可编辑区块链的研究现状与挑战

罗 彬[1], 温金明[1,2], 吴永东[1], 陈 洁[3]

[1]暨南大学信息科学技术学院 广州 中国 510632
[2]密码科学技术国家重点实验室 北京 中国 100878
[3]华东师范大学软件工程学院 上海 中国 200062

摘要 区块链是一种去中心化和不可篡改的数据库, 不仅能有效解决需要第三方介入所引发的信任问题, 还能保证应用执行过程公平公正。然而, 不可篡改性限制了区块链的许多应用。例如, 不同机构需要删除区块链上存储的非法数据或对旧数据进行编辑。因此, 如何设计一种可编辑的区块链, 对链上数据进行合法编辑是拓展区块链应用的一个重要问题。本文旨在对当前主流的可编辑区块链方案进行系统性调查和分析。首先, 总结了这些方案提出的时间脉络和编辑操作的统一流程, 针对编辑共识、编辑处理和编辑证明三个重要环节, 分别从编辑操作的种类、编辑所属权、编辑涉及的链结构、编辑对象粒度、编辑过程使用的数据结构和编辑共识技术六个不同角度对典型方案进行了细粒度分类。其次, 概括了每个方案的核心思想、新颖之处和优缺点, 并对方案涉及的密码学原语、常用技术和共识机制等重要组成部分进行了全面回顾, 包括变色龙哈希函数、秘密共享、委员会选取、激励惩罚机制、工作量证明、权益证明等。再次, 根据身份管理、物联网和外包的不同场景需求, 分类讨论了几个利用可编辑区块链技术的最新应用方案, 以及区块链可编辑功能所产生的潜在价值。最后, 指出了目前方案有待完善或尚未解决的问题, 这些问题将提供新的研究方向、拓展区块链的应用场景、加速区块链与更多领域结合、促进区块链技术的发展。

关键词 可编辑区块链; 变色龙哈希; 投票; 共识机制
中图法分类号 TP311.13 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2023.07.05

State of the Art and Challenges of Redactable Blockchain

LUO Bin[1], WEN Jinming[1,2], WU Yongdong[1], CHEN Jie[3]

[1]College of Information Science and Technology, Jinan University, Guangzhou 510632, China
[2]China State Key Laboratory of Cryptology, Beijing 100878, China
[3]Software Engineering Institute, East China Normal University, Shanghai 200062, China

Abstract As a decentralized and immutable database, blockchain cannot only effectively solves the trust problems caused by the need for third-party intervention, but also ensures the fairness and justice of the application execution process. However, immutability limits a multitude of applications of blockchain. For example, different institutions need to delete illegal data stored on the blockchain or edit old data. Therefore, how to design a redactable blockchain and legally redact the data on the blockchain is an important problem for expanding the application of blockchain. This article aims to systematically investigate and analyze the current redactable blockchain schemes. Firstly, the timeline of these schemes and the unified process of redacting operations are summarized. Aiming at the three important segments of redacting consensus, redacting processing, and redacting proof, the typical schemes are classified in fine-grained from six different perspectives: redacting operation type, redacting right ownership, chain structure involved in redacting, redacting object granularity, the data structure used in the redacting process and redacting consensus technology. Secondly, this article summarizes the main idea, novelty, advantages, and disadvantages of each scheme, and comprehensively reviews the important components of these schemes, such as cryptography primitives, common technologies, and consensus mechanisms, including chameleon hash function, secret sharing, committee selection, incentive, and punishment mechanisms, proof of work and proof of stake, etc. Thirdly, according to the different application requirements of identity management, the Internet of Things, and outsourcing, several latest application schemes using redactable blockchain technology and the potential value generated by the redactable function of blockchain are classified and discussed. Finally, we present some open problems that need to be improved or not solved in the current schemes, which may lead to new research directions, expand the application scenarios of blockchain, accelerate the combination of blockchain and more fields, and promote the development of blockchain technology.

通讯作者: 温金明, 博士, 研究员, Email: jinmingwen1@163.com。

本课题得到国家自然科学基金(No. 11871248、No. 12271215、No. 61972156、No. 61932011), 广东省高等学校珠江学者岗位计划资助项目, 广东省自然科学基金(No. 2021A1515010857), 广东省基础与应用基础研究基金联合基金(No. 2019B1515120010), 广东省重点领域研发项目 (No. 2020B0101090002), 科技部重点研发项目(No. 2020YFB1005600)资助。
收稿日期: 2021-12-24; 修改日期: 2022-02-21; 定稿日期: 2023-04-18

Key words redactable blockchain; chameleon hash; e-voting; consensus

1 引言

区块链是比特币的底层核心技术, 可以看作是一种由包含交易信息的区块有序链接起来的数据结构, 通常被视为一个公开账本。它也是在分布式、不可信环境中, 所有节点通过点对点网络执行某种共识机制, 就公共账本的交易状态达成一致的技术。区块链一般具有以下几个特点: 去中心化、公开透明性以及不可篡改性, 其中去中心化和不可篡改性是区别于其他技术最重要的性质, 赋予了大众对区块链技术的信任, 工商业界也在考虑把区块链技术部署到更多实际应用中, 如医疗[1]、物流[2]、政务[3]等, 同时区块链作为又一次产业革命的新兴要素, 有助于打造未来数字经济生态。

1.1 研究需求和意义

近几年, 随着区块链应用的不断落地, 越来越多案例表明不可篡改性具有两面性: 它虽然保障了区块链数据的安全和可信, 但也在一定程度上限制了区块链的应用。例如, 在许多业务中存在纠正错误数据或删除过时数据的需求, 然而区块链的不可篡改性阻碍了这一功能的实现。此外, 如果恶意方在区块链中存储了一些不恰当的内容, 如非法图片、文件、恶意言语等, 这些会给社会造成不良舆论, 也会给区块链的数据安全带来负面影响。

因此, 虽然区块链的不可篡改性对其安全有着重要作用, 但是该性质也限制了区块链的应用场景, 如政府、服务商需要对存储的公民和用户数据进行管理。如果区块链不可编辑, 那么即使公民和用户的数据资料出现错误也不能更改, 无法在原位对数据纠错; 另一方面, 随着区块链的发展, 其所需的存储空间也在不断增长, 如果任何数据都被允许永久保留在链上而不修剪的话, 可能导致拥有较弱硬件设备的用户无法使用区块链的相关应用; 其次, 在智能合约中, 如果后期发现合约存在漏洞, 现有方法只能通过硬分叉进行漏洞修复[4], 这不仅会影响其他交易的结果, 而且不利于区块链系统的稳定; 在信息管理方面, 近些年报道了在区块链中, 特别是在比特币中, 存在着一些非法数据, 包括图片、链接、文档等[5], 由于区块链的不可篡改性使得去除非法数据变得几乎不可能; 再者在隐私保护方面, 欧盟的数据保护通用条例(General Data Protection Regulation, GDPR)等法规的发布, 要求赋予公民遗忘权(The Right To Be Forgotten), 然而目前的区块链因不可篡改性是无法满足这些法规的要求[6]。综上, 针对许多应用场景, 有待提出一种安全可控和高效编辑区块链数据的方案。

1.2 区块链编辑技术

可编辑区块链是一类具有编辑链上数据功能的区块链, 其编辑操作主要针对区块链的不可篡改性。即在不破坏区块链的其他性质且满足一定条件时, 实现链上数据的删除、修改、插入操作。第一个可编辑方案是在 2017 年由 Ateniese 等人提出[7], 该方案使用变色龙哈希函数代替了区块链原本的哈希函数, 在保持哈希值不变的情况下能够修改区块链数据, 但由于变色龙哈希函数陷门的存在, 引入了密钥管理问题。此外, 该方案使用安全多方计算和秘密共享等一些复杂的密码学技术, 会带来高昂的通信开销, 不适合应用在大规模的公链场景。2019 年 Deuber 等人基于投票机制在公链上提出了可编辑方案[8], 该方案没有使用复杂的密码学工具, 只是在区块中增加一个“旧状态”来维持哈希链的完整, 但方案中是以区块为单位进行修改, 而不是更加细粒度的交易级修改, 且缺乏对编辑权的限制。最近 Tian 等人提出了在公链中细粒度的重写方案[9], 方案主要使用了基于密钥策略的属性基加密实现对编辑权限的控制, 与之前方案相比在公开问责方面有所改进, 然而该方案中没有经过交易拥有者同意也可以修改数据, 可能存在恶意修改者对数据进行任意修改, 而且仍存在密钥管理问题。

在过去的 5 年中, 区块链的可编辑技术作为新兴的研究方向, 针对目前可编辑方案的缺陷, 学者们致力于研究一种安全、可控、高效的编辑方法来插入、删除和修改区块链中的数据, 同时不会影响区块链的完整性。他们先后提出了一些可编辑区块链的方案, 图 1 展示了不同类型的可编辑区块链方案提出的时间表, 包括基于变色龙哈希和基于投票的方案。

在本文中, 我们依据可编辑区块链的几个特征, 对这些方案进行了分类, 如图 2 所示。一般来说, 我们大致可从六个角度对它们进行分类: 编辑操作种类、编辑所属权、编辑涉及的链结构、编辑对象粒度、编辑过程使用的数据结构、编辑共识技术。

此外, 我们根据目前的可编辑区块链方案, 概括出其总体流程如图 3 所示。具体来说, 主要分为以下五个步骤:

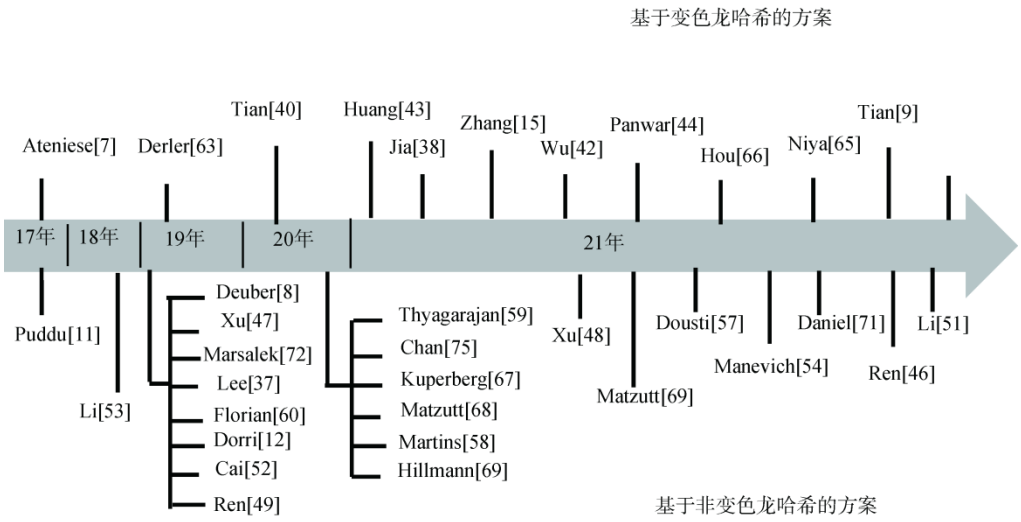

图 1　近五年可编辑区块链发展的时间表

Figure 1　Development schedule of redactable blockchain in recent five year

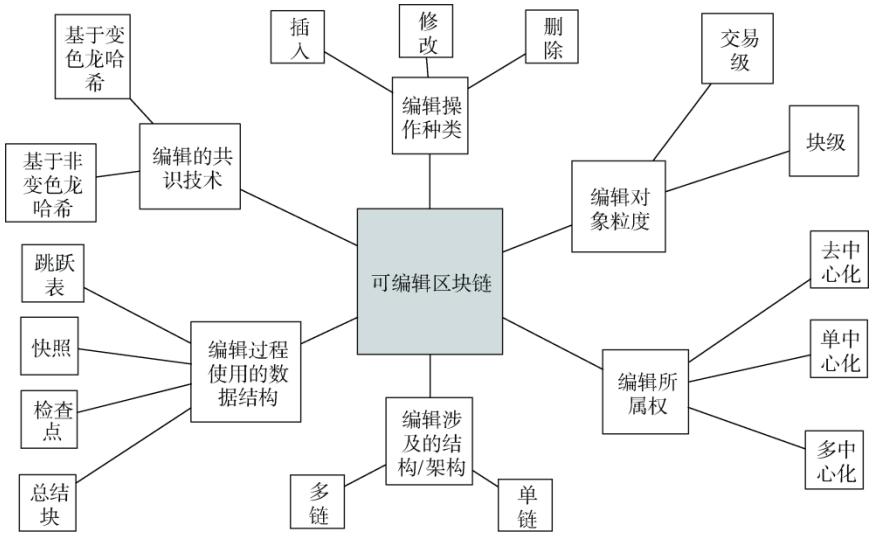

图 2　可编辑区块链研究分类

Figure 2　Classification of redactable blockchain scheme

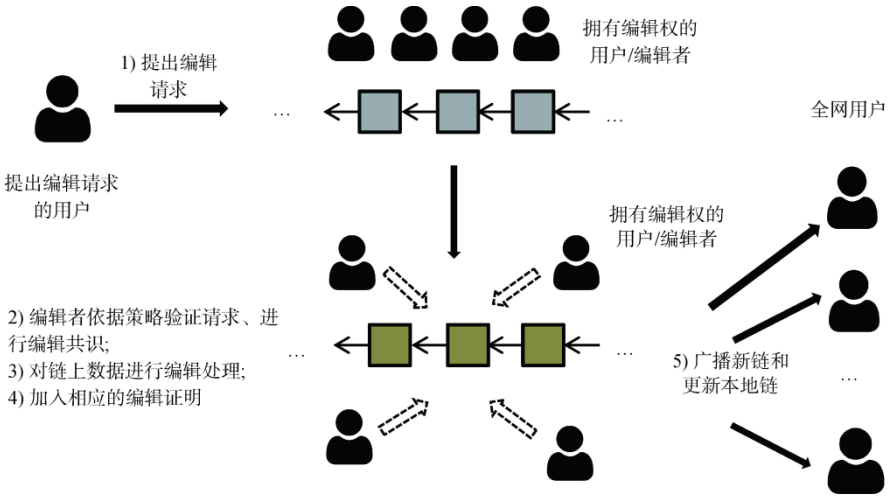

图 3　可编辑区块链方案的总体流程

Figure 3　The overall process of redactable blockchain scheme

1) 区块链用户提出编辑请求。通常请求包括编辑交易或编辑区块的索引、编辑后交易或区块的内容、编辑操作的种类以及其他附加元素。不同方案对提出编辑请求的用户身份具有不同要求, 比如在非许可链中可以是区块链系统中的任何用户, 而在许可链中可能只是交易拥有者或其指定的用户等。

2) 拥有编辑权的用户(编辑者)对请求进行验证和编辑共识。当编辑者收到编辑请求时, 首先依据指定策略对请求内容进行验证, 若验证通过, 则在编辑者之间对是否同意该编辑请求进行共识。

3) 在达成共识后, 编辑者根据编辑请求的内容处理链上数据。一般根据编辑类型和对象, 在编辑处理的过程中采用不同的方法对数据进行编辑。

4) 为了区分未编辑和已编辑的数据, 编辑者在链上需要加入相应的编辑证明, 必要时会借助辅助结构。

5) 在编辑操作完成后编辑者会向全网广播新链。当每个节点收到更新消息时, 依据策略对消息内容进行验证, 验证通过后更新自己的本地区块链。

1.3 相关工作

Politou 等人的可编辑区块链综述[10]侧重于遗忘权问题, 回顾了几个可编辑区块链的解决方案[7-8,11-12]和一些密码学技术[13], 并讨论了在许可或无许可区块链中应用它们的潜力和局限性。但文章并没有把几个可编辑区块链方案进行系统分类讨论, 也没有提供今后可能的研究方向或有待解决的问题。Yuan 等人的综述[14]提出了可编辑区块链的工作框架, 并从数据修改、删除、插入、过滤和隐藏五个环节阐述可编辑区块链的技术与方法, 讨论了该领域亟待解决的一些问题。虽然文章对现有可编辑区块链方案进行了分类讨论, 但主要针对的是编辑操作的类型这一角度, 没有从编辑使用的数据结构等更多角度对可编辑方案进行探讨。Zhang 等人的综述[15]把目前区块链编辑机制分为四类, 分析了每一类的安全和性能, 总结了在区块链中编辑数据的典型方法, 讨论了设计可编辑方案的挑战, 并提出一系列评估标准。但是文章给出四类修改机制中不包括插入操作, 并且也没有对可编辑区块链技术的相关应用案例进行讨论。

一方面, 目前的综述文献中明显缺乏对现有可编辑区块链方案涉及的共识机制的回顾, 以及对方案更加细粒度的分类与讨论。另一方面, 若从应用场景的角度对可编辑方案的实现进行充分探讨, 会有利于可编辑技术的发展, 但现有的综述文章没有讨论可编辑区块链的应用方案, 例如将区块链的可编辑操作应用在指纹识别系统[16], 其中利用变色龙哈希函数对链上数据进行编辑。特别是如图 1 所示, 变色龙哈希函数作为可编辑区块链的重要密码学工具, 近几年来在密码学界中发展迅速[17], 为可编辑区块链方案的安全性和实用性提供了可靠保证。

总的来说, 现有的综述没有提供统一的编辑流程, 也没有对先进的可编辑区块链的应用案例、相关的共识机制和密码学工具进行详细分类和讨论, 我们这篇综述的主要目的就是填补上述不足之处, 针对区块链的可编辑技术进行总结归纳。本文和上述讨论的可编辑区块链综述文章的对比如表 1 所示。

表 1 本文和现有可编辑区块链综述文章的对比

Table 1 Comparison between this article and existing redactable blockchain overview articles

文章	是否进行分类讨论	是否提供未来可能的研究方向	是否讨论密码学技术	是否提供编辑方案的统一流程	是否讨论应用方案	是否讨论共识机制
Politou[10]	×	×	√	×	×	×
Yuan[14]	√	√	√	×	×	×
Zhang[15]	√	√	×	×	×	×
本文	√	√	√	√	√	√

1.4 贡献和文章组织

在本文中, 我们填补了上述综述中的空白, 总结了目前可编辑区块链方案的统一流程, 针对其中三个重要环节对方案进行了细粒度分类, 给出了相应的代表方案, 分析各自的核心思想、创新点以及优缺点, 并对可编辑区块链涉及的密码学工具、常用技术、共识机制等方面进行全面的回顾和讨论, 这些技术是可编辑方案的重要组成部分。此外, 我们还讨论了区块链的可编辑功能所带来的潜在应用价值, 总结目前方案尚待解决或需要完善的问题, 为相关研究人员提供新的见解和未来可能的研究方向, 以拓展区块链的使用场景, 实现更实用的区块链解决方案。

本文的其余部分组织如下: 第 2 节介绍了可编辑区块链的预备知识, 包括相关密码学工具、方案常用的技术等。第 3 节从编辑共识、编辑处理、编辑

证明这三个重要环节对目前可编辑区块链方案进行分类讨论。在第 4 节中, 我们分析了使用可编辑区块链的几个应用场景。最后, 第 5 节总结了目前方案有待完善或解决的问题, 并讨论潜在的研究方向。

2 预备知识

2.1 可编辑区块链涉及的密码学原语

密码学技术作为区块链的基石, 为其安全性提供了保障。其中哈希函数是区块链中最基础, 也是最核心的一个密码学技术, 它能将任意长度的字符串转化为固定长度的输出, 对其输入的微小改动也会导致输出结果的显著变化, 所以通常用于维护区块链数据的完整性和实现认证功能, 是区块链数据不可篡改的重要保障。本节主要介绍目前可编辑区块链方案中涉及的一些常用密码学原语, 包括变色龙哈希函数、秘密共享、安全多方计算、数字签名及它的拓展等。

2.1.1 变色龙哈希函数

变色龙哈希函数(Chameleon Hash Function)是由 Krawczyk 和 Rabin 提出的一种带陷门的单向哈希函数[18], 粗略的说变色龙哈希函数的原理是: 给定一个消息 m 及其哈希值 $H(m)$, 如果没有陷门则很难找到 m 的哈希碰撞, 但若知道陷门,那么持有陷门的实体可以有效地找到消息 m 的碰撞, 即可以找到 m'使得 $H(m)=H(m')$。文献[20]给出了正式定义, 一个变色龙哈希函数通常包含以下五个算法:

1) $pp \leftarrow \mathrm{CHPG}(1^{\lambda})$:公共参数生成算法 CHPG 以安全参数λ作为输入, 输出一个公共参数 pp, 该公共参数作为其他算法的隐式输入;

2) $(pk, sk) \leftarrow \mathrm{CHKG}(pp)$:密钥对生成算法 CHKG 以公共参数 pp 作为输入, 输出一对变色龙哈希的公私钥 pk 和 sk, 其中私钥 sk 也叫做陷门;

3) $(h, r) \leftarrow \mathrm{CHash}(pk, m)$:变色龙哈希生成算法 CHash 以公钥 pk 和消息 m 作为输入, 输出变色龙哈希值 h 和随机数 r;

4) $d \leftarrow \mathrm{CHV}(pk, m, h, r)$:哈希验证算法 CHV 以变色龙哈希的公钥 pk、消息 m、哈希值 h 以及随机数 r 作为输入, 输出一个布尔值 $d \in \{0,1\}$, 若哈希值 h 合法则输出 1, 反之为 0;

