Modeling and Simulation 建模与仿真, 2025, 14(5), 458-471
Published Online May 2025 in Hans. https://www.hanspub.org/journal/mos
https://doi.org/10.12677/mos.2025.145407

Hans汉斯

BCS-FL：基于区块链的工业物联网隐私保护联邦学习框架

王新中

上海理工大学光电信息与计算机工程学院，上海

收稿日期：2025年4月19日；录用日期：2025年5月12日；发布日期：2025年5月19日

摘　要

随着工业物联网的快速发展，在分布式环境下高效训练机器学习模型，同时保障数据隐私与系统安全，已成为亟待解决的关键问题。传统联邦学习虽然在一定程度上缓解了数据泄露风险，但仍面临中心化服务器易受攻击、投毒威胁以及模型更新透明度不足等挑战。为此，文章提出了一种基于区块链的去中心化联邦学习框架(BCS-FL)。该框架结合区块链技术与权益证明(PoS)共识机制，实现联邦学习的去中心化训练、筛选及存储，从而提升系统的安全性、透明度和可靠性。具体而言，在本地设备层，各智能工厂(客户端)基于自身数据训练局部模型，并经过裁剪和差分隐私保护后，生成局部模型包上传至验证层；在去中心化验证层，由PoS选出的验证节点计算局部模型间的余弦相似度以及损失函数差值，以筛选可信、高质量的客户端模型；在区块链存储层，矿工节点验证候选全局模型的哈希值、时间戳以及客户端签名，以确保模型版本的一致性，并通过PBFT共识将其存储至区块链，供所有客户端下载与同步，确保模型更新的可追溯性和抗篡改能力。

关键词

联邦学习，工业物联网，余弦相似度，区块链

BSC-FL: Blockchain-Based Privacy Protection Federated Learning Framework for Industrial Internet of Things

Xinzhong Wang

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Apr. 19th, 2025; accepted: May 12th, 2025; published: May 19th, 2025

文章引用: 王新中. BCS-FL: 基于区块链的工业物联网隐私保护联邦学习框架[J]. 建模与仿真, 2025, 14(5): 458-471.
DOI: 10.12677/mos.2025.145407

Abstract

With the rapid development of the Industrial Internet of Things, efficiently training machine learning models in distributed environments while ensuring data privacy and system security has become a critical challenge. Although traditional federated learning mitigates the risk of data leakage to some extent, it still faces challenges such as the vulnerability of centralized servers to attacks, poisoning threats, and insufficient transparency in model updates. To address these issues, this paper proposes a blockchain-based, decentralized, federated learning framework (BCS-FL). By integrating blockchain technology with the Proof-of-Stake (PoS) consensus mechanism, BCS-FL enables decentralized training, selection, and storage of federated learning models, thereby enhancing system security, transparency, and reliability. Specifically, at the local device layer, intelligent factories (clients) train local models based on their own data and generate local model packages after pruning and differential privacy protection before uploading them to the coordination layer. At the decentralized coordination layer, coordination nodes selected through PoS compute the cosine similarity and loss function differences between local models to filter out reliable and high-quality client models. At the blockchain storage layer, miner nodes verify the hash values, timestamps, and client signatures of candidate global models to ensure model version consistency. The final global model is stored on the blockchain through PBFT consensus, allowing all clients to download and synchronize it, ensuring model update traceability and tamper resistance.

Keywords

Federated Learning, Industrial Internet of Things, Cosine Similarity, Blockchain

1. 引言

随着工业物联网(Industrial Internet of Things, IIoT)的快速发展，智能制造、工业自动化和数据驱动决策正成为现代工业体系的重要组成部分。IIoT 通过部署大量传感器、控制器和智能设备，结合大数据分析、云计算及人工智能技术，实现对工业生产全流程的感知、互联、优化与智能控制。然而，IIoT 系统在带来生产效率提升的同时，也暴露出严重的数据安全与隐私风险[1]。工业数据通常涉及企业核心机密、生产工艺参数及敏感业务信息，一旦泄露，可能导致巨大的经济损失及安全隐患。因此，如何在 IIoT 场景下有效保护数据隐私，同时保障数据价值的充分利用，已成为亟待解决的关键问题[2]。

近年来，联邦学习(Federated Learning, FL) [3]作为一种新兴的分布式机器学习范式，为 IIoT 数据隐私保护提供了一种潜在的解决方案。联邦学习允许多个客户端(如智能工厂)在不共享原始数据的前提下，共同训练全局模型。具体而言，各客户端利用本地数据训练模型，并仅上传模型参数至服务器进行聚合，从而避免数据直接暴露。然而，传统联邦学习框架仍然面临诸多挑战。首先，中心化架构使得服务器成为潜在的安全瓶颈，易受攻击或篡改；其次，数据质量差异问题未被充分考虑，现有方法通常采用加权平均策略进行模型聚合，而未能有效区分数据质量高低，可能导致低质量数据对全局模型的负面影响。此外，联邦学习面临投毒攻击风险，即恶意客户端可通过上传异常梯度干扰全局训练，影响模型收敛或降低其性能。为应对上述挑战，研究者提出了多种隐私保护技术，如 k-匿名(k-Anonymity) [4]、l-多样性

(l-Diversity) [5]、t-相似性(t-Closeness) [6]以及差分隐私(Differential Privacy) [7]。其中，差分隐私成为联邦学习领域最广泛应用的隐私保护方法之一[8]。差分隐私通过在梯度上传前引入噪声，使攻击者难以推断单个客户端的真实数据，从而提升系统的隐私保护能力。

针对上述问题，本文提出了一种基于区块链的去中心化联邦学习框架(Blockchain-Based Collaborative Secure Federated Learning, BCS-FL)，旨在提高模型训练的安全性、可靠性及透明度。BCS-FL 采用区块链技术取代传统联邦学习中的中心服务器，利用权益证明(Proof of Stake, PoS)共识机制和拜占庭容错(Practical Byzantine Fault Tolerance, PBFT)实现去中心化的节点选举与共识机制，并结合差分隐私和余弦相似度筛选以提升数据安全性，从而实现去中心化、抗投毒、透明可追溯的联邦学习过程。具体而言，BCS-FL 主要包括以下关键步骤：

1) 本地训练与上传。智能工厂在本地基于自身数据训练模型，更新加入差分隐私保护后的局部模型，并对更新后的模型计算哈希值、记录时间戳等，并使用私钥进行签名上传至去中心化验证层。

2) 去中心化筛选与模型选择。由 PoS 选出的验证节点计算局部模型间的余弦相似度，以筛选异常客户端，并根据损失函数差值选定最优局部模型作为全局模型，从而降低投毒攻击的影响。验证节点之间通过 PBFT 共识确定最终的全局模型，若无法达成 2/3 以上的共识，则回滚至上一轮的全局模型。

