软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(2):800−827 [doi: 10.13328/j.cnki.jos.006950]

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

区块链互操作技术综述*

段田田[1,2], 张瀚文[1,2], 李 博[1,2], 宋兆雄[1], 李忠诚[1,2], 张 珺[4,5], 孙 毅[1,2,3]

[1](中国科学院 计算技术研究所, 北京 100190)

[2](中国科学院大学, 北京 100049)

[3](山东省区块链金融重点实验室, 山东 济南 250014)

[4](内蒙古大学, 内蒙古 呼和浩特 010021)

[5](雄安新区智能城市创新联合会 区块链实验室, 河北 保定 071700)

通信作者: 张瀚文, E-mail: hwzhang@ict.ac.cn

摘 要: 区块链技术被认为是构建价值互联网的基石, 然而彼此独立的区块链系统形成了数据、价值孤岛. 区块链互操作 (也被称为跨链操作) 是打破链间壁垒、构建区块链网络的关键技术. 在区分狭义与广义区块链互操作的基础上, 重新定义狭义区块链互操作, 并抽象出跨链读与跨链写两类基本操作; 分析总结实现狭义区块链互操作需要解决的 3 个关键技术问题: 跨链信息传输、跨链信任传递、跨链操作原子性保障; 系统梳理这 3 个问题的研究现状, 并分别从多角度进行比较; 在此基础上, 从关键技术问题的角度分析具有代表性的整体解决方案; 最后指出几个值得进一步探索的研究方向.

关键词: 区块链; 区块链互操作; 跨链; 原子性

中图法分类号: TP393

中文引用格式: 段田田, 张瀚文, 李博, 宋兆雄, 李忠诚, 张珺, 孙毅. 区块链互操作技术综述. 软件学报, 2024, 35(2): 800–827. http://www.jos.org.cn/1000-9825/6950.htm

英文引用格式: Duan TT, Zhang HW, Li B, Song ZX, Li ZC, Zhang J, Sun Y. Survey on Blockchain Interoperability. Ruan Jian Xue Bao/Journal of Software, 2024, 35(2): 800–827 (in Chinese). http://www.jos.org.cn/1000-9825/6950.htm

Survey on Blockchain Interoperability

DUAN Tian-Tian[1,2], ZHANG Han-Wen[1,2], LI Bo[1,2], SONG Zhao-Xiong[1], LI Zhong-Cheng[1,2], ZHANG Jun[4,5], SUN Yi[1,2,3]

[1](Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

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

[3](Shandong Key Laboratory of Blockchain Finance, Jinan 250014, China)

[4](Inner Mongolia University, Hohhot 010021, China)

[5](Blockchain Lab, Xiong'an Intelligent City Innovation Federation, Baoding 071700, China)

Abstract: Blockchain is the basis of the Internet of value. However, data and value silos arise from independent blockchain systems. Blockchain interoperability (also known as cross-chain operability) is essential for breaking inter-chain barriers and building a blockchain network. After differentiating between the blockchain interoperability in the narrow sense and that in the broad sense, this study redefines the former concept and abstracts out two primary operations: cross-chain reading and cross-chain writing. Subsequently, it summarizes three key technical problems that need to be resolved for achieving the blockchain interoperability in the narrow sense: cross-chain information transmission, cross-chain trust transfer, and cross-chain operation atomicity guarantee. Then, the study reviews the current research status of the three problems systematically and makes comparisons from multiple perspectives. Furthermore, it analyzes some

* 基金项目: 国家重点研发计划 (2021YFB2700404); 国家自然科学基金 (U22B2032, 61972382); 内蒙古自然科学基金 (2020MS06017); 河北省重点研发计划 (20310105D); 未来区块链与隐私计算高精尖创新中心项目 (GJJ-22-025)
收稿时间: 2022-10-17; 修改时间: 2023-01-13; 采用时间: 2023-04-06; jos 在线出版时间: 2023-09-06
CNKI 网络首发时间: 2023-09-07

representative holistic solutions from the perspective of the key technical problems. Finally, several research directions deserving of further exploration are also presented.

Key words: blockchain; blockchain interoperation; crosschain; atomicity

区块链是一种以安全、可验证、透明的方式在对等网络上存储交易的分布式账本[1], 自 2008 年被提出以来, 经过两个阶段的发展, 从概念走向应用, 逐步与实体经济融合, 开始助力经济社会发展, 被看作是构建未来价值互联网的重要支撑技术. 依据区块链技术的应用范围可以将其大致划分为两个阶段.

(1) 区块链 1.0 阶段, 以分布式账本为标志性技术. 区块链技术起源于比特币[2], 首次从技术上突破了中心化信任模式的桎梏, 解决了在一组互不信任的多方之间实现一致的状态转移问题, 从而在不可信多方间实现了可信的信息/价值流通的应用范式. 本阶段, 区块链技术对业务逻辑的实现能力有限, 因此其应用主要局限在数字货币领域.

(2) 区块链 2.0 阶段, 以支持图灵完备的智能合约为标志性技术. 图灵完备的智能合约技术使得区块链具备了实现上层业务逻辑及承载部分垂直行业应用的能力, 通过与多种信息化技术的集成重构, 在金融、民生、政务等不同行业催生了新的商务及管理模式, 优化了现有生产关系, 出现了用于实际生产环境的应用方案, 开始从不同角度助力经济社会的发展.

经过前两阶段的发展, 基于异构平台实现的面向不同领域的上层应用, 在既有用户和价值积累的基础上, 产生了与其他区块链及区块链应用交互的外延需求. 如何在不破坏区块链去中心、去信任特性的前提下融合异构的底层技术平台, 打破上层应用边界, 成为当今区块链技术发展的迫切需求, 因此, 跨链技术应运而生.

跨链技术是指跨越单一区块链系统的数据可信边界 (共识机制作用范围), 实现区块链互操作, 即互不影响的两个或多个区块链间的有效协作, 进而实现可信的信息/价值跨链流通的技术.

区块链本质上是基于点对点分布式账本技术实现的多副本状态机, 区块链系统的链内操作实际上是对区块链状态的操作. 具体而言, 区块链共识节点基于共识算法对交易序列达成一致, 并生成不可篡改的区块链账本; 区块链全节点 (状态机) 各自在本地维护着区块链状态, 其基于一致的初始状态 (创世区块), 通过按序执行由区块链账本所记录的一致的交易, 实现一致的状态更新, 而全局一致的状态更新则承载了上层应用逻辑的实现. 区块链系统的链内操作可以抽象为不更改区块链状态的读操作和更改区块链状态的写操作两类基本操作.

与链内操作一样, 跨链操作同样可以抽象为跨链读操作与跨链写操作两类基本操作, 但与链内操作不同的是, 跨链操作的对象在另一个区块链系统内, 跨链操作会被拆分为多个区块链链内操作. 为了实现跨链读、写操作, 需要解决几个关键的技术问题.

(1) 跨链信息传输, 即将一个区块链系统的状态数据或者账本数据传送到另一个区块链系统, 其中, 前者被称作源区块链, 后者被称作目的区块链.

(2) 跨链信任传递, 即目的区块链确认所接收的跨链数据已在源区块链中达成一致.

(3) 跨链操作原子性保障, 即保证跨链操作在各区块链上的链内操作同时执行或者同时不执行.

早期跨链研究基于具体的应用需求, 如跨链验证、跨链资产原子交换、跨链资产转移等, 设计解决方案. 根据应用需求的不同, 研究重心也有差异: 跨链验证研究主要解决跨链信息传输与信任传递问题[3−6]; 跨链代币兑换、跨链代币原子交换研究主要解决跨链操作原子性保障问题[7−12]. 随着对跨链需求的深入认识, 跨链研究转向系统级的解决方案, 在设计跨链架构的基础上提出兼顾上述 3 种关键技术问题的整体解决方案[13−16].

目前已有多项综述工作针对跨链研究进行了梳理, 但是其聚焦于跨链整体解决方案, 缺乏技术体系层面系统的分析与梳理. Buterin 对跨链方案进行了分类与回顾, 将跨链方案分为公证人、侧链/中继与哈希时间锁 3 类[17]; Belchior 等人在定义区块链互操作框架与标准的基础上, 对跨链方案进行了更全面的分类与回顾, 将区块链互操作分为公共连接器、区块链的区块链和混合连接器 3 类, 其中公共连接器包含 Buterin 整理的 3 类方案[18]; Singh 等人[19]与 Johnson 等人[20]的工作聚焦于侧链技术, 对相关研究与项目进行了回顾与分析; Zamyatin 等人[21]与 Robinson[22]从跨链通信的角度分析和综述了已有方案; Schulte 等人则是从跨链资产转移与跨链合约调研两类跨链应用的角度分析和综述了已有方案[23]. 为了帮助后续研究者更深入地理解跨链, 从而更快地加入这一领域的研究, 本文首次系统性地分析了跨链需要解决的关键技术问题, 并综述总结了各个关键技术问题的研究现状, 为读者

分析可行的跨链方案打下基础; 同时从关键技术问题的角度分析典型的跨链整体解决方案, 以便读者更好地理解并评价现有研究工作.

本文第 1 节在定义区块链互操作的基础上, 系统性地分析区块链互操作需解决的跨链信息传输、跨链信任传递、跨链操作原子性保障 3 种关键技术问题. 第 2–4 节针对各关键技术问题, 分别综述分析现有研究工作. 第 5 节对代表性的跨链整体解决方案从关键技术问题的角度进行分析与讨论. 最后, 在第 6 节指出跨链研究领域尚待解决的问题与挑战.

1 跨链问题分析

1.1 区块链互操作定义

跨链技术发展过程中, 不同研究团队对区块链互操作有不同诠释, 具体可以分为广义和狭义两类. 广义的区块链互操作包括区块链与区块链的互操作 (也被称为跨链操作) 以及区块链与其他信息系统的互操作, 中国信息与通信技术研究院区块链研究团队认为区块链互操作是指区块链系统实例与其他区块链系统实例及所有区块链外部系统实例交换信息, 并对所交换信息加以使用[24]. 狭义的区块链互操作仅包括不同区块链系统间的互操作, Jin 等人认为跨链互操作定义是在保留区块链自身不可逆转、可追溯等特性的同时, 实现不同区块链系统间有效的通信与直接的信息交换[25], Lafourcade 等人认为跨链互操作是两个区块链系统协同工作[26].

本文仅针对狭义区块链互操作相关研究进行综述, 将区块链互操作定义为跨越区块链数据可信边界 (共识机制作用范围), 在独立区块链系统间实现的、与区块链链内操作效果一致的可信信息获取与协同状态更新.

由于区块链本质上是基于点对点分布式账本技术实现的多副本状态机, 区块链的链内操作实际上是对区块链状态的操作, 因此链内可以抽象为不更改区块链状态的读操作和更改区块链状态的写操作两类基本操作.

(1) 读操作

读操作是指读取区块链数据, 包括读取区块链状态数据 (如读取账户余额) 和读取区块链账本数据 (如读取某条交易信息). 由于所有区块链全节点均在本地维护了区块链状态, 而读操作不更改区块链状态, 因此区块链的读操作可以仅在一个区块链全节点上执行, 而不需要所有区块链全节点均执行. 当任意账户存在读需求时, 一般通过向一个或多个可信的区块链全节点发起请求, 通过获取这些区块链全节点的本地执行结果获取所需数据.

(2) 写操作

写操作是指更改区块链状态数据 (如更新账户余额). 由于写操作需要更改区块链状态, 而区块链的状态由系统内所有全节点在各自本地独立维护, 因此, 当任意账户存在写需求时, 一般以交易的形式将该写操作打包至区块链账本. 所有全节点通过执行打包进账本中的交易, 在本地执行该写操作, 从而一致地完成对区块链状态的指定更改.

与链内操作一样, 跨链操作同样可以抽象为不更改区块链状态的跨链读操作和更改区块链状态的跨链写操作两类基本操作, 而与链内操作不同的是, 跨链操作的对象在另一个区块链系统内, 一个跨链操作会被拆分为多个交互区块链链内的子操作.

(1) 跨链读操作

跨链读操作是指一个区块链系统 $Chain_1$ 读取另一个区块链系统 $Chain_2$ 的数据, 包括读取区块链系统 $Chain_2$ 状态数据和账本数据. 例如, Alice 在司法区块链 $Chain_1$ 上读取存证区块链 $Chain_2$ 上指定的存证信息 x 用于案件判决, 该跨链读操作被拆分为:

- ACTION $Chain_1$: Alice 发起对存证区块链 $Chain_2$ 上存证信息 x 的跨链读请求并获取存证信息 x.
- ACTION $Chain_2$: 读取存证信息 x 并将该信息返回给区块链 $Chain_1$.

(2) 跨链写操作

跨链写操作是指一个区块链系统 $Chain_1$ 更改另一个区块链系统 $Chain_2$ 的状态数据, 包括跨链原子写操作与跨链非原子写操作.

跨链原子写操作包括跨链研究中常见的资产原子交换、跨链资产转移等, 例如: Alice 与 Bob 在区块链

$Chain_1$ 与区块链 $Chain_2$ 上均拥有账户, Alice 在区块链 $Chain_1$ 上有 10 个代币, Bob 在区块链 $Chain_2$ 上有 5 个代币, Alice 通过跨链写操作使用其在区块链 $Chain_1$ 上的 10 个代币交换 Bob 在区块链 $Chain_2$ 上的 5 个代币, 该跨链原子写操作被拆分为:

- ACTION $Chain_1$: Alice 发起与 Bob 进行代币交换的请求, 并将 10 个代币转移给 Bob.
- ACTION $Chain_2$: Bob 将 5 个代币转移给 Alice.

跨链原子写操作的正确执行需要保证子操作的原子性, 即需要保证 ACTION $Chain_1$ 与 ACTION $Chain_2$ 同时执行或者同时不执行.

跨链非原子写操作包括跨链信息同步等, 例如: 区块链 $Chain_1$ 记录了信息 x, 区块链 $Chain_2$ 记录了信息 x 的备份, 区块链 $Chain_1$ 中信息更新后, 跨链更新区块链 $Chain_2$ 中的备份信息, 该跨链写操作被拆分为:

- ACTION $Chain_1$: 区块链 $Chain_1$ 发起对区块链 $Chain_2$ 上信息 x 的跨链更新请求.
- ACTION $Chain_2$: 更新信息 x.

跨链非原子写操作不需要保证子操作的原子性, 若 ACTION $Chain_2$ 因为网络拥塞等原因没有执行, ACTION $Chain_1$ 不需要撤销更新操作.

现有跨链研究根据跨链操作的实现方式可以分为链上触发模式和链下触发模式两种, 链上触发模式在交互区块链上发起跨链请求、解析跨链请求并根据解析结果触发对应的子操作[13−16,27]. 链下触发模式在链下解析跨链需求、拆分跨链操作, 并在此基础上通过按序触发不同区块链的子操作达到预期跨链效果[13,17,28]. 链下触发模式的核心逻辑均在链下, 链上子操作彼此独立, 区块链仅作为资源使用.

1.2 跨链关键问题

跨链读、写操作的实现需要解决 3 个层面的关键问题: (1) 解决区块链系统间信息传输的问题, 实现链间信息互通. (2) 解决区块链系统间信任传递的问题, 保证链间信息可信. (3) 解决跨链操作原子性保障的问题, 保证相互依赖的跨链子操作一致执行或者一致不执行.

(1) 跨链信息传输

跨链读、写操作可以拆分为各交互区块链内部的子操作, 由彼此独立的各交互区块链协同工作实现, 协同工作的基础是必要的信息互通. 但是区块链获取数据需要区块链系统内的所有节点都获取一致的数据, 而即使在外部数据源与区块链网络连通的情况下, 由于无法保证区块链系统中的每一个节点对该外部数据源具有相同的访问权限和视图, 因此无法仅依靠网络传输实现区块链对外部数据的获取.