5) $r' \leftarrow \mathrm{HCol}(sk, m, m', h, r)$:碰撞生成算法 HCol 以变色龙哈希的陷门 sk、旧消息 m、新消息 m'、哈希值 h 和旧随机数 r 作为输入, 输出新随机数 r', 使得 $\mathrm{CHV}(pk, m, h, r) = \mathrm{CHV}(pk, m', h, r')$。

2.1.2 秘密共享

秘密共享(Secret sharing)是由 Shamir 提出的密码学方案[21], 目的是将秘密在多个实体之间安全地共享, 简单来说秘密共享的思想是把秘密分成 n 份并分发给不同实体, 只有不少于 t 个实体一起使用多项式插值才能恢复秘密。方案如下:

1) $(\tau_1, \tau_2, ..., \tau_n) \leftarrow \mathrm{Share}(s)$: 输入一个秘密值 $s \in \mathbb{Z}_p$, 选取随机系数$\alpha_1, \alpha_2, ..., \alpha_{t-1} \in \mathbb{Z}_p$, 生成一个 t–1 次的多项式$f(x) = s + \alpha_1 \cdot x + \alpha_2 \cdot x^2 + ... + \alpha_{t-1} \cdot x^{t-1} \bmod p$, 其中 $\mathbb{Z}$ 表示所有自然数构成的集合, p 是一个素数。然后计算将要分发的 n 个份额: $\tau_i = f(k_i), \forall i \in \{1,2,...,n\}, k_i \in \mathbb{Z}$, 通常$k_i = i$。

2) s 或 $\perp \leftarrow \mathrm{Rec}(\tau_1, \tau_2, ..., \tau_t)$: 输入 t 个份额$(\tau_1, \tau_2, ..., \tau_t)$, 使用插值计算出多项式$\alpha(X) = \alpha_0 + \alpha_1 \cdot X + \alpha_2 \cdot X^2 + ... + \alpha_{t-1} \cdot X^{t-1}$, 则$\alpha(0) = \alpha_0 = s$。

通常, 秘密共享技术需要和安全多方计算一起使用, 这样才能保护各方拥有的份额在重构时不会被泄露。

2.1.3 安全多方计算

安全多方计算(secure multi-party computation, SMPC)主要用于隐私保护的场景。SMPC 是 1982 年由姚期智先生提出的概念[22], 其主要目的是在没有信任基础的参与者间计算某个共同函数且不泄露各自的秘密信息。简单来说, SMPC 就是指一个分布式环境中有 n 个无信任基础的参与者$P_1, P_2, ..., P_n$, 各自拥有秘密值$s_1, s_2, ..., s_n$, 他们都想获取一些函数值 $f_1, f_2, ..., f_k$, 这些函数需要输入各方的秘密值, 即 $f_i(s_1, s_2, ..., s_n)$, 此时运用 SMPC 协议, 各方分别输入秘密值s_i进行协同计算得到所需要的函数值, 在整个计算的过程中, 各方秘密值都没有被泄漏。比如一个群体想计算该群体的平均年龄, 而又不想让别人知道自己的年龄, 此时就可以使用 SMPC 来计算。

2.1.4 数字签名

数字签名(Digital Signature)是区块链的重要密码学原语之一, 消息接收者使用它来验证发送者对消息的签名是否合法。数字签名具有完整性、认证和不可否认性三个重要性质。其中完整性可以通过比较接收消息的哈希值来判断传输过程中签名内容是否发生改变; 认证是指消息接收者可以用发送者的公钥对签名进行验证, 除了签名者, 任何人都无法伪造该签名; 不可否认是指因为只有签名者才知道签名私钥, 所以他不能否认自己的签名。数字签名通常包含以下三个算法:

1) $(pk, sk) \leftarrow \mathrm{Gen}(1^{\lambda})$: 密钥对生成算法 Gen 以安全参数λ作为输入, 输出用户的一对公私钥。其中私钥 sk 用来签名, 公钥 pk 用来让其他用户验证签名的合法性;

2) $\sigma \leftarrow \mathrm{Sign}(sk, m)$: 签名算法以用户私钥和消息 m 作为输入, 输出对消息 m 的签名σ, 一般是对消息 m 的哈希值 $H(m)$进行签名, 而不是直接对 m 签名;

3) $b \leftarrow \mathrm{Ver}(pk, \sigma)$: 验证算法 Ver 以用户公钥 pk 作为输入, 输出一个布尔值 $b \in \{0,1\}$, 若签名合法则输出 1, 反之为 0。

在可编辑区块链方案中, 除了使用普通的数字签名, 还会涉及与它相关的签名方法, 如多重签名[23]、聚合签名[24]、群签名[25]、环签名[26]等。下面分别作简要介绍: 多重签名(Multi-Signature)是指在 n 个用户中, 只有超过m个用户对同一消息进行签名, 该签名才是有效的。聚合签名(Aggregate Signature)是将多个签名聚合在一起, 压缩成一个签名, 这能够显著减少签名的传输量, 实现交易的批量验证, 提高效率。群签名(Group Signature)和环签名(Ring Signature)经常在涉及隐私保护的方案中使用。在群签名中, 任意一个群中的成员能以匿名的形式代表整个群体对指定消息进行签名。环签名和群签名的功能相似, 都是能够隐藏签名者的身份, 但是群签名中存在群管理员这个可信第三方, 能够在必要时揭示签名者的身份, 而在环签名中其他用户是不能识别出签名者的身份, 除非签名者自己揭示身份, 所以具有更好的匿名性。

2.1.5 基于属性的加密

合理分配和管理编辑权是可编辑区块链方案中重要的环节。为了减少冗余操作, 支持细粒度和可控的编辑, 越来越多的方案考虑使用基于属性的加密技术, 因为该原语能够实现数据拥有者自己设置数据的控制策略, 规定谁有权访问加密数据。该原语最先是由 Waters 提出的[27], 基于属性的加密(Attribute Based Encryption, ABE), 被认为是最具前景的支持细粒度访问的加密原语。不同于传统公钥加密方案, ABE 方案使密钥和密文与策略和属性集关联, 这样能够灵活设置对数据的访问策略, 降低开销, 便于加解密; 与基于身份的加密原语相比, 基于属性的原语无需用户在加密时提前知道接收者的身份信息。

ABE 主要分为两类: 基于密文策略的属性基加密(Ciphertext Policy-ABE, CP-ABE)和基于密钥策略的属性基加密(Key Policy-ABE, KP-ABE)。前者的密文和访问策略相关, 如果用户的属性满足该策略则可以成功解密密文; 后者的密文和属性关联, 如果密文关联的属性符合用户的密钥策略, 那么该用户可以解密密文。由于 KP-ABE 和 CP-ABE 相似, 所以这里我们主要介绍 CP-ABE, 其通常由概率多项式时间算法的元组(Setup_{cp},KeyGen_{cp},Enc_{cp},Dec_{cp})组成:

1) (mpk, msk)←$\mathrm{Setup}_{cp}(1^{\lambda})$: 以安全参数λ作为输入, 生成一个系统主私钥 msk 和公钥 mpk, 公钥 mpk 作为其他算法的隐式输入;

2) $sk_U \leftarrow \mathrm{KeyGen}_{cp}(msk, U)$: 以系统主私钥 msk和属性集 U作为算法输入, 输出和属性相关联的私钥sk_U;

3) $c \leftarrow \mathrm{Enc}_{cp}(m, A)$: 以明文 m 和访问结构 A 作为输入, 输出一个密文 c;

4) m 或 ⊥← $\mathrm{Dec}_{cp}(\mathrm{c}, sk_U)$: 以密文 c 和用户属性私钥sk_U作为输入, 如果用户属性 U 满足密文的访问结构 A, 则输出明文 m, 否则返回错误⊥。

2.2 可编辑方案中的常用技术

这节中我们主要介绍在可编辑区块链方案中使用的一些常用技术, 包括公平合理地选取委员会、激励和惩罚机制、智能合约等, 这些技术对可编辑区块链方案的安全、效率和公平公正等方面具有重要的推动作用。

2.2.1 委员会选取

在可编辑区块链方案中, 特别是基于变色龙哈希函数的方案里, 为了减少编辑权的中心化影响, 通常需要对陷门进行划分, 并在一群随机选取的委员会中分发这些陷门碎片, 所以如何合理公平的选取委员会, 而不会让敌手提前知道被选者的身份是一个重要问题。目前方案主要使用可验证的随机函数(Verifiable Random Function, VRF)[28], 这是一种可以生成公开可验证伪随机数的函数, 通常生成的随机数与节点相关联, 任何节点都可以通过 VRF 的验证函数来验证生成该随机数的节点是否具有成为委员会成员的资格。此外, 可验证的延迟函数(Verifiable Delay Functions, VDF)[29]是当前方案未使用但颇有前景的一个函数。粗略来说该函数的特点是计算函数结果需要一段时间, 即使利用并行工具也不能加速计算。

2.2.2 激励与惩罚机制

为了维护区块链的良好环境, 需要加入一些激励或惩罚机制来奖励诚实节点和惩罚作恶节点。目前区块链中的激励与惩罚主要是经济方面, 包括代币、权益、金钱奖励和金钱惩罚等。其中代币是现有区块链系统中最具吸引力的激励方法, 也是一种价值交换的媒介; 权益是通过赋予用户对某些事情的决策权力, 以带给这些用户一定的利益, 从而激励用户积极参与维护区块链系统; 奖励一般是给予用户一些金钱或其他形式的经济奖励, 惩罚是为了遏制在区块链上发生作恶事件的机制, 例如参与活动之前需要缴纳保证金、作恶发现后加入黑名单等。

在可编辑区块链方案中, 需要加入一些合理的激励或惩罚机制来维护编辑操作的正常执行。

2.2.3 智能合约

区块链中智能合约(Smart Contract)[30]是把真实世界合约电子化的一种协议, 可以理解为是提前编写的程序, 若发生了相关事件, 则会触发合约按照一定的流程执行操作。目前公链中以太坊是最主要的智能合约平台之一, 智能合约的出现把区块链的用途从单纯的数字货币交易过渡到可编程社会。一般智能合约编写好后被写入区块链中, 由于不可篡改性, 智能合约在上链后将无法被修改, 实现了可追溯和公开透明。但是在 DAO 事件发生后, 越来越多研究[4]发现正因为智能合约是人为编写的代码, 因此不可避免会出现漏洞。在传统区块链中, 若合约出现漏洞, 将无法及时修复, 这可能带给用户不可挽回的损失。如果允许在可控条件下对区块链进行编辑, 那么就能及时解决此类事件带来的不良影响。