3) 区块链存储与更新。PoS 选出的矿工节点对候选全局模型进行验证，计算其哈希值、时间戳和客户端签名，确保模型版本的一致性。使用 PBFT 共识机制，若 2/3 以上矿工节点投票通过，则将其存入区块链，供所有客户端下载与同步，以确保模型更新的可追溯性和抗篡改能力。

为了验证 BCS-FL 的有效性，本文在 MNIST 和 CIFAR-10 数据集上进行了实验评估。实验结果表明，该框架在分类精度及隐私保护方面均优于传统联邦学习方法，同时能够有效防御投毒攻击，提高全局模型的安全性与可信度。

2. 相关工作

近年来，针对工业物联网(IIoT)中的联邦学习模型聚合方法，研究者们提出了多种应用框架，以提升其安全性、隐私保护能力及资源利用效率。Basu 等人[9]详细探讨了联邦学习在 IIoT 领域中的潜力，分析了其在信息物理系统(CPS)中的应用机遇、关键挑战及潜在解决方案，并展望了其未来发展方向。然而，该研究在联邦学习的具体应用场景及发展路径方面缺乏深入探讨，未充分考虑 IIoT 复杂环境下的实际部署问题。Boobalan 等人[10]进一步从隐私保护、资源分配和数据管理等多个维度总结了联邦学习与 IIoT 的集成方式，阐述了二者结合的动机，并探讨了如何在 IIoT 框架下融合联邦学习、机器学习和深度学习，以构建更具鲁棒性和适应性的安全机制。此外，Hao 等人[11]针对 IIoT 场景提出了一种联邦学习安全隐私保护方案，结合人工智能技术增强系统的安全性，从而有效降低数据泄露与攻击风险。这些研究为 IIoT 中联邦学习的实际应用提供了理论基础，但仍需进一步探索如何优化模型聚合策略、提高通信效率以及增强系统可扩展性，以推动联邦学习在 IIoT 领域的广泛落地。

3. 预备知识

3.1. PBFT 共识

拜占庭容错(PBFT)是一种能够在存在恶意或故障节点的情况下仍然保证系统正常运行的共识机制。PBFT 共识的核心思想是：在 n 个节点中，最多允许 f 个节点作恶，只要系统满足 $n \geq 3f+1$，即超过 2/3 的节点仍然诚实，整个系统就能保持正确运行。在 BCS-FL 中，PBFT 共识主要应用于以下两个环节：

1) 全局模型选择：PoS 选出的 M 个验证节点负责计算局部模型之间的余弦相似度，并筛选出异常客户端。此后，M 个验证节点依据损失函数的差值，选定最优的局部模型 w_{t+1} 作为新的全局模型。该选择

需要经过 PBFT 共识机制的投票，即若≥2/3 的节点同意该选择，则确认 w_{t+1} 为新一轮的全局模型；否则系统将回滚至上一轮全局模型 w_t，以防止恶意节点干扰导致全局模型异常。

2) 模型存储与更新：PoS 选出的 K 个矿工节点对 w_{t+1} 进行哈希验证，并通过 PBFT 共识机制进行最终确认。若≥2/3 的矿工节点投票通过，则 w_{t+1} 被存入区块链，并供所有客户端下载和同步。这一过程确保了联邦学习模型更新的可追溯性和抗篡改能力。

3.2. 直接加权随机抽样的 PoS 选举方法

权益证明(PoS)是一种基于权益权重进行节点选举的共识机制，相较于工作量证明(Proof of Work, PoW)，PoS 具有能耗少、计算成本低、共识效率高等优势。在 BCS-FL 框架中，PoS 机制用于选举验证节点和矿工节点，以确保去中心化联邦学习的安全性和稳定性。本文采用直接加权随机抽样(Direct Weighted Random Sampling, DWRS)进行 PoS 选举，即每个候选节点的当选概率与其所持有的权益成正比。

具体而言，假设系统中共有 K 个候选节点，第 i 个节点的权益值为 S_i，则其被选中的概率 P_i 的计算如公式(1)所示：

$$P_i = \frac{S_i}{\sum_{j=1}^{K} S_j} \tag{1}$$

PoS 选举过程包括以下关键步骤：

1) 权益分配：每个候选节点根据 S_i 确定其选举权重，系统维护一个全局权益池 $S_{\text{total}} = \sum_{j=1}^{K} S_j$ 以供抽样参考。

2) 随机抽样：系统依据 P_i 采用加权随机抽样方法，从 K 个候选节点中选出 M 个验证节点和 N 个矿工节点，确保选举结果的随机性和公平性。

3) 共识达成：验证节点在完成局部模型筛选与全局模型选择后，通过拜占庭容错(PBFT)共识机制进行验证，确保 2/3 以上的验证节点同意最终全局模型的选择。若无法达成共识，则回滚至上一轮全局模型。

4) 区块链存储：矿工节点对候选全局模型进行哈希验证，并利用 PBFT 共识确认 2/3 以上矿工节点投票通过后，存入区块链，以确保模型更新的可追溯性和抗篡改能力。

3.3. 余弦相似度

余弦相似度(Cosine Similarity)是一种常用于衡量两个向量间相似度的度量方法，特别适用于高维空间的特征比较。其核心思想是通过计算两个向量夹角的余弦值，来衡量它们的方向是否一致，而不受向量的绝对大小影响。在 BCS-FL 框架中，余弦相似度用于评估不同客户端提交的局部模型更新方向的一致性，以筛选异常模型并保证联邦学习的稳定性。设两个 d 维向量 X 和 Y，其余弦相似度计算公式如公式(2)所示：

$$\cos(\theta) = \frac{X \cdot Y}{\|X\|\|Y\|} \tag{2}$$

其中：$X \cdot Y$ 表示 X 与 Y 的点积；$\|X\|$ 和 $\|Y\|$ 分别表示向量 X 与 Y 的欧几里得范数(L_2 范数)；计算结果范围在$[-1, 1]$之间，值越接近 1 说明两个向量方向越相似，越接近−1 说明方向相反。在 BCS-FL 评估阶段，验证节点计算所有客户端提交的局部模型更新间的余弦相似度，并设定阈值 τ，用于划分可信客户端集群和异常客户端集群。若某客户端提交的更新与大多数客户端方向偏离过大，则可能是受到攻击或存在异

常，该客户端的模型更新将被拒绝，以提高全局模型的鲁棒性和安全性。

4. 基于区块链的去中心化联邦学习框架

本节介绍一种基于区块链的去中心化联邦学习框架(BCS-FL)，该框架利用区块链技术取代传统联邦学习中的中心服务器，并采用权益证明(Proof of Stake, PoS)机制进行验证节点选举与模型筛选，以提升系统的安全性和鲁棒性。在 BCS-FL 框架中，联邦学习过程被划分为本地设备层、去中心化验证层和区块链存储层三个部分。本节将详细介绍 BCS-FL 框架的安全模型、系统架构及关键工作流程。

