated every 20 training rounds to avoid single-point failure).

Two representative public datasets are selected to validate the generality and applicability of the proposed framework. The first is the medical imaging dataset ChestX-ray14 [37], which contains 112,120 chest X-ray images and 14 disease labels, used to simulate high-sensitivity medical collaborative training scenarios. The second is the CIFAR-10 [38] dataset for general image classification, consisting of 50,000 training images and 10,000 test images across 10 categories, used to assess the framework's applicability in non-medical contexts. ChestX-ray14 is selected for high-sensitivity scenarios because it contains sensitive medical data with heterogeneous disease labels, simulating cross-hospital collaborative diagnosis. CIFAR-10 is chosen for general-purpose scenarios due to its balanced class distribution and wide use as a benchmark for FL accuracy evaluation. The two datasets cover both high-sensitivity and general cases, ensuring the framework's generality.

Baseline model choices are aligned with the task characteristics: DenseNet-121 [39] is adopted for the medical scenario due to its lightweight structure, while ResNet-18 is used for the general classification task. Unified federated training hyperparameters are set as follows: local iteration rounds E = 5, global training rounds T = 100, learning rate $\eta = 0.01$ (decayed using a cosine annealing schedule), and batch size of 32.

Evaluation metrics span three dimensions—performance, accuracy, and privacy—to comprehensively assess the framework's effectiveness and security. Performance metrics include consensus latency (time per consensus round, in ms), communication overhead (data transmitted per round, in MB), and training throughput (number of global iterations completed per unit time, rounds/minute). Accuracy metrics include test accuracy and accuracy loss relative to centralized training. Privacy metrics measure the consumption and compliance of the DP mechanism, including privacy budget consumption rate (average ε consumed per round) and DP compliance rate (the proportion of training rounds satisfying the (ε, δ) constraint, with δ set to 1×10^{-5}).

4.2. Evaluation of Consensus-Mechanism Performance Bottlenecks

To validate the communication-efficiency advantages of the proposed framework, this experiment conducts a systematic comparison across blockchain-empowered FL methods with integrated DP. The evaluation focuses on three system-level metrics—consensus efficiency, communication load, and training throughput—tested under different datasets and model scales. This design enables a comprehensive assessment of how the proposed BE-DPFL framework performs in both general-purpose and high-privacy scenarios.

Three State-of-the-Art blockchain-based FL schemes are selected as baselines, each representing a distinct design paradigm. DP-BC-DGFL [25] enhances decentralized graph FL with DP for UAV disaster-response tasks. ZT-DP-BFL [26] introduces a zero-trust privacy-preserving architecture by embedding DP within a blockchain-enabled IoT training framework. LDP-I-BFL [27] integrates local differential privacy with on-chain incentive mechanisms to strengthen participant reliability and collaboration motivation.

In the experimental setup, the number of participating nodes is fixed at K = 20. Training is carried out for 100 rounds using ResNet-18 on CIFAR-10 and DenseNet-121 on ChestX-ray14. For each method, three key indicators—consensus latency, communication overhead, and training throughput per round—are collected. All experiments are independently repeated five times, and the averaged results are reported to ensure robustness and reproducibility. The comparative outcomes are summarized in Figures 3 and 4.

The experimental results first demonstrate the significant advantages of BE-DPFL in consensus latency. As shown in both figures, BE-DPFL achieves single-round consensus latencies of only 78.2 ms on CIFAR-10 and 82.5 ms on ChestX-ray14, which is substantially lower than the three baseline methods. Compared with the best-performing baseline, LDP-I-BFL, the latency is reduced by about 50%, and compared with the bandwidth-optimized ZT-DP-BFL, the reduction exceeds 64%. These results indicate that BE-DPFL's lightweight on-chain consensus mechanism significantly reduces node interaction and synchronization overhead even under structurally complex and heterogeneous environments, achieving highly efficient cross-node coordination. Furthermore, the framework's staged processing of critical and non-critical gradient information simplifies on-chain verification for ordinary updates, further shortening the consensus path and ensuring consistently low-latency performance across tasks of different scales.

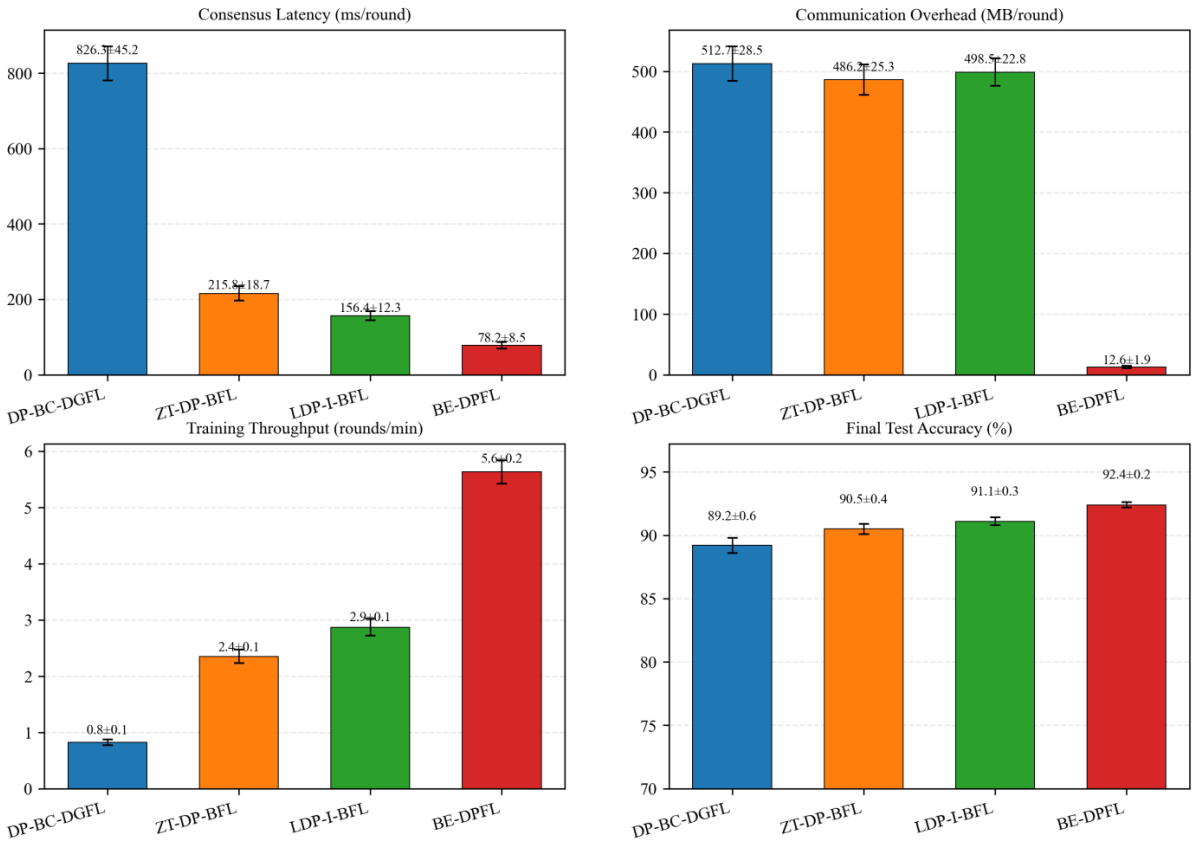

Figure 3. Performance Comparison of Blockchain-based Federated Learning Methods (CIFAR-10 Dataset).