2.2.4 快照与剪枝

快照(snapshot)是记录某时刻系统状态的一种技术, 通常在区块链中用于节点定期保存链上状态或记录指定区块高度的状态, 这样既能减小存储空间, 又能快速获得当前链上的状态数据, 加快节点验证数据和同步新链的过程。如 Sidecoin 就是对比特币进行快照的一种机制[31]。随着时间的推移, 链上存在越来越多无用的数据, 浪费大量的空间, 所以如何合理减少存储空间成为一个关注点。如 Palm 等人提出了一种在区块链中减少存储规模的剪枝(prune)策略[32], 该方法是基于一种剪枝函数, 能够让每位用户独立选择和删除任意交易。但是在该方法中, 如果区块链的所有节点都删除了某些交易, 那么可能导致某些数据丢失, 而且对于新加入的节点来说, 无法分辨链上数据是否被删除过。

剪枝技术是一种特殊的删除操作, 其目标侧重于减少区块链的存储负担, 通常是大规模删除历史数据, 而可编辑里的删除目标侧重于去除非法或不合适的历史数据, 通常是小规模的删除操作。目前的交易级剪枝方案较少, 大多是区块级的剪枝, 且该方法隐式假设剪去的数据和后续交易无关, 因此只适合安全性要求较弱的应用场景。

2.3 安全要求

安全要求主要包括用户输入数据的机密性、完整性和计算的正确性, 以及数据的可用性。输入数据的机密性是指除了数据输入方知道输入的具体内容, 其余用户对这些输入内容是不可见、不可知的, 如可以使用同态加密来实现[33]; 区块链中数据的完整性是指存储在链上的所有数据没有被更改, 一般用数据哈希值验证完整性; 计算的正确性是指涉及对数据的计算, 其结果一定能被证明是正确的, 可以使用合适的技术实现, 如零知识证明[34]; 数据可用性是指合法授权的用户能够访问到相应的数据, 不会出现访问阻塞, 数据不可用的现象, 如使用黑名单、定时检测等方法, 防止恶意用户实施泛洪攻击破坏数据的可用性。

另一个安全要求是对密钥的安全有效管理, 其中密钥主要分为对称和非对称两种, 前者表示用来加解密的密钥是相同的, 后者通常包括一对公私钥, 表示加解密的密钥是不同的。在已有的可编辑区块链方案中, 除了分别使用对称和非对称的密钥方案, 还有两者混合使用的方案。通常密钥的整个生命周期由生成密钥、分发密钥、存储密钥、备份或恢复密钥、更新密钥、使用和撤销密钥五个阶段组成。如何安全高效地管理密钥对可编辑区块链方案是非常重要的, 不同方案对密钥管理的要求也不同。例如在基于变色龙哈希的方案中, 陷门的管理是整个方案的关键。因为谁掌握了陷门相当于拥有区块链数据的编辑权, 所以若授予用户编辑权, 那么需要考虑当用户作恶时, 能够及时地撤销其编辑权限, 防止恶意用户滥用编辑权。若发生了密钥泄漏或是密钥过期, 则需要及时更新密钥。

2.4 安全性质和假设

区块链具有三个重要的安全性质[35]: 链质量、链增长和公共前缀。简单来说, 链质量是指在诚实方持有链的任何一段中, 敌手区块的占比不超过μ, 其中μ是敌手所控资源的份额比例; 链增长是指区块链总是以和时间成正比的速度拓展成更长的新链; 公共前缀在可编辑区块链方案中又叫做可编辑的公共前缀, 是指如果两个诚实节点在不同时刻持有两条链C、C', 那么其中一条较短链C移去最右边的x个区块后得到的链是另一条较长链的前缀, 记作$C^{\lfloor x} \prec C'$。

此外, 区块链还需满足两个性质: 活性和持久性。活性是指如果系统中所有诚实用户都试图将某笔交易纳入各自的账本, 那么在经过交易确认的时间后, 当所有用户诚实地进行查询和回复时, 都会指明交易是稳定的。如果某交易所在区块后面至少存在k个区块, 则说明该交易是稳定的, 其中k是一个系统参数, 表示稳定交易的区块数目。持久性是指如果某个用户声称某一交易成为稳定交易, 那么其余所有用户在账本里的相同位置都报告该交易, 或不把其他冲突交易报告成稳定交易。上述两个性质保证了在一段时间后, 系统里所有诚实用户都会拥

有一致的区块链状态视图, 由诚实用户发布的交易最终将会被包含在区块链内。

由于在实际应用中需要权衡安全和效率两个方面, 所以为了设计更加安全有效的方案, 需要指定区块链的安全假设, 例如许多可编辑方案安全假设是系统中诚实方占大多数。此外, 不同可编辑区块链方案使用的共识不同, 对系统敌手的容错率也不同。目前方案主要涉及的共识机制及其容错率如表 2 所示。

表 2　不同共识机制的容错率

Table 2　Fault tolerance of different consensus algorithms

共识机制	容错率(容忍敌手占比的上限)
Proof Of Work, POW	1/2
Proof Of Stake, POS	1/2
Practical Byzantine Fault Tolerance, PBFT	1/3

(注: 根据文献[36], 有些可编辑区块链方案涉及的 POW 容错率实际上是 1/3)

3　可编辑区块链方案的分类

本节主要根据图 3 可编辑区块链方案的总体流程, 结合图 2 的分类, 分别从编辑共识、编辑处理、编辑证明这三个重要环节对方案进行分类讨论。通常, 在编辑共识中主要涉及编辑所属权和共识技术, 编辑处理环节主要考虑编辑操作种类和编辑对象粒度, 编辑证明环节需要考虑编辑使用的数据结构以及链架构。

3.1　编辑共识

当区块链用户提出编辑请求后, 拥有编辑权的用户(编辑者)需要验证请求是否合法, 如果验证通过, 则需要在编辑者之间对是否同意该编辑进行共识。这是因为某个交易或区块被编辑以后, 区块链的状态就发生了改变, 为了让区块链状态在全部节点之间重新达成一致, 需要一种共识机制。编辑共识环节需要考虑的是编辑所属权和共识技术两个方面, 其中编辑所属权是指编辑权掌握在哪些用户的手中, 共识技术主要分为基于变色龙哈希函数、基于投票共识、基于可变交易以及其他方法。

3.1.1　编辑所属权

在编辑共识中, 由于编辑权赋予了不同的实体, 所以可以将方案分为单中心化、多中心化和去中心化的编辑共识, 三种分类如图 4 所示。

单中心化是指编辑权掌握在一个实体中, 可以是个人, 也可以是一个机构, 这种情况下共识不需

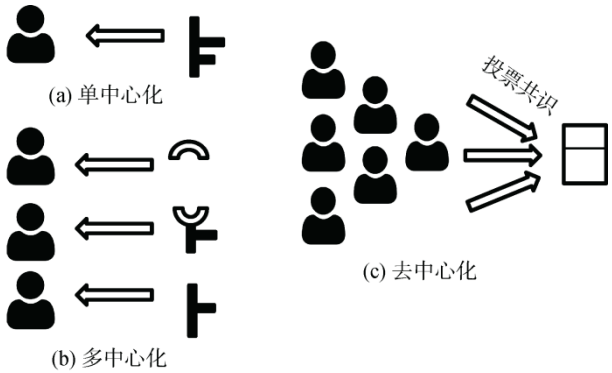

图 4　依据编辑所属权划分的方案类别

Figure 4　Scheme category based on redacting authority

在多方之间进行交互。如 Lee 等人提出的一种在无许可区块链中借助侧链构建单中心化的可编辑方案[37], 在该方案中只允许交易拥有者自己提出编辑操作, 其他用户没有编辑权限。其核心思想是使用方案定义的交易目标值, 而不是交易本身来计算区块的哈希值, 当修改交易时, 根据修改困难度, 交易拥有者把修改请求提交给指定侧链, 侧链上的矿工需要找一个前缀相等的哈希碰撞, 再次进行 POW 生成修改命令块来争夺交易的编辑权, 当侧链最终化时, 依据修改命令块在主链上执行相应的编辑操作, 其中困难度依赖于用户发送交易时的设置, 如果涉及金融资产转账的交易, 方案会限制用户只能把该交易设置为不可篡改的级别, 这样能够使修改交易的方式和传统公链中同意交易的方式保持一致, 最小化对传统链的修改, 适合用于公有链中的金融和云服务场景。由于编辑权是单中心化的, 只有交易拥有者自己才拥有编辑权, 如果拥有者上传了非法数据, 其他用户无法进行更改, 缺乏相应监管, 且方案修改了计算区块哈希值的方法, 不能做到向后兼容。

为了解决编辑权过度中心化的问题, 加强对区块链的监督, Jia 等人构造了一个带有半可信监管者的细粒度可编辑区块链[38], 这是第一个不仅支持监管区块链内容, 而且允许用户管理自己数据的区块链。该方案把交易分为两部分: 标准部分和任意部分。其中标准部分由涉及安全的字段组成, 只有半可信的监管机构可以修改它, 他的修改操作将接受全节点监督; 任意部分允许用户插入任意的数据, 因此数据拥有者可以对其进行修改, 并由监管者进行监督, 这种情况下的编辑共识也不需要多方交互。如果发现恶意用户, 作为半可信的监管机构可以撤销该用户编辑数据的权利。由于引入半可信监管方, 方案适用于联盟链和私有链的场景, 如在基于区块链的

医疗系统中, 链上记录了患者的病原信息, 必要时患者有权将这些敏感数据从交易的任意部分中删除。虽然方案具有可监督性, 但编辑权本质上还是偏向单中心化, 不支持恶意监管者的存在, 且效率较低。

多中心化是把编辑权赋予多个实体, 这种情况下的编辑共识需要在多方之间进行交互。通常编辑权是由交易发送者在指定策略中规定的, 策略一般包括交易可以由谁编辑、何时允许编辑、可以编辑哪些数据等。一般是把编辑密钥在多个实体之间进行秘密共享, 需要修改时再使用安全多方计算恢复密钥。如 2017 年 Ateniese 等人提出的可编辑区块链方案[7], 这是最早提出的一个基于变色龙哈希函数的可编辑区块链方案, 能够实现删除和修改区块的内容, 其区块链结构如图 5 所示。该方案总体思想是利用变色龙哈希函数 $\mathrm{CH}(hk,(s,x);r)$创建区块, 当需要修改块中的交易时, 持有陷门的实体可以找到一对碰撞使得 $\mathrm{CH}(hk,(s,x);r)=\mathrm{CH}(hk,(s,x');r')$, 其中 r 和 r'是区块结构里新增的随机数, s 是前一区块的哈希值, hk 是变色龙哈希函数的公钥, x 和x'分别是编辑前后的区块内容。方案把陷门持有权分为单中心化和多中心化, 分别适用于私有链和联盟链, 前者只有一个可信方拥有陷门, 后者利用 Shamir 秘密共享[21]把陷门分给不同的节点。当需要编辑区块链数据时, 各方对编辑对象达成共识, 并使用安全多方计算恢复陷门, 实现编辑操作。注意到在公有链中, 若把陷门分配给全节点, 则全节点的数量会影响整体性能; 若把陷门分配给指定节点, 则会存在指定目标攻击的危险, 所以该方案不适合用于大规模公有链场景。其次, 该方案的编辑比较粗粒度, 除了执行编辑操作的节点, 其他节点没有对内容进行验证, 包括区块创造者, 这会导致恶意编辑者能够任意修改区块内容。最后, 该方案只使用一个陷门, 一旦编辑者重构陷门, 他就可以多次使用陷门对不同区块数据进行编辑。

