计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
63(3): 685−709, 2026

FedPPB: 基于 PSO 和 Paillier 加密算法的区块链联邦学习方法

沈凡凡[1] 刘梓昂[1] 梁琦玮[1] 徐 超[1] 陈 勇[1] 何炎祥[2]
[1](南京审计大学计算机学院 南京 211815)
[2](武汉大学计算机学院 武汉 430072)
(ffshen@nau.edu.cn)

FedPPB: A Blockchain Federated Learning Method Based on PSO and Paillier Encryption Algorithm

Shen Fanfan[1], Liu Zi'ang[1], Liang Qiwei[1], Xu Chao[1], Chen Yong[1], and He Yanxiang[2]
[1](*School of Computer Science*, *Nanjing Audit University*, *Nanjing* 211815)
[2](*School of Computer Science*, *Wuhan University*, *Wuhan* 430072)

Abstract In the landscape of distributed intelligent systems, achieving a harmonious balance between data privacy and system efficiency remains a paramount challenge. While existing blockchain-based federated learning approaches offer enhanced security, they frequently encounter critical bottlenecks related to static resource allocation, excessive system overhead, and vulnerability to malicious participants. We propose a novel blockchain federated learning framework (FedPPB). FedPPB integrates the particle swarm optimization (PSO) for dynamic role allocation and Paillier homomorphic encryption for robust privacy preservation. The main idea of FedPPB is that PSO adaptively adjusts the number of worker, validator, and miner nodes by leveraging a multi-objective fitness function. Meanwhile, the model accuracy, verification latency, and block generation time are concurrently optimized. For security, worker nodes employ Paillier algorithm to encrypt all parameter updates before transmission. Subsequently, these parameters are decrypted and validated by validator nodes to ensure integrity. Finally, miner nodes finalize the process by generating immutable blocks to update the global model. The security of this architecture is rigorously established through theoretical analysis. Comprehensive experiments on MNIST, Fashion-MNIST, and CIFAR-10 datasets demonstrate that FedPPB significantly outperforms the existing methods, particularly in adversarial environments. Even when confronted with malicious node proportions of 15% and 25%, the proposed framework exhibits superior model accuracy and exceptional robustness, providing a sophisticated and practical solution for building secure, efficient, and adaptive decentralized intelligent systems.

Key words federated learning; blockchain; privacy protection; Paillier; PSO

摘 要 在人工智能快速发展的背景下,数据隐私与系统效率成为分布式智能系统中的核心挑战。现有研究虽在一定程度上缓解了数据泄露问题,但在资源分配、系统开销和安全性方面仍存在显著瓶颈。为此,构建了一种基于粒子群优化(particle swarm optimization, PSO)和 Paillier 加密算法的区块链联邦学习方法,简称 FedPPB,该方法融合 PSO 算法与 Paillier 同态加密算法,实现对系统角色的动态优化分配与训

收稿日期: 2025-06-17; **修回日期**: 2025-12-08
基金项目: 国家自然科学基金项目(61902189, 62472227, U23A20318); 江苏省高等学校基础科学(自然科学)研究项目(22KJA520004, 24KJA520005)
This work was supported by the National Natural Science Foundation of China (61902189,62472227,U23A20318) and the Basic Science (Natural Science) Research Project of Colleges and Universities in Jiangsu Province (22KJA520004, 24KJA520005).
通信作者: 徐超(xuchao@nau.edu.cn)

练参数的加密保护。首先,针对工作节点、验证节点和矿工节点的任务特点,通过 PSO 算法构建包含模型准确率、验证时间和区块生成时间的适应度函数,实现角色数量的动态调整;其次,工作节点通过 Paillier 算法对参数更新进行加密,验证节点解密参数并验证其合法性,矿工节点生成区块并更新全局模型;最后,从理论上证明了 PSO 算法和 Paillier 算法分别在角色分配和参数加密中的安全性。实验表明,在 MNIST, Fashion-MNIST, CIFAR-10 数据集上,当恶意节点的占比为 15% 和 25% 时,FedPPB 显著优于现有方法,展现出更高的准确率与鲁棒性。

关键词 联邦学习;区块链;隐私保护;Paillier;粒子群优化

中图法分类号 TP181; TP309.2; TP309.7

DOI: 10.7544/issn1000-1239.202550483 **CSTR:** 32373.14.issn1000-1239.202550483

物联网是指通过网络相互连接的计算设备、机械设备和数字机器组成的分布式系统,这些设备能够与其他物联网设备以及云端进行数据采集、交换和处理。随着物联网和云计算的快速发展,以及企业数据量的持续增长,大数据生态系统已逐渐形成。在该生态系统中,不同来源的海量数据将在用户之间共享。然而,将数据共享或者外包至云端可能引发隐私泄露和安全威胁。为解决物联网和云计算环境中数据共享带来的隐私安全问题,联邦学习(federated learning, FL)[1]作为一种新兴的分布式机器学习范式被提出,通过本地设备协同训练模型而不共享原始数据[2-3],有效缓解了隐私泄露风险,并在物联网、医疗保健和工业智能等领域展现出广泛应用前景[4]。然而,基于服务器-客户端架构的传统联邦学习系统因依赖中央服务器,存在单点故障风险,且在动态异构网络环境中,资源分配、隐私保护和通信效率等问题仍未得到有效解决。这些问题限制了联邦学习在复杂分布式系统中的可扩展性和鲁棒性[5-6]。

因此,将联邦学习应用于真实的、大规模分布式系统中,仍面临 3 个相互交织的严峻挑战,这些挑战共同制约了其安全与效率。首先,是异构环境下资源利用率低下与系统稳定性差的根本性矛盾,传统的静态角色分配模型无法解决节点异构性与训练过程动态需求之间的内在张力,极易导致性能瓶颈。其次,是贯穿数据处理全流程的隐私泄露风险,共享的模型参数更新构成了潜在的信息泄露信道,易遭受高级推理或模型毒化等攻击。最后,是沉重的系统开销对可扩展性的制约,尤其是在引入区块链以增强去中心化时,其链上共识与验证过程不可避免地会引入显著的额外计算与时间开销,从而导致端到端时延增加。

为了对上述挑战进行深入的量化分析与实证支持,我们进行了针对性的实验与分析。

首先,针对异构环境下的资源利用率低下问题,我们将其定义为由系统瓶颈所导致的“高资源闲置率”与“高系统等待时间占比”同时存在的低效现象。具体而言,如图 1 所示,动机性实验在典型的瓶颈轮次中,高性能节点的 CPU 资源闲置率高达 85% 。如图 2 所示,任务时序分析则进一步揭示,该瓶颈的根源在于单一慢速矿工节点,即单一慢速节点迫使其他节点长时间等待。

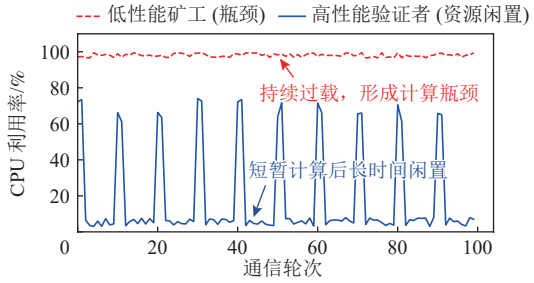


Fig. 1 Divergence diagram of CPU utilization

图 1 CPU 利用率分化图

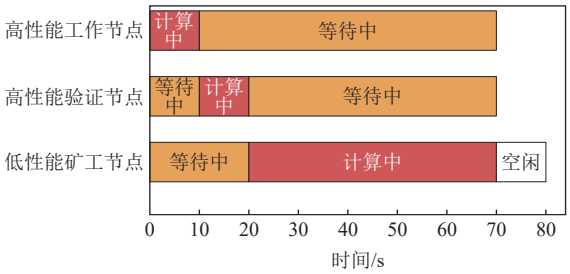


Fig. 2 Task state timing diagram

图 2 任务状态时序图

其次,针对贯穿全局的隐私泄露风险,我们通过模型在恶意攻击下的鲁棒性进行量化。后续实验数据明确揭示了现有方法的脆弱性,在面临 25% 恶意节点攻击的严苛场景下,基线模型 VBFL[7] 的测试准确率会崩溃至仅 10.60% 。

最后，针对高昂的系统开销，我们将其关键衡量指标定义为“区块生成时间”，因为它直接决定了每一轮全局模型更新时的端到端时延，我们的基线测试表明，在静态方案中，单轮的区块生成时间最高可达 18.64 s，这种级别的时延在高频次的联邦学习迭代中会迅速累积，严重制约了系统的可扩展性与训练效率。这些实验从 3 个维度揭示了现有方法的固有缺陷，为本文构建能够协同应对这些挑战的整体性框架提供了强有力的动机。

为应对上述一系列挑战，现有研究虽尝试通过区块链技术实现去中心化以缓解单点故障[8]，或采用加密算法增强隐私保护[9]，但对角色分配优化的系统性研究不足，且未充分平衡隐私保护与系统效率。

为此，本文提出了一种基于粒子群优化（particle swarm optimization, PSO）和 Paillier 加密算法的区块链联邦学习方法——FedPPB，该方法结合 PSO 算法和 Paillier 同态加密算法。PSO 算法的引入旨在动态优化资源分配，减少系统开销；Paillier 同态加密算法则从根本上保障参数传输与聚合过程中的隐私安全。

尽管 PSO 算法和 Paillier 同态加密技术在各自领域均有显著贡献，但将两者无缝融合以解决联邦学习中的三大核心挑战仍面临若干技术难题。例如，如何协调 PSO 的动态优化机制与 Paillier 加密所带来的计算开销，从而在保证隐私安全的同时确保整体训练效率的提升，是实现该跨域技术融合的关键挑战。本文旨在系统性地解决这些问题，为联邦学习提供一个既高效又安全的综合性解决方案。本文的主要贡献有 4 点：

1）提出了一种基于 PSO 和 Paillier 加密算法的区块链联邦学习方法——FedPPB，通过将 PSO 的动态优化能力与 Paillier 的同态加密特性相结合，在区块链框架下有效解决了联邦学习中资源动态分配、隐私保护与系统开销的三大核心挑战。

2）设计了基于 PSO 算法的动态角色分配优化框架，针对工作节点、验证节点和矿工节点的任务特点，构建包含模型准确率、验证时间和区块生成时间的适应度函数，实现角色数量的动态调整。

3）设计了基于 Paillier 同态加密的训练参数保护框架来对模型参数进行加密，确保传输和聚合过程中的隐私安全。并且通过 Paillier 同态加密的语义安全性，从理论上证明了 Paillier 同态加密算法在训练参数传输过程中的安全性。

4）实验表明，FedPPB 在 15% 的恶意设备下，准确率达到了 96.81%；在 25% 的恶意设备下，依然保持了 87.59% 的准确率，区块生成时间降低了 27.6%。FedPPB 有效抵御了恶意设备的攻击，保障了分布式系统的数据隐私性并且显著提升了系统在动态环境中的资源利用效率和鲁棒性。

1 相关工作

1.1 基于区块链的联邦学习

随着数据隐私问题的日益突出，传统集中式机器学习因需传输原始数据而面临隐私泄露风险。McMahan 等人[10] 于 2017 年提出联邦学习，通过本地训练和聚合模型更新实现隐私保护。Konečný等人[11] 通过结构化更新降低通信开销并提升隐私保护。然而，这些方法多依赖服务器-客户端架构，存在单点故障风险[12]。

为解决单点故障问题，研究者们探索将区块链技术引入联邦学习。Korkmaz 等人[13] 提出 Chain FL，利用以太坊和 POA 共识算法增强鲁棒性。高胜等人[14] 设计异步联邦学习框架来兼容隐私保护与对抗攻击防御。刘炜等人[15] 提出基于区块链和动态评估的隐私保护联邦学习模型 SPFLChain，以区块链替代中心服务器解决单点故障，结合稀疏化与差分隐私保障隐私安全，并通过动态评估优化激励公平性。然而，这些方法因频繁数据传输导致通信开销高[16]。

为提升隐私与效率，Fan 等人[17] 提出 LPBFL，结合轻量级签名和同态加密降低计算与通信负担。Yu 等人[18] 通过功能加密实现轻量级隐私保护。Li 等人[19] 引入贝叶斯博弈激励机制提升参与积极性。Chen 等人[7] 提出 VBFL，通过选择性验证和 PoS 共识增强模型可靠性。

尽管上述研究在模型验证机制方面取得一定进展，联邦学习系统仍面临模型参数泄露导致的安全风险，恶意参与者可能通过分析局部模型参数发起推理攻击，这不仅会降低模型精度，还会严重削弱系统的鲁棒性[20]。针对这一挑战，Paillier 同态加密算法提供了一种有效的解决方案。本文采用该算法对模型参数进行加密处理，在不解密的前提下可直接对密文执行加法与数乘运算，从而在保障数据隐私的同时显著提升系统的安全性。

1.2 PSO 算法

PSO 算法是由 Kennedy 等人[21] 提出的通过群体合作和信息共享来寻找最优解的优化算法，该算法中每个参与的个体将被视作一个具有位置和速度的粒子，通过不断更新自身和群体的历史最优位置与

速度来找到全局最优解[22]。为解决传统 PSO 易陷入局部最优和收敛速度慢的问题，Shi 等人[23]引入线性递减惯性权重策略，初期增强全局搜索，后期提升局部搜索精度，实验验证其在单峰和多峰函数优化中的优越性。Yao 等人[22]进一步提出混合策略 PSO，结合自适应权重和柯西变异，显著提升全局和局部搜索能力及算法鲁棒性。PSO 因实现简单、适应性强，在特征选择、神经网络优化等领域得到广泛应用。

在实际场景中，Demir 等人[24]将 PSO 与梯度提升结合，优化地震液化侧向扩展来预测模型的超参数，提升预测精度和泛化能力。Zhang 等人[25]提出基于 PSO 的车辆调度方案，结合差分进化算法优化共享车辆调度效率。Lin 等人[26]融合 PSO 与强化学习，解决动态工业环境中自动化引导车的路径规划问题，利用 PSO 的快速收敛特性提升路径优化效率。

PSO 的全局搜索能力和高效收敛特性使其在联邦学习和区块链领域展现出巨大潜力。Supriya 等人[27]提出 PSO-FL(PSO-enabled federated learning)算法，用于早期森林火灾检测，通过传输最优权重降低通信开销，提升检测准确性和响应速度。然而，权重传输仍可能带来数据开销。Park 等人[28]开发 FedPSO 算法，通过优化权重更新和传输模型性能评分，显著降低通信成本并增强网络不稳定环境下的鲁棒性。另有研究利用 PSO 优化联邦学习中的带宽分配，支持更多客户端参与训练，提升系统性能[29]。Houssein 等人[30]提出 FedPSO 框架，通过模型评分更新全局模型，结合 PSO 优化数据交互，降低通信负载并提高网络波动下的稳定性。

当前，研究者们通过 PSO 算法解决通信开销、模型效率等问题。然而，我们将 PSO 算法应用在区块链的联邦学习系统中，旨在解决系统中角色数量的分配问题。具体而言，通过评估 3 种角色的任务指标，利用 PSO 算法寻找最优的角色分配方案，从而显著提升整个模型训练的效率。

1.3 Paillier 同态加密算法

同态加密技术作为一种支持密文域计算的密码学方法，能够在保证数据隐私的前提下实现特定的运算。该技术允许对加密数据直接执行计算操作，所得结果经解密后与对明文数据进行相同运算的结果一致，这一特性使其成为隐私保护计算领域的关键使能技术[11]。Paillier 同态加密算法作为一种非对称加密算法，以其加法同态特性成为当前最有效的同态加密系统之一，该算法提供了针对选择明文攻击的语义安全[31-32]，由于 Paillier 算法具有同态性，允许在不暴露原始数据的情况下对加密数据进行处理，为数据的隐私安全提供了重要保障[33-34]。

目前许多研究者将 Paillier 同态加密算法应用到联邦学习系统中，以此进一步提高隐私安全性。该算法通过加密模型训练过程中的参数以及梯度，在不泄露敏感数据的前提下确保模型训练的高效性。这使得各参与方无需共享原始数据即可协同训练机器学习模型，且在不损失训练精度的基础上，提供了更强大的隐私保护能力。研究者将 Paillier 同态加密算法与联邦学习相结合，通过 Paillier 算法对模型训练过程中上传的模型梯度进行加密操作，降低了模型训练过程中的隐私泄露风险[35-37]。Fang 等人[35]提出 PFMLP 框架，利用 Paillier 算法加密客户端梯度，即使中央服务器遭受攻击，也仅泄露加密数据，有效抵御推理攻击。Marcolla 等人[33]结合 Paillier 加密与双线性聚合签名，验证共享梯度的正确性，兼顾模型精度与隐私保护。Ku 等人[37]提出基于可信第三方的医疗联邦学习方案，结合同态重加密与 Paillier 算法，支持加密参数安全聚合，保障医疗数据隐私与模型鲁棒性。