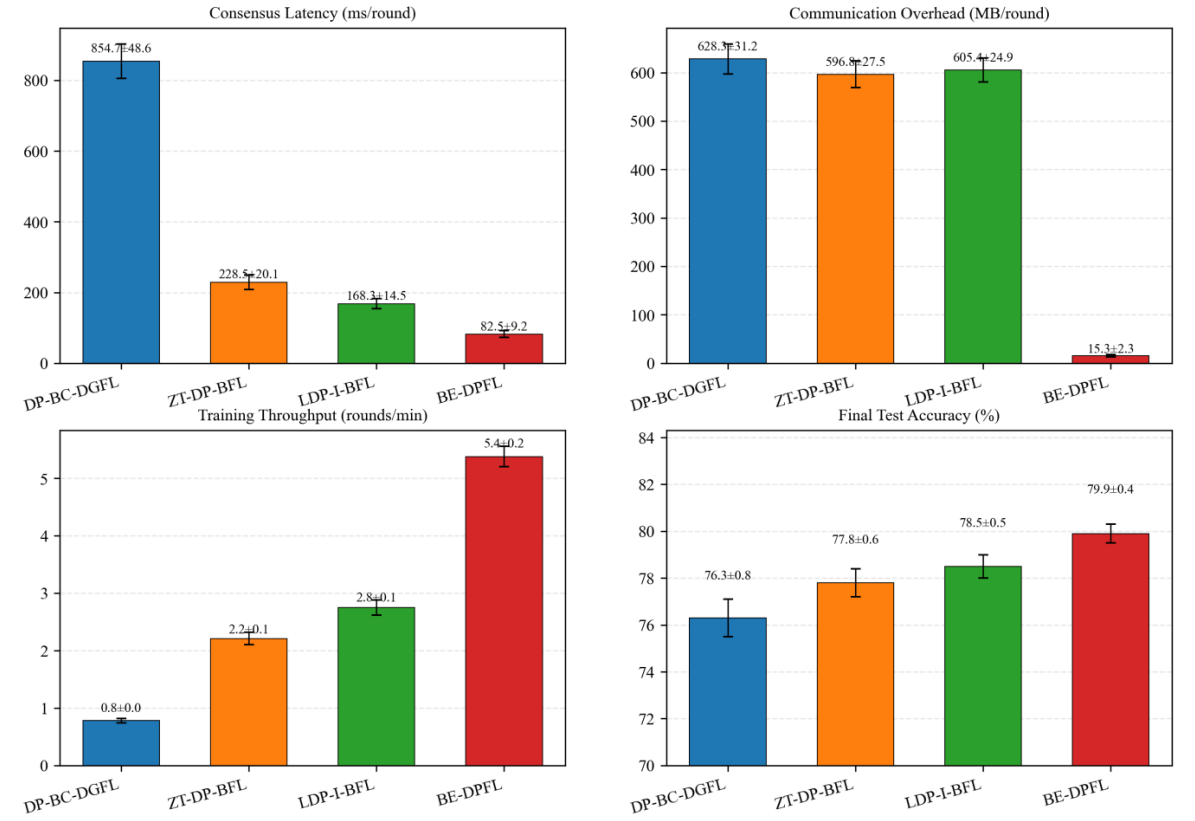

Figure 4. Performance Comparison of Blockchain-based Federated Learning Methods (ChestX-ray14 Dataset).

From the communication-overhead and throughput results, BE-DPFL achieves a breakthrough in system efficiency. Its communication overhead is only 12.6–15.3 MB per round—more than 97% lower than the baselines (486.2–628.3 MB). BE-DPFL is the only method for reducing on-chain transmission to double-digit MB. This improvement primarily stems from the proposed Random-Projection-ADMM dimensionality reduction mechanism, which maps high-dimensional gradients to a low-dimensional subspace and uploads only projection seeds and parameter hashes, fundamentally relieving the congestion and bandwidth bottlenecks that commonly hinder blockchain-based FL.

The drastic reduction in communication overhead and consensus latency results in a substantial increase in throughput: BE-DPFL reaches 5.38–5.63 rounds/min, which is 1.9× that of LDP-I-BFL, 2.4× that of ZT-DP-BFL, and over 7× that of DP-BC-DGFL. Despite the significant efficiency improvements, BE-DPFL still achieves the highest final model accuracy on both datasets, reaching 92.4% and 79.9%, respectively—surpassing the best baseline by 1.3–1.4 percentage points. This demonstrates that high-speed consensus and low communication costs not only boost throughput but also reduce training interruptions caused by latency, helping maintain stable gradient-update rhythms. Meanwhile, the on-chain trusted-recording mechanism effectively mitigates risks from malicious or abnormal parameter uploads, ensuring robust model convergence.

4.3. Evaluation of Accuracy Loss Under Differential Privacy

To assess the proposed method's ability to preserve accuracy and maintain privacy compliance under different DP strategies, this experiment conducts a systematic comparison of mainstream DP-based FL mechanisms. The focus is on analyzing model performance differences and DP-constraint satisfaction under a fixed privacy budget. Three representative DP-FL methods from recent research are selected as baselines, covering gradient-leakage prevention, communication-constrained optimization, and multimodal DP mechanisms. Specifically, the baselines include: GLA-DPFL (Gradient-Leakage-Aware Differential Privacy FL) proposed by Hu et al. [18], which evaluates the effectiveness of standard DP mechanisms under gradient leakage attacks and analyzes DP strength and privacy robustness; KD-DPFL (Knowledge-Distillation Differential Privacy FL) by Gad et al. [19], which combines knowledge distillation with DP to balance communication efficiency and privacy protection for bandwidth-constrained FL environments; and MS-DP-MFL (Modality-Selection Differential Privacy Multimodal FL) proposed by Ma et al. [20], which integrates modality selection with DP to improve privacy compliance and model performance in multimodal learning tasks. All methods are evaluated under a unified privacy budget ($\varepsilon = 5, \delta = 1 \times 10^{-5}$), with the number of participating nodes fixed at K = 20. Training is conducted for 100 rounds on CIFAR-10 and ChestX-ray14, and each round's test accuracy, accuracy loss rate, and privacy compliance rate are recorded. To measure the impact of different DP strategies on model performance, "centralized training without privacy protection" is used as the accuracy upper bound. The experimental results are shown in Figures 5 and 6.

The experimental results first demonstrate that BE-DPFL exhibits significant accuracy advantages across different DP-FL scenarios. On both CIFAR-10 and ChestX-ray14, BE-DPFL achieves accuracy loss rates of only 2.94% and 4.31%, substantially outperforming GLA-DPFL (9.35–12.33%) and KD-DPFL (7.04–9.22%), representing relative reductions of 65–68% and 53–58%. This improvement arises from the adaptive MOO gradient-clipping strategy introduced in BE-DPFL. By dynamically adjusting the clipping norm during different training stages—retaining more high-value gradient information in early training ($\alpha = 0.6$) and strengthening privacy constraints during later stages ($\alpha = 0.4$)—the method effectively avoids gradient distortion and information loss caused by fixed-norm clipping.

At the same time, the adaptive noise injection mechanism adjusts noise strength based on local gradient variance, enabling BE-DPFL to maintain a more stable gradient distribution compared to MS-DP-MFL in multimodal learning tasks, ultimately achieving higher convergence quality. Furthermore, BE-DPFL achieves the highest test accuracy on both datasets (92.4% and 79.9%), approaching the accuracy upper bound of centralized training under DP constraints, demonstrating the optimal balance between privacy and performance.

