软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(2):773−799 [doi: 10.13328/j.cnki.jos.006985]

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

区块链网络综述*

司冰茹[1,2,3,4], 肖 江[1,2,3,4], 刘存扬[1,2,3,4], 戴小海[1,2,3,4], 金 海[1,2,3,4]

[1](大数据技术与系统国家地方联合工程研究中心, 湖北 武汉 430074)

[2](服务计算技术与系统教育部重点实验室, 湖北 武汉 430074)

[3](集群与网格计算湖北省重点实验室, 湖北 武汉 430074)

[4](华中科技大学 计算机学院, 湖北 武汉 430074)

通信作者: 肖江, E-mail: jiangxiao@hust.edu.cn

摘 要: 区块链是典型的分布式系统, 底层网络的性能和安全性至关重要. 区块链网络的本质是 P2P 网络, 然而在安全模型、传输协议和性能指标等方面与传统 P2P 网络存在明显差异. 首先, 针对区块链网络的传输流程进行全面、深入地分析, 阐明区块链网络所面临的瓶颈挑战. 其次, 针对区块链网络拓扑结构和传输协议的最新研究工作, 从节点异构性、编码方案、广播算法和中继网络等方面系统性地分类梳理, 并归纳总结跨链网络实现和网络仿真工具. 最后, 探讨区块链网络的未来研究趋势.

关键词: 区块链; 拓扑结构; 传输协议; 跨链网络; 仿真工具

中图法分类号: TP393

中文引用格式: 司冰茹, 肖江, 刘存扬, 戴小海, 金海. 区块链网络综述. 软件学报, 2024, 35(2): 773–799. http://www.jos.org.cn/1000-9825/6985.htm

英文引用格式: Si BR, Xiao J, Liu CY, Dai XH, Jin H. Survey on Blockchain Network. Ruan Jian Xue Bao/Journal of Software, 2024, 35(2): 773–799 (in Chinese). http://www.jos.org.cn/1000-9825/6985.htm

Survey on Blockchain Network

SI Bing-Ru[1,2,3,4], XIAO Jiang[1,2,3,4], LIU Cun-Yang[1,2,3,4], DAI Xiao-Hai[1,2,3,4], JIN Hai[1,2,3,4]

[1](National Engineering Research Center for Big Data Technology and System, Wuhan 430074, China)

[2](Services Computing Technology and System Lab, Wuhan 430074, China)

[3](Cluster and Grid Computing Lab, Wuhan 430074, China)

[4](School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: Blockchain, as a typical distributed system, its underlying networks highly influences the overall system performance and security. Blockchain networks differ from traditional P2P (peer-to-peer) networks in terms of security models, transmission protocols and performance indicators. This study first systematically analyzes the blockchain network transmission process, i.e., connection establishment and data transmission, and list out the challenging issues. Second, state-of-the-art blockchain topology protocols and data transmission methods are thoroughly investigated and discussed, from the perspective of node heterogeneity, coding scheme, broadcast algorithm and relay network, and etc. Meanwhile, the typical cross-chain network implementation and the network simulation tools are summarized. Finally, we envision the possible future research trends in the realm of blockchain networks.

Key words: blockchain; topology structure; transport protocol; cross-chain network; simulation tool

自 2008 年中本聪首次提出比特币[1]系统以来, 区块链技术引起广泛关注. 区块链技术是由分布式存储[2]、共识机制[3]、点对点传输、智能合约和加密算法等多种技术创新融合, 具有去中心化、开放性、防篡改性、匿名性

* 基金项目: 国家重点研发计划 (2021YFB2700700); 国家自然科学基金 (62072197); 湖北省重点研发计划 (2021BEA164)
收稿时间: 2022-07-30; 修改时间: 2022-12-02, 2023-04-28; 采用时间: 2023-07-12; jos 在线出版时间: 2023-11-01
CNKI 网络首发时间: 2023-11-02

和可追溯性等特征[4]. 随着区块链技术演进发展, 其应用领域已经从最初的数字货币扩展到金融服务[5]、医疗健康[6]、政务民生[7]和溯源管理[8]等领域.

区块链系统的节点具有开放、分布、自治等特性, 所有节点构成扁平式拓扑结构进行通信, 不存在任何中心节点. 通过点对点通信, 节点能够在去信任的竞争环境下不依赖第三方机构实现可信的数据共享和价值传输. 区块链网络节点交互可以分为两个主要步骤: 连接建立和数据传输. 在连接建立阶段, 新节点选择网络中已有节点构建连接并加入网络. 在数据传输阶段, 所有节点利用传输协议验证并转发消息, 以达到数据全网广播的目的. 区块链网络协议制定节点连接建立和数据传输的规则, 支撑一致性共识操作等.

网络协议作为区块链系统底层架构的核心组件, 对区块链的性能和安全性至关重要. 比特币测量报告显示, 一个区块被网络中 90% 节点接收平均花费时间 79 s, 约为验证和中继时间的 40 倍[9]. 该数据表明, 区块传播延迟是限制区块链吞吐量的主要因素之一, 与区块大小、网络拓扑和传输路径密切相关. 传播时间越长, 矿工在旧区块上挖矿的频率越高, 从而导致区块链分叉的概率和孤块的比率越大, 系统的安全性越低.

目前研究人员主要从可信连接[10,11]、交易编码[12−18]和广播算法[19−27]等方面提升区块链网络传输系统的性能和安全性. 随着区块链网络研究的推进, 研究人员开始对研究工作进行梳理: 文献 [28] 讨论了非许可区块链的网络层; 文献 [29] 讨论了 3 种加密货币以太坊[30]、Nano[31]和 IOTA[32]的底层网络和数据传输协议; 文献 [33] 对加密货币网络的背景、技术和挑战进行了概述; 文献 [34] 按区块链架构梳理了主要的网络优化技术. 上述研究工作将网络视为整体讨论, 缺乏对网络传输流程的步骤划分以及各阶段工作的详细对比.

本文中, 我们从区块链网络的传输流程入手, 对各阶段的研究工作进行全面梳理及对比分析, 同时对跨链网络和仿真工具进行归纳总结, 观察区块链网络的发展趋势, 为未来的研究方向提供建议. 具体而言, 在连接建立阶段, 我们主要围绕网络拓扑结构展开讨论, 分析分布式非结构化、分布式结构化和半分布式拓扑的特点及相关应用, 从节点同构和异构角度总结现有的拓扑设计. 在数据传输阶段, 我们分析比特币数据传输机制的缺陷, 从编码方案、广播算法和中继网络角度总结现有的传输协议, 其中重点关注基于交易和区块中继的编码方案. 之后, 我们描述跨链网络通信和网络仿真工具, 总结区块链网络发展趋势以及可能的研究方向, 为研究人员提供有用的参考.

本文第 1 节说明区块链网络的传输流程, 通过与传统对等网络的对比分析说明区块链网络面临的瓶颈挑战. 第 2 节描述区块链网络拓扑结构及其应用系统, 分析比较拓扑设计的现有工作. 第 3 节描述区块链网络传输流程, 分析比较各种传输协议. 第 4 节描述跨链网络通信和技术实现. 第 5 节描述区块链网络仿真需求并比较仿真工具. 第 6 节总结研究发现及展望未来可能的研究方向.

1 区块链网络

随着互联网的快速普及以及用户规模的海量增长, 传统的客户端/服务器 (client/server, C/S) 模式已难以满足流媒体等的新型应用需要. 在这种背景下, 对等网络 (peer-to-peer, P2P) 技术[35]应运而生, 迅速演变成一种重要的网络模式. P2P 网络由大量地位对等的计算机节点通过随机方式或某些特定规则组织而成, 每个节点既是服务的请求者, 也对其他节点的服务请求进行响应. 节点之间直接进行资源调用和数据共享, 避免中心化服务器可能带来的性能瓶颈, 降低对服务器的依赖. 与 C/S 模式相比, P2P 模式具有对等性、可扩展性、健壮性、高效性等优势.

区块链网络本质上是 P2P 网络[36], 所有节点在不互信的环境中采用点对点通信, 通过组网方式、消息传播协议和验证机制等模块实现数据的全网传输和简单验证. 只有当区块数据被网络中大部分节点验证通过后, 才能将其写入区块链. 区块链网络在安全模型、传输协议和性能指标等方面与传统 P2P 网络存在明显差异. 例如区块链网络可能具有更高的安全要求, 可能需要服务于不同的传输流量, 或者性能可能受到更多因素限制等 (详见第 1.3 节).

区块链网络传输流程可以分为两个阶段: 连接建立阶段和数据传输阶段, 如图 1 所示. 在连接建立阶段, 新节点利用发现协议获取网络中其他节点的地址, 连接成功后加入网络, 同步当前账本状态的数据并定期维护自身交易列表. 在数据传输阶段, 节点利用传输协议广播并验证交易和区块. 具体而言, 区块链网络中任何节点可以发起交易, 交易通过 P2P 网络广播. 收到交易的节点验证其有效性, 若有效则加入本地交易池并转发到其他邻居节点,

若无效则直接丢弃. 随着交易不断生成积累, 所有节点争夺记账权. 获得记账权的节点打包交易、生成新区块并广播. 收到区块的节点验证区块及其中交易的有效性, 若有效则更新本地区块链数据并继续转发.

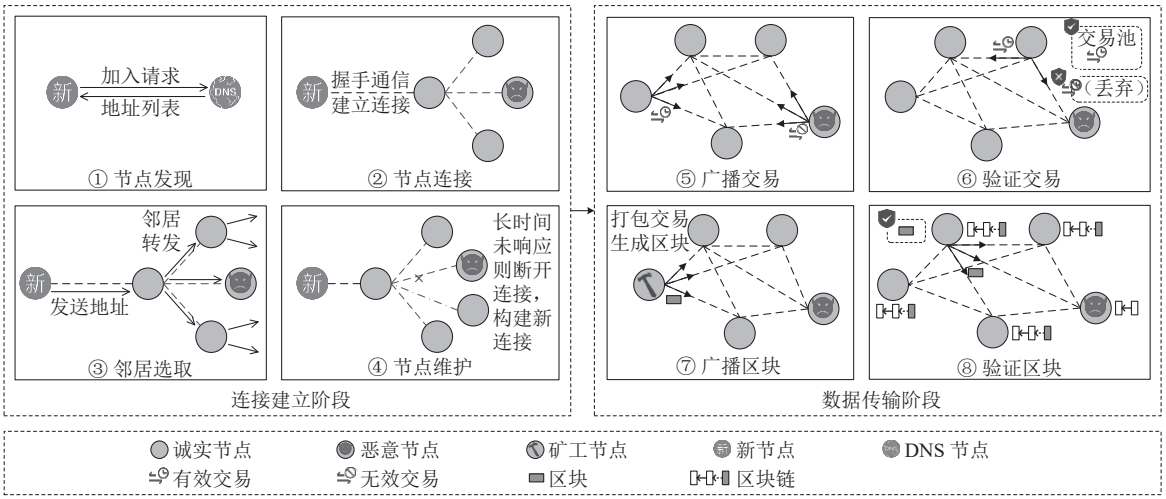


图 1　区块链网络传输流程图

区块链的网络环境主要包括开放公有的公有链 (public blockchain)、受限可控的联盟链 (consortium blockchain) 和私有链 (private blockchain). 公有链中所有节点可以自由加入和退出网络, 参与链上的交易确认和共识机制. 每个节点都有发起交易和读取数据的权限. 由于匿名性, 节点能够很好地隐藏真实身份. 私有链是对某个组织开放的区块链系统. 所有节点必须经过组织授权才能进入网络, 各个节点的写入权限由组织控制, 而读取权限可视情况选择性开放. 联盟链是由若干个组织共同维护的区块链, 每个组织管理一个或多个节点, 其开放程度介于公有链和私有链之间. 联盟链中节点必须获得联盟许可才能加入网络, 各个节点读写和记账权限由联盟规则决定. 鉴于开放程度的不同, 公有链的网络规模通常较大, 网络系统根据节点自由扩展, 而联盟链和私有链的网络规模相对较小. 因此, 本文主要关注公有链网络, 特别是比特币系统的相关研究工作, 对联盟链和私有链网络的讨论较少.

1.1　连接建立阶段

节点连接机制是构建网络的基础. 为了保障网络的可扩展性、负载均衡等特性, 新节点加入应避免大幅网络扰动. 根据节点连接机制, 区块链网络可以形成不同的拓扑结构, 通常采用网络连通性和网络直径来衡量其对数据广播的影响. 网络连通性指网络中每个节点连接的邻居节点的数量. 网络连通性越大, 节点单次可传输数据的邻居越多. 网络直径指网络中任意两个节点之间的最大网络延迟. 网络直径越小, 任意两个节点之间的平均广播时间越短. 本节从区块链连接建立流程的 4 个关键步骤, 包括节点发现、节点连接、邻居选取和节点维护进行详细说明.

1.1.1　节点发现

在 P2P 网络中, 节点初始加入网络需要了解至少一个网络节点的地址信息并以其作为入口. 传统 P2P 网络节点利用泛洪、组播等方式向节点或服务器发送请求以寻找存活的节点[37]. 区块链网络节点发现与传统 P2P 网络类似, 主流采用两种方式. 第 1 种是域名系统 (domain name system, DNS) 种子节点, 即节点向 DNS 服务器发送请求获取种子节点的地址, DNS 服务器返回一个可连接的 IP 地址列表. 第 2 种是硬编码种子节点, 即将一部分长期稳定运行的节点硬编码至代码中. 不同区块链系统的节点发现方式略有差异. 例如, 比特币通过“-dnsseed”选项指定是否使用种子节点, 默认情况下该选项开启; 以太坊通过硬编码种子节点或矿池种子节点 (国内组织和个人分享的稳定运行的节点地址) 接入网络; IOTA 没有自动发现节点机制, 新节点在启动时通过手动配置的方式发现节点.

1.1.2　节点连接

新节点发现可连接的节点地址后, 通过握手通信建立连接. 握手指正在运行的节点互相通信、彼此感知的过

程. 在传统网络中, TCP 协议能够提供可靠的连接服务, 需要 3 次握手进行连接初始化. 在区块链网络中, 节点连接为不可靠的环境提供可靠的数据传输通道. 现有主流区块链系统采用不同的节点连接方式. 例如, 比特币节点发送 version 信息开始握手通信, 收到响应后连接建立成功, 确保节点的兼容性; 以太坊节点通过密钥交换和协议协商两个阶段完成连接, 确定传输过程中的规则; IOTA 节点只要连接协议类型相同即可连接.

1.1.3 邻居选取

邻居选取决定网络拓扑和性能. 当连接成功后, 新节点需要足够数量的邻居确保连接的稳定性, 即新节点将自身的地址信息发送给邻居, 邻居节点进行转发以保证新节点被更多节点发现. 区块链有多种邻居选取方法. 例如, 比特币随机选取邻居构建连接, 这在公有链系统中较为常用; 以太坊利用结构化数据结构存储邻居节点并连接; IOTA 通过 addNeighbors 命令指定邻居节点并添加, 这种方式在联盟链和私有链中较为常见.

1.1.4 节点维护

节点加入网络后必须持续进行节点维护, 在旧连接断开时及时添加新连接, 保证通信路径的可靠. 区块链系统周期性地管理节点的连接. 例如, 比特币和以太坊定期发送探活消息维持连接; IOTA 在每个周期内随机中断部分连接并每分钟检查一次邻居节点的活性. 区块链网络会根据节点和网络的状态动态变化, 不需要经过中心化的控制即可实现网络规模的调整.

1.2 数据传输阶段

在区块链网络的数据传输阶段, 每个节点持续监听网络中广播的新交易和新区块. 当数据生成后, 该生成节点根据传输协议选取邻居节点并广播. 接收数据的节点验证数据后转发, 直到消息到达全网节点. 本节从区块链数据传输流程的 3 个关键步骤, 包括交易广播、区块广播和数据验证进行详细说明.