然而，Paillier 同态加密算法增加计算开销，影响训练效率。为此，Zhang 等人[38]设计批处理加密技术，通过梯度量化和批量编码降低计算与通信开销。陶云亭[39]提出多维数据聚合协议，优化 Paillier 同态加密解密效率。Han 等人[40]针对异构客户端差异，开发高效批量加密方法，显著减少训练时间与通信成本。尽管梯度加密增强了安全性，但无法全面保护模型参数，恶意攻击者可能规避加密机制获取参数信息。为此，Ma 等人[41]提出基于分布式 Paillier 同态加密算法的拜占庭鲁棒机制，保护工作者节点参数传输。其他研究通过 Paillier 同态加密算法的安全聚合协议，防止恶意服务器窃取参数[42]。

鉴于 Paillier 同态加密算法在隐私保护方面的优越性，本研究将其应用于区块链赋能的联邦学习系统，为联邦学习系统在模型训练参数传输过程中的安全性提供更强的保障。

2 预备知识

2.1 PSO 算法

PSO 算法是一种基于群体智能的随机优化技术，通过模拟鸟群觅食行为，使粒子在多维解空间中迭代搜索最优解。本研究选择 PSO 算法来解决区块链联邦学习中的角色数量分配问题，主要源于其实现

简单、计算高效、收敛速度快以及对非线性多维问题具有优异的全局搜索能力。相较于其他优化策略，如收敛缓慢的模拟退火算法[43]、计算开销较大且不适于离散空间搜索的贝叶斯优化[44]、参数复杂的遗传算法[45]、容易陷入局部最优的蚁群算法[46]、实现难度较高的萤火虫算法[47]和训练难度高的强化学习方法[48]，PSO 算法能够以最低的成本高效寻找到最优解。为严谨论证这一选择，我们将主要优化算法的特点与适用性进行了对比，如表 1 所示。

在 PSO 算法中，被视作搜索空间中粒子的每个潜在解都具备特定的速度属性和位置属性。根据粒子的个体极值和全局极值，不断更新自己的位置和速度，最终找到最优解。其核心思想为通过个体极值和全局极值引导整个搜索过程，进而实现全局最优。算法 1 展示了 PSO 算法的完整流程。

算法 1. PSO 粒子群优化算法。

输入：粒子群规模 N，最大迭代次数 T，惯性权重 ω，加速系数 c_1，c_2；

输出：全局最优位置 $gbest$。

① 初始化粒子位置 x_i 和速度 v_i，$i=1, 2, \cdots, N$；

② 评估每个粒子的适应度值 $f(x_i)$；

③ 设置个体最优位置 $pbest_i = x_i$，全局最优位置 $gbest$ 为当前最优粒子位置；

④ for $t = 1$ to T do

⑤ for $i = 1$ to N do

⑥ 更新粒子速度 $v_i^{t+1} = \omega v_i^t + c_1 r_1 \left(pbest_i - x_i^t\right) + c_2 r_2 (gbest - x_i^t)$；

⑦ 更新粒子位置 $x_i^{t+1} = x_i^t + v_i^{t+1}$；

⑧ 评估新位置的适应度值 $f(x_i^{t+1})$；

⑨ if $f\left(x_i^{t+1}\right) < f(pbest_i)$ then

⑩ $pbest_i = x_i^{t+1}$；

⑪ end if

⑫ end for

⑬ if 发现更优解 then

⑭ 更新全局最优位置 $gbest$；

⑮ end if

⑯ end for

⑰ return $gbest$。

算法 1 中，v_i^t 为粒子 i 在时间 t 的速度；ω 为惯性权重，控制粒子在搜索空间中当前速度对于下一位置的影响程度，当惯性权重较大时有助于粒子保持当前的搜索方向即利于全局搜索，而当惯性权重较小时，粒子更容易受到个体极值和全局极值的影响即利于局部搜索；c_1 和 c_2 是学习因子，决定粒子向其个体极值和全局极值移动的程度，学习因子 c_1 决定了粒子向其个体极值靠近的程度，学习因子 c_2 决定了粒子向全局极值靠近的程度；r_1 和 r_2 为 [0,1] 区间内的随机数，$pbest_i$ 为粒子的个体极值，$gbest$ 是全局极值。x_i^t 为粒子 i 在时间 t 的位置。可以看出，粒子通过不断更新速度和位置，在解空间中搜索最优解，兼顾了局部搜索和全局搜索的平衡。

2.2 Paillier 同态加密算法

Paillier 加密算法是由 Pascal Paillier 于 1999 年提出的一种高效的加法同态加密算法，因该算法支持密文的加法和数乘操作，在数据的隐私保护中起到了至关重要的作用。Paillier 算法作为 EIGamal 加密算法的重要变体，基于模 n 乘法群的数学特性实现主要包括密钥的生成、加密和解密 3 个部分。

1）密钥的生成

①选择 2 个相似的大素数 p 和 q，计算 $N = pq$，其中 N 的位数决定密钥的长度；然后根据式（1）计算欧拉函数：

$$\varphi(N) = (p-1)(q-1) \text{ 。} \tag{1}$$

②选择 1 个生成元 $g \in \mathbb{Z}_{N^2}^*$，其表示模 N 的 1 个原根。

③选择 1 个随机数 λ，并计算 1 个模逆 L，满足式

Table 1 Comparative Analysis of Major Optimization Algorithms

表 1 主要优化算法的对比分析

优化算法	核心思想	计算复杂度	收敛速度	实现难度	参数调优
PSO[21]	模拟鸟群觅食行为，通过粒子迭代寻找最优解	较低	较快	简单	参数少，易于调优
模拟退火[43]	模拟物理退火过程，以概率接受劣解	中等	较慢	简单	依赖于退火温度参数，收敛速度慢
贝叶斯优化[44]	基于概率代理模型，高效搜索最优解	较高	较慢	复杂	需构建代理模型和采集函数，计算量大
遗传算法[45]	模拟生物进化过程，通过选择、交叉、变异等操作进行搜索	较高	较慢	较复杂	参数多，调优困难
蚁群算法[46]	模拟蚂蚁寻找食物路径的行为，利用信息素进行协作搜索	中等	较快	较复杂	参数多，对问题依赖强，调优复杂
萤火虫算法[47]	模拟萤火虫根据亮度吸引同伴的行为，通过吸引力进行移动	较低	较快	简单	参数少但对设置较敏感
强化学习[48]	通过智能体与环境交互学习最优策略	较高	不确定	极复杂	需大量环境交互，收敛时间长

(2)和式(3)：

$$\lambda = lcm(p-1, q-1), \tag{2}$$

$$L\lambda \equiv 1 \bmod \varphi(N) \text{ 。} \tag{3}$$

④ 生成公钥$Pubkey = (N, g)$，私钥$Prikey = \lambda$ 。

2)加密

给定明文m，其中$m \in \mathbb{Z}_N$，选择 1 个随机数r，得到加密后的密文C，计算如式(4)所示：

$$C = g^m r^\lambda \bmod N^2 \text{ 。} \tag{4}$$

3)解密

给定密文C，使用私钥进行解密计算出明文m的公式：

$$m = L\left(C^\lambda \bmod N^2\right)\mu \bmod N, \tag{5}$$

其中μ是N的模逆元，用于解密操作，计算公式为

$$\mu = \left(L\left(g^2 \bmod N^2\right)\right)^{-1} \bmod N \text{ 。} \tag{6}$$

Paillier 算法作为一种同态加密算法，允许在加密状态下进行加法和数乘运算。其具有加法同态性，即给定 2 个密文$C_1 = E(m_1)$和$C_2 = E(m_2)$，无需对其进行解密操作便可直接计算出对应的明文之和的密文$E(m_1 + m_2)$。同时，该算法还支持数乘同态性，即给定 1 个密文$C = E(m)$和 1 个标量k，可直接计算出$E(km)$：

$$E(km) = C^k \bmod N^2 \text{ 。} \tag{7}$$

Paillier 加密算法因其具备独特的同态特性，使得数据加密过程不仅更加灵活，并能支持复杂的计算操作。该算法允许在加密数据上直接执行加法和数乘运算，而无需先对数据进行解密。这种特性极大地增强了数据处理的效率，同时避免了数据在传输和处理过程中的泄露风险。通过这种方式，Paillier 算法为数据的安全性和隐私性提供了强有力的保障。

3 FedPPB 方法

3.1 方法概述

本文提出了一种基于 PSO 和 Paillier 加密算法的区块链联邦学习方法——FedPPB。该方法的设计核心在于解决联邦学习中的资源动态分配、隐私保护与系统开销的三大核心挑战。我们首先利用了 PSO 算法解决联邦学习中的资源分配与系统开销问题，通过设计适应度函数动态调整角色数量，以提升训练效率并降低验证实验和区块生成时间。其次，为应对隐私风险，我们集成了 Paillier 同态加密技术，确保所有客户端参数在传输与聚合过程中均处于密文状态，从而在数学上保障数据机密性。这种融合实现了从局部资源优化到全局安全保障的无缝衔接，使得 FedPPB 框架能够高效、安全地完成联邦学习任务。同时，为便于理解和后续分析，本文常用的符号及其含义在表 2 中进行了汇总。FedPPB 的工作流程如图 3 所示。

Table 2 Key Notations Discription

表 2 关键符号说明

符号	符号定义
N	节点数量
d	在 FedPPB 中组成的节点
w	工作节点
v	验证节点
m	矿工节点
R_i	第i轮通信
L_{w_i}	由工作节点训练的局部模型
θ_i	模型训练参数
$\theta_i^{\text{encrypt}}$	使用 Paillier 加密算法加密后的局部模型参数
$\theta_i^{\text{decrypt}}$	使用 Paillier 加密算法解密后的局部模型参数
G_i	第i轮通信R_i的全局模型
R_{w_i}	第i轮通信R_i中工作节点所得到的奖励
Tr_{w_i}	工作节点交易，包含$L_{w_i}, \theta_i^{\text{encrypt}}, R_{w_i}$
$R_{v_i}^{\text{verify}}$	验证节点通过验证在第i轮通信中的工作节点交易所得到的奖励
$R_{v_i}^{\text{validate}}$	第i轮通信R_i中工作节点所得到的投票奖励
$v_i(\theta_i^{\text{decrypt}})$	投票结果
Tr_{v_i}	验证节点交易,包含$R_{v_i}^{\text{verify}}, R_{v_i}^{\text{validate}}, Tr_{w_i}, v_i(\theta_i^{\text{decrypt}})$
R_{m_i}	矿工节点所获得的奖励
$block_i^m$	矿工节点m第i轮通信R_i中获选区块

对于图 3 中的步骤①，在每轮训练开始之前，PSO 优化算法根据当前系统的状态迭代优化角色分配，确保性能与安全的平衡，实现了工作节点、验证节点和矿工节点的动态调整。

对于图 3 中的步骤②，将参与的设备最开始随机划分为工作节点、验证节点和矿工节点这 3 种角色，并通过图 3 中的步骤①得到了每个节点的最优数量，工作节点在本地数据集上训练模型，并利用 Paillier 同态加密算法对参数更新进行加密，生成可支持加法的运算的交易发送至验证节点，以确保隐私安全。

对于图 3 中的步骤③，验证节点接收交易后，使用 Paillier 私钥解密参数并验证其合法性，随后通过投票决定是否接受更新，并将投票结果封装为新交

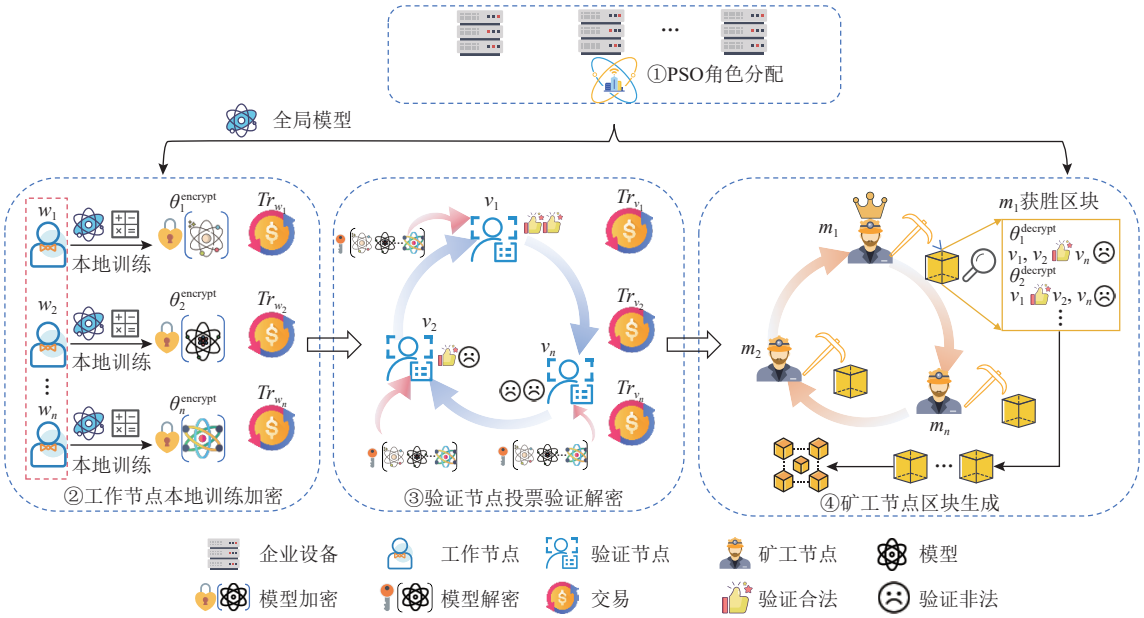


Fig. 3 The framework diagram of FedPPB

图 3 FedPPB 框架图

易提交给矿工节点。

对于图 3 中的步骤④，矿工节点负责验证交易并将通过验证的加密参数和投票结果打包为候选区块，通过权益证明（proof of stake, PoS）共识机制添加到区块链中，进而基于验证通过的参数安全聚合更新全局模型，并分发给工作节点以进行下一轮训练。

FedPPB 通过 PSO 优化提升效率、Paillier 加密保障隐私、区块链技术实现去中心化与安全性，构建了一个安全、高效且自适应的联邦学习系统，有效应对传统联邦学习中的隐私泄露和性能瓶颈挑战。

3.2 基于 PSO 算法的动态角色分配优化框架

本节提出了一种基于 PSO 算法的动态角色分配优化框架。该框架的核心思想在于引入具有卓越全局搜索能力和快速收敛特性的 PSO 算法，以实现联邦学习系统中工作节点、验证节点和矿工节点数量的自适应调整。PSO 算法的优化过程可以利用地形上的粒子运动来理解。如图 4 所示，每个粒子代表一种潜在的角色分配方案（即特定数量的工作节点和验证节点，矿工节点数量由总设备数减去前两者的数量决定），其在 3 维空间中的位置对应着该方案的适应度值（Z 轴）。算法通过迭代更新粒子的速度和位置引导粒子群在适应度地形上逐步趋向最优解。

在区块链联邦学习领域中，现有研究如 VBFL[7]，FedANIL+[49]，VBSFL[50] 等，普遍采用的是静态角色数量分配策略，即设备的角色数量在每个通信轮次中固定不变。这种策略存在明显不足：固定角色数量

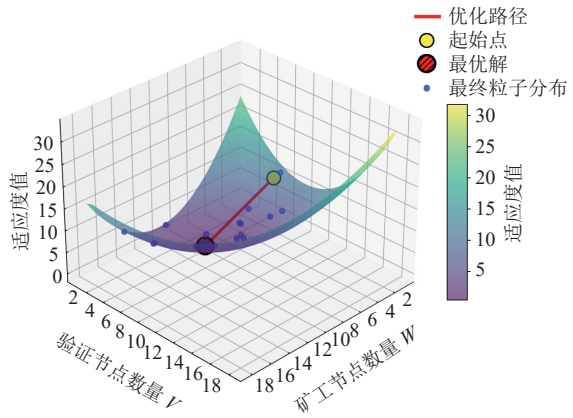


Fig. 4 Visualization of the PSO convergence process

图 4 PSO 收敛过程可视化

限制了系统的动态适应性；静态资源配置制约了模型训练效率。如表 3 所示，主要体现在其静态的角色数量分配机制。为了解决这些问题，本节引入 PSO 算法，提出了一种基于 PSO 算法的动态角色分配优化框架。通过构建目标优化模型实现角色分配的动态调整，从而提升系统的整体性能与资源利用效率。本方案主要分为 6 个部分：初始化粒子群、粒子位置解码、分配角色任务、3 种角色的并行评估、适应度计算以及更新最优解与终止条件的判断。具体如算法 2 所示。

