软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, [doi: 10.13328/j.cnki.jos.006150]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

区块链共识协议综述*

夏 清[1,4], 窦文生[2,4], 郭凯文[3,4], 梁 赓[1], 左 春[5], 张凤军[1]

[1](区块链技术与应用联合实验室(中国科学院 软件研究所),北京 100190)

[2](计算机科学国家重点实验室(中国科学院 软件研究所),北京 100190)

[3](可信计算与信息保障实验室(中国科学院 软件研究所),北京 100190)

[4](中国科学院大学,北京 100049)

[5](中科软科技股份有限公司,北京 100190)

通讯作者: 窦文生, E-mail: wensheng@iscas.ac.cn

摘 要: 共识协议作为区块链的核心技术,已经得到学术界和产业界的广泛重视,并取得一系列研究成果。当前关于共识协议的综述研究一般将共识协议作为整体进行比较分析,缺乏对共识协议中主要步骤的解耦与比较。本文将共识协议分为出块节点选举和主链共识两个主要步骤,并针对每个步骤进行协议间的分析比较。在出块节点选举部分,主要讨论工作量证明和权益证明,分析其中存在的问题、以及相应解决方案的分类比较。在主链共识部分,针对概率性共识和确定性共识,总结其安全目标,并进行安全性分析比较。通过共识协议的系统梳理,最后总结区块链共识协议的发展现状和发展趋势,以及未来重要研究方向。

关键词: 区块链;共识协议;出块节点选举;主链共识;工作量证明

中图法分类号: TP311

中文引用格式: 夏清,窦文生,郭凯文,梁赓,左春,张凤军.区块链共识协议综述.软件学报. http://www.jos.org.cn/1000-9825/6150.htm

英文引用格式: Xia Q, Dou WS, Guo KW, Liang G, Zuo C, Zhang FJ. A survey on blockchain consensus protocol. Ruan Jian Xue Bao/Journal of Software, 2018 (in Chinese). http://www.jos.org.cn/1000-9825/6150.htm

Survey on Blockchain Consensus Protocol

XIA Qing[1,4], DOU Wen-Sheng[2,4], GUO Kai-Wen[3,4], LIANG Geng[1], ZUO Chun[5], ZHANG Feng-Jun[1]

[1](Joint Laboratory of Blockchain Technology and Application (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[2](State Key Laboratory of Computer Science (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[3](State Key Laboratory of Computer Science (Institute of Software, Chinese Academy of Sciences), Beijing 100190, China)

[4](University of Chinese Academy of Sciences, Beijing 100049, China)

[5](Sinosoft Company Limited, Beijing 100190, China)

Abstract: As the core technology of blockchain, consensus protocol has received great attention from academy and industry in recent years. Existing surveys on blockchain generally regard the consensus protocol as a whole, and do not decouple and compare its main components. In this survey, we divide the consensus protocol into two main components, i.e., blocker election and main chain consensus. In the blocker election component, we mainly discuss two mechanisms, i.e., Proof of Work and Proof of Stake. For each mechanism, we analyze the encountered problems and compare their corresponding solutions. In the main chain consensus component, we summarize its security goal and conduct security comparison for probabilistic consensus and deterministic consensus. Through our comprehensive review of the state-of-the-art consensus protocol in blockchain, we finally summarize some important research directions for blockchain consensus.

Key words: blockchain; consensus protocol; blocker election; main chain consensus; proof of work

自 2009 年比特币[1, 2]系统出现以来,其底层区块链技术逐渐受到来自学术界和产业界的关注,并获得快速的发展。当前区块链技术已经得到广泛使用。例如,基于数字货币的跨境支付作为区块链技术的一大应用场景,近年来已经广为人知并开始被商家接受[3];一系列基于区块链技术的金融服务[4, 5]、供应链技术[5, 6]、政府治理[7]、游戏服务[8]等已经出现并获得成功运用。

区块链作为一种链式数据结构,由不断增长的区块利用哈希指针前后链接而成。区块链中的数据只能追加、不可删除或篡改。区块链系统是一个典型的分布式系统,其中每个节点维护一份本地区块链数据备份。在区块链系统中,区块链共识协议制

基金项目: 中国科学院战略性先导 A 类专项(XDA20080200)

Foundation item: Supported by the Strategy Priority Research Program of Chinese Academy of Sciences (XDA20080200)

收稿时间: 2019-09-30; 修改时间: 2020-02-13; 2020-04-26; 采用时间: 2020-09-09; jos 在线出版时间: 2020-10-12

定了一组每个节点必须遵守的规则，最终保证分布式系统中各节点区块链数据备份的一致性。

区块链共识协议分为两个主要步骤：出块节点选举和主链共识。在出块节点选举阶段，某个节点(或多个节点)成为出块节点，提出新区块。由于分布式网络中可能存在的恶意节点及分叉块的影响，其他节点在收到新区块以后不能直接将其加入自己的本地区块链中。所有节点需要利用主链共识对新区块及其构成的主链达成一致。出块节点选举机制和主链共识共同保证了区块链数据的正确性和一致性，从而为分布式环境中的不可信主体间建立信任关系提供技术支撑。

作为区块链系统的核心组件，共识协议同时决定了区块链系统的性能及安全性。例如，流行度最高的比特币系统[9]采用概率性共识协议。为了保证区块在大规模分布式系统中的广泛传播，比特币系统将区块大小限制为 1MB，区块间隔设置为 10 分钟，从而导致比特币系统每秒只能处理约 7 笔交易[1]。当交易量上升时，系统堵塞现象时有发生。此外，对于安全性要求较高的大数额交易，比特币官方客户端 Bitcoin Core 推荐用户等待随后六个区块的确认[10] (约一个小时)，使其具有更高概率安全性。为了改善比特币系统的性能，研究开发人员不断对其共识协议进行优化，比如缩短区块间隔、区块扩容等，然而这些优化同时可能带来对安全性[11-13]的负面影响。自从区块链技术诞生以来，大量研究工作围绕设计与分析共识协议开展，如 Bitcoin-NG[11]、Algorand[14]、Byzcoin[15]、Honeybadger[16]等。同时，也形成多个基于区块链的系统，如 Ethereum[17]、Litecoin[18]、Hyperledger Fabric[19]等。

随着区块链研究工作的推进，研究人员开始对区块链技术从不同方面进行梳理[20-23]。文献[20]讨论了比特币和以太坊工作量证明机制的异同。文献[21]讨论了基于证明的共识协议，包括工作量证明、权益证明等，并比较了工作量证明和权益证明的异同。文献[22]从激励机制的角度讨论了非许可链共识协议。文献[23]按时间顺序梳理了主要的区块链共识协议。上述研究工作一般将区块链共识协议视为整体讨论，缺乏对共识协议的步骤划分以及不同步骤间详细比较分析。据前所述，共识协议分为出块节点选举和主链共识两个主要步骤。这一分类视角在区块链协议设计方面十分常用[2, 14, 15, 24, 25]。基于对共识协议主要步骤的划分，有助于研究及开发人员理解共识协议的运行流程。进而，通过比较各步骤采用机制，清晰准确地理解不同机制的优劣。最后，不同步骤间的机制组合也可以为协议设计提供新的想法。

本文中，我们从区块链共识的两个主要步骤(出块节点选举和主链共识)对共识协议进行解耦分析。通过对每个步骤采用的机制进行综合性梳理、对比和分析，观察共识协议的发展趋势，为研究人员和开发者提供有用建议。具体而言，在出块节点选举机制部分，我们主要围绕目前已得到广泛研究与应用的工作量证明和权益证明展开讨论，总结工作量证明和权益证明机制存在的问题，从解决问题的角度讨论了随后出现的各种替代性证明机制。在主链共识部分，我们根据主链共识的性质，将主链共识分为概率性共识和确定性共识两类。在概率性共识中，我们讨论了各种主链选取规则，包括最长链规则[2]、GHOST[12]、包容性协议[26]等，并从概率性共识的持久性和活性角度对上述规则进行分析比较。在确定性共识中，我们首先讨论了经典的拜占庭协议[27]以及其应用到区块链中需要解决的问题，随后讨论了各种基于拜占庭容错协议达成一致的共识算法，包括 Algorand 的 BA*协议[14]、Byzcoin 的 PBFT 协议[15]、Stellar 的 SCP 协议[28]及 Honeybadger[16]，并从确定性共识的安全性和活性角度对各种协议进行分析比较。最后，我们总结了共识协议发展趋势以及可能的研究方向，为未来研究提供参考。

本文内容结构组织如下：第 1 章介绍了区块链共识协议的运行流程以及区块链系统的分类；第 2 章描述出块节点选举机制，分析比较了工作量证明机制和权益证明机制的现有工作；第 3 章描述了主链共识，并分析比较了各种概率性共识和确定性共识；第 4 章总结了研究发现及未来的展望。

1 区块链共识协议

区块链共识协议属于拜占庭容错协议，保证区块链网络中诚实节点在恶意节点干扰下也能达成共识。在分布式系统中，依据系统对故障组件的容错能力，共识协议分为崩溃容错协议(Crash Fault Tolerant，CFT)和拜占庭容错协议(Byzantine Fault Tolerant，BFT)两大类[20]。CFT 协议保证系统在组件宕机的情况下也能达成共识，适用于中心化的分布式数据集群，例如 Google 分布式锁服务 Chubby[29]、Paxos[30]协议等。BFT 协议由 Leslie Lamport 在 1982 年提出，保证分布式系统在故障组件[31, 32]的干扰下依然可以达成一致性。由于区块链网络的开放性质，区块链共识协议需要抵御恶意节点干扰，因此属于 BFT 协议。

[1]比特币区块大小约 1MB，区块间隔维持在 10 分钟左右，每笔交易大小约 250 字节，因此理论上每秒交易数 $=(1\times1024\times1024\div250)\div(10\times60)\approx7$

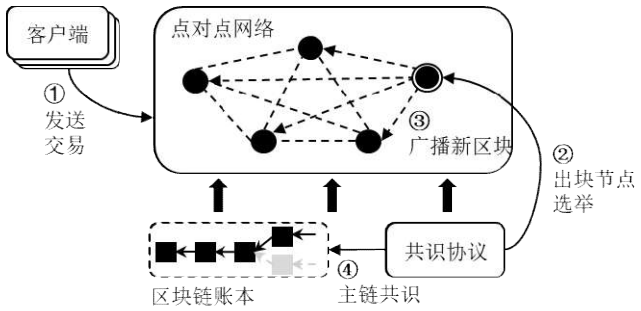

Fig. 1 Blockchain system execution flow diagram

图 1 区块链系统运行流程图

区块链系统的运行流程如图 1 所示。用户利用区块链客户端(钱包、网页等)发起交易，交易通过点对点(peer-to-peer，P2P)网络[2, 33]广播，节点验证交易格式和内容，若无误则加入本地交易池中并广播给其他节点；根据共识协议的出块节点选举规则，网络中的某节点成为出块节点；出块节点将交易打包、生成新区块，并通过点对点网络广播新区块；节点收到区块后，验证区块及交易的格式和内容正确性，若无误则更新本地区块链数据；节点若收到同一高度的多个区块，则根据共识协议的主链共识对区块链数据达成一致。通过以上分析可看出，区块链共识协议主要分为两个步骤：出块节点选举以及主链共识[2, 14, 15, 25]。前者用于选举分布式系统中的出块节点、生成新区块，后者则对区块链数据达成一致。

根据节点准入机制，区块链系统分为非许可链(permissionless blockchain)和许可链(permissioned blockchain)两类[34, 35]。非许可链系统中没有许可机构对节点进行身份审查，节点以匿名形式任意加入或退出系统，因此非许可链又被称为公开链(public blockchain)。基于这种开放性质，非许可链系统规模通常较大，共识节点甚至可达上万[36, 37]。许可链系统中节点需经过中心机构的准入审查，获得授权后才能加入系统。因而，许可链系统规模往往较小，节点数通常为几十到几百[16, 38]。针对不同应用场景，许可链又分为联盟链(consortium blockchain)和私有链(private blockchain)[34, 39]。联盟链通常由具有相同行业背景的多家不同机构组成，共识节点来自联盟内各个机构，区块链数据在联盟机构内部共享。私有链通常部署在单个机构内部，共识节点来自机构内部，类似于传统的分布式数据集群。由于区块链共识协议的相关研究主要针对非许可链，因此，我们在本文中主要分析非许可链共识协议，同时也包括一些典型的许可链共识协议。