1.2.1 交易广播

交易是区块链网络的最小数据传输单元, 用户可以简单地使用区块链客户端如钱包发起交易. 比特币中, 节点将交易发送给除发送节点和其他已收到交易的节点之外的所有邻居节点. 以太坊中, 节点记录发送给每个邻居的交易信息并创建交易缺失列表, 每次仅向邻居节点广播一批交易中缺少的交易. IOTA 中, 由于账本结构的不同, 交易验证先前的两笔交易后再进行广播.

1.2.2 区块广播

区块是组成区块链分布式账本的基本数据结构, 由区块头和一组交易组成. 在比特币系统中区块生成需要各节点通过工作量证明共识机制达成一致, 这实际上将区块的创建限制在矿池中. 比特币中节点将新生成的区块发送所有邻居节点. 以太坊中节点随机选取部分邻居节点广播整个区块, 向剩余节点广播区块哈希, 收到区块哈希的节点将向源节点请求下载区块数据.

1.2.3 数据验证

数据验证对区块链系统安全性极为重要. 节点一旦接收到新交易和新区块, 首先验证其有效性. 若有效则进行转发, 否则立即丢弃, 以免浪费资源. 对于新交易的验证, 根据预先拟定的验证协议进行, 如交易的格式和内容的正确性等. 对于新区块的验证, 通过区块头验证可确保父区块哈希、区块编号、时间戳和工作量证明正确无误.

1.3 与传统 P2P 网络的对比

P2P 技术诞生于文件共享、分布式存储等应用, 现已广泛扩展到即时通信、移动网络等众多领域. 区块链网络是 P2P 网络的一种新型通信范式, 对于区块链应用和系统的安全可信与性能至关重要. 表 1 对传统 P2P 网络和区块链 P2P 网络进行对比分析.

(1) 信任假设. 传统 P2P 网络和区块链 P2P 网络的根本区别在于节点之间的信任关系. 传统 P2P 网络节点互相信任协作, 需要对恶意行为采取一定的措施以保证系统正常运行. 区块链网络节点互不信任, 需要兼顾恶意节点行为以保证系统公平性.

(2) 异常容错. 由于网络的扰动问题, 节点和数据的动态删减会使节点状态变得陈旧, 与实际网络应有的状态不一致. 传统 P2P 网络采用故障容错机制, 通过冗余和周期性检测等方法应对和抑制恶意行为, 存在崩溃宕机的风

险. 区块链网络的安全模型为拜占庭容错, 需确保系统在存在恶意节点的故障容忍性. 当发现可能发起攻击的可疑节点时, 区块链网络可以手动配置可信节点连接, 同时将可疑节点加入黑名单.

表 1　传统 P2P 网络和区块链 P2P 网络对比

分析维度	特征	传统P2P网络	区块链P2P网络
安全模型	信任假设	互信	不互信
	异常容错	故障容错(crash fault tolerance, CFT)	拜占庭容错(Byzantine fault tolerance, BFT)
	攻击类型	女巫攻击、信息窃取、DDoS攻击等	DDoS攻击、日蚀攻击、双花攻击、自私挖矿等
传输协议	传输需求	数据共享、资源查找	同步更新、数据一致性
	拓扑结构	中心化、分布式结构化、非结构化、半分布式	分布式结构化、非结构化、半分布式
	广播算法	数据在资源需求方和资源提供方之间广播	数据全网节点广播
性能指标	传播延迟	资源分布情况、搜索算法等	区块大小、拓扑结构、广播算法、网络规模等
	传输效率	链路带宽、传输能力等硬件限制	网络带宽、数据冗余度、区块链分叉等
	可扩展性	整体资源和服务能力同步扩充	数据同步和验证限制、网络共识

(3) 攻击类型. P2P 网络广泛应用的同时, 其安全问题也逐渐显露出来. 传统 P2P 网络的安全问题包括节点信任问题、节点通信安全问题和系统安全问题, 例如女巫攻击[38]、信息窃取[39]和分布式拒绝服务 (distributed denial of service, DDoS) 攻击[40]等. 区块链网络的安全性问题除针对 P2P 网络的攻击外, 还包括拜占庭环境下利用网络特性破坏共识的攻击行为, 如通过日蚀攻击[41]控制节点进而执行双花攻击[42]和自私挖矿[43]等.

(4) 传输需求. 传统 P2P 网络的主要任务是数据共享和资源查找. 根据分布式系统性质, 资源分散地存储于整个网络中, 资源备份和恢复较为复杂. 区块链网络的主要目标是实现数据在全网节点的一致性同步更新. 区块链技术将全网数据同时存储在所有节点上, 每笔数据必须经过签名验证和全网共识后才可写入区块链. 数据一旦被记录, 任何人不可篡改和否认.

(5) 拓扑结构. 传统 P2P 网络拓扑结构可分为中心化 (如 Napster[44])、分布式非结构化 (如 Gnutella[45])、分布式结构化 (如 Kademlia[46]) 和半分布式 (如 KaZaA[47]) 这 4 类. 拓扑研究重点在于网络扰动问题, 如节点加入离开、失效恢复策略、系统维护策略等. 鉴于区块链去中心化的特征, 区块链网络拓扑主流采用分布式非结构化 (如比特币)、分布式结构化 (如以太坊) 和半分布式 (如超级账本 Fabric[48]) 这 3 类. 拓扑研究重点在于对网络可扩展性和共识速度的影响. 为适用不同的应用需求, 区块链网络拓扑仍在不断演进.

(6) 广播算法. 在传统 P2P 网络中, 资源需求方通过泛洪或结构化广播等方式向邻居节点发送请求, 邻居节点转发消息直到到达资源提供方. 一般情况下, 查询请求受到次数和时间限制, 可能无法通知到网络中的所有节点. 在区块链网络中, 交易和区块的生成者向全网节点广播消息. 例如, 比特币采用基于泛洪的广播算法, 以太坊采用基于 Gossip[49]的广播算法. 这种网状广播算法实现简单, 路径容错性好, 但会产生大量冗余数据.

(7) 传播延迟. 在传统 P2P 网络中, 传播延迟指请求从资源需求方传播到资源提供方所需要的时间, 与资源在网络中的分布情况和搜索算法等密切相关. 在区块链网络中, 理论上传播延迟指交易或区块到达网络中所有节点所需要的时间, 与区块大小、拓扑结构、传输协议和网络规模等密切相关.

(8) 传输效率. P2P 网络取消中间环节直接建立传输通道, 传输效率受链路带宽、传输能力等网络硬件条件的限制. 传统 P2P 系统广泛使用多源传输和多路由传输技术, 通过并行化传输提高传输效率. 在区块链网络中, 除网络硬件限制, 数据冗余和无效传输及区块链分叉等现象也将造成系统传输效率低下, 使其无法应用于现实世界的支付和其他大规模应用.

(9) 可扩展性. 随着网络规模的扩大, 传统 P2P 网络的整体资源和服务能力也在同步扩充, 始终能够满足用户不断增长的需求, 响应速度快, 可扩展性强. 然而, 目前大多数区块链平台的整体性能随着网络规模的扩大而下降. 由于节点加入导致数据同步和验证的开销增多, 网络达成共识的时间显著增加, 区块链网络的可扩展性较差.

1.4 挑　战

目前, 对传统 P2P 网络的拓扑结构和传输路由等研究已相对成熟, 但对区块链网络的研究仍处于初步阶段. 由

于区块链网络的特殊性和两者的差异, 区块链网络面临以下挑战.

(1) 安全性

安全问题是区块链系统面临的最主要挑战之一. 虽然去中心化特性为缺乏信任的环境提供更高的安全性, 但仍引入新的安全威胁. 区块链网络面临的安全问题主要是针对 P2P 网络的攻击行为和拜占庭环境下利用网络特性破坏共识的攻击行为. 在针对 P2P 网络的攻击行为中, 攻击者可以通过网络窃听获取节点身份和网络标识等信息, 通过资源占用干扰网络的正常运行, 或通过路由劫持改变节点的网络视图等. 在拜占庭环境下的攻击行为中, 攻击者可以通过操纵网络或不诚实转发的行为破坏共识. 一旦攻击者持有超过一半的算力, 就能以较大的优势获得记账权并主导区块链系统的共识, 从而发起各种类型的攻击, 如双花攻击和自私挖矿等.

不同的攻击场景和攻击目标导致攻击方式多样, 区块链网络安全及其防御技术尚未成熟[50]. 区块链网络攻击主要依靠网络固有的安全问题以及区块链的实现机制, 构建更系统的防御体系以实现更可靠的通信环境是区块链网络研究面临的一大技术难题.

(2) 可扩展性

区块链可扩展性是限制区块链技术大规模应用开发的主要因素[51]. 根据不可能三角, 区块链系统至多只能满足去中心化、安全性和可扩展性中的两个属性. 现已提出许多提高可扩展性的解决方案, 如侧链[52]、分片[53]和 DAG (directed acyclic graph)[54]等, 但需要在去中心化和安全性之间做出权衡. 例如, 侧链技术利用支付通道技术将链上交易转移到链下执行提升吞吐量, 但由于链下部分采用传统的中心化分布式系统, 削弱了去中心化的特性. 分片技术和 DAG 技术利用并行处理提高交易速度和网络吞吐量, 但存在一定的安全性风险.

在考虑去中心化和安全性的前提下, 区块链可扩展性面临的挑战主要来源于 3 个方面: 网络传输性能、账本一致性和节点性能限制. 从网络传输性能角度, 优化底层数据传输协议可以从根本上解决可扩展性问题, 既使效率提升不受限于单机性能, 又不改变区块链的基础架构. 具体而言, 网络传播延迟和带宽利用是影响区块链网络传输性能的两个重要因素, 在下文进行详细描述. 目前区块链网络传输协议仍有很大的优化空间, 值得研究学者进一步探索.

(3) 传播延迟

对于任何去中心化平台, 网络延迟是一个不确定因素. 区块链网络传播延迟通常指交易或区块到达网络中大多数节点所需的平均时间, 快速传播有助于加快交易被包含在区块中并得到确认. 传播延迟应保持在合理的小范围内. 较大的传播延迟可能造成区块链分叉, 为攻击者执行双花和隐藏区块等攻击创造时间条件, 危及系统安全.

传播延迟与区块大小、拓扑结构、广播算法和网络规模等密切相关. 在区块大小方面, 小区块的确认延迟更小, 但带宽利用率远不及大区块. 在拓扑结构方面, 随机拓扑结构无法确定延迟性能; 结构化拓扑的传输优势可以改善延迟时间, 但构建成本和维护难度随着网络规模的扩大而增加. 在广播算法方面, 先验证后转发的模式使消息在传播路径的每个节点处都将产生验证延迟, 这进一步加剧了传播延迟, 尤其是对于连通性较好或规模较大的网络. 受物理网络带宽和链路延迟等自身条件的限制, 无论上层协议如何设计, 底层网络的传播延迟始终制约着系统性能.

(4) 带宽利用

对于区块链和其他分布式应用而言, 最小化数据同步所需的网络带宽是其基本需求. 在有限的网络资源下, 高效的带宽利用有助于降低交易和区块的同步时间, 提高网络传输效率. 此外, 降低带宽消耗可以提高节点的参与度, 尤其是能力有限的节点.

现有的降低区块链网络带宽利用的方法主要有两类. 第 1 类是设计编码方案, 降低单个数据传输的网络需求, 提高有效数据的传输量. 该方法引入了额外的工作且牺牲了一部分准确性, 甚至可能破坏系统安全. 第 2 类是设计广播算法, 降低冗余数据传输. 目前大多数区块链采用基于泛洪的广播算法, 大量消息重复传输会造成资源浪费. 随着网络连接性的增加, 带宽消耗也将线性增加. 受共识协议速度限制, 目前带宽造成的影响并不明显. 但区块链技术正在不断发展与成熟, 有限的网络资源会限制系统性能. 合理分配和高效利用带宽是进一步提升性能的重要

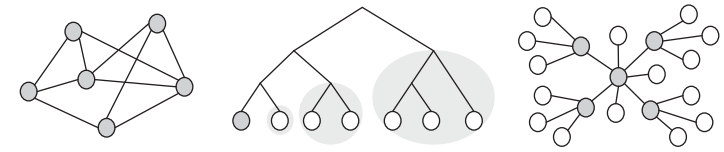
技术手段.

1.5 小 结

本节说明区块链网络的传输流程, 并通过与传统 P2P 网络进行对比分析, 说明区块链网络面临的挑战. 针对这些挑战, 研究学者提出一些网络层优化方案以提高区块链系统性能. 本文将关注重点放在拓扑结构和传输协议上, 分别在第 2 节和第 3 节中详细阐述.

2 区块链网络拓扑结构

拓扑构造是网络研究和实践的核心内容之一, 是实现信息交互和路由驱动的前提. 区块链网络拓扑结构包括分布式非结构化、分布式结构化和半分布式拓扑, 如图 2 所示. 本节从拓扑结构特点、代表项目以及现有的拓扑设计对区块链网络拓扑进行详细说明.



(a) 分布式非结构化拓扑 (b) 分布式结构化拓扑 (c) 半分布式拓扑

图 2 区块链网络拓扑结构图

2.1 分布式非结构化拓扑

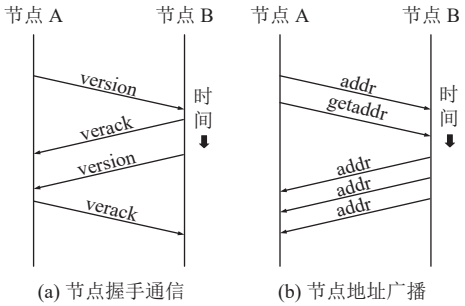
2.1.1 特 点

分布式非结构化拓扑移除了中心节点, 节点连接呈随机状态, 即新节点随机选择其他节点建立连接通道, 形成无结构化的随机拓扑. 该结构实现简单, 易于维护, 具有良好的匿名性和容错性. 然而, 随机性使系统性能无法保证. 随着网络规模不断扩大, 当发生消息风暴时, 非结构化拓扑网络可能瞬间瘫痪, 可扩展性差.

2.1.2 比特币

比特币开启了区块链时代, 旨在建立一个去中心化的点对点支付系统. 比特币网络采用随机拓扑结构, 节点通常使用 TCP 协议、8333 端口与已知的邻居节点连接通信. 默认情况下, 每个节点最多保持 125 个连接, 包括最多 8 个传出节点和最多 117 个传入连接[55].

比特币网络连接建立流程如图 3 所示. 新节点启动后, 节点利用发现协议获得网络中其他节点的地址. 建立连接时, 节点发送一个包括基本认证内容的 version 消息开始握手通信, 并对响应的 verack 消息进行确认. 如果接收节点需要连回起始节点, 也会传回该节点的 version 消息 (见图 3(a)). 节点接入网络后, 新节点将一条包含自身 IP 地址的消息发送给邻居节点, 邻居节点再将此消息依次转发给各自的邻居节点以便被更多节点接受. 新节点还可以向邻居节点发送消息, 请求其返回已知的节点地址列表 (见图 3(b)). 通过这种方式, 新节点可以向网络发布自身信息, 保证连接的稳定性.



(a) 节点握手通信 (b) 节点地址广播

图 3 比特币网络连接建立流程图

比特币采用地址数据库保存并维护节点. 如果已建立的连接没有数据通信, 所有节点会定期发送探活消息以维持连接. 如果节点长达 90 min 没有任何通信, 则被认为从网络中断开, 随机查找新节点构建新连接. 比特币网络将发现的节点信息和断开的节点信息分别保存在不同的本地文件中. 节点再次启动时, 默认与上次连接成功的节点通信. 如果节点没有收到任何响应, 则使用种子节点进行重启动.