算法 2. 基于 PSO 算法的动态角色分配优化框架。

输入：设备总数 N，PSO 参数（粒子数 S，迭代次数 T，惯性权重 ω，学习因子 c_1，c_2），约束参数 $V_{\min}$；

Table 3 Comparison of Key Functions of Related Methods

表 3 相关方法的关键功能对比

方法	使用区块链技术	优化系统开销	支持同态加密	动态角色数量分配
VBFL[7]	√	√		
VBSFL[50]	√	√		
FedANIL+[49]	√	√	√	
FedPPB（本文）	√	√	√	√

输出：全局最佳角色分配$gbestPos(W_{opt}, V_{opt}, M_{opt})$全局最佳适应度$gbestFit$。

① $gbestPos \leftarrow$ NULL, $gbestFit \leftarrow -\infty$;
② for $j = 1$ to S do /*初始化*/
③ $x[j] \leftarrow RandomPosition(N, V_{min})$, $v[j] \leftarrow 0$, $f[j] \leftarrow f(x[j])$;
④ $pbestPos[j] \leftarrow x[j]$, $f_{pbest}[j] \leftarrow f[j]$;
⑤ if $f_{pbest}[j] > gbestFit$ then
⑥ $gbestFit \leftarrow f_{pbest}[j]$, $gbestPos \leftarrow pbestPos[j]$;
⑦ end if
⑧ end for
⑨ for $t = 1$ to T do /*优化迭代*/
⑩ for $j = 1$ to S do
⑪ $f[j] \leftarrow f(x[j])$, $\widetilde{f}[j] \leftarrow$式(8)添加扰动;
⑫ end for
⑬ $\mu, \sigma \leftarrow Mean(\{\widetilde{f}\}), StdDev(\{\widetilde{f}\})$; /*Z-Score 统计量*/
⑭ for $j = 1$ to S do
⑮ $Z \leftarrow (\widetilde{f}[j] - \mu)/\sigma$;
⑯ if $|Z| > 3$ then
⑰ $x[j] \leftarrow RandomPosition(N, V_{min})$, $f[j] \leftarrow f(x[j])$, $pbestPos[j] \leftarrow x[j]$, $f_{pbest}[j] \leftarrow f[j]$;
⑱ else if $f[j] > f_{pbest}[j]$ then
⑲ $f_{pbest}[j] \leftarrow f[j]$, $pbestPos[j] \leftarrow x[j]$;
⑳ end if
㉑ if $f_{pbest}[j] > gbestFit$ then
㉒ $gbestFit \leftarrow f_{pbest}[j]$, $gbestPos \leftarrow pbestPos[j]$;
㉓ end if
㉔ $r_1, r_2 \leftarrow Rand(0,1)$;
㉕ $v[j] \leftarrow$式(9)更新速度;
㉖ $x[j] \leftarrow$式(10)更新位置;
㉗ end for
㉘ end for
㉙ return $gbestPos, gbestFit$。

PSO 粒子群优化算法是一种模拟鸟类群体觅食行为的优化算法，通过粒子间的协作和信息共享来寻找最优解。在该算法中，每一个粒子均为包含位置和速度 2 个核心变量的问题潜在解。由于 PSO 算法具有简单易实现且收敛速度快等特点，本节将其应用于传统的基于区块链的联邦学习系统中，借助该算法的特性进一步提升系统的模型性能。下文将对算法 2 进行详细介绍。

1)初始化阶段，位于算法 2 的行②~⑥，在该阶段中系统遍历所有的粒子，为每个粒子生成初始的角色数量。首先随机生成粒子位置$x[j]$，并满足$W_j + V_j + M_j = N$，初始化粒子速度$v[j] = 0$。其次基于设计的适应度函数计算初始的适应度值$f[j]$，并立即初始化该粒子的个体最优解($pbestPos[j]$, $f_{pbest}[j]$)，同时更新种群的全局最优解($gbestPos$, $gbestFit$)。这一阶段为后续的迭代优化过程建立了完整的初始状态。

2)优化阶段，优化阶段实现了粒子群算法的核心迭代过程，位于算法 2 的行⑨~⑳。该阶段首先执行隐私保护评估，采用式(8)对所有粒子的适应度添加拉普拉斯噪声，随后计算均值μ和标准差σ，并利用 Z-score 标准化方法过滤异常粒子。

$$\widetilde{f}[j] \leftarrow f[j] + Lap(0, \Delta f/\epsilon) \text{。} \tag{8}$$

在粒子交互更新阶段，位于算法 2 的行㉑~㉘，首先使用粒子的真实适应度$f[j]$更新其个体最优解($pbestPos[j]$, $f_{pbest}[j]$)，并随之更新全局最优解($gbestPos$, $gbestFit$)。根据式(9)更新速度$v[j]$，按照式(10)更新位置$x[j]$，位置更新后仍需满足W_j, V_j, $M_j \geqslant 1$, $V_j \geqslant V_{min}$。

$$v[j] \leftarrow \omega v[j] + c_1 r_1 (pbestPos[j] - x[j]) + c_2 r_2 (gbestPos - x[j]), \tag{9}$$

$$x[j] = x[j] + v[j] \text{。} \tag{10}$$

全局最优更新机制通过持续追踪并记录迭代过程中的最优适应度值，如果当前粒子的适应度值已优于目前的全局最优值，则对其更新并记录每轮迭代的最优解。

3.3 基于 PSO 算法的适应度函数设计

本节提出了基于 PSO 算法的适应度函数设计，粒子群优化算法的性能表现很大程度上取决于适应度函数的设计质量，该函数直接决定了优化过程的收敛特性和最终解的质量。本节将系统性地阐述适应度函数的设计原理及其数学表达形式。

本节所提出的 FedPPB 方法依据所划分 3 类角色

的工作任务进行适应度函数的目标设计。具体而言，工作节点负责基于本地数据集进行模型的训练与更新，因此将其在本地数据集上的准确性作为评估指标；验证节点承担交易内容的验证任务，确保提交至系统的数据和模型更新符合预定规则和标准，其工作效率通过完成验证的时间来评估；矿工节点负责将验证后的交易打包进新的区块，并尝试工作量证明以生成候选区块，其工作表现通过生成区块的时间来衡量。因此，该适应度函数旨在实现模型精度最大化、系统开销最小化以及系统安全性保障 3 个核心目标，其中模型精度通过优化角色分配提升联邦学习效果，系统开销通过降低验证时延和区块生成时间来提高系统效率，而系统安全性则通过设置验证者数量下限来确保。这一设计在保证模型性能的同时，兼顾了系统运行效率和安全性要求。

本节设计的适应度函数如式(11)所示，采用了多目标优化框架。

$$f(W,V,M)=\alpha Accuracy-\beta CommCost+\gamma SafetyPenalty(V)\ 。\tag{11}$$

适应度函数包含 3 个关键组成部分：其一为模型精度项，通过工作节点在本地测试集上的评估准确率来反映联邦学习效果。其二为系统开销项，包含验证节点对交易内容的验证时间t_v和矿工生成区块的时间t_m两个关键指标，如式(12)所示。

$$CommCost=t_vV+t_mM\ 。\tag{12}$$

其中，V和M分别表示验证节点的数量和矿工节点的数量。

其三为安全性惩罚项，基于 PBFT(practical Byzantine fault tolerance)[51]共识机制对验证者数量进行约束，如式(13)所示。

$$SafetyPenalty(V)=\begin{cases}-k, & V<V_{\min},\\ 0, & V\geqslant V_{\min},\end{cases}\tag{13}$$

其中 PBFT 作为一种高效的状态机复制算法，能够在$N\geqslant 3f+1$个节点中容忍f个恶意节点。系统中验证者需通过 PBFT 协议对加密参数进行投票以确保在多数节点同意后参数才能够被其余节点所接受，当验证者总数$V\geqslant 3f+1$时，可容忍最多f个恶意验证者节点进一步确保验证过程系统的安全性和可靠性。

3.4 基于 Paillier 同态加密的训练参数保护框架

本节提出了基于 Paillier 同态加密的训练参数保护框架。其核心设计思想在于引入 Paillier 同态加密算法，构建一个端到端的加密保护框架。该框架旨在确保模型训练参数始终处于加密状态，从而有效抵御潜在的窃听、篡改和推理攻击。通过利用 Paillier 算法的加法同态特性，系统能够在不解密敏感参数的前提下进行必要的计算与验证，极大地增强了联邦学习过程的安全性与可信度，同时配合区块链的去中心化特性，共同构建了一个鲁棒、高效且隐私友好的联邦学习生态。如图 3 的②③④所示，这 3 个步骤清晰展示了训练参数从工作节点生成、经 Paillier 公钥加密，到加密参数在网络中传输，再由验证节点在密文域进行验证或由矿工节点在密文域进行聚合的全过程。

在 3.1 节中，我们简单阐述了参与的角色划分以及后续的训练、验证、更新步骤。在 3.2 节中，通过 PSO 优化算法计算出了本轮的最佳角色分配个数，之后将进行参与的角色划分、模型训练与参数加密、交易验证与参数解密、区块创建与模型更新。在本节我们将详细阐述节点的更新流程。

在我们的 FedPPB 框架中，工作节点负责模型的训练，并通过 Paillier 同态加密算法对模型参数进行加密，从而确保数据在传输过程中的安全性，防止恶意攻击设备窃取数据。验证节点则负责验证加密后的模型参数，有效阻止恶意设备提交有害更新。矿工节点负责区块链的维护和区块的添加，为系统提供稳定且不可篡改的数据支持。在该系统架构中，通过角色划分机制巧妙地规避了单点故障的风险，从而显著增强系统的弹性。这种角色划分策略能够确保在个别设备出现故障时，系统的完整性和连续性依然得以保持，不受任何影响。此外，该策略不仅有效提升系统的运行效率和管理便捷性，还为系统带来更高的安全性与透明度，有力地推动联邦学习过程向更高效率和可靠性的方向迈进。具体的每种角色的分工任务如下：

1)工作节点。在 FedPPB 系统中，工作节点承担本地模型的训练任务。训练完成后，更新的模型参数通过 Paillier 同态加密算法进行加密处理，确保数据在传输过程中的安全性，防止被潜在的恶意攻击设备窃取。随后，加密后的参数被传输至验证节点。

2)验证节点。验证节点接收加密的模型参数后利用私钥验证参数的有效性，确保其真实性和完整性。

3)矿工节点。矿工节点的主要职责是构建和验证区块。验证节点完成模型验证后，会将验证结果和投票结果发送给矿工节点。矿工节点对这些结果进行进一步验证，随后将验证结果打包成区块，并将其安全地添加到区块链中。这一过程不仅保障了数

据的不可篡改性,还为系统的稳定运行提供了支持。

如算法 3 所示,基于 Paillier 同态加密的训练参数保护机制可划分为 3 个主要部分:模型训练与参数加密、交易验证与参数解密以及区块创建与模型更新。

算法 3. 基于 Paillier 同态加密的训练参数保护框架。

输入:设备总数 N, Paillier 算法公钥,私钥对 $(Pubkey, Prikey)$;

输出:更新的全局模型 G_{i+1}。

① for i in N do /*遍历每个节点*/
② 随机为每个节点分配工作节点、验证节点、矿工节点;
③ end for
④ for $R_i, i \in N$ do /*通信轮次遍历*/
⑤ for 每个工作节点 i do /*本地模型训练并加密参数*/
⑥ 利用本地数据进行模型训练→局部模型参数 θ_i;
⑦ $paillierEncrypt(Pubkey, \theta_i) \rightarrow (\theta_i^{\text{encrypt}})$; /*参数加密*/
⑧ 计算工作节点 i 的奖励值 R_{w_i};
⑨ 创建交易 $Tr_{w_i} = (\theta_i^{\text{encrypt}}, L_{w_i}, R_{w_i})$;
⑩ 将计算得到的交易 Tr_{w_i} 发送给验证节点;
⑪ end for
⑫ for 每个验证节点 v do /*仅负责验证,不更新参数*/
⑬ while 接收到来自工作节点发送的交易结果 Tr_{w_i} do
⑭ 广播交易内容 Tr_{w_i}; /*每个验证节点可以收到所有工作节点的参数更新*/
⑮ if 验证 Tr_{w_i} 中的签名成功 then
⑯ 授予奖励 $R_{v_i}^{\text{verify}}$; /*第 1 类奖励:签名验证奖励*/
⑰ 提取交易中的 $\theta_i^{\text{encrypt}}$;
⑱ else
⑲ continue; /*签名验证失败,跳过当前交易*/
⑳ end if
㉑ $paillierEncrypt(Privkey, \theta_i^{\text{encrypt}}) \rightarrow (\theta_i^{\text{decrypt}})$; /*参数解密*/
㉒ 根据 $\theta_i^{\text{decrypt}}$ 进行验证投票得到投票结果 $v_i(\theta_i^{\text{decrypt}})$; /*通过阈值进行判断更新是否合法*/
㉓ if 投票结果为正 then
㉔ 授予奖励 $R_{v_i}^{\text{validate}}$; /*第 2 类奖励:投票验证奖励*/
㉕ end if
㉖ 计算验证节点 v 的奖励值 R_{v_i}; /*验证节点总奖励为 2 类奖励之和*/
㉗ 创建交易 $Tr_{v_i} = (R_{v_i}^{\text{verify}}, R_{v_i}^{\text{validate}}, v_i(\theta_i^{\text{decrypt}}), Tr_{w_i})$;
㉘ 将计算得到的交易 Tr_{v_i} 发送给矿工节点;
㉙ end while
㉚ end for
㉛ for 每个矿工节点 m do /*生成区块与全局模型更新*/
㉜ while 接收到来自验证者发送的交易结果 Tr_{v_i} do
㉝ if 验证结果有效 then
㉞ $PaillierDecrypt(Privkey, Tr_{v_i}(\theta_i^{\text{decrypt}}))$; /*参数解密*/
㉟ 授予奖励 R_{m_i} 并创建新区块;
㊱ 使用 POS 共识机制挖掘新区块 $block_i^m$;
㊲ 将新区块添加到区块链中;
㊳ 更新全局模型 G_{i+1},完成新区块的模型更新; /*对于合法的模型参数使用 FedAvg 更新全局模型参数*/
㊴ else
㊵ continue; /*验证结果无效,跳过当前交易*/
㊶ end if
㊷ end while
㊸ end for
㊹ end for

FedPPB 中基于 Paillier 同态加密的训练参数保护框架具体细节如算法 3 所示。基于功能角色的任务划分,FedPPB 算法架构可划分为 3 个主要部分:模型训练与参数加密、交易验证与参数解密以及区块创建与模型更新。具体而言,算法 3 的行⑤~⑪定义了工作节点的模型训练和参数加密流程;行⑫~㉚规定了验证节点的交易验证和参数解密机制;行㉛~㊸则明确了矿工节点的区块生成和模型更新规范。下文将针对这 3 个功能模块分别展开详细的技术阐述。

1)模型训练与参数加密

在算法 3 中,行⑤~⑪详细阐述了工作节点进行模型训练与参数加密的完整流程。在图 3 中的步骤

②完整展现了这一过程，系统依次遍历所有参与训练的工作节点，每个工作节点均独立执行后续操作，具体包括基于本地数据的模型训练、局部模型参数的加密处理，以及交易的创建与提交。这一流程不仅确保了每个工作节点在分布式训练中的独立性，同时也保障了数据隐私的安全性。