我们将非许可链和许可链共识协议的不同特征总结如表 1 所示。针对准入机制，非许可链共识协议允许节点自由准入，许可链共识协议要求节点审查准入。基于此特性，非许可链一般应用于公开链的场景，而许可链应用于联盟链、私有链的场景。据上所述，非许可链共识协议一般具有较高的网络规模，许可链共识协议的网络规模相对较低。通常情况下，分布式系统网络规模越大，达成共识的难度越高，因而非许可链吞吐量通常较低，许可链较高。在一致性方面，非许可链共识协议通常以一定概率确保数据一致，实现弱一致性，许可链通常采用确定性方式确保数据一致，实现强一致性。

2 出块节点选举机制

区块链共识协议的出块节点选举机制与传统分布式协议中的领导人选举(leader election)问题[40]类似。该问题 1977 年由 Gérard Le Lann[40]正式提出，指分布式系统中采用某种机制选出一个领导人节点，该节点负责发起提案并发送给其他节点，其他节点基于提案更新数据，以此提升分布式系统运行效率。领导人选举思想应用于随后一系列的分布式系统共识协议研究[30, 41, 42]。在这些研究工作中，领导人节点的生命周期较长，通常持续到节点宕机，因此也被称为强领导人[42]。由于区块链系统的出块节点负责发起区块提案并发送给其他节点，以此完成区块链数据的更新，因此，出块节点选举机制类似于领导人选举问题。有所不同的是，区块链共识协议的出块节点选举机制需要抵御开放网络环境中的恶意节点。通过在 P2P 网络中伪造大量虚拟节点，恶意节点可以发起女巫攻击[43]，从而控制区块链系统。为了解决这一问题，区块链系统在出块节点选举环节通常采用“身份定价”机制，例如工作量证明[11, 15, 17, 18]、权益证明[24, 44, 45]等。下面分别对这两种机制进行综述分析。

Table 1 Comaprison between permissionless and permissioned blockchains

表 1 非许可链和许可链特征比较

特征	非许可链	许可链
应用场景	公开链	联盟链，私有链
准入机制	自由准入	审查准入
网络规模	高	低
吞吐量	低	高
一致性	概率(弱)一致性	确定(强)一致性

Table 2 Problems and corresponding solutions of PoW mechanism

表 2 工作量证明机制的问题及其解决方案

问题	解决方案	示例
算力中心化	内存密集型哈希函数； 外包困难难题； 智能合约矿池	莱特币[18]、狗狗币[47]、以太坊[17]、大零币[48]、小零币[49]； 文献[50]； SmartPool[51]
资源浪费	提供有用服务； 其他特定能力证明	素数币[52]、有用工作证明[53]； 权益证明机制{, #109}、空间证明[54]、权威证明[55]、信誉证明[56]
性能	缩短区块间隔； 微块	莱特币[18]、狗狗币[47]、以太坊[17]； Bitcoin-NG[11]、Byzcoin[15]

2.1 工作量证明机制

比特币[2]首次使用工作量证明(Proof of Work，PoW)机制进行出块节点选举，随后的大量区块链研究工作及系统都采用这一机制。工作量证明概念最早由 Markus Jakobsson 和 Ari Juels 在 1999 年提出[46]，用于实现可验证的计算任务。工作量证明包括证明者和验证者两个角色，证明者向验证者出示证据，表明自己在某时间段内完成了一定数量的计算任务。由于产生证据需消耗一定量计算资源，工作量证明可用于缓解垃圾邮件和其他拒绝服务攻击问题。

定义 1. 比特币工作量证明难题 给定全网统一的难度值 D，区块元数据 blockData，寻找满足条件的 Nonce(Number only used once)值，使得根据哈希函数 SHA-256 计算得到的区块哈希 blockHash 低于目标难度值 D。

$$\text{blockHash} = \text{Hash}(\text{blockData}, \text{Nonce}) \le \text{D}$$

比特币基于难题形式实现工作量证明机制。如定义 1 所示，比特币节点寻找满足条件的 Nonce 值，使区块哈希低于目标难度值 D。解决该工作量证明难题的节点将成为出块节点，负责发起区块提案。

由于哈希算法具备的输入敏感和抗碰撞[57, 58]特点，节点唯有不断调整输入值(Nonce、交易数据等)以寻找满足条件的 Nonce。因此，节点解决难题从而成为出块节点的概率与其可用的计算资源成正比。计算资源的投入可被视为一种身份定价机制[46, 59]，即便攻击者伪造大量虚拟身份，也无法提升计算资源，从而增加成为出块节点的优势。因此，工作量证明难题解决了分布式系统中的女巫攻击问题[43]。另一方面，由于哈希算法具备的正向快速和逆向困难[58]特点，验证节点可利用出块节点寻找的解快速验证正确性。因此，工作量证明难题实现了匿名分布式网络中的可公开验证。

作为首个区块链系统，比特币系统采用的工作量证明机制被应用到大量区块链共识协议研究[11, 14-16, 25, 26, 60, 61]及新的区块链系统[44, 45, 62]中。随着区块链研究工作的推进，研究人员逐渐发现比特币工作量证明机制的不足，并在此基础上进行改进。表 2 总结了比特币工作量证明机制(以下简称比特币 PoW)的不足及相应改进工作，下面分别对这些不足进行综述分析。

2.1.1 算力中心化

比特币的工作量证明机制(PoW)具有计算密集型特点，容易导致网络算力中心化。在比特币白皮书中，中本聪(Satoshi Nakamoto)提出“一处理器一票”(One-CPU-One-Vote)概念。在中本聪的设想中，节点使用个人电脑即可进行 PoW 运算，参与出块节点选举，并获得相应报酬。然而，随着比特币价格上涨，出块节点获得的区块奖励吸引了大量算力加入，比特币网络中的哈希算力呈指数级增长趋势[63]。共识节点参与 PoW 运算的物理设备从早期的个人电脑转换为 GPU，再演变为目前广泛使用的专用集成电路(Application-Specific Integrated Circuits，ASIC)矿机。

由于比特币 PoW 具备的计算资源可聚集和可外包特点，大多数节点选择加入矿池以保证收入的稳定性[1, 64]。伴随计算资源的聚集，比特币网络出现多个大型矿池。矿池由矿池管理员和矿工构成。如定义 2 所示，矿池管理员将计算子任务下发给矿工，子任务难度值 d 远低于全网统一难度值 D。矿工找到子任务难题解后，提交给矿池。由于部分子任务难题解也是定义 1 中比特币 PoW 难题解，矿池将获得区块奖励，并根据矿工根据提交的子任务解的数量分配报酬。矿池子任务难题的设计保证矿工收入的稳定性。截止 2019 年 7 月 9 日，占比第一的 BTC.com 矿池拥有 21.9%的算力，前两大矿池拥有 33.9%的算力。算力中心化会带来一系列的安全问题[64-66]，例如发动双花攻击、自私挖矿攻击等。

定义 2. 矿工工作量证明难题 给定难度值 d($d \ll D$)，区块元数据 blockData，寻找满足条件的 Nonce 值，使根据哈希函数 SHA-256 计算得到的区块哈希 blockHash 低于目标难度值 d。

$$\text{blockHash} = \text{Hash}(\text{blockData}, \text{nonce}) \le \text{d}$$

针对比特币 PoW 算力中心化问题，一些研究工作和区块链系统提出改进措施，包括替换 SHA-256 哈希函数、设计外包困难的 PoW 难题、去中心化矿池等。针对 SHA-256 哈希函数计算密集型特点，一些区块链系统选择用内存密集型哈希函数替代原有函数。例如，莱特币(Litecoin)[18]和狗狗币(Dogecoin)[47]采用 Scrypt 算法[67]、以太坊[17]采用 Ethash 算法[68]、大零币(ZeroCash)[48]和小零币(ZeroCoin)[49]采用 Equihash 算法[69]。内存密集型哈希函数由于计算时占用内存多、难以并行计算，能在一定程度上降低 ASIC 矿机的算力优势。针对比特币 PoW 难题可外包特点，研究人员修改难题形式使其外包困难，达到区块链系统去中心化的目标。例如，文献[50]重新设计比特币 PoW 难题，使矿池管理者将计算任务分发给矿工后，矿工可修改计算任务中获取奖励的地址，并不被矿池管理者发现。文件[51]实现了基于智能合约的去中心化矿池 SmartPool，矿池可自动执行子任务难题分发与

确认工作，替代矿池管理员，矿工在获得稳定收入的前提下，共同维护 SmartPool，从而保持算力的去中心化。

2.1.2 资源浪费

比特币工作量证明机制导致的算力资源浪费问题一直被广为诟病。从 2016 年开始，比特币网络的哈希率(哈希/秒)呈指数级增长。截至 2019 年 7 月，哈希率达到 70EH/s(百亿亿次哈希/秒)。现有文献[70]估计比特币网络年用电量与爱尔兰或奥地利年用电量相当。

为了解决资源浪费问题，现有研究工作和区块链系统主要提供了两种改进措施，即提供有用服务和其他特定能力证明。一些区块链系统利用 PoW 计算过程中消耗的算力提供有用服务。例如，素数币(Primecoin)[52]将 PoW 难题改进为寻找符合要求的素数，供公众使用，进而促进数学领域发展。素数币 PoW 难题的解包括三种形式的素数，即第一类坎宁安链(Cunningham chain of first kind)，第二类坎宁安链(Cunningham chain of second kind)和双链(bi-twin chain)。有用工作证明(Proof of Useful Work，PoUW)[53]提出基于广泛计算问题的 PoW 难题，解决正交向量、最短路径等问题。

除利用算力提供有用服务外，大量共识协议利用其他特定能力证明，如权益证明(Proof of Stake，PoS) [24, 44, 60, 71, 72]、空间证明(Proof of Space，PoSp)[54][1]、权威证明(Proof of Authority，PoAu)[55]、信誉证明(Proof of Reputation，PoR)[56]替代工作量证明。这些特定能力证明中，节点成为出块节点的概率分别与其拥有的某种稀缺资源相关，如权益(即加密货币数量)、内存或硬盘存储空间、权威、信誉相关，与算力无关。例如，文献[54]的空间证明用内存消耗型难题替代 PoW 算力难题。PoAu 与 PoR 思想类似，只有具有较高权威或信誉度的节点才能成为出块节点，由于区块带有节点签名，节点被检测到作恶后会丧失出块资格。因此，PoAu 与 PoR 只能用于具有准入机制的许可链系统中，无法用于非许可链系统。在这几类特定能力证明中，权益证明受到广泛研究与实际应用，因此，我们将在第 2.2 节详细讨论权益证明机制。

2.1.3 性能

比特币工作量证明机制是算力竞争型的出块节点选举机制，限制了出块环节的性能提升。如前所述，由于比特币系统平均区块间隔为 10 分钟，区块大小限制为 1MB，因此理论上交易吞吐量约每秒 7 笔交易。低吞吐量限制了比特币系统的广泛应用。随着比特币系统关注度上升，网络中未确认交易数增多。截至 2019 年 7 月 10 日，比特币网络存在近五万笔未确认交易[73]。性能问题成为比特币 PoW 中亟待解决的问题。

针对比特币 PoW 的低性能问题，一些研究工作和区块链系统通过修改参数和改进出块节点选举机制提升效率。例如，以太坊、莱特币、狗狗币系统分别将比特币 PoW 机制中的区块间隔调整为 2.5 分钟、1 分钟和 15 秒[74]，以此加速交易处理速度。缩短区块间隔看起来是改善性能的可行方案。然而，一些研究工作发现缩短区块间隔存在安全隐患。文献[11-13]指出，足够长的区块间隔保证区块数据在 P2P 网络中广泛传播，区块间隔缩短会削弱系统安全性。例如，当攻击者掌握 30%系统算力时，为了达到和比特币系统同等程度安全性，以太坊、莱特币、狗狗币系统需要分别等待至少 37、28、47 个区块长度确认[74]。