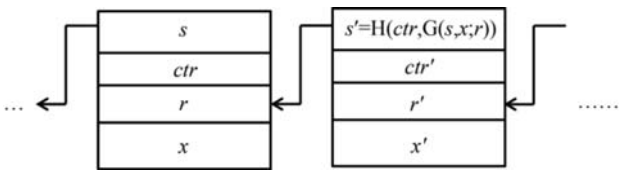

图 5 Ateniese 方案的区块链结构

Figure 5 Blockchain structure of Ateniese's scheme

去中心化一般指所有实体都具有编辑权, 但还需通过某种机制选出特殊团体, 利用投票方式对编辑请求达成共识并修改相应数据, 这样才能被认为是合法的编辑操作。Deuber 等人提出在无许可区块链中基于矿工投票的可编辑区块链方案[8], 该方案具有公开可验证和可问责性, 且没有依赖繁重的密码学工具或额外的信任假设, 其核心是在区块头中增加一个区块的旧状态, 即旧区块的默克尔根哈希值。具体来说, 方案的编辑共识思想是: 当网络中的矿工收到包含候选块的编辑请求时, 首先使用旧状态验证候选块, 然后验证候选块是否包含前一个区块的正确哈希值、是否已解决了工作量证明问题、是否会使链中的下一个区块失效。如果通过上述验证, 则认为候选块是有效的, 那么在一定的投票期内, 矿工可以把合法请求的哈希值包含在他们挖掘的下一个区块中, 作为同意此次编辑的投票。在投票期结束后, 网络中的每个节点都可以根据指定策略验证编辑请求是否获得批准, 比如通过检查收到的投票数是否超过阈值。若被批准, 则说明在去中心化的环境下对该编辑请求已经达成共识, 之后就可以用候选块替换原始块来执行编辑操作。该方案适合应用于公有链场景, 如合理移除比特币现存的恶意数据。然而方案属于粗粒度的块级修改, 只能改变 coinbase 和 OP_RETURN 字段里的非法数据部分, 而且方案的投票期过长, 效率和可扩展性较差, 不能实现及时修改。

3.1.2 基于变色龙哈希函数

除了上述的 Ateniese 方案[7]使用基于变色龙哈希函数的共识技术, Ashritha 等人[39]在 Ateniese 方案思想的基础上, 提出使用增强的变色龙哈希函数来修改区块内容。简单来说就是在划分陷门时使用了非线性的秘密共享方法, 当对区块内容进行编辑时, 编辑者除了需要从陷门碎片持有者那里获取重构的陷门, 还需要从待编辑区块的创建者那里获得临时陷门, 只有同时拥有两个陷门才能对区块进行编辑。此外, 方案的密钥恢复机制使用哈希时间锁技术, 使得临时密钥能够在一段时间后被公开, 可用于解决持有临时密钥的区块创建者或陷门碎片持有者离线而导致区块不可编辑的问题, 适用于许可链中节点灵活离线的场景。然而该方案依然是区块级编辑, 虽然可以使用秘密共享减小持有陷门实体的中心化影响, 但是由于秘密共享和安全多方计算需要高昂的通信开销, 所以不适合用于大规模的公有链场景。另外, 即使区块内容被编辑过, 也不知道编辑者的身份和编辑时间, 无法对编辑操作进行追责。

之后, 为了增强对编辑操作的访问控制, 许多方案使用了基于属性的加密原语。但在这些方案中, 其他交易的编辑者也可能满足嵌入在本交易中的访

问策略, 陷门持有者可能会滥用重写权限, 且恶意重写不会被标识, 无法进行问责。为了实现合理的问责功能, Tian 等人提出了一个基于变色龙哈希函数和 CP-ABE 的可编辑区块链方案[40], 同时实现了问责和匿名功能。其中利用黑盒来实现问责是该方案的一大亮点, 当产生争议时, 一个权威实体能把修改的交易和修改者链接起来, 就像群签名里的管理者一样。此外, Tian 等人还提出一个在无许可环境下实现公开问责和安全重写的区块链通用架构[9]。该架构使用了多个用户组成的可验证动态委员会, 周期性地刷新密钥, 以多中心化的方式进行编辑共识。此外, 编辑者的身份通过数据签名来识别, 通过承诺方案将编辑者的公钥与动态委员会进行链接, 以确定编辑者是从委员会那里获得的编辑权, 使得方案不仅能识别修改交易的用户是谁, 还能判断该用户是否具有交易修改权。该方案适用于需要问责功能的可编辑公有链场景, 如在供应链或物流链的运输管理中, 可以对数据进行纠错和追踪, 一旦出现问题, 各方能够根据指定策略对编辑者进行问责, 使各方损失降到最低。但是, 该方案引入了一个委员会, 需要额外考虑选取委员会成员的公平性和随机性。

从另一方面来看, 密钥的合理分配与管理对于变色龙哈希函数和可编辑区块链都是至关重要的。密钥泄漏问题在被 Ateniese 等人解决之后[41], 出现了许多双陷门无密钥泄漏的变色龙哈希方案, 这些方案包含长期密钥和一次性密钥两个陷门, 而哈希碰撞最多只会泄露一次性密钥, 这就是由 Camenisch 等人提出的临时陷门[20]。在之前的方案中, 没有基于抗量子假设的方案。为了解决该问题, Wu 等人提出了基于有格陷门和无格陷门的两个无密钥泄漏的变色龙哈希函数, 给出了它们在可编辑区块链中的应用[42], 并提出两种机制以防止误用区块链的可编辑功能: 一个是基于投票共识, 提供公共问责性; 另一个是基于门限秘密共享, 提供不可区分的修改功能, 而且该方案也可用来构造抗量子的变色龙签名。不同于 Ateniese 的构造, 该方案不需要把非对称加密和非交互零知识证明等复杂密码学工具进行结合, 但是需要对区块链进行一定的更改。

Huang 等人针对可编辑区块链提出了两个新的密码学方案[43]: 时间可更新的变色龙哈希(TUCH)和可链接可编辑的环签名(LRRS), 分别用于修改数据和在匿名签名中防止双花攻击。将 TUCH 和 LRRS 分别作为哈希函数和数字签名可以构建一个可扩展、可编辑的区块链。不过 LRRS 是以效率换取匿名性, 且区块的修改权是完全掌握在打包该区块的矿工手上, 这样不仅需要矿工额外存储区块对应的陷门, 而且修改权过于集中可能导致单点故障问题。Panwar 等人针对重写区块链缺乏匿名性、效率低以及缺少陷门撤销机制的问题, 提出了一个有效重写许可区块链的框架 ReTRACe[44], 框架包括一个带临时陷门的可撤销变色龙哈希方案、一个可撤销的快速属性基加密方案和一个动态群签名方案。利用前两个方案, 授权用户可通过临时陷门来编辑交易, 根据实际应用需求能撤销对临时陷门的访问。利用动态群签名方案, 授权用户能够匿名发布和编辑交易, 必要时管理员能够揭露编辑者的身份。但由于群签名中存在群管理员, 会引入第三方实体, 且框架修改了区块链的结构, 不能兼容现有结构。

之前所述方案主要存在如下两个问题: 一旦用户被授予编辑权, 那么他可能会多次对区块链内容进行任意的修改, 且系统对恶意行为没有合适的奖惩机制; 此外, 基于属性的变色龙哈希方案可能存在共谋攻击。为了解决这些问题, Xu 等人提出了一个带有金钱惩罚机制的有限修改次数和基于 *epoch* 的可编辑区块链(KERB)[45]。粗略来说, 为了减少恶意行为, 由权威机构 CA 规定编辑者最多只能修改交易 k 次, 并且每个编辑者都需要在链上放置时间锁保证金, 如果在规定时间内编辑者作恶, 那么 CA 可以取走保证金作为惩罚, 否则时间到后可以由编辑者自己取走。该方案的安全模型考虑了交易拥有者和编辑者共谋的情况, 且没有使用特别复杂的密码学原语, 而是使用了数字签名和普通的变色龙哈希函数, 比较适用于许可链的场景, 如某些应用会根据用户充值的金额设置用户对数据的编辑次数或访问次数上限, 如果超过阈值, 则会扣除用户的账户余额。但是, 在该方案中, 如果交易拥有者设置了只允许他自己才能对数据进行编辑的访问策略, 那么这将不利于数据的监管。

通常来说, 陷门管理问题是基于变色龙哈希函数方案的核心。在现有方案中, 区块级的编辑方案一般需要修改区块头部的数据结构, 与传统区块链不兼容; 而交易级的编辑方案一般不用修改区块结构, 能够实现后向兼容。

3.1.3 基于投票共识

目前基于投票的可编辑区块链方案主要涉及工作量证明(Proof Of Work, POW)、权益证明(Proof Of Stake, POS)、空间证明(Proof Of Space, POSpace)等共识机制, 这些方案一般不会涉及过多复杂的密码学原语, 相比基于变色龙哈希函数的方案要更支持插入、删除、修改的动态操作和动态节点的加入离开,

以及更宽松的密钥管理。通常基于投票的可编辑方案总体流程如图 6 所示:

在 POW 共识机制中, 节点能否获得区块链的记账权与其拥有的算力大小有关, 通常节点需要解决一个 POW 的困难问题, 若找到一个合适的随机数, 则向全网广播自己获得了一个新区块的记账权。基于 POW 共识的可编辑区块链方案较多, 但由于部署在公链上, 容易暴露用户的交易内容和身份, 以及编辑者的身份。对此 Ren 等人改进了 Deuber 的方案[8], 提出了一个适用于物联网隐私保护的可编辑区块链方案[46]。方案的主要思想是: 数据拥有者把交易数据用对称密钥加密, 以此来保护交易数据, 只有数据拥有者才能对交易进行编辑。使用环签名来保证用户的身份隐私, 如果用户提出编辑请求, 则矿工在投票期内对请求进行投票, 当票数超过门限环签名的阈值 t 时, 让这 t 个矿工形成环, 进行门限环签名。为了在匿名环境下证明编辑交易的合法性, 发送者需要揭示过期的用户身份和交易数据。Ren 的方案相比于 Deuber 的方案增加了隐私保护功能, 且实现了交易级的编辑, 适用于公有链场景, 尤其是涉及物联网设备的商业应用, 使用该方案能够在编辑数据的过程中, 不会将编辑者的个人信息或数据拥有者持有的秘密泄露给竞争对手。不过只有交易发送者才拥有对交易的编辑权, 缺乏对数据的监管以及相应的问责机制。

