计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202330274
60(11): 2504−2519, 2023

一种支持自适应联邦学习任务的可信公平区块链框架

张宝晨[1] 黄 月[1] 孔兰菊[1,2] 李庆忠[1,2] 李文全[1] 郭秋曼[1]
[1](山东大学软件学院 济南 250101)
[2](山大地纬软件股份有限公司 济南 250101)
(baochenzhang@mail.sdu.edu.cn)

A Trustworthy and Fair Blockchain Framework Supporting Adaptive Federated Learning Task

Zhang Baochen[1], Huang Yue[1], Kong Lanju[1,2], Li Qingzhong[1,2], Li Wenquan[1], and Guo Qiuman[1]
[1](*School of Software*, *Shandong University*, *Jinan* 250101)
[2](*Dareway Software Co.*, *Ltd.*, *Jinan* 250101)

Abstract Consensus mechanism is an important part of blockchain technology, but the mainstream consensus mechanisms, especially proof-of-work consensus mechanisms, suffer from problems such as wasted computing power and low throughput. Federated learning as a distributed machine learning method, the local training of learning models and the final calculation of participant contributions require a large amount of computing power. Therefore, we propose a trusted and fair blockchain framework, called TFchain, supporting adaptive federated learning tasks to explore how to utilize the wasted arithmetic power in the original consensus mechanism to improve the efficiency of federated learning. First, we design a new consensus mechanism PoTF (proof of trust and fair) based on blockchain and federated learning, which sets the nodes of the blockchain as the participants of federated learning and transfers a large amount of ineffective arithmetic power used in the original consensus mechanism for Hash computation to federated learning for training of local models and evaluation of participants' contributions. Second, while improving the throughput of blockchain transactions, the participants of federated learning are evaluated and incentivized with reasonable contributions. Finally, an algorithm is designed to prevent nodes from being evil. The experimental results show that the TFchain proposed in this paper can effectively improve the transaction processing performance of the blockchain while recycling the arithmetic power, and provide effective positive incentives to the participants who actively participate in federated learning.

Key words blockchain; federated learning; consensus algorithm; rewards distribution; contribution calculation; incentive mechanism; bad-behavior detection; computing power recycling

摘 要 共识机制是区块链技术的重要组成部分,但是主流的共识机制尤其是工作量证明共识机制都存在算力过度耗费和吞吐量低等问题.而联邦学习作为一种分布式机器学习方法,学习模型的本地训练和最终的参与方贡献度计算都需要消耗大量算力资源.因此,提出了一种支持自适应联邦学习任务的可信

收稿日期: 2023-04-03; **修回日期**: 2023-06-12
基金项目: 国家重点研发计划项目(2021YFF0704102); 国家社会科学基金项目 (20BJY131); 山东省重大科技创新项目 (2020CXGC010106, 2021CXGC010108); 泉城产业领军人才项目; CCF-华为胡杨林基金项目
This work was supported by the National Key Research and Development Program(2021YFF0704102), the National Social Science Fund of China (20BJY131), the Major Science and Technology Innovation of Shandong Province (2020CXGC010106, 2021CXGC010108), the Industrial Experts Program of Spring City and CCF-Huawei Populus Grove Fund.
通信作者: 孔兰菊(klj@sdu.edu.cn)

公平区块链框架 TFchain，探索如何利用原本共识机制中耗费的大量算力来提高联邦学习的效率. 首先，设计了基于区块链和联邦学习的全新共识机制 PoTF（proof of trust and fair），该共识机制将区块链的节点设置为联邦学习的参与方，将原本共识机制中用于哈希计算的大量无效算力转移到联邦学习中，进行本地模型的训练和参与方贡献度的评估；其次，在提高区块链交易吞吐量的同时，对联邦学习的参与方进行了合理的贡献度评估和激励；最后，设计了防止节点作恶的算法. 实验结果表明，提出的 TFchain 能够在回收算力的同时有效提升区块链的交易处理性能，对积极参与联邦学习的参与方进行有效正向的激励.

关键词 区块链；联邦学习；共识算法；奖励分配；贡献计算；激励机制；作恶检测；算力资源回收

中图法分类号 TP181；TP311

随着数字时代的发展和人们对数据的日益重视，隐私保护、信息安全和数据所有权已成为人们关注的重要问题. 区块链[1]作为一种去中心化的分布式账本技术，其核心特性是去中心化和不可篡改性，这意味着没有单个实体可以掌控整个系统，同时也意味着区块链上的数据是无法被篡改的，这些特点使得区块链能够在众多领域都具有广泛的应用前景.

在分布式系统中，一致性是非常重要的一个问题，因为不同节点之间可能存在数据不一致的情况，导致系统出现错误. 区块链通过使用共识算法，保证了不同节点之间的数据一致性. 例如，在比特币区块链中，采用了工作量证明（proof of work, PoW）[2]算法来实现共识，这样所有的节点可以达成一致的结果. 因此，区块链能够有效地解决分布式系统中的一致性问题.

在 PoW 中，挖掘区块的矿工节点需要解决一道难题，获得正确答案后才能提交区块. 其他节点需要验证这个难题是否解决来决定是否接受区块，从而达成所有节点之间的数据一致性. 虽然 PoW 算法可以实现一致性，但是它也存在着一些问题，其中最显著的问题就是算力的过度耗费. PoW 算法中矿工需要通过不断进行无意义的哈希计算来寻找正确的答案，这导致 PoW 算法会消耗大量的能源，目前以 PoW 为共识算法的比特币挖矿消耗的总能源已经超过了奥地利每年的总用电量，这造成了极大的能源浪费[3].

另外，比特币等基于 PoW 的公有区块链平台，为了控制区块的出块速度，会设置目标难度值. 这是一个衡量挖矿难度的参数，该参数根据比特币网络的总算力（即全网的矿机算力）来动态调整的. 目标难度值的变化对应着全网矿工的算力变化. 如果全网算力增加，目标难度值也会增加，反之亦然. 定期调整目标难度值，能够保证每个区块的挖矿难度始终保持在一个稳定的水平，但是这也导致了 PoW 挖矿越来越难的现象. 截止 2023 年 3 月想要得到一个比特币需要 46.84 万亿次哈希计算，如图 1 所示.

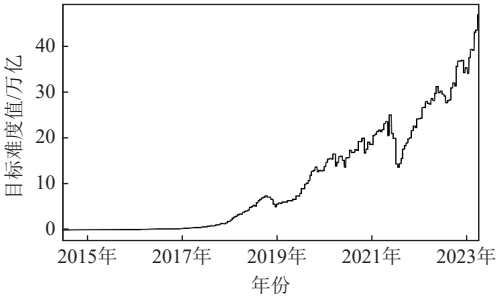


Fig. 1 Changes in bitcoin's difficulty values

图 1 比特币的目标难度值变化

目前，大多数公有链系统采用的共识机制仍然是 PoW，但是其高能耗和低吞吐量的现状已经成为了区块链发展的瓶颈.

与此同时联邦学习（federated learning, FL）[4]作为一种新兴的机器学习技术，可以在不暴露原始数据的情况下，让多个参与方通过模型的联合训练，实现模型的共同提升. 与传统的集中式机器学习相比，联邦学习具有更高的安全性和隐私保护性[5]，但是联邦学习训练过程中需要大量计算资源，同时也缺乏一种公平有效的参与方激励机制，而区块链的公开透明、不可篡改等特性保证了区块链能够提供一种公平的节点激励机制. 因此，很多研究考虑将联邦学习引入区块链领域，探索一种新的共识机制和联邦学习激励机制[6].

因此，针对上文提出区块链共识算法中的算力过度耗费问题和联邦学习缺乏有效公平激励的问题，本文引入联邦学习技术，提出了一种支持自适应联邦学习任务的可信公平区块链框架（a trustworthy and fair blockchain framework supporting adaptive federated learning task）——TFchain，并设计了一种新的共识机制——PoTF（proof of trust and fair），该机制不仅可以减少算力的耗费，而且可以通过引入联邦学习和贡献度的计算，进行 PoW 和 PoTF 的共识切换，从而提高整个区块链系统的效率和安全性. 与此同时，本文基于区块链的设计架构，对联邦学习的参与方进行

了合理的贡献度评估和激励. 最后，本文还设计了防止联邦学习参与方作恶的算法，有效地保证了区块链的安全性和联邦学习的效率.

本文的主要贡献包括 4 个方面：

1) 提出了一种支持自适应联邦学习任务的可信公平区块链框架 TFchain. TFchain 将服务节点和普通节点组成联邦系统,完成所有联邦学习任务发起者提出的训练任务. 通过将算力使用在联邦任务上，避免了传统 PoW 的算力过度耗费问题.

2) 提出新的共识机制 PoTF. 在每个联邦学习训练任务开始时，将上次任务贡献度最高的节点作为服务节点. 普通节点进行训练并且上传模型更新参数，服务节点收集参数然后进行聚合，并且在聚合的同时计算普通节点的贡献度，然后生成区块，既保障了模型参数的及时更新，又提高了区块的吞吐量.

3)提出了一种基于区块链和贡献度计算的联邦学习奖励分配机制，保证服务节点的流动性，实现奖励的公平分配，鼓励参与方积极参与到联邦任务中. 同时提出了一种节点作恶检测策略，通过投票更换在训练中排除作恶的服务节点，保证 TFchain 的安全性和可信度. 此外，在训练中还会排除贡献度异常的普通节点，进一步提升了奖励分配的公平性.