除以上工作外，Bitcoin-NG[11]通过修改比特币的出块节点选举机制提升交易性能。Bitcoin-NG 将区块分为关键块(key block)和微块(micro block)两类。关键块包含比特币 PoW 难题的解，体现出块节点的工作量证明。微块包含关键块对应的出块节点签名，但不包含难题的解，不体现工作量证明。节点生成关键块后，负责在随后的区块间隔时间内将交易打包进微块并签名。通过验证节点签名，其他节点判断微块合法性。通过在区块间隔连续产生微块，Bitcoin-NG 实现在出块节点选举环节加速交易处理。关键块和微块的概念随后也在 Byzcoin[15]中得到应用。

2.1.4 总结

工作量证明机制广泛应用于区块链共识协议的出块节点选举环节。针对算力中心化、资源浪费、低性能等问题，一些研究工作和区块链系统针对工作量证明机制进行改进。在工作量证明机制的调研过程中，我们发现一些可行研究点。首先，工作量证明改进工作多以白皮书形式提出，缺乏理论及实验数据支撑。例如，以太坊系统等白皮书大多从概念层面进行论述，没有相关实验数据支撑，也没有进行安全性证明。其次，众多的工作量证明改进机制间缺乏相互比较，无法判读优劣势。例如，Scrypt[67]、Ethash[68]、Equihash[69]在内的众多内存密集型哈希函数没有相互比较，尚不清楚这些算法针对算力中心化问题的改进程度。值得注意的是，目前已出现针对以上内存密集型哈希函数的专用矿机。再者，由于分布式系统的各方面复杂因素，对协议进行参数调整需要加以严格的安全证明。例如，莱特币等调整参数的改进方案看似可行，但被证明存在安全隐患[13]，最终没能达到预期效果。

---

[1] 空间证明(Proof of Space，PoSp)又被称为容量证明(Proof of Capacity，PoC)、存储证明(Proof of Storage，PoSt)

Table 3 PoS mechanisms based on random algorithms

表 3 基于随机算法的权益证明机制工作

工作	随机算法	随机种子
活动证明(PoA)[60]	follow-the-satoshi	PoW 空区块哈希
活动链(CoA)[71]	follow-the-satoshi	前 N 个区块哈希组合
Ouroboros[24]	follow-the-satoshi	安全多方计算更新
Algorand[14]	可验证随机函数	可验证随机函数更新

2.2 权益证明机制

针对工作量证明机制的资源浪费问题，比特币社区[75]在 2011 年首次提出权益证明机制(Proof of Stake，PoS)，根据节点掌握的比特币数量而不是算力作为权重选举出块节点。权益证明机制的安全性基于权益拥有者比矿工更有动力维护网络安全的假设[44, 71, 75]，当区块链系统遭到攻击，权益拥有者自身利益更容易受损。2012 年，权益证明机制首次在点点币(peercoin/ppcoin)[44]系统中得到应用。点点币以权益作为选举权重，提出权益证明难题。

定义 3. 点点币权益证明难题 给定全网统一的难度值 D，区块元数据 blockData，寻找满足条件的时间戳 timeStamp，使根据哈希函数 SHA-256 计算得到的区块哈希 blockHash 低于目标难度值。目标难度值为全网统一难度值 D 和币龄 coinDay 的乘积。币龄(coinDay)是节点持有权益(即节点持有的数字货币数量，coin)与持有时间(day)的乘积。

$$\text{blockHash} = \text{Hash}(\text{blockData}, \text{timeStamp}) \le \text{D} * \text{coinDay}$$

与比特币工作量证明难题相比，点点币权益证明难题主要有两处不同：哈希运算中移除随机数 Nonce，引入币龄调整难题难度。由于移除随机数 Nonce，点点币权益证明难题减轻了工作量证明难题算力竞争问题。在给定元数据 blockData 情况下，共识节点在求解点点币权益证明难题中可尝试的只有时间戳变量。由于点点币采用以秒计数的 UNIX 时间戳，节点求解难题时尝试空间有限。因此，点点币权益证明难题大大缩小了工作量证明难题的计算尝试空间，减缓了算力竞赛带来的资源浪费问题。

点点币以币龄作为权重，实现根据权益选举出块节点的目标，币龄的概念随后在披风币(Cloakcoin)[76]和新星币(Novacoin)[77]中也得到应用。币龄是用户权益和持有时间的乘积。假设用户 A 拥有 10 个点点币并持有 90 天，累计 900 币龄。用户 B 拥有 10 个点点币并持有 45 天，累计 450 币龄。根据点点币权益证明难题，用户 A 解决难题的可能性两倍于用户 B。

点点币权益证明难题创新性地使用币龄概念衡量权益，使得持有较多数字货币并活跃的节点更积极参与系统运行，但也存在不足。不活跃节点可能通过长期持有权益累积大量币龄，提高自己成为出块节点的可能性，从而等待发动攻击的时机。针对这一问题，未来币[45]和黑币[78]在权益证明难题中以权益(coin)替代币龄(coinDay)，维理币[79]使用类似币龄的权益时间(stakeTime)概念，节点离线后权益时间会逐渐减少，活动证明(Proof of Activity，PoA)[60]将权益证明与工作量证明结合，使得只有在线的活跃节点才能获得挖矿收益和交易费，这几种方法都用于改进点点币系统中不活跃节点问题。

2.2.1 基于随机函数的权益证明

以上所述的权益证明机制在一定程度上缓和了工作量证明机制的算力浪费问题，但采用的仍是基于难题求解的竞争性选举机制，为了进一步解决算力浪费问题、提高出块节点选举效率，随后的大量研究工作[14, 24, 71]采用基于随机函数的权益证明机制。这类机制采用以权益作为权重的随机算法确定出块节点，同时其他节点可通过随机算法验证出块节点身份的正确性。由于不再利用算力竞争成为出块节点，基于随机函数的权益证明属于非竞争性选举机制。

活动证明[60]，活动链(Chains of Activity，CoA)[71]和 Ouroboros[24]利用 follow-the-satoshi 算法[60]随机选举出块节点。聪(satoshi)是比特币的最小货币单位，follow-the-satoshi 算法将零和比特币发行总量(以聪为单位)间的一个随机数作为输入，通过追溯区块数据，找到目前持有该货币的节点，该节点即成为出块节点。假设用户 A 持有 10 个比特币，用户 B 持有 5 个比特币，用户 A 被 follow-the-satoshi 算法选中的概率两倍于用户 B。因此，follow-the-satoshi 算法实现根据权益进行出块节点选举。

表 3 对基于随机函数实现权益证明的研究工作及系统进行总结。PoA[60]，CoA[71]和 Ouroboros[24]采用不同方式更新 follow-the-satoshi 算法的随机种子。PoA[60]中，出块节点首先需要生成满足 PoW 的空区块[1]哈希，将该哈希作为随机算法输入，选出一组背书节点。出块节点搜集一定数量背书节点签名后才能打包交易、生成合法区块。因此，PoA[60]的出块节点选举机制实质是 PoW 和 PoS 的结合，PoW 区块哈希的不可预测保证了 PoS 选举结果的不可预测。CoA[71]将当前区块的前 N 个区块哈希作为随机算法输入，选出后 N 个区块的出块节点。Ouroboros[24]基于安全多方计算(Multi-Party Computation，MPC)更新随机种子。Ouroboros 将多个区块间隔称为一个纪元(epoch)，纪元内的出块节点共同组成组委员(committee)，组委会节点参与 MPC 算法，合作更新随机种子。Algorand[14]基于可验证随机函数(Verifiable Random Function，VRF)进行出块节点选举，各节点利用自己的私钥和全网统一的随机种子作为算法输入，判断自己是否是本轮的出块节点。若成为出块节点，节点将同时出示算法生成的选举证明，供其他节点验证。前一区块间隔的出块节点利用 VRF 函数更新下一间隔的随机种子。除了以上基于随机函数的出

[1] 不包含交易，只包含区块元数据的空区块。

Table 4 Problems and corresponding solutions of PoS mechanism

表 4 权益证明机制的问题及其解决方案

问题	描述	解决方案
粉碎攻击	节点尝试随机参数， 提高被选中为出块节点概率	参数不可尝试； 随机种子不直接依赖区块信息
无权益攻击	节点在多个分叉块后生成区块	保证金制度；安全硬件
长程攻击	节点从某一高度区块分叉	检查点机制；密钥演进加密技术

块节点选举机制外，Tendermint[80]采用确定性轮询算法决定出块节点，每个区块间隔将确定性产生一个出块节点。

2.2.2 问题及解决方案

在出块节点选举环节，PoS 以权益替代算力作为选举权重，解决了算力浪费问题。随着对权益证明工作的推进，研究人员发现 PoS 也存在一些问题，主要包括粉碎攻击、无权益攻击和长程攻击。表 4 对权益证明机制的问题及其解决方案进行总结。

粉碎攻击(grinding attack)指节点通过尝试随机参数提高成为出块节点概率的一类攻击形式。例如，攻击者通过尝试点点币权益证明难题中的时间戳(timeInStamp)和拥有的多个账户币龄(coinDay)提高成为出块节点概率{, #109}。与点点币类似，未来币中也有时间戳尝试和公钥尝试问题。除此以外，基于随机函数的权益证明可提前尝试随机种子，提高随后被选中为出块节点的概率[24, 60]{, #109}。由于随机种子基于以前的区块哈希，CoA[71]可能会遭到粉碎攻击[24]。

粉碎攻击的解决方案主要包括参数不可尝试以及随机种子限制。前者指权益证明难题中不包括可尝试参数。后者指随机种子尽量不依赖区块本身信息，否则存在随机算法偏向某一节点的可能。例如，Ouroboros[24]和 Ouroboros Praos[81]利用多方安全计算产生随机算法种子，既解决了可尝试参数问题，同时保证随机性与区块信息无关。

无权益攻击(nothing-at-stake)指被选中的出块节点在同一高度产生多个区块，导致区块分叉无法解决的问题。无权益攻击是出块节点出于利益最大化作出的选择。如图 2 所示，基于权益证明的区块链系统在高度 h 处出现分叉，节点 A 是高度 h+1 的出块节点，由于尚不确定在高度 h 上被最终确认的区块，节点 A 选择同时在多个分叉块后产生区块，因此，无论最终哪一区块是合法区块，节点 A 都可获利。

无权益攻击的攻击成本低，出块节点无需任何代价即可生成多个区块，最终导致多个分叉区块齐头并进，永远无法达成共识。无权益攻击的现有解决方案主要包括保证金制度和安全硬件。保证金制度指共识节点需在账户中存取一定金额保证金，若节点被监测发动无权益攻击，则罚没其保证金，从经济激励角度解决无权益攻击。例如，以太坊的 PoS 提案 Slasher[82]和 Casper[83]都引入保证金制度。类似地，Tendermint[84]也引入了保证金机制，且保证金比例和共识票数成正比。除此以外，文献[85]基于可信执行环境(Trusted Execution Environments，TEEs)的安全硬件限制节点在同一高度只能生成一个区块。

长程攻击(long range)指攻击者试图从某一高度区块后重新生成后续所有区块，覆盖这一区间区块数据，也被称为历史攻击(history attack)。由于长程攻击要求攻击者的区块生成速度快于其他节点，因此理论上只有掌握超过 50%权益的攻击者才能发动长程攻击，然而实际上，攻击者可通过控制或贿赂历史时刻拥有大量权益的节点[85]发动长程攻击。例如，攻击者 A 拥有 21%的权益，节点 B 在某一历史时刻拥有 30%的权益，随后将 30%权益转让他人，攻击者可通过控制或贿赂的手段，利用节点 B 在历史时刻拥有的权益，从该时刻重新生成区块。