4.1. 安全模型设定

在 BCS-FL 框架中，假设所有参与方(如智能工厂、PoS 选出的验证节点及矿工节点)为半诚实(Semi-honest)实体[12]。这种假设在去中心化联邦学习研究中较为常见[13]，并具有以下特性：

1) 协议遵循性。所有客户端和验证节点均按照协议规定的步骤执行联邦学习任务，不会主动偏离或中途退出。

2) 数据完整性。所有参与方不会恶意篡改本地输入数据或计算过程中生成的中间结果。

3) 隐私窥探风险。尽管参与方遵循协议执行，但可能会尝试分析所接触到的中间数据，以推测其他客户端的隐私信息。

此外，本文假设所有智能工厂均配备充足的计算资源和通信能力，能够独立完成本地训练任务，并拥有对自身设备的完全控制权限。

4.2. 模型数据包

模型数据包是 BCS-FL 框架中用于传输和存储模型更新信息的基本单位，各客户端在本地完成模型训练后，会将训练结果封装为数据包并上传至去中心化验证层。数据包由模型信息、哈希值、时间戳、数据包序号、客户端标识和签名组成。其具体含义如下：

1) 数据包序号(Packet ID)：为每个数据包分配唯一的序号，以便在联邦学习过程中管理和追踪不同轮次的模型更新。

2) 模型信息：包含客户端在本地训练得到的模型参数，用于后续的筛选和全局更新。

3) 哈希值：模型数据的唯一标识，用于验证数据包的完整性和防篡改能力。

4) 时间戳(Timestamp)：标记数据包的生成时间，确保模型更新的时序一致性。

5) 客户端标识(Client ID)：表示数据包的提交者，便于在共识过程中进行身份验证。

6) 签名(Signature)：客户端使用私钥对当前数据包的哈希值进行签名，防止数据在网络传输过程中被篡改，同时供其他节点验证数据的真实性。

4.3. 共识步骤

在 BCS-FL 框架中，每个客户端在完成本地训练后，需要对更新后的模型进行哈希计算、时间戳记录和签名，以确保数据的完整性、可追溯性和安全性。其关键共识步骤如下：

1) 计算哈希值。每个训练客户端 k 计算本地更新模型 $\tilde{w}_{kn}$ 的哈希值，以确保数据在传输和存储过程中不被篡改，如公式(3)所示：

$$H\left(\tilde{w}_{kn}\right)=Hash\left(\tilde{w}_{kn}\right) \tag{3}$$

2) 记录时间戳。生成当前时间戳 $T\left(\tilde{w}_{kn}\right)$ 以标记提交时间，确保模型更新的时序可验证，如公式(4)所示：

$$T\left(\tilde{w}_{kn}\right)=Timestamp\left(current\ time\right) \tag{4}$$

3) 使用私钥对哈希值进行签名。客户端使用其私钥对哈希值进行签名，以验证数据的来源及完整性，如公式(5)所示：

$$Sign\left(H\left(\tilde{w}_{kn}\right)\right)=Encrypt\left(H\left(\tilde{w}_{kn}\right),PrivateKey_k\right) \tag{5}$$

4) 训练客户端将模型更新及相关信息封装为数据包$\left\{\tilde{w}_{kn},H\left(\tilde{w}_{kn}\right),T\left(\tilde{w}_{kn}\right),ID,Sign\left(H\left(\tilde{w}_{kn}\right)\right)\right\}$，提交至区块链以供筛选、评估和验证。

5) 每个客户端 k 在最初加入网络时，会注册自己的公钥 $PublicKey_k$。矿工节点使用公钥验证接收到的候选全局模型包签名的真实性，如公式(6)所示：

$$Verify\left(H\left(\tilde{w}_{kn}\right),Sign\left(H\left(\tilde{w}_{kn}\right)\right)\right),PublicKey_k \tag{6}$$

4.4. 安全通信与密钥管理

在 BCS-FL 框架中，客户端训练后的本地模型需要加密后提交，以确保数据安全性。为了保证矿工能够正确解密对应的模型，我们采用公私钥加密(PKI)机制，并使用客户端 ID 进行公钥映射。具体流程如下：

1) 密钥生成与存储。每个客户端 C_i 生成公私钥对$\left(pk_i,sk_i\right)$，其中私钥 sk_i 由客户端持有，并用于加密本地模型。公钥 pk_i 存储在区块链中，供矿工查询。

2) 模型加密与提交。客户端本地训练模型 W_i，然后使用私钥加密：$E_i=Enc\left(sk_i,W_i\right)$。客户端将加密后的模型 W_i 和自身的唯一标识 ID 一起提交。

3) 矿工解密与验证。矿工接收到加密模型 W_i 后，首先读取客户端 ID。矿工从区块链查询该 ID 对应的公钥 pk_i，然后使用公钥进行解密 $W_i=Dec\left(pk_i,E_i\right)$。成功解密后，矿工可以对模型进行验证，并存储到区块链中。

4.5. 系统架构

如图 1 所示，BCS-FL 框架由本地设备层、去中心化验证层和区块链存储层组成。本地设备层由多个智能工厂(联邦学习客户端)组成，每个智能工厂配备边缘计算设备或本地计算资源，负责本地模型训练、隐私保护及模型上传。去中心化验证层由 PoS 选出的验证节点组成，负责全局模型的同步、局部模型的筛选以及恶意客户端的筛选。采用余弦相似度筛选机制过滤异常客户端，确保聚合模型的可信性。区块链存储层由 PoS 共识机制选出的矿工节点负责全局模型的存储和验证。主要训练过程如下：

1) 同步阶段。在每一轮全局模型的迭代开始前，通过 PoS 机制从所有客户端中使用加权随机抽样的 PoS 选举方法，选择验证节点和矿工节点。PoS 选出的矿工节点从区块链获取最新的全局模型包，并将其广播给所有节点，确保训练开始时的模型一致性。

2) 训练阶段。每个客户端基于本地数据集训练模型，裁剪并加入差分隐私，生成更新后的局部模型包。并对更新后的模型进行处理，包括计算哈希值、记录时间戳，并使用私钥进行签名，以保证数据的完整性和可追溯性。随后，客户端将封装好的数据包上传至区块链网络。

3) 评估阶段。验证节点对所有客户端提交的局部模型进行评估，计算所有客户端提交的局部模型间的余弦相似度，衡量模型更新方向的一致性。根据设定的相似度阈值 τ，将客户端划分为可信客户端集群和异常客户端集群。对可信客户端集群根据损失之差 ΔE 进行筛选，选择最佳模型作为候选的全局模型 w_{t+1}。在此过程中，第一个完成计算的验证节点会先广播其计算结果，其他验证节点若尚未完成计算，则