首先每个工作节点使用本地数据集在本地执行模型的训练过程，采用随机梯度下降方法（SGD）即式（14）更新模型参数θ_i。

$$\theta_i = \theta - \alpha \nabla L(\theta, D_i) \text{。} \tag{14}$$

其次，通过 Paillier 同态加密算法对模型参数加密处理，得到加密后的模型参数$\theta_i^{\text{encrypt}}$。在完成上述工作后，工作节点会根据本地训练量预先计算预期的奖励R_{w_i}，作为其参与联邦学习过程贡献的激励，最终实际分发的奖励仍然需要依据验证节点的投票结果进行调整。

$$R_{w_i} = Ne_{w_i} \times |train_w| \times r, \text{ if } \left[N_{v^+}\left(\theta_i^{\text{decrypt}}\right) \geqslant N_{v^-}\left(\theta_i^{\text{decrypt}}\right)\right], \tag{15}$$

$$R_{w_i} = 0, \text{ if } \left[N_{v^+}\left(\theta_i^{\text{decrypt}}\right) < N_{v^-}\left(\theta_i^{\text{decrypt}}\right)\right] \text{。}$$

式（15）为工作节点的奖励计算公式，其中Ne_{w_i}表示在第i轮通信轮次R_i中局部训练时期的数量；$N_{v^+}(\theta_i^{\text{decrypt}})$为正向投票的数量；$N_{v^-}(\theta_i^{\text{decrypt}})$为负向投票的数量。工作节点在完成上述工作内容后，生成包含有$\theta_i^{\text{encrypt}}$，$L_{w_i}$，$R_{w_i}$的交易$Tr_{w_i}$，并将交易发送给验证节点。

2）交易验证与参数解密

算法 3 的行⑫~㉚为系统内验证节点的具体职责，即交易验证与参数解密过程。图 3 中的步骤③完整展现了这一过程，当验证节点接收到工作节点发送的交易Tr_{w_i}后，系统内的各验证节点之间会相互广播，以确保每一个验证节点均参与到验证流程中。

由于每个交易均由工作节点使用 Paillier 同态加密算法的私钥进行了签名以此保证交易内容的完整与真实性，因此验证节点首先验证交易的签名，当验证成功时，验证节点将通过式（16）计算获得签名验证奖励$R_{v_i}^{\text{verify}}$，并从验证成功的交易中提取本地模型的更新L_{w_i}来训练参数$\theta_i^{\text{encrypt}}$，此时这些训练参数处于加密状态，验证节点需要使用私钥并应用式（5）对其解密获得$\theta_i^{\text{decrypt}}$。随后验证节点将解密后的训练参数应用到当前的全局模型，使用本地验证数据集对更新后的模型进行评估，检查模型准确率等性能指标，并将更新后的模型性能与更新前的模型性能进行比较，判断参数更新的有效性。验证节点依据上述的验证结果进行投票，当训练参数合法且模型的性能达到预期则对交易内容投正向票并给予投票验证奖励$R_{v_i}^{\text{validate}}$，反之则为负向票，如计算式（17）。将通过式（16）计算出来的验证节点的签名验证奖励和通过式（17）计算出来的验证节点的投票验证奖励通过式（18）进行相加得到验证节点的总奖励值。完成上述验证与投票后将投票结果$v_i(\theta_i^{\text{decrypt}})$、验证奖励$R_{v_i}$以及工作节点的交易内容$Tr_{w_i}$汇总并创建生成交易$Tr_{v_i}$，最后将交易内容发送给矿工节点。

$$R_{v_i}^{\text{verify}} = \left\{Tr_{w_i}\right\} \times r, \tag{16}$$

$$R_{v_i}^{\text{validate}} = \left\{v_i\left(\theta_i^{\text{decrypt}}\right)\right\} \times r, \tag{17}$$

$$R_{v_i} = R_{v_i}^{\text{verify}} + R_{v_i}^{\text{validate}} \text{。} \tag{18}$$

3）区块创建与模型更新

在 FedPPB 中，矿工节点承担区块链的创建以及全局模型更新的关键职责，对应于算法 3 的行㉛~㊸。在图 3 中的步骤④完整展现了这一过程。

在这一过程中，首先，矿工节点接收来自验证节点发送的交易Tr_{v_i}，这些交易包含加密模型参数的验证奖励及投票结果等内容，并且每个矿工之间相互广播。接收完成后，使用 Paillier 同态加密算法的私钥对交易中的加密训练参数进行解密操作，确保只有经过验证的模型参数才会被用于全局模型的更新过程，从而有效防止恶意节点提交的无效或有害参数对全局模型造成影响。解密操作完成后，矿工节点进一步检验验证节点对模型训练参数的投票结果，以确保每个参数均为合法且通过验证者的验证。若验证结果有效，矿工节点将继续处理这些交易并获取相应的验证奖励R_{m_i}，如计算式（19）所示，反之若验证结果无效，则矿工节点将丢弃这些交易。

$$R_{m_i} = \left|\left\{Tr_{v_i}\left(\theta_i^{\text{decrypt}}\right)\right\}\right| \times r \text{。} \tag{19}$$

当矿工节点确认了交易的有效性后，会将其打包成候选区块。生成的候选区块包含加密的模型训练参数、验证节点对训练参数的投票结果以及各角色在完成各自任务后所获得的奖励。最终新区块的生成依据 PoS 共识机制来确定本轮中生成区块的矿工节点，具体而言，对基于矿工节点的权益权重进行加权随机选择，权益越高的矿工节点被选中的概率越大。其中，矿工节点的权益S_i由币龄值、历史贡献和任务奖励 3 部分构成。币龄值的计算如式（20）所示，币龄值越大，节点生成新区块的概率越高。

$$coinage = num \times time, \tag{20}$$

其中$coinage$为币龄，num为币的个数，$time$为代币的

使用时间。

历史贡献则根据矿工节点生成区块的次数以及验证交易的准确性来衡量。任务奖励为矿工节点完成任务所获得的奖励R_{m_i}。依据这3个指标进一步确定矿工节点的权益值S_i，如式(21)所示，每个矿工节点被选中的概率与权益权重成正比。

$$S_i=\alpha\times coinage+\beta\times contribution+\gamma\times R_{m_i},$$
$$P_i=\frac{S_i}{\sum_i S_i}。\tag{21}$$

为确保生成的区块不被篡改，矿工节点将利用自身的私钥对新区块进行签名操作，以此保障区块的完整性与真实性。在完成上述流程后，新区块将被正式添加至区块链中。随后，系统将依据区块中提取的经过验证的训练参数，与当前的全局模型进行聚合操作以生成新的全局模型。最终，完成聚合的全局模型将被广播至系统内的所有节点。

4 安全性分析

4.1 Paillier同态加密算法安全性分析

FedPPB提出的基于区块链的联邦学习系统中，各工作节点在完成本地模型训练后，采用Paillier同态加密算法对训练得到的模型参数进行加密处理。随后，这些经过加密的训练参数被传输至验证节点进行验证。类似地，矿工节点从验证节点接收的封装交易信息亦为加密后的结果。在此过程中，工作节点利用公钥对训练参数进行加密，而验证节点和矿工节点则借助私钥执行部分解密操作。鉴于Paillier加密系统具备语义安全性，本文所构建的基于区块链的联邦学习系统通过引入训练参数加密机制有效保障了训练参数在传输过程中的隐私性。

1) 安全目标设定

FedPPB系统中的Paillier同态加密算法的核心目标在于保护联邦学习过程中训练参数的隐私性与完整性，具体包括3个方面：

①训练参数的隐私保护。确保本地模型训练参数在传输过程中不被恶意节点或外部攻击者窃取或逆向推导，防止敏感数据的泄露。

②抵御恶意行为。通过加密机制和角色隔离的设计，有效防范恶意节点伪造、篡改或注入无效参数，确保全局模型更新的合法性与可信性。

③语义安全性。基于Paillier加密算法的语义安全特性保证加密后的模型训练参数在传输和验证过程中无法被区分或破解，即使敌手截获密文也无法获取明文信息。

2) 安全性证明

FedPPB中提出的训练参数加密机制的安全性依赖于Paillier加密算法的使用。如果该加密系统的语义安全性得到保证，那么本文提出的基于区块链的联邦学习系统也将具有语义安全性。

假设存在一个敌手$\mathcal{A}$，能够破坏Paillier加密系统的语义安全性，从而获取私有数据。

步骤1. 初始化：系统生成公钥和私钥对(*Pubkey*, *Prikey*)，并将公钥分发给所有的工作者节点。

步骤2. 本地模型训练：每个工作节点使用本地数据训练本地模型L_{w_i}，并生成训练参数θ_i。

步骤3. 加密传输：工作节点使用公钥对生成的训练参数进行加密，使用式(22)得到$\theta_i^{\text{encrypt}}$，然后将其传输给验证节点。

$$\theta_i^{\text{encrypt}}=E(\theta_i,N,g)=g^{\theta_i}\times r^{\lambda}\bmod N^2。\tag{22}$$

步骤4. 验证选择：验证节点随机选择2组加密的训练参数，分别记为$\theta_1^{\text{encrypt}}$, $\theta_2^{\text{encrypt}}$。这些参数可能来自不同的工作节点。

步骤5. 解密验证：验证节点使用私钥对接收到的加密训练参数进行解密，使用式(23)获得明文$\theta_i^{\text{decrypt}}$，并随后验证其合法性。

$$\theta_i^{\text{decrypt}}=D\left(\theta_i^{\text{encrypt}},\lambda,N\right)=L((\theta_i^{\text{encrypt}})^{\lambda}\bmod N^2)\times\mu\bmod N。\tag{23}$$

步骤6. 猜测挑战：敌手$\mathcal{A}$试图区分$\theta_1^{\text{encrypt}}$和$\theta_2^{\text{encrypt}}$是否来自$\theta_1$或$\theta_2$的加密训练参数。

步骤7. 构建敌手：如果敌手$\mathcal{A}$无法以不可忽略的优势ε区分$\theta_1^{\text{encrypt}}$和$\theta_2^{\text{encrypt}}$，则系统在传输过程中保持了隐私性。构造一个新的敌手$\mathcal{A}'$，$\mathcal{A}'$可以利用敌手$\mathcal{A}$来破坏Paillier加密方案的语义安全性。敌手$\mathcal{A}'$的攻击策略如下：

① $\mathcal{A}'$接受$\mathcal{A}$的挑战，并能够获取Paillier的公钥*Pubkey*。

② 敌手$\mathcal{A}'$生成2个不同的合法模型训练参数θ_1和θ_2，然后通过式(24)(25)对这2个训练参数进行加密，得到$\theta_1^{\text{encrypt}}$, $\theta_2^{\text{encrypt}}$，并将这2个加密参数发送给$\mathcal{A}$。

$$\theta_1^{\text{encrypt}}=E(\theta_1,N,g)=g^{\theta_1}\times r^{\lambda}\bmod N^2,\tag{24}$$

$$\theta_2^{\text{encrypt}}=E(\theta_2,N,g)=g^{\theta_2}\times r^{\lambda}\bmod N^2。\tag{25}$$

步骤8. 敌手$\mathcal{A}$试图区分$\theta_1^{\text{encrypt}}$和$\theta_2^{\text{encrypt}}$，并随后输出一个猜测值$b$，表示$\theta_1^{\text{encrypt}}$和$\theta_2^{\text{encrypt}}$是否来自$\theta_1$或$\theta_2$。

步骤9. 如果敌手$\mathcal{A}$能够以优势ε区分$\theta_1^{\text{encrypt}}$和

$\theta_2^{\text{encrypt}}$, 那么敌手$\mathcal{A}'$能够以相同的优势$\varepsilon$对$b$的来源进行猜测。

步骤 10.根据角色符合决策复合剩余假设(decisional composite residuosity assumption, DCRA)：即使提供了 2 个随机选择的元素g^a和g^b, 敌手也无法在多项式时间内区分g^{a+b}和g^r, 即不存在任何敌手$\mathcal{A}'$能够以不可忽略的优势区分 Paillier 加密, 其中$a+b=r \bmod n$。

因此, 敌手$\mathcal{A}$也无法以不可忽略的优势ε区分加密的合法模型。FedPPB 中的 Paillier 加密算法通过加密训练参数传输与角色隔离机制, 确保敌手即使截获密文也无法推断明文信息或破坏系统的完整性。

4.2 PSO 优化算法安全性分析

本节从理论分析的角度对 FedPPB 中的 PSO 优化算法的安全性进行系统性论证, 下文将从符号定义与假设、加密机制的安全性、角色分配优化的安全性约束和动态分配过程中的抗攻击性 4 个方面详细阐述。

4.2.1 符号定义与假设

1)系统参数

①设备总数为N, 角色分配方案为(W,V,M)并满足$W+V+M=N$;

②最低验证设备数为$V_{\min}$;

③使用 Paillier 加密密钥对$(Pubkey, Prikey)$, 其中$Pubkey=(N,g)$, $Prikey=\lambda$;

2)敌手能力

①设置敌手$\mathcal{A}$控制不超过f个恶意节点($f<V_{\min}$);

② $\mathcal{A}$可窃听通信链路, 但无法破解 Paillier 加密或获取私钥;

3)安全目标

①数据隐私性: 敌手无法从加密参数$\theta_i^{\text{encrypt}}$推断出明文$\theta_i$;

②模型完整性: 验证者将检测并拒绝恶意设备所提交的伪造参数。

4.2.2 加密机制的安全性

在本文提出的系统中, Paillier 同态加密算法被用于对模型训练过程中所传输的参数进行加密处理, 以确保参数的安全性。Paillier 算法的安全性基于 DCRA。根据 4.1 节所提供的 Paillier 加密算法的安全性证明, 该算法对任意概率多项式时间敌手$\mathcal{A}$均满足 IND-CPA 安全, 即式(26)。

$$|Pr[\mathcal{A}(E(\theta_0))=1]-Pr[\mathcal{A}(E(\theta_1))=1]| \leqslant negl(k), \tag{26}$$

其中k为安全系数, $negl(k)$为可忽略函数。

4.2.3 角色分配优化的安全性约束

在本节中提出设置验证节点的数量下限用于确保系统面对恶意设备时的容错能力, 具体而言, 如果 PSO 优化满足$V \geqslant V_{\min}$这一条件, 那么系统则可以最多容纳$f=\lfloor (V-1)/3 \rfloor$个恶意验证节点。

证明. 根据 PBFL 容错条件$V \geqslant 3f+1$, 可以解得$f \leqslant (V-1)/3$; 由于f为整数, 因此我们取$f=\lfloor (V-1)/3 \rfloor$。 证毕.