4) 通过实验证明了 TFchain 的有效性和可行性，同时证明 TFchain 是一种可插拔联邦模型的区块链组件，可以方便地加入不同的联邦学习任务. TFchain 能够在节约算力的同时，提升区块链的吞吐量，并且为联邦学习提供公平的激励机制.

1 相关工作

1.1 区块链

在区块链中，为了确保所有节点账本数据的正确性，使得所有节点达成一致并防止恶意节点提交假数据，需要使用共识机制. 目前，主要使用的共识机制有 PoW[2]、权益证明(proof of stake, PoS)[7]、代表权益证明(delegated proof of stake, DPoS)[8] 和实用拜占庭容错(practical Byzantine fault tolerance, PBFT)[9] 共识协议等.

在较大规模的区块链平台中，比如以比特币[2] 和以太坊[10] 为代表的公有链平台，使用的是 PoW 共识机制. 然而，这种机制中的工作量只是大量重复且无意义的哈希计算过程，导致系统中 99% 以上的算力被耗费，同时也浪费了大量电力[11-12]，这一缺陷一直受到诟病. 此外，为了防止恶意节点进行女巫攻击，PoW 中设定的哈希计算难度值往往很大，导致基于该共识机制设计的区块链平台的交易吞吐量通常很低[3]. 以比特币为例，其吞吐量一直维持在每秒 6～7 笔，完全无法满足需要处理大量高频交易的应用场景.

1.2 联邦学习

由于联邦学习的分散性，目前学习过程中存在客户端不积极或者作恶等问题，这些问题可能会导致模型性能下降或模型不可用等后果.

客户端不积极是指在训练过程中，有些客户端可能不愿意或者不能积极参与到模型的训练中. 联邦学习的客户端不愿意在没有回报下参与联邦学习训练，所以如何激励客户参与很重要. 客户端不积极也可能是硬件性能、网络带宽等因素造成的. 在联邦学习中，如果有太多的客户端不积极，就会导致整个模型的性能下降或者训练失败[13]. 另外一些恶意的客户端可能会恶意破坏模型更新，传输一些错误数据，导致全局模型偏离正常模型. 交换梯度对参与者来说不够安全，例如一个对抗性参与者可以推断出在其他人的训练数据中存在准确的数据点(成员推断).

为了解决客户端不积极或者作恶的问题，研究者们提出了许多方法. 常见的方法是通过设计合适的奖励机制来激励客户端积极参与训练. 文献 [14] 基于博弈论设计了一种有效的激励机制，选择最有可能提供可靠数据的用户，并补偿其隐私泄露的成本. 文献 [15] 提出了质量意识激励机制，在预算范围内，建立逆向拍卖问题的模型，以鼓励高质量学习用户的参与.

此外，还可以采用多方安全计算技术来保证模型的隐私性和安全性，从而更加公正地给参与训练的客户端提供奖励或者惩罚. 文献 [16] 设计一个基于贝叶斯博弈论的激励机制，确保参与者提供真实的有用预测. 文献 [17] 提出了一种分层的两级激励机制设计，有效地分配数据所有者和联邦学习参与者的资源，以完成编码联邦学习任务.

1.3 区块链和联邦学习结合

在联邦学习中，数据不需要集中存储在一个中心节点，而是分布在各个设备上，这与区块链的分布式存储结构相似，因此将区块链技术与联邦学习相结合是一个研究热点[18].

区块链的技术可以实现联邦学习参与者的贡献度评价和公平激励. 在区块链网络中，每个矿工的贡献度都可以通过计算能力来评价. 矿工的贡献度越高，获得奖励的概率也越高. 这样可以激励矿工提供

更多的计算资源，并保证整个网络的安全和稳定. 在联邦学习中，每个设备也可以通过贡献度评价来获得相应的奖励. 例如，在联邦学习中，每个设备可以通过提供更多的数据或更准确的模型来提高贡献度，从而获得更多的奖励. 这样可以激励设备提供更多的计算资源，并保证联邦学习的分布式和公平性. 通过区块链对联邦学习参与者的贡献度进行评价的方法主要有基于信誉度和基于博弈论 2 种.

联邦学习中的信誉度[19]是一个用于衡量参与者参与联邦学习的贡献和表现的指标. 文献 [20] 提出了一种用于联邦学习的区块链激励机制，该机制通过评估参与者的信誉和贡献指标，公平奖励高效率的节点，同时惩罚恶意节点. 文献 [21] 提出了用于去中心化联邦学习的共识机制，通过基于信誉的激励措施，激励矿工的诚实参与并提高共识效率. 文献 [22] 提出了一种用于跨设备联合学习的去中心化参数聚合链，该链利用区块链技术来保护中间参数的隐私，并使用基于信誉的激励机制来激励跨设备联合学习中的协作节点，利用智能合约来实现可靠的参与者选择和激励机制.

博弈论[23]是研究决策者之间的相互作用和影响，以及他们在这种相互作用下做出的最优决策，因此，博弈论可以用来分析和解决联邦学习中的协作和竞争问题. 文献 [24] 提出了一种基于区块链的支付系统 FedCoin，它通过沙普利值（Shapley value，*SV*）评估联邦学习参与者的贡献. 文献 [3] 引入了一种名为联邦学习证明的新型能量回收共识算法，该算法将最初耗费在 PoW 中的能量再投资于联邦学习，并且基于博弈论进行参与者的贡献度计算和激励分配. 文献 [25] 提出了一种基于贡献证明的共识机制，该机制避免了挖矿过程造成的区块生成延迟，并异步缓解了模型参数验证中的拥塞. 并且根据每个节点生成的区块数量以及其生成的区块被其他节点引用的次数来计算每个节点的贡献.

文献 [19−25] 所述的研究都是侧重于区块链对于激励的分配方面，且大部分激励是基于参与方的信誉和参与者间的博弈关系设计的，而没有将区块链的共识过程与联邦学习的训练过程进行有效地结合. 例如 FedCoin，联邦学习过程仍然是在链外进行的，只是将训练完成的结果和过程参数上链记录并且计算参与方的贡献度.

另外，区块链还可以提高联邦学习的安全性，为联邦学习进行参数的记录存储和训练过程提供隐私保护. 由于联邦学习中涉及到模型数据传输，因此存在数据隐私泄露的风险. 通过将联邦学习与区块链相结合，可以实现更高的安全性. 文献 [26] 提出了一种名为基于区块链的异步联合学习的新方法，使用区块链来确保模型数据不会被篡改. 文献 [27] 提出了一种用于安全的多方机器学习的方法，该方法使用区块链和加密技术来协调对等客户之间保护隐私的机器学习过程. 文献 [28] 提出了一种去中心化联合学习方法，该方法使用区块链技术来存储模型. 文献 [29] 提出一个双层区块链驱动的联邦学习框架，该框架由多个分片网络和基于有向无环图的主链组成. 文献 [30] 提出一项新聚合规则，该规则使用区块链促进透明的流程和法规的实施. 文献 [31] 提出了一种基于区块链的声誉感知细粒度联邦学习方法，以确保在移动边缘计算系统中进行值得信赖的协作训练.

在文献 [26−31]，区块链更多充当记录和见证的技术，而不是将区块链的节点算力用于联邦学习的训练.

2 预备知识

2.1 区块链的共识机制

共识机制是指在分布式系统中实现一致性的算法. 区块链共识机制是指在区块链中节点之间达成共识的算法，以决定哪个节点可以添加下一个区块到区块链中. 共识机制可以分为基于权益的共识机制、基于权威的共识机制和基于工作量的共识机制等不同类型.

PoW 是一种基于工作量的共识机制，最早被用于比特币中，它要求节点通过计算一个特定难度的哈希函数来寻找一个符合要求的数字，并将其添加到区块链中，其公式为：

$$hash(nonce + data) = target, \tag{1}$$

其中 *nonce* 是随机数，*data* 是待添加数据，*target* 是目标哈希值，通过不断尝试不同的 *nonce* 值，直到找到符合要求的数字，从而完成添加区块的过程. PoW 机制的主要缺点是需要大量的计算资源，因此存在一定的安全性问题.

目前，基于工作量的共识机制已经被广泛应用于各种区块链系统中，如以太坊、莱特币等. 此外，还出现了其他类型的共识机制，如基于权益证明的共识机制和拜占庭容错共识机制等，以满足不同场景下的共识需求.

2.2 联邦学习的目标函数和聚合算法

联邦学习的目标函数主要作用是衡量模型的预

测结果与真实结果之间的误差,并提供一个用于优化模型参数的方向.不同的联邦学习算法和任务可能使用不同的目标函数.

在联邦学习中,模型的目标函数通常定义为

$$\min_{w\in W} F(w)=\frac{1}{n}\sum_{i=1}^{n} f_i(w), \tag{2}$$

其中 w 表示模型参数,W 表示参数的取值范围,n 表示客户端的数量,$f_i(w)$ 表示第 i 个客户端的目标函数,其形式为

$$f_i(w)=\frac{1}{N_i}\sum_{(x,y)\in D_i} \ell(w,x,y), \tag{3}$$

其中,N_i 表示客户端 i 的本地数据集的大小,(x,y) 表示一个数据样本,$\ell(w,x,y)$ 表示损失函数.每个客户端的目标函数形式相同,但是本地的数据集是不同的.