2.2 分布式结构化拓扑

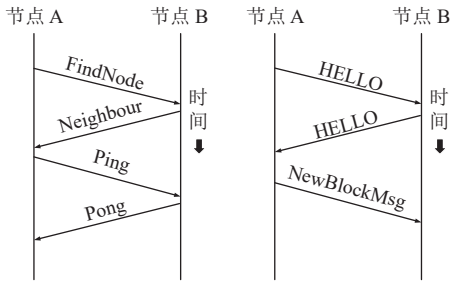
2.2.1 特　点

分布式结构化拓扑中节点的连接遵循一定的规则, 主要采用分布式哈希表 (distributed hash table, DHT) 技术. 该结构能够自适应节点的动态变化, 具有良好的可扩展性和鲁棒性. 通过确定性拓扑结构, DHT 可以提供精准定位节点以实现正确查询, 系统的可靠性得到保证. 然而, 结构化拓扑去中心化程度低, 拓扑构建和维护的成本随网络规模显著增加, 无法适应高频的节点变化.

2.2.2 以太坊

与比特币不同, 以太坊的目的是建立一个去中心化的、永不停机的世界计算机. 以太坊采用类 Kademlia 算法实现结构化 P2P 网络, 节点发现使用未加密的 UDP 协议[56], 数据传输使用加密的 TCP 协议[57]. 以太坊网络的特点是快速准确地查找地址. 节点在首次启动时, 随机生成固定且唯一的节点 ID 用于身份标识和资源定位. 路由表使用 K 桶构造, 用于记录已知的邻居节点信息, 包括节点 ID、距离、端点和 IP 等.

以太坊网络连接建立流程如图 4 所示. 新节点利用发现协议加入网络后, 节点不主动广播自身地址, 而是通过 FindNode 命令向其他节点查询与目标节点距离接近的节点, 收到 Neighbour 响应后建立双向通信. 若节点 Ping-Pong 握手通过, 则认为节点在线, 比较距离后将收到的节点存入 K 桶 (见图 4(a)). 节点通信协议与节点发现协议并行运行, 负责节点间数据传输和交互. 一旦节点连接成功, 节点交换 HELLO 消息, 检查节点是否运行相同的应用层协议、是否支持彼此的版本以及是否达成一致. 如果所有条件成立, 节点可以进行信息交换和传输, 否则将断开连接 (见图 4(b)).



(a) 节点发现协议　　(b) 节点通信协议

图 4　以太坊网络连接建立流程图

以太坊使用 K 桶实现邻居维护. 节点定期随机选择一个桶, 向其末尾节点发送探活消息. 如果节点发出响应, 则探活成功, 否则将被替换. 此外, 系统每隔 7200 ms 刷新一次 K 桶, 每次刷新所查找的节点均在距离上向随机生成的目标节点收敛.

2.3 半分布式拓扑

2.3.1 特　点

半分布式拓扑根据 CPU、内存、带宽等资源, 将节点分为超级节点和普通节点. 整个网络可以看作层次式结构, 第 1 层是超级节点构成的类似随机拓扑结构, 第 2 层是每个超级节点和若干个普通节点构成的中心化拓扑. 该结构结合中心化和分布式结构的优点, 能够根据节点能力和资源状况合理决定节点地位, 具有对异构网络的高度适应性. 然而, 半分布式拓扑对超级节点的依赖性较大, 系统安全性和容错性易受影响.

2.3.2 超级账本 Fabric

超级账本 Fabric 是为企业级分布式应用设计的操作系统, 旨在实现一个通用的许可区块链底层框架. Fabric 采用许可制, 参与者必须注册身份并通过审核才能加入网络, 不是完全无信任的和匿名的. Fabric 网络节点具有不同的身份, 通过节点功能和组织机构的划分提高了网络效率和系统性能, 适合在企业内和企业间构建联盟链.

Fabric 网络组件和交易传输流程如图 5 所示. 客户端节点发布交易之前, 需要获取 CA (certification authority) 节点签发的证书, 凭借该证书在网络中进行交易. 发起交易后, 客户端将交易提案发送给一组背书节点, 背书节点的数量和来源由背书策略决定. 背书节点模拟执行并将结果和签名返回客户端. 客户端收集足够数量的背书后将交易发送到排序节点. 排序节点对所有交易进行排序, 按照规则打包成新区块并广播到各组织的主节点. 主节点负责在组织内部同步区块, 每个组织中只能有一个. 所有节点都是记账节点, 负责在本地检验交易和区块的合法性, 更新并维护区块链账本状态. 锚节点代表组织和其他组织进行信息交换, 可以被同一通道的其他任何节点发现和通信.

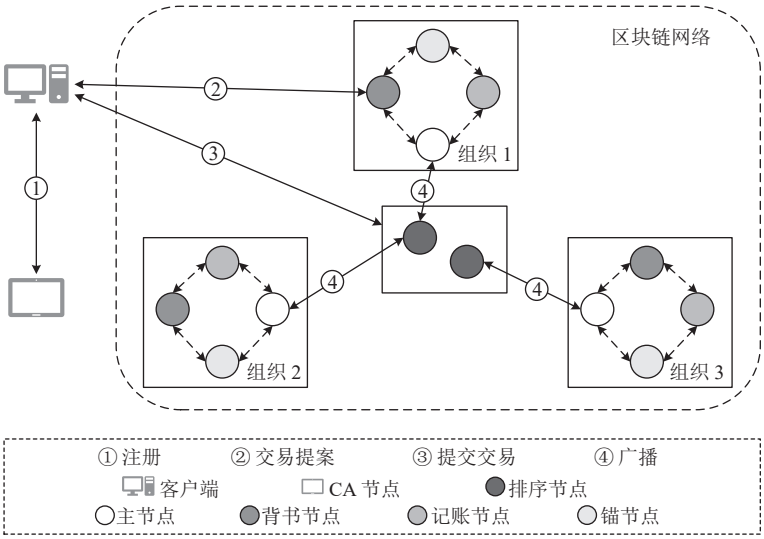


图 5 Fabric 网络组件和交易传输流程图

新节点加入网络前, Fabric 通过读取配置文件得到一个可连接的节点列表. 节点向列表中所有节点发送 MembershipRequest 消息. 收到消息的节点验证消息的准确性, 无误后将其添加到存活节点列表中, 更新检测到的存活时间. 之后节点定时发起节点查找任务广播地址, 在节点列表中随机选择某些节点, 请求存活节点列表信息并同步列表, 同时将自身信息发送给其他节点. 所有节点周期性地维护节点列表, 如有超时未响应节点则移出到离线节点列表, 并不断尝试连接.

2.4 拓扑设计

比特币网络随机选择邻居通信, 节点之间的传播延迟无法衡量. 交易和区块在传播过程中会受到各种因素影响, 例如链路延迟、验证延迟等, 但主要因素还是底层网络的拓扑结构. 研究表明, 增加网络连接性可以使网络更加健壮[58]. 本节从节点同构和节点异构两方面梳理现有的拓扑设计.

2.4.1 面向同构节点的拓扑设计

BCBSN (bitcoin clustering based super node) 协议[59]根据节点的信誉度选取超级节点, 各超级节点与地理位置相近的节点聚类, 负责维护整个集群. 所有超级节点相互连接, 普通节点仅与一个超级节点连接. 结果表明, BCBSN 协议中传播延迟的降低与经过的路由跳数最小化相关. BCBSN 协议没有考虑通信链路距离因素的影响, 性能提升并不显著.

LBC (locality based clustering) 协议[60]根据节点的 IP 地址计算节点之间的地理距离, 将地理距离接近的节点连接分组, 增强网络连接的局部性. 结果表明, LBC 协议中传播延迟的降低与节点地理位置相关. 与 BCBSN 协议

相比, 随着网络连接性的增加, LBC 协议传播延迟增长幅度较小. 但是, 覆盖网络中地理位置相近的节点在真实的物理网络中可能相距很远, LBC 协议未考虑真实物理网络的通信情况.

BCBPT (bitcoin clustering based ping time) 协议[61]使用节点之间的 Ping 延迟判断节点的邻近度, 将物理网络距离相近的节点聚类, 降低实际通信链路的开销. 结果表明, BCBPT 协议中传播延迟的降低与节点物理网络距离相关, 物理网络距离阈值越小, 传播延迟越小. 与 BCBSN 和 LBC 协议相比, BCBPT 协议更有效地降低了传播延迟. 但是由于节点需要反复计算与其他所有节点的网络距离, 计算开销较大.

BlockP2P 协议[62]设计了一种高连通和小直径的结构化拓扑结构, 利用 K-means 聚类算法和网络坐标系划分集群. 在数据传输时, 各集群的路由节点负责集群内和集群间的数据转发, 利用并行化广播加快传播速度. 结果表明, 与比特币相比, BlockP2P 在区块同步达到 90% 时将区块传播延迟降低了 90%. BlockP2P 协议能够感应网络变化并快速重建集群, 网络的稳定性和可扩展性更好.

同构网络节点表现出相同的功能和效用, 拓扑设计简单, 结构健壮. 面向同构节点的拓扑设计较为理想, 与实际的网络环境不符. 例如, 除链路延迟、地理位置等显示因素, 节点自身的处理能力和计算资源也是影响传播延迟的关键. 为了真实地反映网络状况, 节点的异构性不容忽视.

2.4.2 面向异构节点的拓扑设计

Perigee 协议[63]是一种面向异构节点的拓扑设计方案, 仅基于节点与邻居的交互作用自适应调整到适合当前网络的最佳拓扑. 受多臂匪徒问题的启发, 节点根据邻居评分算法定期评估当前邻居集, 在保留良好连通性的旧邻居与探索潜在更优连通性的新邻居之间权衡取舍. 邻居的评分越低, 节点在下一轮保留该节点的偏好越高.

Perigee 节点通过查看区块到达时间来量化与邻居节点的交互. 该方法自动考虑了链路延迟、验证延迟和节点带宽异构性等因素, 使高性能的节点具有更优的连通性. 由于节点很难获取区块生成及传输到邻居的准确时间, 节点使用区块被不同邻居转发的到达时间的相对时间差表示区块的传播延迟. 相对时间差集合计算为:

$$\widetilde{O}_v=\left\{\left(b,u,t^b_{u,v}-t^b_v\right):u\in\Gamma_v,b\in B\right\} \tag{1}$$

$$\widetilde{T}_{u,v}:=\left(\widetilde{t}:\left(b,u,\widetilde{t}\right)\in\widetilde{O}_v\right) \tag{2}$$

其中, Γ_v 表示节点 v 的邻居集合, B 表示一轮中广播的区块集合, t^b_v 表示节点 v 首次收到区块 b 的时间, $t^b_{u,v}$ 表示节点 v 从邻居 u 收到区块 b 的时间. 邻居 u 的评分计算为 $90percentile\left(\widetilde{T}_{u,v}\right)$. 为了避免挖矿随机性和样本不足导致的意外断开, Perigee 利用过去可用的相对时间重新计算各邻居的评分和置信区间, 根据所有邻居置信区间的上下限判断需要断开的邻居. 这种情况下, 相对时间差集合计算为:

$$\widetilde{\widetilde{T}}_{u,v}:=\left(\widetilde{t}:\left(b,u,\widetilde{t}\right)\in\bigcup_{i=0}^{-r_{u,v}}\widetilde{T}_{u,v}(-i),\widetilde{t}<\infty\right) \tag{3}$$

其中, $r_{u,v}$ 表示节点 u 是节点 v 邻居的轮次数, $\widetilde{T}_{u,v}(-i)$ 表示当前轮次的前 i 轮的相对时间集合. 邻居 u 的评分计算为 $90percentile\left(\widetilde{\widetilde{T}}_{u,v}\right)$. 邻居 u 的置信区间的上限 ucb 和下限 lcb 分别计算为:

$$ucb(u)=90percentile\left(\widetilde{\widetilde{T}}_{u,v}\right)+c\sqrt{\frac{\log\left(\left|\widetilde{\widetilde{T}}_{u,v}\right|\right)}{2\times\left|\widetilde{\widetilde{T}}_{u,v}\right|}} \tag{4}$$

$$lcb(u)=90percentile\left(\widetilde{\widetilde{T}}_{u,v}\right)-c\sqrt{\frac{\log\left(\left|\widetilde{\widetilde{T}}_{u,v}\right|\right)}{2\times\left|\widetilde{\widetilde{T}}_{u,v}\right|}} \tag{5}$$

为了减少评估所有邻居的计算开销, Perigee 协议还设计了邻居组评分规则, 逐个选择要保留的邻居. 首先, 在相对时间 $\widetilde{T}_{u,v}$ 中选择一个最佳的邻居 u_1. 假设已选择 k 个邻居 $\{u_1,u_2,\ldots,u_k\}$, 对于其他邻居 u 计算邻居组的相对时间差集合.

$$\widetilde{\widetilde{O}}_v(u_1,u_2,\ldots,u_k)=\left\{\left(b,u,\min\left(\widetilde{t}^b_{u,v}-\min_{1\leqslant i\leqslant k}\widetilde{t}^b_{u_i,v}\right)\right):u\in\Gamma^o_v\backslash\{u_1,u_2,\ldots,u_k\},b\in B\right\} \tag{6}$$

$$\widetilde{\widetilde{T}}_{u,v}(u_1,u_2,\ldots,u_k)=\left(\widetilde{\widetilde{t}}:\left(b,u,\widetilde{\widetilde{t}}\right)\in\widetilde{\widetilde{O}}_v(u_1,u_2,\ldots,u_k)\right) \tag{7}$$

其中, Γ_v^o 表示节点 v 的出度邻居集合. 最后选择 $90percentile\left(\widetilde{\widetilde{T}}_{u,v}\right)$ 最佳的节点作为第 $(k+1)$ 个邻居. 一旦邻居数量达到预设值, 节点还随机选择少量邻居来探索更优的连接.

结果表明, 与随机选取邻居相比, Perigee 邻居组算法和置信区间算法分别将传播延迟降低了 33% 和 11%. 该协议能够在各种设置下实现全局优化拓扑. 对于较小的区块处理延迟, 协议能够发现性能优于随机延迟的拓扑. 随着处理延迟的增加, 性能逐渐接近随机拓扑, 与最短路径的节点数相关. 对于矿池和低延迟网络的模拟, 性能接近于理想情况, 即完全拓扑连接中区块传播时间的理论下限.

异构网络节点的功能和效用不同, 拓扑设计复杂, 实现难度较大. 随着节点硬件设施和网络架构的不断升级, 节点的计算能力、存储空间和网络带宽等资源也在不断变化. 面向异构节点的拓扑设计能够融合更多的节点信息, 感应节点变化并动态调整拓扑结构更能满足实际网络需求.

2.4.3 安全分析

拓扑分区型区块链具有良好的可扩展性, 系统的整体性能随着分区数量的增加而提高. 但是当遵循一定的规则聚类拓扑时, 系统的抗攻击能力也被分解. 首先, 攻击者可能通过某种手段将恶意节点集中在一个集群中以低成本发起网络攻击. 例如, LBC 协议根据节点的 IP 地址获取节点的位置信息, 攻击者可以利用 VPN、代理服务器等进行地理位置欺骗以影响聚类结果. 其次, 分区型区块链为分割攻击[64]创造了条件, 攻击者很容易将网络划分成多个不相交的子网络以实现区块链分叉并从中获利.

Perigee 协议始终保持一个随机选择的邻居子集, 网络更加健壮. 但是如果攻击者总是比其他节点更早地提供区块, 攻击者可以获得节点的信任并控制其邻居. Perigee 协议的最佳拓扑结构是逐步形成的, 如果在此过程中发生节点搅动和高频变化现象, 协议的性能及拓扑收敛情况尚未可知.