4.2.4 动态分配过程的抗攻击性

假设 PSO 优化算法的适应度函数$f(W,V,M)$满足以下条件, 则敌手$\mathcal{A}$无法通过伪造适应度值影响 PSO 输出的角色分配方案:

1)输入数据的完整性。准确率与系统开销均通过区块链进行验证, 不可篡改。

2)异常检测。通过 Z-score 基于统计学的原理进行异常数据的过滤操作($|x-\mu|>3\sigma$时将异常数据丢弃)。

3)隐私保护。对适应度函数添加拉普拉斯噪声$\eta \sim Lap(\Delta f/\varepsilon)$。

下面将分别证明上述 3 个假设:

1)区块链技术具有不可篡改的特性因此可以保证输入数据的真实性。

2)使用 Z-score 方法进行异常值检测, 该方法用于衡量数据点与均值的偏差程度$Z=\dfrac{x-\mu}{\sigma}$, 当$|Z|>3$时, 则判定数据点x为异常并丢弃。基于正态分布假设, 99.7% 的数据位于$\mu \pm 3\sigma$内。

3)添加拉普拉斯噪声后的适应度函数$\widetilde{f}$满足(ε,δ)-差分隐私, 即对任意的数据集D_1和D_2以及可能的输出角色数量分配方案(W, V, M)都满足式(27)。

$$Pr[\mathcal{A}(\widetilde{f}(D_1))=(W,V,M)] \leqslant e^{\varepsilon} Pr[\mathcal{A}(\widetilde{f}(D_2))=(W,V,M)]+\delta。 \tag{27}$$

其中ε为隐私预算, 控制着隐私保护的强度, 隐私预算的值越小, 隐私保护越强; δ表示失败的概率, 允许在极少数的情况下发生隐私泄露。

首先, 我们定义 2 个角色分配方案D_1和D_2, 这些方案仅在 1 个节点的角色分配上存在差异, 而其余节点的角色分配和训练数据均保持一致。具体而言D_1中某个节点为工作者而在D_2中这一节点为验证者。

其次, 使用式(28)计算适应度函数的敏感度Δf, 即在D_1和D_2数据集上适应度函数输出的最大差异。根据所设计的适应度函数(式(11))可知单个节点的角色变化对于准确率、通信时间以及验证者数量变化所带来的安全性惩罚差异具有一定的影响, 因此将全局敏感度设置为式(29)。

$$\Delta f = \max_{D_1,D_2} |f(D_1) - f(D_2)|, \tag{28}$$

$$\Delta f = \alpha \times \Delta Accuracy + \beta \times \Delta t + \gamma \times \Delta SafetyPenalty 。 \tag{29}$$

然后, 采用拉普拉斯机制, 向适应度函数中添加拉普拉斯噪声(式(8)), 其中噪声尺度为$b = \Delta f/\varepsilon$, 拉普拉斯分布满足的概率密度函数如式(30)所示。

$$Lap(x|b) = \frac{1}{2b} e^{-\frac{|x|}{b}} 。 \tag{30}$$

最后, 证明适应度函数$\widetilde{f}$满足(ε,δ)-差分隐私。对于任意的输出s, 添加噪声后的概率比如式(31)所示。鉴于$|f(D_1) - f(D_2)| \leqslant \Delta f$, 根据式(32)可得, 概率比满足式(33), 这表明添加噪声后的适应度函数满足ε-差分隐私($\delta = 0$)。由此可知, 敌手无法通过观察$\widetilde{f}$区分D_1和D_2, 满足隐私保护的目标。

$$\frac{Pr\left[\widetilde{f}(D_1) = s\right]}{Pr\left[\widetilde{f}(D_2) = s\right]} = \frac{e^{-\frac{|s-f(D_1)|}{b}}}{e^{-\frac{|s-f(D_2)|}{b}}} = e^{\frac{|s-f(D_2)|-|s-f(D_1)|}{b}}, \tag{31}$$

$$|s - f(D_2)| - |s - f(D_1)| \leqslant |f(D_1) - f(D_2)| \leqslant \Delta f, \tag{32}$$

$$\frac{Pr\left[\widetilde{f}(D_1) = s\right]}{Pr\left[\widetilde{f}(D_2) = s\right]} \leqslant e^{\frac{\Delta f}{b}} = e^{\varepsilon} 。 \tag{33}$$

综上, 通过对3个方面假设展开全面的理论分析, 能够明确本节所提出的FedPPB框架中的PSO优化算法切实保障了角色分配的安全性。此外, 框架中所采用的噪声机制进一步强化了系统安全性, 为整个系统构建起多重安全防护体系, 有力提升了系统在面对各类安全挑战时的稳健性与可靠性。

5 实验结果与分析

5.1 实验平台与数据集

本文的实验旨在通过系统化的角色分配和恶意设备数量的动态调整, 深入分析分布式训练场景下模型的性能表现。实验在1台设备(Intel® Gold 6326 CPU(主频2.90 GHz))和NVIDIA GTX 3090 Ti的高性能计算设备上进行。软件环境包括Python 3.7.0和Pytorch 1.13.1, 采用3个广泛使用的数据集MNIST, Fashion-MNIST, CIFAR-10用于验证模型的有效性。

MNIST数据集由Lecun等人[52]在1998年提出, 包含了10个种类的手写数字, 共计70 000个样本。在该数据集中, 有60 000个样本被用于训练数据, 另外10 000个样本则被用作测试数据。MNIST数据集作为评估机器学习模型性能的标准参考, 已被广泛地应用于各类实验研究中。

Fashion-MNIST数据集由Xiao等人[53]提出, 其结构与MNIST数据集相似, 包括图像大小、格式以及训练和测试数据集的划分。该数据集涵盖了70 000张28×28像素的灰度图像, 这些图像代表了10种不同的时尚产品类别, 并且来源于不同的性别分类: 男性、女性、儿童以及中性。每个类别下均包含7 000张图像。Fashion-MNIST数据集同样被分为训练集和测试集, 分别包含60 000张和10 000张图像。

为进一步验证所提方法在更复杂场景下的泛化能力, 我们还引入了CIFAR-10数据集[54]。CIFAR-10是一个广泛用于评估图像识别算法的彩色图像数据集, 包含10个类别的32×32像素3通道图像, 其数据复杂度和类内差异性远高于MNIST和Fashion-MNIST, 对模型的鲁棒性和有效性提出了更高要求。

1)参数设置。在模型验证阶段, 本研究设计了一个包含100轮通信的分布式训练实验来模拟真实场景下的联邦学习过程。实验环境配置了20台虚拟计算设备, 各节点的本地训练参数经过优化设定: 批量大小为10, 学习率固定为0.01, 每个通信周期内执行5轮本地训练迭代。这些超参数的设置经过严格测试, 能够在保证训练效率的同时实现模型的稳定收敛。此外, 在安全性测试方面, 实验设置了3种恶意设备的数量, 分别模拟无攻击(存在0%的恶意设备)、中等攻击(存在15%的恶意设备)和强攻击(存在25%的恶意设备)场景。通过动态调整恶意设备数量全面评估模型在不同威胁等级下的防御能力, 从而更全面地评估模型的鲁棒性。表4为本文实验中的超参数的设置。

2)对比方法。为了系统性地评估FedPPB框架在不同恶意设备数量渗透水平下的性能表现, 我们进行了全面且公平的模型准确率对比实验。具体而言, 实验将FedPPB与普通联邦学习(vanilla FL)和验证区块链的联邦学习(VBFL)进行对比, 结果表明FedPPB在安全性方面具有显著优势。通过控制角色分配和恶意节点数量等变量, 验证了FedPPB在保证准确率的同时有效提升了系统的鲁棒性。下面将详细介绍具有代表性的2个基线模型。

1)vanilla FL方法[17]的核心理念在于将训练数据保留在客户端, 从而在保护数据隐私的同时实现分布式训练。该方法通过聚合本地计算的更新来促进共享模型的学习, 为联邦学习的基本架构奠定了基础。

2)VBFL方法[7]则是在联邦学习领域中提出的一个增强框架。该框架通过整合区块链技术, 有效解决了传统联邦学习场景中集中式架构所面临的漏

Table 4 Experimental Parameters Setting

表 4 实验参数设置

参数	赋值
设备数量	20
批量大小	10
学习率	0.01
本地训练轮数	5
粒子数S	30
惯性权重ω	0.8
适应度函数权重α,β,γ	$\alpha=1.0$，$\beta=0.5$，$\gamma=10.0$
迭代次数T	50
学习因子c_1,c_2	$c_1=c_2=2.0$
最小验证者数量	$V_{\min}$
恶意节点比率ρ	0%，15%，25%

洞问题。这些漏洞主要包括单点故障的易感性以及无法验证本地模型合法性的问题。通过引入区块链技术，VBFL 能够防止恶意设备通过恶意行为破坏模型训练过程，从而显著提升了联邦学习系统的安全性和可靠性。

5.2 模型结构与评价方法

由于 MNIST 和 Fashion-MNIST 数据集的通道数是相同的，所以我们设计了相同的网络结构。该网络模型由 3 个卷积层（卷积核大小是 5×5，第 1 层输入通道为 1、输出通道为 32，第 2 层输入通道为 32、输出通道为 64，第 3 层输入通道为 64、输出通道为 64）、3 个池化层（池化窗口大小为 2×2）、2 个全连接层（第 1 个全连接层神经元个数为 512，第 2 个全连接层神经元个数为 10，激活函数为 ReLU）组成。对于 CIFAR-10 数据集，由于其为 3 通道彩色图像，我们将网络模型第 1 个卷积层的输入通道数相应地调整为 3，其余网络结构的输入通道数和超参数保持一致，以确保对比的公平性。由于 MNIST，Fashion-MNIST 和 CIFAR-10 数据集都是 10 分类问题，本文采用测试准确率（test accuracy）作为评价指标。对于系统开销，我们在 5.4 节消融研究中通过区块的生成时间来进行对比。

1）模型鲁棒性分析

为了全面评估 FedPPB 模型在面对恶意节点攻击时的鲁棒性，本节从恶意节点数量的角度展开研究，实验分别设置恶意设备数量为 0，3，5，即占所有设备总数的 0%，15%，25%。通过逐步增加恶意设备的数量，系统地检验 FedPPB 方法在聚合全局模型方面的有效性。

设计 2 组实验，每组包含 100 轮通信，以确保结果的稳定性和可靠性。在每轮通信结束后，由 20 台设备共同生成的全局模型精度被记录下来。为了模拟分布式训练环境，实验将这 20 台设备划分为 3 种角色：工作节点、验证节点和矿工节点。在 FedPPB 中，采用了基于 PSO 粒子群优化算法的动态分配方案。该方案依据每种角色所设定的任务评估指标自动将实验中预先设置的 20 个设备合理分配至 3 种不同的角色类别。借助这种自动化的最优分配策略，我们得以高效地模拟区块链联邦学习系统中的分工协作机制，进而对 PSO 优化算法在提升系统运行效率和增强系统鲁棒性方面的实际效用进行验证。

图 5 分别呈现了在 MNIST 和 Fashion-MNIST 这 2 个具有代表性数据集上，FedPPB 模型于全局模型训练进程中的准确率动态变化情况。实验所得结果有力地表明，即使面临恶意节点数量逐步递增的挑战时，FedPPB 模型依旧能够保持相对较高的准确率水平。

具体来看，当恶意节点数量为 0，即处于无恶意节点干扰的理想状态时，FedPPB 模型在 MNIST 数据集上实现了 98.72% 的准确率，在 Fashion-MNIST 数据集上的准确率则达到了 96.51%。随着恶意节点数量增加至 3 个（占总设备数量的 15%），模型在 MNIST 数据集上的准确率下降至 96.81%，在 Fashion-MNIST 数据集上的准确率降至 86.23%。当恶意节点数量进一步攀升至 5 个（占总设备数量的 25%）时，模型在 MNIST 数据集上的准确率为 87.59%，而在 Fashion-

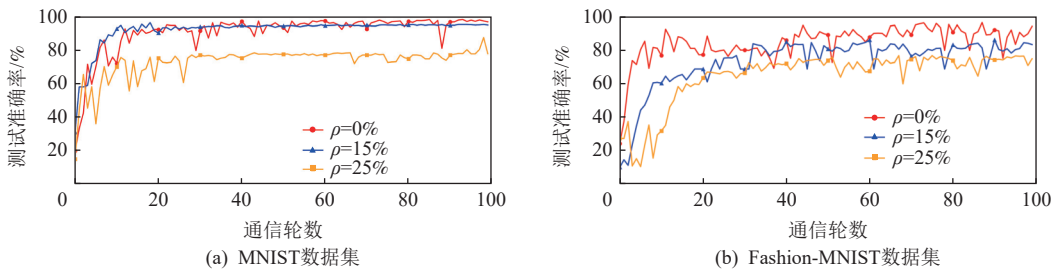


Fig. 5 Test accuracy under different numbers of malicious nodes across different datasets

图 5 不同数据集在不同恶意节点数量下的测试准确率

MNIST 数据集上的准确率则为 76.65%。这些数据直观地反映了模型在不同恶意节点干扰程度下的性能表现, 进一步验证了 PSO 算法优化策略的有效性。

实验结果表明, 2 个数据集的训练性能存在一定差异, 其中 Fashion-MNIST 的准确率略低于 MNIST, 这一差异主要来源于 Fashion-MNIST 的数据集包含的 10 类时尚物品具有更高的分类复杂度。可以看出, 即便面对更复杂的分类任务, FedPPB 仍能保持与 MNIST 相近的相对准确率水平, 这说明加密机制对于模型收敛的影响与数据分布无关, 且加密传输过程并未损害模型的特征学习能力。

PSO 算法对模型角色分配的动态优化以及噪声机制也起到了作用。首先, 通过适应度函数中的模型准确率指标, 系统能够优先选择计算能力强、数据质量高的设备作为工作节点, 从而显著提升局部模型的质量。其次, 验证节点数量的动态调整有效降低了恶意参数通过验证的概率, 显著减少了攻击对系统的影响。最后, 将矿工节点生成区块的时间作为优化目标, 不仅缩短了全局模型的更新延迟, 还加速了模型的整体收敛速度。此外, 适应度函数中引入的噪声机制虽然可能轻微影响模型精度, 但能有效抵御恶意设备的攻击, 使系统在存在 15% 甚至 25% 恶意设备的情况下仍能保持较高的准确率。

2)模型比较分析

为系统性地评估 FedPPB 在不同复杂度数据集上的性能, 我们在 MNIST, Fashion-MNIST, CIFAR-10 上, 将其与 vanilla FL 和 VBFL 方法在不同ρ下进行了对比分析, 平均测试准确率结果如表 5 所示。

如图 6(a)所示, 这张图清晰地展示了在没有恶意设备的情况下, 采用 FedPPB 方法构建的全局模型的准确率与 vanilla FL 和 VBFL 相当, 这表明 Paillier 加密算法和 PSO 优化算法的引入并未对模型的整体性能产生负面影响。而图 6(b)展示了 Fashion-MNIST 数据集下在没有恶意设备的情况下, FedPPB 优于其他的方法。

当恶意设备数量占总设备数量的 15% 时, 如图 6(c)(d)所示, 即存在 3 个恶意设备, FedPPB 模型的准确率分别达到了 96.81% 和 86.23%。在 MNIST 数据集下, FedPPB 实验曲线比 VBFL 方法的实验曲线更加平滑; 而在 Fashion-MNIST 数据集下, FedPPB 显著优于 VBFL 和 vanilla FL。这一结果表明, FedPPB 在面对恶意设备攻击时能够保持较高的模型准确率。

进一步的实验结果如图 6(e)(f)所示, 当恶意设备的比例增加到总设备数量的 25%, 即存在 5 个恶意设备时, VBFL 在 2 个数据集的准确率分别急剧下降至 10.60% 和 12.82%, 而 FedPPB 的准确率仍稳定维持在 87.59% 和 76.65%。这一对比结果充分证明 FedPPB 在面对恶意设备攻击时的优越性能。由于在ρ=15% 时, vanillaFL 的准确率急剧下降, 所以在ρ=25% 的实验中我们没有和 vanilla FL 进行对比。通过采用 PSO 优化算法进行动态角色分配和 Paillier 同态加密算法对训练参数进行加密, FedPPB 不仅确保了训练参数的安全性, 还有效地抵御了恶意节点对模型性能的潜在破坏。表 5 展示了在这些设置下, FedPPB 与基线方法在测试集上的分类准确率对比。从表 5 可以看出, 在 MNIST 和 Fashion-MNIST 数据集上, FedPPB 在ρ=0%, 15%, 25% 下均取得了最高的分类准确率。随着ρ的增加, 所有方法的准确率有所下降, 然而, FedPPB 在ρ=15% 和ρ=25% 下仍能保持较高的性能, FedPPB 显著高于基线方法, 表明其在对抗恶意干扰方面的有效性和稳定性。这些结果验证 FedPPB 显著高于基线方法, 表明其在对抗恶意干扰方面的有效性和稳定性。这些结果验证了 FedPPB 在分布式学习环境中处理恶意客户端的能力, 相较于现有方法具有显著优势。

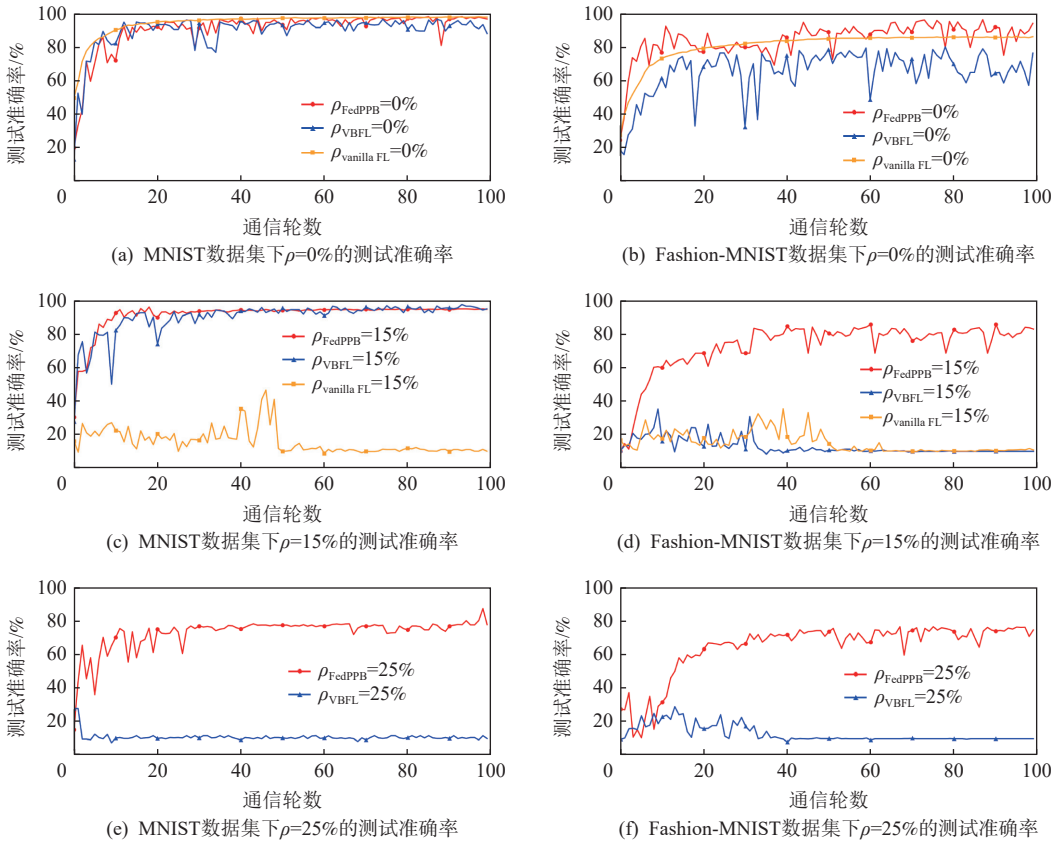
Table 5 Mean Test Accuracy for Different Methods on MNIST, Fashion-MNIST, and CIFAR-10 Datasets