在联邦学习中,由于每个客户端的本地数据集都不同,因此不能像传统的机器学习一样在全局数据集上进行训练,而是需要采用一些特殊的技术来解决这个问题.其中一种常用的方法是联邦平均算法 FedAvg[32], FedAvg 的本地更新方式为:

$$w_t \leftarrow w_t-\eta\nabla\ell(w_t,x,y). \tag{4}$$

FedAvg 的全局更新公式为:

$$w_{t+1}=\sum_{i=1}^{m}\frac{N_t}{N}w_{i+1}^{t}, \tag{5}$$

其中,w_i^t 表示客户端 i 在 t 轮更新后的参数,m 表示客户端的数量.在每一轮更新时,客户端通过本地训练更新参数,并将更新后的参数上传到服务节点.服务节点通过聚合算法将这些参数集成到全局模型中.

2.3 沙普利值

贡献度是指一个变量对于整个系统输出的影响程度,它在很多领域中都有着广泛的应用,例如特征选择和优化、网络分析中的节点重要性评估等.贡献度的计算方法有很多种,其中一种常用的方法是使用沙普利值[33].

沙普利值是一种用来解决合作利益分配问题的方式,它由诺贝尔奖得主劳埃德·斯托韦尔·沙普利(Lloyd Stowell Shapley)提出.其目标是构造一种综合考虑冲突各方要求的折中的效用分配方案,从而保证分配的公平性.沙普利 SV 的定义为

$$SV_i=\sum_{j=1}^{m}\frac{\left|\dfrac{\partial y_j}{\partial x_i}\right|}{\sum\limits_{k=1}^{m}\left|\dfrac{\partial y_j}{\partial x_k}\right|}, \tag{6}$$

其中,系统共有 m 个参与者,i 代表系统的第 i 个参与者,$i\in\{1,2,\ldots m\}$. x_i 和 y_i 分别是第 i 个参与者的输入变量和输出变量. SV_i 则表示第 i 个输入变量对于系统输出的影响程度,$SV_i\in[0,1]$, SV_i 越大表示该变量对于系统输出的影响越大.

3 支持自适应联邦学习任务的可信公平区块链框架 TFchain

本节分为 6 个部分:

第 1 部分介绍了支持自适应联邦学习任务的可信公平区块链框架 TFchain,该框架包含了链上链下的协作完成联邦学习任务的过程,以及 TFchain 和 PoW 的切换机制.

第 2 部分介绍了 TFchain 中的区块链的数据结构.

第 3 部分介绍了全新区块链数据结构下,一种高效且安全的共识机制 PoTF,它通过引入沙普利值来正向激励参与者.

第 4 部分具体介绍了 PoTF 中不同角色的任务和运行流程.

第 5 部分介绍了基于联邦学习贡献度的节点奖励方法.

第 6 部分介绍了一种应对服务节点作恶策略,即将节点的行为记录在区块链上,以提高节点的透明度和可信度,当发现服务节点时,在训练中排除作恶的服务节点.

3.1 TFchain 的整体框架

TFchain 的整体架构如图 2 所示,TFchain 可以整体划分为链上和链下 2 个部分,其中链上负责交易的处理、学习参数的收集和聚合、区块的打包等工作,链下负责进行联邦学习的参与方贡献度计算和本地模型训练等工作.

TFchain 中区块链的共识节点同时也是联邦学习的参与方,最终会根据联邦学习的训练过程和结果对各个参与方进行贡献度评价,最终分配区块链的记账权和学习的奖励.

在每次训练任务开始时,TFchain 会根据历史贡献度选择 PoTF 的服务节点.服务节点负责收集训练后的参数并进行聚合,然后打包,其包含每个普通节点训练参数和普通交易的区块,在完成每次的训练任务后,服务节点会在本地计算每个普通节点的贡献度,在本次训练周期 Epoch 的最后一个区块中,服务节点会上传计算的所有参与方本次学习任务的最终贡献度并且分配奖励.而在 TFchain 中,普通节点会作为联邦学习的参与方,使用私有数据集训练本

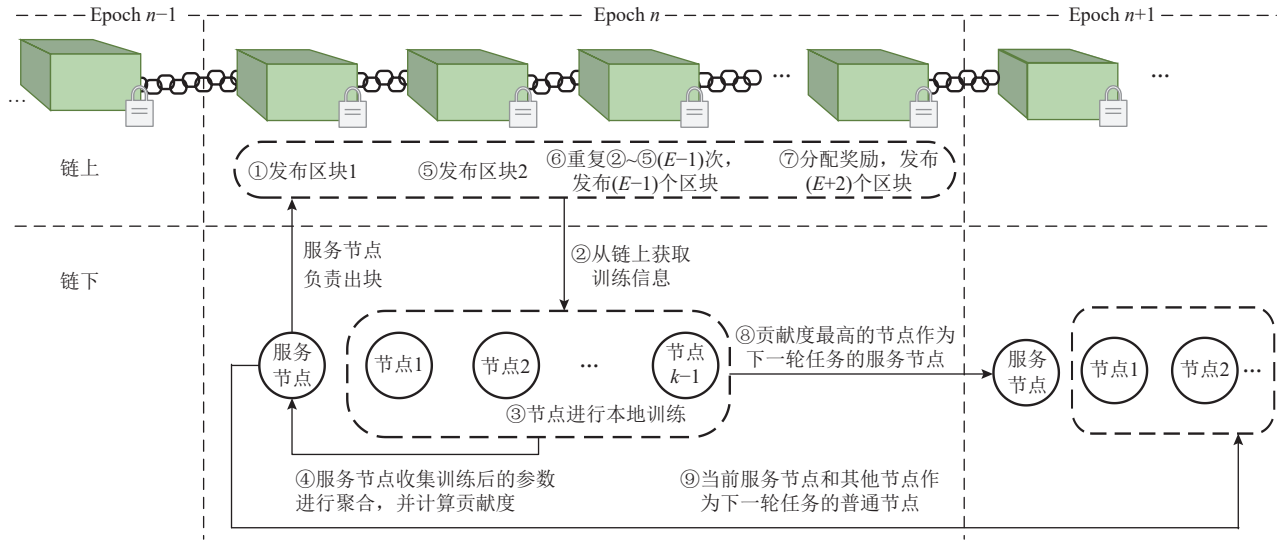


Fig. 2 Structure of and flowchart TFchain

图 2 TFchain 结构与流程图

地模型并上传训练后的模型参数，并根据贡献领取相应的奖励.

如果服务节点有作恶行为，普通节点也能够进行投票并永久排除本轮服务节点. 贡献度最高的本轮学习的参与方会成为下一轮任务的服务节点，然后进行下一轮学习任务的训练周期 Epoch，直到交易池中的学习任务列表为空. TFchain 可以通过奖励机制和投票机制来确保服务节点的公正性和参与方的利益. 因此基于 PoTF 的 TFchain 可以显著避免算力过度耗费，并进行公平的奖励分配.

下面介绍 TFchain 的共识切换策略. 本文在区块链交易池中独立建立了一个联邦学习任务列表. 学习任务发起方可以通过向区块链发送交易的方式发布学习任务，交易内容是学习任务的具体信息，包括任务名称、任务描述、任务数据集、参与方信息等. 在此处本文采用一种 PoTF 和 PoW 的共识切换机制，具体如图 3 所示.

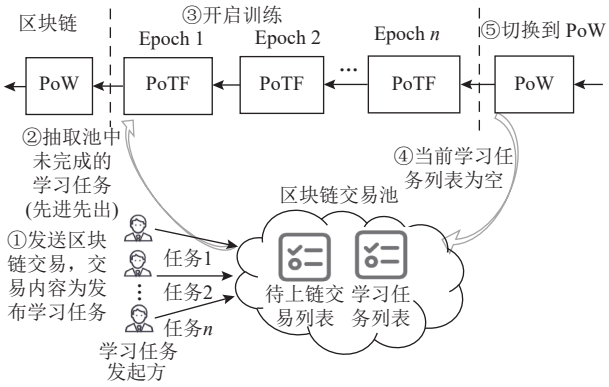


Fig. 3 Switched consensus mechanism of TFchain

图 3 TFchain 的共识切换机制

在这个机制中，TFchain 在运行过程中是基于交易池中的学习任务列表进行实时切换的，当新的任务加入到交易池中的学习任务列表时，应该采用先进先出的方式在交易池中进行存储. 而区块链对交易池的学习任务列表进行实时监测，当交易池中存在未训练的联邦学习任务时，将其取出，开启 PoTF 并且进行任务的训练，直到所有任务训练完毕. 当任务池中为空后，则切换到 PoW 共识.

此外，虽然本文框架主要服务于联邦学习任务的高效和安全运行，但是不参与联邦学习的普通区块链节点也可以加入到共识过程中. 这种不参与联邦学习的节点在 TFchain 的不同共识阶段可以执行不同的功能：在无联邦学习任务时，不参与联邦学习的节点可以参与到切换的正常 PoW 共识中，执行共识节点的任务；在系统中有联邦学习任务，需要进行 PoTF 共识时，不参与联邦学习的节点可以作为区块链的账本节点，同步共识区块链账本，并且进行交易和账本的正确性验证，进一步加强联邦学习期间的系统安全性.

以上就是 TFchain 的整体框架，这种框架不仅可以实现高效快速的区块链交易处理性能，确保联邦学习的数据隐私保护和模型快速更新，而且还能提供公平的联邦学习奖励以激励参与方的积极性.

3.2 PoTF 共识过程中的区块结构

PoTF 是本文提出的一种全新的基于联邦学习的共识算法，为了适应联邦学习的相关特性，对区块链的整体数据结构也进行了全新的设计.