先接受并存储该信息，待所有验证节点完成计算后开始 PBFT 共识，并投票选出 w_{t+1}，若无法达成 2/3 以上的共识，则回滚到上一轮的全局模型。

4) 验证阶段。矿工节点对候选的全局模型 w_{t+1} 进行验证，使用密钥进行解密，检查哈希值是否正确、时间戳是否一致。若 2/3 以上矿工节点投票通过，则进入存储阶段，否则回滚至上一轮模型并重新评估和训练。

5) 存储阶段。质押量最高的矿工负责将新的全局模型包存入区块链。其他矿工节点对存储过程进行验证，确保模型的正确性。所有客户端可以从区块链获取最新的全局模型，以便进行下一轮的训练。

在未达到迭代收敛或者最大迭代次数之前，重复执行上述五个阶段。

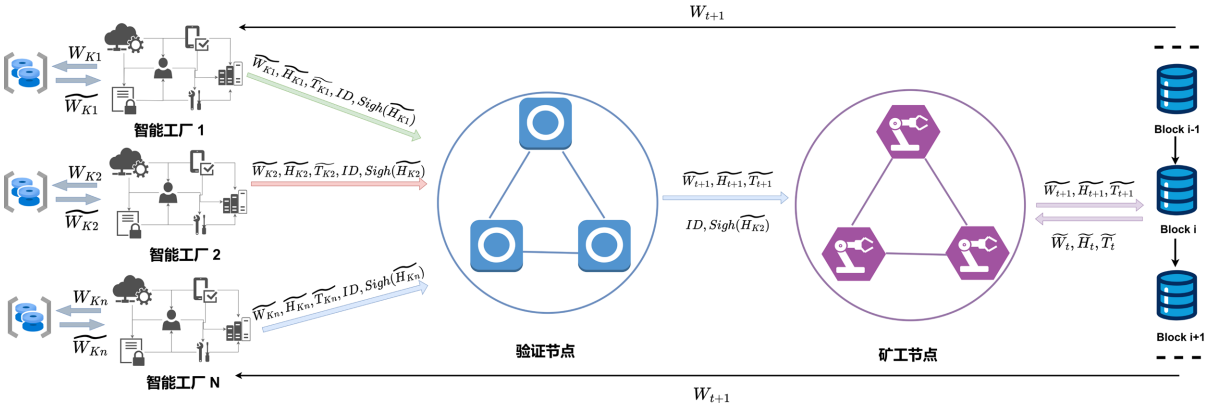

Figure 1. Framework diagram for information protection using BCS-FL in smart factories
图 1. 智能工厂中应用 BCS-FL 进行信息保护的框架图

4.6. BCS-FL 算法

本节介绍基于区块链的去中心化联邦学习(BCS-FL)训练过程，如算法 1 所示。

算法 1 基于区块链的去中心化联邦学习(BCS-FL)

输入：客户端总数量 K；初始模型参数 w_t；相似度筛选阈值 τ；本地训练轮次 E；学习率 lr；最大训练轮次 T。

输出：最终全局模型参数 w_{t+1}。

初始化全局模型参数 w_t, $t=0$
while $t < T$ *do*:
    for 客户端 k *in* $\{1,\cdots, K\}$?*do*:
        从 K 个候选节点中选出 M 个验证节点和 N 个矿工节点
        矿工节点从区块链获取最新的全局模型 w_t，并广播给所有客户端
        训练节点训练本地模型 $\tilde{w}_{kn} \leftarrow LocalUpdate(w_t, D_k, lr, E)$
        计算哈希值 $H(\tilde{w}_{kn})$
        记录时间戳 $T(\tilde{w}_{kn})$
        使用私钥生成签名 $Sign(H(\tilde{w}_{kn}))$
        封装数据包 $\{\tilde{w}_{kn}, H(\tilde{w}_{kn}), T(\tilde{w}_{kn}), ID, Sign(H(\tilde{w}_{kn}))\}$ 提交数据包至区块链
        验证节点计算所有客户端上传模型的余弦相似度矩阵
        根据相似度阈值 τ 划分 S_{major} 和 S_{minor}

续表

13	仅保留 S_{major} 作为可信客户端集群
14	在 S_{major} 根据损失函数差值 ΔE 选择最佳模型 w_{t+1}
15	第一个完成计算的验证节点广播计算结果，其他验证节点接受并存储该信息
16	进行 PBFT 共识，确认唯一 w_{t+1}，否则回滚至 w_t 并重新评估
17	矿工节点计算并验证 $Sign\left(H\left(w_{t+1}\right)\right), H\left(w_{t+1}\right), T\left(w_{t+1}\right)$ 是否有效
18	进行 PBFT 共识，确认 w_{t+1}，否则回滚至 w_t 并重新评估和训练
19	质押量最高的矿工将 $\left\{w_{t+1}, H\left(w_{t+1}\right), T\left(w_{t+1}\right), Sign\left(H\left(w_{t+1}\right)\right)\right\}$ 存入区块链
20	广播新全局模型 w_{t+1} 给所有客户端
21	$t = t + 1$
22	*return* w_{t+1}

5. 实验综述

为了评估基于区块链的去中心化联邦学习(BCS-FL)框架的性能，本节通过计算机仿真实验模拟 IIoT 场景下的智能工厂节点行为，并在两个真实数据集上进行实验分析。所有实验均采用 Python 的 PyTorch 框架实现，重点评估 BCS-FL 在抗投毒攻击能力及吞吐量和时延方面的表现。

5.1. 实验环境

实验配备在以下硬件配置的个人计算机上进行。其中详细信息为 CPU：13th Gen Intel (R) Core (TM) i7-14700 K、GPU：NVIDIA GeForce RTX 4080S、内存：32 GB RAM、操作系统：Windows 10、深度学习框架：PyTorch 2.0.1 + Python 3.11。为了模拟 IIoT 场景下的去中心化联邦学习过程，实验采用单机模拟多个智能工厂节点，并通过 Python 线程池控制客户端的并发行为。此外，实验选用 ResNet-50 作为训练模型，并在客户端侧使用随机梯度下降 SGD 算法进行局部模型更新。

5.2. 数据集

实验选取了两个广泛应用于工业物联网场景的图像分类数据集，用于评估 BCS-FL 在不同任务中的表现，包括手写数字数据集 MNIST 和图像识别数据集 CIFAR-10。在实验中，数据集被非独立同分布(Non-IID)分配给各客户端，以模拟 IIoT 设备环境下的异构数据分布。