由于节点转移权益后不再有维护系统安全的动机，攻击者可通过贿赂历史时刻拥有大量权益的节点发动长程攻击，针对这一问题的解决方案主要包括移动检查点机制和密钥演进加密技术。移动检查点机制[86]将某一历史高度的区块作为检查点，检查点前的区块不可篡改。移动检查点机制在点点币[44]、未来币[45]、黑币[78]、snow white[87]中得到使用。点点币和黑币依赖中心服务器定期发布检查点。未来币[45]不接受 720 个区块以前的历史区块分叉。与未来币类似，snow white[87]不接受对距离当前区块太远的历史区块分叉。除了检查点机制以外，Ouroboros Praos[81]采用密钥演进加密技术解决长程攻击[86]。节点随着时间需要不

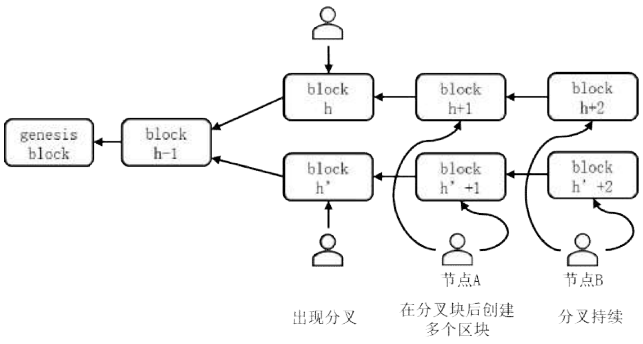

Fig. 2 The illustration of nothing-at-stake attack

图 2 无权益攻击图示

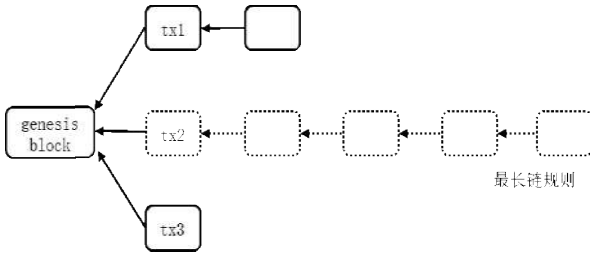

Fig. 3 The illustration of the longest chain rule

图 3 最长链规则图示

断演变私钥，当攻击者盗取了节点目前的密钥时，由于不能伪造过去的签名，无法利用节点的历史权益发送长程攻击。

除以上安全问题外，委托权益证明(Delegated Proof of Stake，DPoS)[88]通过投票机制缩小共识节点范围，使 PoS 在大规模网络中得以高效应用。DPoS 中的票数与权益成正比，权益所有者投票选出一部分节点作为候选出块节点，这些节点再利用 PoS 的随机算法成为出块节点。节点若在给定时间段内未完成出块，将被移出候选出块节点列表。因此，持有权益较少的节点可通过投票维护系统安全，而不必购买专业硬件设备成为共识节点。投票机制还可用于权益所有者修改系统参数，包括交易大小、区块间隔、交易费规则等，实现了区块链系统自治。

2.2.3 总结

权益证明机制由早期基于难题的竞争性机制，逐渐演变为基于随机函数的非竞争性出块节点选举机制，后者由于安全且高效，是目前权益证明共识协议的重点研究方向。我们在调研过程中发现，尽管研究成果众多，目前尚没有权益证明机制及其安全问题的综述研究。因此，本文对权益证明及其三种主要攻击方法和对应解决方案进行总结。此外，相比工作量证明机制，采用权益证明机制的区块链系统仍较少。再者，一些区块链系统采用工作量证明和权益证明结合的方式，前期利用工作量证明完成权益的初始分配，再逐渐过渡到权益证明机制，例如以太坊、点点币等。但是，目前尚没有关于工作量证明如何安全过渡到权益证明的相关研究。

3 主链共识

主链共识指分布式网络节点对区块数据达成一致的过程。如前所述，在区块链系统运行流程中，分布式网络中的节点根据出块节点选举机制成为出块节点、生成新区块并广播。由于出块节点选举机制存在同一高度产生多个区块的可能，节点本地维护区块形成的树状结构，即区块树(block tree)[12]，因此节点需要利用主链共识对区块树中最终的区块链数据达成一致。

根据区块数据是否满足最终一致性，主链共识可分为概率性共识和确定性共识。概率性共识中区块数据以一定概率达成一致，随着时间推移概率逐渐提高，不能保证区块数据将来不可更改，这种一致性也称为弱一致性。确定性共识中一旦区块数据达成一致便不可更改，又被称为强一致性。

3.1 概率性共识

非许可链系统广泛使用达成概率性共识的主链选取规则。由于非许可链网络规模较大[36, 37]，消息传输时间长、传输代价高，因此通常采用一轮广播即可达成共识的主链选取规则。在这类规则中，出块节点将生成的新区块广播给其他节点，节点使用主链选取规则从本地区块树中确定主链区块，各节点的主链区块随着时间推移接近一致。

3.1.1 最长链规则

最长链规则在比特币白皮书[2]中首次提出，选取区块树中的最长分支作为主链。由于比特币采用工作量证明机制，最长链累积着最多的工作量证明，根据比特币白皮书中“一处理器一票”的思想，最长链可被视为分布式网络中大部分节点投票做出的决定。因此，只要大部分算力由诚实节点掌握，分布式网络就可利用最长链规则对区块数据达成一致。最长链规则是目前应用最广泛的主链选取规则，在一系列研究工作[11, 24, 60, 71]和区块链系统[17, 18]中得到使用。

假设节点的本地区块树如图 3 所示，区块树中最长的分支(虚线分支)将成为主链。交易确认数指交易所在区块的长度，交易未被打包时称为 0 次确认(0-confirmation)，交易被区块包含称为 1 次确认(1-confirmation)，随着区块被后续区块不断链接，确认数不断增加。交易确认数越多，一致性概率越高、安全性越高。在图 3 中，交易 3(tx3)为 1 次确认，交易 1(tx1)为 2 次确认，交易 2(tx2)则是 5 次确认。因此，交易 2 安全性高于交易 1，交易 1 高于交易 3。比特币白皮书[2]中分析表明，假设攻击者拥有 10%系统算力，6 次确认交易的安全性高于 99.9%。为兼顾系统安全性与系统效率，，比特币客户端根据交易金额大小为交易推荐不同确认数[89]，金额较大交易确认数越多，保证交易安全性。

最长链规则性能受到协议参数和网络环境的影响，如区块间隔、区块大小、交易大小、网络规模、带宽等。如表 5 所示，

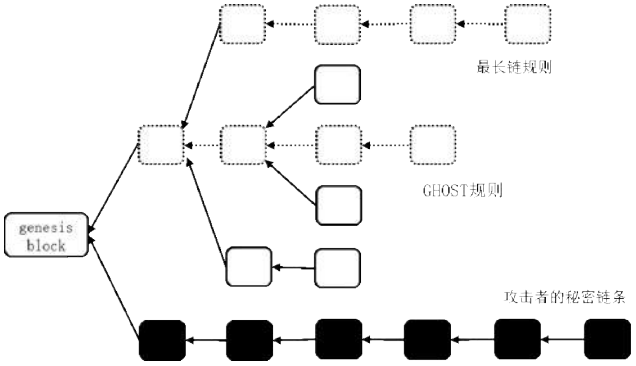

Fig. 4 The illustration of GHOST rule

图 4 GHOST 规则图示

在比特币系统中，当区块大小为 1MB，区块间隔为 10 分钟，交易大小为 250 字节时，交易吞吐量(Transaction Per Second，TPS)约等于 7，比特币系统从 2018 年 1 月至今的平均交易确认时延大于 10 分钟[90]。

3.1.2 GHOST规则

文献[12]发现在交易请求增多时，比特币不得不频繁创建大区块以提高交易吞吐量。大区块将导致区块传输时间延长，使得分叉块增多、诚实节点算力分散，此时恶意节点将更容易发动攻击。为了在交易请求增多时依然保证较高安全性，Yonatan Sompolinsky 和 Aviv Zohar 提出 GHOST(the Greedy Heaviest-Observed Sub-Tree)[12]规则作为最长链规则的替代。

图 4 展示了一种网络中出现多个分叉块的情形，当诚实节点的算力被分散时，最长链规则的安全性降低。例如，攻击者秘密生成一条私有区块链(黑色链条)，当私有区块链长度超过网络中公开的最长链时将其发布，根据最长链规则，攻击者的私有区块链将成为主链。攻击者可通过私有区块链发动双花攻击，从而破坏区块链系统安全性。在最长链规则中，主链外区块都被视为分叉块抛弃，不用于维护系统安全性。与最长链不同，GHOST 规则将分叉块纳入主链选取规则，区块树中最重子树的区块将构成主链，又被称为最重链。由于最重链代表网络中的大部分算力，文献[12]认为，只要诚实节点掌握大多数算力，GHOST 规则在网络交易吞吐量高的情况下也能保证安全性。如图 4 所示，尽管攻击者生成更长的私有区块链，由于没有累积足够多的工作量证明，无法替代 GHOST 规则选出的最终链。GHOST 规则随后在包容性协议[26]和 Conflux 中用来选择主链，但据我们所知，该规则目前并未直接应用于非许可链系统中[1]。

如表 5 所示，当区块间隔为 1 秒，区块大小为 1MB，网络规模为 1000 节点时，GHOST 协议的吞吐量约为 200，文献[12]中未给出交易确认时延的实验数据。值得注意的是，文献[12]在相同实验条件下对比了 GHOST 与最长链规则的性能，实验结果显示在不同参数设置中，GHOST 的吞吐量都略低于最长链规则，但其在处理高吞吐量交易时拥有更高安全性能。GHOST 的安全性将在 3.1.6 节中详细讨论。

3.1.3 包容性协议

包容性协议[26]将 GHOST 规则与有向无环图(Directed acyclic graph，DAG)结合，进一步提高交易吞吐量。包容性协议修改了以比特币为代表的传统区块链数据结构，区块可以指向多个父区块而不是唯一一个，新区块将所有没有被指向的区块(即叶子区块)作为父区块，因此，包容性协议中区块构成了有向无环图而不是区块树。基于该有向无环图，包容性协议首先利用 GHOST 规则选出主链，遍历主链区块的多个分叉父区块，如果分叉块中的交易和主链交易没有冲突，则将分叉块也纳入主链中。通过利用分叉块交易内容，包容性协议进一步提升交易通量，并且对于网络连接差、不能及时广播区块的节点更加友好。

如表 5 所示，当区块间隔为 1 秒，区块包含 50 笔交易，网络规模为 100 节点时，包容性协议的吞吐量约为 30，文献[26]中未给出交易确认时延的实验数据。

Table 5 Performance analysis of probabilistic consensus protocols

表 5 概率性共识协议的性能分析

规则	参数设置	吞吐量(TPS)	交易确认时延
最长链规则	区块间隔 10min，区块大小 1MB，交易大小 250Byte，网络规模 1 万节点	≈7	＞10min
GHOST	区块间隔 1s，区块大小 1MB，网络规模 1000 节点	≈200	无
包容性协议	区块间隔 1s，区块大小为 50 笔交易，网络规模 100 节点	30	无
SPECTRE	区块间隔 0.1s，区块大小 100KB	无	≈11s
Conflux	区块间隔 5s，区块大小 4MB，交易大小 250Byte，网络规模 1 万节点	3200	10min

[1] 我们注意到以太坊声称使用 GHOST 规则来选取主链，并实现了一个 GHOST 的简化版本[12]，但项目代码[91]显示目前采用的仍是最长链规则。

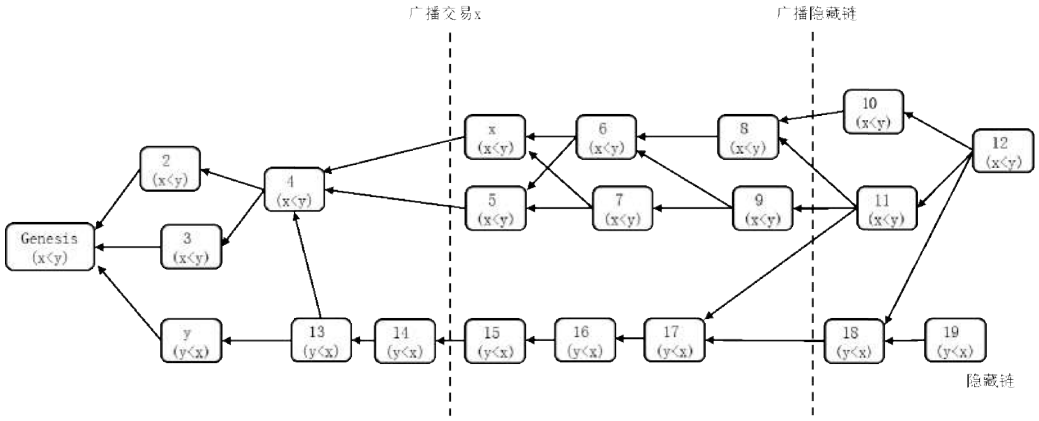

Fig. 5 The illustration of SPECTRE protocol

图 5 SPECTRE 协议图示

3.1.4 SPECTRE协议

SPECTRE[61]协议利用成对投票(pair voting)解决冲突区块问题，提高系统吞吐量并保证安全性。与包容性协议类似，SPECTRE 中新区块指向多个父区块，形成有向无环图，在此基础上运行投票规则。