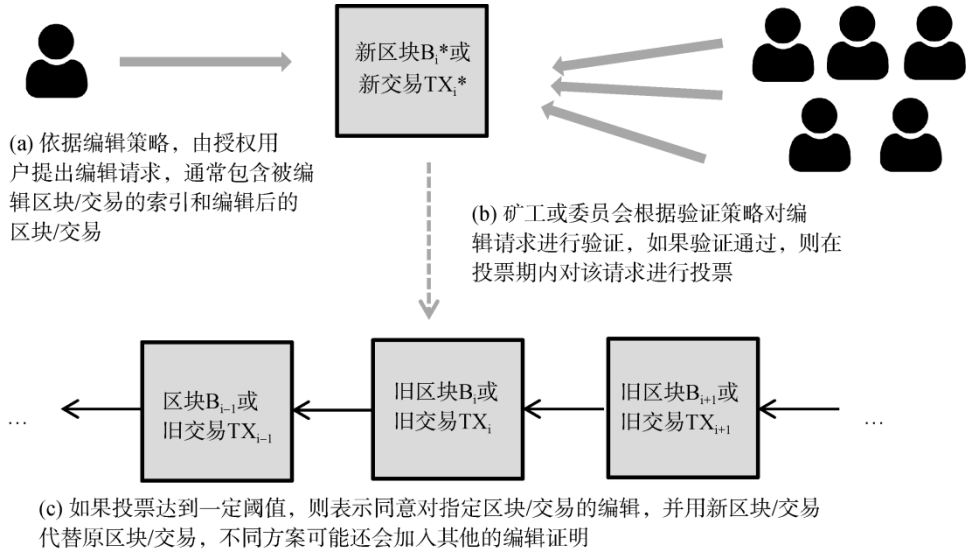

图 6　基于投票的可编辑区块链方案总体流程

Figure 6　Overall process of voting based redactable scheme

POS 是一种权益证明的共识机制, 记账权与持币量和持币天数有关, 拥有的权益越多, 获得记账的机会就越大。与 POW 这种需要较高算力、消耗资源的共识机制相比, POS 共识机制不需要耗费大量的电力和能源, 且基本上对计算机硬件没有过高要求, 适合运用在更多场景中。为了解决 Deuber 方案的编辑时效性问题, Xu 等人基于该方案相继提出了运用在 POW 和 POS 公有链及时可编辑的通用方法[47]。方案的编辑共识是基于 POW 和 POS 的委员会投票机制, 能够加快编辑处理的速度。具体来说: 如果有编辑请求提出, 那么把它加入编辑池, 并将投票期划分成几个 slot, 这是一种数量连续的离散时间单位。每个 slot 有一个领导者, 如果编辑池为空, 则领导者像传统区块链一样提出新块, 否则在投票期内需要收集委员会成员对编辑请求中候选块的投票, 然后由领导者在该 slot 使用聚合签名生成证明, 打包成块, 广播上链。方案利用算力或股权来随机选取委员会, 具有公开可验证性和问责性。不足的是该方案依旧是区块级的编辑操作, 且没有设置对委员会投票的激励机制, 不利于维护区块链系统的稳定。

另一种共识机制是 POSpace, 该机制是由证明者发送给验证者一小块数据, 该数据用于确认证明者拥有一定量的存储空间, 数据占用的空间越大, 说明用户的付出越大。POSpace 是使用物理硬盘空间作为付出的证明, 这种共识机制也解决了 POW 浪费大量资源的问题, 但和其他共识机制不一样的是: 在基于 POSpace 的可编辑区块链方案中, 每个区块被划分为三个子块, 分别是证明子块、签名子块和交易子块。其中需要维持的是签名链的完整, 而不是哈希链的完整。POSpace 区块链结构如图 7 所示。

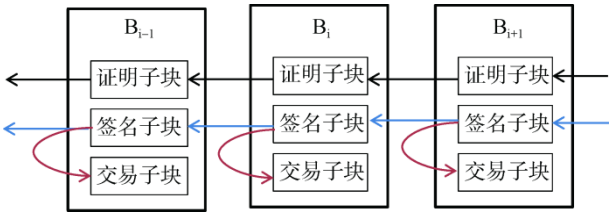

图 7　基于 POSpace 的区块链结构

Figure 7　Blockchain structure based on POSpace

以下几个方案都适用于基于 POSpace 结构的公有链场景: Ren 等人使用改进的门限环签名提出了一种数据可删除方案[49]。当需要删除某些区块数据时, 相关节点提出删除请求, 其他节点广播删除意见, 只有超过一定阈值的用户同意该请求, 且生成有效的门限环签名才能对数据进行删除。然而在该方案中, 用户的身份和交易内容都是公开可见的, 这些数据可能会泄露用户个人隐私。为了保护用户的隐私安全, Li 等人结合无证书的聚合签名[50], 设计了一种匿名的可编辑区块链方案[51]。其中无证书签名是指密钥生成中心只会生成用户的部分私钥, 接下来用部分私钥和用户自己持有的秘密信息结合得到最终用户私钥。在该方案中, 首先由用户自己生成一对公私钥, 并将其真实身份发送给注册中心, 注册中心返回给用户一个伪身份和部分私钥, 用户使用伪身份能够在系统中实现匿名功能, 之后在签名时需同时使用私钥和部分私钥。方案改进了 POSpace 区块链结构, 在签名子块中加入了一个对交易签名的副本。若存在编辑数据的需求, 则只有在同意编辑的用户数量达到阈值后, 触发相应的智能合约, 对新数据进行签名并取代原副本, 链上的数据才能被修改或删除。Cai 等人也提出了一种基于 POSpace 隐私保护的可删除区块链[52]。方案的编辑共识主要是基于投票机制, 使用文章提出的可链接多重签名和群签名方案, 允许多个用户使用一次性地址生成有效签名, 隐蔽交易内容和双方地址, 以实现匿名功能。同时, 因为使用了可追踪的环签名, 如果两个子签名是由同一用户生成的, 则可以将这两个子签名链接在一起。根据不同的数据删除原因, 分别使用可追踪的环签名或 Pedersen 承诺方案来揭示用户的真实身份或交易内容, 使得方案具有公开可验证性和问责性, 且不依赖任何可信方。然而, 方案的删除操作是一次性的, 在删除之后只能验证删除的合法性, 但不能验证删除数据的完整性, 而且没有考虑到被删数据可能导致后续交易不合法的情况, 也没有相应的惩罚或激励机制。

3.1.4　基于可变交易

在传统区块链中, 一旦智能合约出现漏洞, 为了挽回损失只能进行硬分叉, 这会降低区块链的受信任度以及可用性。在这种情况下, 除了利用主流的基于变色龙哈希函数和投票的编辑共识机制, 还可以利用可变交易的方法修复合约漏洞。可变交易(Mutable Transactions)是由 Puddu 等人提出的概念[11]。所有可变交易都被指定了一个时间窗口, 在时间窗口内的交易是可变或可拓展的, 超过时间窗口的交易就是不可变的。总的来说, 方案的总体思想是: 交易发送方可以选择发送的交易是可变或是不可变的, 如果是可变的(mutable), 那么交易 T 包含一个交易集 Γ。其中, Γ由所有可能的交易版本$\{\tau 1, \tau 2, ..., \tau k\}$组成, 但只有一个交易版本$\tau a$ 是活跃状态, 其余都是对用户不可见的非活跃状态, 交易集在一段时间Δt 内可以由策略 P 授权的修改者 M 进行拓展, 增加新交易版本, 也可以通过发布一个特殊的元交易$T' = (ref_T, a')$来拓展或修改区块链历史, 其中ref_T指示要修改的交易是 T, a'是指把活跃的可变交易变更为$\tau a'$。每个可变交易集包含一个空交易 *nope*, 它不会修改区块链的状态, 当把 *nope* 状态设置为活跃时, 可以将一个可变交易从区块链中移除。Puddu 的可变交易方案如图 8 所示。

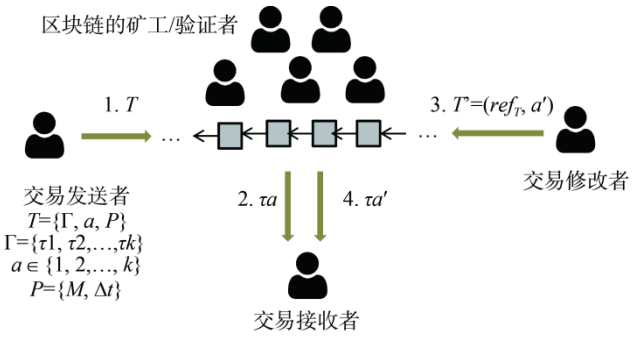

图 8　Puddu 的可变交易方案

Figure 8　Puddu's mutable transaction scheme

此外, 该方案还讨论了因为更改交易导致相关账户余额为负的问题, 并给出了一种解决方法: 把普通交易余额划分为可变余额(mutable balance)和不可变余额(immutable balance)两部分, 其中使用不可变交易只能花费不可变余额, 使用可变交易能花费两种类型, 但回退交易余额只能从可变交易余额中进行增减。该方案不仅适用于联盟链, 也适用于公有链场景, 如在公共存证服务中, 能够支持用户在知识产权等领域进行数据的存储、查找和编辑操作。虽然该方案提供了一个新颖的可编辑方法, 但是该

方案中的时间窗口不易设置, 编辑也是比较粗粒度的, 且没有做到真正在区块链上删除该交易, 只是让可变交易对用户不可见, 该交易的所有版本依旧存储在区块链上, 这会带来存储问题以及密钥管理问题。

3.1.5 其他方案

与公有链相比, 联盟链只允许指定用户或机构参与, 且数量有限, 因此共识效率较高, 适用于联盟机构或商企之间, 特别是金融行业。目前越来越多的组织根据业务需求搭建了不同的联盟链平台, 同时也提出了许多和商企相关的联盟链项目。然而在传统联盟链中, 平台无法更改出现错误的数据, 使得基于联盟链的项目变得不切实际。在权衡基于变色龙哈希函数和投票共识各自的利弊后, Li 等人结合两者提出了基于联盟链的可编辑方案[53], 使得平台可以在数据出现错误时, 能依据指定策略对历史数据进行编辑。方案中的每个人都能获取变色龙陷门和拥有修改权, 但是一次成功的编辑操作还需要进行投票共识。为减少多方交互, 采用随机选出一个用户对数据进行修改的方式, 每个用户都拥有参与投票的权利和被选为修改者的机会。方案的共识思想是: 如果联盟链的某位用户有编辑请求, 那么该用户把请求广播给其他用户, 其他用户对其进行投票, 若票数超过一半, 则表示同意该请求, 然后根据分布式随机数生成协议随机选择一位用户对历史区块的数据进行修改。虽然方案减轻了变色龙哈希函数陷门管理的负担, 但因为通过随机数来选取由谁负责修改区块, 方案的安全性转移到随机数的选取上, 所以随机数不能被敌手提前知道, 否则可能存在共谋、贿赂等攻击。此外方案只是块级修改, 如果系统人数多, 通信代价较大。