表 2 对以上讨论的拓扑设计进行总结. 除 Perigee 协议以外, 上述协议利用节点的显示属性和聚类算法设计拓扑. 目前仅存在 BCBSN、LBC 和 BCBPT 协议的性能对比工作, BlockP2P 和 Perigee 协议与比特币随机拓扑对比. 目前, 区块链网络拓扑设计在研究工作中备受关注, 但缺乏实际的系统应用, 安全分析和对比工作较少.

表 2 区块链网络拓扑优化协议

协议	传输协议		性能指标		安全性
	拓扑结构	邻居选取规则	影响因素	传播延迟	
BCBSN[59]	半分布式拓扑	超级节点和地理距离	路由跳数	略微降低	依赖超级节点
LBC[60]	结构化拓扑	地理距离	链路延迟	优于BCBSN协议	地理位置欺骗
BCBPT[61]	结构化拓扑	物理网络距离	网络延迟	优于LBC协议	高频节点变化
BlockP2P[62]	结构化拓扑	K-means和网络坐标系	拓扑和广播	与比特币相比, 降低90%	网络干扰问题
Perigee[63]	动态调整拓扑	区块到达时间	节点异构性	与随机拓扑相比, 降低33%	节点搅动现象

2.5 小 结

本节以区块链网络拓扑结构为基准, 说明 3 大主流区块链系统的网络连接建立流程, 梳理拓扑设计的现有工作. 拓扑结构对区块链系统的安全性至关重要, 攻击者可以通过网络窃听监测节点的连接状态, 获取节点的隐私和路由信息以实施攻击. 拓扑设计需要适合分散的网络环境, 除安全性外, 还要考虑网络复杂性、维护难度和可扩展性等因素.

3 区块链网络传输协议

区块链网络传输协议的主要目标是以速度更快、带宽密度更低的方式实现数据在全网节点的一致性同步更新. 比特币数据传输流程如图 6 所示. 为了避免重复传输, 节点首先验证收到的交易或区块, 通过后发送 INV 消息(包含交易或区块的哈希值) 到邻居节点. 缺少这些交易或区块的邻居发送 Getdata 消息获取完整的数据, 使用哈希

值进行确认; 否则将忽略 INV 消息[58]. 大量研究表明, 该传播方式存在资源浪费、数据冗余和效率低下等缺陷. 本节从编码方案、广播算法和中继网络这 3 方面对现有的数据传输协议进行全面梳理.

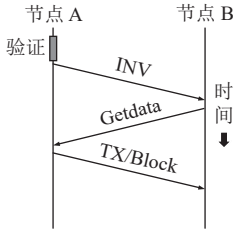


图 6 比特币数据传输流程图

3.1 编码方案

3.1.1 基于交易中继的编码方案

比特币最初提出交易哈希公告方法解决交易中继的冗余问题, 但交易公告仍占据着总带宽的 30%−50%. Erlay[12]是针对比特币交易中继提出的新协议, 利用低扇出泛洪和集对账方法尽可能降低与交易公告相关的带宽. 在低扇出泛洪中, 节点仅向一部分邻居节点公告交易. 为了确保交易到达整个网络, 节点根据选定的对账计划定期与所有出度节点进行集对账, 以发现缺失的交易. 协议规定, 只有连接性良好的公共节点才能通过出度连接中继交易, 而私有节点仅在集对账过程中传递交易, 即将自身的交易添加到转发给其他节点的交易中一起转发. 低扇出泛洪将交易中继到每个节点的少量跳数内, 便于后续的集对账操作能够在少量轮次内完成.

Erlay 数据传输流程如图 7 所示. 节点首先发送自身交易集的大小和系数 q 到出度节点, 其中 q 反映上一轮对账的特征并在本轮对账结束后根据真实的差异值更新. 出度节点利用双方交易集大小和系数 q 估算交易集的差异值 d, 并返回一个交易草图 $sketch(N, d)$. 最后节点计算自身的交易草图 $sketch(M, d)$, 通过对两个草图进行异或操作得到交易集的差异 (见图 7(a)). 如果 d 预估准确, 草图解码成功, 进入交换过程 (见图 7(b)). 节点请求自身没有的交易, 发送出度节点丢失的交易. 如果 d 预估错误, 草图解码失败, 节点采用二分法重新计算交易差异, 最多可恢复 $2d$ 个交易差异 (见图 7(c)). 该方法重用了之前的传输信息, 避免重新计算造成的带宽浪费和低效现象. 如果二分法也失败, Erlay 将回退到原始的 INV-Getdata 模式获取交易数据. 由此可见, 集对账过程依赖于估算值的准确性, 过低会导致解码失败, 过高则会引入带宽消耗.

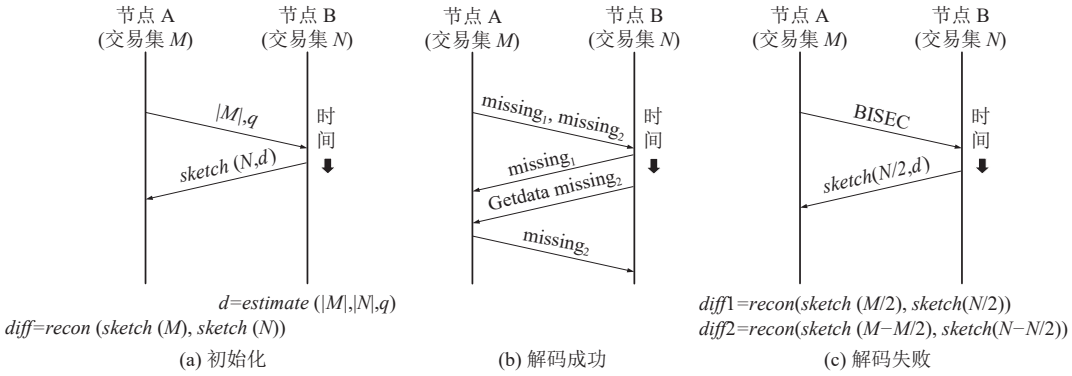


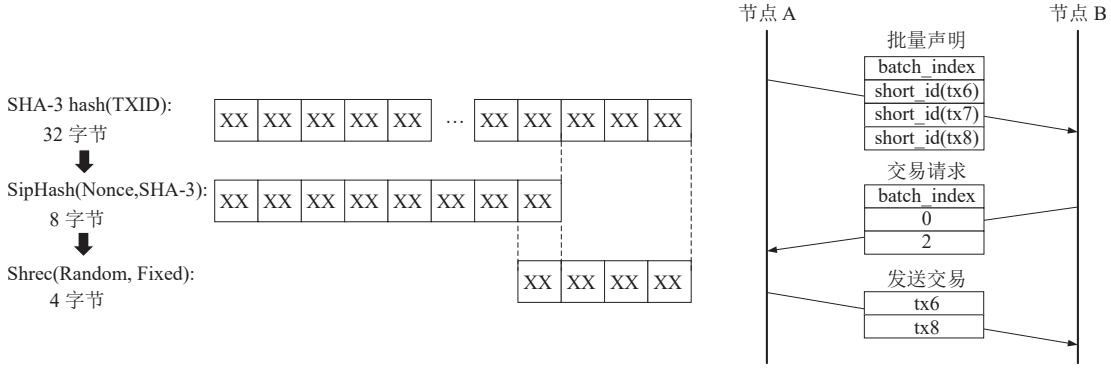
图 7 Erlay 数据传输流程图

Erlay 旨在不牺牲安全性的情况下平衡低延迟和低带宽, 即使节点的连接数量增加, 带宽几乎保持不变. 虽然该协议降低了交易公告所需的网络流量和节点运行的压力, 但由于交易的批处理传播延迟有所增加. 此外, 集对账协议的计算成本很高. 对于相同的对账周期, 更高的交易率将导致对账双方收到更多交易. 如果两个交易集之间的差异较大, 解码成功的概率低.

Shrec[13]是为高吞吐量区块链系统设计的交易中继协议, 通过混合哈希编码和低扇出泛洪实现高传输效率和

强安全性能, 同时保证合理的传播延迟和计算成本. 每个节点维护一个接收池来记录接收到的交易公告、一个飞行池来记录未完成的交易请求和一个发送池来缓存已发出公告的交易. 接收池和飞行池利用滑动时间窗口进行管理, 允许系统丢弃接收时间很久的信息.

Shrec 数据传输流程如图 8 所示. Shrec 结合 SHA-3 哈希[65]和 SipHash[66]的值, 为每个交易构造 4 字节的短 ID, 其中 3 个固定字节来自 SHA-3 哈希, 1 个随机字节来自 SipHash (见图 8(a)). 为了高效地利用带宽, Shrec 缓存新收到的交易, 定期将交易的短 ID 批处理公告并广播到一组随机节点. 收到交易公告的节点在接收池和飞行池中进行交易匹配以减少冗余交易传输. 当发生冲突时, 节点不会立即认为已经请求该交易, 而是等待结果并重构交易编码. 如果两个编码结果相同则交易已收到, 否则节点发出交易请求. 最后, 接受方以批处理的方式返回所有交易请求, 包括批次索引以及未匹配交易的偏移量. 同样的, 发送方以批处理方式发送所有请求交易 (见图 8(b)).



(a) 交易短 ID 编码方案　　(b) 节点交互流程

图 8　Shrec 数据传输流程图

在有限的网络带宽限制下, Shrec 的性能优于所有的替代方案, 如 BTCFlood、SipHash 和 Erlay 等. BTCFlood 和 SipHash 基于泛洪广播, 其性能明显受限于网络带宽. 当可用带宽增加时, 吞吐量和带宽消耗也线性增加. Erlay 协议依赖于集对账, 受 CPU 的限制无法充分利用网络带宽, 因此即使增加带宽其性能也不会提高. Shrec 协议在不同的带宽限制和扇出设置下表现出相似的性能, 系统受限于交易执行的吞吐量. 在相同吞吐量下, 减少扇出数量可以降低带宽消耗, 但会相应增加传播延迟. 与交易确认时间相比, 延迟增量几乎可以忽略不计.

3.1.2　基于区块中继的编码方案

比特币中新创建的区块包含的大多数交易已经存储在大部分全节点的内存池中, 直接中继整个区块可能会导致节点传输的带宽峰值. 瘦块策略旨在利用请求方内存池中已经存在的交易来重构区块, 而不是传输整个区块. 但是由于网络故障、交易传播故障等因素, 各节点内存池并不完全同步, 发送方需要知道接收方内存池中缺少哪些交易以实现快速重构区块.

XThin (xtreme thinblocks)[14]采用简单的布隆过滤器 (Bloom filter, BF) 和交易往返策略来重构区块. XThin 数据传输流程如图 9 所示. 当接收方请求一个区块时, 首先接受方以自身内存池中的交易为种子构建一个布隆过滤器, 并与 Getdata 消息一起发送给发送方. 之后发送方返回一个瘦区块, 其中包含区块头、区块中包含的所有交易哈希以及任何与布隆过滤器不匹配的交易. 最后接收方根据这些信息重构区块. 如果仍缺少交易, 则通过 CThinBlockTx 直接请求丢失的交易, 通常只需要一次往返即可完成区块. XThin 技术将交易哈希缩短为 8 字节, 即 1 MB 的区块可以缩小为 10–25 KB 的瘦区块, 同时能够避免内存池中的交易冲突.

受 XThin 的启发, Xthinner[15]利用 LTOR 技术将区块中的交易以数字或字典顺序排序并进行熵编码, 使其能够唯一地标识交易即可. 接收方收到后进行解码操作, 在内存池中查找相同编码前缀的交易并只希望找到一个, 之后将其附加到区块中. Xthinner 优化 99.6% 的区块空间, 传播效率进一步提高.

Compact Block[16]协议不借助额外的数据结构, 根据节点和带宽的使用情况, 设计了两种数据传输模式.

Compact Block 数据传输流程如图 10 所示. 在高带宽模式下, 发送方可能在区块验证完成之前, 直接发送新区块公告 (见图 10(a)). 该模式只为少数节点启用. 在低带宽模式下, 区块验证完成后, 接收方通过 Getdata 消息获取新区块. 若仍缺少交易, 则在后续的操作中单独请求交易 (见图 10(b)). 相较于 XThin, Compact Block 进一步将交易哈希缩短为 6 字节, 网络成本较低; 但是请求缺失的交易将增加往返时间, 如果缺失交易过多其性能可能会变差.

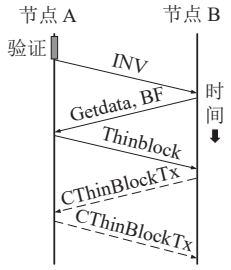


图 9 XThin 数据传输流程图

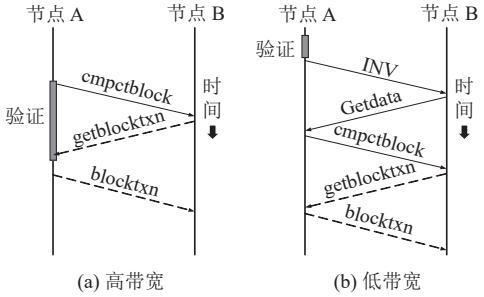


图 10 Compact Block 数据传输流程图

Txilm[17]协议使用短交易哈希和规范交易排序规则 (canonical transaction ordering rule, CTOR) 将交易压缩为 4 字节, 区块传输带宽需求降低到原来的 1/80. 当使用短哈希进行交易搜索时, 可能找到多个匹配的交易. 这时接收方收集所有匹配的交易编码, 计算 Merkle 根以寻找正确的交易. 如果未找到, 接收方将回退到原始算法, 请求完整的交易 ID 列表. Txilm 协议解决交易的二义性可能产生额外延迟, 消耗大量的 CPU 时间, 与发生哈希碰撞的概率相关. 通过 CTOR 技术, 区块和内存池中的交易根据交易哈希排序, 实现了更低的碰撞率和更高的压缩率.

Graphene[18]是一种节点间进行交互式集对账的协议, 通过布隆过滤器和可逆布隆查找表 (invertible Bloom lookup table, IBLT) 的新颖组合, 将数据传输所需的网络带宽的一部分用于单向或双向同步. 由于布隆过滤器和 IBLT 是概率型数据结构, 因此 Graphene 是一种概率型编码方案, 接收方解码 IBLT 的能力取决于发送方编码 IBLT 的大小、发送方与接收方的内存池之间的同步量以及一定程度的运气. 为了避免出错造成的资源浪费, Graphene 提出了高效的搜索算法参数化布隆过滤器和 IBLT, 确保高可调的解码成功率, 并能够在传输数据大小、传输复杂性和解码成功率之间进行权衡.

Graphene 数据传输流程如图 11 所示. 当接收方请求区块时, 发送方以区块中所有交易为种子构建 BF_A 和 $IBLT_A$, 并发送到接收方. 之后接收方使用 BF_A 测试内存池中的所有交易, 得到交易列表 S. 由于布隆过滤器存在假阳性, 因此列表中的交易数量可能会多于实际数量. 鉴于此, 接收方构造该交易列表的 $IBLT_S$, 并与发送方发送的 $IBLT_A$ 进行解码, 识别误报的交易, 重构区块 (见图 11(a)). 但是在实际的网络条件下, 区块中的某些交易可能尚未包含在内存池中, IBLT 将无法解码. 这时接收方以交易列表 S 为种子构造 BF_S. 之后发送方使用 BF_S 测试区块中的所有交易, 并将不在 BF_S 中的交易 (即交易列表 P) 和 $IBLT_A$ 直接发送给接收方. 接收方以交易列表 P 和 S 为种子构造 $IBLT_{S+P}$, 与 $IBLT_A$ 解码, 识别误报和缺失的交易 (见图 11(b)). 最后, 接收方以正确的交易顺序重构区块.