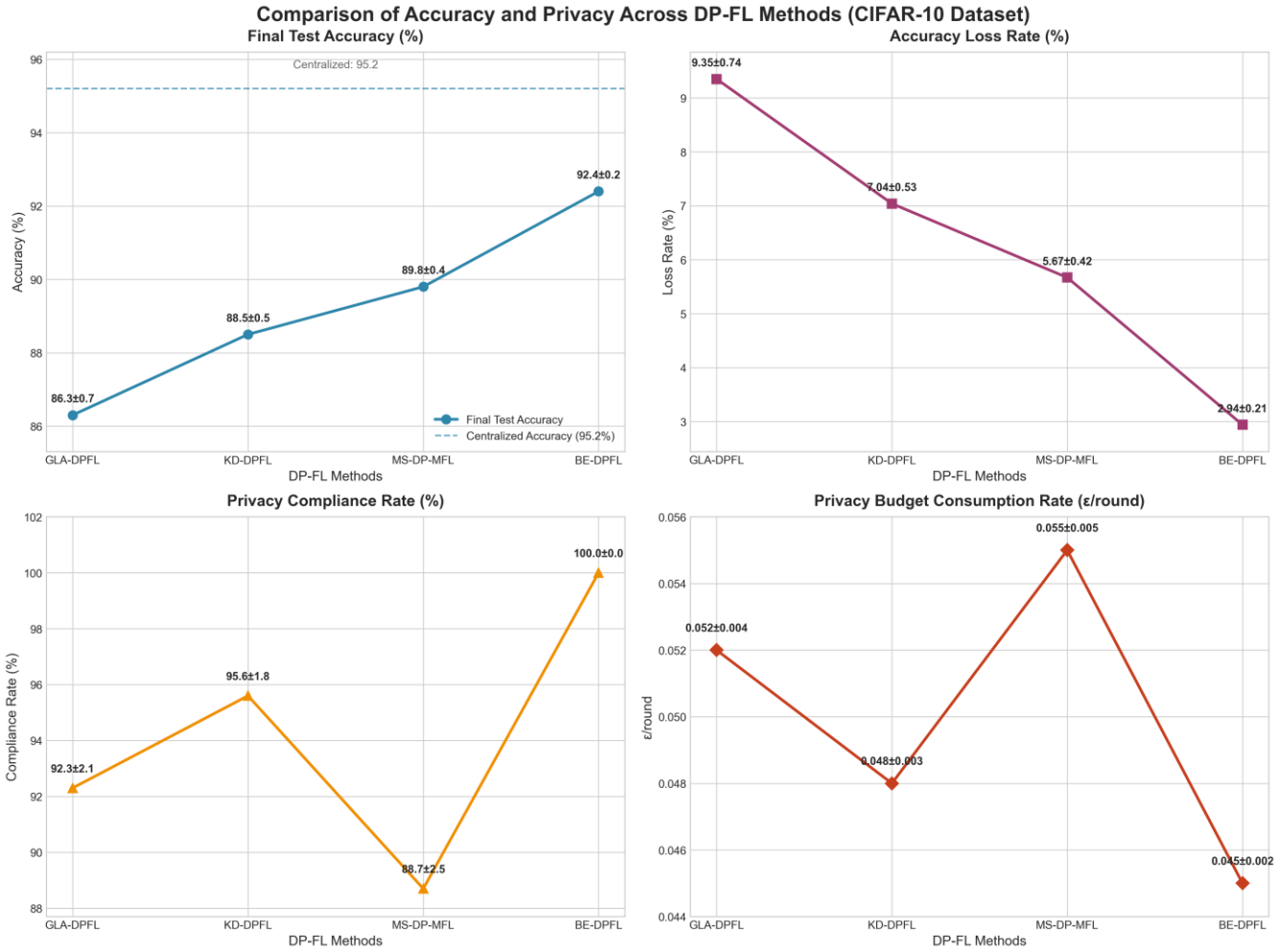

Figure 5. Comparison of Accuracy and Privacy Across DP-FL Methods (CIFAR-10 Dataset).

In terms of privacy protection and budget sustainability, BE-DPFL also shows leading performance. Its privacy compliance rate reaches 100% on both datasets, significantly surpassing GLA-DPFL (91.5–92.3%) and MS-DP-MFL (87.9–88.7%). This is primarily due to the Privacy Budget Contract mechanism introduced in BE-DPFL, which conducts on-chain auditing to verify clipping norms, noise variance, and per-round budget usage in real time (ensuring $\varepsilon_i \leq \varepsilon_{\text{alloc}}^t$). This prevents situations where participants illegally reduce noise to improve accuracy, ensuring that the entire training process satisfies (ε, δ)-DP constraints. Meanwhile, BE-DPFL achieves a lower privacy budget consumption rate (0.045–0.047 ε/round), a reduction of 5–17% compared to the baselines, reflecting higher budget efficiency. Since the allocation strategy is correlated with $\sqrt{n_i} \cdot s_i$, nodes with smaller data volumes or higher gradient noise do not excessively consume privacy budget. As a result, the model retains 0.3–0.5 ε of unused budget after 100 training rounds, supporting further secure fine-tuning or incremental learning tasks. Overall, BE-DPFL achieves the best performance in accuracy preservation, privacy compliance, and

budget efficiency, demonstrating comprehensive superiority in efficiency, sustainability, and privacy protection.