因此, 为了实现跨链读、写操作, 首先需要解决跨链信息传输问题, 保证区块链系统可以获取其他区块链系统的数据. 例如在上述跨链读操作的例子中, 司法区块链 $Chain_1$ 需要能够获取存证区块链 $Chain_2$ 内部的存证信息 x.

跨链信息传输是指将一个区块链系统的状态数据或者账本数据传送到另一区块链系统中. 为了更好地描述跨链信息传输, 本文将一次跨链传输中发起跨链信息传输的区块链系统称为源区块链, 将接收跨链信息的区块系统链称为目的区块链.

(2) 跨链信任传递

区块链的数据是指区块链系统内所有节点达成一致的数据, 由于区块链共识机制只能保证共识范围内节点对区块链账本数据与状态数据达成一致, 而协同完成跨链操作的区块链彼此独立, 区块链无法直接确认其获得的对端区块链账本数据与状态数据的可信性. 在上述跨链读操作中, 司法区块链 $Chain_1$ 获取存证信息 x 后并不能确定这是在存证区块链系统 $Chain_2$ 中达成了一致的数据.

因此为了实现跨链读、写操作, 在跨链信息传输的基础上, 还需要打破区块链可信性的边界, 实现区块链之间的信任传递问题, 让司法区块链 $Chain_1$ 的节点可以确认通过跨链传输获取的区块链 $Chain_2$ 状态数据或者账本数据的可信性.

跨链信任传递是指在跨链传输的基础上, 实现目的区块链对跨链信息的可信性确认, 即确认所获取的跨链信息已在源区块链中达成一致.

(3) 跨链操作原子性保障

读操作与写操作作为区块链系统的元操作, 其执行只有成功与失败两种状态, 若操作执行失败, 则区块链系统需要回滚至操作执行前的初始状态. 由于读操作在执行过程中并不会更新区块链状态, 因此只有写操作在执行失败时会进行区块链状态的回滚.

跨链读、写操作的执行也只有成功与失败两种状态, 跨链写操作执行失败时, 跨链系统需要进行状态的回滚. 跨链系统的状态是交互区块链状态的并集, 一次跨链写操作可能导致目的区块链或者目的区块链与源区块链的状态更新. 若跨链写操作同时涉及目的区块链与源区块链的状态更新, 则在跨链写操作执行失败时, 需要同时回滚目的区块链与源区块链上已有的状态更新. 在上述资产交换例子中, 若 ACTION $Chain_1$ 正确执行, 将 Alice 的 10 个代币转移给了 Bob, 而 ACTION $Chain_2$ 由于网络拥塞、攻击等原因执行失败, 则 Bob 需要在区块链 $Chain_1$ 上将已收到的 10 个代币退还给 Alice, 否则将会导致 Alice 损失 10 个代币, 不符合资产交换成功或者失败的两种预期结果.

因此为了实现跨链写操作, 面向跨链转账、跨链资产交换等同时涉及源区块链与目的区块链状态更新的场景, 还需要保证跨链操作的原子性, 这类写操作就是跨链原子写操作.

跨链操作的原子性是指拆分在多个区块链系统内执行的子操作要么全部执行成功, 要么均不执行.

2 跨链信息传输关键技术

跨链信息传输是指将一个区块链系统的状态数据或者账本数据传送到另一区块链系统中. 由于区块链系统是一个封闭的系统, 无法主动向外部系统发送数据, 只能记录向外部系统发送特定数据的意图[29], 因此跨链信息传输方案一般通过引入链下转发实体实现区块链系统间的数据转发.

跨链信息传输的基本流程可以总结抽象为图 1, 主要包括获取跨链信息与转发跨链信息两个阶段.

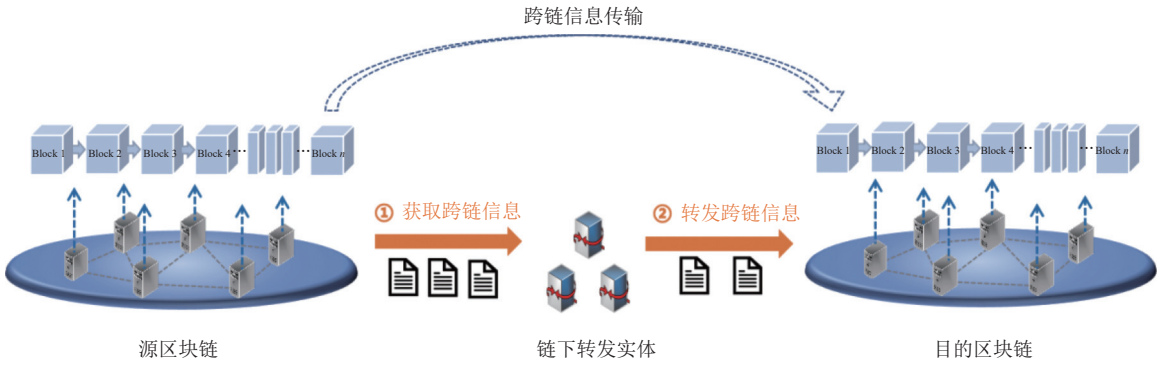

图 1 跨链信息传输示意图

(1) 获取跨链信息

链下转发实体通过监听源区块链获取待传输的跨链信息. 具体地, 链下转发实体监听源区块链记录跨链信息的对象, 包括交易、收据以及状态. 通过交易记录跨链信息是指用户在源区块链发起交易时将跨链信息直接存储在交易的某些字段中[6,30−32]; 通过收据记录跨链信息是指源区块链在处理用户跨链请求时, 以事件的形式将用户的跨链意图存储在区块链的收据日志中[14,15,33,34]; 通过状态记录跨链信息是指源区块链基于智能合约将跨链信息存储在区块链状态中[15].

(2) 转发跨链信息

链下转发实体将待传输的跨链信息转发至目的区块链. 链下转发实体一般通过在目的区块链上发起包含跨链信息的交易 (后文称为转发交易) 实现消息转发, 这是因为区块链作为多节点组成的系统, 需要保证系统中的所有节点均获取相同的信息, 而基于共识机制可以对要执行的交易 (即待处理的跨链操作) 达成一致.

2.1 跨链信息传输方案

BTC Relay[3]在以太坊上构建比特币轻客户端, 使得以太坊具备验证比特币交易的能力, 其在传输层面实现了两类信息从比特币到以太坊的单向跨链传输: 一类是比特币交易信息, 通过用户监听、转发特定交易实现; 一类是比特币区块头, 通过基于激励机制引入的链下转发节点 Relayer 监听、转发比特币区块头实现, 为以太坊上比特币交易的验证提供基础. 在 BTC Relay 中, 用户在以太坊上验证比特币交易需要向提交交易所在区块头的 Relayer 支付手续费, 从而激励 Relayer 参与跨链传输与 BTC Relay 的构建. BTC Relay 支持多 Relayer, 若多个 Relayer 向以太坊转发同一高度的比特币区块头, 仅第 1 笔被以太坊打包的转发交易生效, 以保证跨链信息仅被处理一次, 即保证跨链传输的幂等性. Peace Relay[4]、ETH Relay[5]仿照 BTC Relay 设计、实现了类以太坊区块链之间的跨链交互. RSK[6]、Loom[35]等侧链研究同样基于用户实现跨链信息在主链、侧链之间的传输.

Cosmos、BitXHub、Polkadot 等方案针对跨链信息传输问题提出了专门的传输协议.

Cosmos 提出的区块链间通信协议 (IBC 协议) 是一种端到端、面向连接、有状态的协议, 实现了独立分布式账本上模块之间的可靠、有序和认证通信[29]. IBC 协议抽象了轻客户端、连接 (connection)、通道 (channel) 以及转发节点 (Relayer). 轻客户端抽象封装验证区块链数据的属性及方法, 为跨链信任传递提供支撑. 连接维护区块链间的关联状态, 与轻客户端一起实现跨链信任传递. 通道维护不同区块链的应用模块间信息传输的状态, 提供有序传输与无序传输两种模式, 其中有序模式基于 IBC 数据包序列号与跟踪传输状态实现, 并基于 IBC 数据包的超时时间提供丢包检测功能. 转发节点完成区块链间的信息传输, 与 BTC Relay 类似, Cosmos 通过激励机制引入转发节点, 但其核心协议中不包括对应激励机制, 由应用提供具体激励机制. 同样地, 当多个转发节点转发同一跨链信息时, 为了避免目的区块链重复处理, 仅最先提交的生效. Cosmos 的跨链数据包括两类: 一类是记录了应用跨链信息的 IBC 数据包, 一类是用于跨链信息验证的区块头. 特别地, 在获取跨链数据阶段, Cosmos 支持监听事件与通道状态两种方法; 在转发跨链数据阶段, 转发节点可以将多个 IBC 数据包打包在一笔目的链交易中.

BitXHub 提出的链间消息传输协议 (IBTP 协议) 是一种类似 TCP/IP 的链间传输协议, 其同样通过引入转发节点 (跨链网关, Pier) 实现了记录了跨链信息的 IBTP 数据包的传输[14,36]. IBTP 在不同的区块链交互模式, 跨链传输的实现方式不同.

● 区块链直接进行交互时, 源区块链跨链网关通过 P2P 网络将 IBTP 数据包传输给目的区块链跨链网关, 再由目的区块链跨链网关将 IBTP 数据包以交易的形式提交给目的区块链.

● 区块链借助中继链进行交互时, 跨链传输需要通过多跳传输实现: 源区块链跨链网关将 IBTP 数据包以交易的形式提交给中继链, 再由目的区块链跨链网关将 IBTP 数据包以交易的形式提交给目的区块链. 在大规模跨链场景下, 还需要实现 IBTP 数据包在中继链间的传输.

与 Cosmos 在区块链上实现跨链传输的有序性不同的是, BitXHub 在跨链网关上借助缓冲池对 IBTP 数据包的排序实现跨链传输的有序性. 同时, BitXHub 考虑到单跨链网关可能产生恶意行为, 采用跨链网关集群的方式增强跨链网关可信度, 并设计了主备节点模式保证交互区块链间只有一个活跃的主节点处理跨链请求, 从而避免跨链信息的重复传输. 备用节点在主节点宕机时主动升级为主节点处理跨链请求.

Polkadot 针对区块链在可扩展性、可伸缩性、安全性等方面普遍存在的问题, 分离共识架构中的一致性和有效性, 由各平行链并行出块, 中继链统一共识[16]. 为此, Polkadot 设计了收集者角色收集平行链交易, 打包平行链区块. Polkadot 设计的平行链间跨链信息传输协议 XCMP 也需要借助各平行链收集者实现连接同一中继链的平行链间的信息传输[37]. 具体地, 源区块链收集者将跨链信息、目的链与时间戳一起放入源区块链的出队列, 目的区块链收集者通过定期向源区块链收集者请求跨链数据获取目的链与其匹配的跨链信息, 并将其放入目的区块链入队列, 收集者出块时将出、入队列信息打包在内, 从而实现跨链信息传输. XCMP 的目标是高效、有序、可验证、无遗漏的传输, 但是由于目前仍在设计阶段, 具体的实现方式还不明确. 由于 XCMP 仍在设计实现阶段, 目前平行链通信使用资源开销更大的水平中继路由信息传输 (HRMP) 借助中继链通过多跳传输实现.

WeCross[13]、HTLC[8]等方案尽管实现了跨链交互, 但是并没有直接实现区块链间的信息传输, 而是通过链下触发的方式, 由目的区块链的特殊用户/转发节点接收跨链信息并触发目的区块链上对应的操作.

WeCross 借助由部署区块链的机构为区块链搭建的、可信的跨链路由 (对应转发节点) 实现链下触发. 如图 2 所示, 源区块链跨链路由通过监听源区块链输出队列获取记录了目标区块链资源路径、目标接口、调用参数等信息的跨链操作请求, 并通过网络将跨链操作转发给目的区块链的跨链路由, 实现跨链信息从源区块链到目的区块链跨链路由的传输. 目的区块链跨链路由根据获取的跨链操作请求在目的区块链上触发对应操作.

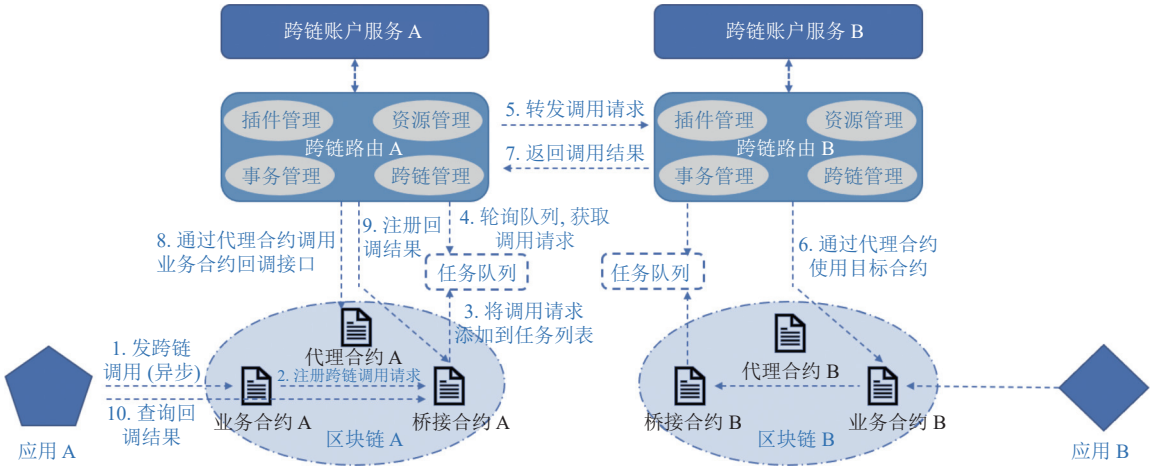

图 2 WeCross 交互示意图[13]

HTLC 借助参与跨链交互的用户实现链下触发, 特别地由于用户同时在源区块链上与目的区块链上, 因此没有引入额外的角色进行跨链信息传输, 而是由用户自行监听源区块链获取跨链信息并触发目的区块链对应操作.

Xiao 等人[38]、Luo 等人[39]、Belchior 等人[40]提出的方案同样引入转发节点实现链下触发. 其中, Luo 等人[39]提出的方案还引入了路由区块链, 用于维护区块链的路由节点信息 (对应转发节点), 从而支撑源、目的区块链路由节点间的通信. Belchior 等人进一步提出了容忍崩溃的网关节点 Hermes[41], 支持网关节点自恢复和主备两种崩溃恢复方法.

2.2 跨链信息传输方案评价

(1) 评价指标

根据本节汇总的跨链信息传输方法, 我们整理得到表 1, 从转发节点安全假设、幂等性、传输开销、可扩展性、有序性以及可靠性几个方面进行分析与评价.

表 1 跨链信息传输方案评价

文献	转发节点安全假设	幂等性	传输开销	可扩展性	有序性	可靠性			
						转发节点数量	转发节点奖励机制	转发节点恢复机制	跨链传输丢包处理
[3−5]	拜占庭	可保证	高	低	不支持	多个	用户向转发节点支付服务费	无	不支持
[14]	拜占庭	可保证	低	低	支持	多个 (主备)	无	无	不支持
[15]	拜占庭	可保证	高	高	支持	多个	传输协议不包含激励机制, 应用模块提供	无	仅支持丢包检测
[16]	拜占庭	可保证	低	低	支持	多个	传输协议不包含激励机制, 平行链提供	无	状态根保证
[13]	诚实	可保证	低	低	不支持	单个	无需激励	无	不支持
[38]	诚实	可保证	低	低	不支持	单个	无	无	不支持
[39]	诚实	可保证	低	低	不支持	单个	无	无	不支持
[41]	诚实	可保证	低	低	不支持	多个 (主备)	无	基于日志的主备节点恢复	不支持

1) 转发节点安全假设: 安全假设包括诚实节点假设以及拜占庭节点假设. 诚实节点假设下, 节点只存在崩溃问题; 拜占庭节点假设下, 节点还存在伪造跨链信息、丢弃跨链信息等问题.