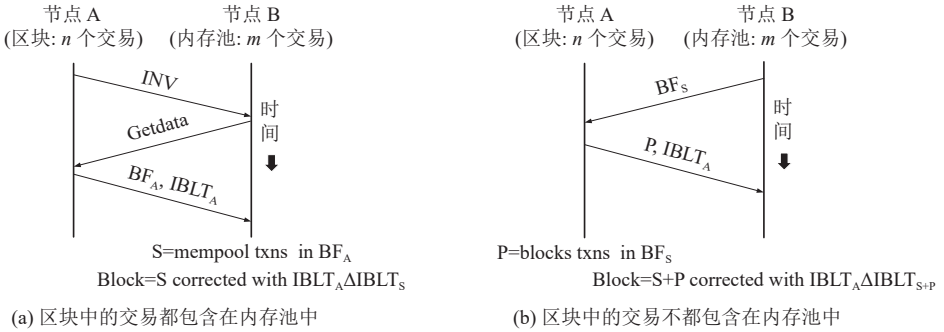


图 11 Graphene 数据传输流程图

与 XThin 和 Compact Block 相比, 无论接收方是否丢失交易, Graphene 在同步区块时消耗的带宽较低, 随着区块大小增加成本增长缓慢. 但是如果协议解码失败, 节点需要使用额外的往返时间请求丢失的交易. 在恶意矿工产生包含大量尚未广播的交易的区块的情况下, Graphene 协议可能无法解码. 在 BCH 压力测试中显示, Graphene 使用布隆过滤器和 IBLT 传递与交易顺序相关的信息, 在传输区块时有 86% 的数据用于编码交易顺序[67], 严重加大了传输成本. 如果结合 CTOR 技术, 其压缩性能将进一步提高.

3.1.3 安全分析

编码方案采用短标识符的一个主要问题是哈希冲突, 攻击者很容易构造与已有短交易哈希相匹配的新交易以攻击网络, 并且节点不会因此而受到惩罚. 交易冲突可能降低交易的广播和确认速度. 为了防御冲突攻击, 一种常用的方法是在计算编码时引入随机数, 同时也带来新的安全隐患.

Txilm 协议在计算短标识符时引入盐 (Salt), 该盐值特定于携带这些交易的区块并被包含在编码数据中, 如 Merkle 树根作为盐, 或随机生成一个 4 字节字段. 即使交易已知, 攻击者很难在区块被矿工广播之前构造恶意交易并将其快速传播到全网. 一旦区块被确认, 该恶意交易也将失效, 这使得攻击者发起攻击的时间非常受限, 经济效益也十分低效. 加盐短哈希显著提高了攻击难度和成本, 但如果攻击者不计成本, 大规模碰撞攻击仍有可能发生. 接收方可以请求原始交易列表以避免创建孤块造成浪费算力.

Erlay 协议也采用加盐短哈希, 每对节点使用不同的盐计算短哈希. 该盐值为添加在哈希函数中的随机数, 由发起连接的节点确定. 在不同的集对账过程中, 请求连接的节点可以使用同一盐值将加盐短哈希与泛洪过程中从其他节点收到的交易哈希进行对比, 以避免同一交易的重复传输. 此外, Erlay 规定只通过出度连接中继交易的方式可以防御时序攻击[68]. 在时序攻击中, 攻击者可以连接到某个节点并监听该节点入站连接中发送的所有交易. 如果攻击者从多个节点收到交易, 通过交易的到达时间可以猜测交易的发起者. 比特币通过引入泛洪延迟实现这一点, 而 Erlay 协议并不需要.

随机数的引入使交易冲突独立于不同的对等节点, 但仍然无法避免同一交易的冗余传输. Shrec 协议采用混合哈希编码方式, 在每对节点传输交易公告之前确定一个随机数作为 SipHash 函数的输入, 并与 SHA-3 一起计算交易编码. 不同节点对对同一交易的编码结果可能不同. 由于交易只有在所有对等节点上发送冲突才会被阻止传播, 因此冲突率随着对等节点数量的增加呈指数下降. 然而, 仅依靠编码方案仍不能完全避免哈希冲突攻击. 例如攻击者在离线状态下按交易的固定字节对所有交易进行分组并伪造交易, 只要分组中的交易足够多, 冲突的概率就越大. 当攻击者在线观察到一笔交易时, 迅速将具有相同固定字节的交易组广播到全网以达到攻击特定交易的目的. 针对这种情况, Shrec 设置了一个阈值来表示接受池中具有相同固定字节交易的数量上限. 只有当数量少于该阈值时, 才使用交易短哈希检查交易, 否则请求交易的完整 SHA-3 哈希. Shrec 协议利用混合编码缓解网络带宽瓶颈, 为区块链系统提供了更高的安全保障.

Graphene 是一种概率性协议, 通过高效的参数化算法和加强的 IBLT 性能实现高解码成功率和安全性. 然而, 生成一个 IBLT 很简单, 这也为攻击者创造条件. 例如, 攻击者可以创建错误格式的 IBLT 导致解码无限循环. 同样的, 这类攻击也很容易解决: 一旦检测到 IBLT 中的项目被解码两次, 攻击将被阻止, 接收方可以丢弃或禁止发送方. 对于交易冲突, 不论是否使用 SipHash 来防御对单个节点的攻击, Grapene 协议对哈希碰撞的抵抗力比 XThin 和 Compact Block 都要高.

后文表 3 对以上讨论的编码方案进行总结. 上述协议利用不同的编码技术缩短交易和区块大小, 降低数据同步时对网络带宽的需求, 现已部署在实际的系统中. 为了缓减交易冲突, 各协议制定了不同的安全防御策略, 能够在现有的攻击模型下保持安全性和鲁棒性.

3.2 广播算法

广播涉及区块链的多个层次, 是系统性能和可靠性的基础. 现有的区块链系统如比特币采用简单的广播策略, 要求中继节点验证所有收到的交易和区块, 并将有效的交易和区块转发给所有邻居节点. 虽然该策略有助于数据同步, 但数据的冗余传输和重复验证大大降低广播速度和带宽利用率. 现有方案主要从验证转发、Gossip 协议优化和结构化广播 3 个方面加快广播进程.

表 3 区块链网络编码优化协议

协议	编码技术	交易大小 (字节)	安全防御策略	性能	部署平台
Erlay[12]	PinSketch	8	加盐短哈希、公共节点中继	带宽减少40%, 交易传播延迟增加2.6 s	Bitcoin core
Shrec[13]	混合编码	4	设置冲突阈值	带宽减少60%, 吞吐量提高90%	Conflux[69]
XThin[14]	BF	8	降低布隆过滤器误报	区块大小减小为原来的1/40到1/100	BU client
Xthinner[15]	LTOR、熵编码	不定	基于堆栈的状态机	优化99.6%的区块空间	Bitcoin cash
Compact Block[16]	SipHash	6	引入随机数	减少区块中继带宽和区块传播延迟	BU client
Txilm[17]	CTOR	4	加盐短哈希、回退原始算法	区块传输带宽降低为原来的1/80	Monoxide[70]
Graphene[18]	BF、IBLT	8 (IBLT)	参数化算法、增强IBLT性能	优于XThin和Compact Block, 成本随区块增大呈亚线性增长	Bitcoin cash (协议1)

在传统的传输模式中, 节点必须对所有接收到的交易和区块进行验证, 只有验证通过后才能进行转发. 然而, 这种模式效率低下, 一些研究工作从验证转发的角度优化交易和区块的广播过程. Santiago 等人提出了一种基于节点权重选择邻居的替代方法[19], 该方法根据发送消息和接收消息之间的时间差评估节点之间传输通道的质量, 并将测得的延迟与其他节点进行比较来分配节点权重, 以更优化的方式选择邻居. Zhang 等人设计一个基于信誉机制的中继协议 RepuLay[20], 实现概率性验证和选择性转发交易. 根据拥有的本地信息, 每个节点独立地对邻居评分并自行决定转发节点, 保证传输质量的同时加快交易的传播速度. Chen 等人引入基于信任机制的担保人角色决定是否直接验证区块, 从而减少传播并降低分叉的概率[21]. 此外, Ayinala 等人利用流水线和分块技术并行化区块的传播和验证[22], 该方案显著增强了区块链网络的可扩展性, 其效率随着区块链网络的规模而增加.

在 Gossip 协议优化中, 一些策略试图限制 Gossip 算法引入的重复量或消耗的时间. He 等人提出一种适用于半分布式区块链网络的 HNA-Gossip 算法, 通过动态记录历史节点信息降低选择重复节点发送消息的概率[23]. Yu 等人提出了类似的算法, 不同之处在于它还包括区块在网络中的传输路径[24]. 传输节点根据传输路径过滤已接收到数据的节点, 从而避免冗余转发. Berendea 等人以 Fabric 为背景, 详细分析 Gossip 算法在 Fabric 中的弊端并改进传输模型, 同时优化了传播时间、尾部延迟和带宽消耗等[25]. Xu 等人在 Fabric 中引入密度聚类的思想提出 DC-Gossip 广播算法, 为区块链网络层构建具有高密度连接的稳定网络架构[26]. 在未来的研究工作中, 可以将对 Gossip 算法的性能研究及其对数据一致性的影响扩展到其他区块链系统.

鉴于结构化网络和 DHT 技术的优势, 结构化广播算法能够实现带宽高效的数据传输, 其中树状广播算法可以加快广播收敛时间并减少冗余流量. 在树状网络中, 消息能够以不到 2 倍树高的跳数传递到最远的节点, 流量浪费少, 但存在单点故障问题. 因此, Kan 等人提出一种基于树簇结构的传播模型[27], 该模型将单个节点扩展到集群组, 组内节点彼此连接. 在传输过程中, 消息可以从任何分支节点开始, 转发到除传入邻居之外的其他所有邻居, 并在集群组内传递. 该方法结合树状网络和 Gossip 的优势, 既实现多路径传输, 加快消息的传播速度, 又提高网络的集成能力, 保证整个树状网络的可靠性.

区块链网络不涉及复杂的资源定位和路由查找, 通过简单的广播算法即可实现消息的全网广播, 然而单从广播这一方向优化性能具有局限性. 一些研究工作如 Erlay、Graphene 等, 将广播算法与集对账相结合, 降低数据必须全网广播的需求, 并在后续操作中借助某种方法达成最终一致性. 广播算法的设计可以考虑与拓扑、编码等优化方案相结合, 更有助于提升网络性能和降低冗余现象.

3.3 中继网络

中继网络是独立于区块链网络的一种外部网络, 以减少比特币网络的区块传播时间和孤块率. 现有比特币技术采用的中继网络包括 BFRN[71]、Falcon[72]、FIBRE[73]和 bloXroute[74]. 这些中继网络之间彼此不同, 但基本功能在某种程度上相同. 中继网络由中继服务器组成, 通过在不同区域部署中继服务器实现数据传输. 理想情况下, 中继网络分 3 步将区块传播到不同区域的节点[75]. 首先, 节点将区块发送到连接的中继服务器; 其次, 进

行区域间传播, 即在中继服务器之间进行传播, 服务器之间的传输速度较快; 最后, 中继服务器将区块传播给连接的节点.

BFRN (bitcoin fast relay network) 是 Corallo 建立的第 1 个比特币中继网络[71], 通过建立具有低延迟连接和块压缩两种功能的网关减少区块的传播时间. BFRN 由全球各地的 9 台服务器构成一个集中星型结构, 矿工连接到离它们最近的中继节点, 通过连接的中继节点发送和接收区块. 之后被 Falcon 取代.

Falcon 由分布在全球各地的 10 台服务器组成[72], 在最小验证和优化拓扑结构的基础上使用直通路由加快区块的传播速率. 为了提高区块的传播效率, 节点不会检查接收到的所有 IP 数据包, 只需检查区块头便可验证区块的有效性. 但上述过程中节点在接收到数据包后才能验证区块的有效性, 因此可能存在不诚实的矿工通过网络传输无效 IP 数据包现象以浪费竞争对手的资源.

为了解决 Falcon 传输无效数据包的问题, FIBRE (fast Internet bitcoin relay engine) 中继网络[73]被提出. FIBRE 由分布在全球各地的 6 台服务器组成, 结合致密区块和直通路由提高区块的传播速率. 由于传统比特币系统使用 TCP 协议存在数据丢失问题, 因此 FIBRE 使用 UDP 为注册用户提供转发纠错机制. 当传输过程中数据丢失时, FIBRE 使用 UDP 的前向纠错技术重建传输的数据.

bloXroute[74]在上述协议的基础上提高了安全性, 通过增加块的大小、减小出块的时间间隔提高区块链的吞吐量. 为了保证区块内容的安全性, bloXroute 传播加密区块, 且加密密钥只有在该区块通过网络传播后才会显示. 同时, 为了增加区块来源的安全性, 网关不直接将块传播到 bloXroute, 而是通过网络节点中继区块以掩盖区块的来源. 此外, bloXroute 提出解决错误控制的方案, 在系统完全故障的情况下仍支持区块链操作.

中继网络结合拓扑、编码和直通路由等以高效的方式快速传输区块链消息, 在不改变区块链协议或共识机制的情况下解决区块链扩容问题. 目前中继网络仍在发展中, 其目标是不断增加对新区块链项目的支持, 成为对所有区块链开放的中立平台. 中继网络的设计除要解决网络加速和扩容问题外, 还要考虑网络中立性, 即第三方网络运营商不得对区块链节点进行干预以及窃取区块内容.

3.4 小　结

本节从编码方案、广播算法和中继网络 3 个方面梳理现有的数据传输协议, 其中重点分析编码方案的安全性问题. 在数据传输阶段, 高效地中继交易和区块对于区块链达成共识、减少存储膨胀及提高系统性能至关重要. 在有限的网络资源下, 设计高效且快速的传输协议以应对冗余数据和高并发传输, 是区块链系统实现低带宽和高吞吐的关键.

4 区块链跨链网络

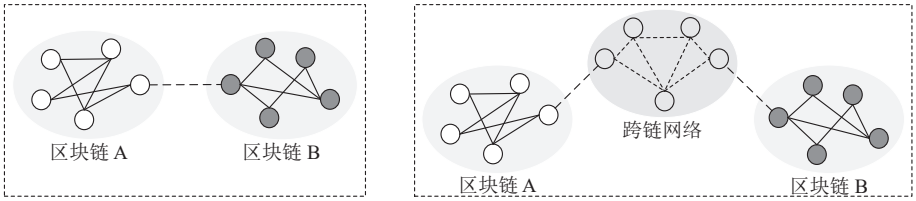
随着区块链技术的发展, 工业界和学术界涌现大量区块链平台. 跨链技术能够解决不同链间资产及状态相互交换和转移的难题, 避免价值孤岛现象[52]. 由于各区块链在网络协议、设计理念等方面的差异, 不同区块链平台很难直接进行通信, 需要跨链网络作为通信桥梁. 本节从跨链网络拓扑、通信流程和跨链网络实现这 3 个方面总结跨链技术.

4.1 跨链网络拓扑结构

跨链网络由许多中继节点组成, 提供跨链消息传输的通道. 跨链网络拓扑结构主要包括点对点拓扑和第三方网络拓扑, 如图 12 所示.

点对点拓扑是最简单的跨链结构, 不同区块链系统直接通过网关节点建立网络连接, 通常以手动配置邻居节点的方式实现. 该结构配置简单, 通信速度快. 但是当数据交互量很大时, 一条通信连接可能无法满足网络需求造成网络拥塞现象, 网关节点很容易成为跨链网络的性能瓶颈.