MNIST 数据集：包含手写数字图像，可用于模拟工业生产中的质量检测和数字识别任务。该数据集用于评估 BCS-FL 在低复杂度分类任务下的表现。

CIFAR-10 数据集：包含多种彩色物体图像，可用于模拟工业领域中的目标检测和物体识别任务。该数据集用于评估 BCS-FL 在复杂场景下的泛化能力。

5.3. 系统设计

在区块链系统中，区块是承载数据的核心单元，负责存储链上所有的交易信息。每个区块的结构如图 2 所示，主要由区块头(Block Header)和交易数据(Transactions)两部分组成。区块头包含多个关键字段，包括当前区块的序号、前一区块的哈希值、地址、当前区块的哈希值、区块生成的时间戳以及区块生产者对当前区块哈希值的签名。其中地址指的是改区块生成者的地址，即该区块内包含的全局模型训练节点的公钥，用于解密其用私钥加密的签名，验证其内部信息是否被篡改，包含的这些信息确保了区块的完整性和链式连接的安全性。区块链中的数据存储依赖于这些区块的顺序连接，通过哈希值关联前后区

块，形成一条不可篡改的链式结构。这种设计不仅保证了交易记录的安全性和不可逆性，还确保了整个区块链系统的可信性和稳定性。

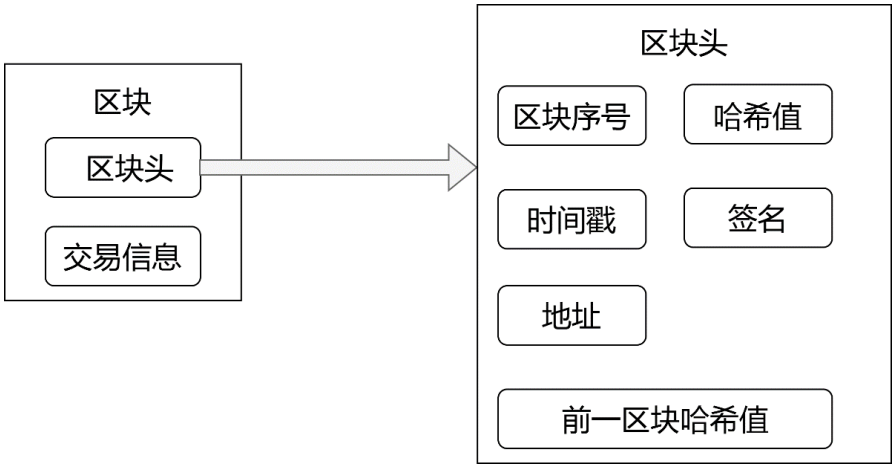

Figure 2. Block structured
图 2. 区块结构

整个系统主要可分为两部分，联邦学习部分以及区块链部分。联邦学习部分代码主要用 Python 编写，区块链部分主要用 JavaScript 编写。每个节点在创建后可提供如表 1 所示的接口。总计提供四个对外接口：1) /getBlockchain：通过该接口可获取区块链上存储的所有区块信息。2) /getLatestModel：通过该接口可获取区块链中最新的区块信息。3) /postModelUpdate：通过该接口可向对应节点提交模型数据包。

Table 1. Interface information
表 1. 接口信息

接口路径 API	解释
/getBlockchain	获取区块链上所有的区块信息
/getLatestModel	获取区块链最新的区块信息
/postModelUpdate	提交数据模型包

Table 2. Data item information
表 2. 数据项信息

数据项	解释
hash	区块的 hash 值
lastHash	上一个区块的 hash 值
timestamp	生成时间戳
data	区块内包含的信息
privace	公钥
sign	私钥对 hash 值签名后的结果
sequenceNo	区块序号

节点在生成本地个性化参数时，调用/getLatestModel 接口，获取最新区块信息，如图 3 所示。其中各

个数据项的说明如表 2 所示。主要有九个数据。hash 和 lastHash 分别表示本区块的哈希值和上一区块的哈希值，如果当前区块为创世区块，hash 值为 genesis-hash，lastHash 值为空。timestamp 表示区块生成的时间戳。data 里保存了区块包含的模型更新信息，在每一轮训练中，节点通过获取最新区块中所有模型更新信息里的模型参数和训练数据量等计算下一轮全局模型参数。public 表示区块生成者的公钥。sign 表示客户端用自身私钥对 hash 值签名后的结果。sequenceNo 表示当前区块序号。


```
{
    "hash": "46b8cdb3515a1512b95564f49dc7d8f65c6cf70faa77cbaa044a24f696372cb3",
    "lastHash": "472c04951571d1d9f1e13011a4439668643a16905bc86fa80d74e3ea37a6ff61",
    "timestamp": 1742799876420,
    "data": [
        {
            "id": "d9ad4001-7fd3-11eb-8d39-f7a7d91efaad",
            "from": "14ebfc6ea1286c8dc1201214cc0d065864f1e7d7e37e958826398c21ca29b386",
            "input": {
                "data": {
                    "model-params": "[...]",
                    "model-acc": 0.9108439764,
                    "num": 15346
                },
                "timestamp": 1742799871420
            },
            "hash": "4813f6262a1c61d5d3d7339a9d48512f782d985663465ddbe8d714962151d239",
            "sign": "F99496CDEF38A95331E65C4894468B813ECD886E5096444A72E636E6D9423AF3FB8DB2E41B7484FC1C588D437F73835D58A468B7481741FFF2A8EE259581E70D"
        },
        {
            "id": "0949d500-7fd4-11eb-8d39-f7a7d91efaad",
            "from": "14ebfc6ea1286c8dc1201214cc0d065864f1e7d7e37e958826398c21ca29b386",
            "input": {
                "data": {
                    "model-params": "[...]",
                    "model-acc": 0.90468137168,
                    "num": 15346
                },
                "timestamp": 1742799868420
            },
            "hash": "e52f805485cb9507268732a0ad4ad9d8218d2ab0e26239d674791a2ed64aa01c",
            "sign": "4F6208BF2F1578777C41992AC315C7E31C16E82CB07FAABE059C265F2DB6663D6DF957828E70DA8168551FBB7E7DEB301FB4339592ADFF42C07E45D7DE35D70A"
        }
    ],
    "public": "5d0173efd66bf38c1962f38ac79acb43035a47fc60f15b1e5195c8a03c2ef4c3",
    "sign": "88F4AE237836CBC12BB359BA50B97962206E88108F05131F11A99F7C919CE57FB9F22479E1A777E96D039CCC97836FD5DF2D8EDDFC0AC48EE069A393A06AE50B",
    "sequenceNo": 6
}
```


Figure 3. Block Information
图 3. 区块信息

5.4. 实验结果与分析

本实验基于 BCS-FL 框架，评估了余弦相似度筛选机制在实验环境下对抗投毒攻击的有效性。实验在 MNIST 和 CIFAR-10 数据集上进行，并针对不同的投毒攻击强度(10%、30%、50%恶意客户端)进行了系统性分析。为验证 BCS-FL 的效果，我们设置了无筛选(FedAvg 直接聚合)和筛选(BCS-FL 结合余弦相似度阈值)两种实验对比，并考察其对模型最终准确率和收敛稳定性的影响。实验结果如图 4(a)~(d)所示。