2) 幂等性: 区块链系统不会重复处理重复提交的跨链信息.

3) 传输开销: 一次跨链信息传输在目的区块链上的计算、存储开销, 若交互区块链间存在多个转发节点且均执行转发操作, 则传输开销包含所有转发节点在目的区块链上的计算、存储开销.

4) 可扩展性: 即跨链信息传输方案扩展至其他、多元应用的能力.

5) 有序性: 目的区块链系统按照源区块链系统发送跨链信息的顺序进行处理.

6) 可靠性: 确保跨链信息能够被转发到目的区块链上, 若跨链信息无法被转发到目的区块链, 源区块链能够及时得知. 我们面向转发节点可靠性与传输协议可靠性两个角度, 分别从转发节点数量、转发节点奖励机制、转发节点恢复机制与跨链传输丢包处理几个方面进行评估.

a) 转发节点数量: 交互区块链间用于转发跨链信息的节点数量 (包括主备节点), 用于评估跨链传输方案对转发节点崩溃问题的容忍程度. 跨链传输最基本的需求是交互区块链间至少有一个诚实的转发节点, 转发节点越多, 对转发节点崩溃问题的容忍程度越高.

b) 转发节点奖励机制: 转发节点提供跨链传输服务的动力. 跨链传输服务需要大量成本, 转发节点缺乏提供跨链传输服务的动力, 因此需要额外引入奖励机制, 保证交互区块链间有足够的转发节点提供跨链传输服务.

c) 转发节点恢复机制: 转发节点崩溃宕机后的恢复机制, 用于评估跨链传输方案对转发节点崩溃问题的容忍程度.

d) 跨链传输丢包处理: 包括丢包检测方法与对丢包的恢复方法, 用于评估跨链传输方案对丢包问题的.

(2) 方案对比

● 幂等性方面: 幂等性是跨链交互对跨链传输的基本要求, 若跨链信息被重复转发至目的区块链并被重复处理, 将影响跨链交互的正确性. BTC Relay[3]、Peace Relay[4]、ETH Relay[5]、Cosmos[15]这几种方案通过目的区块链识别并忽略重复的跨链信息保证跨链传输的幂等性, 其他方案通过可靠的单一转发节点或者主备节点保证跨链信息仅被传输一次. 现有方案均能保证幂等性.

● 转发节点安全性假设方面: WeCross [13]等基于代理传输的方案均采用诚实节点假设, 而 BTC Relay 等基于链上传输的方案均采用拜占庭节点假设, 与前者相比在实现可靠传输时需要考虑更多的节点可靠性问题.

● 传输开销方面: BTC Relay、ETH Relay、Peace Relay、Cosmos 几种方案交互区块链间有多个转发节点处理跨链信息, 目的区块链会接收多笔该跨链信息的转发交易, 因此传输开销高. 其他方案目的区块链只会接收到一笔该跨链信息的转发交易, 因此传输开销低.

● 可扩展性方面: 仅 Cosmos 的 IBC 协议采用松耦合的设计模式, 区分链间连接与应用间通道, 支持多个通道对连接的复用, 使得应用层面的扩展无需关注链间连接, 可扩展性较高.

● 有序性方面: Cosmos、BitXHub 与 Polkadot[16]支持有序传输, 其中 Cosmos 基于通道内传输的 IBC 数据包序列号以及消息确认机制, 保证跨链信息传输的有序性; BitXHub 在跨链网关上借助缓冲池对 IBTP 数据包的排序实现跨链传输的有序性, Polkadot 则是通过输入输出队列保证跨链信息传输的有序性.

● 可靠性方面: 1) 采用诚实节点假设的方案. 仅 Hermes[41]在交互区块链间引入了多个转发节点, 且针对转发节点可能存在的崩溃宕机问题设计了基于日志的恢复机制, 可以容忍转发节点的崩溃, 其他方案在转发节点崩溃或者被攻击时将无法提供跨链传输服务. 除 WeCross 转发节点由部署区块链的机构搭建, 无需考虑动力问题外, 其他方案均不提供转发节点奖励机制, 在实际应用中可能存在可用性问题. 上述方案均未设计丢包检测机制与丢包恢复机制, 无法处理因目的区块链拥塞等原因导致丢包问题. 2) 采用拜占庭节点假设的方案. 上述方案均在交互区块链间引入了多个转发节点, 但是均未设计转发节点崩溃恢复机制, 对崩溃问题的容忍程度与网关节点数量相关. 除 BitXHub 外, 其他方案均有对转发节点奖励机制的讨论, BTC Relay、ETH Relay、Peace Relay 针对转发节点设计了跨链服务奖励, 但是缺乏对激励机制有效范围与可行性的进一步讨论与分析, 在实际应用中存在可用性问题: BTC Relay 中 Relayer (对应转发节点) 的跨链服务奖励远小于其成本, 项目上线一年后所有 Relayer 都撤出平台; Cosmos 与 Polkadot 的核心机制并不包括对转发节点的激励, 而是将这部分工作留给了应用与平行链. 仅

Cosmos 与 Polkadot 提供了丢包处理方法, 其中 Cosmos 仅在有序模式下支持丢包检测, 其基于 IBC 数据包超时时间实现, 若出现丢包则关闭通道; Polkadot 基于中继链上存储的状态根, 验证跨链数据的完整性, 从而避免丢包; 其他方案即使转发节点向目的区块链发送了跨链信息, 也可能因为目的区块链拥塞等原因而“丢包”.

3 跨链信任传递关键技术

跨链信任传递是指在跨链传输基础上, 实现目的区块链对跨链数据可信性的确认, 保证跨链数据已在源区块链中达成共识并写入主链. 其中, 跨链数据包括源区块链账本数据及状态数据: 账本数据一般指交易, 验证源区块链账本数据的可信性本质上是验证源区块链交易的存在性, 即一笔交易是否被写入源区块链主链; 状态数据是各区块链节点在本地维护的全局一致的区块链状态.

为了确保目的区块链的所有节点对跨链数据达成一致, 最需要关注的两个问题.

(1) 由谁来执行跨链数据验证?

(2) 如何实现跨链数据验证?

下文将围绕上述问题, 从跨链数据验证主体以及跨链数据验证方法两个角度分别综述当前研究, 并进行总结评价. 为实现对跨链研究的思路整理及扩展, 所综述的方案不局限于各跨链方案, 还包括其他区块链研究中可用于跨链信任传递的相关技术.

3.1 跨链数据验证主体

目的区块链对跨链数据的验证, 既可以通过目的区块链自行验证跨链数据实现, 也可以通过目的区块链确认跨链数据的验证结果实现. 前者在目的区块链链上进行跨链数据验证, 后者在链下进行跨链数据验证, 根据验证发生的位置可以将二者分别称为链上验证模式与链下验证模式.

(1) 链上验证模式

链上验证模式中, 目的区块链将自行验证跨链数据可信性, 即目的链区块链节点各自完成跨链数据验证并对验证结果进行共识最终达成一致. 链上验证模式的跨链信任流可以抽象为如图 3, 信任从源区块链直接传递至目的区块链.

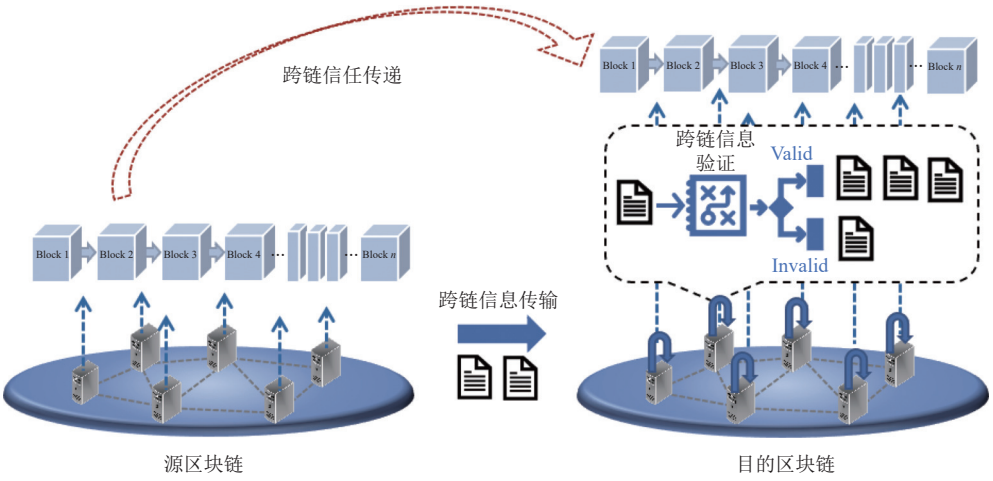

图 3 跨链信任传递链上验证模式示意图

链上验证模式一般以在目的区块链上部署源区块链数据验证智能合约的形式实现. 例如: BTC Relay 在以太坊上部署智能合约实现对比特币区块头与交易的验证[3]; Cosmos 的 IBC 协议抽象出客户端模块实现对对端区块链的验证[29].

链上验证模式下验证算法公开透明, 安全程度高. 然而, 该方法只适用于支持智能合约的区块链, 整体性能与开销取决于源块链数据验证算法, 若源区块链验证算法复杂, 则整体性能较低且需要大量的链上计算资源与存储

资源: ETH Relay 指出在类以太坊区块链上实现以太坊轻客户端的开销极大, 即使是优化后的验证方案, 仅以太坊共识主体的验证就需要大约 300 万 gas (一笔普通转账交易大约需要 21 000 gas)[5].

(2) 链下验证模式

链下验证模式中, 目的区块链将跨链数据可信性验证委托给一个或者一组链下代理, 目的区块链节点仅确认链下代理的验证结果. 链下验证模式的信任流可以抽象为图 4, 源区块链与目的区块链间的信任传递由源区块链与链下代理间的信任传递以及链下代理与目的区块链间的信任传递组合实现.

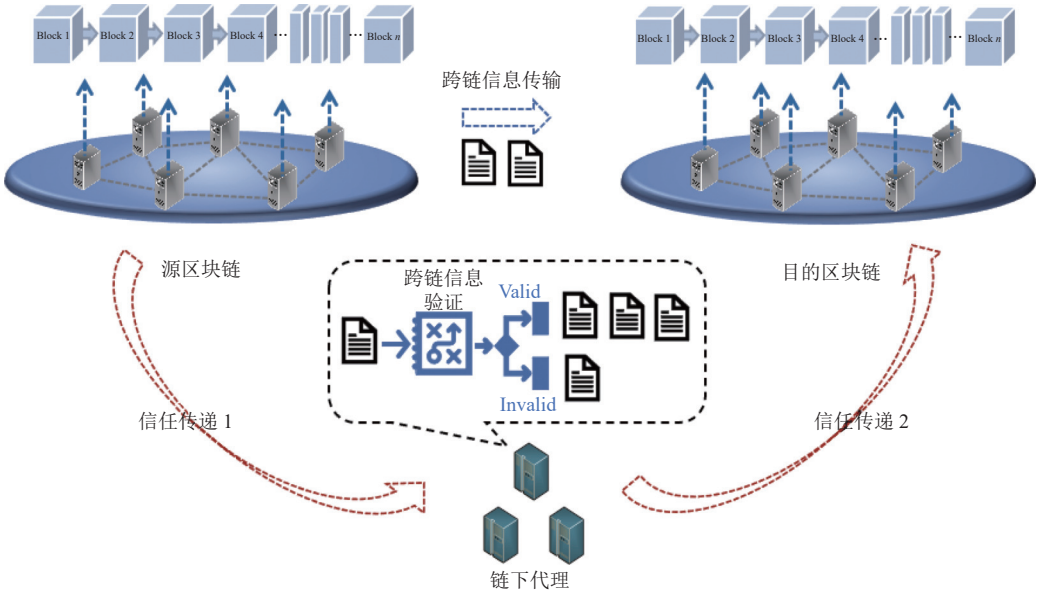

图 4 跨链信任传递链下验证模式示意图

链下代理的选择将影响链下代理与目的区块链间信任传递的方式, 根据目的区块链与链下代理信任传递构建的原理, 可以将链下验证模式分为基于身份的链下验证模式以及基于行为验证的链下验证模式.

1) 基于身份的链下验证模式

在基于身份的链下验证模式中, 目的区块链信任链下代理, 仅需验证收到的跨链信息是否有正确的链下代理背书. 其中, 链下代理可以是单点也可以是多点.

WeCross 的链下代理是区块链部署机构所搭建的跨链路由, 区块链信任其对应跨链路由[13,42]. 跨链路由代替目的区块链完成跨链数据验证后触发目的区块链子操作, 目的区块链仅需检测跨链子操作由跨链路由发起. 该方案链上操作简单, 整体性能较高, 但是仅面向联盟链场景, 需要信任单一的跨链路由, 存在单点故障风险.

Loom 的链下代理是网关 Oracle, 其一般运行在参与区块链 DPoS 共识的委托人节点上[35]. 当代币从侧链撤回至主链时, 网关 Oracle 基于用户提交的跨链交易与相关证明进行验证, 并为验证通过的交易签名, 主链仅需检测跨链交易具有网关 Oracle 签名. 与 WeCross 类似, 该方案引入了中心化信任的网关 Oracle, 同时网关 Oracle 的处理将会影响跨链交互的性能.

Drivechain[43]的链下代理是参与联合挖矿的矿工节点, 同时为源区块链与目的区块链出块的矿工组在目的区块链上对跨链数据的可信性进行投票, 实现目的区块链对跨链数据的确认. 该方案在不支持智能合约的区块链上也适用, 同时通过多矿工投票避免了单点故障. 但是 Drivechain 的安全性仅由联合挖矿矿工保证, 只适用于区块链强相关的场景 (主链与侧链), 无法满足独立区块链跨链交互的需求, 同时若联合挖矿算力不够高, 攻击者可以轻易地实现对 Drivechain 的攻击.

RSK[44]在不同阶段使用不同的链下代理: 初期使用可信联邦, 即多家社会中有较高声誉的组织、公司作为链下代理, 通过多签投票向目的区块链提供跨链数据背书; 随着参与联合挖矿算力的不断提高, 加入联合挖矿的矿工

节点作为链下代理, 并逐渐增大矿工投票比重直至最终只有矿工投票, 在此过程中去中心化程度不断提升.

基于身份的链下验证模式中, 目的区块链信任链下代理, 链上仅需验证链下代理的签名, 与链上模式相比方案易于实现、性能显著提升且链上开销低. 但是该模式引入了对链下代理的信任, 不适用于不存在可信链下代理的跨链场景.

2) 基于行为验证的链下验证模式

在基于行为验证的链下验证模式中, 目的区块链不信任代替它进行数据验证的链下代理, 而是信任链下代理的验证行为本身. 通过技术手段, 目的区块链确认链下代理的验证行为后, 信任可以传递到目的链. 当前主要基于可信执行环境与零知识证明技术保证链下代理行为可验证.

可信执行环境 (trusted execution enviroment, TEE) 是通过硬件隔离手段对运算和操作进行保护的技术, 在不破解硬件的前提下可以保证执行不可篡改. 可信执行环境中的代码是公开、可审计的, 通过提前审计以及可信执行环境签名可以保证输出是特定程序的执行结果[45].

Robinson 等人提出的跨链方案基于 TEE 实现链下代理[46]. 链下代理在 TEE 中部署源区块链数据验证方法, 并生成存储在 TEE 中的公私钥对. 目的区块链审计 TEE 中的程序, 并存储 TEE 公钥. 链下代理在 TEE 中完成跨链数据验证后, 使用 TEE 私钥对跨链数据验证结果进行签名背书. 目的区块链根据交易的 TEE 签名可以确认链下代理在 TEE 中执行了已审计的程序, 完成了对源区块链数据的验证.