由图 4 可知，PoTF 的区块头包含 7 方面内容：

1)前一区块哈希(Pre_Hash)；

2)当前的区块高度(Height)；

3)当前区块的交易默克尔树根(Merkle_Root)；

4)当前服务节点地址(Master_Node)，当本次学习任务结束，结束本 Epoch 的 PoTF 时，当前 Epoch 的最后一个区块要进行服务节点地址的更新，更新为选定的下一轮学习任务的服务节点；

5)本轮联邦学习的聚合参数(Con_parameter)，每个 Epoch 的区块 0 中设置为默认 0；

6)当前的任务信息(Task)；

7)其他相关信息(Additional_Data).

而 PoTF 中区块体主要包含的就是联邦学习中参与方每轮的本地训练参数和普通的区块链交易. 但是需要说明的是，在本 Epoch 的最后一个区块中，服务节点需要在区块体中公布本次学习任务中所评估的所有参与方的贡献度和奖励分配情况.

3.3 PoTF 共识算法

PoTF 共识算法是在 TFchain 检测到交易池中联邦学习任务列表中有未完成任务时开启的共识机制，该共识机制能够有效地提升区块链的吞吐量，在实际应用中，我们可以根据联邦学习应用领域的不同，依据 TFchain 框架建立多个基于 PoTF 共识的区块链分片，每个分片负责进行一个应用领域的联邦学习任务. 由于不同的应用领域学习任务差异大，也不存在跨分片交易问题，因此区块链的吞吐量与分片数量呈正相关趋势，能够大大提升区块链的可扩展性. 由于不同分片的工作流程是相同的，因此，本文统一使用单分片的架构进行描述，但是在实际应用中可以拓展到多应用场景、多分片的架构.

PoTF 算法总体的共识流程在图 2 的 TFchain 的总体框架图中有所展示，具体有 9 个步骤：

第 1 步，服务节点选择交易池中未完成的一个学习任务，并打包任务的发起方发布的任务信息和相关的模型信息到本 Epoch 的第 1 个区块(区块 0)中.

第 2 步，普通节点从区块 0 中同步训练任务信息及训练参数，训练参数应该包括训练轮数、学习率、损失函数等.

第 3 步，普通节点进行本地训练. 在本地训练过程中，客户端使用自己的数据集进行训练，然后将训练结果返回给服务节点.

第 4 步，服务节点收集训练后的参数并进行聚合. 在聚合过程中，服务节点记录聚合过程中的参数，然后计算参与到每次聚合的普通节点的贡献度.

第 5 步，服务节点发布区块. 区块包含默认信息、每个客户端的参数以及聚合后的参数. 此外，区块头还应包括前一个区块的哈希值、默克尔树根、参数的哈希值等信息.

第 6 步，重复第 2~5 步. 在每一轮训练中，客户端会使用新的训练参数进行本地训练，并将训练结果返回给服务节点. 服务节点收集这些结果，进行聚合并发布新的区块. 这个过程重复进行，直到训练轮数达到预设值.

第 7 步，服务节点依据参与节点的贡献度确定奖励的分配方式，并发布本 Epoch 的最后一个区块.

第 8 步，更新下一轮学习任务的服务节点(选择贡献度最高的参与方作为下一轮任务的服务节点，第一次联邦学习任务时，随机选择一个节点作为服

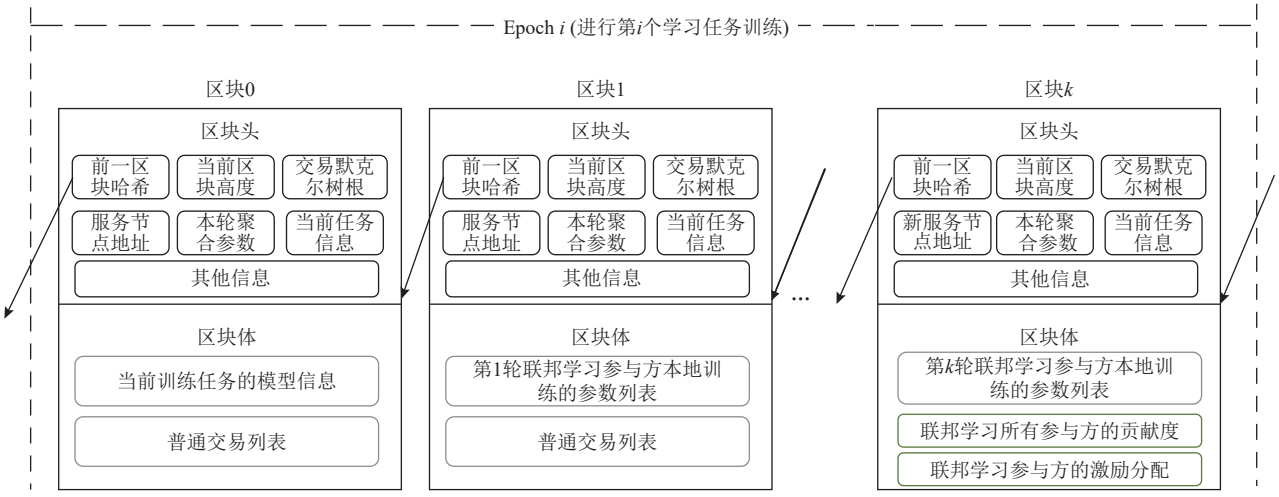


Fig. 4 Block structure of TFchain

图 4 TFchain 的区块结构

务节点)，并结束本次联邦学习任务.

第 9 步，当前服务节点则在下一轮任务中作为普通节点.

3.4 TFchain 的节点任务划分

本节将进一步对 3.3 节的 PoTF 中服务节点和普通节点的任务进行详细描述.

如图 5 所示，服务节点主要有 2 项并行任务，其一是每轮参数的收集和区块的打包，另一项是参与方的贡献度计算. 服务节点的具体任务执行流程如算法 1 所示.

具体而言，算法 1 主要分为 3 部分：1)任务初始化. 从交易池的学习任务列表中获取一个任务，并将初始信息打包出块. 2)训练. 在一定的时间内，接收参与节点发送的训练数据，并按照某个联邦聚合函数(可插拔可替换)对这些数据进行聚合，计算出全局模型，同时通过沙普利值计算贡献度，然后发布区块. 3) 贡献度计算及分配奖励. 根据沙普利值计算参与方的贡献度，并且分配奖励，其中服务节点领取 50% 的奖励，并按照各个节点的贡献度按比例分配剩余奖励给其他节点，并且将贡献度最高的普通节点设置为新的服务节点.

算法 1. 服务节点执行训练过程.

输入：交易池 *pool* 中的任务信息；

输出：经过良好训练的全局模型w，按照贡献进行的奖励分配 R.

(任务初始化)

① $modelMsg \leftarrow pop(pool)$; /*获取任务*/

② $addBlock(header(modelMsg, \cdots), body)$;

/*初始信息上链*/

(训练)

③ for each round t =1,2,⋯ do

④ $N_t \leftarrow$ (在规定时间内响应的参与节点);

⑤ for each round $k \in N_t$ do

⑥ $w_t^k \leftarrow getMsg(w_t, k)$;

⑦ $w_t \leftarrow \dfrac{\sum_{i=1}^{m} N_i w_i^t}{\sum_{i=1}^{m} N_i}$; /*聚合函数可以替换*/

⑧ if $MASE(SV_{t-timetemp}, \cdots, SV_{t-1}) > \Delta$

⑨ $SV_t^N \leftarrow \left(N_t, f\left(w_t^N\right)\right)$; /*式(5)*/

⑩ end if /*SV 计算与聚合并行*/

⑪ end for

⑫ $addBlock(header(w_t, \cdots), body(w_t^N, \cdots))$;

/*发布区块，其结构如图 4 所示*/

⑬ end for

(贡献度)

⑭ $R_n = \dfrac{\sum_{t=1}^{n} SV_t^n}{|t|}$; /*统计贡献度，并且发布区块*/

⑮ $addBlock(header(R_N, SV_T^N, \cdots), body)$;

⑯ $\_Ledger$ += $reward \times 50\%$;

/*服务节点拿走 50% 的奖励*/

⑰ for each round $n \in N$ do

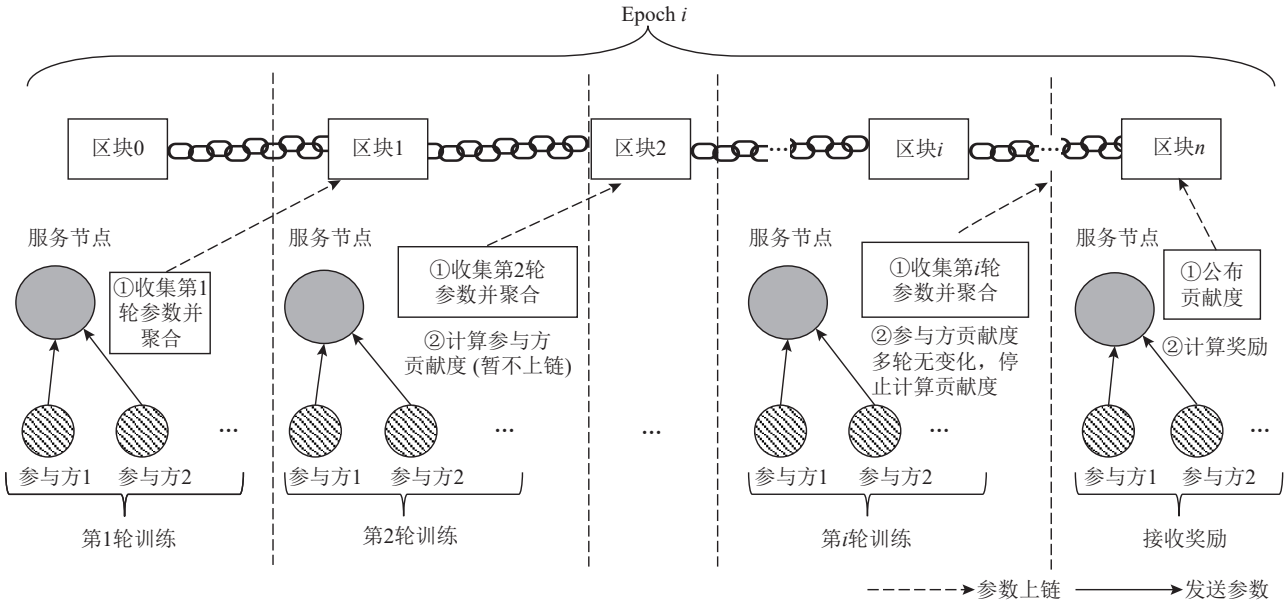


Fig. 5 Illustration of parallel task execution of service nodes in PoTF

图 5 PoTF 中服务节点的并行任务执行示意图

⑱ $sentReward(n, reward\times 50\%\times R_n)$;

⑲ end for /*按照贡献分配奖励给其他节点*/

⑳ $sentServerNode(max(R_n))$.