第三方网络拓扑指不同区块链系统与第三方跨链网络相连达到跨链通信的目的. 第三方跨链网络包含两种组织形式: 随机拓扑和中心化交易所. 随机拓扑灵活度和可靠性高, 能够适应不同的应用场景, 但网络带宽消耗和系统性能无法保证. 中心化交易所部署简单, 扩展方便, 但对中心化交易所的安全性和可靠性要求更高. 在跨链交互时, 不同的区块链系统与跨链网络相连, 在跨链网络上完成数据的传输和验证.



(a) 点对点拓扑　　(b) 第三方网络拓扑

图 12　跨链网络拓扑结构图

4.2　跨链网络通信流程

对于所有参与跨链通信的区块链, 提供数据的链称为源链, 接收数据的链称为目标链. 数据必须具有不可逆性和可追溯性, 以保证数据到达目标链且被记录后不再更改. 与单链网络传输流程类似, 跨链网络传输流程也分为两个阶段: 连接建立阶段和数据传输阶段, 如图 13 所示.

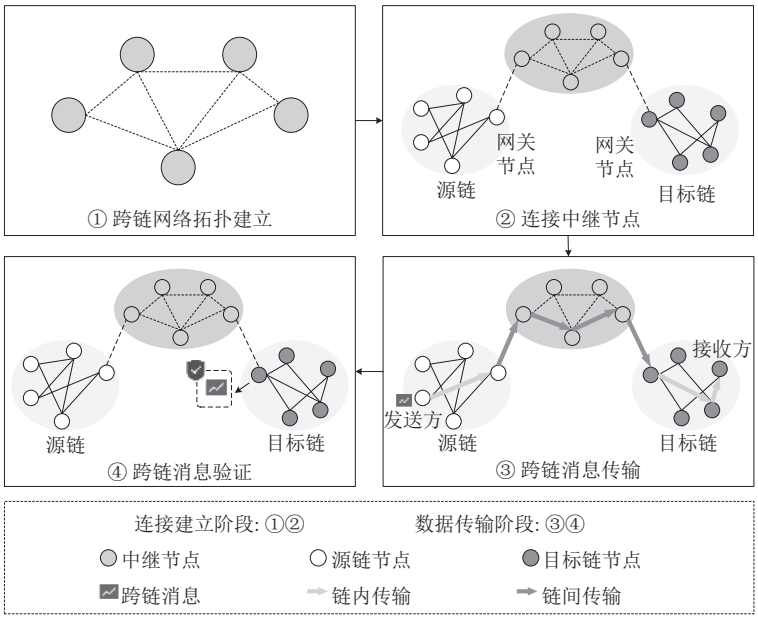


图 13　跨链网络传输流程图

节点连接建立阶段的主要任务是构建跨链网络并连接源链和目标链. 每个区块链都有独立的网络, 具有相应的网络服务和内部通信. 为实现不同区块链之间的信息交流, 源链和目标链需要选取一个特定的节点作为网关节点连接到跨链网络中, 构建数据传输的通道.

在数据传输阶段, 跨链数据经由某一路径从源链发送方传递到目标链接收方. 跨链传输路径可分为链内传输和链间传输. 链内传输指源链和目标链内部的传输过程, 不涉及跨链网络节点. 链间传输指跨链网络内部的传输过程, 通过网关节点与源链和目标链通信. 在跨链交易实现中, 网关节点接收到跨链交易后对该交易的有效性进行验证, 若有效则将交易提交到相应的接受方. 这种交易验证方式依赖网关节点保证交易的安全性. 若网关节点作恶, 例如篡改跨链交易信息并伪造符合条件的其他交易, 跨链交易的安全性无法保证.

4.3　跨链网络实现

区块链跨链技术的研究尚处于初级阶段, 不断拓展的应用场景和实际需求推动着跨链技术的发展和创新. 目前, 主流跨链技术实现可以分为公证人机制、侧链/中继、哈希锁定和分布式私钥控制 4 种类型. 公证人机制通常选举一个或多个节点作为公证人对双方交易进行确认和验证. 该机制能够灵活地支持多样化区块链结构, 但依赖

公证人的可靠性, 存在中心化风险. 侧链/中继技术是最常用的跨链方式, 通过部署相应的己方或第三方节点形成一个第三方链, 使得独立的区块链系统可以彼此通信, 实现监听并验证其他区块链的状态及交易, 实现难度较大. 哈希锁定利用哈希算法的不可逆性, 结合智能合约技术对链上资产进行锁定. 当跨链交互的两条链在规定时间内使用哈希值的原像触发解锁操作后, 智能合约自动执行资产转移. 分布式私钥控制采用分布式节点管理资产私钥, 将资产的使用权和控制权分开, 同时将原链的资产映射到跨链中. 通过门限签名技术, 当收集到的私钥达到一定数量时解除资产锁定, 实现资产流动和价值转移. 现有跨链网络通信模式可分为中继链和路由器[76]. 中继链是独立于源链和目标链的第三方区块链, 源链和目标链必须信任中继链以保障跨链通信可靠. Cosmos[77]和 Polkadot[78]是中继链方式的两个代表性项目. Cosmos 网络由许多独立并行的区块链组成, 其中中继链称为 Hub, 其他并行链称为 Zone. 各区块链通过链间通信 (inter-blockchain communication, IBC) 协议和 Hub 实现跨链操作. Cosmos 允许任何使用快速确定性共识算法 (如 Tendermint) 的区块链通过适配 IBC 建立连接, 而对于异构区块链则需要通过 Peg-Zone 进行对接. 如果接入的链太多, Cosmos 使用多个 Hub 管理并行链. Polkadot 是一个可扩展的异构多链系统, 网络由一个中继链和若干平行链组成. 各平行链通过跨链消息传递 (interchain message passing, ICMP) 机制进行通信, 并由中继链随机分配的验证者进行区块验证, 实现共享安全性. 当平行链数量增多时, Polkadot 采用 Hyper-cube 路由策略缓减中继链压力.

基于路由器的跨链网络由一个中心化的路由器或多个分布式路由器组成, 不是完全去中心化的. 代表项目有 WeCross[79]和 Interledger[80]. WeCross 为每个区块链设计一个跨链适配器, 抽象区块链资源. 跨链路由通过配置跨链适配器与区块链对接, 配置多个适配器能够达到连接多条区块链的效果. 在跨链传输时, 跨链路由根据 iPath 将请求路由到相应区块链上, 访问跨链网络中的资源. Interledger 网络使用多个连接器构建不同账本之间的路由路径, 每个连接器都会被激励以寻找最佳的传输路径. 在传输过程中, Interledger 通过哈希时间锁合约 (hashed timelock contract, HTLC) 确保安全的支付路由和资产转移, 用户可以在自己选择的网络上进行本地交易, 而无需利用其他第三方机构. 与中继链模式不同, 路由器模式不依赖任何第三方, 也无需额外的信任假设, 仅适用于节点较为可信的环境.

表 4 对以上讨论的跨链网络进行总结. 跨链网络充分考虑各区块链平台的差异, 在尽可能不改变原有设计的前提下实现区块链的融合连通, 通过验证机制确保跨链交互的安全可靠. 基于中继链的跨链网络设计了特定的共识算法, 容易受到某些破坏共识的攻击, 降低跨链交互的安全性. 基于路由器的跨链网络根据路由表内容转发消息, 路由器的存储和处理能力以及路由器之间的连接影响跨链交互的效率. 此外, 部分跨链项目设计路由策略代替传统泛洪路由, 但仅降低了链间传输的压力.

表 4　跨链网络实现

跨链项目	中心化程度	拓扑结构	路由策略	共识算法	验证机制
Cosmos[77]	去中心化	随机拓扑	—	Tendermint	并行链独立验证
Polkadot[78]	去中心化	随机拓扑	Hyper-cube	BABE+GRANDPA	中继链分配验证节点最终验证
WeCross[79]	弱中心化	随机拓扑	iPath	—	额外传输证明数据进行可信验证
Interledger[80]	弱中心化	点对点拓扑	—	—	加密算法、资金托管

4.4 小　结

本节从跨链网络拓扑、通信流程及跨链网络实现 3 个方面总结区块链跨链技术. 跨链技术是实现区块链互联互通的重要手段. 通过跨链技术, 区块链之间可以自由通信, 极大拓展区块链的应用前景. 目前跨链网络实现尚存在许多技术难题, 研究学者也在寻求多种技术融合以增强跨链系统的安全性. 未来, 跨链技术一定会在区块链的大规模应用中发挥重要作用.

5 区块链网络仿真

区块链具有可扩展性, 部分优势只能在大规模部署下体现, 例如网络规模达到成千上万个节点. 然而, 构建一

个真实的区块链网络进行性能评估具有挑战性, 不仅时间和资源成本高昂, 而且网络配置一旦部署成功不能轻易更改. 本节主要讨论另一种简单且有效的评估方法, 即区块链网络仿真.

5.1 网络仿真需求

为了比较各区块链平台, 研究学者提出多种不同的解决方案以评估区块链性能, 包括基准测试框架、实时性能监测、自行设计实验和仿真工具等. 基准测试框架通过实现真实的系统以评估各项性能指标, 通常需要一个标准化的环境和记录良好的工作负载, 如 Blockbench[81]、Hyperledger caliper[82]和 Dagbench[83]. 实时性能监测记录公共系统在实际工作负载下的性能, 准确性高但运行成本高昂. 自行设计实验需要在良好稳定的测试环境下比较系统性能. 上述 3 种测试方法都需要系统的可用性, 但并不是所有被评估的系统都是可用的. 这种情况下, 通常采用仿真工具模拟系统实现.

仿真以低成本方式复现真实系统的运行过程和性能, 允许用户设置不同的参数和配置研究不同操作场景下系统的性能. 区块链仿真工具为一组特定的资源如网络带宽、延迟等设计不同的模型, 每个模型由特定的参数进行描述. 用户只需更改模型的相关条件或参数即可全面模拟系统, 无需中断系统的运行或更改整个配置.

区块链网络层是系统的基础, 提高交易和区块的传播速度和稳定性是其基本需求. 无论哪种区块链类型, 节点发现的速度和账本同步直接依赖于网络, 丢包率和延迟时间等网络因素也可能对系统的性能产生影响. 对于区块链网络, P2P 网络仿真是一种合适的模拟方法[84], 无须进行实际的节点部署也可以很好地测试系统性能. 该方法已被用于测试区块链网络协议.

5.2 网络仿真工具

现有的研究工作将区块链系统分为 5 层架构[85], 自底向上依次是数据层、网络层、共识层、合约层和应用层. 由于区块链仿真工具设计和开发过程较为复杂, 大部分区块链仿真工具也遵循区块链多层范式, 即通常仅实现系统的一个或多个层面, 而简化与研究内容相关性较小的层面. 例如, 对共识行为的仿真测试往往缺乏对网络层的考虑, 仅使用简单的随机拓扑和泛洪广播代替. 研究表明, 大多数情况下涉及层次越少的模拟器具有更复杂的模型.

根据仿真工具源代码的可用性、引用量以及网络模拟情况, 选取 6 个区块链网络仿真工具: Shadow[86]、Bitcoin-simulator[87]、BlockSim[62]、SimBlock[88]、DAGsim[89]和 local bitcoin network simulator[90], 对比侧重于网络层的模拟, 忽略共识层、数据层等的分析.

(1) 洋葱网络仿真器 Shadow

Shadow 是一个并行离散事件的网络模拟器, 可以在一台机器上运行类似洋葱网络和比特币的真实应用程序和具有数千个节点的分布式系统. Shadow 将仿真的准确性与模拟的高效性相结合, 通过模块化设计和插件架构简化模拟和执行, 通过操作系统内核级别的仿真如排队和多线程实现并行化. 仿真器提供各种网络仿真参数, 如路由、延迟和带宽等, 但未定义网络的拓扑连接, 无法研究拓扑结构对网络性能的影响.

(2) 比特币网络仿真器 Bitcoin-simulator

Bitcoin-simulator 是一个命令行形式的开源区块链仿真器, 用于测试各种 PoW 区块链系统的性能和共识安全性, 目前仅支持比特币仿真. Bitcoin-simulator 考虑节点地理位置及网络延迟和带宽模拟网络层, 有效地量化不同参数对双花攻击和自私挖矿等区块链安全性的影响. 但是, 仿真器未定义交易相关参数和交易传播机制, 不适用于交易传输的模拟.

(3) 通用区块链网络仿真器 BlockSim

BlockSim 是基于 PeerSim 平台开发的通用区块链网络仿真器, 通过层次化架构设计实现不同区块链系统. BlockSim 根据区块链的网络、共识和存储机制设定不同的参数, 以模拟实际的网络环境. 在网络模块, 仿真器提供拓扑连接、延迟设置和广播算法等接口, 用户可以通过调整图模型结构实现结构化和非结构化拓扑结构, 并根据配置文件设置网络延迟. 仿真器能够适应网络的动态变化, 具有良好的可扩展性.

(4) 事件驱动的区块链仿真器 SimBlock

SimBlock 是一个事件驱动的区块链网络仿真器, 可以轻松改变节点行为以研究节点行为对区块链的影响, 其

中区块的生成和消息的传输视为事件. SimBlock 定义了区块参数、节点参数和网络参数, 可以非常准确的模拟各种区块链系统. 另外, 仿真器还阐明邻居选择算法和中继网络对区块传播时间的影响. 但是由于网络拓扑结构在短时间内不会发生显著变化, 仿真器没有针对拓扑的研究.

(5) 图式区块链仿真器 DAGsim

DAGsim 是一个异步的时间连续型多主体仿真框架, 用于评估所设计的共识算法在稳定性和安全性方面的效果. DAGsim 考虑以交易和区块为单位的两种 DAG 模型, 制定一系列参数和规则以实现图式区块链结构. 节点网络是加权无向图, 节点之间的网络延迟保存在距离矩阵中. DAGsim 存在一定的局限性, 假定节点是诚实和半诚实的, 未考虑恶意节点存在的情况, 缺乏针对特定攻击的模拟.

(6) 轻量级比特币本地网络仿真器 local bitcoin network simulator

Local bitcoin network simulator 利用轻量级虚拟化技术 Docker 构建一个经过微调的比特币本地测试网络, 用于测试和评估修改比特币后的影响. 仿真器引入多个网络约束评估不同现实网络环境下的比特币性能, 如不同的拓扑结构、延迟和挖矿算力分布和自动调整挖矿难度级别等. 仿真器还研究哈希能力、网络延迟和挖矿难度对网络一致性和公平性的影响. 虽然仿真器具有高度可配置性, 但仅支持少量节点间的简单设定, 难以在成千的节点规模下构建高效的网络连接.

表 5 对以上讨论的仿真工具进行总结. 由于基于 PoW 的区块链系统占数字加密货币总市值的 90% 以上[87], 因此大多数区块链仿真工具致力于模拟各种 PoW 区块链平台, 其中比特币网络的模拟是主要焦点. 仿真工具的功能有限, 旨在模拟实际区块链的某些关键部分. 相比之下, Bitcoin-simulator 是最全面的仿真模型, 提供各种输入参数和输出指标, 但已无法准确模拟最新的比特币网络. Shadow 和 SimBlock 也具有丰富的功能, 但缺乏针对拓扑结构的研究. BlockSim 提供多样化接口和多层架构全面模拟区块链系统, 但未考虑网络节点的异构性. Local bitcoin network simulator 仅支持节点数量较少的研究. DAGsim 研究图式区块链中共识算法的稳定性和安全性. 所有仿真器的缺点是通常采用固定数量的节点模拟网络, 不符合公有区块链网络性质. 从跨链角度, 现有的区块链仿真工具仅关注单链网络模拟, 无法支持多条不同区块链交互, 不能有效地进行跨链网络协议测试.