此类方案保留链下验证高性能、低开销优势的同时通过代码审计以及硬件保证削减了对链下代理的信任, 但是由于可信执行环境内的认证密钥以及硬件隔离均由制造商提供, 转而需要信任制造商不会留有后门, 而由于可信执行环境存储空间有限, 链下代理的功能在一定程度上会被限制.

零知识证明技术是一种证明者在不向验证者提供任何有用信息情况下, 通过生成一个数学证明, 使得验证者相信对于某个待验证的问题, 其持有对应答案的技术[47].

Zkrelay 基于零知识证明实现链下验证[48], 其链下代理对源区块链跨链信息进行验证, 并在成功时生成证明其链下行为的零知识证明文件, 同时将该证明文件转发至目的区块链. 目的区块链基于提前存储的验证密钥以及证明文件可以确认跨链信息确实通过了链下代理的验证.

基于零知识证明技术构建链下代理无需引入任意程度的信任, 但是链下代理生成零知识证明文件的过程非常复杂, 而链上对于零知识证明文件的验证也比许多跨链数据验证的复杂度更高, 整体性能低.

3.2 跨链数据验证主体评价

(1) 评价指标

根据本节综述的各方案, 我们整理出表 2 对相关研究工作进行总结与对比, 其中的维度包括去中心化程度、信任锚点、开销、性能以及各自独特的优劣.

表 2 跨链数据验证主体评价

文献	验证主体分类	去中心化程度	信任锚点	开销		性能		其他
				存储开销	计算开销	吞吐量	时延	
[3]	链上	区块链	区块链共识	高	高	中	中	—
[4]	链上	区块链	区块链共识	高	高	低	高	未实现
[15]	链上	区块链	区块链共识	高	高	中	中	—
[13,42]	链下 (身份)	单节点	可信跨链路由	链上低 链下高	链上低 链上高	高	低	—
[44]	链下 (身份)	少数节点/ 区块链	联邦/ 区块链共识	链上低 链下高	链上低 链下低	高	中	局限性强, 只适用于挖矿节点同时为两条链工作的场景
[46]	链下 (行为)	单节点/ 少数节点	可信执行环境	链上低 链下高	链上低 链下高	高	低	TEE可能存在后门; 存储空间有限
[48]	链下 (行为)	单节点	密码学	链上低 链下高	链上高 链下非常高	高	高	可以通过降低验证、同步源链数据的频率, 降低开销

1) 去中心化程度: 即验证主体的去中心化程度, 用于描述方案抵御验证主体单点故障与联合作恶的能力.

2) 信任锚点: 即当前方案对验证主体的信任来源, 信任锚点的选择会影响跨链信任传递的安全性.

3) 开销: 包括存储开销与计算开销, 链下验证模式将分别标注链上与链下的存储、计算开销.

4) 性能: 包括吞吐量与时延, 其中吞吐量是指单位时间内完成验证的次数, 时延是指完成一次验证的时间, 为了能够横向比较不同验证主体的性能, 假定所有方案跨链信息验证复杂度一致.

5) 其他: 各方案独特的优缺点.

(2) 方案对比

● 去中心化程度方面: 链上验证方案去中心化程度与区块链相同, 去中心化程度较高. 链下验证方案中 RSK[41] 与 Robinson 等人的方案[46]引入了多个验证节点, 但是 RSK 后期依赖联合挖矿, 理论上若所有矿工均加入联合挖矿则其去中心化程度与区块链相当. WeCross[13,42]与 Zkrelay[48]仅引入单个验证节点, 若验证节点故障、被攻击或者作恶, 则无法提供链下验证功能, 甚至提供虚假的验证结果, 影响跨链操作安全性.

● 信任锚点方面: 链上验证方案的信任来源是区块链本身. 链下验证方案中, 基于身份的链下验证方案的信任来源是验证者本身, 其中, RSK 后期的验证者是联合挖矿矿工, 矿工是区块链信任构建的基础, 若 RSK 有足够的联合挖矿矿工, 则后期其信任来源等同于区块链; 基于行为验证的链下验证方案的信任来源是验证链下行为的技术, Robinson 等人的方案的信任源自 TEE 技术, 但是由于 TEE 很有可能存在厂商预留的后门, 因此一定程度上该方案还需要信任 TEE 厂商, 而 Zkrelay 的信任源自密码学, 无需其他额外信任.

● 开销方面: 链上验证方案需要在链上存储源链数据并完成验证, 一般计算开销与存储开销较大, 这极大制约了方案的落地. 链下验证方案中, 基于身份的链下验证方案与基于 TEE 技术的链下验证方案的链下部分承担了源区块链跨链数据存储与验证功能, 链上仅需验证签名, 开销远小于链上验证方案. 然而基于零知识证明的链下验证方案链下需要存储源链数据并基于密码学生成证明文件, 计算过程非常复杂, 计算开销显著高于其他方案, 一般通过批处理降低验证频率, 保证该方案的开销可被接受.

● 性能方面: 链上验证方案在区块链上进行跨链数据验证, 具体吞吐量与时延取决于区块链本身的设计, 采用确定性共识、模组化的 Cosmos[15]性能均优于采用概率性共识的类以太坊 (Peace Relay), 但是由于链上计算需要多节点共识, 整体性能均较低. 链下验证方案除 Zkrelay 外, 其他方案性能均优于链上方案, Zkrelay 链下证明生成过程非常复杂, 性能较低, 当跨链验证较为简单时, 其性能甚至不如链上验证.

3.3 跨链数据验证方法

跨链数据验证 (可信性验证) 是指验证跨链数据在源区块链中是否达成了一致, 其中跨链数据包括源区块链账本数据及状态数据. 账本数据一般指交易, 验证源区块链账本数据的可信性本质上是验证源区块链交易的存在性, 即一笔交易是否被写入源区块链主链. 状态数据是各区块链节点在本地维护的全局一致的区块链状态 (如: 比特币区块链的 UTXO 集合[2], 以太坊区块链的状态树[49]), 区块链节点通过按序执行账本中的交易完成本地维护的区块链状态的更新. 面向账本数据验证和状态数据验证两个问题, 下面将分别综述其验证方法.

(1) 账本数据验证

账本数据验证一般需要解决如下两个问题: 1) 交易打包验证, 即验证交易 Tx 是否被打包在区块 B_n 中. 2) 区块一致性验证, 即验证区块 B_n 是否在源区块链主链上.

1) 交易打包验证

验证跨链交易 Tx 是否被打包在区块 B_n 中最直接的方法是验证者查询交易 Tx 是否在区块 B_n 包含的全部交易序列$\{Tx\}$中, 然而这种方法需要验证者存储包含交易序列$\{Tx\}$的区块 B_n, 而一个区块可能包含几百甚至几千笔交易, 存储开销相对较大, 尤其是在链上验证模式中, 所有区块链全节点都是验证者, 整体存储开销随区块链规模线性增长.

BTC Relay、Cosmos、WeCross 等大多数跨链研究均基于区块链交易树进行交易打包验证[3,13−15]. 大多数区块链在生成新区块 B_n 时, 会以该区块包含的交易序列 $\{Tx\}$ 为叶子节点构建 Merkle Tree[50]、Merkle Patricia Tree[51] 等形式的交易树, 并将交易树树根存储在区块头 BH_n 中. 验证者基于交易树树根、跨链交易以及对应的验证路径即可验证跨链交易是否在交易树中. 这种方法仅需要验证者存储包含交易树树根的区块头 BH_n, 与存储完整区块

的方案相比开销较小, 但是无法适用于 Fabric 等没有交易树的区块链.

2) 区块一致性验证

验证者如何验证区块是否在源区块链主链上, 取决于源区块链的共识算法. 区块链的共识算法分为确定性共识以及概率性共识两大类[52], 前者只会形成一条区块链, 区块一旦形成不可更改; 后者需要在众多区块链中确定一条主链, 由于主链可能发生变更, 因此链上区块包含的交易也可能被回滚.

当源区块链的共识算法是 PBFT[53]等确定性共识算法时, 因为只有一条区块链存在, 验证者仅需验证区块是否符合源区块链出块规则. 例如, Cosmos 通过验证区块头是否拥有足够的可信验证者签名投票实现对采用 tendermint 共识的区块链的验证[29]. Cosmos 检验区块头的验证者的投票权重是否超过可信验证者集合总投票权重的 2/3, 如果通过验证则认定为有效区块. 同时, 如果该区块的验证者集合与目的区块链维护的可信验证者集合不一致, 则在认定该区块有效后, 将可信验证者集合更新为该区块的验证者集合, 从而完成可信验证者的更新.

当源区块链的共识算法是工作量证明类[2,54]、权益证明类[55−57]等概率性共识算法时, 验证者还需基于源区块链主链确定规则进一步验证区块是否在源区块链主链上. 例如, BTC Relay 实现了对比特币区块链的验证. 比特币区块链采用 PoW 共识算法, 并通过最长链规则确定主链. BTC Relay 从一个确定的区块高度开始不断同步后续区块头; 同时设置了确认期, 保证目标区块后足够多的有效区块, 从而保证所同步的比特币区块头链为最长链.

由于验证概率性共识区块链中的区块要同步该区块前的所有区块, 验证开销与区块高度呈线性正相关. 即使仅同步、验证源区块链区块头, 随着区块高度的不断增加以及跨链场景下区块链数目的增多, 验证者依旧需要消耗大量的存储、计算资源.

为了降低跨链场景下验证者的开销, ETH Relay 通过乐观接收区块头的方法实现区块一致性验证[5]. ETH Relay 中目的区块链乐观地接收转发节点在每个区块高度提供的第 1 个源区块链区块, 并对该区块进行公示. 公示期间任意链下客户端可以对该区块进行链下审核, 并在审核不通过时在目的区块链上发起挑战, 只有当链下客户端发起挑战时, 目的区块链才依据源区块链规则验证该区块是否有效. 这种优化方法在合适的奖惩机制的配合下可以很好地降低目的区块链上的计算开销, 但是无法降低目的区块链上的存储开销.

Garoffolo 等人基于 NIPoPoW 构建简洁的区块一致性证明, 实现无需信任的 PoW 侧链[32]. NIPoPoW 是一种超轻客户端, 其实现了链外节点对区块链一致性的简洁的验证[58]. PoPoW 是 NIPoPoW 的基础版本[59].

PoPoW 设计了一种基于跳表的结构体 Interlink, 添加在每个区块中. 该结构体结合每个区块在共识过程中求解 PoW 的结果, 指向前序中的一些其他区块, 形成如图 5 所示的多级索引. 验证者不需要完整的区块链, 而是仅基于所有未确认的 (如比特币中默认 6 个区块后确认) 区块以及各级索引中一定数目的区块, 即可在一定安全程度上完成区块一致性验证, 该方法可以将存储、验证开销从线性降低到对数级, 但是该方案只可用于难度不变的 PoW 区块链, 且需要验证者与全节点反复通信.

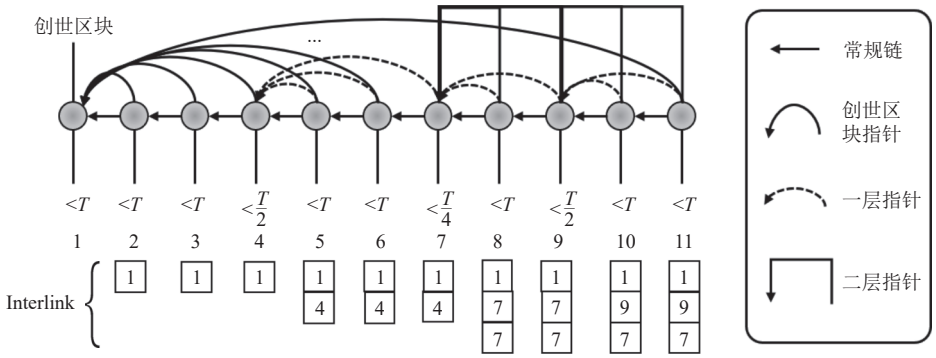

图 5 Interlink 结构示意图

NIPoPoW 是 PoPoW 的改进版, 其同样基于 Interlink 构造由各级索引中一定数目的区块组成的证明. 验证者对比多个证明者提交的证明, 找到多个证明的最低公共祖先 B, 并对公共祖先 B 之后的区块按照区块索引层级进

行打分. NIPoPoW 认为综合得分最高的证明有较高可能性来自于长度最长且真实有效的区块链. NIPoPoW 可以用于难度调整的区块链, 减少了轻节点与完整节点通信的次数, 实现了更加快速、方便的区块头有效性验证.

FlyClient[60]是另一种超轻客户端, 其为每个区块引入如图 6 所示的默克尔山脉 (Merkle mountain range, MMR) 的根, 并基于 MMR 扩展不影响前述叶子节点默克尔根的特效, 通过区块抽样将区块一致性验证的存储、验证开销从线性降低到对数级. 与 PoPoW 及 NIPoPoW 相同, 这种校验是基于一定安全程度进行的猜测, 而非确定性的校验.

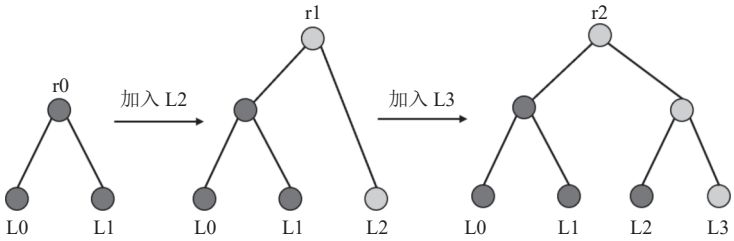

图 6 默克尔山脉示意图[60]

超轻客户端研究应用于跨链领域时, 需要源区块链已经包含有指定的数据结构, 或通过分叉的方式进行更新, 这制约了他们在跨链系统中的实施性. 但是, 随着越来越多的区块链项目对超轻客户端的支持, 如 zcash[61], 它在跨链研究中的使用前景也越发明朗.

(2) 状态数据验证

状态数据验证 (可信性验证) 是指验证者验证状态数据是否在源区块链中达成了一致. 由于区块链系统中节点就区块数据进行共识, 因此如果区块结构中包含状态数据, 验证者可以通过验证状态是否包含在源链主链中实现, 而如果区块数据中不包含状态数据, 验证者则需要通过其他方法实现.

1) 区块包含状态数据

当源区块链的区块数据包含状态数据 (如, 状态树) 时, 状态数据的验证可以拆分成:

a) 状态打包验证, 即验证状态 S 是否被打包在区块 B_n 中.

b) 区块一致性验证, 即验证区块 B_n 是否在源区块链主链上.

其中, 区块一致性验证在账本数据验证部分已有详细总结, 此处不再赘述. 下面将综述总结状态打包验证相关研究.

Westerkamp 等人提出的智能合约移植[62]是指将源区块链上的智能合约复制到目的区块链上, 智能合约移植完成后, 源区块链和目的区块链均能通过交易更新合约状态. 在移植过程中, 目的链需要验证目的区块链上重构的智能合约状态是否与源区块链智能合约状态一致. 该方案通过链下重放交易构造待移植合约的状态, 并通过直接比较源、目的区块链中该合约的存储根是否一致, 实现对智能合约状态的验证. 但是该方案仅当源区块链与目的区块链状态树构建的方式一样时才可使用.

与交易数据一样, 部分区块链状态数据也以状态树的形式存储在区块头中[49], 因此验证者同样可以基于轻客户端思想, 通过状态树树根以及状态 S 对应的验证路径实现状态打包验证.

Fynn 等人提出的智能合约转移[63]是指将源区块链上的智能合约移动到目的区块链上, 智能合约转移完成后, 源区块链只能读取而不能更新被锁定的合约状态, 目的区块链则可以通过交易更新合约状态. 在移动过程中, 目的区块链同样需要验证智能合约状态. 该方案基于轻客户端思想, 通过源区块链区块头中存储的状态树根以及合约状态所在路径实现状态打包验证.

由于状态数据是全局数据而非特定区块的数据, 数据量大且类型负责复杂, 与交易验证相比, 基于轻客户端思想进行状态验证复杂度更高、开销更大、耗时更长.