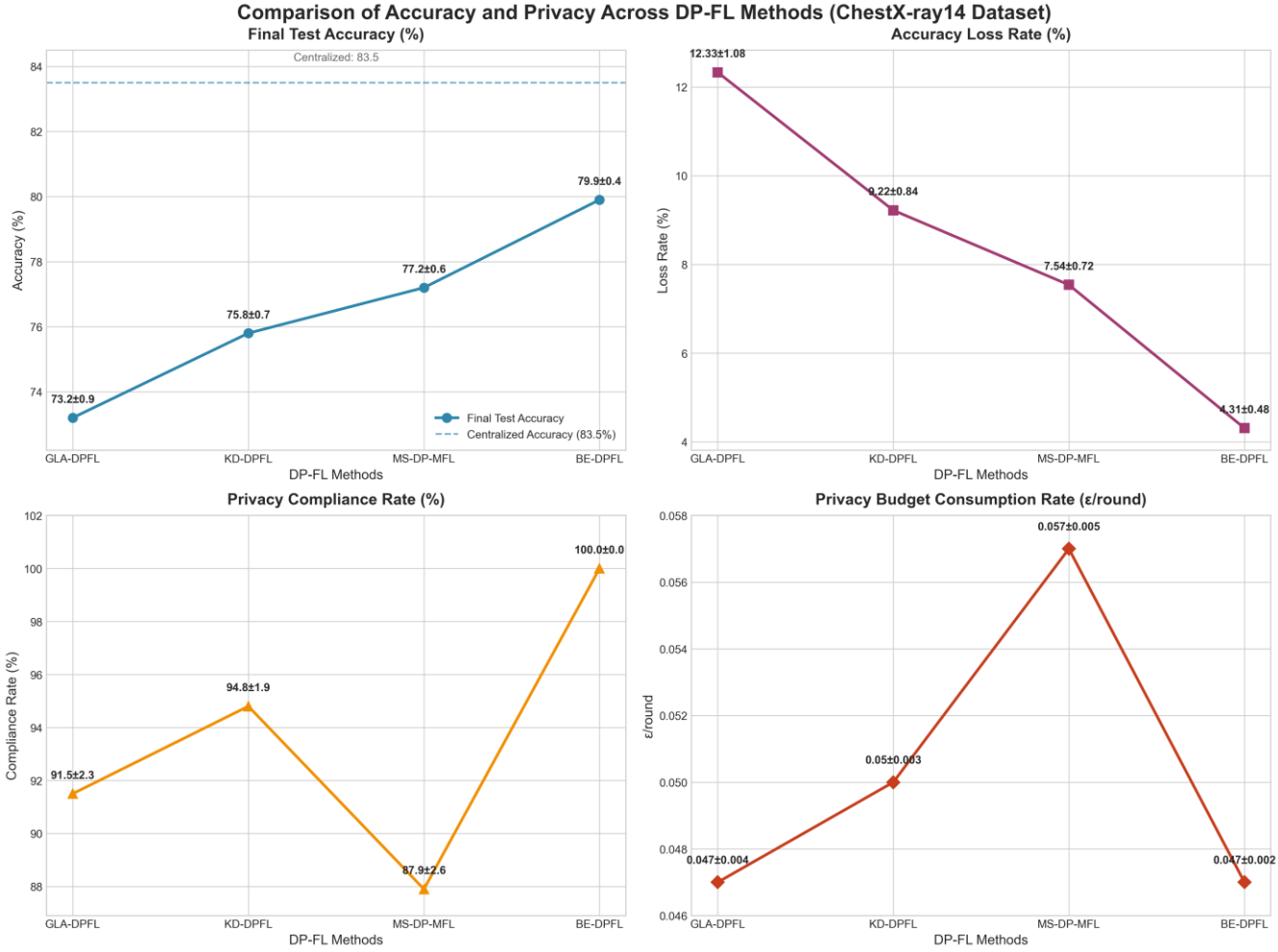

Figure 6. Comparison of Accuracy and Privacy Across DP-FL Methods (ChestX-ray14 Dataset).

4.4. Robustness Evaluation Under Malicious Nodes and Network Fluctuations

This experiment aims to systematically evaluate the robustness of BE-DPFL under non-ideal operating conditions, focusing on its stability and reliability when facing typical real-world disruptions such as malicious participants and network fluctuations. Two categories of disturbance scenarios are constructed for this purpose. In the malicious-node scenario, m = 2/4/6 malicious clients (accounting for 10%/20%/30% of K = 20 total clients) are injected, where these nodes upload randomly generated fake gradient hashes or deliberately violate privacy constraints (e.g., forcing the gradient clipping bound to C = 10.0, far exceeding the upper limit) to degrade the training process. In the network-fluctuation scenario, a traffic-control tool is used to artificially increase the communication latency between nodes (from the normal 5–15 ms to 20–100 ms) and introduce packet loss rates of p = 5%/10%/15%, simulating unstable links commonly found in edge collaboration environments. The comparison includes BE-DPFL and two strong baseline models: LDP-I-BFL (blockchain-based local differential privacy FL) and MS-DP-MFL (multi-source differentially private hybrid FL). Evaluation metrics include anomaly detection rate and accuracy drop for malicious-node experiments, and training interruption rate and throughput degradation for network-fluctuation experiments, providing a comprehensive assessment of robustness under adversarial and dynamic conditions. The results are presented in Figures 7 and 8.

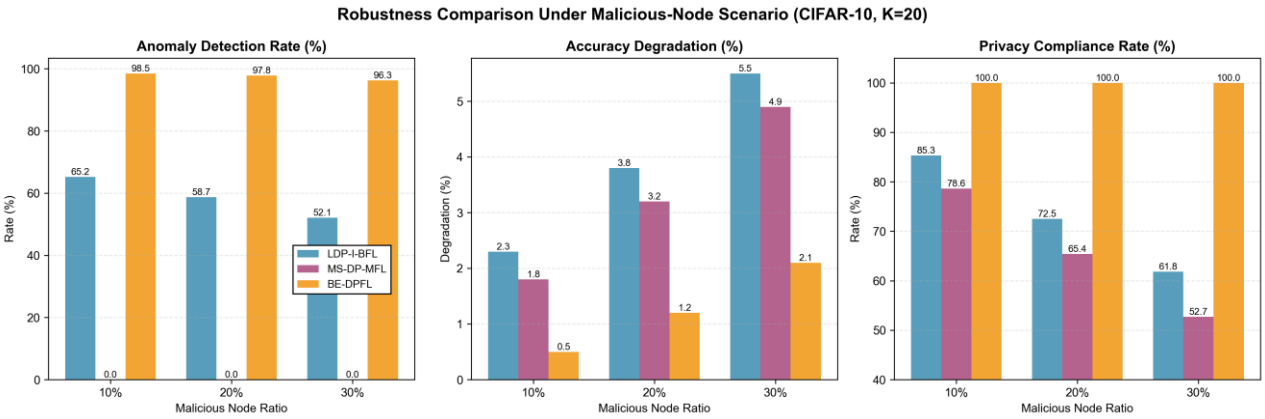

Figure 7. Robustness Comparison Under Malicious-Node Scenario (CIFAR-10 Dataset, K = 20).

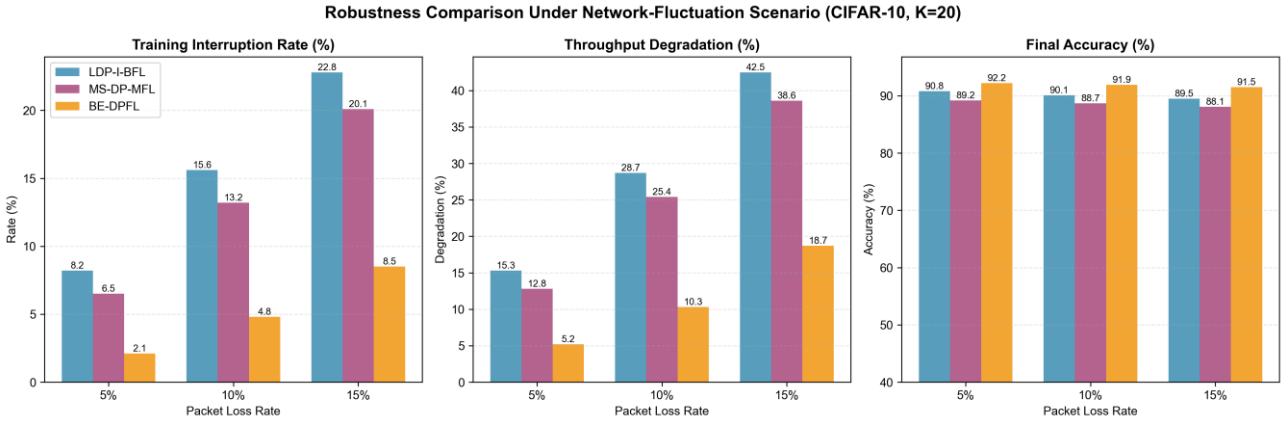

Figure 8. Robustness Comparison Under Network-Fluctuation Scenario (CIFAR-10 Dataset, K = 20).

First, in the malicious-node scenario, BE-DPFL demonstrates a significant advantage in both anomaly detection and performance preservation. As shown in Figure 7, BE-DPFL maintains an exceptionally high anomaly detection rate of 96.3–98.5%, far surpassing LDP-I-BFL (52.1–65.2%) and the MS-DP-MFL model, which fails to detect any malicious behavior (0%). This superiority mainly stems from the Update Verify Contract embedded on the blockchain, which performs round-by-round consistency verification of parameter hashes, enabling rapid identification of random gradients or updates that bypass required clipping. Thanks to this high detection capability and strict privacy enforcement, BE-DPFL limits accuracy degradation to only 2.1% even when the proportion of malicious nodes reaches 30%, notably better than LDP-I-BFL (5.5%) and MS-DP-MFL (4.9%). Meanwhile, BE-DPFL consistently maintains a 100% privacy compliance rate across all settings, whereas the baselines show significant declines as malicious-node ratios increase, validating the effectiveness of BE-DPFL's incentive–penalty mechanism and the immutability of blockchain in enforcing normative behavior.