表 5 区块链网络仿真工具

仿真工具	特点	功能	模拟平台	局限性
Shadow[86]	仿真与模拟相结合	路由定制	比特币	缺乏拓扑结构定义
Bitcoin-simulator[87]	可用于测试共识安全性	拓扑、路由定制	PoW区块链	缺乏交易参数和传播机制
BlockSim[62]	支持结构化拓扑和广播	拓扑、路由定制	PoW区块链	未考虑异构硬件配置
SimBlock[88]	基于事件驱动机制	路由定制	PoW区块链	未针对拓扑研究
DAGsim[89]	图式区块链仿真	共识算法	DAG	缺乏特定攻击模拟
Local bitcoin network simulator[90]	利用轻量级虚拟化技术	拓扑定制	比特币	仅支持在少量节点间设定

5.3 小 结

本节说明区块链网络仿真需求并对比网络仿真工具. 大部分仿真工具提供数据定制、网络参数设定和大规模模拟等功能, 但在拓扑和路由模拟方面存在不足, 应用场景受功能限制. 目前尚不存在适用于广泛场景的通用仿真器, 需要根据设计特点和实现功能选择适合的仿真工具. 跨链交互基于现有的区块链平台, 潜在的节点规模十分庞大, 开发适用于跨链交互网络的仿真工具仍然是市场空缺.

6 总结与展望

本文中, 我们重点关注区块链网络传输流程, 系统梳理区块链网络在拓扑结构和传输协议上的研究工作, 归纳总结跨链网络实现, 对比分析网络仿真工具. 随着区块链技术的不断成熟和网络规模的不断扩大, 在网络通信方面尚存在许多优化工作. 本节对未来的研究工作提出一些可能的方向, 值得研究人员广泛关注.

(1) 基于激励惩罚的网络信任机制研究

一些工作强调区块链的去信任化, 但事实上区块链并不是完全去信任的. 节点可以根据先前的行为增加彼此信任度, 通过算法等方式保障. 现有的信任机制存在以下问题: 首先, 区块链网络研究中, 大部分工作假设节点是无私奉献的, 但实际网络中节点可能存在合作, 缺乏有效的网络监管和激励机制. 其次, 节点信任机制仅区分节点行为, 缺乏对恶意行为的惩罚措施, 如利用多次小额交易获取高信任度进行大额交易欺诈、恶意评价或主观评价导致信任度不准确等行为.

激励惩罚即通过一定的奖励措施鼓励节点提供服务和资源, 引导主动性不强的节点并严重惩罚恶意行为节点. 网络信任机制需要提供合理的激励惩罚机制以鼓励节点产生良好的网络行为, 促进网络节点的合作互利. 信任机制的引入可以增强网络安全性, 同时也为拓扑构造和数据传输协议提供新的方向, 例如利用博弈论方法分析节点在纳什均衡时的行为, 从经济学的角度改善系统性能等.

(2) 基于节点感知的自适应网络拓扑研究

拓扑结构提供数据传输的通道. 结构化网络传输效率高, 但去中心化程度低, 节点易遭到攻击. 非结构化网络去中心化程度高, 但传输效率低, 不易监管. 相比之下, 结构化网络性能更优, 但在特定场景和需求下仍然需要非结构化网络. 现有的区块链网络拓扑模型结构单一, 无法感知节点变化或节点搅动等现象, 缺乏对拜占庭容错、良好连通性和小网络直径等理想属性的理论保证.

在实际网络环境中, 节点具有不同的处理能力和加入/离开等自主行为, 自适应拓扑研究能够感应节点能力并将节点自组织为相对高效的拓扑. 例如, 对于负载性能较好的节点构建结构化拓扑提高传输效率, 对于负载性能较差的节点构建非结构化拓扑满足去中心化特性, 以此达到负载均衡的目的. 为了适应实际网络环境和确保网络安全, 基于节点感知的自适应网络拓扑研究是必然趋势.

(3) 基于 NDN 的数据传输协议研究

传输协议决定数据同步的性能. 理想的数据传输协议应在现有的安全模型下保持健壮, 具有合理的传输延迟, 尽可能充分利用网络资源, 最大化传输效率和系统性能. 现有的区块链网络基于传统的 TCP/IP 网络架构, 利用多播方式实现数据的同步传输. 由于该架构是在两个节点间进行端到端数据交换, 区块链中交易和区块的传输会产生大量冗余流量, 带宽利用率低且传输延迟高.

与传统 TCP 架构以主机为中心的通信模式不同, 命名数据网络 (named data networking, NDN) 以内容为中心, 所有数据采用名字标识, 按照名称在全网范围内搜索. 这一性质恰好符合区块链节点的匿名性和数据的开放性特征, 由此开展基于 NDN 的数据传输协议研究. 如果采用 NDN 作为区块链网络底层架构, 交易和区块的传播均由 NDN 支持能够有效减少传输开销和冗余流量, 关键问题在于区块链泛洪广播机制和 NDN 兴趣传播机制之间的融合.

(4) 基于区块并发的图式区块链网络研究

图式区块链通过 DAG 技术和异步并发记账解决链式区块链效率低下的问题, 与链式结构相比具有确认速度快、吞吐量高和可扩展性强等优势, 适合小额结算、即时通信等领域. 然而, 图式区块链也存在许多缺陷. 首先, 图式区块链的验证机制是新交易验证旧交易, 在节点交易量较少的情况下可能导致交易长时间无法确认问题. 虽然一些协议引入超级节点或见证人等, 但带来中心化风险. 其次, 图式区块链采用异步通信算法, 不存在一个全局的排序机制, 一致性较弱. 最后, 图式区块链应用场景有限, 很难用于需要同步或一致性要求较高的系统中, 安全性有待时间验证.

图式区块链在并发性上有显著优势, 有望突破传统区块链的性能瓶颈. 从账本拓扑结构角度, 如果网络延迟过高, 可能会导致拓扑纵向增长; 如果网络吞吐量过低, 可能会导致拓扑横向增长. 设计良好的网络并发管理方案以优化交易发布率, 为高吞吐量和高区块并发的图式区块链系统提供网络支持.

区块链的出现是一次技术的革命, 其去中心化、分布式、开放自治的特性为治理活动提供了重要的技术支撑. 为了保证区块链系统的稳定运行, 信息交换至关重要. 网络协议作为区块链技术的底层实现, 直接影响着上层共识速度和系统性能, 逐渐成为关键的研究方向之一. 通过本文的综述与介绍, 希望对本领域研究者有所启发, 开

展更多关于区块链网络协议设计的研究工作.

References:

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. 2008. https://bitcoin.org/bitcoin.pdf

[2] Cai CJ, Weng J, Yuan XL, Wang C. Enabling reliable keyword search in encrypted decentralized storage with fairness. IEEE Trans. on Dependable and Secure Computing, 2021, 18(1): 131–144. [doi: 10.1109/TDSC.2018.2877332]

[3] Abraham I, Malkhi D, Nayak K, Ren L, Yin MF. Sync HotStuff: Simple and practical synchronous state machine replication. In: Proc. of the 2020 IEEE Symp. on Security and Privacy. San Francisco: IEEE, 2020. 106–118. [doi: 10.1109/SP40000.2020.00044]

[4] Jin H, Xiao J. Towards trustworthy blockchain systems in the era of "Internet of value": Development, challenges, and future trends. Science China Information Sciences, 2022, 65(5): 153101. [doi: 10.1007/s11432-020-3183-0]

[5] Hyperchain. 2016. https://www.hyperchain.cn/

[6] FiMAX. 2018. https://fimax.ocft.com

[7] Ant financial blockchain. 2019 (in Chinese). https://antchain.antgroup.com/

[8] IBM supply chain. 2019. https://www.ibm.com/blockchain/supply-chain

[9] Croman K, Decker C, Eyal I, Gencer AE, Juels A, Kosba A, Miller A, Saxena P, Shi E, Sirer EG, Song D, Wattenhofer R. On scaling decentralized blockchains. In: Proc. of the 2016 Int'l Workshops on Financial Cryptography and Data Security. Christ Church: Springer, 2016. 106–125. [doi: 10.1007/978-3-662-53357-4_8]

[10] Liu MD, Shi YJ, Chen ZN. Distributed trusted network connection architecture based on blockchain. Ruan Jian Xue Bao/Journal of Software, 2019, 30(8): 2314−2336 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5764.htm [doi: 10.13328/j.cnki.jos.005764]

[11] Ghosh BC, Bhartia T, Addya SK, Chakraborty S. Leveraging public-private blockchain interoperability for closed consortium interfacing. In: Proc. of the 2021 IEEE Conf. on Computer Communications. Vancouver: IEEE, 2021. 1–10. [doi: 10.1109/INFOCOM42981.2021.9488683]

[12] Naumenko G, Maxwell G, Wuille P, Fedorova A, Beschastnikh I. Erlay: Efficient transaction relay for bitcoin. In: Proc. of the 2019 ACM SIGSAC Conf. on Computer and Communications Security. London: ACM, 2019. 817–831. [doi: 10.1145/3319535.3354237]

[13] Han YL, Li CX, Li PL, Wu M, Zhou D, Long F. Shrec: Bandwidth-efficient transaction relay in high-throughput blockchain systems. In: Proc. of the 11th ACM Symp. on Cloud Computing. ACM, 2020. 238–252. [doi: 10.1145/3419111.3421283]

[14] Tschipper P. BUIP010 Xtreme Thinblocks. 2016. https://bitco.in/forum/threads/buip010-passed-xtreme-thinblocks

[15] Toomim J. Benefits of LTOR in block entropy encoding. 2018. https://jtoomim.medium.com/benefits-of-ltor-in-block-entropy-encoding-or-8d5b77cc2ab0

[16] Corallo M. BIP152: Compact block relay. 2016. https://github.com/bitcoin/bips/blob/master/bip-0152.mediawiki

[17] Ding DH, Jiang X, Wang JP, Wang H, Zhang XB, Sun Y. Txilm: Lossy block compression with salted short hashing. arXiv:1906.06500, 2019.

[18] Ozisik AP, Andresen G, Levine BN, Tapp D, Bissias G, Katkuri S. Graphene: Efficient interactive set reconciliation applied to blockchain propagation. In: Proc. of the 2019 ACM Special Interest Group on Data Communication. Beijing: ACM, 2019. 303–317. [doi: 10.1145/3341302.3342082]

[19] Santiago C, Lee C. Accelerating message propagation in blockchain networks. In: Proc. of the 2020 Int'l Conf. on Information and Communication Technology Convergence. Jeju Island: IEEE, 2020. 157–160. [doi: 10.1109/ICTC49870.2020.9289312]

[20] Zhang MQ, Cheng YK, Deng XT, Wang B, Xie J, Yang YY, Zhang JR. Accelerating transactions relay in blockchain networks via reputation. In: Proc. of the 29th IEEE/ACM Int'l Symp. on Quality of Service. Tokyo: IEEE, 2021. 1–10. [doi: 10.1109/IWQOS52092.2021.9521324]

[21] Chen JL, Qin Y. Reducing block propagation delay in blockchain networks via guarantee verification. In: Proc. of the 29th IEEE Int'l Conf. on Network Protocols. Dallas: IEEE, 2021. 1–6. [doi: 10.1109/ICNP52444.2021.9651926]

[22] Ayinala K, Choi BY, Song SJ. PiChu: Accelerating block broadcasting in blockchain networks with pipelining and chunking. In: Proc. of the 2020 IEEE Int'l Conf. on Blockchain. Rhodes: IEEE, 2020. 221–228. [doi: 10.1109/Blockchain50366.2020.00035]

[23] He XW, Cui YJ, Jiang YC. An improved gossip algorithm based on semi-distributed blockchain network. In: Proc. of the 2019 Int'l Conf. on Cyber-enabled Distributed Computing and Knowledge Discovery. Guilin: IEEE, 2019. 24–27. [doi: 10.1109/CyberC.2019.00014]

[24] Yu B, Li XF, Zhao H, Zhou T. A scalable blockchain network model with transmission paths and neighbor node subareas. Computing, 2022, 104(10): 2253–2277. [doi: 10.1007/s00607-021-00913-1]

[25] Berendea N, Mercier H, Onica E, Rivière E. Fair and efficient gossip in hyperledger fabric. In: Proc. of the 40th IEEE Int'l Conf. on

Distributed Computing Systems. Singapore: IEEE, 2020. 190–200. [doi: 10.1109/ICDCS47774.2020.00027]

[26] Xu ZG, Ye KZ, Dong XH, Han HM, Yan ZZ, Chen XX, Liao DY, Wang HT. DC-Gossip: An enhanced broadcast protocol in hyperledger fabric based on density clustering. In: Proc. of the 17th Int'l Conf. on Wireless Algorithms, Systems, and Applications. Dalian: Springer, 2022. 3–19. [doi: 10.1007/978-3-031-19211-1_1]

[27] Kan J, Zou LY, Liu B, Huang X. Boost blockchain broadcast propagation with tree routing. In: Proc. of the 1st Int'l Conf. on Smart Blockchain. Tokyo: Springer, 2018. 77–85. [doi: 10.1007/978-3-030-05764-0_8]

[28] Neudecker T, Hartenstein H. Network layer aspects of permissionless blockchains. IEEE Communications Surveys and Tutorials, 2019, 21(1): 838–857. [doi: 10.1109/COMST.2018.2852480]

[29] Katkuri S. A survey of data transfer and storage techniques in prevalent cryptocurrencies and suggested improvements. arXiv: 1808.03380, 2018.

[30] Buterin V. A next-generation smart contract and decentralized application platform. 2014. https://ethereum.org/en/whitepaper/

[31] LeMahieu C. Nano: A feeless distributed cryptocurrency network. 2018. https://nano.org/en/whitepaper

[32] Popov S. The tangle. 2018. http://www.descryptions.com/Iota.pdf

[33] Dotan M, Pignolet YA, Schmid S, Tochner S, Zohar A. Survey on cryptocurrency networking: Context, state-of-the-art, challenges. arXiv:2008.08412, 2020.

[34] Antwi R, Gadze JD, Tchao ET, Sikora A, Nunoo-Mensah H, Agbemenu AS, Obour Agyekum KOB, Agyemang JO, Welte D, Keelson E. A survey on network optimization techniques for blockchain systems. Algorithms, 2022, 15(6): 193. [doi: 10.3390/a15060193]

[35] Zhou WL, Wu XF. Survey of P2P technologies. Computer Engineering and Design, 2006, 27(1): 76−79 (in Chinese with English abstract). [doi: 10.16208/j.issn1000-7024.2006.01.023]

[36] Donet JAD, Pérez-Solà C, Herrera-Joancomartí J. The bitcoin P2P network. In: Proc. of the 2014 Int'l Workshops on Financial Cryptography and Data Security. Christ Church: Springer, 2014. 87–102. [doi: 10.1007/978-3-662-44774-1_7]

[37] Tang JY, Li L, Tang DQ, Xiao WD. Research on P2P bootstrapping mechanism. Computer Science, 2009, 36(5): 27–29, 44 (in Chinese with English abstract). [doi: 10.3969/j.issn.1002-137X.2009.05.006]

[38] Douceur JR. The Sybil attack. In: Proc. of the 1st Int'l Workshop on Peer-to-peer Systems. Cambridge: Springer, 2002. 251–260. [doi: 10.1007/3-540-45748-8_24]

[39] Dai HN, Wang H, Xiao H, Li XR, Wang Q. On eavesdropping attacks in wireless networks. In: Proc. of the 2016 IEEE Int'l Conf. on Computational Science and Engineering and the IEEE Int'l Conf. on Embedded and Ubiquitous Computing and the 15th Int'l Symp. on Distributed Computing and Applications for Business Engineering. Paris: IEEE, 2016. 138−141. [doi: 10.1109/CSE-EUC-DCABES.2016.173]

[40] Lau F, Rubin SH, Smith MH, Trajkovic L. Distributed denial of service attacks. In: Proc. of the 2000 IEEE Int'l Conf. on Systems, Man and Cybernetics. Nashville: IEEE, 2000. 2275–2280. [doi: 10.1109/ICSMC.2000.886455]

[41] Heilman E, Kendler A, Zohar A, Goldberg S. Eclipse attacks on Bitcoin's peer-to-peer network. In: Proc. of the 24th USENIX Conf. on Security Symp. Washington: USENIX Association, 2015. 129–144.