SmartSync 在智能合约移植的基础上, 通过定期更新智能合约状态, 实现目的区块链对源区块链指定智能合约状态数据的同步[64]. 为了减少状态同步、验证的开销, 该方案在同步过程中并不同步、验证合约的所有状态, 而是仅基于轻客户端思想同步并验证更新的状态, 同时通过将证明路径中的新状态替换成目的区块链存储的当前状态, 并将生成的状态根与目的区块链当前的合约状态根对比, 确保状态更新同步的完整性.

2) 区块不包含状态数据

当源区块链的区块数据不包含状态数据时, 验证状态数据性最直接的方法是验证者执行区块链主链交易, 在本地生成并维护源区块链的可信状态, 但是这种方法会有持续的计算开销与不断增长的存储开销.

部分跨链解决方案将状态数据验证转换成账本数据验证: 通过将目标状态写入区块链事件 event, 实现本地状态信息写入区块链账本, 目的区块链通过交易信息验证方法实现对交易与事件的验证, 并从事件中可以获取目标状态信息. 但是将区块链状态验证转换成交易验证的方法需要在区块链上通过额外的交易将状态信息写入账本中, 增长了跨链流程, 增大了跨链时延与开销.

此外, 无状态客户端[65]研究对此场景下的状态数据验证具有借鉴意义. 目前该方向的研究可以分为基于哈希树累加器的无状态客户端[66−68]、基于 RSA 累加器的无状态客户端[68,69]以及基于双线性映射累加器的无状态客户端[70,71]. 其基本原理是不存储区块链的完整状态, 而是基于上述累加器聚合并存储区块链状态, 在此基础上, 借助对应的状态承诺可以证明区块链状态有效性. 然而现有无状态客户端研究聚焦于降低存储开销, 其计算复杂度较高, 无法直接应用于跨链场景.

3.4 跨链数据验证方法评价

(1) 评价指标

根据本节综述的区块链账本数据跨链验证方法与区块链状态数据跨链验证方法, 我们整理出表 3、表 4, 展示验证方法的分类、方法所解决的问题, 并从开销、安全性、实现难度、适用场景、验证主体以及各自独特的优劣进行评价.

表 3 区块链账本数据跨链验证方法评价

文献	验证主体	安全性	适用区块链	实现难度	开销		其他
					存储开销	计算开销	
[3,15]	链上	哈希	支持轻客户端	中	线性	线性	—
[32,58]	链上/链下	概率	PoW共识	高	对数级	对数级	单次通信
[59]	链上/链下	概率	PoW共识 (难度固定)	高	对数级	对数级	多次通信
[60]	链上/链下	概率	概率性共识	高	对数级	对数级	单次通信
[5]	链上	哈希	支持轻客户端	中	线性	非线性	—

表 4 区块链状态数据跨链验证方法评价

文献	验证主体	安全性	适用区块链	实现难度	开销		其他
					存储开销	计算开销	
[62]	链下	哈希	区块有状态根	低	常数级	低	状态根构造相同链间
[63]	链上	哈希	区块有状态根	中	线性	较高	—
[64]	链上	哈希	区块有状态根	中	线性	中	—
[66−68]	链下	哈希	UTXO	高	对数级	较高	—
[68,69]	链下	RSA	账户/UTXO	高	常数级	高	10亿账户; 非成员验证
[70]	链下	ECC	账户	高	常数级	高	可聚合承诺

1) 验证主体: 即为前文所述的验证主体, 包括链上链下两种.

2) 安全性: 即评价该跨链数据验证方案的安全性是如何保障的, 例如有的验证方案因为其算法, 只能概率性地保证安全性; 有的方案的安全性通过密码学算法予以保障.

3) 适用区块链: 本文所综述的方案不局限于各跨链方案, 还包括了一些非跨链场景技术的借鉴, 其中很多方案对于区块链本身因为作出了较强的假设, 因此, 此处主要评价这些方案所适用的区块链场景.

4) 实现难度: 方案在跨链场景下的实现难度, 用以评价方案的可实施性.

5) 开销: 包括存储开销与计算开销. 对于解决不同问题的方案, 其开销的评价角度、比较对象不一而同, 难以跨问题评价, 我们将在后面的统一评价中进行展开.

6) 其他: 各方案独特的优缺点.

(2) 方案对比

1) 区块链账本数据验证

● 安全性方面: BTC Relay[3], ETH Relay[5], Cosmos[15]等大部分跨链方案通过维护源区块链轻客户端, 实现对源区块链账本数据的验证, 其安全性来源于哈希计算的防碰撞性; PoPoW[59]、NIPoPoW[58]、FlyClient[60]基于“抽取”的思想, 不再同步所有区块头, 而仅通过同步对数级数目的区块头确定主链, 但是其对于主链的确定是概率性的, 抽取的区块头越多, 确定的概率越大, 安全性越高.

● 适用区块链方面: BTC Relay, ETH Relay, Cosmos 等通过构建源区块链轻客户端的方案需要源区块链支持轻客户端; PoPoW、NIPoPoW、FlyClient 是针对概率性共识区块链验证方案的优化, 其中 PoPoW、NIPoPoW 方案仅适用于 PoW 共识区块链.

● 实现难度方面: BTC Relay, ETH Relay, Cosmos 等方案的具体难度取决于源区块链轻客户端构建难度; PoPoW、NIPoPoW、FlyClient 这 3 种方案均修改了区块链结构, 尽管已有一些区块链进行了升级适配[61], 但是无法直接用于大多数区块链的跨链验证, 需要源区块链分叉进行支持, 实现难度较大.

● 开销方面: BTC Relay 等大部分跨链方案通过逐个验证并存储区块头的方式实现源区块链轻客户端的构建与维护, 其存储开销与计算开销均线性增长, 具体开销与实现复杂度取决于源区块链; ETH Relay 采用乐观接收区块头的方式, 可以降低链上计算开销, 但是存储开销依然线性增长; PoPoW、NIPoPoW、FlyClient 基于“抽取”的思想, 将存储开销与计算开销降低到对数级别, 对跨链验证不频繁场景下的跨链验证优化研究具有一定参考价值.

2) 区块链状态数据验证

● 安全性方面: 基于 RSA 累加器的无状态客户端[68,69]与 Pointproofs[70]的安全性分别来源于 RSA 与 ECC 椭圆曲线; 而智能合约移植[62]、智能合约转移[63]、智能合约同步[64]以及基于哈希树累加器的无状态客户端[66−68]均通过哈希计算聚合、验证状态, 相对 RSA 与 ECC 较弱.

● 实现难度方面: 智能合约移植、智能合约转移、智能合约同步实现难度较小, 而 3 类基于累加器的无状态客户端方案因为其数学计算的复杂程度, 缺乏在链上的可实现性, 应用在跨链场景时更适用于链下代理作为验证主体的方案.

● 开销方面: 智能合约移植不进行状态打包验证, 通过比较源、目的区块链合约存储根验证状态是否一致, 开销小但是需要引入合适的奖惩机制保证方案的可用性, 且仅适用于源、目的区块链状态树设计一致的场景; 智能合约转移以及智能合约同步基于链上维护的源区块链轻客户端, 在链上根据状态树实现状态打包验证, 具体开销与复杂度取决于源区块链状态树设计与状态树大小, 但由于状态树包含源区块链全局状态数据, 因此一般开销较大, 耗时较长, 其中智能合约同步通过仅验证更新状态降低计算开销. 3 类基于累加器的无状态客户端无需存储区块链的状态, 而是通过获取的证据直接验证状态, 因此, 表格中此处的存储开销, 主要指每次通信时, 客户端需要获取的证据大小. 基于哈希树累加器的无状态客户端每次通信的证据大小, 相比所有区块数目是对数级的, 其计算开销主要来源于哈希计算, 是 3 类无状态客户端中最小的; 基于 RSA 累加器的无状态客户端可以将每次提交的证据大小优化为常数大小, 显著优于哈希累加器方案的表现, 并且拓宽了方案的适用范围, 其计算开销主要来源于 RSA 的验证计算, 计算复杂度高; Pointproofs 使用 ECC 椭圆曲线来充当双线性映射, 相比 RSA 累加器方案, 也提供了常数级的证据大小, 并且在相同证据大小时, 能够提供更高的安全性, 当然, 这也带来了更高的验证计算复杂度.

4 跨链操作原子性保障关键技术

跨链操作原子性是指跨链操作拆分成的在多个区块链上子操作执行的原子性, 即多个区块链上的子操作需要一致执行或者一致不执行.

原子性问题在数据库领域已经历了数十年的研究, 这一问题的基本解决思路是两阶段提交.

(1) 准备阶段

各节点执行本地操作后进入准备状态, 等待其他节点进入准备状态.

(2) 提交阶段

1) 提交操作: 如果所有节点都可以提交, 则所有节点提交本地操作.

2) 回退操作: 如果存在节点不能提交, 则所有节点撤销本地操作.

跨链操作原子性问题的基本解决思路同样可以归纳为两阶段提交.

(1) 准备阶段

各区块链通过执行跨链子操作证明其具备执行跨链子操作的能力, 并且将子操作相关的区块链状态锁定进入准备状态, 等待其他区块链的准备状态.

(2) 提交阶段

1) 提交操作: 如果所有跨链操作相关区块链都进入准备状态, 即相关区块链都具备执行跨链子操作的能力, 则所有相关区块链均提交子操作状态.

2) 回滚操作: 如果存在跨链操作相关区块链未进入准备状态, 即存在相关区块链不具备执行跨链子操作的能力, 则所有相关区块链均回滚子操作状态.

两阶段提交的关键在于根据准备阶段的状态协调各区块链完成一致的提交或者回滚操作. 依据协调对象的不同, 可以将跨链原子性保障关键技术分为基于用户自协调的原子性保障机制与第三方协调的原子性保障机制. 下面将分别综述各类方案的研究现状, 并提出评价指标分类评估现有方案.

4.1 基于用户自协调的原子性保障机制

基于用户自协调的原子性保障机制是指由参与跨链操作的用户作为协调者依据跨链操作相关区块链的准备状态, 对提交阶段的操作进行决策的方案.

哈希时间锁协议 (hashed timelock contract, HTLC) 是基于用户自协调保障原子性的经典方案[8], 其使用哈希锁[72]与时间锁[73]保证多个操作的原子执行, 一般用于解决跨链资产原子交换问题[74].

具体地, HTLC 需要选举出一个用户担任发起者与协调者, 并由该用户在进行跨链操作前设置哈希锁 $hash(x)$, 哈希锁原像 x 仅有该用户持有. 准备阶段, 用户使用相同的哈希锁 $hash(x)$ 与差异化的时间锁 T_1 与 T_2 锁定资产, 交易对手需要使用哈希锁原像 x 才能进行提交, 用户需要等待时间锁计时完毕才能进行回滚. 提交阶段, 若时间锁到期前协调者确认所有跨链操作相关区块链都进入准备状态, 则释放哈希锁原像 x 完成提交操作, 其他参与用户获取哈希锁原像 x, 随后完成提交操作; 若时间锁到期, 用户执行回滚操作取回锁定资产.

HTLC 基于相同的哈希锁保证各个区块链上的子操作可以一致提交, 基于差异化的时间锁, 保证跨链操作可终止, 且参与者有足够的时间提交操作. 但是 HTLC 的提交操作与回滚操作并不是完全互斥: 攻击者在执行完提交操作后, 可以通过审查攻击[75−77]、洪泛攻击[78]等方法阻止其他用户执行提交操作, 并在超时后执行回滚操作, 这种情况下 HTLC 无法保证跨链操作的原子性. 目前在实际应用中一般通过将时间锁与其差值设置得足够长减少审查攻击、洪泛攻击带来的影响. 但是 HTLC 需要参与者串行地在各个区块链上构造正确的哈希时间锁, 足够长的时间锁设定会导致跨链操作完成的最大时延过长, 而当跨链操作涉及多个参与者时, 这种现象将更突出[79].

HTLC 的实现需要参与区块链均支持脚本或者智能合约, 但是并非所有区块链都支持智能合约 (或者脚本). Zie 等人对哈希时间锁协议进行了扩展, 通过智能合约技术 (或者脚本) 与多签技术实现了支持智能合约的区块链与不支持智能合约的区块链间的原子交换[80].

该方案由在不支持智能合约 (或者脚本) 的区块链上锁定资产的用户担任发起者与协调者. 下面基于跨链原子写操作例子进行描述, 其中区块链 $Chain_1$ 不支持智能合约, Alice 为协调者.

在准备阶段, Alice 将资产锁定在多签账户中, 需要 Alice 与 Bob 的签名才能进行解锁; Bob 提前为区块链 $Chain_1$ 上的提交操作 tx_1 (将 $Chain_1$ 锁定资产转给 Bob) 与回滚操作 tx_2 (将 $Chain_1$ 锁定资产退给 Alice) 签名, 并将资产锁定在智能合约中. 在提交阶段, 若 Alice 确认 Bob 进入准备状态, 则为区块链 $Chain_1$ 上的提交操作签名, 并使用该签名触发区块链 $Chain_2$ 上的预提交操作 (将锁定资产标记为解锁), Bob 学习到该签名后使用该签名触发 $Chain_1$ 上的提交操作, 一定时间后 Alice 触发 $Chain_2$ 上的提交操作 (将 $Chain_2$ 锁定资产转给 Alice); 否则, Alice 为区块链 $Chain_1$ 上的回滚操作签名, 并使用该签名触发区块链 $Chain_2$ 上的回滚操作 (将锁定资产退给 Bob).

该方案拓展了基于用户自协调的原子性保障机制的适用范围, 但是由于协调者在提交阶段可以通过审查攻击在不同区块链上触发相反的操作, 从而破坏跨链操作的原子性, 即跨链操作的原子性依赖于同为参与者的协调者遵守协议.

基于用户自协调的方案依赖于参与者之间的同步假设, 需要各个参与者实时在线并关注区块链的更新情况, 对跨链操作参与者提出了较高的要求.

4.2 基于第三方协调的原子性保障机制

基于第三方协调的原子性保障机制是指引入外部第三方作为协调者依据跨链操作相关区块链的准备状态, 对提交阶段的操作进行决策的方案. 根据第三方的性质, 基于第三方协调的原子性保障机制又可以分为公证人作为协调者以及区块链作为协调者两类.

(1) 基于公证人协调的原子性保障机制

以跨链资产原子交换为典型的原子性问题出现之初, 主要通过公证人解决原子性问题[7,79,81].

AC^{3}TW[79]借助跨链操作参与者共同选择的公证人 Trent 保证跨链操作的原子性. 准备阶段, 参与者将资产锁定智能合约中, 该智能合约根据公证人 Trent 的指令 RD 与 RF 分别执行提交操作与回滚操作. 在任意时刻, 跨链操作参与者都可以向 Trent 发起提交请求或者回滚请求从而进入提交阶段. 提交阶段, Trent 指令一旦发布就不会再更改, 保证了提交操作与回滚操作互斥, 从而保证跨链操作的原子性.

该方案完全依赖于公证人 Trent, 一旦 Trent 不遵守协议, 则无法保证跨链操作的原子性. 然而, 由于各个区块链在组成节点、安全等级等方面存在较大差异, 并不是所有跨链操作都能找到一个可信的公证人.

Lipton 等人提出的方案[81]借助交互区块链各自选择的公证人保证跨链操作的原子性, 为了保证多公证人间无争议的协调, 其基于经典的两阶段提交协议[82,83]或者其他变体 (例如三阶段提交协议) 实现, 具体过程如图 7 所示. 交互区块链的网关节点首先建立连接并交换跨链信息, 接着源区块链网关节点 G1 向目的区块链网关节点 G2 提供源链待转移资产已锁定的相应证明, 同时 G2 返回该证据的签名收据, 随后 G1 与 G2 通过两阶段提交保证交互区块链上同时提交.