表 5 不同方法在 MNIST, Fashion-MNIST, CIFAR-10 数据集上的平均测试准确率 %

方法	数据集	ρ=0%	ρ=15%	ρ=25%
vanilla FL	MNIST	90.78±0.12	16.05±0.07	-
	Fashion-MNIST	80.82±0.10	15.13±0.06	-
	CIFAR-10	60.84±0.18	17.36±0.08	-
VBFL	MNIST	89.72±0.11	90.40±0.10	10.60±0.03
	Fashion-MNIST	62.25±0.13	12.88±0.05	12.82±0.05
	CIFAR-10	58.12±0.16	11.45±0.11	9.85±0.06
FedPPB (本文)	MNIST	**98.72±0.07**	**96.81±0.09**	**87.59±0.09**
	Fashion-MNIST	**96.51±0.10**	**86.23±0.15**	**76.65±0.16**
	CIFAR-10	**82.65±0.13**	**78.92±0.15**	**69.28±0.17**

注: 黑体数值表示最佳结果; "-"表示模型训练崩溃。

在较为简单的 MNIST 与 Fashion-MNIST 数据集上, FedPPB 已展现出显著优势。当我们将实验扩展到更具挑战性的 CIFAR-10 数据集上时, FedPPB 的优越性得到了进一步的印证。在ρ=0% 时, FedPPB 的准确率达到 82.65%, 已优于基线方法。尤为关键的是, 当ρ提升至 25% 时, VBFL 的准确率已崩溃至 9.85%, 而 FedPPB 仍能维持较高准确率(69.28%)。这一结果强有力地证明, 我们所提出的结合 PSO 优化和



(a) MNIST数据集下ρ=0%的测试准确率　(b) Fashion-MNIST数据集下ρ=0%的测试准确率

(c) MNIST数据集下ρ=15%的测试准确率　(d) Fashion-MNIST数据集下ρ=15%的测试准确率

(e) MNIST数据集下ρ=25%的测试准确率　(f) Fashion-MNIST数据集下ρ=25%的测试准确率

Fig. 6　Test accuracy comparison of various methods on different datasets

图 6　不同数据集下各方法的测试准确率对比

Paillier 加密的框架，不仅在简单任务上有效，在处理复杂数据和抵御高强度恶意攻击时具备更卓越的鲁棒性和泛化能力。

5.3　动态网络环境适应性分析

为进一步检验 FedPPB 框架在真实世界部署中应对网络拓扑动态变化的能力，本节设计了一系列节点动态加入与退出的场景。我们引入了 FedPPB+，这是对原 FedPPB 框架的增强，其核心在于 PSO 优化前增加了一个网络拓扑动态感知机制，能主动扫描并更新当前节点总数N，从而实现对节点变化的端到端自适应。实验在一个包含 20 个节点且ρ=15% 的系统中进行，将 FedPPB+与 FedPPB 及静态分配方案（w/o PSO）进行对比。

为清晰展示并量化该自适应能力，我们的实验在训练中期（如第 40 轮通信）引入了 1 次占节点总数 15%（即 3 个节点）的拓扑突变。我们做出如下设计考量：

1）扰动时机。选择在训练中期施加扰动，旨在评估模型在已初步收敛但尚未完全稳定时的动态调整与恢复能力。相较于训练初期（模型尚未稳定）或末期（模型已过拟合或收敛），中期的扰动更能有效检验算法的适应性与鲁棒性。

2）扰动幅度。选择 15% 的节点数量变化，旨在构成一个对系统稳定性的挑战，其规模足以对 w/o PSO 造成显著性能冲击，从而凸显动态适应机制的必要性与优越性。

3）节点退出场景分析。该实验模拟网络规模收缩，在第 40 轮通信时从系统中移除 3 个节点（占节点数量总数的 15%）。如图 7（a）所示，节点退出时，w/o PSO 的准确率出现断崖式下跌，因其固定的角色分配无法适应资源池的减少。FedPPB 框架的准确率也受到一定冲击，但仍优于 w/o PSO，这得益于 PSO 算法会尝试在剩余节点中寻找最优角色配比。相比之下，FedPPB+在感知到节点总数变化后，迅速调整了优化搜索空间，准确率仅出现轻微波动便快速恢复至高位，展现出卓越的容错能力。

从系统效率角度来看，如图 7（b）所示，节点退出导致 w/o PSO 的区块生成时间剧烈波动且显著增加。FedPPB 的区块生成时间也有所上升。而 FedPPB+通过动态重新分配矿工角色，有效地将区块生成时间稳定在较低水平，确保了系统在节点退出后的高效运行。

3）节点加入场景分析

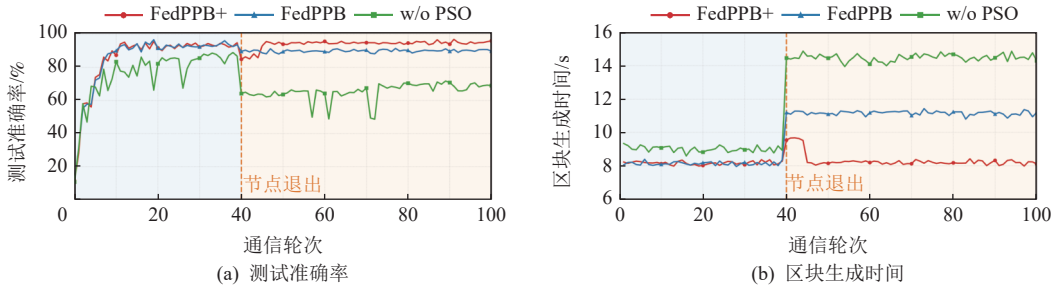


Fig. 7 Comparison of test accuracy and block generation time of various methods under node dropout scenario

图 7 节点退出场景下各方法的测试准确率与区块生成时间对比

该实验通过在第 40 轮通信时向系统注入 3 个新节点(占节点总数的 15%)来衡量框架的可扩展性。如图 8(a)所示, FedPPB+在节点加入后, 准确率曲线相较于 FedPPB 和 w/o PSO 呈现出更优的性能和更快的收敛趋势。这是因为 FedPPB+能够立即将新节点纳入资源池并参与 PSO 优化, 可以将计算能力更强的新节点分配为工作节点, 从而提升模型训练效果。而 FedPPB 和 w/o PSO 无法利用这些新资源, 其性能曲线在节点加入后未表现出任何增益。

如图 8(b)所示, FedPPB+在识别到新节点后, 通过优化角色分配, 区块生成时间甚至呈现小幅下降趋势, 验证了其高效的资源利用能力。其余 2 种方案的区块生成时间则保持在原有水平, 未能从网络扩张中获益。

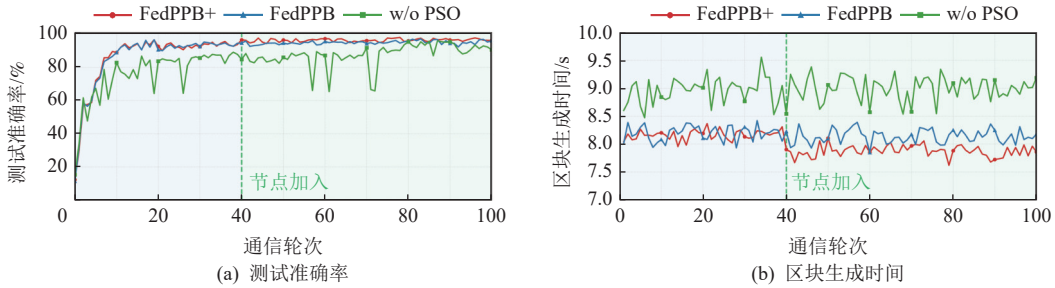


Fig. 8 Comparison of test accuracy and block generation time of various methods under node joining scenario

图 8 节点加入场景下各方法的测试准确率与区块生成时间对比

4) 连续拓扑变化场景分析

为模拟更复杂的、连续变化的动态环境, 本实验设计了在不同训练阶段的连续拓扑扰动: 在训练前期(第 30 轮)模拟网络扩张(加入 3 个节点), 并在训练后期(第 60 轮)模拟网络收缩(移除 3 个节点)。这种设计旨在全面评估 FedPPB+在整个训练生命周期中, 面对连续网络拓扑变化时的长期稳定性与持续自适应能力。如图 9 所示, 在整个动态变化过程中, FedPPB+的测试准确率和区块生成时间始终保持最优且最稳定。无论网络扩张还是收缩, FedPPB+均能迅速自适应调整, 维持系统的高性能和高效率。相比之下, FedPPB 表现出一定的适应能力但稳定性不足, 而 w/o PSO 则在 2 次拓扑变化时均出现显著的性能恶化。

综合上述动态实验结果, 我们认为, 尽管 FedPPB 的 PSO 机制提供了角色分配的动态性, 但其性能仍

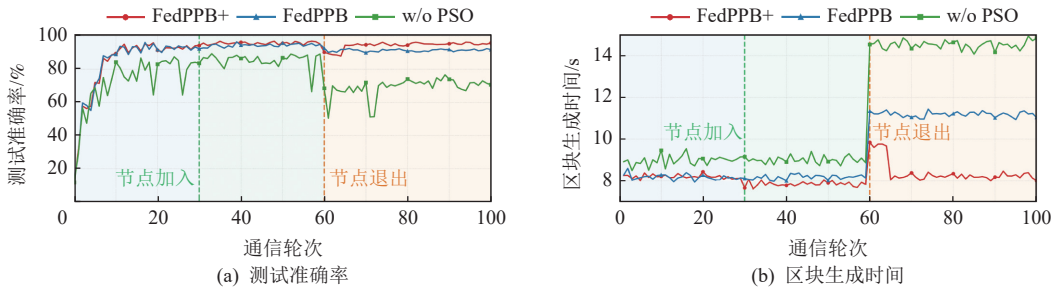


Fig. 9 Comparison of test accuracy and block generation time of various methods under continuous topology change scenario

图 9 连续拓扑变化场景下各方法的测试准确率与区块生成时间对比

受限于对固定网络规模的假设。通过引入网络拓扑动态感知机制，FedPPB+有效弥补了这一不足，在面对节点动态加入/退出时展现出强大的鲁棒性与适应性，这对于联邦学习系统在真实、多变的网络环境中实现长期稳定运行至关重要。

5.4 消融研究

1）PSO 和 Paillier 算法对模型性能的影响

为了评估 FedPPB 中 PSO 和 Paillier 加密算法的关键作用，我们通过消融研究，在 MNIST, Fashion-MNIST, CIFAR-10 数据集上测试了以下配置：①完整模型 FedPPB（包含 PSO 优化算法和 Paillier 加密算法）；②移除 PSO 的模型（仅使用 Paillier 加密算法），用 w/o PSO 来表示。

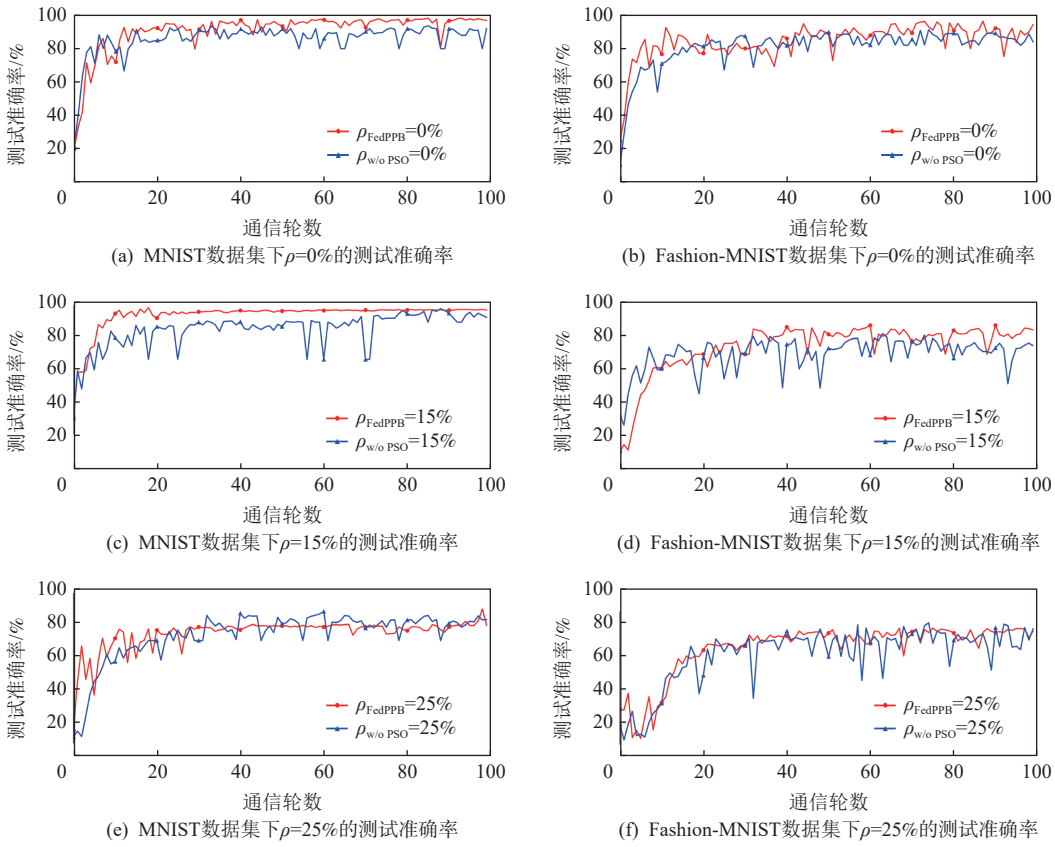
由于使用 PSO 优化算法可以动态调整角色的数量，在进行 w/o PSO 的实验时，我们采用固定的 20 个节点数量，其中工作节点数量为 12、验证节点数量为 5、矿工节点数量为 3。

如图 10(a)(b)所示，在 ρ=0% 时，使用 MNIST 数据集和 Fashion-MNIST 进行测试时，使用 PSO 的模型的收敛速度比不使用 PSO 的模型更快。准确率曲线表明，PSO 显著加速了收敛速度并提升了最终的分类准确率，相较于不使用 PSO 的模型来说，模型准确率提升了 6.94 个百分点。

如图 10(c)所示，在面对 ρ=15% 时的 MNIST 数据集，使用 PSO 的模型相比于不使用 PSO 的模型取得了显著优越的性能。使用 PSO 的方法在初始训练轮次中表现出更快的收敛速度，并在整个训练过程中保持更高的准确率。值得注意的是，不使用 PSO 的模型在训练过程中经历了几次急剧的准确率下降，表明其对恶意攻击的脆弱性；而使用 PSO 的模型则保持了更稳定的性能，波动幅度最小。

如图 10(d)所示，在更具挑战性的 Fashion-MNIST 数据集上观察到了类似的趋势。使用 PSO 的模型表现出对恶意节点攻击的增强鲁棒性，相比不使用 PSO 的方法实现了更高的准确率并展现出更大的稳定性。不使用 PSO 模型显示出更明显的性能退化和频繁的准确率振荡，特别是在训练后期阶段，这突出了 PSO 优化在对抗条件下维持模型性能的有效性。