[42] Karame GO, Androulaki E, Capkun S. Double-spending fast payments in bitcoin. In: Proc. of the 2012 ACM Conf. on Computer and Communications Security. Raleigh: ACM, 2012. 906−917. [doi: 10.1145/2382196.2382292]

[43] Sapirshtein A, Sompolinsky Y, Zohar A. Optimal selfish mining strategies in bitcoin. In: Proc. of the 20th Int'l Conf. on Financial Cryptography and Data Security. Christ Church: Springer, 2016. 515–532. [doi: 10.1007/978-3-662-54970-4_30]

[44] Fanning S, Parker S. Napster. 1999. https://us.napster.com/

[45] Frankel J, Pepper T. Gnutella. 2001. https://zh.wikipedia.org/wiki/Gnutella

[46] Maymounkov P, Mazières D. Kademlia: A peer-to-peer information system based on the XOR metric. In: Proc. of the 1st Int'l Workshop on Peer-to-peer Systems. Cambridge: Springer, 2002. 53–65. [doi: 10.1007/3-540-45748-8_5]

[47] Sharman Networks Ltd. Kazaa media desktop. 2001. http://www.kazaa.com/

[48] Androulaki E, Barger A, Bortnikov V, *et al*. Hyperledger fabric: A distributed operating system for permissioned blockchains. In: Proc. of the 13th EuroSys Conf. Porto: ACM, 2018. 30. [doi: 10.1145/3190508.3190538]

[49] Demers A, Greene D, Hauser C, Irish W, Larson J, Shenker S, Sturgis H, Swinehart D, Terry D. Epidemic algorithms for replicated database maintenance. In: Proc. of the 6th Annual ACM Symp. on Principles of Distributed Computing. Vancouver: ACM, 1987. 1–12. [doi: 10.1145/41840.41841]

[50] Tian GH, Hu YH, Chen XF. Research progress on attack and defense techniques in block-chain system. Ruan Jian Xue Bao/Journal of Software, 2021, 32(5): 1495−1525 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6213.htm [doi: 10.13328/j.cnki.jos.006213]

[51] Tsai WT, Yu L, Wang R, Liu N, Deng EY. Blockchain application development techniques. Ruan Jian Xue Bao/Journal of Software, 2017, 28(6): 1474−1487 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5232.htm [doi: 10.13328/j.cnki.jos.005232]

[52] Li F, Li ZR, Zhao H. Research on the progress in cross-chain technology of blockchains. Ruan Jian Xue Bao/Journal of Software, 2019, 30(6): 1649−1660 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5741.htm [doi: 10.13328/j.cnki.jos.005741]

[53] Huang HW, Kong W, Peng XW, Zheng ZB. Survey on blockchain sharding technology. Computer Engineering, 2022, 48(6): 1−16 (in Chinese with English abstract). [doi: 10.19678/j.issn.1000-3428.0063887]

[54] Gao ZF, Zheng JL, Tang SY, Long Y, Liu ZQ, Liu Z, Gu DW. State-of-the-art survey of consensus mechanisms on dag-based distributed ledger. Ruan Jian Xue Bao/Journal of Software, 2020, 31(4): 1124−1142 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5982.htm [doi: 10.13328/j.cnki.jos.005982]

[55] Miller A, Litton J, Pachulski A, Gupta N, Levin D, Spring N, Bhattacharjee B. Discovering bitcoin's public topology and influential nodes. 2015. https://www.cs.umd.edu/projects/coinscope/coinscope.pdf

[56] Kim S K, Ma Z, Murali S, Mason J, Miller A, Bailey M. Measuring Ethereum network peers. In: Proc. of the 2018 Internet Measurement Conf. Boston: ACM, 2018. 91–104. [doi: 10.1145/3278532.3278542]

[57] Alex L. The RLPx transport protocol. 2017. https://github.com/ethereum/devp2p/blob/master/rlpx.md

[58] Decker C, Wattenhofer R. Information propagation in the bitcoin network. In: Proc. of the 2013 Int'l Conf. on Peer-to-peer Computing. Trento: IEEE, 2013. 1−10. [doi: 10.1109/P2P.2013.6688704]

[59] Fadhil M, Owenson G, Adda M. A Bitcoin model for evaluation of clustering to improve propagation delay in bitcoin network. In: Proc. of the 2016 IEEE Int'l Conf. on Computational Science and Engineering and the IEEE Int'l Conf. on Embedded and Ubiquitous Computing and the15th Int'l Symp. on Distributed Computing and Applications for Business Engineering. Paris: IEEE, 2016. 468–475. [doi: 10.1109/CSE-EUC-DCABES.2016.226]

[60] Fadhil M, Owenson G, Adda M. Locality based approach to improve propagation delay on the bitcoin peer-to-peer network. In: Proc. of the 2017 IFIP/IEEE Symp. on Integrated Network and Service Management. Lisbon: IEEE, 2017. 556–559. [doi: 10.23919/INM.2017.7987328]

[61] Fadhil M, Owenson G, Adda M. Proximity awareness approach to enhance propagation delay on the bitcoin peer-to-peer network. In: Proc. of the 37th IEEE Int'l Conf. on Distributed Computing Systems. Atlanta: IEEE, 2017. 2411–2416. [doi: 10.1109/ICDCS.2017.53]

[62] Hao WF, Zeng JJ, Dai XH, Xiao J, Hua QS, Chen HH, Li KC, Jin H. BlockP2P: Enabling fast blockchain broadcast with scalable peer-to-peer network topology. In: Proc. of the 14th Int'l Conf. on Green, Pervasive, and Cloud Computing. Uberlândia: Springer, 2019. 223–237. [doi: 10.1007/978-3-030-19223-5_16]

[63] Mao YF, Deb S, Venkatakrishnan SB, Kannan S, Srinivasan K. Perigee: Efficient peer-to-peer network design for blockchains. In: Proc. of the 39th Symp. on Principles of Distributed Computing. Virtual Event: ACM, 2020. 428–437. [doi: 10.1145/3382734.3405704]

[64] Sun YX, Edmundson A, Vanbever L, Li O, Rexford J, Chiang M, Mittal P. RAPTOR: Routing attacks on privacy in Tor. In: Proc. of the 24th USENIX Conf. Security Symp. Washington: USENIX Association, 2015. 271−286.

[65] Dworkin MJ. SHA-3 standard: Permutation-based hash and extendable-output functions. NIST, Federal Information Processing Standards Publication, NIST FIPS-202, 2015. [doi: 10.6028/NIST.FIPS.202]

[66] Aumasson JP, Bernstein DJ. SipHash: A fast short-input PRF. In: Proc. of the 12th Int'l Conf. on Cryptology in India. Kolkata: Springer, 2012. 489–508. [doi: 10.1007/978-3-642-34931-7_28]

[67] Toomim J. Block propagation data from bitcoin cash's stress test. 2018. https://jtoomim.medium.com/block-propagation-data-from-bitcoin-cashs-stress-test-5b1d7d39a234

[68] Neudecker T, Andelfinger P, Hartenstein H. Timing analysis for inferring the topology of the bitcoin peer-to-peer network. In: Proc. of the 2016 Int'l IEEE Conf. on Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress. Toulouse: IEEE, 2016. 358–367. [doi: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0070]

[69] Li CX, Li PL, Zhou D, Yang Z, Wu M, Yang G, Xu W, Long F, Yao ACC. A decentralized blockchain with high throughput and fast confirmation. In: Proc. of the 2020 USENIX Conf. on USENIX Annual Technical Conf. USENIX Association, 2020. 515–528.

[70] Wang JP, Wang H. Monoxide: Scale out blockchain with asynchronous consensus zones. In: Proc. of the 16th USENIX Conf. on Networked Systems Design and Implementation. Boston: USENIX Association, 2019. 95–112.

[71] BFRN. 2014. http://bitcoinrelaynetwork.org/

[72] Basu S, Eyal I, Sirer EG. Falcon: Relay network for bitcoin blocks. 2016. https://www.falcon-net.org/

[73] FIBRE. 2016. http://bitcoinfibre.org/

[74] Klarman U, Basu S, Kuzmanovic A, Sirer EG. bloXroute: A scalable trustless blockchain distribution network whitepaper. 2018. https://

bloxroute.com/wp-content/uploads/2019/11/bloXrouteWhitepaper.pdf

[75] Otsuki K, Banno R, Shudo K. Quantitatively analyzing relay networks in bitcoin. In: Proc. of the 2020 IEEE Int'l Conf. on Blockchain. Rhodes: IEEE, 2020. 214–220. [doi: 10.1109/Blockchain50366.2020.00034]

[76] Zeng JJ. Research on structured hierarchical cross-chain network topology model [MS. Thesis]. Wuhan: Huazhong University of Science and Technology, 2021 (in Chinese with English abstract). [doi: 10.27157/d.cnki.ghzku.2021.005267]

[77] Kwon J, Buchman E. Cosmos whitepaper. 2016. https://v1.cosmos.network/resources/whitepaper

[78] Wood G. Polkadot: Vision for a heterogeneous multi-chain framework. 2016. https://polkadot.network/PolkaDotPaper.pdf, 2016.

[79] WeBank. WeCross whitepaper. 2019 (in Chinese). https://wecross.readthedocs.io/zh_CN/latest/

[80] Thomas S, Schwartz E. A protocol for interledger payments. 2015. https://interledger.org/interledger.pdf

[81] Dinh TTA, Wang J, Chen G, Liu R, Ooi BC, Tan KL. Blockbench: A framework for analyzing private blockchains. In: Proc. of the 2017 ACM Int'l Conf. on Management of Data. Chicago: ACM, 2017. 1085–1100. [doi: 10.1145/3035918.3064033]

[82] Hyperledger caliper. 2019. https://www.hyperledger.org/use/caliper

[83] Dong ZL, Zheng E, Choon Y, Zomaya AY. DAGBENCH: A performance evaluation framework for dag distributed ledgers. In: Proc. of the 12th IEEE Int'l Conf. on Cloud Computing. Milan: IEEE, 2019. 264–271. [doi: 10.1109/CLOUD.2019.00053]

[84] Montresor A, Jelasity M. PeerSim: A scalable P2P simulator. In: Proc. of the 9th IEEE Int'l Conf. on Peer-to-peer Computing. Seattle: IEEE, 2009. 99–100. [doi: 10.1109/P2P.2009.5284506]

[85] Shao QF, Jin CQ, Zhang Z, Qian WN, Zhou AY. Blockchain: Architecture and research progress. Chinese Journal of Computers, 2018, 41(5): 969–988 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2018.00969]

[86] Miller A, Jansen R. Shadow-bitcoin: Scalable simulation via direct execution of multi-threaded applications. In: Proc. of the 8th USENIX Conf. on Cyber Security Experimentation and Test. Washington: USENIX Association, 2015.

[87] Gervais A, Karame GO, Wüst K, Glykantzis V, Ritzdorf H, Capkun S. On the security and performance of proof of work blockchains. In: Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security. Vienna: ACM, 2016. 3–16. [doi: 10.1145/2976749.2978341]

[88] Aoki Y, Otsuki K, Kaneko T, Banno R, Shudo K. SimBlock: A blockchain network simulator. In: Proc. of the 2019 IEEE Conf. on Computer Communications Workshops. Paris: IEEE, 2019. 325–329. [doi: 10.1109/INFCOMW.2019.8845253]

[89] Zander M, Waite T, Harz D. DAGsim: Simulation of DAG-based distributed ledger protocols. ACM SIGMETRICS Performance Evaluation Review, 2019, 46(3): 118–121. [doi: 10.1145/3308897.3308951]

[90] Alsahan L, Lasla N, Abdallah M. Local bitcoin network simulator for performance evaluation using lightweight virtualization. In: Proc. of the 2020 IEEE Int'l Conf. on Informatics, IoT, and Enabling Technologies. Doha: IEEE, 2020. 355–360. [doi: 10.1109/ICIoT48696.2020.9089630]

附中文参考文献:

[7] 蚂蚁链. 2019. https://antchain.antgroup.com/

[10] 刘明达, 拾以娟, 陈左宁. 基于区块链的分布式可信网络连接架构. 软件学报, 2019, 30(8): 2314−2336. http://www.jos.org.cn/1000-9825/5764.htm [doi: 10.13328/j.cnki.jos.005764]

[35] 周文莉, 吴晓非. P2P技术综述. 计算机工程与设计, 2006, 27(1): 76–79. [doi: 10.16208/j.issn1000-7024.2006.01.023]

[37] 唐九阳, 李榴, 汤大权, 肖卫东. P2P入网机制研究. 计算机科学, 2009, 36(5): 27–29, 44. [doi: 10.3969/j.issn.1002-137X.2009.05.006]

[50] 田国华, 胡云瀚, 陈晓峰. 区块链系统攻击与防御技术研究进展. 软件学报, 2021, 32(5): 1495−1525. http://www.jos.org.cn/1000-9825/6213.htm [doi: 10.13328/j.cnki.jos.006213]

[51] 蔡维德, 郁莲, 王荣, 刘娜, 邓恩艳. 基于区块链的应用系统开发方法研究. 软件学报, 2017, 28(6): 1474−1487. http://www.jos.org.cn/1000-9825/5232.htm [doi: 10.13328/j.cnki.jos.005232]

[52] 李芳, 李卓然, 赵赫. 区块链跨链技术进展研究. 软件学报, 2019, 30(6): 1649–1660. http://www.jos.org.cn/1000-9825/5741.htm [doi: 10.13328/j.cnki.jos.005741]

[53] 黄华威, 孔伟, 彭肖文, 郑子彬. 区块链分片技术综述. 计算机工程, 2022, 48(6): 1–16. [doi: 10.19678/j.issn.1000-3428.0063887]

[54] 高政风, 郑继来, 汤舒扬, 龙宇, 刘志强, 刘振, 谷大武. 基于DAG的分布式账本共识机制研究. 软件学报, 2020, 31(4): 1124–1142. http://www.jos.org.cn/1000-9825/5982.htm [doi: 10.13328/j.cnki.jos.005982]

[76] 曾家杰. 面向跨链交互的区块链结构化分层网络拓扑模型研究 [硕士学位论文]. 武汉: 华中科技大学, 2021. [doi: 10.27157/d.cnki.ghzku.2021.005267]

[79] 微众银行区块链团队. WeCross技术白皮书. 2019. https://wecross.readthedocs.io/zh_CN/latest/

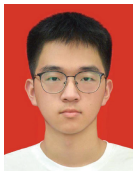
[85] 邵奇峰, 金澈清, 张召, 钱卫宁, 周傲英. 区块链技术: 架构及进展. 计算机学报, 2018, 41(5): 969–988. [doi: 10.11897/SP.J.1016.2018.00969]

司冰茹(1997－), 女, 硕士生, 主要研究领域为区块链网络.

戴小海(1992－), 男, 博士, CCF 专业会员, 主要研究领域为区块链可扩展性.

肖江(1988－), 女, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为区块链, 分布式计算.

金海(1965－), 男, 博士, 教授, 博士生导师, CCF 会士, 主要研究领域为分布式系统.



刘存扬(2001－), 男, 硕士生, 主要研究领域为区块链分片.