投毒攻击实验设置：为了模拟联邦学习环境下的投毒攻击，本实验引入了标签翻转攻击(Label Flipping Attack)作为常见的投毒攻击方式。在每轮训练过程中，恶意客户端会对其本地数据进行系统性篡改，具体操作如下：

1) MNIST 数据集：恶意客户端将部分类别的标签进行翻转，使模型学习到错误的类别映射。某些数字标签会被篡改为不同的错误类别，例如恶意客户端将数字“0”翻转为“9”，“1”翻转为“8”，以此类推，导致全局模型在测试时出现高误分类率。

2) CIFAR-10 数据集：恶意客户端将某些类别的标签进行错误映射，把某些动物类别的标签互换，或

将交通工具类别的标签进行混淆，例如恶意客户端将“猫”标签翻转为“狗”，或将“飞机”翻转为“汽车”，使得全局模型在测试时出现分类错误。

在无攻击(Baseline)环境下，实验采用 FedAvg 作为基准方法，即所有客户端的本地模型更新均被无差别地聚合，不进行任何筛选或异常检测。实验结果表明，在无攻击情况下，联邦学习能够充分学习数据特征，并在 MNIST 和 CIFAR-10 数据集上均达到较高的分类准确率。这表明在没有恶意客户端干扰的情况下，模型能够稳定收敛，并取得最优的训练效果。

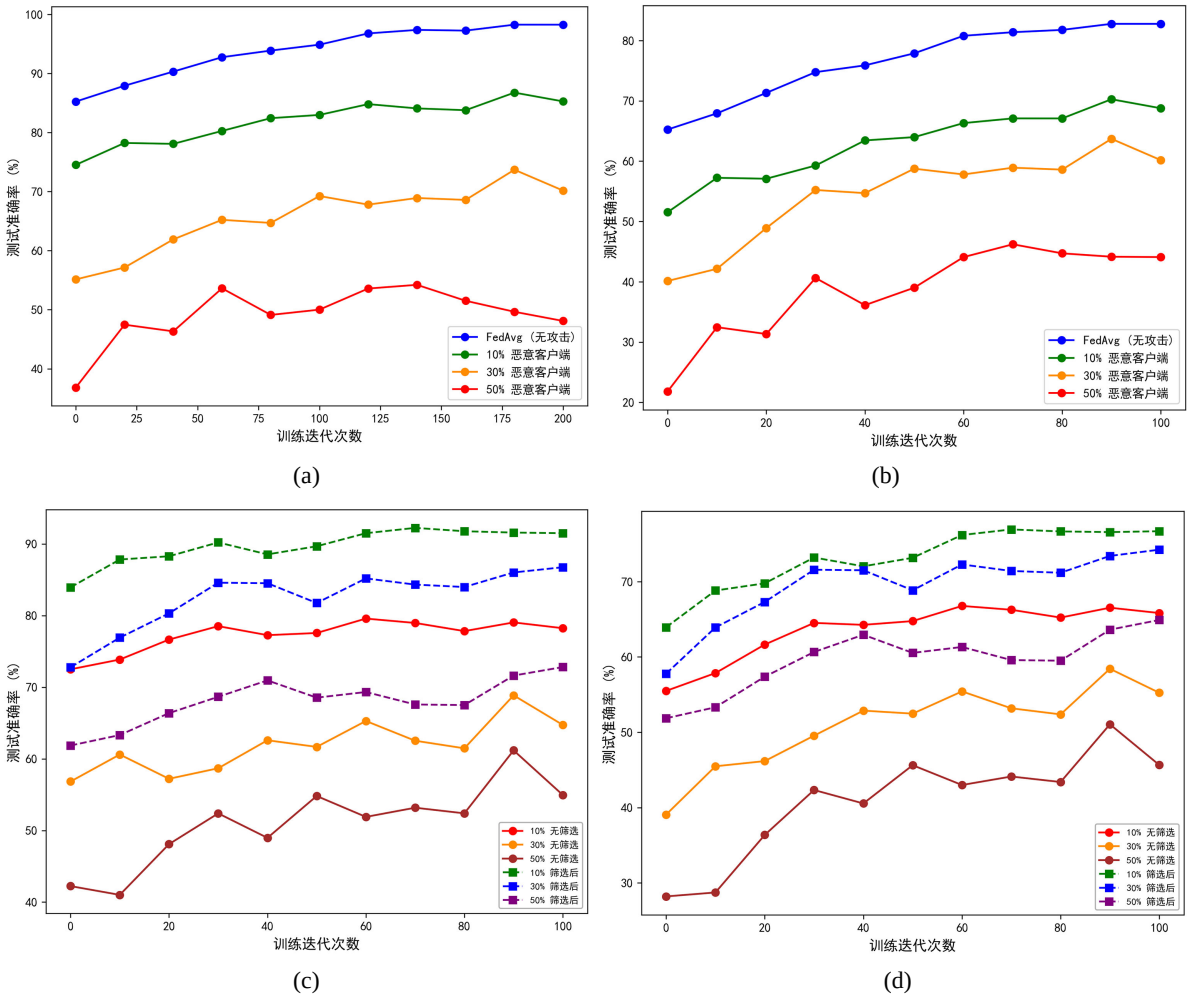

Figure 4. The impact of poisoning attacks on different schemes for MNIST and CIFAR-10 datasets
图 4. 对于 MNIST、CIFAR-10 数据集，投毒攻击在不同方案下的影响

如图 4(a)所示，在 MNIST 数据集上的实验结果表明，当恶意客户端比例较低(10%)时，模型仍能学习到有效的特征表示，尽管准确率略有下降，但整体收敛趋势保持稳定。当恶意客户端比例增加至 30%时，模型的准确率开始出现显著下降，并且在训练过程中出现更大幅度的收敛波动，表明攻击已开始干扰全局优化方向，部分错误更新影响了模型的梯度聚合效果。在 50%恶意客户端的情况下，模型准确率不仅急剧下降，甚至在训练后期出现持续恶化的趋势。这表明大量错误更新可能导致全局模型逐步朝着错误方向优化，破坏模型的泛化能力，最终使得模型在测试数据上的表现大幅下降。

如图 4(b)所示，在 CIFAR-10 数据集上的实验结果表明，当恶意客户端占比较低(10%)时，模型仍能

部分学习有效特征，最终准确率略有下降。相比 MNIST 任务，CIFAR-10 由于特征空间更复杂，即使在低比例的投毒攻击下，模型的泛化能力也会受到一定影响。当恶意客户端比例达到 30%时，模型的测试准确率出现更明显下降，并且训练过程中波动增大，表明攻击已开始干扰全局优化方向。在 50%恶意客户端的极端攻击环境下，全局模型的分类性能大幅下降，甚至在训练后期无法收敛到合理水平。这说明大量错误更新不仅影响了模型的优化方向，还可能导致模型学习到错误的特征表示，使其难以正确分类测试数据。特别是在 CIFAR-10 任务中，数据类别之间的相似度较高(如猫和狗、飞机和汽车)，投毒攻击使模型的决策边界变得混乱，从而导致分类错误率急剧上升。