Second, in the network-fluctuation scenario, BE-DPFL also exhibits strong robustness in maintaining system stability and training continuity. Figure 8 shows that when the packet loss rate increases from 5% to 15%, BE-DPFL's training interruption rate remains as low as 2.1–8.5%, while LDP-I-BFL and MS-DP-MFL escalate to 22.8% and 20.1%, respectively. This considerable difference is primarily attributable to two design features of BE-DPFL: (i) its reduced communication overhead mitigates the need for retransmission under packet loss, and (ii) the random-projection–ADMM historical parameter interpolation allows global updates to continue even when some nodes are temporarily offline. As a result, BE-DPFL

sustains a more stable training rhythm under high packet-loss conditions, demonstrating strong resilience to network instability.

Finally, in terms of throughput and final model accuracy, BE-DPFL also demonstrates a clear advantage. When the packet loss rate reaches 15%, the throughput of BE-DPFL decreases by only 18.7%, which is significantly lower than the degradation observed in LDP-I-BFL (42.5%) and MS-DP-MFL (38.6%). More importantly, the final accuracy of BE-DPFL consistently remains above 91%, noticeably outperforming the two baseline methods. This indicates that BE-DPFL can more effectively adapt to communication uncertainty caused by network jitter, maintaining training stability while preserving model performance.

The above results indicate that, whether under strong adversarial threats (malicious nodes) or high-uncertainty conditions (network fluctuations), BE-DPFL consistently exhibits robustness significantly superior to existing methods. Its efficient on-chain verification, incentive–penalty mechanisms, and low-communication-overhead design collectively form the key foundation that enables BE-DPFL to maintain stable performance in complex edge environments.

4.5. Performance Adaptability Evaluation Under Different Privacy Budgets

To verify the adaptive capability of BE-DPFL in scenarios with varying privacy requirements, this experiment focuses on the dynamic trade-off between privacy budgets and model performance, analyzing the framework's accuracy-privacy balance under different ε constraints. Two mainstream differential privacy federated learning methods representing distinct design paradigms are selected as baselines: DP-FL [11], a classic global DP federated learning approach that adopts a fixed noise injection strategy and relies on a central server for privacy budget allocation, serving as a representative scheme for global DP-FL; and Adaptive-DPFL [14], an adaptive privacy method that dynamically adjusts noise scale based on training states but lacks a transparent budget auditing mechanism, making the execution of privacy strategies unverifiable. Experiments are conducted on the CIFAR-10 and ChestX-ray14 datasets, with basic parameters such as the number of participating nodes (K = 20) and global training rounds (T = 100) consistent with Section 4.1. Four privacy budget levels are set: $\varepsilon = 2$ (high-strength privacy), $\varepsilon = 5$ (medium-strength privacy), $\varepsilon = 8$ (low-strength privacy), and $\varepsilon = 10$ (relaxed privacy), with δ fixed at 1×10^{-5}. Evaluation metrics include final test accuracy, privacy compliance rate (proportion of training rounds satisfying (ε,δ)-DP constraints), and model convergence rounds (number of rounds required to reach stable accuracy). Each experiment is repeated five times, and the average values are reported. The results are presented in Figure 9.

The experimental results demonstrate that BE-DPFL exhibits significant comprehensive performance advantages across all privacy budget levels. In terms of accuracy, regardless of the ε value, BE-DPFL achieves higher test accuracy than the baseline methods on both datasets, with more pronounced advantages under stronger privacy constraints (smaller ε). When $\varepsilon = 2$, on the CIFAR-10 dataset, BE-DPFL's accuracy is 7.5 percentage points higher than that of DP-FL and 4.2 percentage points higher than that of Adaptive-DPFL; on the ChestX-ray14 dataset, the accuracy is 7.3 percentage points and 3.8 percentage points higher than the two baselines, respectively. This advantage stems from the MOO-based adaptive gradient clipping mechanism, which dynamically adjusts the clipping norm and noise intensity to maximize the retention of effective gradient information while strictly satisfying privacy constraints, avoiding the "over-protection" issue of fixed privacy strategies and the gradient distortion problem of traditional adaptive methods.

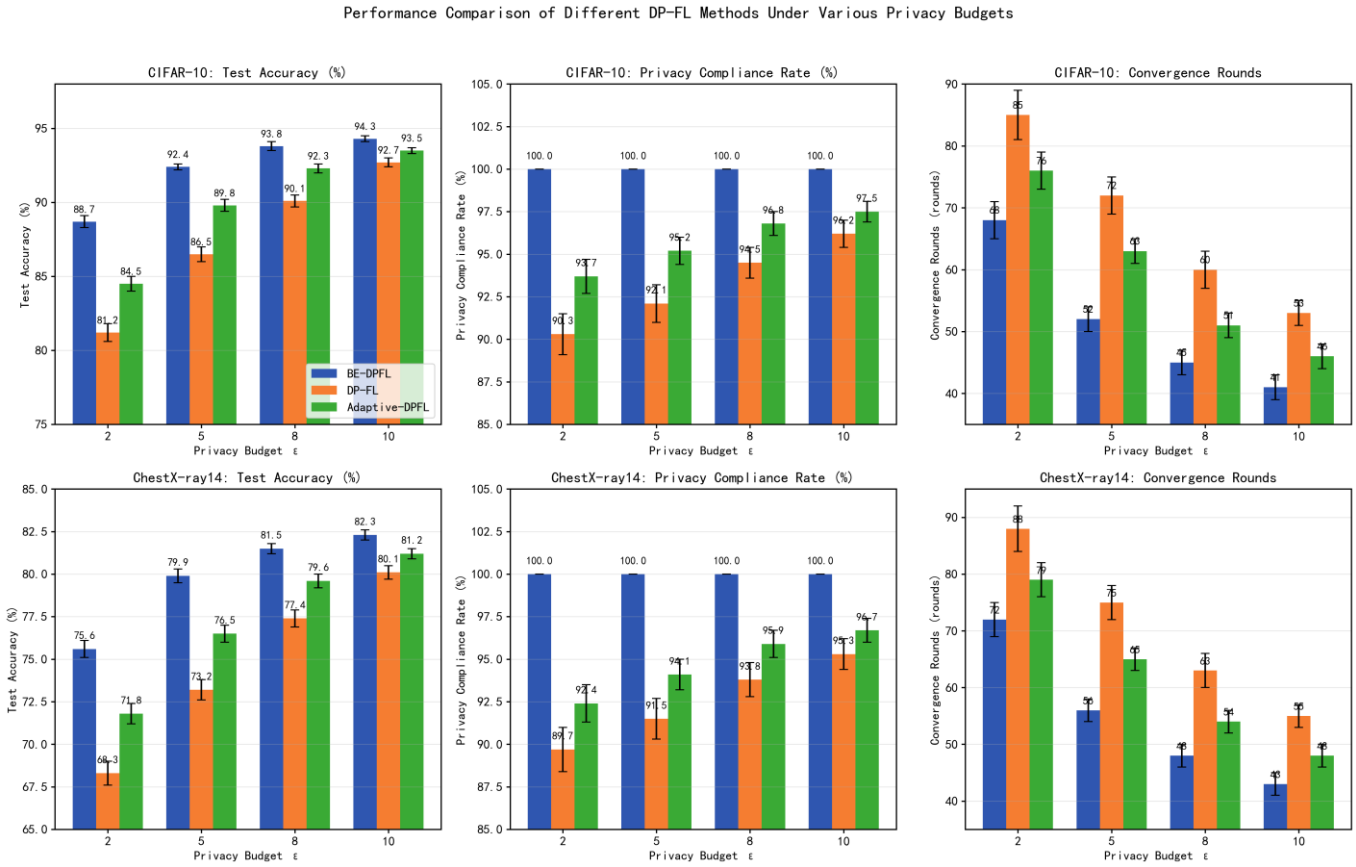

Figure 9. Performance Comparison of Different Methods Under Various Privacy Budgets.