/*贡献度最高的节点为服务节点*/

需要说明的是,在多轮的联邦学习中,参与训练的节点对全局模型的影响会越来越小,其贡献度的变化也会逐渐变小.因此,当参与训练的节点贡献度变化过小时,会停止每一轮贡献度的计算,然后根据前期贡献度来计算每个参与方本次学习任务中的最终贡献度.

具体而言,任务初始化部分发布的区块对应图 5 中的区块 0,训练部分发布的区块对应区块 1~n−1,贡献度和奖励分配相关发布的区块对应区块 n.具体的区块结构如图 4 所示.

算法 1 使用的联邦聚合函数可以根据实际情况进行替换,从而提高算法的适用性和性能.算法 1 中,贡献度最高的普通节点可以成为新的服务节点,从而保证了服务节点的动态变化.

算法 2 详细介绍了 PoTF 的普通节点执行训练过程.

算法 2. 参与节点执行训练过程.

输入:区块链中的任务信息;

输出:经过良好训练的本地模型w,按照贡献领取奖励.

(任务初始化)

① $modelMsg \leftarrow getBlock()$; /*获取任务*/

(训练)

② for each round t =1,2,⋯ do

③ $w \leftarrow getBlock()$; /*获取初始信息 */

④ $B \leftarrow$ (将数据集 D_k 划分为 B 大小的批次);

⑤ for each round $b \in B$ do

⑥ $w \leftarrow w-\eta\nabla\ell(w, b)$;

⑦ end for

⑧ $sentMsg(w, serverNode)$;

⑨ end for

(贡献度)

⑩ $\_\ Ledger$ += $getReward()$.

算法 2 与算法 1 对应,同样分为 3 部分:1)任务初始化.从区块链中获取初始信息,设置模型.2)本地训练.在一定的时间内,将数据集 D_k 划分为大小为 B 的批次,并在每个批次上进行训练,得到本地模型.3)贡献度获取.节点完成训练后,根据节点的贡献度领取相应的奖励.

算法 2 相对于算法 1 来说,更加简单易懂,易于实现和部署,因此适用于部署在区块链的普通节点中,作为联邦学习的参与方参与到共识和联邦学习中.同时,算法 2 中节点之间只需传输本地模型,而不需要传输所有数据,因此节省了通信量,减少了网络传输时间.

3.5 TFchain 的贡献计算及奖励分配

在 PoTF 运行的过程中,服务节点会使用沙普利值的计算方式,在链下持续计算每个参与方在整个联邦学习过程中的贡献度,在每次收到参与节点的本地模型,对全局模型进行更新后,计算更新后的模型的提升程度,该值将用于计算参与节点的沙普利值.并且在打包本 Epoch 的最后一个区块时,将每个节点的最终贡献度公布,并且根据贡献度分配奖励.

本文设定的奖励分配机制是:本次学习任务中 50% 的奖励将分配给本轮共识的服务节点,剩下 50% 的奖励按照其他参与方的贡献度进行等比例分配,并且参与方中贡献度最高的节点自动成为下一轮学习任务和 PoTF 的服务节点(本轮服务节点由于不参与本地模型的训练,因此不会成为连续当选下一轮的服务节点),能够在下一轮 PoTF 共识中获得成为服务节点的 50% 的奖励.

3.6 TFchain 的服务节点作恶检测机制

服务节点作恶可能表现为恶意修改参数信息或者伪造贡献度信息,因此如果最近 m 个区块的参数信息、贡献度不符合预期,就有可能发现服务节点作恶.

针对贡献度计算和区块打包中忽略某些客户端参数的作恶行为,本文设计了基于投票的服务节点作恶检测和更换机制,当普通节点(联邦学习的参与方)发现服务节点作恶(贡献分配不平均或多次忽视某一客户端的参数不进行聚合)后,该节点会发起更换服务节点的投票并广播给全部节点的方式进行服务节点的更换.

本文设计的服务节点作恶检测和更换机制可以有效防止服务节点作恶.如果一个服务节点无法按照预期工作,其他节点可以通过发起投票来选举新的服务节点,保证共识算法的正常运行.在本文中,我们设定投票需要 2/3 的节点同意才能进行服务节点变更,这个规则是为了保证新的服务节点能够获得足够多的支持,从而使整个系统能够继续正常工作.同时,这也是防止恶意节点随意发起更换服务节点的关键措施.

基于以上思路,本文提出了算法 3,用于服务节点的作恶检测和更换.

算法 3. 参与节点发现主节点作恶后的更换过程.

输入：质疑信息 *Qmsg*；

输出：投票同意服务节点作恶的节点大于 2/3，如果是则更换服务节点.

（发起质疑）

① *sentMsg*(*Qmsg*, *allNode*)；

（投票）

② *Msgcount* = 0；

③ while *nowTime*<*endTime* 或 *Msgcount*<*N*×2/3:

④ if *receiveMsg* ≠ Null

⑤ *Msgcount*++；

⑥ end if

⑦ end while

（同意服务节点作恶的节点小于 2/3，向其他节点确认更换主节点）

⑧ if *Msgcount* > *N*×2/3

⑨ *sentMsg*(*changeServerNode*, *allNode*)；

⑩ *Changecount* = 0；

⑪ while *Changecount* < *N*×2/3

⑫ if *receiveMsg* ≠ Null

⑬ *Changecount* ++；

⑭ end if

⑮ end while

（主节点替换）

⑯ *sentServerNode*(*max*($R_{N\text{-}ServerNode}$)).

⑰ end if

算法 3 描述了一个基于投票的机制来替换作恶的服务节点的过程，主要分为 5 部分：

1）发起质疑. 某个节点发现服务节点作恶后，会向其他节点发起质疑，引发投票替换服务节点.

2）投票. 节点收到质疑后，在一定时间内，如果有超过 2/3 的节点同意替换服务节点，则进入下一步.

如果同意替换的节点不足 2/3，则认为投票结果不足以作为更换服务节点的依据，返回 Null.

3）向其他节点确认更换服务节点. 如果投票结果符合条件，则通知其他节点确认更换服务节点.

4）服务节点替换. 如果有超过 2/3 的节点同意更换服务节点，则选择贡献度最高的节点作为新服务节点，替换原服务节点.

5）切换服务节点. 完成服务节点替换后，返回新服务节点.

通过算法 3 的投票机制能够确保服务节点的公正性和可信度. 如果投票通过，会重新选择贡献度第 2 高的节点作为新服务节点，能够保证整个网络的稳定性和效率；如果投票结果不足以作为更换服务节点的依据，算法会终止，能够防止虚假投票的发生.

4 安全性分析

4.1 服务节点的作恶

服务节点的作恶行为可以分为联邦学习中作恶、贡献度计算中作恶、区块打包中忽略某些客户端参数作恶等 3 种行为.

针对联邦学习中服务节点投毒攻击等作恶方式，由于服务节点不参与本轮联邦学习的本地模型训练，只负责参数的聚合，所以无法进行这种方式的作恶.

针对服务节点的以下 2 种作恶行为：贡献度计算中作恶和区块打包中忽略某些客户端参数作恶，我们设计了有效的服务节点作恶检测和更换策略. 而根据策略的设计，从理论上来说，如果有超过 1/3 的节点是恶意的，那么算法就无法保证安全性和正确性. 因为恶意节点可以通过阻止投票数超过 2/3，来阻止服务节点变更的发生，从而导致整个系统无法正常工作. 本文的共识安全性是建立在节点数量和诚实节点比例的基础之上的.

在实际应用中，如果网络中存在大量恶意节点，那么就需要采取其他措施来提高系统的安全性，但是本文的环境是相互协作的联邦学习客户端作为节点，即使有作恶节点也很难超过总节点数的 1/3.

4.2 普通节点的作恶

普通节点的作恶行为可以分为拒绝学习作恶和参数篡改作恶 2 种行为. 前者节点可能会故意中断其与服务节点的通信；后者节点可能会篡改模型，使其产生错误的输出结果或引导全局模型发生偏移.

针对普通节点拒绝学习作恶，即节点不进行本地训练更新参数，影响总体联邦学习训练效率的行为，本文通过公正的激励机制来鼓励参与者的积极参与，从而减少参与者的作恶动机.