如图 4(c)所示，在 MNIST 数据集上，引入 BCS-FL 结合余弦相似度筛选机制(阈值 = 0.7)。实验结果表明，BCS-FL 在不同攻击强度下均能有效提升模型的最终准确率，并减少因投毒攻击导致的训练波动。对于低攻击比例(10%)，筛选机制基本能够抵消恶意更新的影响，使模型恢复至接近无攻击情况下的性能，表明余弦相似度能够有效识别并过滤少量恶意更新。当恶意客户端比例达到 30%时，模型的测试准确率仍然有所下降，但相比无筛选时有明显提升，说明 BCS-FL 能够有效过滤异常更新，使模型更稳定地收敛。在极端攻击环境下(50%恶意客户端)，BCS-FL 仍然能够部分恢复模型性能，但最终模型的准确率仍然低于无攻击环境较多。

如图 4(d)所示，在 CIFAR-10 数据集上，引入 BCS-FL 结合余弦相似度筛选机制(阈值 = 0.5)。实验结果表明，该机制在不同攻击强度下均能显著提升模型的鲁棒性。对于低比例投毒攻击(10%)，BCS-FL 使模型的分类准确率几乎恢复至无攻击情况下的水平，证明该机制能有效识别并过滤少量恶意更新。当恶意客户端比例增加至 30%时，筛选机制仍能显著提升最终准确率，并减少训练过程中由于异常更新导致的不稳定性。即使在极端攻击条件下(50%恶意客户端)，BCS-FL 依然能够部分恢复模型性能，表明余弦相似度筛选可以有效削弱大规模投毒攻击的影响。

吞吐量是衡量共识机制性能的重要指标，通常以每秒可处理的交易数(TPS)表示。本文在网络节点数为 20，每个区块包含 50 笔交易的条件下，分别对 PBFT 和 BCS-FL 共识算法的吞吐量进行了测试，结果如图 5 所示。在相同实验条件下，PBFT 和 BCS-FL 在交易吞吐量(TPS)方面表现出不同的特性。PBFT 作为一种经典的拜占庭容错共识算法，广泛用于联盟链环境，以确保安全性和一致性，而 BCS-FL 结合了区块链技术与联邦学习，不仅保障了数据完整性，还提升了去中心化协作的可信度。实验结果表明，BCS-FL 的吞吐量约为 470 TPS，而 PBFT 的吞吐量约为 330 TPS，BCS-FL 相较于 PBFT 具有更高的吞吐能力。由于本文共识机制应用于去中心化联邦学习，每轮模型训练完成后才进行共识，因此系统对吞吐量的需求相对较低。BCS-FL 采用了更加轻量级的共识机制，使其在大规模协作环境下更加适用，尤其适用于需要高可信数据存储和去中心化计算的场景。

交易时延是衡量共识机制性能的重要指标，本文将交易时延定义为从交易发出到区块共识完成所经历的时间。交易时延的高低直接影响共识机制的效率，为了保证测试的公平性，本文在实验过程中确保所有测试均在相同的网络环境下进行。此外，为了保证 PBFT 和 BCS-FL 共识机制在相同条件下进行比较，实验中 BCS-FL 共识算法的节点数量等同于整个网络中的节点数量。实验分别在 5、10、20、30、40、50 个节点的情况下，对整个网络的交易时延进行了测试。每种节点规模下均测试 30 次，并取其平均值，以减少网络波动对实验结果的影响。

实验结果如图 6 所示，可以看出 PBFT 和 BCS-FL 共识机制的交易时延均随节点数量的增加而上升。然而，由于 PBFT 采用全网广播和多轮投票机制，使得其通信开销较大，因此 PBFT 的交易时延增长呈指数级趋势，在节点数达到 50 时，平均时延已接近 6 秒。相比之下，BCS-FL 通过引入 PoS 机制优化了共识流程，减少了共识节点间的通信复杂度，从而有效降低了交易时延。实验结果表明，在相同节点规

模下，BCS-FL 共识机制的时延始终小于 PBFT，且随着网络规模的扩大，这种优势更为明显。当节点数量达到 50 时，BCS-FL 的交易时延保持在 2.7 秒左右，远低于 PBFT 的 6 秒左右。由于本实验的应用场景是去中心化联邦学习(BCS-FL)，在每轮训练完成后才会进行共识，相比于传统 PBFT 在大规模网络下因高通信开销导致的显著时延增长，BCS-FL 由于采用轻量级共识机制，使其在大规模协作环境下更具适应性，能够更高效地支持分布式模型训练与更新。

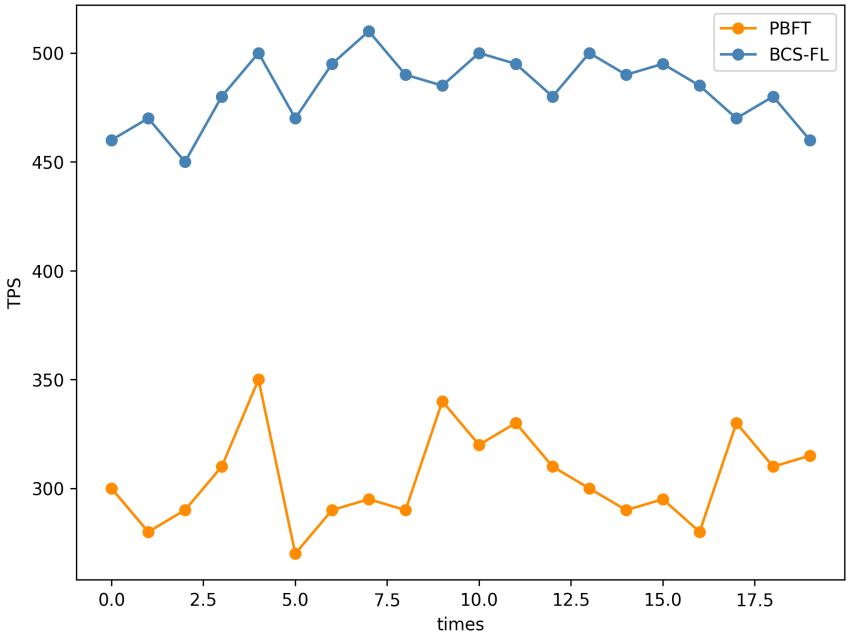

Figure 5. Testing the transaction throughput of BCS-FL and PBFT
图 5. 对于 BCS-FL 和 PBFT 交易吞吐量的测试