在许多商业场景下, Execute-Order-Validate 相比于其他架构更适合应用在实际中, 因为该模式具有较好的拓展性, 能够增加交易吞吐量。为了同时满足商业需求和隐私策略, Manevich 提出了第一个 Execute-Order-Validate 模式的可编辑区块链[54]。该模式共有三步: 先执行, 得到的结果发送给排序节点, 排序节点产生以块为单位的有序链码输出序列, 这些块被广播到验证节点, 然后依据最新的链状态对其进行验证, 验证通过后会提交这些输出, 并更改相应的本地状态。因为和普通交易一样, 编辑交易需使用原子广播来达成编辑共识, 因此删除数据时用零值代替。该方案的共识机制和 Deuber 方案[8]不一样: 后者关注的是 order-execute 模式的区块链, 不可移除能被花费的交易数据, 而 Manevich 的方案可以做到, 但此方案验证时即使只有一个交易无效, 也会导致整个块验证失败。考虑到许多可编辑区块链方案构造都很复杂, Grigoriev 等人基于 RSA 加密方案[55], 提出一个共识简单的可编辑区块链的方案[56]。该方案把区块分为前缀、内容和后缀三部分, 在需要编辑区块数据时, 重新计算区块的哈希值, 并依据 RSA 难题计算后缀值。其中, 区块链的不可篡改性是基于 RSA 难题, 而其他方案是基于底层哈希函数, 虽然方案简单, 但该方案只适用于私有链, 且 Dousti 的文章[57]指出该构造是不安全的。

除了交易的编辑技术, 还有交易的回退机制也属于区块链领域的一种恢复机制。前者通常用于插入、删除和修改区块链上不合适的数据, 这是可编辑区块链的目标; 而后者通常用于私钥泄漏时挽回资金被盗的损失, 其目标是取消某个操作或将区块链状态更改到上一个时间点。在解决交易回退的问题上, 许多工作都依赖交易尚未提交或确认时进行替换或取消交易, 这可能直接破坏区块链的基本属性, 所以针对区块链的恢复机制, Martins 等人在 EVM 上实现第一个用于恢复以太坊 token 的解决方案[58]。在方案中, 由提议者提交恢复请求, 使用区块链的争议解决机制决定能否执行恢复请求, 如果同意该请求, 则执行恢复操作。其恢复操作的共识不依赖于投票机制或密码学方法, 也不需要修改底层区块链协议, 而是使用智能合约, 让钱包拥有者恢复由于攻击或意外造成的数字资产损失, 同时保证区块链的基本属性不被破坏。除了以太坊, 该方案同样适用于以太坊经典、Cardano 等运行 EVM 的公有链, 以及 Quorum、Hyperledger Besu 等运行 EVM 的私有链。

总的来说, 目前的编辑共识方案可以分为基于变色龙哈希函数和非变色龙哈希函数, 其中后者包括投票、可变交易以及其他共识方法。近几年, 由于结合了许多复杂的密码学原语来改善区块链的可编辑功能, 使用变色龙哈希的方案越来越朝向复杂化方向发展, 所以基于变色龙哈希函数的方案一般更适合用于小规模的许可区块链, 即联盟链或私有链; 基于投票共识的方案一般是矿工或指定团体对编辑请求进行投票, 如果票数达到一定阈值即同意对区块链数据进行编辑, 这种方案比较适合无许可区块链, 即公有链。本文对基于变色龙哈希的方案和基于非变色龙哈希的方案总结如表 3 和表 4 所示。

3.2 编辑处理

各方对编辑请求达成共识后, 进入编辑处理环节, 编辑者将依据策略和请求内容对链上数据进行相应的处理。其中主要考虑编辑操作种类和编辑对象粒度两个方面。

表 3 基于变色龙哈希方案的总结

Table 3 Summary of chameleon hash based schemes

方案提出年份、名称		是否基于属性	是否有问责性	架构(单/多)	交易/块级	是否实现	是否有匿名性
2017 年	Ateniese[7]	否	是	单链	块级	是	否
2019 年	Derler[63]	是	否	单链	交易级	是	否
2020 年	Tian[40]	是	是	单链	交易级	是	是
	Tian[9]	是	是	单链	交易级	是	否
	Jia[38]	否	是	单链	交易级	是	否
	Wu[42]	否	是	单链	交易级	否	否
	Huang[43]	否	是	单链	块级	是	是
2021 年	Panwar[44]	是	是	单链	交易级	是	是
	Xu[45]	否	是	单链	交易级	是	否
	Niya[65]	否	是	单链	交易级	否	否
	Hou[66]	是	否	单链	交易级	是	否
	Zhang[73]	否	是	多链	块级	是	否

表 4 基于非变色龙哈希方案的总结

Table 4 Summary of non-chameleon hash based schemes

方案提出年份、名称		是否有问责性	架构(单/多)	交易级/块级	是否实现	是否有匿名性	是否向后兼容	适用于许可链或无许可链
2017 年	Puddu[11]	是	单链	交易级	是	否	否	都可以
2018 年	Li[53]	---	单链	块级	是	---	否	许可
2019 年	Deuber[8]	是	单链	块级	是	否	否	无许可
	Dorri[12]	---	单链	块级	是	---	否	都可以
	Lee[37]	是	多链	交易级	否	否	否	无许可
	Xu[47]	是	单链	块级	是	否	否	无许可
	Ren[49]	是	单链	块级	是	是	否	无许可
	Cai[52]	是	单链	交易级	是	是	否	无许可
2020 年	Florian[60]	---	单链	块级	是	---	是	许可
	Marsalek[72]	是	多链	块级	是	是	否	无许可
	Martins[58]	---	单链	交易级	是	否	是	都可以
	Thyagarajan[59]	是	多链	交易级	是	否	是	都可以
	Kuperberg[67]	---	多链	交易级	否	---	否	无许可
	Matzutt[68]	---	单链	块级	是	---	是	无许可
	Hillmann[70]	---	单链	交易级	是	---	否	都可以
	Chan[75]	---	单链	交易级	是	---	否	许可
2021 年	Ren[46]	是	单链	交易级	是	是	否	无许可
	Xu[48]	是	单链	块级	是	否	否	无许可
	Li[51]	是	单链	块级	是	是	否	许可
	Manevich[54]	是	单链	块级	是	否	否	许可
	Dousti[57]	---	单链	块级	否	否	否	许可
	Matzutt[69]	---	单链	块级	是	---	是	无许可
	Daniel[71]	---	单链	块级	否	否	否	都可以

(注: “---”表示没有说明)

3.2.1 编辑操作种类

通常编辑操作主要分为插入、删除和修改, 这些操作基本都会引起区块链状态的变更, 三种操作如图 9 所示。其中插入操作是指在区块链的历史数据中插入交易或区块, 通常用于数据在某个时刻因网络时延或其他原因没有被插入到区块链里, 这时可以使用插入操作对区块链进行编辑, 不过目前只有极少文章涉及插入操作, 大部分方案主要实现的是

修改和删除操作。

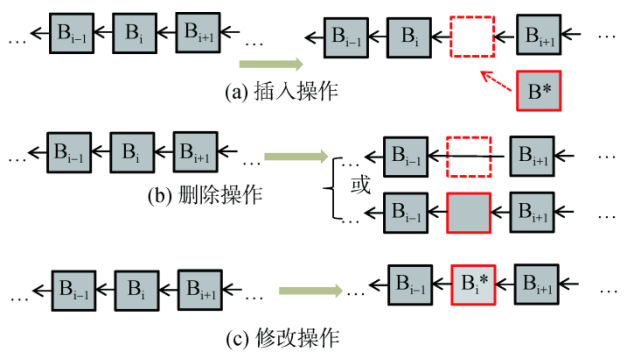

图 9　三种区块链编辑操作

Figure 9　Three blockchain redactable operations

(注: 存在两种删除操作, 一种是直接删除块, 而另一种是保留块结构, 只是删除内容)

在修改操作方面, 许多方案都是构建了一条新链, 无法与比特币、以太坊等这类现有区块链兼容, 因此 Thyagarajan 等人提出了一种通用的可编辑协议 Reparo[59], 它作为一个公开可验证层 (publicly verifiable layer), 可以在许可或无许可的区块链环境下高效运行, 能够实现及时修复有漏洞的智能合约或从链中移除非法内容。即使编辑对象有许多关联交易, 方案能够以较低开销修复以太坊的 DAO 漏洞。该协议和 Deuber 的方案[8]思想相似, 但不同之处在于, 该协议不是把旧状态存储在区块头里, 而是把旧交易的哈希值或完整的旧交易集合与旧状态一起存储在单独的一层数据结构中, 并提供相应的接口。和其他方案相比, 该方案具有公共可验证性和问责性, 且不会修改区块的数据结构, 能与比特币和以太坊等现有运行的区块链进行集成。但是协议缺少适当的访问控制策略, 且在必要时还需级联修改受影响的后续交易。

在删除操作方面, 为了减少区块链用户的存储空间和避免承担携带非法数据的责任, Florian 等人提出了允许全节点在本地擦除不合适数据的一个实用方法[60], 这与考虑全局擦除数据的想法相反。方法的主要思想是标记为擦除的数据块从本地存储器中物理擦除或进行混淆操作, 不再以可重构形式存储, 但节点能够分辨链中已擦除的数据和根本不存在的数据。对已擦除数据的引用需要存储在擦除数据库中, 以应对后续需要使用被擦除数据的情况, 在重新接收时对其进行过滤。该方法能够有效减少本地存储数据的负荷, 适用于轻节点较多的私有链或联盟链场景。不过方案只能更改 UTXO 的脚本和擦除非见证部分, 不能运用在挖矿节点上, 且后续的验证过程可能需要可信第三方的参与。

在插入操作方面, Dousti 等人引入第三种编辑方式: 块插入[57]。方案构造了一个三元组来标识每个区块, 若编辑了区块, 那么元组中需包含一个合法的数字签名。由于在插入块后可能会导致区块链变得不合法, 因而带来一些攻击, 为了使区块链能支持插入操作, 方案提出了一种模型并构造一个区块链来证明该模型是安全的。同时, 他们还提出了针对现有可编辑区块链方案的 4 个攻击[61]。文献[62]调查了一些解决区块链中不合适内容被插入的方法, 发现仅依靠过滤方案防止非法数据进入区块链是不够严谨的。因此, 文章提出了概念性对策: 通过缴纳适当的修改数据费用和减少交易的互操作性, 能够有效减少有害数据插入区块链以及增强数据的可用性。