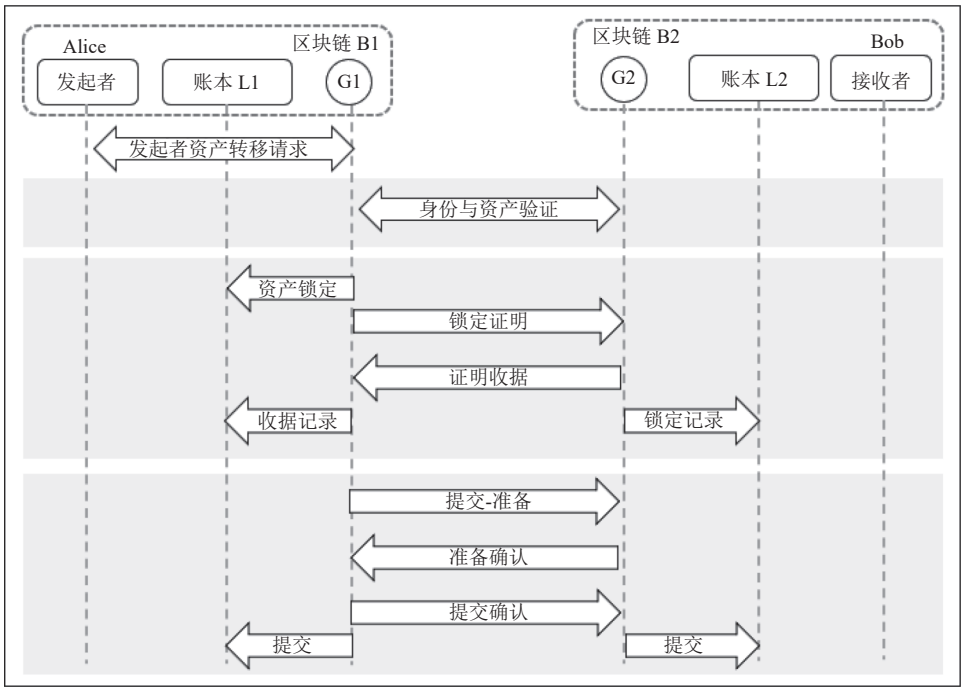

图 7 Lipton 等人方案流程图[81]

该方案拓展了基于公证人协调的原子性保障机制的适用范围, 但是这种“多公证人”的模式并没有降低用户对公证人的信赖, 反而需要更强的安全假设: 由于交互区块链预锁操作顺序执行且不提供回滚操作, 如果存在后手公证人不按照协议执行或者遭受攻击, 则跨链操作无法终止.

上述两类方案存在单点故障的隐患, 为了降低原子性保障方案对单一可信公证人的依赖, Interledger 原子模式[7]将单一公证人扩展成公证人集合, 同时通过拜占庭容错算法实现公证人集合一致的协调, 从而降低对单一公证人的信任程度.

基于公证人协调的原子性保障机制与基于用户自协调的原子性保障机制相比, 不需要用户保持同步, 降低用户复杂度的同时可以实现准备阶段与提交阶段中各用户并行操作, 在多参与者的场景中可以有效降低整体时延, 但是引入了额外的可信公证人, 增强了安全假设.

(2) 基于区块链协调的原子性保障机制

虽然 Interledger 将单公证人协调拓展成了多公证人协调, 提高了协调者的去中心化程度, 提升了原子性保障机制的容错能力, 但是公证人集合大小不能无限扩展, 去中心化程度有限. 为了进一步提升原子性保障机制的容错能力, 不少研究将公证人扩展成公证链.

AC^{3}WN[79]借助一条非许可链 (公证链) 进行协调保证跨链操作的原子性, 其中, 公证链需要部署与公证人 Trent 有相同协调功能的智能合约. 该方案原子性保障的原理及流程与前文所述的中心化可信公证人模式一致, 只是将公证人 Trent 替换成公证链, 相应的, 参与者与 Trent 间的通信也替换成了交互区块链与公正链间通信.

除了使用第三方区块链外, 参与跨链交互的区块链也可以作为协调者保障跨链操作的原子性. Zhao 等人的方案[84]基于链间传输同步假设, 借助任意一个交互区块链作为 leader, 通过分布式系统两阶段提交, 在崩溃容忍场景下保证跨链操作的原子性, 同时通过被动心跳方法检测节点故障, 从而避免阻塞.

与基于公证人协调的原子性保障机制相比, 基于区块链协调的原子性保障机制保持用户友好、支持并行等优势的同时, 进一步提升了协调者的去中心化程度, 同时这类机制基于智能合约实现协调协议, 协议内容公开透明, 支持用户进行验证与确认, 将对公证人身份的信任转换成对区块链技术的信任. 相对应的, 协调功能需要在区块链上实现部署, 复杂度提升, 而由于引入链间通信, 跨链操作时延增大.

4.3 原子性保障方案评价

(1) 评价指标

综合前文介绍的各原子性保障方案, 我们整理得到表 5, 从基础要求与扩展要求两个层面对其进行分析与评价.

表 5 原子性保障方案评价

文献	安全性	可终止性	去中心化程度	用户实时在线要求	智能合约支持要求	实现难度	多参与方原子性保障扩展	性能		
								回滚模式	执行模式	协调时间开销
[79]*	满足	满足	公证人	不需要	双方	低	适用	主动	并行	低
[81]	满足	不满足	公证人	不需要	双方	低	适用	无	并行	低
[79]**	满足	满足	公证区块链	不需要	双方	中	适用	主动	并行	高
[84]	满足	满足	公证区块链	不需要	双方	中	适用	无	并行	高
[8]	不满足	满足	单用户	需要	双方	低	改进后适用	时间锁	串行	低
[80]	不满足	满足	单用户	需要	单方	低	不确定	主动+时间锁	串行	低

注: *是AC^{3}TW模式, **是AC^{3}WN模式

原子性保障方案的基础要求, 包括安全性与可终止性, 一般而言, 只有当方案同时满足了安全性与可终止性, 才实现了跨链操作的原子性的保障.

1) 安全性: 即使在客户端崩溃故障、网络分区、消息延迟、参与者偏离协议的环境下, 协议执行完成后, 所有参与方的终止状态仍满足“all or nothing”的需求, 即要么所有参与方均回滚至初始状态, 要么都迁移至执行子操作后的状态.

2) 可终止性: 所有跨链操作相关区块链都会进入提交阶段而不会永久地停留在准备阶段.

原子性保障方案在实现与应用方面的扩展性要求, 具体包括如下几个方面.

1) 性能: 在实际系统中, 因为跨链操作原子性保障方案不是独立实现的, 需要综合诸多其他关键技术, 例如跨链信息传输技术, 因此, 评价原子性保障机制本身的性能只能从有限的几个角度入手, 此处我们从回滚模式、执行模式和协调时间开销 3 个角度评估其性能.

a) 回滚模式: 回滚模式分为主动回滚和超时回滚, 主动回滚是指在准备阶段的任意时刻参与者可以发起回滚操作, 跨链操作执行进入提交阶段; 超时回滚是指, 参与者需要等待准备阶段设置的定时器超时后才能发起回滚操作.

b) 执行模式: 即各个阶段各区块链子操作间的执行模式, 包括串行执行与并行执行.

c) 协调时间开销: 即原子性保障方案中, 协调各部分工作所需要的时间开销, 包括但不限于协调者拆分子操作的时间开销、协调者对于参与方提交的签名等信息的确认时间.

2) 去中心化程度: 即协调者的去中心化程度, 用于描述方案抵御协调者单点故障与联合作恶的能力.

3) 智能合约支持: 即该方案的实现是否需要参与方双方所在区块链都支持智能合约, 用于评估该方案的适用范围.

4) 实现难度: 该指标意为该方案在跨链场景下的实现难度, 用以评价方案的可实施性.

5) 用户实时在线: 即用户是否需要实时在线并在区块链间同步信息、处理信息, 用于评估该方案对用户的要求高低.

6) 多参与方原子性保障: 即是否能保证多参与方跨链操作的原子性多, 分为不适用/适用/推测适用/不确定 3 种, 不确定即为文章没有明确表示是否适用, 但是本论文中也将给出自己的理解.

(2) 方案对比

● 安全性方面: HTLC[8]及其衍生出的 Zie 方案[80], 由于提交操作与回滚操作并不是完全互斥, 在洪泛攻击、审查攻击等场景下, 无法保证安全性, 进而无法保证方案原子性, 其他方案都可以有效保证安全性.

● 可终止性方面: Lipton 等人[81]提出的方案由于不提供回滚机制, 在参与者偏离协议时无法保证方案的可终止性.

下面是原子性保障机制扩展性要求的分析与评价.

● 去中心化程度方面: 基于公证人或用户协调的方案与基于区块链协调的方案相比, 去中心化程度更低, 更容易受协调者故障的影响, 协调者作恶破坏原子性的成本更低.

● 用户实时在线方面: HTLC 及 Zie 等人[80]提出的方案因为其自协调特性, 需要用户保证实时在线进行信息同步与协调处理, 对用户要求更高.

● 智能合约支持方面: 除 Zie 方案外, 其他方案都需要参与双方所在区块链都支持智能合约 (或脚本), 一定程度上限制了方案的实现.

● 实现难度方面, 基于单用户与公证人的方案依赖用户协调, 实现难度低; 基于区块链协调的方案需要在链上实现协调方法, 难度较前者更高. 多参与方原子性保障方面: 除 Zie 等人提出的方案外的其他各机制, 都可以有效扩展到多方资产交换的场景中, 其中 HTLC 作为最早的原子性保障方案, 最初仅实现了双方资产交换, 但是已有工作将其扩展至多方资产交换的场景. 而当一笔复杂的多方交换事务可以拆分成多笔智能合约链和非智能合约链的子操作时, 我们认为 Zie 等人提出的方案无法满足所有多方资产交换的需求, 只有在一个参与者在不支持智能合约的区块链上时才能实现多方资产交换.

● 性能方面: 1) 主动回滚相比超时回滚, 在参与者偏离协议等异常情况下可以更早地完成跨链操作执行, 性能表现更好. 2) 并行执行方案相比串行执行方案, 其原子性事务的执行耗时更短, 在越复杂、流程越长的跨链操作中性能表现更好. 3) 基于公证人协调或用户协调的方案与基于区块链协调的方案相比, 由于在链下完成协调功能, 其协调时间开销更低, 性能表现更好.

5 跨链整体解决方案

早期跨链解决方案主要集中于两个区块链交互的场景, 随着对跨链需求的深入认识, 研究焦点逐渐拓展到多区块链交互的场景. 针对这些场景, 已有多种跨链解决方案被提出, 根据是否需要借助其他区块链可以分为两大类: 一

类是基于直连的跨链解决方案, 其基本思想是有跨链需求的区块链直接实现区块链互操作; 一类是基于中继链的跨链方案, 其基本思想是有跨链需求的区块链借助其他区块链作为中继, 通过多跳跨链操作实现指定区块链互操作.

5.1 基于直连的跨链解决方案

直连跨链解决方案的基本思想是有跨链需求的区块链直接实现区块链互操作, 即在有跨链需求的区块链间解决上述 3 个跨链关键问题.

(1) BTC Relay

BTC Relay 被认为是最早的跨链方案[3], 旨在完善以太坊基础设施、帮助以太坊实现更大的创新, 其实现了比特币区块链与以太坊区块链的单向跨链操作——支持在以太坊上验证比特币交易. BTC Relay 在以太坊上部署可以接收并处理比特币区块头与交易的智能合约, 并通过 Relayer 将比特币区块头跨链传输至以太坊实现基于最长链规则的比特币轻客户端, 为比特币交易的链上验证提供基础. 用户将比特币交易传输至以太坊, 并附加比特币交易打包证明 (Merkle path), 实现以太坊对比特币交易的验证. BTC Relay 不解决跨链操作原子性保障问题.

(2) RSK

RSK 是锚定比特币区块链的一个开源智能合约平台, 其目标是将智能合约以可操作的形式带入比特币系统, 实现即时支付以及高扩展性, 从而为比特币生态系统增加价值和功能. 与 BTC Relay 实现的单向跨链操作不同的是, RSK 实现了与比特币的双向跨链操作, 支持在 RSK 上根据比特币资产锁定交易生成资产以及在比特币上根据 RSK 资产锁定交易解锁资产[11]. 在此过程中需要在 RSK 上验证比特币交易、在比特币上验证 RSK 交易: 在比特币至 RSK 方向, 公证人联盟或者用户将比特币交易传输至 RSK 中, 并基于 RSK 链上维护的比特币轻客户端进行验证; 在 RSK 至比特币方向, RSK 交易并未传输至比特币链上, 而是由公证人联盟代替比特币接收、验证该信息, 并在比特币上签署对应解锁交易. RSK 的公证人联盟由 15 家公司组成, 在比特币参与 RSK 联合挖矿矿工比例足够时公证人联盟将切换为联合矿工. RSK 并没有考虑跨链操作原子性保障问题, 然而这对 RSK 是必要的, 目前 RSK 有可能面临转账不可终止的情况.

(3) HTLC

HTLC 是常用的跨链操作原子性保障技术[8], 但是作为面向跨链资产原子交换问题的完整解决方案, 其同样完成了跨链信息在链间的传输以及目的区块链对跨链信息的可信性验证. HTLC 实现了在目的区块链上验证特定的源区块链交易, 不过与其他方案不同, 特定的源区块链交易并没有直接被传输到目的区块链上, 仅有其包含的哈希锁密钥被用户传输到目的区块链上. 目的区块链、根据预先设置的哈希锁可以验证哈希锁密钥的正确性, 从而间接验证哈希锁密钥对应的特定源区块链交易有效. HTLC 基于参与者间的同步假设与各区块链上相同的哈希锁保证安全性, 基于差异化的时间锁, 保证跨链操作可终止, 然而由于 HTLC 的提交操作与回滚操作并不是完全互斥, 其在审查攻击、洪泛攻击等场景下无法保证跨链操作的原子性, 目前在实际应用中一般通过将时间锁设置得足够长减少审查攻击、洪泛攻击带来的影响.

(4) WeCross

WeCross 面向区块链互联互通问题, 旨在构建一套未来区块链互联基础设施[13], 虽然其支持多链跨链, 但是由于跨链交互不需要借助其他区块链, 本质上依旧是基于直连跨链架构的方案. WeCross 的核心组件是跨链路由, 每条参与跨链交互的区块链都有一个由部署该区块链的机构搭建的跨链路由, 区块链强信任其跨链路由. WeCross 基于跨链路由解决上述 3 个跨链关键问题: 通过跨链路由间的信息传输实现跨链信息传输, 通过在跨链路由内构建其他区块链的轻客户端实现对跨链信息的可信性验证, 通过跨链路由担任协调者实现基于可信公证人的原子性保障. 此外, WeCross 还集成了哈希时间锁技术 (HTLC) 解决跨链操作原子性保障问题.

5.2 基于中继链的跨链方案

一类是基于中继链的跨链方案, 其基本思想是有跨链需求的区块链借助其他区块链作为中继, 通过两跳跨链操作实现区块链互操作. 单个中继链的处理能力有限, 当存在大规模跨链需求时, 中继链可以进一步扩展成中继链网络, 有跨链需求的区块链借助中继链网络, 通过多跳跨链操作实现区块链互操作.

(1) Cosmos

Cosmos 针对区块链存在的扩展性、可用性以及独立性问题, 提出了构建区块链互联网的设想, 为此设计了一种区块链网络架构, 并提出了该架构下的跨链交互方案[15].

Cosmos 的区块链网络架构如图 8 所示, 独立区块链被称作分区 (zone), 连接分区的特殊分区被称作枢纽 (Hub), 分区借助枢纽实现跨链交互, 在大量分区存在交互需求时, 这种架构可以减少区块链之间的适配复杂度.