假设区块 x 和 y 是一对包容冲突交易的区块，z 区块将对冲突区块进行投票，投票规则如下：

1) z 是 x 的后代区块，不是 y 的后代区块，z 投票给 x，表示为$x \leq y$。

2) z 同时是 x 和 y 的后代区块，则根据本区块之前有向无环图的区块投票结果确定。如果投票数量相等，则根据预定义规则决定(区块哈希的字典顺序等)。

3) z 既不是 x 的后代区块，也不是 y 的后代区块，则根据本区块之后有向无环图的区块投票结果确定。如果投票数量相等，则根据预定义规则决定。

图 5 展示了 SPECTRE 协议的运行规则。交易 x 和 y 是攻击者发起的两笔冲突交易，交易 x 是攻击者支付给商家的正常交易，交易 y 是恶意交易。将交易 x 广播之前，攻击者预先生成包含交易 y 的区块及之后两个区块(区块 13、14)并隐藏。为了欺骗商家获得收益，攻击者首先广播交易 x，并等待交易 x 被包含到区块中。商家确认交易 x 后，攻击者将隐藏区块链发布出来，企图替代交易 x 所在的区块链。

在 SPECTRE 协议中，节点可并行产生区块，随后再用投票规则处理冲突区块，整体上提升交易处理速率，同时投票规则保证攻击者的隐藏区块链无法颠覆主链。由图 5 可知，除攻击者产生的所有区块都认可交易 x。区块 6~10 是区块 x 但不是区块 y 的后代，根据规则 1 投票给 x。区块 1~5 不是 x 和 y 的后代，根据规则 3 投票给 x。区块 11~12 是 x 和 y 的后代，根据规则 2 投票给 x。SPECTRE 基于工作量证明机制选举出块节点，文献[61]认为，只要网络中诚实节点掌握大多数算力，投票机制可不断提升历史交易优势，从而消除来自恶意节点隐藏区块链的威胁[92]，保证系统安全性。

如表 5 所示，当区块间隔为 0.1 秒，区块大小为 100KB 时，SPECTRE 协议的交易确认时延约为 11 秒，论文[61]中未给出交易吞吐量的实验数据。

3.1.5 Conflux协议

Conflux[25]协议基于有向无环图计算出区块内交易的全局顺序，通过剔除冲突交易，使所有分叉块内的交易得到利用，从而提升系统吞吐量。Conflux 的区块间有两类指向关系，父边(parent edge)和引用边(reference edge)。父边指向父区块，每个区块只有一条父边。除父边外，区块可以有多条引用边，引用边指向所有目前没有被父边引用的叶子区块，引用边代表时间上的发生在先(happen-before)关系。Conflux 利用 GHOST 规则选出有向无环图中的主链，利用主链和两类指向关系对所有区块进行全局排序。基于区块的全局排序，区块内的交易也达成全局排序。随后，Conflux 从交易全局排序中剔除掉重复和冲突的交易，对余下交易达成共识。

与包容性协议和 SPECTRE 协议相比，Conflux 将共识粒度从区块细化到交易，利用交易排序算法将所有分叉块里的交易得以利用，因而提升了交易吞吐量。文献[25]将 Conflux 协议与工作量证明的出块节点选举机制结合，同时 Conflux 也可与其他出块节点选举机制结合，例如权益证明机制。

如表 5 所示，当区块间隔为 5 秒，区块大小 4MB，交易大小 250 字节，网络规模 1 万节点时，Conflux 的吞吐量为 3200，远高于相同实验参数设置下的最长链和 GHOST 规则，但由于使用 GHOST 规则选举主链，Conflux 的交易确认时延和 GHOST 一致，都是 10 分钟。

Table 6 Analysis of the security properties of probabilistic consensus protocols

表 6 概率性共识协议的安全性质分析

规则	持久性	活性	自私挖矿攻击
最长链规则	满足	满足	存在
GHOST	满足	满足但弱于最长链规则	存在
包容性协议	未证明	未证明	存在
SPECTRE	满足	弱活性	未讨论
Conflux	满足	满足但弱于最长链规则	存在

3.1.6 安全分析

由于大量共识协议[14, 15, 24]围绕持久性和活性进行安全性分析，本节主要从持久性和活性两方面对概率性共识进行安全性分析。文献[13, 93, 94]基于传统分布式协议的安全性和活性目标，提出并总结了区块链共识协议中持久性和活性目标。由于大部分概率性共识与工作量证明机制结合，因此概率型共识的安全性分析主要基于工作量证明的出块节点选举机制。

- **持久性(persistence)** 持久性衡量概率性共识中区块链数据的一致性。持久性指如果某区块在节点的本地区块链中拥有 k 个区块的深度，该区块在其他节点的本地区块链中(极大概率)也拥有 k 个区块的深度。由于网络传播等限制，各个节点的本地区块链可能暂时不一致，但 k 个区块之前的数据(极大概率)是一致的。
- **活性(liveness)** 活性衡量概率性共识中区块链系统的可用性。活性指诚实节点发起的交易最终被打包进节点区块链中，并满足持久性。由于网络吞吐量等限制，诚实节点发起的交易可能不会立即被处理，但最终会被处理。

诚实节点掌握大多数算力的情况下，最长链规则在同步和异步网络中都满足持久性和活性要求。比特币白皮书[2]对基于工作量证明的最长链规则首先进行了非正式分析，在网络同步且诚实节点掌握大多数算力时，协议满足持久性要求。随后，文献[13]对最长链规则进行理论证明，在网络同步且诚实节点拥有大多数算力的情况下，最长链规则满足持久性和活性要求。在此基础上，文献[95]证明最长链规则在延迟时间有上界的异步网络中，且诚实节点拥有大多数算力的情况下，满足持久性和活性要求，并可以适应算力动态变化。

在诚实节点掌握大多数算力情况下，GHOST 规则满足持久性和活性要求，但活性弱于最长链规则。GHOST 规则[12]作为最长链规则的替代，只要诚实节点掌握大部分算力，在网络交易吞吐量高的情况下依然可以保证安全性。文献[96]理论证明 GHOST 协议满足持久性和活性。然而，文献[96]认为在网络交易吞吐量高的情况下，GHOST 规则相对最长链规则并没有性能优势。除此以外，文献[96]提出一种针对区块链活性的攻击方法，通过干扰交易和区块的传播，拖延交易的确认时间，从而破坏活性。在这种攻击方式下，GHOST 的活性弱于最长链规则。由于 Conflux 利用 GHOST 协议选择主链，因此安全性质和 GHOST 相同[25]。

在诚实节点掌握大多数算力情况下，包容性协议满足持久性和活性。文献[26]未对包容性协议进行理论证明，但通过实验方式显示协议可满足持久性和活性。

在诚实节点掌握大多数算力情况下，SPECTRE 协议可保证区块链系统的持久性和弱活性[61][1]。SPECTRE 协议在交易内容互不冲突时，能保证交易在有限时间内被打包到区块中，即满足活性要求。然而，当攻击者同时发起两笔冲突交易时，合法交易只能满足弱活性，即交易最终会被打包到区块中，但时间有明显延迟。

在与工作量证明机制结合的情况下，大部分概率性共识协议存在激励相容(incentive compatibility)问题。激励相容问题指节点的个人目标与系统目标不一致，其中最典型的自私挖矿问题。在工作量证明机制的基础上，为鼓励节点加入网络维护安全，系统通常以数字货币形式给出块节点一定报酬。然而，大量研究工作[64, 65, 97, 98]发现当节点掌握算力达到一定程度时，理性节点选择不遵守协议将获得更高收益。这些节点被称为自私挖矿节点，他们将生成的新区块隐藏起来，并选择在将来的适当时机公开，从而获得更高收益，这些隐藏区块可能会颠覆网络的公开区块，引发安全问题。

文献[64]首次发现最长链规则中的自私挖矿问题。在自私挖矿影响下，最长链规则的安全边界从 50%降低到 33%，即只能抵御算力不超过 33%的恶意节点。随后，一系列研究工作[65, 97]在最长链规则上拓展不同的自私挖矿策略，进一步降低系统安全边界。与最长链规则类似，GHOST 规则和包容性协议也受到自私挖矿影响[26, 65]。SPECTRE 协议没有说明是否存在自私挖矿问题。此外，由于 Conflux 协议基于 GHOST 规则选择主链，也有同样的自私挖矿问题。

表 6 对以上讨论的概率性共识协议安全性质进行总结。除包容性协议以外，以上概率性共识都从理论上证明满足持久性和活性目标。目前仅有最长链和 GHOST 规则在持久性和活性上的对比分析工作，其他概率性共识之间缺乏对比。其次，GHOST 规则在研究工作中备受关注，但缺乏区块链系统应用，以太坊目前仍采用最长链规则。再者，几乎所有基于工作量证明的概率性共识都存在自私挖矿问题，目前自私挖矿问题在最长链规则上得到大量研究，但在其他概率性共识上的研究较少。

[1]SPECTRE 协议中未对持久性(persistence)进行证明，但其中的一致性、安全性与持久性同一定义，因此我们将其看做对持久性的证明。

Table 7 Performance analysis of deterministic consensus protocols

表 7 确定性共识协议的性能分析

规则	参数设置	吞吐量(TPS)	交易确认时延
Algorand(BA*)	区块间隔 1min，区块大小 1MB，网络规模 5 万节点	327MB/h	＜1s
Byzcoin(PBFT)	关键块间隔 10min，区块大小 32MB，网络规模 144 节点	974	68s
Stellar(SCP)	无		
HoneyBadger	交易大小 250Byte，网络规模 104 节点	1500	＜6min
Tendermint	区块大小为 1 万笔交易，交易大小 250Byte，网络规模 64 节点	≈4000	≈2s

3.2 确定性共识

概率性共识在交易延迟与安全性存在天然的权衡问题[99]，限制了区块链技术的应用场景，因此，一些研究工作采用确定性共识确保区块数据的强一致性。概率性共识的权衡问题源于区块数据的一致性概率随着时间推移逐渐提高，为保证交易安全性，用户不得不等待多个区块确认，带来明显的交易延时。延时问题限制了基于概率性共识的区块链系统的商业应用，因此，一些研究工作[14-16, 28]采用确定性共识替代概率性共识。如上所述，在确定性共识中，区块一旦写入节点本地区块链，就不存在随后被改变的可能性。确定性共识有两个明显优势[15]，首先，用户不用等待较长时间确保交易安全性。其次，由于同一高度仅有一个合法区块，节点不用在分叉区块上浪费计算资源。

拜占庭容错协议用于解决分布式系统中的拜占庭将军问题，在存在恶意节点的情况下达成一致性。拜占庭将军问题由 Leslie Lamport 在 1982 年[31]提出，诚实将军在叛徒将军的干扰下对进攻命令达成一致。拜占庭将军问题是分布式系统中正常组件在故障组件干扰下达成一致的抽象描述。

经典的拜占庭容错协议通常面向中心化的分布式集群达成确定一致性，如 PBFT[27]、Byzantine Paoxs[100]等，但无法直接应用在区块链系统中。在这些协议中，共识节点数量固定或者变化缓慢[27]，节点之间需要多轮广播通信[101]，通信复杂度较高，然而区块链系统中的节点数量不断动态变化，区块链系统(特别是非许可链)的网络规模也不支持节点间的多轮广播通信，因此区块链拜占庭容许协议需要适应系统特点进行改进。