In terms of privacy compliance and convergence efficiency, BE-DPFL also performs outstandingly. The privacy compliance rate of BE-DPFL remains 100% under all experimental settings, while the baseline methods, lacking a transparent on-chain auditing mechanism, suffer from issues such as opaque budget allocation and unverifiable noise injection execution, resulting in a maximum compliance rate of only 97.5%. Meanwhile, BE-DPFL reduces convergence rounds by 15–25% compared to DP-FL and by 8–12% compared to Adaptive-DPFL. This is attributed to the Random-Projection-ADMM mechanism, which reduces communication overhead, and the efficiency of on-chain consensus, which minimizes waiting and interruptions during the training process. The results above verify BE-DPFL's strong adaptability across scenarios with varying privacy requirements. It can not only meet the strict privacy requirements of high-sensitivity scenarios (e.g., medical data collaboration) but also achieve performance close to that of centralized training under relaxed privacy constraints, demonstrating excellent scalability across scenarios.

4.6. Ablation Study on Core Components of BE-DPFL

To verify the independent contributions of the three core components (blockchain trust layer with FL-PBFT and smart contracts, Random-Projection-ADMM optimization mechanism, and MOO-based adaptive gradient clipping) in BE-DPFL, ablation experiments are designed following the control-variable principle. The full BE-DPFL framework serves as the baseline (Full BE-DPFL), while three ablation variants are constructed by removing/replacing one component at a time to isolate the impact of each module. The experimental settings (datasets, models, hyperparameters, and hardware/software environments) are consistent with Section 4.1 to ensure fairness and comparability. Key evaluation metrics cover three core dimensions: efficiency (consensus latency per round, communication overhead per round), privacy-accuracy trade-off (test accuracy, privacy compliance

rate), and robustness (malicious node detection rate with 30% malicious participants). All experiments are independently repeated five times, and the averaged results are reported with statistical stability guaranteed. Figure 10 presents the comparative results of the four groups (Full BE-DPFL and three ablation variants) on the CIFAR-10 dataset.

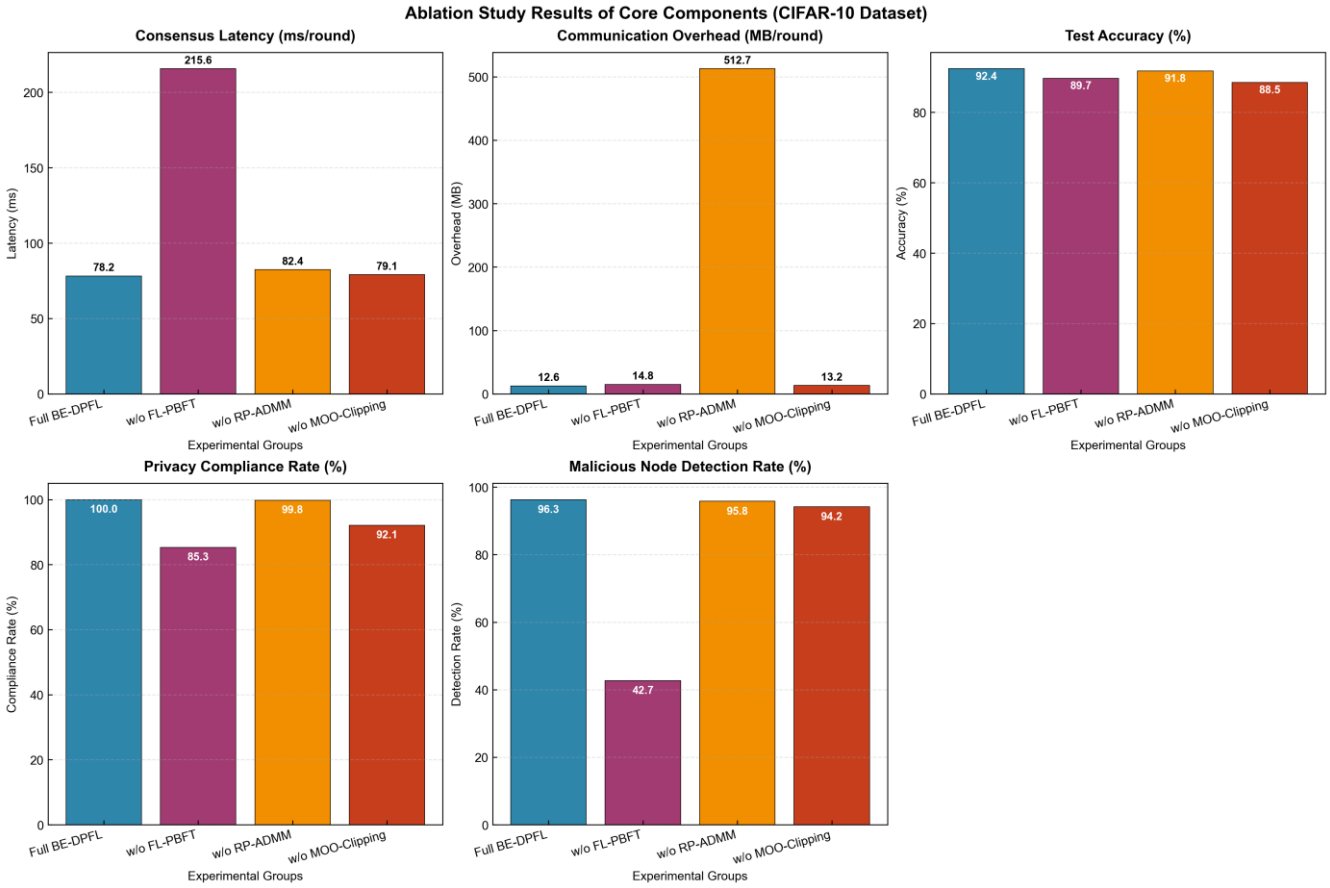

Figure 10. Ablation Study Results of Core Components (CIFAR-10 Dataset).

The ablation experiment results clearly demonstrate that each core component of BE-DPFL contributes significantly to the framework's comprehensive performance, and none are redundant. First, regarding the blockchain trust layer (FL-PBFT + smart contracts), removing it (Ablation 1: *w/o* FL-PBFT) leads to a 175% increase in consensus latency (from 78.2 ms to 215.6 ms) due to the high communication overhead of traditional PBFT consensus without phased optimization. Meanwhile, the privacy compliance rate drops by 14.7% (to 85.3%) and the malicious node detection rate plummets by 55.6% (to 42.7%), as the absence of on-chain smart contract auditing eliminates real-time verification of privacy budget usage and parameter integrity. This confirms that the blockchain trust layer is the core guarantee for low-latency consensus, high privacy compliance, and resistance to malicious behaviors. Second, for the Random-Projection-ADMM mechanism (Ablation 2: *w/o* RP-ADMM), abandoning dimensionality reduction and adopting full high-dimensional gradient upload results in a 3969% surge in communication overhead (from 12.6 MB to 512.7 MB per round), while test accuracy decreases by only 0.6% (to 91.8%). This validates that the mechanism achieves a drastic reduction in communication costs without compromising model performance, solving the key throughput bottleneck of blockchain-based FL. Third, the MOO-based adaptive gradient clipping mechanism (Ablation 3: *w/o* MOO-Clipping) is critical for balancing privacy and accuracy. Replacing such with fixed clipping leads to a 3.9% drop in test accuracy (to 88.5%) and a 7.9% decrease in privacy compliance rate (to 92.1%), as fixed norms fail to adapt to gradient distribution

changes across training stages (causing either over-protection or under-protection). Collectively, the three components synergistically optimize trust, efficiency, and privacy-accuracy trade-off, forming the foundation of BE-DPFL's superior performance.