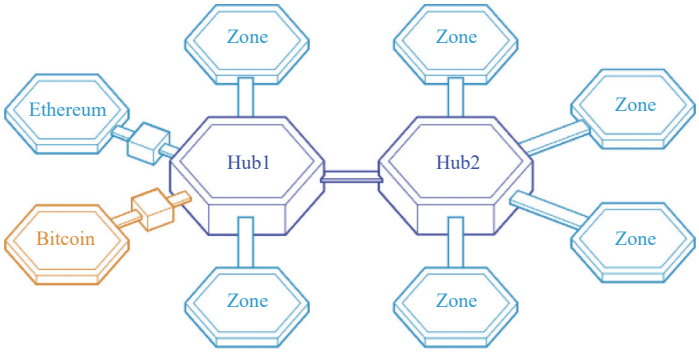

图 8 Cosmos 架构示意图

Cosmos 提出的 IBC 协议实现了相邻两个区块链间的信息传输与信任传递. 其中, 信息传输通过转发节点监听转发跨链信息实现, 信任传递通过在区块链上构建对端区块链轻客户端 (Client 模块) 实现. 尽管目前 IBC 协议仅适用于相邻两个区块链 (单跳), 但是未来将通过指定跨链信息传输的路径, 实现基于枢纽的多跳处理, 以支持区块链互联网的构建. Cosmos 的核心协议并不解决跨链操作的原子性保障问题, 而是将该问题交由具体应用解决.

在跨链交互方面, zone-zone 交互与 zone-Hub 交互并没有区别, Hub (中继链) 仅解决了适配复杂度的问题, 即当多个 zone 存在相互跨链需求时不需要两两适配, 仅需要每个 zone 与 Hub 适配.

(2) Polkadot

Polkadot 针对区块链在可扩展性、可伸缩性、安全性等方面普遍存在的问题, 设计了一种打破原有区块链共识架构中一致性与有效性紧密结合的网络架构, 旨在通过这种方法设计出一种完全可伸缩、可扩展的异构多链系统, 在保证最小化的安全性和传输功能的前提下将共识架构中的一致性和有效性分开[16].

Polkadot 的网络结构如图 9 所示, 由一条/多条中继链 (relay chain) 以及多条平行链 (parachain) 构成. 其中, 平行链负责具体业务的执行; 中继链负责与其直接连接的所有平行链的最终性共识. 为了在保证安全性的前提下实现中继链上复杂的操作, Polkadot 设计了 4 种角色共同维护网络: 收集者、渔夫、提名者以及验证者.

Polkadot 基于 XCMP 协议实现跨链信息传输, 并基于中继链对平行链区块最终有效性的验证与共识, 在共识层面完成对打包进目的平行链区块的跨链信息的可信性确认, 实现跨链信任传递. Polkadot 的核心协议并不解决原子性保障问题.

Polkadot 的中继链为跨链交互提供了信任传递功能, 平行链基于中继链对平行链区块最终有效性的验证与共识, 实现对对端链状态的确认.

(3) Interledger

Interledger 是 Ripple 面向交互区块链不存在可用中间人的跨链资产转移问题, 提出的不受区块链连接拓扑限制的安全支付协议[7]. 其核心思想如图 10 所示, 利用多跳同时在两条区块链上有账户的连接者连接跨链交易的发送者和接受者, 通过在源链、连接者所在区块链、以及目的链上的一组包括跨链交易发送者与连接者、连接者与连接者以及连接者与跨链交易接受者的交易, 实现借助一条或者多条区块链完成跨链支付. 虽然 Interledger 的目标是实现两个区块链间的交互, 但是实施过程中有多条区块链进行了跨链交互, 本质上属于基于中继链的跨链方案, 连接者所在区块链均为“中继链”.

聚焦到任意相邻区块链, 同时在两个区块链上拥有账户的连接者根据源区块链跨链请求, 在目的区块链上发

起对应交易, 完成单跳跨链操作. 在此过程中, 跨链信息并未传输到目的区块链并进行处理, 而是在链下由连接者代理接收并处理, 从而在应用层面达到跨链操作效果. 具体地, 连接者通过监听源区块链获取跨链请求, 通过接入源区块链实现对跨链请求的验证与确认. 由于连接者在目的区块链上的操作仅涉及其自身状态的更新, 因此连接者代理接收、处理跨链请求不需要目的区块链额外的授权与信任.

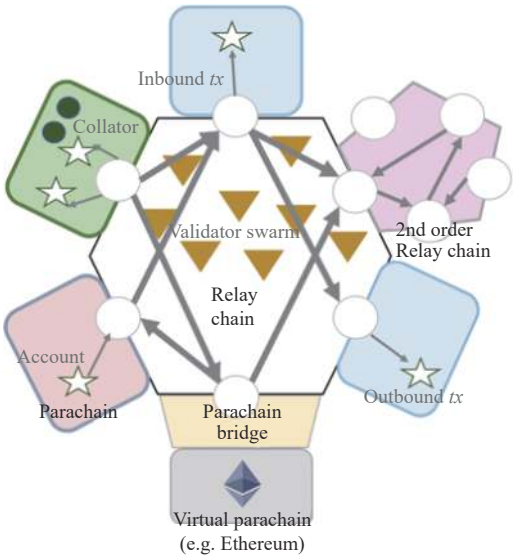

图 9 Polkado 络架构示意图[16]

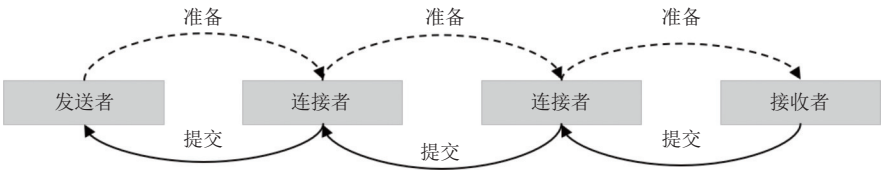

图 10 Interledger 多跳交易示意图

Interledger 包含通用模式和原子模式, 其中通用模式基于用户自协调保证跨链操作的原子性, 而原子模式基于一组公证人的协调保证跨链操作的原子性.

Interledger 的“中继链”与交互区块链在跨链交互为对等关系, 仅作为交互区块链打通跨链交互的通路.

(4) HyperService

HyperService 面向跨链合约调用问题, 提出了为异构区块链提供互操作性与可编程性的平台, 其具有 3 个核心组件 (如图 11 所示)[85].

1) 全局状态区块链 (NSB): 独立于应用区块链的区块链, 保存其他区块链的轻量状态信息以及交易执行信息, 负责监管 Dapp 的正确执行.

2) 保险智能合约 (ISC 合约): 部署在 NSB 上的智能合约, 基于 NSB 存储的信息, 能够对 Dapp 执行进行正确的决策, 保证跨链合约的可结算性与金融原子性.

3) 可验证执行系统 (VES): 可验证执行系统处理用户提交的 HSL, 将其编译成可执行的 DAG, 并根据生成的 DAG 在 NSB 上部署 ISC, 最后将生成的 DAG 与部署的 ISC 发送给用户验证与确认.

HyperService 由 VES 驱动跨链交易的执行, 并由 NSB 与 ISC 保证跨链交易的正确执行. 由于 VES 根据 DAG 在链下触发各个应用链上的操作, 属于链下触发模式 (具体定义见第 1.1 节), 区块链间没有直接的跨链交互, 因此不讨论应用链间对应 3 个关键问题的解决方案.

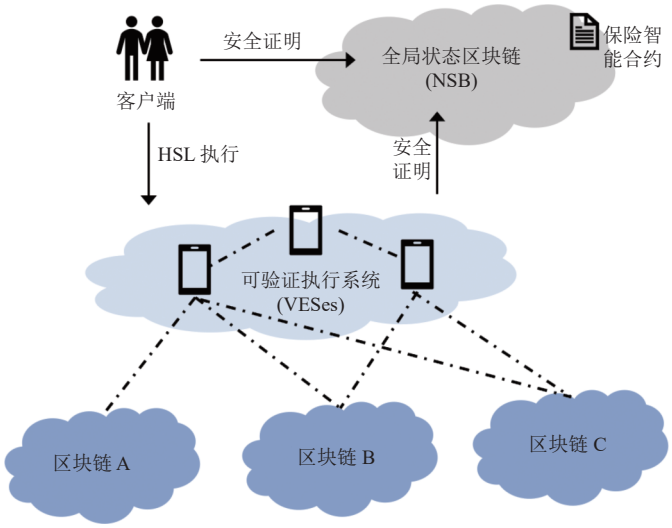

图 11 HyperService 架构图[85]

6 挑战与总结

6.1 跨链研究挑战

跨链技术已经取得了显著的进步, 但是它仍然是一个充满挑战的研究领域. 本文在此列出一些重要并且充满挑战性的问题.

(1) 跨链交互分层解耦: 跨链交互需要解决跨链信息传输、跨链信任传递与跨链操作原子性保障 3 个关键问题, 然而现有跨链方案信息传输与信任传递关键技术紧耦合, 跨链操作原子性保障与应用紧耦合, 限制了各类关键技术的独立发展, 制约了各类关键技术在功能、性能方面的优化, 增大了应用设计开发复杂度. 因此, 未来跨链研究需要解耦跨链交互, 针对各关键问题独立地深入研究解决方案, 实现功能完善与性能提升, 进一步可以通过通用层间接口的抽象与设计, 实现跨链交互分层协议栈.

(2) 可靠跨链传输机制: 跨链信息传输是跨链交互的基础, 现有跨链方案在传输层面仅能实现不可靠传输, 部分方案通过冗余的转发节点提升跨链传输的可靠性, 但是这种方法效果有限且会增大跨链开销. 可靠的跨链传输能有效支撑跨链其他关键技术研究, 有助于提升跨链服务质量, 并能满足未来更多跨链应用的需求. 因此, 未来需要研究容忍拜占庭节点、容忍崩溃等不同程度的可靠、低冗余的跨链传输机制.

(3) 轻量跨链信任传递技术: 现有跨链方案大多采用在链上构建对端链轻客户端的方法实现跨链信任传递, 然而这种方法链上开销与对端链成正比, 整体信任传递开销线形增长, 当跨链交易比例较小时, 每笔跨链交易平均信任传递开销较大. 未来可以根据跨链交易比例, 在现有区块链超轻量验证的基础上, 研究亚线形的、轻量的跨链信任传递技术.

(4) 高效原子性保障技术: 现有跨链方案中, 单跨链操作在多主体、长流程的场景下, 不同区块链上的处理串行执行, 导致单事务耗时长; 因区块链交易串行执行, 互不相关的多跨链操作在同一区块链内均串行执行, 导致区块链系统事务处理效率低, 未来可以从单跨链操作并行与多跨链操作并发等角度研究高效原子性保障技术.

(5) 跨链隐私与安全: 当前跨链研究聚焦于跨链方案的研究与实现, 对跨链技术打通区块链间壁垒后带来的跨链双花等安全性问题以及原有单链隐私保护、权限控制技术无法满足跨链需求的跨链隐私保护问题关注较少, 未来需要分析跨链技术带来的隐私保护问题与安全性问题, 并研究对应的方案.

6.2 总 结

跨链技术是打破当前区块链间数据孤岛现状的关键技术, 是当今区块链技术发展的迫切需求. 本文首先在区

分狭义与广义区块链互操作问题的基础上, 重新定义狭义区块链互操作, 并分析实现狭义区块链互操作需要解决的关键技术问题. 随后, 本文分别综述、讨论了各关键技术问题的研究现状, 包括跨链信息传输、跨链信任传递以及跨链操作原子性保证, 并进一步分析了当前具有代表性的跨链整体解决方案. 最后, 本文指出了几个值得进一步探索的研究方向. 总体而言, 跨链技术研究还在起步节点, 目前, 保证区块链可以相互操作尚有许多理论与技术问题亟待解决, 实现可以实际应用的、安全高效的区块链互操作则更是充满挑战.

References:

[1] Dabbagh M, Sookhak M, Safa NS. The evolution of blockchain: A bibliometric study. IEEE Access, 2019, 7: 19212–19221. [doi: 10.1109/ACCESS.2019.2895646]

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

[3] BTC Relay: Ethereum contract for Bitcoin SPV. 2017. https://github.com/ethereum/btcrelay

[4] Peace Relay. 2017. https://github.com/loiluu/peacerelay

[5] Frauenthaler P, Sigwart M, Spanring C, Sober M, Schulte S. ETH Relay: A cost-efficient relay for Ethereum-based blockchains. In: Proc. of the 2020 IEEE Int'l Conf. on Blockchain (Blockchain). Rhodes: IEEE, 2020. 204–213. [doi: 10.1109/Blockchain50366.2020.00032]

[6] Lerner SD. RSK white paper overview. 2015. https://bravenewcoin.com/assets/Whitepapers/RootstockWhitePaperv9-Overview.pdf

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

[8] Herlihy M. Atomic cross-chain swaps. In: Proc. of the 2018 ACM Symp. on Principles of Distributed Computing. Egham: ACM, 2018. 245–254. [doi: 10.1145/3212734.3212736]

[9] HTLC implementation in the wallet. 2017. https://github.com/bitcoin/bips/blob/master/bip-0199.mediawiki

[10] Decred-compatible cross-chain atomic swapping. 2021. https://github.com/decred/atomicswap

[11] The Komodo Organization. BarterDEX—Atomic swap decentralized exchange of native coins. 2017. https://github.com/SuperNETorg/komodo/wiki/barterDEX-Whitepaper-v2

[12] Zyskind G, Kisagun C, Fromknecht C. Enigma catalyst: A machine-based investing platform and infrastructure for crypto-assets. 2018. https://assets.coingecko.com/paper_documents/documents/222/open-uri20180806-11-sc59sg.?1533563479

[13] WeCross. 2021. https://wecross.readthedocs.io/zh_CN/latest/

[14] BitXHub. 2023. https://bitxhub.cn

[15] Kwon J, Buchman E. Cosmos whitepaper. 2019. https://wikibitimg.fx994.com/attach/2020/12/16623142020/WBE16623142020_55300.pdf

[16] Wood G. Polkadot: Vision for a heterogeneous multi-chain framework. 2016. https://assets.polkadot.network/Polkadot-whitepaper.pdf

[17] Buterin V. Chain interoperability. 2016. https://r3.com/reports/chain-interoperability/

[18] Belchior R, Vasconcelos A, Guerreiro S, Correia M. A survey on blockchain interoperability: Past, present, and future trends. ACM Computing Surveys, 2021, 54(8): 168. [doi: 10.1145/3471140]

[19] Singh A, Click K, Parizi RM, Zhang Q, Dehghantanha A, Choo KKR. Sidechain technologies in blockchain networks: An examination and state-of-the-art review. Journal of Network and Computer Applications, 2020, 149: 102471. [doi: 10.1016/j.jnca.2019.102471]

[20] Johnson S, Robinson P, Brainard J. Sidechains and interoperability. arXiv:1903.04077, 2019.