3.2.1 非许可链拜占庭容错协议

为了在网络规模较大的非许可链系统中达成确定性共识，一些研究工作[14, 15]将拜占庭容错协议和区块链的出块节点选举机制相结合，被称为混合协议[102]。混合协议也分为出块节点选举和主链共识两个阶段。在出块节点选举阶段，混合协议采用区块链的选举机制，抵御开放网络中的女巫攻击等问题。在主链共识阶段，混合协议通常让多个出块节点构成组委会(committee)，运行拜占庭容错协议，对新区块达成一致。组委会成员通常随着时间变化[102]。

Algorand[14]是权益证明和拜占庭容错协议结合的混合协议。在出块节点选举阶段，Algorand 利用基于随机函数的权益证明选出一组出块节点，每个节点发起区块提案(block proposal)并广播，各提案附有随机优先级，每个节点只保留优先级最高的区块。随后，节点再运行一轮拜占庭一致性协议(BA*)，将自己接收到的最高优先级区块作为输入，对区块达成共识。

Algorand 中的拜占庭一致性协议分为两个阶段：归约(reduction)和二进制一致性(binary agreement)。规约阶段保证各节点持有相同的最高优先级区块，解决网络传输导致的节点本地最高优先级区块可能不一致的问题。在规约阶段，所有节点广播自己本地最高优先级的区块哈希，接收到其他节点的区块哈希后，节点统计每个区块的票数，认定票数最高的区块为最高优先级区块，没有票数最高区块时，将空区块作为最高优先级区块。归约阶段达成一致的区块将作为二进制一致性阶段的输入。

二进制一致性阶段对规约阶段生成的区块达成确定性共识。在二进制一致性阶段，出块节点选举环节发起区块提案的节点形成组委会，对规约阶段的区块投票。区块收到一定数量票数后，就被确认为最终区块。所有节点将该区块更新到本地区块链

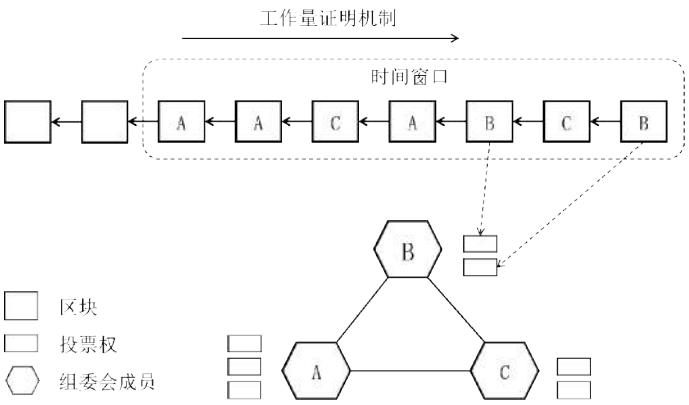

Fig. 6 Byzcoin hybrid consensus

图 6 Byzcoin 混合协议图示

中，达成确认性共识。由于网络原因，二进制一致性阶段的投票可能会重复多次。如表 7 所示，当区块间隔为 1 分钟，区块大小为 1MB，网络规模为 5 万节点时，Algorand 的交易吞吐量达到 327MB/h，交易确认时延小于 1 秒。

Byzcoin[15]是工作量证明和实用拜占庭容错协议结合的混合协议。Byzcoin 首先利用工作量证明机制选举出块节点、生成新区块，随后再利用实用拜占庭容错协议对新区块达成确定性共识。如图 6 所示，在出块节点选举阶段，节点利用工作量证明生成新区块并广播。一个时间间隔(一天或一周，可调整)内的出块节点构成组委会。组委会成员票数为在该时间间隔内的出块数量，成员利用实用拜占庭容错协议对新区块投票达成共识。出块节点广播新区块，组委会成员验证区块无误后返回签名作为投票，出块节点搜集至少 2/3 票数后，广播组委会成员签名，证明新区块已经被组委会接收并验证。组委会成员接收到广播信息后，再次返回签名，表示同意将该区块写入区块链中，出块节点搜集至少 2/3 票数后，再次广播区块，并写入区块链中。至此，共识节点对该区块达成确定性共识。由于通信过程中涉及大量签名和验签操作，为提高效率，Byzcoin 引入集体签名技术，可一次同时验证多个签名。如表 7 所示，当关键块间隔为 10 分钟，区块大小为 32MB，网络规模为 144 节点时，Byzcoin 的 TPS 为 974，时延为 68 秒。

不同于 Algorand 和 Byzcoin 采用混合协议，Stellar 共识协议[28]采用联邦拜占庭协议(Federated Byzantine Agreement，FBA)达成共识。Stellar 是一个开放的实时跨境支付系统[103]，为了使拜占庭协议支持非许可链中开放成员的需求，引入仲裁系统分片(quorum slice)达成共识。在拜占庭协议中，仲裁系统指可达成共识的一组节点。仲裁系统分片是仲裁系统子集。Stellar 的仲裁系统基于某种标准划分，例如声誉或权益，节点可同时加入多个仲裁系统分片。仲裁系统分片保持交集，保证共识达成。

Stellar 共识协议主要分为投票、接收和确认三个阶段。在投票阶段，节点对接收到的交易进行投票并广播投票信息。在接收阶段，若节点 v-blocking 集合中的所有节点都投票给该交易，则接收该交易。v-blocking 是和该节点所在的全部仲裁分片有交集的节点集合。在确认阶段，节点间通过消息交互对接收阶段的交易达成最终共识。仲裁分片互相影响，最终保证所有诚实节点对交易达成确定性共识。文献[28]中未给出 Stellar 协议的实验数据。

3.2.2 许可链拜占庭容错协议

如上所述，许可链根据应用场景不同分为联盟链和私有链，其中企业级联盟链是目前应用最广的许可链系统。由于网络规模限制、共识一致性要求高，许可链更适合采用拜占庭容错协议[38]。目前的一些研究工作[16]和系统[19, 80, 104]探索拜占庭容错协议在许可链中的应用。

HoneyBadger[16]首次将实践拜占庭容错协议应用到纯异步许可链中。HoneyBadger 系统中，共识节点身份已知且数量固定，节点两两建立经过认证的可信通道。为了消除出块节点广播区块这一环节的带宽瓶颈，HoneyBadger 没有采用出块节点选举机制，取而代之的是，各节点在每轮出块开始时从本地交易缓冲池中选择部分交易进行广播。为了避免拜占庭节点故意忽略某些交易从而影响系统活性，节点不广播交易内容本身而是经门限加密后的交易密文。所有节点收到密文集合后，HoneyBadger 通过拜占庭协议对一组位向量(bit vector)达成共识，假设位向量第 N 位为真，则将密文集合对应的第 N 位密文还原，并将其中包含的交易写入区块。如表 7 所示，当交易大小为 250 字节，网络规模为 104 节点时，HoneyBadger 的 TPS 为 1500，时延小于 6 分钟。

联盟链系统 Tendermint[80]采用基于轮询机制的实用拜占庭容错协议对新区块达成共识。在出块节点选举环节，Tendermint 采用确定性轮询机制决定出块节点。由于未采用类似工作量证明的身份定价机制，为防止拜占庭节点发动女巫攻击，系统规定节点必须在账户存入保证金才能参与拜占庭容错协议的投票过程，保证金数额与票数成正比。在网络弱同步且诚实节点掌握至少 2/3 票数的情况下，Tendermint 满足安全性和活性。如表 7 所示，当区块大小为 1 万笔交易，交易大小为 250 字节，网络规模为全球 64 节点时，Tendermint 的 TPS 约为 4000，延迟约等于两秒[80]。

除使用拜占庭容错协议外，一些企业级区块链系统采用 CFT 协议而非 BFT 协议达成确定性共识。2016 年初，Linux 基金会发起 Hyperledger 项目，旨在建立企业级区块链框架[105]，目前已有超过 270 个机构加入[106]。Hyperledger Fabric(以下简称 Fabric)是 Hyperledger 项目中备受关注的一个子项目，打造面向许可链的分布式数据平台。Fabric v0.5 采用了 PBFT 协议在共识节点之间对交易内容实现共识[105]，目前最新的 Fabric v1.4 用 Raft 和 Apache Kafka 两种 CFT 协议实现共识[107]。

3.2.3 安全分析

确定性共识需要满足安全性和活性两个基本性质对应概率性共识中的持久性和活性。此外，由于不存在分叉，确定性共识不存在概率性共识中的自私挖矿问题[14, 15]。

- **安全性** 安全性(safety) 衡量确定性共识中区块链数据的一致性[108]。如果某区块被写入节点的本地区块链中，该区块也被其他节点写入本地区块链中。即各节点在同一高度拥有相同区块。
- **活性** 活性衡量确定性共识中区块链系统的可用性。活性指由诚实节点发起的交易最终会被打包进区块链中、并且满足持久性。由于网络吞吐量等限制，诚实节点发起的交易可能不会立即被处理，但最终会被处理。

Table 8 Analysis of security properties of deterministic consensus protocols

表 8 确定性共识协议的安全性质分析

协议	场景	安全性	活性	网络假设	节点数假设
Algorand(BA*)	非许可链	满足	满足	弱同步	诚实节点掌握 2/3 权益
Byzcoin(PBFT)	非许可链	满足	满足	弱同步	节点总数 3f+2
Stellar(SCP)	非许可链	满足	满足*	同步	仲裁分片相交
HoneyBadger	许可链	满足	满足	异步	节点总数 3f+1
Tendermint	许可链	满足	满足	弱同步	诚实节点掌握 2/3 保证金

*恶意节点故意过滤交易会影响活性

在强同步网络及诚实节点掌握至少 2/3 权益的情况下，Algorand 满足安全性和活性。Algorand 拜占庭一致性协议(BA*)基于强同步网络，大多数(如 95%)诚实节点发送的消息在已知时间间隔内可发送给其他大多数诚实节点。BA*在弱同步网络情况下不能达成一致性，需要等待网络变为强同步。Algorand 采用权益证明的出块节点选举机制，要求诚实节点掌握系统中至少 2/3 的权益。最好情况下，即每个节点收到的最高优先级区块一致时，Algorand 需要节点间的四轮交互达成共识，最坏情况下则需要 13 轮。文献[14]利用 1,000 台虚拟机模拟 50,000 用户运行 Algorand 协议，实验显示确认 1MB 大小的区块需要约 22 秒延迟，区块大小为 10MB 时，Algorand 吞吐量为 750MB/h。

在弱同步网络及节点总数至少 3f+2 的情况下，Byzcoin 满足安全性和活性。Byzcoin 是实用拜占庭容错协议和工作量证明的混合协议，网络环境要求和实用拜占庭容错协议一致，即消息延迟有上限但上限不可知的弱同步环境。当网络存在 f 个恶意节点时，Byzcoin 需要至少 3f+2 节点数才能达成共识。文献[15]在 1008 个节点上运行 Byzcoin 协议，当区块大小为 1-2MB 时，交易延迟为 10-20 秒，交易吞吐量为 700 笔每秒。

在同步网络及节点选择足够多的仲裁分片的情况下，Stellar 满足安全性，但恶意节点故意过滤交易时，Stellar 活性受到影响。Stellar 分为投票、接收和确认三个阶段，投票和接收阶段要求同步网络，确认阶段可以是异步网络，总体上，Stellar 需运行在同步网络中。在节点选择足够多的仲裁分片，且仲裁分片中声誉或权益高的节点是诚实节点时，文献[28]证明 Stellar 满足安全性，但恶意节点在 Stellar 的交易投票阶段故意过滤交易，会影响交易活性。文献[28]未对 Stellar 协议的性能表现开展实验。