图 10(c)(d)的对比分析揭示了 PSO 优化组件在增强联邦学习系统对恶意节点攻击的抵抗能力方面发挥了关键作用。在 2 个数据集上观察到的一致性验证了所提出的 PSO 集成在对抗参与者存在情况下



(a) MNIST数据集下ρ=0%的测试准确率　(b) Fashion-MNIST数据集下ρ=0%的测试准确率

(c) MNIST数据集下ρ=15%的测试准确率　(d) Fashion-MNIST数据集下ρ=15%的测试准确率

(e) MNIST数据集下ρ=25%的测试准确率　(f) Fashion-MNIST数据集下ρ=25%的测试准确率

Fig. 10　Comparison of test accuracy between FedPPB and w/o PSO on different datasets

图 10　不同数据集上 FedPPB 与 w/o PSO 的测试准确率对比

维持模型准确率和训练稳定性的有效性。

如图 10(e)(f)所示，在ρ=25% 的更严苛攻击环境下，MNIST 和 Fashion-MNIST 数据集中，PSO 增强模型相比于不使用 PSO 模型展现出了更加显著的性能优势。PSO 集成方法在训练初期实现了更快速的收敛，并在整个训练过程中维持了相对稳定的高准确率水平。不使用 PSO 模型在中后期训练阶段表现出较为明显的性能波动，而 PSO 增强模型则保持了更加平稳的训练曲线，体现出对增强恶意攻击的良好抵抗能力。

随着ρ从 15% 增加到 25%，攻击强度的提升使得 PSO 优化组件的重要性更加凸显。实验结果表明，PSO 不仅能够提升模型在正常条件下的性能，更重要的是显著增强了联邦学习系统在高强度对抗环境下的鲁棒性和稳定性。这一发现验证了 PSO 优化在构建抗攻击联邦学习框架中的关键作用。

为了进一步验证 PSO 进行动态角色分配的必要性，我们在不使用的条件下，之前的工作节点、验证节点、矿工的节点的数量分别为 12, 5, 3, 接下来参考了文献 [55] 中关于角色分配的建议，我们将 3 种角色的数量重新调整为 8 个工作节点、9 个验证节点和 3 个矿工节点。

如图 11 所示，通过调整角色数量，我们在ρ=15% 时比较了 MNIST 数据集和 Fashion-MNIST 数据集的测试准确率，重新调整的角色分配方案的性能明显低于之前的角色分配方案，准确率存在显著的差距，重新调整的角色分配方案的准确率如表 6 所示。

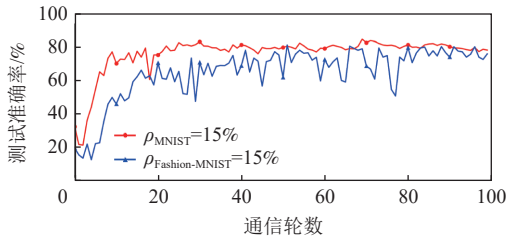


Fig. 11 Comparison of test accuracy on different datasets under different numbers of nodes

图 11 不同节点数量下不同数据集的测试准确率对比

2 种不同静态角色分配方案的性能差异表明，角色分配比例对模型性能具有显著影响，固定的角色分配策略无法适应训练过程中的动态变化需求，PSO 动态角色分配机制能够根据训练状态自适应调整角色比例，避免了静态分配中因比例设置不当而导致的性能损失。这种自适应能力使得模型能够在不同训练阶段采用最优的角色配置。

Table 6 Comparative Analysis of Different Role Allocation Schemes on MNIST and Fashion-MNIST Datasets

表 6 MNIST 和 Fashion-MNIST 数据集下不同角色分配方案的对比分析

节点数量	平均测试准确率/%	
工作节点、验证节点、矿工节点	MNIST	Fashion-MNIST
12，5，3	92.60	81.10
8，9，3	81.25	79.77

为了进一步验证 PSO 和 Paillier 的高效性，我们移除了 Paillier 的模型(仅使用 PSO 优化算法，用 w/o Paillier 表示)，并且移除了 PSO 优化算法和 Paillier 加密算法的模型(用 w/o Paillier+PSO 表示)。并对其进行了消融研究，结果如表 7 所示，完整模块 FedPPB 在所有测试场景下均取得了最高的平均测试准确率，表明 PSO 和 Paillier 的协同作用对模型性能至关重要。在 MNIST 数据集上，FedPPB 在ρ=0% 下达到 98.72% 的准确率，相较于 w/o Paillier+PSO, w/o Paillier, w/o PSO 分别提升了 9.00 个百分点、6.57 个百分点、6.94 个百分点；在ρ=15% 和ρ=25% 下，准确率分别为 96.81% 和 87.59%，同样显著优于其他配置，分别提升约 6.41 个百分点和 76.99 个百分点(相较于 w/o Paillier+PSO)。在 Fashion-MNIST 数据集上，FedPPB 在ρ=0% 下达到 96.51% 的准确率，相较于其他配置提升约 34.26 个百分点(w/o Paillier+PSO)、20.71 个百

Table 7 Effect of Key Modules of FedPPB on the Accuracy of MNIST, Fashion-MNIST and CIFAR-10 Datasets

表 7 FedPPB 关键模块对 MNIST，Fashion-MNIST，CIFAR-10 数据集的准确率影响 %

配置	数据集	ρ=0%	ρ=15%	ρ=25%
w/o Paillier+PSO	MNIST	89.72±0.11	90.40±0.10	10.60±0.03
	Fashion-MNIST	62.25±0.13	12.88±0.05	12.82±0.05
	CIFAR-10	71.35±0.16	9.67±0.08	6.15±0.06
w/o Paillier	MNIST	92.15±0.09	85.20±0.12	45.30±0.18
	Fashion-MNIST	75.80±0.13	35.50±0.14	28.90±0.16
	CIFAR-10	76.82±0.14	28.45±0.18	22.15±0.19
w/o PSO	MNIST	91.78±0.10	92.60±0.11	86.00±0.15
	Fashion-MNIST	88.10±0.09	81.10±0.09	72.63±0.17
	CIFAR-10	78.24±0.15	74.67±0.16	67.85±0.18
FedPPB	MNIST	**98.72±0.07**	**96.81±0.09**	**87.59±0.09**
	Fashion-MNIST	**96.51±0.10**	**86.23±0.15**	**76.65±0.16**
	CIFAR-10	**82.65±0.13**	**78.92±0.15**	**69.28±0.17**

注：黑体数值表示最佳结果。

分点(w/o Paillier)、8.41 个百分点(w/o PSO)；在 ρ=15% 和 ρ=25% 比率下，准确率分别为 86.23% 和 76.65%，分别提高约 73.35 个百分点和 63.83 个百分点(相较于 w/o Paillier+PSO)。在 CIFAR-10 数据集上，FedPPB 在 ρ=0% 下达到 82.65% 的准确率，相较于其他配置提升约 11.3 个百分点(w/o Paillier+PSO)、5.83 个百分点(w/o Paillier)和 4.41 个百分点(w/o PSO)；在 ρ=15% 和 ρ=25% 比率下，准确率分别为 78.92% 和 69.28%，分别高出约 69.25 个百分点和 63.13 个百分点(相较于 w/o Paillier+PSO)。

w/o Paillier+PSO 导致性能显著下降，尤其在高恶意客户端比率下，MNIST，Fashion-MNIST，CIFAR-10 的准确率分别降至 10.60%，12.82%，6.15%(ρ=25%)，表明两者缺失对模型鲁棒性的破坏性影响。w/o Paillier 在 ρ=0% 比率下仍保持较高准确率，MNIST 为 92.15%，Fashion-MNIST 为 75.80%，CIFAR-10 为 76.82%，但随着 ρ增加，准确率急剧下降至 45.30%，28.90%，22.15%，反映了 Paillier 加密在对抗恶意干扰中的重要性。w/o PSO 在低恶意设备比率场景下表现较高准确率，MNIST 为 92.60%，Fashion-MNIST 为 81.10%，CIFAR-10 为 74.67%，但在 ρ=25% 下准确率降至 86.00%，72.63%，67.85%，表明 PSO 在高对抗性环境下的优化能力至关重要。

经过上述的实验验证，PSO 和 Paillier 是 FedPPB 方法的核心组件，其协同作用显著提升在 MNIST，Fashion-MNIST，CIFAR-10 数据集上的性能和鲁棒性。特别是在高恶意客户端比率下，完整模块的优越性验证了方法设计的合理性，为其在分布式学习中的实际应用提供了强有力的支持。

2)PSO 优化算法对区块生成时间的影响

为验证我们提出的 PSO 动态分配机制能够有效解决引言中所揭示的资源利用效率瓶颈，并量化其在降低系统开销方面的效果，我们进行了本项实验。将区块生成时间作为衡量系统开销的核心指标，对比了完整 FedPPB 模型与 w/o PSO 在不同 ρ 下的性能。实验结果如图 12 所示。

图 12 清晰地展示了 2 个核心结论：

1)在平均性能上，FedPPB 在所有测试场景下的平均区块生成时间均显著低于 w/o PSO。特别是在面临 ρ=25% 的严苛环境下，FedPPB 能够在 Fashion-MNIST 中将作为系统开销核心指标的区块生成时间从 18.64 s 缩短至 13.25 s，性能提升 28.9 个百分点 。

2)在系统稳定性上，FedPPB 的误差线(error bar)也一致地比 w/o PSO 更短。这强有力地证明，我们的

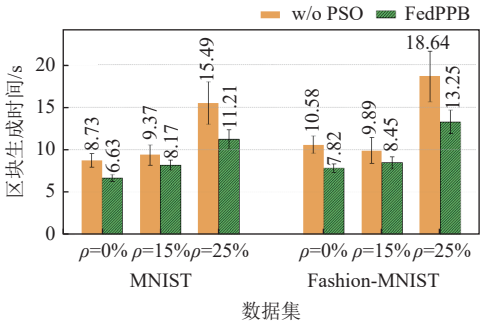


Fig. 12 Comparison of block generation time between FedPPB and w/o PSO schemes

图 12 FedPPB 与 w/o PSO 方案的区块生成时间对比

动态优化机制不仅提升了平均效率，更关键的是增强了系统的稳定性，有效抑制了因随机分配不当而导致的极端延迟事件。

本实验的统计数据证明，FedPPB 方法通过 PSO 的自适应宏观调控系统性地降低了这类瓶颈事件发生的频率和影响，使得系统的平均性能(均值更低)和稳定性(方差更小)都得到了显著提升。因此，系统开销的量化降低，是我们成功缓解了资源利用率瓶颈的直接证明。

如表 8 所示，在处理更为复杂的 CIFAR-10 模型参数时，PSO 的优化效果依然显著。在 ρ=25% 的高负载场景下，FedPPB 的区块生成时间为 15.83 s，相比 w/o PSO 所需的 21.74 s，效率提升了 27.2%。这直接证明了我们将区块生成时间纳入 PSO 适应度函数的设计是有效的，能够跨越不同复杂度的任务，切实降低系统开销，提升整体运行效率。

Table 8 Comparison of Block Generation Time for Key Modules of FedPPB on MNIST, Fashion-MNIST and CIFAR-10 Datasets

表 8 FedPPB 关键模块在 MNIST，Fashion-MNIST，CIFAR-10 数据集的区块生成时间对比 s

配置	数据集	ρ=0%	ρ=15%	ρ=25%
w/o PSO	MNIST	8.73	9.37	15.49
	Fashion-MNIST	10.58	9.89	18.64
	CIFAR-10	12.45	13.82	21.74
FedPPB	MNIST	**6.63**	**8.17**	**11.21**
	Fashion-MNIST	**7.82**	**8.45**	**13.25**
	CIFAR-10	**9.18**	**11.29**	**15.83**

注：黑体数值表示最佳结果。

本节没有对 w/o Paillier 的区块生成时间进行分析，因为 Paillier 算法是对模型参数进行加密，与没有加入 Paillier 算法的区块生成时间差距很小。

3)PSO 优化算法的开销分析

为系统性地评估本文所提出的 PSO 动态角色分配机制的计算开销,我们设计了实验以量化其在系统单轮通信总耗时中的占比。实验在ρ=15% 的对抗环境下进行,分别测试了网络规模为N=20 和N=40 的场景,并记录了连续 100 个通信轮次中各关键阶段的耗时。

实验结果表明,PSO 算法的搜索开销在计算上是可忽略的。如图 13 所示,在N=20 和N=40 的配置下,PSO 搜索耗时在单轮通信总耗时中的平均占比均低于 1.5%。这一结果符合预期,其根本原因在于计算复杂度的量级差异:PSO 的优化过程主要涉及在低维整数空间(W,V,M)中的轻量级向量运算,其计算量远低于模型训练中的梯度反向传播。

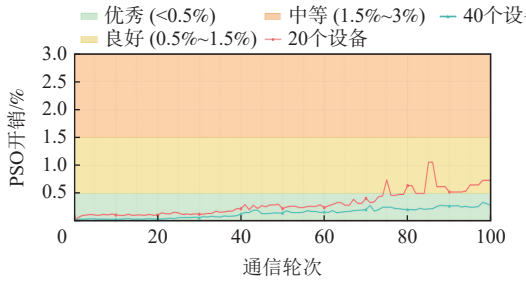


Fig. 13 Analysis of computational overhead of PSO algorithm under different network scales

图 13 PSO 算法在不同网络规模下的计算开销分析

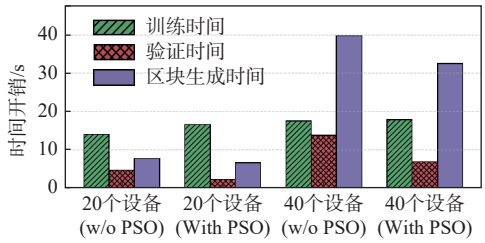
更重要的是,这一微小的计算成本为系统性能带来了显著的增益。如图 14 所示,在N=20 和N=40 的配置下,FedPPB 的验证时间和区块生成时间相较于 w/o PSO 显著降低。因此,PSO 算法能够智能地寻找到满足系统安全和共识所需的最小验证者与矿工数量,从而系统性地压缩了验证时间与区块生成时间,将更多资源高效地分配给模型训练任务。

综上分析,经过系统的实验验证,本研究提出的 FedPPB 方法在 Paillier 加密算法和 PSO 优化算法展现出显著优势。

5.5 可扩展性分析

为了评估 FedPPB 框架在更大规模网络环境下的性能与开销,我们进行了可扩展性分析,将网络规模从 20 个节点扩展至 100 和 200 个节点。实验结果清晰地表明,FedPPB 具备良好的可扩展性与鲁棒性。

在模型性能方面,随着网络规模的扩大,FedPPB 的准确率表现出平滑、可控的下降趋势。如表 9 所示,在 200 个节点和ρ=25% 的严苛环境下,模型在



(a) 各阶段时间开销对比

场景	训练时间/s	验证时间/s	区块生成时间/s
20个设备 (w/o PSO)	14.01	4.69	7.68
20个设备 (With PSO)	16.61	2.22	6.63
40个设备 (w/o PSO)	17.51	13.76	39.72
40个设备 (With PSO)	17.82	6.85	32.47

(b) 各阶段时间开销统计表

Fig. 14 Comparative analysis of time overhead of each stage under different network scales

图 14 不同网络规模下各阶段的时间开销对比分析

Table 9 Performance and Overhead Comparison of Different Methods at Various Network Scales

表 9 不同网络规模下各方法的性能与开销对比

网络规模	评估指标	方法	ρ=0%	ρ=15%	ρ=25%
节点总数 N=100	模型准确率/%	vanilla FL	87.32±0.22	12.87±0.12	-
		VBFL	86.18±0.19	85.64±0.17	9.15±0.08
		FedPPB (本文)	**96.45±0.15**	**94.23±0.18**	**84.72±0.21**
	区块生成时间/s	w/o PSO	21.45	23.78	38.92
		FedPPB	**16.32**	**20.15**	**28.67**
	PSO 搜索时间/s	FedPPB	0.150 6	0.154 7	0.158 5
节点总数 N=200	模型准确率/%	vanilla FL	83.25±0.45	9.87±0.18	-
		VBFL	82.14±0.42	83.76±0.35	8.42±0.12
		FedPPB (本文)	**94.18±0.28**	**90.45±0.32**	**81.23±0.38**
	区块生成时间/s	w/o PSO	45.82	52.14	86.73
		FedPPB	**34.26**	**42.89**	**63.35**
	PSO 搜索时间/s	FedPPB	0.260 4	0.267 8	0.274 4