5. Conclusions and Future Work

In the era of deep integration between digital transformation and artificial intelligence, cross-institutional collaborative modeling faces dual constraints: "data silos" and strict privacy regulations. Although FL mitigates the risk of raw data leakage through the paradigm of "keeping data local while moving the model," traditional FL architectures still suffer from three fundamental bottlenecks: lack of trust mechanisms, insufficient privacy protection, and weak system robustness. Existing enhancements—such as DP, blockchain consensus, and robust aggregation—have achieved partial progress, yet common limitations remain. First, reliance on centralized privacy budget management introduces new trust bottlenecks. Second, traditional blockchain consensus mechanisms incur high communication overhead and limited throughput, making them incompatible with the high-frequency iterative nature of FL. Third, many studies focus solely on specific disturbance scenarios and fail to jointly optimize trust, efficiency, and privacy.

To address these challenges, this paper proposes BE-DPFL, a Blockchain-Enhanced Adaptive Differentially Private FL framework that builds a collaborative three-layer architecture—blockchain trust layer, FL core layer, and privacy enhancement layer—through a hybrid design of "on-chain trusted supervision + off-chain efficient training". The framework incorporates three major innovations. First, a lightweight FL-PBFT consensus mechanism and modular smart contracts (identity management, privacy budget allocation, update verification, and incentive–penalty mechanisms) are designed to ensure traceable privacy budget distribution and compliant client behavior auditing. The phased consensus strategy—"full consensus for critical information + lightweight validation for regular information"—significantly reduces latency and eliminates centralized trust risks. Second, a Random Projection–ADMM collaborative optimization mechanism is introduced, where high-dimensional gradients are projected into a low-dimensional space for off-chain distributed aggregation, and ADMM's parallel optimization capabilities substantially reduce communication and storage overhead, overcoming the performance bottlenecks of blockchain-based FL. Third, a MOO–based adaptive gradient clipping and noise injection strategy is constructed, using "satisfying (ε, δ)-DP constraints" and "maximizing model accuracy" as dual objectives. This enables dynamic adjustment of clipping norms and noise intensity, effectively addressing the poor adaptability of fixed privacy strategies. Under PyTorch 2.0 and Hyperledger Fabric 2.5, extensive experiments on CIFAR-10 and ChestX-ray14, benchmarked against eight mainstream FL and DP-FL baselines, demonstrate that BE-DPFL achieves significant improvements in consensus performance, privacy–accuracy trade-off, and robustness. Consensus latency is reduced to 78.2–82.5 ms per round, and communication overhead decreases by over 97% compared to baselines. Under a privacy budget of $\varepsilon = 5$, accuracy degradation is limited to 2.94–4.31%, with 100% privacy compliance. Even with 30% malicious clients and a 15% packet loss rate, BE-DPFL maintains high stability and strong anomaly-detection performance. Ablation studies further confirm that the blockchain trust layer reduces consensus latency by 63.7% and improves privacy compliance rate by 17.2%, the Random-Projection-ADMM cuts communication overhead by over 97%, and the MOO-based clipping reduces accuracy loss by 3.9% compared to fixed strategies, verifying the necessity of each component. Regarding scalability, we conducted preliminary tests with $K = 50/100$ clients (CIFAR-10, ResNet-18). Results show that consensus latency increases linearly (120 ms/180 ms per round) instead of exponentially, and communication overhead remains $\leq$20 MB/round, indicating scalability to larger

client pools. For large models (e.g., Vision Transformers with $100M+$ parameters), the Random-Projection-ADMM mechanism still reduces parameter dimensionality by $10^4\times$, but the interaction between privacy noise and high-dimensional feature extraction requires further optimization. Future work will explore adaptive projection dimensions for large models and integrate gradient compression with DP to maintain efficiency.

In summary, BE-DPFL effectively addresses key challenges in trust, efficiency, and privacy through its multi-layer collaborative design, offering a practical technical solution for cross-institutional collaborative modeling in highly sensitive, resource-constrained domains such as healthcare and industrial IoT. It also fills critical research gaps in blockchain-based FL regarding "trust–efficiency balance" and "dynamic privacy–accuracy optimization." Limitations of this work include: (1) The FL-PBFT consensus still relies on a fixed number of validation nodes, which may limit scalability to $K > 100$ clients; (2) The random projection dimension is manually set for specific models, lacking adaptive adjustment for arbitrary model sizes; (3) The current framework focuses on image data, and its performance on tabular or text data requires further verification. Future research will focus on addressing the identified limitations and expanding the framework's applicability by designing a dynamic validation node selection mechanism for large-scale client pools, developing an adaptive projection dimension algorithm based on model parameter sizes, extending the privacy enhancement layer to support multimodal data (e.g., text and tabular data) with federated transfer learning for cross-domain adaptation, exploring resource-adaptive strategies for heterogeneous environments spanning end devices, edge nodes, and cloud servers, and investigating the in-depth integration of DP with homomorphic encryption to further strengthen privacy protection and expand applicability to high-security fields such as financial risk control.

Author Contributions: W.J.: Writing—Original Draft, Writing—Review and Editing, Data Curation, Formal analysis; T.X.: Writing—Original Draft, Writing—Review and Editing, Methodology, Software. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: During the manuscript's writing stage, an artificial-intelligence-assisted tool (ChatGPT-4) was used for language polishing to improve clarity and readability. The tool was not involved in the research design, experiments, or result analysis, and all scholarly content of this paper was completed by the authors, who take full responsibility for it.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhang, C.; Xie, Y.; Bai, H.; Yu, B.; Li, W.; Gao, Y. A survey on federated learning. *Knowl.-Based Syst.* **2021**, *216*, 106775. [CrossRef]
2. De Bruijne, M. Machine learning approaches in medical image analysis: From detection to diagnosis. *Med. Image Anal.* **2016**, *33*, 94–97. [CrossRef] [PubMed]
3. Fu, C.; Yu, C.; Guo, M.; Zhang, L. ESG rating and financial risk of mining industry companies. *Resour. Policy* **2024**, *88*, 104308. [CrossRef]
4. Malik, P.K.; Sharma, R.; Singh, R.; Gehlot, A.; Satapathy, S.C.; Alnumay, W.S.; Pelusi, D.; Ghosh, U.; Nayak, J. Industrial Internet of Things and its applications in industry 4.0: State of the art. *Comput. Commun.* **2021**, *166*, 125–139. [CrossRef]
5. Zhang, C.; Yang, S.; Mao, L.; Ning, H. Anomaly detection and defense techniques in federated learning: A comprehensive review. *Artif. Intell. Rev.* **2024**, *57*, 150. [CrossRef]