[21] Zamyatin A, Al-Bassam M, Zindros D, Kokoris-Kogias E, Moreno-Sanchez P, Kiayias A, Knottenbelt WJ. SoK: Communication across distributed ledgers. In: Proc. of the 25th Int'l Conf. on Financial Cryptography and Data Security. Springer, 2021. 3–36. [doi: 10.1007/978-3-662-64331-0_1]

[22] Robinson P. Survey of crosschain communications protocols. Computer Networks, 2021, 200: 108488. [doi: 10.1016/j.comnet.2021.108488]

[23] Schulte S, Sigwart M, Frauenthaler P, Borkowski M. Towards blockchain interoperability. In: Proc. of the 2019 BPM Blockchain and CEE Forum on Business Process Management: Blockchain and Central and Eastern Europe Forum. Vienna: Springer, 2019. 3–10. [doi: 10.1007/978-3-030-30429-4_1]

[24] China Academy of Information and Communications Technology. White Paper for Blockchain Interoperability. Beijing: China Academy of Information and Communications Technology, 2020 (in Chinese). http://www.caict.ac.cn/kxyj/qwfb/bps/202012/P020201230759713827891.pdf

[25] Jin H, Dai XH, Xiao J. Towards a novel architecture for enabling interoperability amongst multiple blockchains. In: Proc. of the 38th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Vienna: IEEE, 2018. 1203–1211. [doi: 10.1109/ICDCS.2018.00120]

[26] Lafourcade P, Lombard-Platet M. About blockchain interoperability. Information Processing Letters, 2020, 161: 105976. [doi: 10.1016/j.

ipl.2020.105976]
[27] WeBankBlockchain. WeCross: The blockchain interoperability platform white paper. 2020 (in Chinese). http://www.d-long.com/eWebEditor/uploadfile/20200416161636165570 31.pdf
[28] Chainmaker. 2022. https://docs.chainmaker.org.cn/index.html
[29] Goes C. The interblockchain communication protocol: An overview. arXiv:2006.15918, 2020.
[30] Zamyatin A, Harz D, Lind J, Panayiotou P, Gervais A, Knottenbelt W. XCLAIM: Trustless, interoperable, cryptocurrency-backed assets. In: Proc. of the 2019 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2019. 193–210. [doi: 10.1109/SP.2019.00085]
[31] Dilley J, Poelstra A, Wilkins J, Piekarska M, Gorlick B, Friedenbach M. Strong federations: An interoperable blockchain solution to centralized third party risks. arXiv:1612.05491, 2016.
[32] Garoffolo A, Kaidalov D, Oliynykov R. Zendoo: A zk-SNARK verifiable cross-chain transfer protocol enabling decoupled and decentralized sidechains. In: Proc. of the 40th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Singapore: IEEE, 2020. 1257–1262. [doi: 10.1109/ICDCS47774.2020.00161]
[33] Teutsch J, Straka M, Boneh D. Retrofitting a two-way peg between blockchains. arXiv:1908.03999, 2019.
[34] Kiayias A, Zindros D. Proof-of-work sidechains. In: Proc. of the 2019 Int'l Workshops on Financial Cryptography and Data Security. St. Kitts: Springer, 2019. 21–34. [doi: 10.1007/978-3-030-43725-1_3]
[35] Duffy M J. PlasmaChain, GameChain, SocialChain: The loom network universe expounded. 2018. https://medium.com/loom-network/plasmachain-gamechain-socialchain-the-loom-network-universe-expounded-5c672617a333
[36] Wang H, He D, Gao Y, Wang XY, Xu CC, Qiu WW, Yao YY, Wang Q. Research on data verification and exchange of heterogeneous blockchains for electricity application. Journal of Physics: Conf. Series, 2020, 1631(1): 012154. [doi: 10.1088/1742-6596/1631/1/012154]
[37] Web3 Foundation Research. Polkadot's Messaging Scheme. 2020. https://medium.com/web3foundation/polkadots-messaging-scheme-b1ec560908b7
[38] Xiao XT, Yu Z, Xie K, Guo SY, Xiong A, Yan Y. A multi-blockchain architecture supporting cross-blockchain communication. In: Proc. of the 6th Int'l Conf. on Artificial Intelligence and Security. Hohhot: Springer, 2020. 592–603. [doi: 10.1007/978-981-15-8086-4_56]
[39] Luo K, Yu W, Muhammad AH, Wang SY, Ling CG, Hu K. A multiple blockchains architecture on inter-blockchain communication. In: Proc. of the 2018 IEEE Int'l Conf. on Software Quality, Reliability and Security Companion (QRS-C). Lisbon: IEEE, 2018. 139–145. [doi: 10.1109/QRS-C.2018.00037]
[40] Belchior R, Vasconcelos A, Correia M, Hardjono T. Enabling cross-jurisdiction digital asset transfer. In: Proc. of the 2021 IEEE Int'l Conf. on Services Computing (SCC). Chicago: IEEE, 2021. 431–436. [doi: 10.1109/SCC53864.2021.00062]
[41] Belchior R, Vasconcelos A, Correia M, Hardjono T. Hermes: Fault-tolerant middleware for blockchain interoperability. Future Generation Computer Systems, 2022, 129: 236–251. [doi: 10.1016/j.future.2021.11.004]
[42] Hyperchain. BitXHub: The cross-chain technology platform white paper. 2022 (in Chinese). https://upload.hyperchain.cn/BitXHub 白皮书.pdf
[43] Sztorc P. Drivechain—The simple two way peg. 2015. https://www.truthcoin.info/blog/drivechain/#drivechain-a-simple-spv-proof
[44] Sergio DL. Drivechains, sidechains and hybrid 2-way peg designs. 2016. https://docs.rsk.co/Drivechains_Sidechains_and_Hybrid_2-way_peg_Designs_R9.pdf
[45] Intel Corporation. Intel® software guard extensions programming reference. 2014. https://www.intel.com/content/dam/develop/external/us/en/documents/329298-002-629101.pdf
[46] Robinson P, Ramesh R. General purpose atomic crosschain transactions. In: Proc. of the 3rd Conf. on Blockchain Research & Applications for Innovative Networks and Services (BRAINS). Paris: IEEE, 2021. 61–68. [doi: 10.1109/BRAINS52497.2021.9569837]
[47] Goldreich O, Oren Y. Definitions and properties of zero-knowledge proof systems. Journal of Cryptology, 1994, 7(1): 1–32. [doi: 10.1007/BF00195207]
[48] Westerkamp M, Eberhardt J. zkRelay: Facilitating sidechains using zkSNARK-based chain-relays. In: Proc. of the 2020 IEEE European Symp. on Security and Privacy Workshops (EuroS&PW). Genoa: IEEE, 2020. 378–386. [doi: 10.1109/EuroSPW51379.2020.00058]
[49] Buterin V. Ethereum whitepaper: A next-generation smart contract and decentralized application platform. 2014. https://ethereum.org/en/whitepaper/
[50] Merkle RC. A digital signature based on a conventional encryption function. In: Advances in Cryptology. Berlin: Springer, 1988. 369–378. [doi: 10.1007/3-540-48184-2_32]
[51] Kerala Blockchain Academy. Merkle Patricia trie in Ethereum: A silhouette. 2021. https://kbaiiitmk.medium.com/merkle-patricia-trie-in-ethereum-a-silhouette-c8d04155b490
[52] Xia Q, Dou WS, Guo KW, Liang G, Zuo C, Zhang FJ. Survey on blockchain consensus protocol. Ruan Jian Xue Bao/Journal of Software,

2021, 32(2): 277−299 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6150.htm [doi: 10.13328/j.cnki.jos.006150]

[53] Castro M, Liskov B. Practical Byzantine fault tolerance. In: Proc. of the 3rd Symp. on Operating Systems Design and Implementation. New Orleans: USENIX Association, 1999. 173–186.

[54] Eyal I, Gencer AE, Sirer EG, Van Renesse R. Bitcoin-NG: A scalable blockchain protocol. In: Proc. of the 13th USENIX Symp. on Networked Systems Design and Implementation. Santa Clara: USENIX Association, 2016. 45–59.

[55] 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. [doi: 10.1145/2695533.2695545]

[56] Bentov I, Gabizon A, Mizrahi A. Cryptocurrencies without proof of work. In: Proc. of the 2016 Int'l Workshops on Financial Cryptography and Data Security. Christ Church: Springer, 2016. 142–157. [doi: 10.1007/978-3-662-53357-4_10]

[57] Gilad Y, Hemo R, Micali S, Vlachos G, Zeldovich N. Algorand: Scaling byzantine agreements for cryptocurrencies. In: Proc. of the 26th Symp. on Operating Systems Principles. Shanghai: ACM, 2017. 51–68. [doi: 10.1145/3132747.3132757]

[58] Kiayias A, Miller A, Zindros D. Non-interactive proofs of proof-of-work. In: Proc. of the 24th Int'l Conf. on Financial Cryptography and Data Security. Kota Kinabalu: Springer, 2020. 505–522. [doi: 10.1007/978-3-030-51280-4_27]

[59] Kiayias A, Lamprou N, Stouka AP. Proofs of proofs of work with sublinear complexity. In: Proc. of the 2016 Int'l Workshops on Financial Cryptography and Data Security. Christ Church: Springer, 2016. 61–78. [doi: 10.1007/978-3-662-53357-4_5]

[60] Bünz B, Kiffer L, Luu L, Zamani M. FlyClient: Super-light clients for cryptocurrencies. In: Proc. of the 2020 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2020. 928–946. [doi: 10.1109/SP40000.2020.00049]

[61] zcash. 2023. https://github.com/zcash/zcash

[62] Westerkamp M. Verifiable smart contract portability. In: Proc. of the 2019 IEEE Int'l Conf. on Blockchain and Cryptocurrency (ICBC). Seoul: IEEE, 2019. 1–9. [doi: 10.1109/BLOC.2019.8751335]

[63] Fynn E, Bessani A, Pedone F. Smart contracts on the move. In: Proc. of the 50th Annual IEEE/IFIP Int'l Conf. on Dependable Systems and Networks (DSN). Valencia: IEEE, 2020. 233–244. [doi: 10.1109/DSN48063.2020.00040]

[64] Westerkamp M, Küpper A. SmartSync: Cross-blockchain smart contract interaction and synchronization. In: Proc. of the 2022 IEEE Int'l Conf. on Blockchain and Cryptocurrency (ICBC). Shanghai: IEEE, 2022. 1–9. [doi: 10.1109/ICBC54727.2022.9805524]

[65] Buterin V. The stateless client concept. 2017. https://ethresear.ch/t/the-stateless-client-concept/172

[66] Dryja T. Utreexo: A dynamic hash-based accumulator optimized for the Bitcoin UTXO set. 2019. https://eprint.iacr.org/2019/611

[67] Bailey B, Sankagiri S. Merkle trees optimized for stateless clients in Bitcoin. In: Proc. of the 2021 Int'l Conf. on Financial Cryptography and Data Security. Springer, 2021. 451–466. [doi: 10.1007/978-3-662-63958-0_35]

[68] Chepurnoy A, Papamanthou C, Srinivasan S, Zhang YP. Edrax: A cryptocurrency with stateless transaction validation. Cryptology ePrint Archive, 2018.

[69] Boneh D, Bünz B, Fisch B. Batching techniques for accumulators with applications to IOPs and stateless blockchains. In: Proc. of the 39th Annual Int'l Cryptology Conf. on Advances in Cryptology. Santa Barbara: Springer, 2019. 561–586. [doi: 10.1007/978-3-030-26948-7_20]

[70] Gorbunov S, Reyzin L, Wee H, Zhang ZF. Pointproofs: Aggregating proofs for multiple vector commitments. In: Proc. of the 2020 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2020. 2007–2023. [doi: 10.1145/3372297.3417244]

[71] Tomescu A, Abraham I, Buterin V, Drake J, Feist D, Khovratovich D. Aggregatable subvector commitments for stateless cryptocurrencies. In: Proc. of the 12th Int'l Conf. on Security and Cryptography for Networks. Amalfi: Springer, 2020. 45–64. [doi: 10.1007/978-3-030-57990-6_3]

[72] Bitcoin Wiki. Hashlock. 2019. https://en.bitcoin.it/wiki/Hashlock

[73] Bitcoin Wiki. Timelock. 2022. https://en.bitcoin.it/wiki/Timelock

[74] Belotti M, Moretti S, Potop-Butucaru M, Secci S. Game theoretical analysis of cross-chain swaps. In: Proc. of the 40th IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Singapore: IEEE, 2020. 485–495. [doi: 10.1109/ICDCS47774.2020.00060]

[75] Winzer F, Herd B, Faust S. Temporary censorship attacks in the presence of rational miners. In: Proc. of the 2019 IEEE European Symp. on Security and Privacy Workshops (EuroS&PW). Stockholm: IEEE, 2019. 357–366. [doi: 10.1109/EuroSPW.2019.00046]

[76] Tsabary I, Yechieli M, Manuskin A, Eyal I. MAD-HTLC: Because HTLC is crazy-cheap to attack. In: Proc. of the 2021 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2021. 1230–1248. [doi: 10.1109/SP40001.2021.00080]

[77] Nadahalli T, Khabbazian M, Wattenhofer R. Timelocked bribing. In: Proc. of the 25th Int'l Conf. on Financial Cryptography and Data Security. Springer, 2021. 53–72. [doi: 10.1007/978-3-662-64322-8_3]

[78] Harris J, Zohar A. Flood & loot: A systemic attack on the lightning network. In: Proc. of the 2nd ACM Conf. on Advances in Financial Technologies. New York: ACM, 2020. 202–213. [doi: 10.1145/3419614.3423248]

[79] Zakhary V, Agrawal D, Abbadi AE. Atomic commitment across blockchains. Proc. of the VLDB Endowment, 2020, 13(9): 1319–1331. [doi: 10.14778/3397230.3397231]

[80] Zie JY, Deneuville JC, Briffaut J, Nguyen B. Extending atomic cross-chain swaps. In: Proc. of the 2019 Int'l Workshops on Data Privacy Management, Cryptocurrencies and Blockchain Technology. Luxembourg: Springer, 2019. 219–229. [doi: 10.1007/978-3-030-31500-9_14]

[81] Lipton A, Hardjono T. Blockchain intra- and interoperability. In: Babich V, Birge JR, Hilary G. Innovative Technology at the Interface of Finance and Operations: Vol. II. Cham: Springer, 2022. 1–30. [doi: 10.1007/978-3-030-81945-3_1]

[82] Gray J. The transaction concept: Virtues and limitations. In: Proc. of the 7th Int'l conf. on Very Large Data Bases. Cannes: VLDB Endowment, 1981. 144–154.

[83] Traiger IL, Gray J, Galtieri CA, Lindsay BG. Transactions and consistency in distributed database systems. ACM Trans. on Database Systems, 1982, 7(3): 323–342. [doi: 10.1145/319732.319734]

[84] Zhao DF, Li TL. Distributed cross-blockchain transactions. arXiv:2002.11771, 2020.

[85] Liu ZT, Xiang YX, Shi J, Gao P, Wang HY, Xiao XS, Wen BH, Hu YC. HyperService: Interoperability and programmability across heterogeneous blockchains. In: Proc. of the 2019 ACM SIGSAC Conf. on Computer and Communications Security. London: ACM, 2019. 549–566. [doi: 10.1145/3319535.3355503]

附中文参考文献:

[24] 中国信息通信研究院. 可信区块链推进计划: 区块链互操作白皮书. 北京: 中国信息通信研究院, 2020. http://www.caict.ac.cn/kxyj/qwfb/bps/202012/P020201230759713827891.pdf

[27] 微众银行区块链团队. WeCross技术白皮书: 区块链跨链协作平台. 2020. http://www.d-long.com/eWebEditor/uploadfile/20200416161636165570 31.pdf

[42] 趣链科技. BitXHub白皮书V2.0: 区块链跨链技术平台. 2022. https://upload.hyperchain.cn/BitXHub 白皮书.pdf

[52] 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, 2021, 32(2): 277–299. http://www.jos.org.cn/1000-9825/6150.htm [doi: 10.13328/j.cnki.jos.006150]

段田田(1996－), 女, 博士生, CCF 学生会员, 主要研究领域为区块链, 分布式系统.

李忠诚(1962－), 男, 博士, 研究员, 博士生导师, CCF 高级会员, 主要研究领域为计算机网络, 区块链.

张瀚文(1981－), 女, 博士, 副研究员, CCF 高级会员, 主要研究领域为计算机网络, 区块链.

张珺(1975－), 女, 博士, 副教授, CCF 高级会员, 主要研究领域为区块链, 下一代互联网, 信息安全.

李博(1996－), 男, 博士生, 主要研究领域为区块链, 数据定价与性能分析.

孙毅(1979－), 男, 博士, 研究员, 博士生导师, CCF 杰出会员, 主要研究领域为区块链, 分布式系统, 网络体系结构.

宋兆雄(1993－), 男, 工程师, CCF 专业会员, 主要研究领域为区块链, 分布式系统.