注:黑体数值表示最佳结果;"-"表示模型训练崩溃。

MNIST 数据集上的准确率仍能维持在 81.23%,与 20 个节点时的 87.59% 相比性能稳定,且始终显著优于 VBFL 等基线方法,后者在同等攻击下准确率已降至 8.42%。这证明了 FedPPB 的防御机制在大规模网络中依然有效。

在系统开销方面,我们重点分析了区块生成时间和 PSO 搜索时间。如表 9 所示,当节点数从 20 增加到 200 时,FedPPB 的区块生成时间(在ρ=0% 下)从 6.63 s 增加到 34.26 s(仅增长约 4.2 倍),呈现出显著的亚线性(sub-linear)增长[①]特征。同样,PSO 算法

① 一个量的增长速度慢于另一个相关量的增长速度。

自身的搜索开销也表现出亚线性增长，从 100 节点的约 0.15 s 增加到 200 节点的约 0.26 s。这一关键结果有力地证明，我们基于 PSO 的动态角色分配机制能够有效抑制因网络规模扩大带来的通信和计算瓶颈，避免了系统开销的失控增长。

综上所述，本节可扩展性分析验证了 FedPPB 不仅在小规模网络中表现优越，更重要的是，其核心机制能够在扩展至数百节点的网络中，依旧保持高水平的模型鲁棒性和可控的系统开销，展现了其在真实、大规模分布式环境中的应用潜力。

6 结　论

本文提出的 FedPPB 机制通过将粒子群优化(PSO)算法与 Paillier 同态加密算法结合，在区块链赋能的联邦学习系统中实现了模型训练参数的加密保护与客户端角色分配的最优化。在性能提升方面，PSO 算法针对 3 类角色任务设置多维评估指标并构建适应度函数，显著提升了全局模型的准确性与系统运行的稳定性。而在安全性方面，Paillier 加密算法保障了模型参数在传输过程中的隐私性，有效防御了模型推断与参数投毒等攻击风险；实验结果表明，FedPPB 不仅在恶意设备环境下保持高鲁棒性，且在模型精度和区块生成时间上均优于现有方案，充分验证了 FedPPB 在安全性与效率之间实现了良好平衡。

尽管本研究在提高联邦学习系统的安全性与性能方面取得了积极进展，但仍存在若干值得深入探讨的问题。未来的研究可考虑以下方向：其一，在保证同态加密安全性的前提下，进一步优化加密与解密的计算效率，以提升系统在大规模分布式环境中的可扩展性；其二，PSO 算法尽管具备良好的全局搜索能力，但在动态网络环境中的收敛速度和稳定性仍有改进空间，后续可探索引入混合优化策略以增强算法的适应性；其三，可进一步扩展模型至异构数据分布、多任务学习和跨域联邦学习等复杂场景，增强方案的通用性与实用性；上述方向将为构建更加安全、高效、可信的联邦学习系统奠定坚实基础。

作者贡献声明：沈凡凡提出研究框架与整体架构，并指导研究方向；刘梓昂负责 PSO 动态角色分配框架的设计、建模与实现，主导实验设计与论文撰写；梁琦玮负责 Paillier 同态加密算法的集成与安全性分析；徐超搭建实验环境，执行多数据集对比实验并完成结果可视化；陈勇参与区块链机制与 PoS 共识流程建模；何炎祥统筹项目进度，并对论文理论深度与最终稿进行修改。

参　考　文　献

[1] Yang Qiang, Liu Yang, Chen Tianjian, et al. Federated machine learning: Concept and applications[J]. ACM Transactions on Intelligent Systems and Technology, 2019, 10(2): 1−19

[2] Kairouz P, McMahan H B, Avent B, et al. Advances and open problems in federated learning[J]. Foundations and Trends in Machine Learning, 2021, 14(1/2): 1−210

[3] Zhang Weishan, Lu Qianghua, Yu Qiuyu, et al. Blockchain-based federated learning for device failure detection in industrial IoT[J]. IEEE Internet of Things Journal, 2020, 8(7): 5926−5937

[4] Brisimi T S, Chen Ruidi, Mela T, et al. Federated learning of predictive models from federated electronic health records[J]. International Journal of Medical Informatics, 2018, 112: 59−67

[5] Ma Chuan, Li Jun, Shi Long, et al. When federated learning meets blockchain: A new distributed learning paradigm[J]. IEEE Computational Intelligence Magazine, 2022, 17(3): 26−33

[6] Kim H, Park J, Bennis M, et al. Blockchained on-device federated learning[J]. IEEE Communications Letters, 2019, 24(6): 1279−1283

[7] Chen Hang, Asif S A, Park J, et al. Robust blockchained federated learning with model validation and proof-of-stake inspired consensus[J]. arXiv preprint, arXiv: 2101.03300, 2021

[8] Li Xiaoqi, Jiang Peng, Chen Ting, et al. A survey on the security of blockchain systems[J]. Future Generation Computer Systems, 2020, 107: 841−853

[9] Peng Changgen, Long Yangyang, Chen Yuling. Verifiable distributed key generation protocol based on blockchain and CP-ABE[J]. Chinese Journal of Computers, 2025, 48(6): 1342−1355 (in Chinese)
(彭长根, 龙洋洋, 陈玉玲. 基于区块链与 CP-ABE 的可验证分布式密钥生成协议[J]. 计算机学报, 2025, 48(6): 1342−1355)

[10] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Proc of the 20th Int Conf on Artificial Intelligence and Statistics. New York: PMLR, 2017: 1273−1282

[11] Konečný J, McMahan H B, Yu F X, et al. Federated learning: Strategies for improving communication efficiency[J]. arXiv preprint, arXiv: 1610.05492, 2016

[12] Qi Yuanhang, Hossain M S, Nie Jiangtian, et al. Privacy-preserving blockchain-based federated learning for traffic flow prediction[J]. Future Generation Computer Systems, 2021, 117: 328−337

[13] Korkmaz C, Kocas H E, Uysal A, et al. Chain FL: Decentralized federated machine learning via blockchain[C]//Proc of the 2nd Int Conf on Blockchain Computing and Applications (BCCA). Piscataway, NJ: IEEE, 2020: 140−146

[14] Gao Sheng, Yuan Liping, Zhu Jianming, et al. A blockchain-based privacy-preserving asynchronous federated learning[J]. SCIENTIA SINICA Informationis, 2021, 51(10): 1755−1774 (in Chinese)
(高胜, 袁丽萍, 朱建明, 等. 一种基于区块链的隐私保护异步联邦学习[J]. 中国科学: 信息科学, 2021, 51(10): 1755−1774)

[15] Liu Wei, Tang Congke, Ma Jie, et al. Privacy-preserving federated learning model based on blockchain and dynamic evaluation[J]. Journal of Computer Research and Development, 2023, 60(11): 2583−2593 (in Chinese)
(刘炜, 唐琮轲, 马杰, 等. 基于区块链和动态评估的隐私保护联邦学习模型[J]. 计算机研究与发展, 2023, 60(11): 2583−2593)

[16] Miao Yinbin, Liu Ziteng, Li Hongwei, et al. Privacy-preserving Byzantine-robust federated learning via blockchain systems[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 2848−2861

[17] Fan Mochan, Ji Kailai, Zhang Zhaofeng, et al. Lightweight privacy and security computing for blockchained federated learning in IoT[J]. IEEE Internet of Things Journal, 2023, 10(18): 16048−16060

[18] Yu Boan, Zhao Jun, Zhang Kai, et al. Lightweight and dynamic privacy-preserving federated learning via functional encryption[J]. IEEE Transactions on Information Forensics and Security, 2025, 20: 2496−2508

[19] Li Bing, Lu Jianfeng, Cao Shuqin, et al. RATE: Game-theoretic design of sustainable incentive mechanism for federated learning[J]. IEEE Internet of Things Journal, 2024, 12(1): 81−96

[20] Wang Kaiqi, Hong Ruiqi, Mao Yunlong, et al. Building a secure and decentralized federated learning scheme based on blockchain[J]. SCIENTIA SINICA Informationis, 2024, 54(2): 316−334 (in Chinese)
(王恺祺, 洪睿琦, 毛云龙, 等. 基于区块链构建安全去中心化的联邦学习方案[J]. 中国科学: 信息科学, 2024, 54(2): 316−334)

[21] Kennedy J, Eberhart R. Particle swarm optimization[C]//Proc of Int Conf on Neural Networks(ICNN'95). Piscataway, NJ: IEEE, 1995: 1942−1948

[22] Yao Jicheng, Luo Xiaonan, Li Fang, et al. Research on hybrid strategy particle swarm optimization algorithm and its applications[J]. Scientific Reports, 2024, 14(1): 24928

[23] Shi Y, Eberhart R. A modified particle swarm optimizer[C]//Proc of 1998 IEEE Int Conf on Evolutionary Computation. Piscataway, NJ: IEEE, 1998: 69−73

[24] Demir S, Sahin E K. Predicting occurrence of liquefaction-induced lateral spreading using gradient boosting algorithms integrated with particle swarm optimization: PSO-XGBoost, PSO-LightGBM, and PSO-CatBoost[J]. Acta Geotechnica, 2023, 18(6): 3403−3419

[25] Zhang Baowei, Song Jiuxiang, Wang Yonghua. PSO-DE-based regional scheduling method for shared vehicles[J]. Automatic Control and Computer Sciences, 2023, 57(2): 167−176

[26] Lin Shiwei, Wang Jianguo, Huang Bomin, et al. Bio particle swarm optimization and reinforcement learning algorithm for path planning of automated guided vehicles in dynamic industrial environments[J]. Scientific Reports, 2025, 15(1): 463

[27] Supriya Y, Gadekallu T R. Particle swarm-based federated learning approach for early detection of forest fires[J]. Sustainability, 2023, 15(2): 964

[28] Park S, Suh Y, Lee J. FedPSO: Federated learning using particle swarm optimization to reduce communication costs[J]. Sensors, 2021, 21(2): 600

[29] Zhao Zichao, Xia Junjuan, Fan Lisheng, et al. System optimization of federated learning networks with a constrained latency[J]. IEEE Transactions on Vehicular Technology, 2021, 71(1): 1095−1100

[30] Houssein E H, Sayed A. Boosted federated learning based on improved particle swarm optimization for healthcare IoT devices[J]. Computers in Biology and Medicine, 2023, 163: 107195

[31] Mohammed S J, Taha D B. Paillier cryptosystem enhancement for homomorphic encryption technique[J]. Multimedia Tools and Applications, 2024, 83(8): 22567−22579

[32] Mohammed S J, Taha D B. Performance evaluation of RSA, ElGamal, and paillier partial homomorphic encryption algorithms[C]//Proc of 2022 Int Conf on Computer Science and Software Engineering (CSASE). Piscataway, NJ: IEEE, 2022: 89–94

[33] Marcolla C, Sucasas V, Manzano M, et al. Survey on fully homomorphic encryption, theory, and applications[J]. Proceedings of the IEEE, 2022, 110(10): 1572−1609

[34] Paillier P. Public-key cryptosystems based on composite degree residuosity classes[C]//Proc of the 9th Int Conf on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 1999: 223–238

[35] Fang Haokun, Qian Quan. Privacy preserving machine learning with homomorphic encryption and federated learning[J]. Future Internet, 2021, 13(4): 94

[36] Zhang Xianglong, Fu Anmin, Wang Huaqun, et al. A privacy-preserving and verifiable federated learning scheme[C]//Proc of 2020 IEEE Int Conf on Communications (ICC). Piscataway, NJ: IEEE, 2020: 1−6

[37] Ku Hanchao, Willy S, Zhang Yudi, et al. Privacy-preserving federated learning in medical diagnosis with homomorphic re-encryption[J]. Computer Standards & Interfaces, 2022, 80: 103583

[38] Zhang Chengliang, Li Suyi, Xia Junzhe, et al. BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning[C]// Proc of 2020 USENIX Annual Technical Conf (USENIX ATC'20). Berkeley, CA: USENIX Association, 2020: 493−506

[39] Tao Yunting. Research on Internet of things privacy-protecting data aggregation protocol based on Paillier homomorphic encryption [D]. Jinan: Shandong University, 2022 (in Chinese)
(陶云亭. 基于 Paillier 同态加密的物联网隐私保护数据聚合协议研究[D]. 济南: 山东大学, 2022)

[40] Han Junhao, Yan Li. Adaptive batch homomorphic encryption for joint federated learning in cross-device scenarios[J]. IEEE Internet of Things Journal, 2023, 11(6): 9338−9354

[41] Ma Xu, Zhou Yuqing, Wang Laihua, et al. Privacy-preserving Byzantine-robust federated learning[J]. Computer Standards & Interfaces, 2022, 80: 103561

[42] Yang Wenqiang, Liu Bin, Lu Changlei, el al. Privacy preserving on updated parameters in federated learning[C]// Proc of ACM Turing Celebration Conf–China (ACM TURC'2020). New York: ACM, 2020: 27–31

[43] Kirkpatrick S, Gelatt C D, Vecchi M P. Optimization by simulated annealing[J]. Science, 1983, 220(4598): 671−680

[44] Jones D R, Schonlau M, Welch W J. Efficient global optimization of expensive black-box functions[J]. Journal of Global Optimization, 1998, 13(4): 455−492

[45] Holland J H. Adaptation in Natural and Artificial Systems[M]. Cambridge, MA: MIT, 1992

[46] Dorigo M, Maniezzo V, Colorni A. Ant system: Optimization by a

colony of cooperating agents[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 1996, 26(1): 29−41

[47] Yang X S. Firefly algorithm, stochastic test functions and design optimisation[J]. International Journal of Bio-Inspired Computation, 2010, 2(2): 78−84

[48] Kaelbling L P, Littman M L, Moore A W. Reinforcement learning: A survey[J]. Journal of Artificial Intelligence Research, 1996, 4: 237−285

[49] Fotohi R, Aliee F S, Farahani B. A lightweight and secure deep learning model for privacy-preserving federated learning in intelligent enterprises[J]. IEEE Internet of Things Journal, 2024, 11(19): 31988−31998

[50] Aziz G, Mistry S, Krishna A. VBSFL: A robust blockchained split-fed learning model for secured distributed learning[J]. IEEE Transactions on Industrial Informatics, 2025, 21(7): 5431−5439

[51] Castro M, Liskov B. Practical Byzantine fault tolerance[C] //Proc of the 3rd USENIX Symp on Operating Systems Design and Implementation (OSDI'99). Berkeley, CA: USENIX Association, 1999: 173–186

[52] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J] Proceedings of the IEEE, 1998, 86(11): 2278−2324

[53] Xiao Han, Rasul K, Vollgraf R. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms[J]. arXiv preprint, arXiv: 1708.07747, 2017

[54] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[J]. Handbook of Systemic Autoimmune Diseases, 2009, 1(4): 7

[55] Yu Feng, Lin Hui, Wang Xiaoding, et al. Blockchain-empowered secure federated learning system: Architecture and applications[J]. Computer Communications, 2022, 196: 55−65



Shen Fanfan, born in 1987. PhD, associate professor, master supervisor. Senior member of CCF. His main research interests include edge intelligence and storage systems.

沈凡凡, 1987 年生。博士、副教授、硕士生导师。CCF 高级会员。主要研究方向为边缘智能、存储系统。



Liu Zi'ang, born in 2001. Master candidate. His main research interests include federated learning and privacy protection.

刘梓昂, 2001 年生。硕士研究生。主要研究方向为联邦学习、隐私保护。



Liang Qiwei, born in 1998. Master. Her main research interests include federated learning and privacy protection.

梁琦玮, 1998 年生。硕士。主要研究方向为联邦学习、隐私保护。



Xu Chao, born in 1980. PhD, professor, PhD supervisor. Distinguished member of CCF. His main research interests include trusted software and blockchain technology.

徐　超, 1980 年生。博士、教授、博士生导师。CCF 杰出会员。主要研究方向为可信软件、区块链技术。



Chen Yong, born in 1986. PhD, associate professor, master supervisor. Member of CCF. His main research interests include compilation technology and smart contract.

陈　勇, 1986 年生。博士、副教授、硕士生导师。CCF 会员。主要研究方向为编译技术、智能合约。



He Yanxiang, born in 1952. PhD, professor, PhD supervisor. Fellow of CCF. His main research interests include trusted software and compiler technology.

何炎祥, 1952 年生。博士、教授、博士生导师。CCF 会士。主要研究方向为可信软件、编译技术。