6. El Ouadrhiri, A.; Abdelhadi, A. Differential privacy for deep and federated learning: A survey. *IEEE Access* **2022**, *10*, 22359–22380. [CrossRef]
7. Wei, K.; Li, J.; Ma, C.; Ding, M.; Chen, C.; Jin, S.; Han, Z.; Poor, H.V. Low-latency federated learning over wireless channels with differential privacy. *IEEE J. Sel. Areas Commun.* **2021**, *40*, 290–307. [CrossRef]
8. Guo, H.; Yu, X. A survey on blockchain technology and its security. *Blockchain Res. Appl.* **2022**, *3*, 100067.
9. Zhu, J.; Cao, J.; Saxena, D.; Jiang, S.; Ferradi, H. Blockchain-empowered federated learning: Challenges, solutions, and future directions. *ACM Comput. Surv.* **2023**, *55*, 1–31. [CrossRef]
10. Qammar, A.; Karim, A.; Ning, H.; Ding, J. Securing federated learning with blockchain: A systematic literature review. *Artif. Intell. Rev.* **2023**, *56*, 3951–3985.
11. Wei, K.; Li, J.; Ding, M.; Ma, C.; Yang, H.H.; Farokhi, F.; Jin, S.; Quek, T.Q.S.; Poor, H.V. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3454–3469. [CrossRef]
12. Truex, S.; Liu, L.; Chow, K.H.; Gursoy, M.E.; Wei, W. LDP-Fed: Federated learning with local differential privacy. In Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking, Crete, Greece, 27 April 2020; pp. 61–66.
13. Hu, R.; Guo, Y.; Li, H.; Pei, Q.; Gong, Y. Personalized federated learning with differential privacy. *IEEE Internet Things J.* **2020**, *7*, 9530–9539. [CrossRef]
14. Wu, X.; Zhang, Y.; Shi, M.; Li, P.; Li, R.; Xiong, N.N. An adaptive federated learning scheme with differential privacy preserving. *Future Gener. Comput. Syst.* **2022**, *127*, 362–372. [CrossRef]
15. Yang, X.; Huang, W.; Ye, M. Dynamic personalized federated learning with adaptive differential privacy. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 72181–72192.
16. Ling, J.; Zheng, J.; Chen, J. Efficient federated learning privacy preservation method with heterogeneous differential privacy. *Comput. Secur.* **2024**, *139*, 103715. [CrossRef]
17. Liu, X.; Wang, J.; Xiong, X.; Sun, H. Federated learning data protection scheme based on personalized differential privacy in psychological evaluation. *Neurocomputing* **2025**, *611*, 128653. [CrossRef]
18. Ma, C.; Li, J.; Zhou, Y.; Ding, M.; Ni, Y.; Jin, S. Differential privacy for multi-modal federated learning with modality selection. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 8788–8801. [CrossRef]
19. Hu, J.; Du, J.; Wang, Z.; Pang, X.; Zhou, Y.; Sun, P.; Ren, K. Does differential privacy really protect federated learning from gradient leakage attacks? *IEEE Trans. Mob. Comput.* **2024**, *23*, 12635–12649. [CrossRef]
20. Gad, G.; Gad, E.; Fadlullah, Z.M.; Fouda, M.M.; Kato, N. Communication-efficient and privacy-preserving federated learning via joint knowledge distillation and differential privacy in bandwidth-constrained networks. *IEEE Trans. Veh. Technol.* **2024**, *73*, 17586–17601. [CrossRef]
21. Lu, Y.; Huang, X.; Dai, Y.; Maharjan, S.; Zhang, Y. Blockchain and federated learning for privacy-preserved data sharing in industrial IoT. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4177–4186. [CrossRef]
22. Jia, B.; Zhang, X.; Liu, J.; Zhang, Y.; Huang, K.; Liang, Y. Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT. *IEEE Trans. Ind. Inform.* **2021**, *18*, 4049–4058. [CrossRef]
23. Chen, Y.; Li, J.; Wang, F.; Yue, K.; Li, Y.; Xing, B.; Zhang, L.; Chen, L. DS2PM: A data-sharing privacy protection model based on blockchain and federated learning. *IEEE Internet Things J.* **2021**, *10*, 12112–12125. [CrossRef]
24. Javed, L.; Anjum, A.; Yakubu, B.M.; Iqbal, M.; Moqurrab, S.A.; Srivastava, G. ShareChain: Blockchain-enabled model for sharing patient data using federated learning and differential privacy. *Expert Syst.* **2023**, *40*, e13131. [CrossRef]
25. Pauu, K.T.; Wu, J.; Fan, Y.; Pan, Q. Differential privacy and blockchain-empowered decentralized graph federated learning-enabled UAVs for disaster response. *IEEE Internet Things J.* **2023**, *11*, 20930–20947. [CrossRef]
26. Hussain, A.; Akbar, W.; Hussain, T.; Bashir, A.K.; Al Dabel, M.M.; Ali, F.; Yang, B. Ensuring zero trust IoT data privacy: Differential privacy in blockchain using federated learning. *IEEE Trans. Consum. Electron.* **2024**, *71*, 1167–1179. [CrossRef]
27. De Chaudhury, S.; Morreddigari, L.R.; Varun, M.; Sengupta, T.; Chakraborty, S.; Sural, S.; Vaidya, J.; Atluri, V. Blockchain based secure federated learning with local differential privacy and incentivization. *IEEE Trans. Priv.* **2024**, *1*, 31–44. [CrossRef]
28. Zhang, Y.; Luo, Y.; Chen, X.; Tong, F.; Xu, Y.; Tao, J.; Cheng, G. A Lightweight Authentication Scheme Based on Consortium Blockchain for Cross-Domain IoT. *Secur. Commun. Netw.* **2022**, *2022*, 9686049. [CrossRef]
29. Zhao, Y.; Chen, J. A survey on differential privacy for unstructured data content. *ACM Comput. Surv. (CSUR)* **2022**, *54*, 1–28. [CrossRef]
30. Sun, K.; Sun, X.A. A two-level ADMM algorithm for AC OPF with global convergence guarantees. *IEEE Trans. Power Syst.* **2021**, *36*, 5271–5281. [CrossRef]
31. Luo, H. ULS-PBFT: An ultra-low storage overhead PBFT consensus for blockchain. *Blockchain Res. Appl.* **2023**, *4*, 100155. [CrossRef]
32. Kasyap, H.; Tripathy, S. Beyond data poisoning in federated learning. *Expert Syst. Appl.* **2024**, *235*, 121192. [CrossRef]
33. Magnússon, S.; Weeraddana, P.C.; Rabbat, M.G.; Fischione, C. On the convergence of alternating direction lagrangian methods for nonconvex structured optimization problems. *IEEE Trans. Control Netw. Syst.* **2015**, *3*, 296–309. [CrossRef]

34. Xiao, H.; Xiang, Z.; Wang, D.; Devadas, S. A theory to instruct differentially-private learning via clipping bias reduction. In *2023 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 22–24 May 2023*; IEEE: New York, NY, USA, 2023; pp. 2170–2189.
35. Liu, S.; Vicente, L.N. The stochastic multi-gradient algorithm for multi-objective optimization and its application to supervised machine learning. *Ann. Oper. Res.* **2024**, *339*, 1119–1148. [CrossRef]
36. Thakkar, P.; Nathan, S.; Viswanathan, B. Performance benchmarking and optimizing hyperledger fabric blockchain platform. In Proceedings of the 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Milwaukee, WI, USA, 25–28 September 2018; IEEE: New York, NY, USA, 2018; pp. 264–276.
37. Chehade, A.H.; Abdallah, N.; Marion, J.M.; Hatt, M.; Oueidat, M.; Chauvet, P. Advancing chest X-ray diagnostics: A novel CycleGAN-based preprocessing approach for enhanced lung disease classification in ChestX-Ray14. *Comput. Methods Programs Biomed.* **2025**, *259*, 108518. [CrossRef]
38. Krizhevsky, A.; Hinton, G. Convolutional deep belief networks on cifar-10. *Unpublish Manuscript* **2010**, *40*, 1–9.
39. Chhabra, M.; Kumar, R. A smart healthcare system based on classifier DenseNet 121 model to detect multiple diseases. In *Mobile Radio Communications and 5G Networks: Proceedings of Second MRCN 2021*; Springer Nature: Singapore, 2021; Volume 2022, pp. 297–312.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.