为了证明本节的分析，我们可以通过为参与者提供一定的奖励来激励其诚实地参与联邦学习，我们实现了不同的作恶节点比例下的联邦学习的训练和贡献度的计算过程. 实验结果表明，在本文设计的框架下，参与者如果拒绝参与联邦学习，则贡献度与诚实节点将会逐渐拉大，无法获得相应的奖励，从而减少其作恶动机. 由图 6 可知，如果系统中没有节点作恶，诚实节点正常进行本地模型的聚合并且发送参数给服务节点，每个参与方的贡献基本与其训练精度相对应，但是一旦出现节点作恶，随着作恶节点

比例的提升,诚实节点的贡献度会被评估得越来越高,并且根据图 7 可以发现,虽然作恶节点的总体比例在上升,但是这些节点的总贡献度占全体贡献度的比例却很少,从而保证诚实节点拿到的系统激励远高于作恶节点.

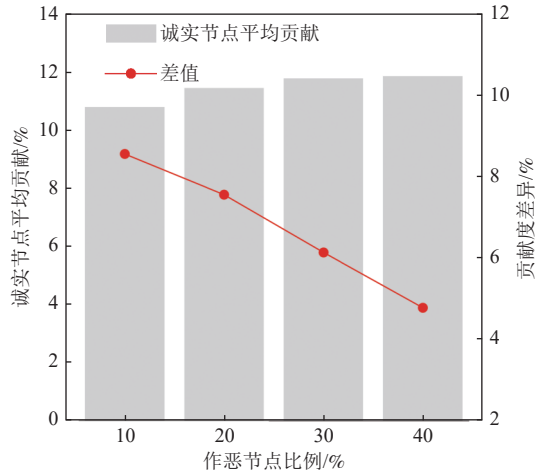


Fig. 6 The average contribution of nodes with different percentage of bad behaviors

图 6 不同作恶比例下节点的平均贡献度

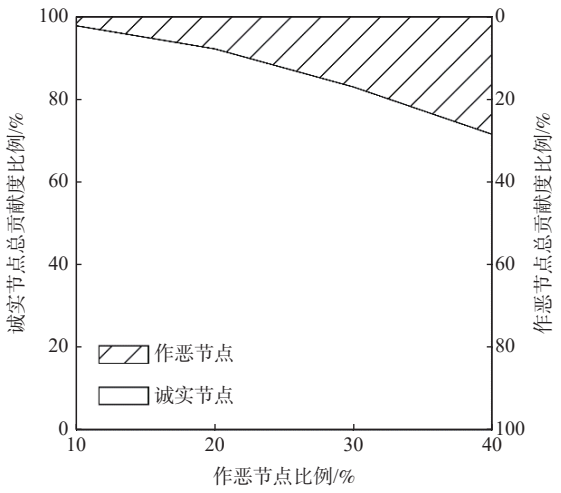


Fig. 7 All contribution of nodes with different percentage of bad behaviors

图 7 不同作恶比例下节点的总贡献度

如果普通节点不在意奖励,只是想破坏全局模型进行作恶,通常会选择参数篡改作恶.针对普通节点参数篡改作恶,由图 6 可以观察到,作恶节点与诚实节点之间的贡献度差异会被逐渐拉大,最终该作恶节点将会被评估为低贡献度,累计到一定程度后被剔除出联邦学习.在这种机制下,普通节点无法进行这种方式的作恶.

5 实 验

5.1 实验设置

在本节中将 TFchain 与基于 PoW 的区块链系统、基于联邦学习贡献度的区块链 FedCoin 进行比较. PoW[2] 是大多数公有链系统采用的共识机制. FedCoin[24] 是一种基于区块链的支付系统,其中引入了沙普利值来评估参与者的贡献度,但是其联邦学习过程是在链外进行的.而且 FedCoin 只将训练结果和过程参数上链记录,然后再计算参与方的贡献度.因此,FedCoin 中区块链和联邦学习部分是分离的.

本文在 2 个图像数据集上进行实验: MNIST[34] 和 EMNIST[35]. MNIST 是一个手写数字数据库,通常用于训练各种图像处理系统. EMNIST 是 MNIST 的扩展版.

下面的实验如果没有特殊说明,基本设置为:

本文共识切换中设置的 PoW 算法挖矿难度为 4,对比的 PoW 算法也将挖矿难度设置为 4, FedCoin 与 TFchain 进行的联邦学习训练任务是相同的.

联邦学习任务中运行 200 个全局通信轮,设置了总参与用户数量为 20,每次通信的用户数量为 10. 采用局部更新步骤 T= 20,每一步使用一个大小为 B=32 的迷你批处理. 设置的基本聚合函数是 FedAvg[32],本地模型是一个基本 CNN 模型. 实验结果采样自 3 个不同随机种子的均值和标准差.

本节所有的实验均在 Intel® Core™ i9-10900k CPU @ 3.70 GHz 3.60 GHz 且具有 64GB RAM 的 PC 平台上完成.

5.2 算力分析

目前在比特币、以太坊等公有链平台进行 PoW 共识的过程中,99% 的算力都是无效的哈希运算,并且随着挖矿难度的提升,挖掘出一个区块的哈希计算时间呈指数级增长,无效算力的比例也会逐渐增加,导致区块链的交易吞吐量显著减少, PoW 中基本所有的工作量都是在进行无意义的哈希计算,以达到设定的难度值 x(最终的哈希结果前 x 位为 0),通过图 8 和图 9 可以得知,随着目标难度值 x 的增加, PoW 挖掘出一个块的时间显著增加,尤其是难度值大于 6 之后,计算时间高达 160 s 以上,相应的交易吞吐量也呈现指数级下降.

由图 8 可知,在难度值为 4 的情况下,吞吐量下降到 400 左右;而难度值为 5 时,交易的吞吐量仅为

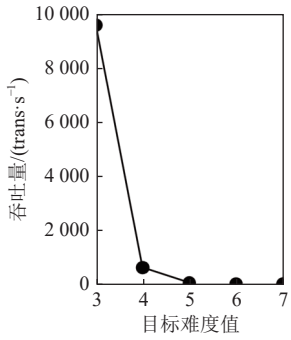


Fig. 8 PoW troughputs

图 8 PoW 吞吐量

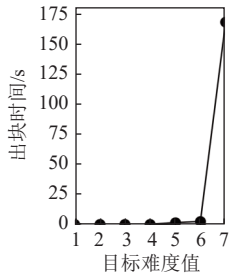


Fig. 9 PoW computing time for building a block

图 9 PoW 出块计算时间

42. 在后续的实验中为了方便衡量和计算，我们选择了难度值为 4 作为实验基准，根据图 9 可知，在难度值为 4 的情况下进行 6 500 次哈希能够挖掘出一个块. 当然，实际的比特币和以太坊等区块链平台中，目标难度值已经非常大，目前需要 46.84 万亿次哈希计算才能挖掘出一个区块.

根据图 10 和表 1，即使在难度值仅为 1 的情况下，无效哈希计算的占比也高达 95% 以上；而在难度值到达 3 时，基本无效算力的比例已经在 99% 以上了，此时哈希计算的次数在 4 500 左右；而当难度值设定为 7 时，哈希计算的次数为 9 千万以上. 由此可见，PoW 共识过程中，大量算力都用在了没有意义的哈希计算中.

根据图 8 可知，设定难度值为 7 的情况下，同样时间下，PoW 挖掘出一个区块的时间为 168 s，而根据图 11 可知，对 MNIST 和 EMNIST 数据集各自进行 200 轮聚合的联邦学习仅需要 25s 和 78 s 左右. 本文的共识是每进行 1 轮联邦学习的参数聚合就打包 1 个区块，因此本文的共识机制能够有效地将 PoW 耗费的大量算力应用在联邦学习中进行高效地训练，大大提升区块链的交易处理性能、降低交易延迟.

5.3 区块链的吞吐量和交易延迟

我们分别实现了基于 PoW 的区块链系统、基于联邦学习贡献度的区块链 FedCoin 和本文提出的基于联邦学习的 TFchain，并且我们同时实现了 TFchain 的学习训练池中持续有联邦学习任务只进行 PoTF 共识的情况和当学习训练池中任务列表为空需要切换共识算法 TFchain-Switch 这 2 种情况下的交易吞吐量和交易延迟情况.

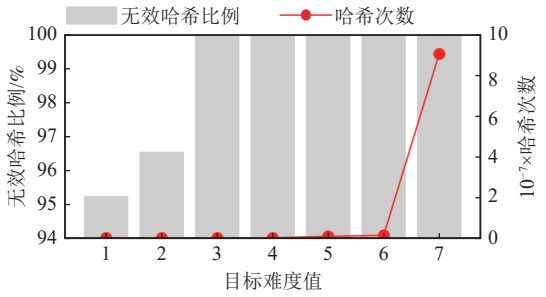


Fig. 10 Invalid hashes' percentage and counts of hashes

图 10 无效哈希比例和哈希次数

Table 1 Invalid Hashes' Percentage and Counts of Hashes

表 1 无效哈希比例和哈希次数

目标难度值	哈希次数	无效哈希比例/%
1	21	95.24
2	29	96.55
3	4 407	99.98
4	6 565	99.98
5	749 278	100.00
6	1 255 485	100.00
7	90 700 714	100.00

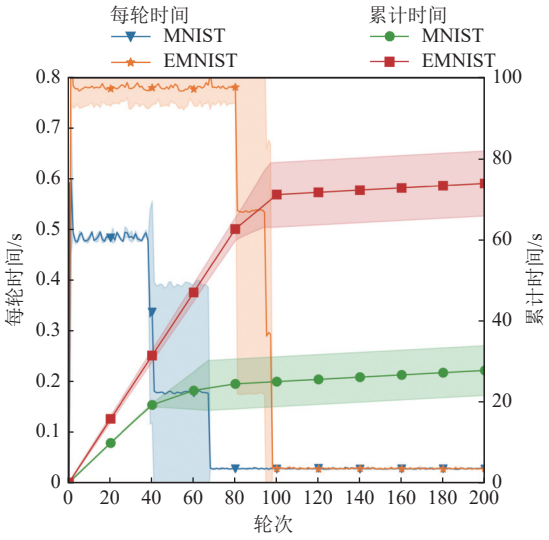


Fig. 11 Learning time curve of TFchain

图 11 TFchain 的学习时间曲线

根据 5.2 节中对 PoW 的算力分析，我们将 PoW 的挖矿难度设置为 4，即要求哈希计算中得到的字符串前 4 位为 0.