在纯异步网络及节点总数至少为 3f+1 时，HoneyBadger 满足安全性和活性。HoneyBadger 要求节点身份已知且数量固定，节点间需要两两建立可信通道，在纯异步通信环境下，HoneyBadger 保持活性。由于采用拜占庭容错协议，为抵御 f 个恶意节点，HoneyBadger 需要至少 3f+1 个节点保证安全性。HoneyBadger 可用于广域网中的联盟链系统，系统可扩展至上百节点数。在跨五个洲的 104 个节点上运行 HoneyBadger 协议时，交易吞吐量达到 1,500 笔每秒，延迟不到六分钟；在 32 个节点上运行时，交易吞吐量最高达到 20,000 笔每秒，延迟为 30 秒左右[16]。

3.2.4 总结

表 8 总结了以上所述确定性共识协议的安全性质。Algorand、Byzcoin 和 Stellar 协议面向非许可链场景，HoneyBadger、Tendermint 协议面向网络规模受限的许可链应用场景。Algorand 等协议基本都满足安全性和活性，但 Stellar 系统在存在恶意节点时系统活性会受到破坏。在网络层面上，Algorand 和 Byzcoin 要求弱同步网络，Stellar 要求同步网络，HoneyBadger 可在纯异步网络中运行。尽管对网络要求低，HoneyBadger 只能在节点身份已知且数量不变的非许可链环境中运行，且要求节点两两建立可信通道，对节点间通信要求较高。在安全边界上，大多数确定性共识协议继承了拜占庭协议 3f+1 的安全边界，Stellar 则要求仲裁分片相交。此外，由于各协议在实验设计方面缺乏统一标准，包括节点数目、节点硬件配置、区块和交易大小设置等，以致于文献中各协议的性能表现实验结果难以横向对比。

4 总结与展望

本文中，我们将区块链共识协议分为出块节点选举和主链共识两个主要步骤，通过对每个步骤采用机制进行综合梳理、对比和分析，发现如下关键问题，值得广大研究人员关注。

在出块节点选举机制，我们围绕工作量证明和权益证明展开讨论。区块链的出块节点选举机制类似传统分布式协议的领导人选举问题。与之不同的是，为抵御开放网络环境中的恶意节点，出块节点选举机制通常基于“身份定价”机制，包括工作量证明和权益证明。

- 工作量证明通过物理资源的投入抵御恶意节点，存在算力中心化、资源浪费、选举性能低等若干问题。为了解决这些问题，一些研究工作用内存密集型函数替代计算密集型函数、利用算力提供有用服务、调整工作量证明难题参数等。部分方案改变了系统性质从而引出了新的研究问题，包括改变区块大小、出块间隔、难度调整算法等，研究人员可针对已有攻击方式在改进系统中的新型攻击手段、安全边界展开研究。
- 权益证明被用于解决工作量证明的资源浪费问题，由早期的竞争性难题机制逐渐演变为基于随机函数的非竞争性机制。非竞争性机制由于安全和高效，是目前权益证明的重点研究方向。权益证明也存在粉碎攻击、无权益攻击和长程攻击等问题，但目前尚没有对权益证明及其安全问题的综述研究。

在主链共识步骤，根据区块数据的一致性性质，我们将主链共识分为概率性共识和确定性共识，并分别展开讨论。

- 概率性共识从早期基于区块树的选取规则逐渐演变为基于有向无环图的选取规则，且共识粒度从区块细化为交易，部分有向无环图协议对交易全局顺序达成概率性共识。当前大部分概率性共识通过理论证明满足安全性和活性。然而，概率性共识之间缺乏对比工作，目前仅有最长链和 GHOST 规则的安全性对比分析。在调研过程中我们发现，最长链规则仍是区块链系统的主导规则，GHOST 规则在研究工作中备受关注但目前缺乏实际应用。此外，几乎所有基于工作量证明的概率性共识都存在自私挖矿问题，但尚未有研究工作对除最长链规则以外的其他概率性共识度量自私挖矿攻击的影响。
- 为满足非许可链开放成员及网络规模要求，大多数确定性共识将出块节点选举机制与拜占庭容错协议相结合，实现区块链系统的确定一致性。这些协议通常运行于异步网络中，并要求恶意节点不超过节点总数的 1/3。据我们所知，目前尚没有可运行于大规模纯异步网络中的确定性共识协议。HoneyBadger 作为纯异步共识协议，只适用于节点身份和数量都已知的非许可链系统。

区块链具有去信任、开放自治、匿名可溯源、信息不可篡改等特性，是构建可信数字经济的重要基础设施。共识协议作为区块链核心技术，近年来得到广泛关注和大量研究。本文对现有区块链共识协议进行综合梳理，为研究人员和开发者提供有用参考。

References:

[1] Bonneau J, Miller A, Clark J, Narayanan A, Kroll J A, Felten E W. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. IEEE Symposium on Security and Privacy 2015: 104-121.[doi: 10.1109/SP.2015.14]

[2] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. White paper, 2008

[3] Merchants accept bitcoin payment. https://99bitcoins.com/who-accepts-bitcoins-payment-companies-stores-take-bitcoins/. Accessed on: 2019.8.9.

[4] IBM financial services. https://www.ibm.com/blockchain/industries/financial-services. Accessed on: 2019.8.9.

[5] Ant financial blockchain. https://tech.antfin.com/blockchain. Accessed on: 2019.8.9.

[6] IBM supply chain. https://www.ibm.com/blockchain/industries/supply-chain. Accessed on: 2019.8.9.

[7] Government blockchain association https://www.gbaglobal.org/. Accessed on: 2019.8.9.

[8] Ethereum dapp market. https://www.stateofthedapps.com/zh/rankings/platform/ethereum. Accessed on: 2019.8.9.

[9] Cryptocurrency market cap. https://coinmarketcap.com/. Accessed on: 2019.8.9.

[10] Bitcoin confirmation. https://en.bitcoin.it/wiki/Confirmation. Accessed on: 2019.8.9.

[11] Eyal I, Gencer A E, Sirer E G, Van Renesse R. Bitcoin-ng: A scalable blockchain protocol. 13th USENIX Symposium on Networked Systems Design and Implementation 2016: 45-59.

[12] Sompolinsky Y, Zohar A. Secure high-rate transaction processing in bitcoin. International Conference on Financial Cryptography and Data Security 2015: 507-527.[doi: 10.1007/978-3-662-47854-7_32]

[13] Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol: Analysis and applications. Annual International Conference on the Theory and Applications of Cryptographic Techniques 2015: 281-310.[doi: 10.1007/978-3-662-46803-6_10]

[14] Gilad Y, Hemo R, Micali S, Vlachos G, Zeldovich N. Algorand: Scaling byzantine agreements for cryptocurrencies. Proceedings of the 26th Symposium on Operating Systems Principles 2017: 51-68.[doi: 10.1145/3132747.3132757]

[15] Kogias E K, Jovanovic P, Gailly N, Khoffi I, Gasser L, Ford B. Enhancing bitcoin security and performance with strong consistency via collective signing. 25th USENIX Security Symposium 2016: 279-296.

[16] Miller A, Xia Y, Croman K, Shi E, Song D. The honey badger of BFT protocols. ACM SIGSAC Conference on Computer and Communications Security 2016: 31-42.[doi: 10.1145/2976749.2978399]

[17] Ethereum. https://github.com/ethereum/wiki/wiki/White-Paper. White paper, 2020.

[18] Coblee. Litecoin - a lite version of Bitcoin. Launched! https://bitcointalk.org/index.php?topic=47417.0. Accessed on: 2019.8.9.

[19] Androulaki E, Barger A, Bortnikov V, Cachin C, Christidis K, De Caro A, Enyeart D, Ferris C, Laventman G, Manevich Y. Hyperledger fabric: a distributed operating system for permissioned blockchains. EuroSys Conference 2018: 30.[doi: 10.1145/3190508.3190538]

[20] Gramoli V. From blockchain consensus back to byzantine consensus. Future Generation Computer Systems, 2020, 107: 760-769.[doi: 10.1016/j.future.2017.09.023]

[21] Nguyen G-T, Kim K. A Survey about Consensus Algorithms Used in Blockchain. Journal of Information processing systems, 2018, 14(1).[doi: 10.3745/JIPS.01.0024]

[22] Wang W, Hoang D T, Hu P, Xiong Z, Niyato D, Wang P, Wen Y, Kim D I. A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks. IEEE Access, 2019, 7 22328-22370.[DOI: 10.1109/ACCESS.2019.2896108]

[23] Yuan Yong, Wang Fei-Yue. Blockchain: The state of the art and future trends. Acta Automatica Sinica, 2016,42(4):481-494. [doi: 10.16383/j.aas.2016.c160158]

[24] Kiayias A, Russell A, David B, Oliynykov R. Ouroboros: A provably secure proof-of-stake blockchain protocol. Annual International Cryptology Conference 2017: 357-388.[doi: 10.1007/978-3-319-63688-7_12]

[25] Li C, Li P, Zhou D, Yang Z, Wu M, Yang G, Xu W, Long F, Yao A C-C. A Decentralized Blockchain with High Throughput and Fast Confirmation. USENIX Annual

Technical Conference 2020: 515-528.

[26] Lewenberg Y, Sompolinsky Y, Zohar A. Inclusive block chain protocols. International Conference on Financial Cryptography and Data Security 2015: 528-547.[doi: 10.1007/978-3-662-47854-7_33]

[27] Castro M, Liskov B. Practical Byzantine fault tolerance. USENIX Symposium on Operating Systems Design and Implementation 1999: 173-186.

[28] Mazieres D. The stellar consensus protocol: A federated model for internet-level consensus. Stellar Development Foundation, 2015.

[29] Burrows M. The Chubby lock service for loosely-coupled distributed systems. Proceedings of the 7th symposium on Operating systems design and implementation 2006: 335-350.

[30] Lamport L. Paxos made simple. ACM Sigact News, 2001, 32(4): 18-25.

[31] Lamport L, Shostak R, Pease M. The Byzantine generals problem. ACM Transactions on Programming Languages and Systems, 1982, 4(3): 382-401.[doi: 10.1145/3335772.3335936]

[32] Pease M, Shostak R, Lamport L. Reaching agreement in the presence of faults. Journal of the ACM, 1980, 27(2): 228-234.

[33] Decker C, Wattenhofer R. Information propagation in the bitcoin network. International Conference on Peer-to-Peer Computing, 2013: 1-10.[doi: 10.1109/P2P.2013.6688704]

[34] Bitfury Group J G. Public versus Private Blockchains. SSRN Electronic Journal, 2015. [doi: 10.2139/ssrn.3486578]

[35] Vukolić M. Rethinking permissioned blockchains. Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts 2017: 3-7.[doi: 10.1145/3055518.3055526]

[36] Bitcoin nodes distribution. https://bitnodes.earn.com/. Accessed on: 2019.8.9.

[37] Ethereum nodes distribution. https://www.ethernodes.org/network/1. Accessed on: 2019.8.9.

[38] Shao QF, Zhang Z, Zhu YC, Zhou AY. Survey of Enterprise Blockchains. Journal of Software, 2019, 30(9): 2571-2592.[doi: 10.13328/j.cnki.jos.005775]

[39] Zheng Z, Xie S, Dai H, Chen X, Wang H. An overview of blockchain technology: Architecture, consensus, and future trends. IEEE International Congress on Big Data 2017: 557-564.[doi: 10.1109/BigDataCongress.2017.85]

[40] Le Lann G. Distributed Systems-Towards a Formal Approach. International Federation for Information Processing congress, 1977: 155-160.

[41] Dwork C, Lynch N, Stockmeyer L. Consensus in the presence of partial synchrony. Journal of the ACM, 1988, 35(2): 288-323.[doi: 10.1145/42282.42283]

[42] Ongaro D, Ousterhout J. In search of an understandable consensus algorithm. USENIX Annual Technical Conference, 2014: 305-319.

[43] Douceur J R. The sybil attack. International workshop on peer-to-peer systems, 2002: 251-260.[doi: 10.1007/3-540-45748-8_24]

[44] King S, Nadal S. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. White paper, 2012.

[45] Nxt community. https://nxtwiki.org/wiki/Whitepaper:Nxt. White paper. Accessed on: 2019.8.9.

[46] Jakobsson M, Juels A. Proofs of work and bread pudding protocols. Secure Information Networks; Springer. 1999: 258-272.[doi: 10.1007/978-0-387-35568-9_18]

[47] Dogecoin Project. https://github.com/dogecoin/dogecoin. Accessed on: 2019.8.9.