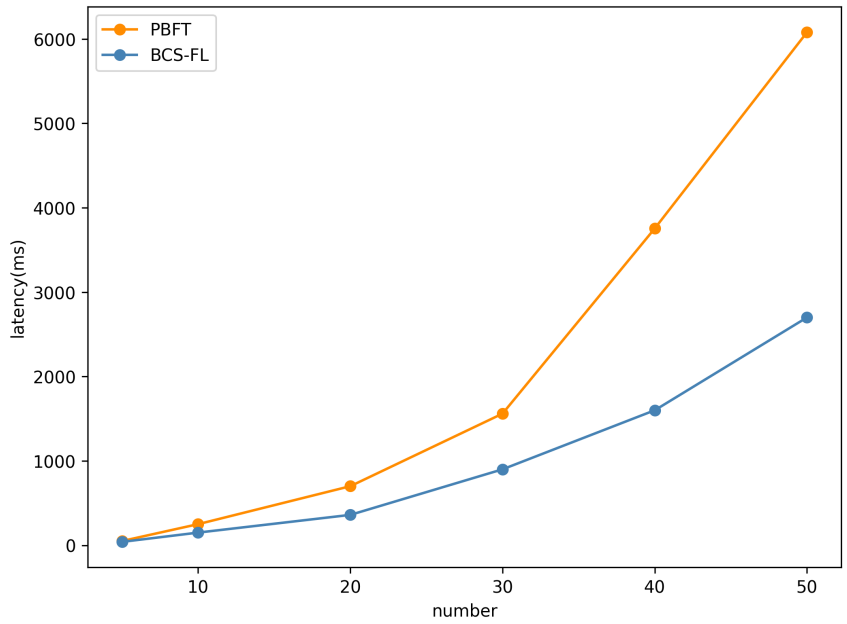

Figure 6. Testing the latency of BCS-FL and PBFT under different numbers of nodes
图 6. 对于 BCS-FL 和 PBFT 在不同节点数量下时延的测试

6. 总结与展望

本文提出了一种基于区块链的去中心化联邦学习框架 BCS-FL，旨在提升联邦学习的安全性、可靠性和透明度。该框架通过区块链技术取代传统中心化服务器，并结合 PoS 共识机制对模型更新进行筛选与聚合，从而有效防范投毒攻击，确保数据的完整性与可追溯性。实验结果表明，与传统联邦学习方法和区块链方法相比，BCS-FL 具备更优异的分类精度和隐私保护能力。具体而言，BCS-FL 结合余弦相似度和差分隐私机制有效剔除异常模型并包含模型信息，使得全局模型在面对恶意节点时仍能保持较高的准确率。此外，在 MNIST 和 CIFAR-10 数据集上的实验进一步验证了 BCS-FL 在抗攻击能力、数据隐私保护和模型优化方面的有效性，展现出其在去中心化联邦学习场景下的优势。未来的研究可以进一步优化区块链存储效率，如引入去中心化存储(如 IPFS)以减少模型存储开销，并提升系统的吞吐量。此外，BCS-FL 还可扩展至更广泛的应用场景，例如智能制造、医疗数据共享和金融风控，以推动区块链与联邦学习的深度融合，促进数据安全与协作智能的发展。

参考文献

[1] Wan, J., Li, J., Imran, M., Li, D. and Fazal-e-Amin (2019) A Blockchain-Based Solution for Enhancing Security and Privacy in Smart Factory. *IEEE Transactions on Industrial Informatics*, **15**, 3652-3660. https://doi.org/10.1109/tii.2019.2894573

[2] Tange, K., De Donno, M., Fafoutis, X. and Dragoni, N. (2020) A Systematic Survey of Industrial Internet of Things Security: Requirements and Fog Computing Opportunities. *IEEE Communications Surveys & Tutorials*, **22**, 2489-2520. https://doi.org/10.1109/comst.2020.3011208

[3] Mcmahan, B., Moore, E., Ramage, D., *et al*. (2017) Communication-Efficient Learning of Deep Networks from Decentralized Data. In: *The 20th International Conference on Artificial Intelligence and Statistics*, PMLR, 1273-1282.

[4] Sweeney, L. (2002) k-Anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **10**, 557-570. https://doi.org/10.1142/s0218488502001648

[5] Machanavajjhala, A., Gehrke, J., Kifer, D. and Venkitasubramaniam, M. (2006) L-Diversity: Privacy Beyond K-Anonymity. 22*nd International Conference on Data Engineering* (*ICDE*'06), Atlanta, 3-7 April 2006, 24-35. https://doi.org/10.1109/icde.2006.1

[6] Li, N., Li, T. and Venkatasubramanian, S. (2007) T-Closeness: Privacy Beyond K-Anonymity and L-Diversity. 2007 *IEEE* 23*rd International Conference on Data Engineering*, Istanbul, 15-20 April 2007, 106-115. https://doi.org/10.1109/icde.2007.367856

[7] Dwork, C. (2006) Differential Privacy. In: Bugliesi, M., *et al*., Eds., *Automata, Languages and Programming*, Springer, 1-12. https://doi.org/10.1007/11787006_1

[8] 宋成, 程道晨, 彭维平. 一种高效的联邦学习隐私保护方案[J]. 西安电子科技大学学报, 2023, 50(5): 178-187.

[9] Basu, D., Ghosh, U. and Datta, R. (2022). 6G for Industry 5.0 and Smart CPS: A Journey from Challenging Hindrance to Opportunistic Future. 2022 *IEEE Silchar Subsection Conference* (*SILCON*), Sircar, 4-6 November 2022, 1-6. https://doi.org/10.1109/silcon55242.2022.10028927

[10] Boobalan, P., Ramu, S.P., Pham, Q., Dev, K., Pandya, S., Maddikunta, P.K.R., *et al*. (2022) Fusion of Federated Learning and Industrial Internet of Things: A Survey. *Computer Networks*, **212**, Article ID: 109048. https://doi.org/10.1016/j.comnet.2022.109048

[11] Hao, M., Li, H., Luo, X., Xu, G., Yang, H. and Liu, S. (2020) Efficient and Privacy-Enhanced Federated Learning for Industrial Artificial Intelligence. *IEEE Transactions on Industrial Informatics*, **16**, 6532-6542. https://doi.org/10.1109/tii.2019.2945367

[12] Xiao, X., Wu, T., Chen, Y. and Fan, X. (2019) Privacy-Preserved Approximate Classification Based on Homomorphic Encryption. *Mathematical and Computational Applications*, **24**, Article No. 92. https://doi.org/10.3390/mca24040092

[13] Castor, M. and Liskov, B. (1999) Practical Byzantine Fault Tolerance. *Proceedings of the* 3*rd Symposium on Operating Systems Design and Implementation* (*OSDI*), New Orleans, 22-25 February 1999, 173-186.