由图 12 和图 13 可以看出，4 种共识算法都会随

着一个区块中可以打包的交易数量 Blocksize 的增加呈近线性增加，而交易的处理延迟随着 Blocksize 的增加而减少，这是因为随着一个区块可以打包的交易数量的增加，交易就可以更快地上链，从而增加吞吐量和降低交易延迟. 其中，PoTF 的吞吐量是 4 种共识算法中最高的，而交易延迟是最低的，当 Blocksize 为 1 000 笔时，PoTF 的吞吐量可以高达 18 000 trans/s 左右，而交易延迟仅为 0.55 s 左右.

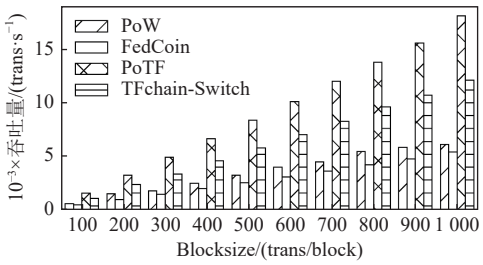


Fig. 12 Comparison of throughputs

图 12 吞吐量对比图

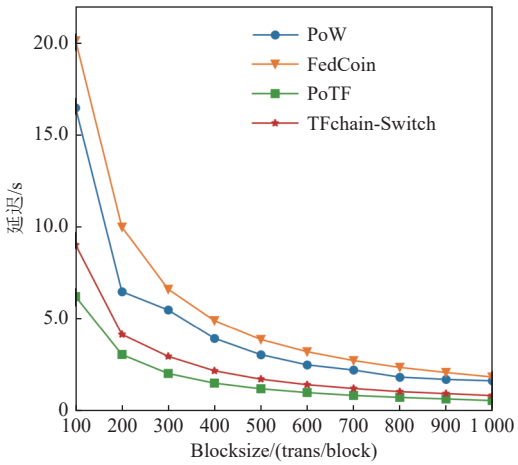


Fig. 13 Comparison of latency

图 13 延迟对比

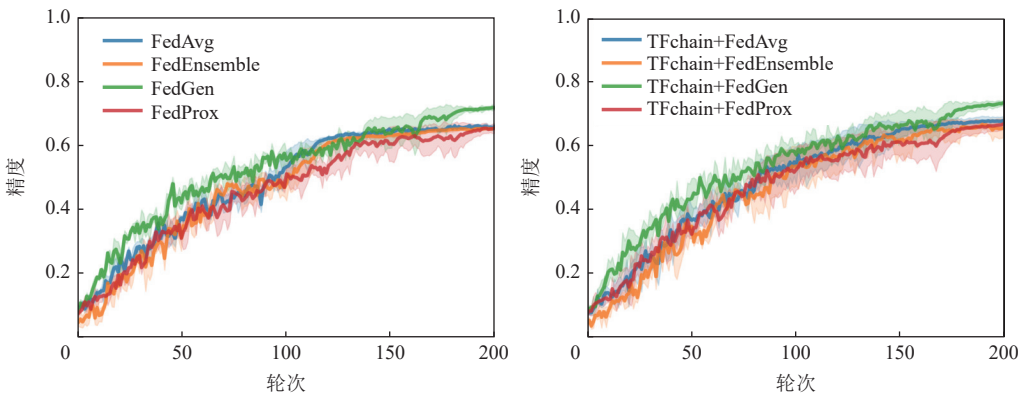
FedCoin 是 4 种方法中吞吐量最低且交易延迟最大，这是由于 FedCoin 中联邦学习的过程和参数聚合的过程是在区块链外进行的，仅仅将沙普利值的计算用于区块工作量计算的证明，但是每挖掘一个区块都需要等待学习的结果，因此 FedCoin 的交易吞吐量比较低. PoW 共识算法在挖矿难度值为 4 的设置下吞吐量排在第三，而 TFchain-Switch 之所以吞吐量相对于 PoTF 降低且交易延迟增加，就是因为在没有学习任务的情况下会切换共识算法到共识效率比较慢的 PoW 算法.

5.4 联邦学习精度分析

本文使用 FedAvg[32], FedEnsemble[36], FedGen[37] 和 FedProx[38] 等 4 种联邦学习的常用模型独立对 MNIST[34] 和 EMNIST[35] 数据集进行了联邦学习的 CNN 模型训练，同步测试了将这些联邦学习模型嵌入到本文设计的区块链框架中的模型训练情况，并且将 2 种情况进行了对比分析.

使用区块链来发布联邦学习参数应该不会影响联邦学习的精度. 区块链可以提供分散的、可验证的存储，使得参数可以被多个参与方共同访问. 由于区块链的去中心化特性，每个参与方都可以验证自己收到的参数是否与其他参与方收到的参数一致，从而保证了参数的一致性. 根据图 14 和图 15，在将 4 种模型嵌入到本文的区块链框架中后，2 个数据集的模型训练精度都变化不大.

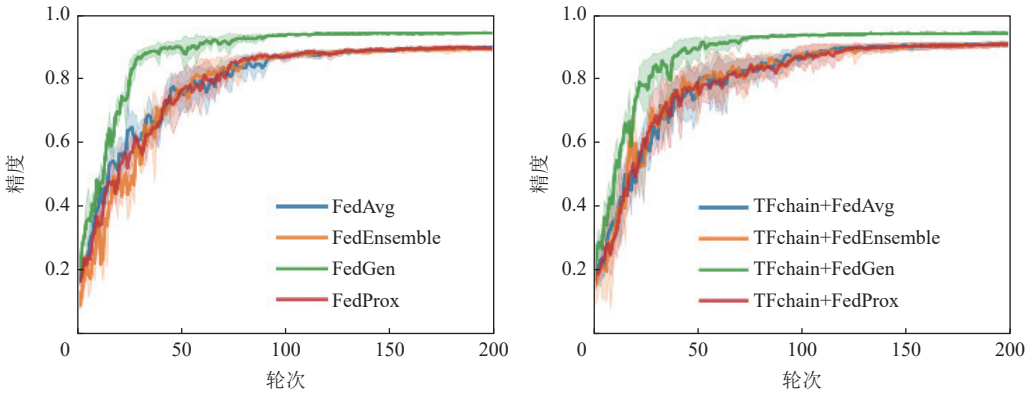
在 200 轮联邦学习后，MNIST 数据集的模型训练精度在使用了 4 种联邦模型的情况下都达到了 90% 以上，其中 FedGen 表现最好，EMNIST 的最终精度都在 60% 以上，也是 FedGen 表现最好. 嵌入到本文的区块链框架后，不仅没有影响联邦学习的精度，表现反而比原来的训练略有提升. 这是因为区块链



(a) EMNIST上的学习曲线 (b) 结合TFchain后在EMNIST上的学习曲线

Fig. 14 Comparison of learning curves on EMNIST

图 14 EMNIST 上的学习曲线对比



(a) MNIST上的学习曲线 (b) 结合TFchain后在MNIST上的学习曲线

Fig. 15 Comparison of learning curves on MNIST

图 15 MNIST 上的学习曲线对比

一方面通过不可篡改的账本保证了参数的一致性，提供了更安全、更可靠、更分散的参数共享方式；另一方面我们的算法通过服务节点的作恶检测和激励机制一定程度上降低了联邦参与方的作恶行为，因此能够让联邦学习参与方更积极地参与到模型的训练中.

6 结 论

本文提出了一种基于区块链和联邦学习的全新共识机制和可信区块链框架，旨在解决传统共识机制存在的算力过度耗费和吞吐量低等问题，同时提高联邦学习的效率. 该框架将区块链的节点设置为联邦学习的参与方，通过有效利用原本共识机制中耗费的大量算力来进行本地模型的训练和参与方贡献度的评估，实现了对区块链交易吞吐量的提升以及对联邦学习参与方的合理评估和激励. 同时，本文还设计了一种防止节点作恶的算法，保障了系统的安全性和可靠性. 实验结果表明，所提出的共识机制不仅有效节约了算力资源，提升了区块链的交易处理性能，而且能够防止联邦学习中参与方的作恶行为，并对积极参与联邦学习的参与方进行有效正向的激励.

本文研究为区块链技术和联邦学习的应用提供了一种全新的解决方案，适用于所有需要进行联邦学习的应用，例如医疗信息诊断、物联网信息溯源分类等，本文提出的 TFchain 在增加联邦学习安全可信性的同时，也增加了区块链的吞吐量，具有实际应用价值和推广意义.