[48] Sasson E B, Chiesa A, Garman C, Green M, Miers I, Tromer E, Virza M. Zerocash: Decentralized anonymous payments from bitcoin. IEEE Symposium on Security and Privacy, 2014: 459-474.[doi: 10.1109/SP.2014.36]

[49] Miers I, Garman C, Green M, Rubin A D. Zerocoin: Anonymous distributed e-cash from bitcoin. IEEE Symposium on Security and Privacy 2013: 397-411.[doi: 10.1109/SP.2013.34]

[50] Miller A, Kosba A, Katz J, Shi E. Nonoutsourceable scratch-off puzzles to discourage bitcoin mining coalitions. Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security 2015: 680-691.[doi: 10.1145/2810103.2813621]

[51] Luu L, Velner Y, Teutsch J, Saxena P. Smartpool: Practical decentralized pooled mining. USENIX Security Symposium 2017: 1409-1426.

[52] King S. Primecoin: Cryptocurrency with prime number proof-of-work. White paper, 2013.

[53] Ball M, Rosen A, Sabin M, Vasudevan P N. Proofs of Useful Work. IACR Cryptology ePrint Archive, 2017, 203.

[54] Ateniese G, Bonacina I, Faonio A, Galesi N. Proofs of space: When space is of the essence. International Conference on Security and Cryptography for Networks, 2014: 538-557.[doi: 10.1007/978-3-319-10879-7_31]

[55] Stefano Tempesta. Deploy Ethereum Proof-of-Authority. Introduction to Blockchain for Azure Developers. Apress, 2019. [doi: 10.1007/978-1-4842-5311-3_2]

[56] Gai F, Wang B, Deng W, Peng W. Proof of reputation: A reputation-based consensus protocol for peer-to-peer network. International Conference on Database Systems for Advanced Applications 2018: 666-681.[doi: 10.1145/3343147.3343169]

[57] Ateniese G, Magri B, Venturi D, Andrade E. Redactable blockchain--or--rewriting history in bitcoin and friends. IEEE European Symposium on Security and Privacy 2017: 111-126.[doi: 10.1109/EuroSP.2017.37]

[58] Narayanan A, Bonneau J, Felten E, Miller A, Goldfeder S. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. 1st Edition, Princeton University Press, 2016, 24-30.

[59] Aspnes J, Jackson C, Krishnamurthy A. Exposing computationally-challenged Byzantine impostors. Vol. 6. Technical Report, Yale University Department of Computer Science, 2005.

[60] Bentov I, Lee C, Mizrahi A, Rosenfeld M. Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake. ACM SIGMETRICS Performance Evaluation

Review, 2014, 42(3): 34-37.

[61] Sompolinsky Y, Lewenberg Y, Zohar A. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. IACR Cryptology ePrint Archive, 2016, 1159.

[62] Bitcoin Cash. https://en.wikipedia.org/wiki/Bitcoin_Cash. Accessed on: 2019.8.9.

[63] Bitcoin hash rate. https://bitinfocharts.com/comparison/bitcoin-hashrate.html. Accessed on: 2019.8.9.

[64] Eyal I, Sirer E G U, N. Majority is not enough: Bitcoin mining is vulnerable. Communications of the ACM, 2018, 61(7): 95-102.[doi: 10.1145/3212998]

[65] Sapirshtein A, Sompolinsky Y, Zohar A. Optimal selfish mining strategies in bitcoin. International Conference on Financial Cryptography and Data Security 2016: 515-532.[doi: 10.1007/978-3-662-54970-4_30]

[66] Heilman E, Kendler A, Zohar A, Goldberg S. Eclipse attacks on bitcoin's peer-to-peer network. 24th USENIX Security Symposium 2015: 129-144.

[67] Alwen J, Chen B, Pietrzak K, Reyzin L, Tessaro S. Scrypt is maximally memory-hard. Annual International Conference on the Theory and Applications of Cryptographic Techniques 2017: 33-62.[doi: 10.1007/978-3-319-56617-7_2]

[68] Vujičić D, Jagodić D, Ranđić S. Blockchain technology, bitcoin, and Ethereum: A brief overview. 2018 17th International Symposium INFOTEH-JAHORINA 2018: 1-6.[doi: 10.1109/INFOTEH.2018.8345547]

[69] Biryukov A, Khovratovich D. Equihash: Asymmetric proof-of-work based on the generalized birthday problem. Ledger, 2017, 2 1-30.[doi: 10.5195/LEDGER.2017.48]

[70] De Vries A. Bitcoin's growing energy problem. Joule, 2018, 2(5): 801-805.[doi: 10.1016/j.joule.2018.04.016]

[71] Bentov I, Gabizon A, Mizrahi A. Cryptocurrencies without proof of work. International Conference on Financial Cryptography and Data Security 2016: 142-157.[doi: 10.1007/978-3-662-53357-4_10]

[72] Badertscher C, Gaži P, Kiayias A, Russell A, Zikas V. Ouroboros genesis: Composable proof-of-stake blockchains with dynamic availability. ACM SIGSAC Conference on Computer and Communications Security, 2018: 913-930.[doi: 10.1145/3243734.3243848]

[73] Unconfirmed transactions in bitcoin. https://www.blockchain.com/btc/unconfirmed-transactions. Accessed on: 2019.8.9.

[74] Gervais A, Karame G O, Wüst K, Glykantzis V, Ritzdorf H, Capkun S. On the security and performance of proof of work blockchains. ACM SIGSAC conference on computer and communications security, 2016: 3-16.[doi: 10.1145/2976749.2978341]

[75] Proof of stake instead of proof of work. https://bitcointalk.org/index.php?topic=27787. Accessed on: 2019.8.9.

[76] Cloakcoin. A private, secure and untraceable transaction system for cloakcoin. White paper, 2018.

[77] Novacoin project. https://github.com/novacoin-project/novacoin/wiki/Proof-of-stake. Accessed on: 2019.8.9.

[78] Vasin P. BlackCoin's Proof-of-Stake Protocol. White paper, 2014.

[79] Douglas p, Patrick n, David b, Daniel g, Steve w, Joshua m. Proof-of-Stake-Time. White paper, 2015.

[80] Buchman E. Tendermint: Byzantine fault tolerance in the age of blockchains. Query date, 2018, 02-26.

[81] David B, Gaži P, Kiayias A, Russell A. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. Annual International Conference on the Theory and Applications of Cryptographic Techniques 2018: 66-98.[doi: 10.1007/978-3-319-78375-8_3]

[82] Buterin V. Slasher: A punitive proof-of-stake algorithm. Technical report, 2014.

[83] Buterin V, Reijsbergen D, Leonardos S, Piliouras G. Incentives in Ethereum's hybrid casper protocol. IEEE international conference on blockchain and cryptocurrency. 2019: 236-244.

[84] Kwon J. Tendermint: Consensus without mining. Technical report, 2014.

[85] Li W, Andreina S E, Bastien, Bohli J-M, Karame G. Securing proof-of-stake blockchain protocols. Data Privacy Management, Cryptocurrencies and Blockchain Technology; Springer. 2017: 297-315.[doi: 10.1007/978-3-319-67816-0_17]

[86] Gaži P, Kiayias A, Russell A. Stake-bleeding attacks on proof-of-stake blockchains. 2018 Crypto Valley Conference on Blockchain Technology 2018: 85-92.[doi: 10.1109/CVCBT.2018.00015]

[87] Daian P, Pass R, Shi E. Snow white: Robustly reconfigurable consensus and applications to provably secure proof of stake. International Conference on Financial Cryptography and Data Security. Springer, Cham, 2019: 23-41.[doi: 10.1007/978-3-030-32101-7_2]

[88] Larimer D. Delegated proof-of-stake. Bitshare whitepaper, 2014.

[89] Bitcoin transaction confirmation. https://www.buybitcoinworldwide.com/confirmations/. Accessed on: 2019.8.9.

[90] Rudden J. Average confirmation time of Bitcoin transactions from January 2018 to April 2020. https://www.statista.com/statistics/793539/bitcoin-transaction-confirmation-time/. Accessed on: 2019.8.9.

[91] Ethereum adopts the longest chain rule. https://github.com/ethereum/go-ethereum/blob/525116dbff9168254639313611f75e75e955c12e2/core/blockchain.go. Accessed on: 2019.8.9.

[92] Luu L, Saha R, Parameshwaran I, Saxena P, Hobor A. On power splitting games in distributed computation: The case of bitcoin pooled mining. IEEE Computer Security Foundations Symposium 2015: 397-411.[doi: 10.1109/CSF.2015.34]

[93] Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol with chains of variable difficulty. Annual International Cryptology Conference 2017: 291-323.[doi: 10.1007/978-3-319-63688-7_10]

[94] Garay J A, Kiayias A, Panagiotakos G. Proofs of Work for Blockchain Protocols. IACR Cryptology ePrint Archive, 2017, 775.

[95] Pass R, Seeman L, Shelat A. Analysis of the blockchain protocol in asynchronous networks. Annual International Conference on the Theory and Applications of Cryptographic Techniques 2017: 643-673.[doi: 10.1007/978-3-319-56614-6_22]

[96] Kiayias A, Panagiotakos G. On trees, chains and fast transactions in the blockchain. International Conference on Cryptology and Information Security in Latin America. Springer, Cham, 2017: 327-351.[doi: 10.1007/978-3-030-25283-0_18]

[97] Nayak K, Kumar S, Miller A, Shi E. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. IEEE European Symposium on Security and Privacy 2016: 305-320.[doi: 10.1109/EuroSP.2016.32]

[98] Ritz F, Zugenmaier A. The impact of uncle rewards on selfish mining in ethereum. IEEE European Symposium on Security and Privacy Workshops 2018: 50-57.[doi: 10.1109/EuroSPW.2018.00013]

[99] Kiayias A, Panagiotakos G. Speed-Security Tradeoffs in Blockchain Protocols. IACR Cryptology ePrint Archive, 2015, 1019.

[100] Lamport L. Fast byzantine paxos. IEEE Transactions on Dependable and Secure Computing, 2006, 3(3) 202-215.

[101] Cachin C. Yet another visit to Paxos. IBM Research, Zurich, Switzerland, Technical Report, 2009.

[102] Pass R, Shi E. Hybrid consensus: Efficient consensus in the permissionless model. International Symposium on Distributed Computing, 2017.

[103] Mazieres D. The stellar consensus protocol: A federated model for internet-level consensus. White paper, 2015.

[104] Li CX, Chen S, Zheng LS, Zuo C, Jiang BY, Liang G. RepChain—A permissioned blockchain toolkit implemented by reactive programming. Ruan Jian Xue Bao/Journal of Software, 2019,30(6):1670−1680.[doi: 10.13328/j.cnki.jos.005743]

[105] Cachin C. Architecture of the hyperledger blockchain fabric. Workshop on distributed cryptocurrencies and consensus ledgers 2016: 310(4).

[106] Hyperledger. https://www.hyperledger.org/. Accessed on: 2019.8.9.

[107] Hyperledger fabric. https://hyperledger-fabric.readthedocs.io/en/release-1.4/orderer/ordering_service.html. Accessed on: 2019.8.9.

[108] Viriyasitavat W, Hoonsopon D. Blockchain characteristics and consensus in modern business processes. Journal of Industrial Information Integration, 2019, 13 32-39. [doi: 10.1016/j.jii.2018.07.004]

附中文参考文献

[23] 袁勇, 倪晓春, 曾帅, 王飞跃. 区块链共识算法的发展现状与展望. 自动化学报, 2018, 44(11): 2011-2022. [doi: 10.16383/j.aas.2016.c160158]

[38] 邵奇峰, 张召, 朱燕超, 周傲英. 企业级区块链技术综述. 软件学报, 2019 ,30(9):2571−2592. [doi: 10.13328/j.cnki.jos.005775]

[104] 李春晓, 陈胜, 郑龙帅, 左春, 蒋步云, 梁赓. 响应式许可链基础组件——RepChain. 软件学报, 2019,30(6): 1670-1680. [doi: 10.13328/j.cnki.jos.005743]