未来工作将从 2 方面开展研究：1）本文的共识切换机制是将 PoTF 与 PoW 进行切换，后续可以考虑将切换的共识机制变为可插拔的，例如可以变为 PoTF 与 PBFT 共识算法的切换等；2）为了应对强人工智能学习背景下大量联邦学习任务急需执行的情况，可以结合区块链的分片技术，设计全新的区块链结构，将不同的联邦学习任务分流到不同的区块链分片上训练.

作者贡献声明：张宝晨和黄月作为共同第一作者，提出论文整体思路，完成实验，撰写论文；孔兰菊和李庆忠提出指导意见；李文全和郭秋曼疏理算法思路，撰写论文.

参 考 文 献

[1] Fu Xiang, Wang Huaimin, Shi Peichang. A survey of blockchain consensus algorithms: Mechanism, design and applications[J]. Science China Information Sciences, 2021, 64: 1−15

[2] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[OL]. [2022-09-03]. https://bitcoin.org/bitcoin.pdf

[3] Qu Xidi, Wang Shengling, Hu Qin, et al. Proof of federated learning: A novel energy-recycling consensus algorithm[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(8): 2074−2085

[4] Singh P, Singh M K, Singh R, et al. Federated learning: Challenges, Methods, and Future Directions[M]//Federated Learning for IoT Applications. Cham, Switzerland: Springer, 2022: 199−214

[5] Li Li, Fan Yuxi, Tse Mike, et al. A review of applications in federated learning[J]. Computers & Industrial Engineering, 2020, 149: 106854

[6] Li Dun, Han Dezhi, Weng Tien-Hsiung, et al. Blockchain for federated learning toward secure distributed machine learning systems: A systemic survey[J]. Soft Computing, 2022, 26(9): 4423−4440

[7] King S, Nadal S. PPCoin: Peer-to-Peer crypto-currency with proof-of-stake[OL]. [2022-09-05]. https:// bitcoin.org/bitcoin.pdf

[8] Yang Fan, Zhou Wei, Wu Qingqing, et al. Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism[J]. IEEE Access, 2019, 7: 118541−118555

[9] Castro M, Liskov B. Practical Byzantine fault tolerance and proactive recovery[J]. ACM Transactions on Computer Systems, 2002, 20(4): 398−461

[10] Wood G. Ethereum: A secure decentralised generalised transaction ledger[J]. Ethereum Project Yellow Paper, 2014, 151(2014): 1−32

[11] Feng Jie, Yu F R, Pei Qingqi, et al. Joint optimization of radio and computational resources allocation in blockchain-enabled mobile edge computing systems[J]. IEEE Transactions on Wireless Communications, 2020, 19(6): 4321−4334

[12] Xu Xiaolong, Zhang Xuyun, Gao Honghao, et al. BeCome: Blockchain-enabled computation offloading for IoT in mobile edge computing[J]. IEEE Transactions on Industrial Informatics, 2019, 16(6): 4187−4195

[13] Khan L U, Pandey S R, Tran N H, et al. Federated learning for edge networks: Resource optimization and incentive mechanism[J]. IEEE Communications Magazine, 2020, 58(10): 88−93

[14] Hu Rui, Gong Yanmin. Trading data for learning: Incentive mechanism for on-device federated learning[C] //Proc of Global Communications Conf (GLOBECOM). Piscataway, NJ: IEEE, 2020: 1−6

[15] Deng Yongheng, Lyu Feng, Ren Ju, et al. Fair: Quality-aware federated learning with precise user incentive and model aggregation[C] //Proc of IEEE Conf on Computer Communications (INFOCOM). Piscataway, NJ: IEEE, 2021: 1−10

[16] Weng Jiasi, Weng Jian, Huang Hongwei, et al. Fedserving: A federated prediction serving framework based on incentive mechanism[C] //Proc of IEEE Conf on Computer Communications (INFOCOM). Piscataway, NJ: IEEE, 2021: 1−10

[17] Ng J S, Lim W Y B, Xiong Zehui, et al. A hierarchical incentive design toward motivating participation in coded federated learning[J]. IEEE Journal on Selected Areas in Communications, 2021, 40(1): 359−375

[18] Wang Zhilin, Hu Qin. Blockchain-based federated learning: A comprehensive survey[J]. arXiv preprint, arXiv: 2110.02182, 2021

[19] Zhang Jingwen, Wu Yuezhou, Pan Rong. Incentive mechanism for horizontal federated learning based on reputation and reverse auction[C] //Proc of the Web Conf (WWW). New York: ACM, 2021: 947−956

[20] Gao Liang, Li Li, Chen Yingwen, et al. FGFL: A blockchain-based fair incentive governor for federated learning[J]. Journal of Parallel and Distributed Computing, 2022, 163: 283−299

[21] Wang Yuntao, Peng Haixia, Su Zhou, et al. A platform-free proof of federated learning consensus mechanism for sustainable block-chains[J]. IEEE Journal on Selected Areas in Communications, 2022, 40(12): 3305−3324

[22] Zhu Jianming, Zhang Qinnan, Gao Sheng, et al. Privacy preserving and trustworthy federated learning model based on blockchain[J]. Chinese Journal of Computers, 2021, 44(12): 2464−2484 (in Chinese)
(朱建明, 张沁楠, 高胜, 等. 基于区块链的隐私保护可信联邦学习模型[J]. 计算机学报, 2021, 44(12): 2464−2484)

[23] Shubik M. Game Theory in the Social Sciences: Concepts and Solutions[M]. Cambridge, MA: MIT Press, 1982

[24] Liu Yuan, Ai Zhengpeng, Sun Shuai, et al. FedCoin: A Peer-to-peer Payment System for Federated Learning[M]//Federated Learning: Privacy and Incentive. Cham, Switzerland: Springer, 2020: 125−138

[25] Qiao Shaojie, Lin Yufeng, Han Nan, et al. Decentralized federated learning framework based on proof-of-contribution consensus mechanism[J]. Journal of Software, 2023, 34(3): 1148−1167(in Chinese)
(乔少杰, 林羽丰, 韩楠, 等. 新型基于贡献度证明共识机制的去中心化联邦学习框架[J]. 软件学报, 2023, 34(3): 1148−1167)

[26] Feng Lei, Zhao Yiqi, Guo Shaoyong, et al. BAFL: A blockchain-based asynchronous federated learning framework[J]. IEEE Transactions on Computers, 2021, 71(5): 1092−1103

[27] Shayan M, Fung C, Yoon C J M, et al. Biscotti: A blockchain system for private and secure federated learning[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 32(7): 1513−1525

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

[29] Yuan Shuo, Cao Bin, Peng Mugen, et al. ChainsFL: Blockchain-driven federated learning from design to realization[C] //Proc of 2021 IEEE Wireless Communications and Networking Conf (WCNC). Piscataway, NJ: IEEE, 2021: 1−6

[30] 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

[31] Rehman M H, Salah K, Damiani E, et al. Towards blockchain-based reputation-aware federated learning[C] //Proc of IEEE Conf on Computer Communications Workshops (INFOCOM Workshops). Piscataway, NJ: IEEE, 2020: 183−188

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

[33] Roth A E. The Shapley Value: Essays in Honor of Lloyd S. Shapley[M]. Cambridge, UK: Cambridge University Press, 1988

[34] Yann L, Corinna C, Christopher J C B. The MNIST database of handwritten digits[DB/OL]. [2018-09-15]. http://yann.lecun.com/exdb/mnist/

[35] Cohen G, Afshar S, Tapson J, et al. EMNIST: An extension of MNIST to handwritten letters [DB/OL]. [2018-09-15]. http://arxiv.org/abs/1702.05373

[36] Shi N, Lai F, Kontar R A, et al. FedEnsemble: Improving generalization through model ensembling in federated learning[J]. arXiv preprint, arXiv: 2107.10663, 2021

[37] Zhu Zhuangdi, Hong Junyuan, Zhou Jiayu. Data-free knowledge distillation for heterogeneous federated learning[C] //Proc of Int Conf on Machine Learning(ICML). New York: PMLR, 2021: 12878−12889

[38] Li Tian, Sahu A K, Zaheer M, et al. Federated optimization in heterogeneous networks[J]. Proceedings of Machine Learning and Systems, 2020, 2: 429−450

Zhang Baochen, born in 1992. PhD candidate. Student member of CCF. Her main research interests include sharding structure of blockchain, verification of blockchain transactions, and blockchain consensus algorithms.

张宝晨, 1992 年生. 博士研究生. CCF 学生会员. 主要研究方向为区块链分片架构、区块链交易验证、区块链共识算法.

Huang Yue, born in 1996. PhD candidate. Student member of CCF. Her main research interest includes federated learning.

黄　月, 1996 年生. 博士研究生. CCF 学生会员. 主要研究方向为联邦学习.

Kong Lanju, born in 1978. PhD, professor, PhD supervisor. Senior member of CCF. Her main research interests include blockchain and cloud computing.

孔兰菊, 1978 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为区块链和云计算.

Li Qingzhong, born in 1965. PhD, professor, PhD supervisor. Senior member of CCF. His main research interests include blockchain and privacy protection.

李庆忠, 1965 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为区块链和隐私保护.

Li Wenquan, born in 1995. PhD candidate. Student member of CCF. His main research interests include distributed systems, blockchain, and transaction concurrency.

李文全, 1995 年生. 博士研究生. CCF 学生会员. 主要研究方向为分布式系统、区块链、交易并发.

Guo Qiuman, born in 2000. Master candidate. Student member of CCF. Her main research interests include blockchain consensus and transaction concurrency.

郭秋曼, 2000 年生. 硕士研究生. CCF 学生会员. 主要研究方向为区块链共识和交易并发.