3.2.2　编辑对象粒度

在处理编辑时, 根据编辑对象的粒度可以分为块级和交易级。块级的编辑方案是以区块为最小编辑单位, 即使只对区块中某笔交易进行编辑也需要使用新的区块内容进行整体代换, 属于粗粒度的编辑方式, 例如 Ateniese 方案[7]的编辑对象就是区块级。交易级的编辑方案是以交易为最小编辑单位, 通常只需更改指定交易, 属于细粒度的编辑方式, 例如 Xu 等人的方案[45]。两种编辑粒度的对比如图 10 所示。

除了上文讨论的部分方案, 还有许多方案都是基于区块级的编辑, 例如在物联网等大规模网络中, Dorri 等人通过改变计算块哈希的方式, 提出了一种内存优化和灵活的通用区块链 MOF-BC[12], 它依赖于中心实体实现在大规模网络中移除或汇总指定时间内链上的一些交易, 适用于私有链和联盟链场景, 可以在现有或将来的任何区块链上实现, 使物联网用户和服务提供商能够及时删除或汇总他们的交易。为了在可编辑区块链上支持细粒度的访问控制, Derler 等人提出了基于策略的变色龙哈希(PCH)概念, 并给出了 PCH 的一个通用构造[63]。PCH 主要由基于密文策略的属性基加密(CP-ABE)和变色龙哈希函数组成, 在计算默克尔树根的时候才使用 PCH。Derler 方案的核心思想是交易拥有者可以设置编辑交易的策略, 每个用户都可以获得与其属性相关的密钥, 当用户属性满足指定策略时就可以获得交易对应的临时陷门, 只有同时使用长期陷门和临时陷门, 才能对交易进行编辑。然而 ABE 方案在密文空间和计算陷门复杂度方面引起了较大的开销, 且用户一旦获取临时陷门, 就可以多次进行编辑, 没有陷门撤销机制。针对如何把陷门进行合理分发的问题, Sartori 提出了一个允许在特殊情况下对交易进行编辑的区块链架构[64], 其主要使用了带有临时陷门的变

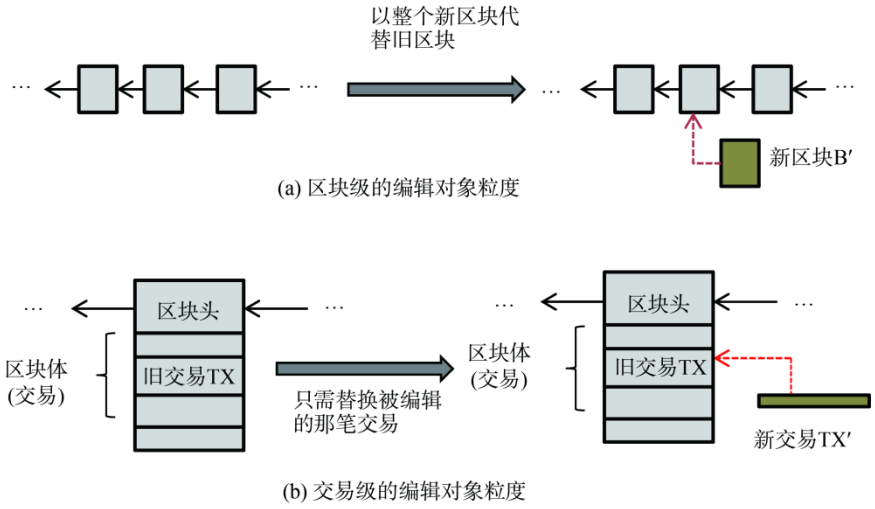

图 10　两种编辑粒度的对比

Figure 10　Comparison of two redactable granularity

色龙哈希函数，且陷门是按照权重来分配的，使得数据拥有者能获得最大的陷门份额。

为了使物联网场景下的区块链符合 GDPR 规范，同时支持对区块链中交易级的数据字段进行细粒度的更新和擦除操作，Niya 等人提出一种新颖的解决方案[65]。与之前方案不同的是，修改操作会记录在区块链中，实现操作的可追溯，而且陷门是由数据拥有者自己持有，用户可以请求修改个人的物联网数据并能立即执行，不过正因为只有用户自己才具有修改权，所以存在恶意用户任意修改数据的问题，这会给数据监管带来困难。最近 Hou 等人提出了一种同时实现细粒度和可控修改的区块链[66]。利用基于策略的变色龙哈希函数，交易发布者能够指定编辑者和编辑范围，当有修改交易的需求时，收集足够的矿工投票才能对交易进行编辑，此外，提供包含有害数据的块索引的用户可以获得一定奖励，如果在一段时间内没有发放奖励，则把恶意用户加入黑名单，以此实现强制去除恶意数据。

3.3　编辑证明

处理完被编辑的数据后，需要在链上加入相应的编辑证明，使得用户能够区分未编辑和已编辑的部分，在这个环节中除了需要考虑编辑使用的数据结构，还需要考虑编辑涉及的区块链架构。

3.3.1　编辑使用的数据结构

在编辑证明环节需要特别考虑使用的数据结构，因为选择一个合适的数据结构，不仅能维护区块链的完整性，提供状态改变的证明，还可以提高编辑效率。目前的编辑方案中使用了快照、检查点(checkpoint)、跳跃表、总结块等数据结构，它们共同的特点是存储了区块链不同时刻的状态和证明该状态正确的数据，相当于在某时刻之前的状态都是经过验证，是合法有效的。其中快照、检查点和总结块都能够加快验证或减少存储空间，例如在 Kuperberg 等人提出的新架构中[67]，把剪枝分为前缀和后缀两种方法。其中支持前缀剪枝的一个方法就是定期创建检查点，不过因为前缀修剪一般运用在链式结构里，这样被删除的交易可能分布在不同区块，需要对多个区块同时进行修剪。

面对区块链的拓展性问题，如有限的交易吞吐量、高时延和不断增长的链大小等，如今的区块链意在设计对旧区块进行剪枝或使用快照技术，而像简单支付验证(SPV)这种轻量级应用，虽然允许用户在本地删除一些旧交易来减少存储负荷，但后续验证还需依靠全节点，这样导致一些非法数据仍然存储在其他节点那里。针对上述问题，Matzutt 等人提出了适用于比特币的剪枝方案 CoinPrune[68]，其中的自举(bootstrapping)过程是通过定期从比特币 UTXO 集合中创建快照来连接节点，支持 CoinPrune 的矿工在区块链上相互重申(mutually reaffirm)快照的正确性来建立信任。在执行同步操作后，普通节点可以删除快照前的区块，只需保留快照、区块头和快照之后的全部区块，为了能够查询和验证旧数据，完整的区块链副本将由特定的存档节点保留。另外，Hillmann 等人提出了定期创建总结块并有选择地删除区块链数据的概念[70]。其中，总结块是每个节点周期性创造的，比如每一百个区块创造一个总结块。当收集的区块超过一定数量后，在遵循轮询(round-robin)的原则下，第一个序列的内容复制到新的总结块，并删除第一个序列的内容。随着不断删除区块，创世块可能会随之移动。但是如果没有足够的区块达到指定数量就

无法删除数据, 为了解决该问题, 方案使用了填充空块的方法, 因此方案不能做到及时删除, 而是需要等待一段时间。

同样从数据结构的角度, Daniel 提出了可变区块链结构的概念和一个以可移动方式在链上存储数据的机制[71]。方案没有使用替代块或投票的方式来移除区块数据, 而是把区块分为永久块和可移动块, 结合类似于跳跃表的跳跃链(skipchains), 能够移除某些区块, 同时保持整条链的可验证性。其数据结构是让一个永久块的哈希指针指向另一个永久块和一个可移动块, 而可移动块直接指向前一个块, 两个相邻永久块之间的可移动块数量称为间隔。如果想要删除交易或者区块, 可以通过删除包含相应块的完整间隔来删除可移动块。该方案不仅适用于联盟链场景, 同样也适用于拥有大量节点的公有链场景, 例如在某些应用的内容管理中, 可移动交易包含一些私人数据, 授权用户能根据规则删除这些数据。不过只能做到删除操作, 不能实现插入或修改区块。

3.3.2 编辑涉及的区块链架构

目前可编辑区块链方案涉及的链结构主要分为单链和多链。单链是指不需要辅助链即可完成编辑操作, 如 Deuber 的方案[8]。而多链结构是指需要借助被编辑链之外的另一条或多条链来实现编辑操作。有些方案中使用多链的目的主要是并行处理编辑操作, 提高速度和增加吞吐量, 以侧链为例, 它是连接不同区块链的桥梁, 能够协助异构链之间信息的交互, 实现不同数字资产或数据信息在多个区块链间的转移和传输, 从而拓展区块链的功能, 如上文讨论的 Lee 等人提出的方案[37], 该方案的矿工需要在挖出的区块里进行数字签名并附带公钥作为编辑证明, 以实现后续的验证、问责和追溯功能, 同时方案使用由一条主链和多条侧链组成的多链结构来提高交易修改的效率; 另外, 使用多链结构也是为了辅助存储编辑证明, 使方案能够更好地兼容现有区块链, 如 Thyagarajan 等人提出的方案[59]。

在使用多链架构的方案中, Marsalek 等人提出使用相同创世块的两条链来实现编辑操作的方法[72], 其中, 一条是原始数据链, 和传统区块链一样。另一条是修正链, 该链主要用来存储修改后的新数据。Marsalek 方案的区块链结构如图 11 所示。

Marsalek 方案的编辑证明主要由修正链上的纠正块和原始数据链上的矿工投票组成, 其大体思想是: 用户提交的编辑请求中指明了具体要修改哪个区块以及修改后是什么样, 并包含能够构建出纠正块的所有必须数据。然后每个矿工验证该请求是否

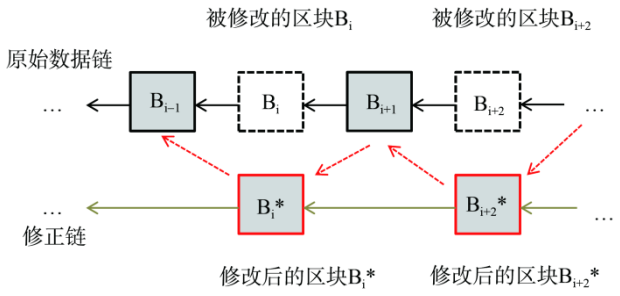

图 11 Marsalek 方案的区块链结构

Figure 11 Blockchain structure of Marsalek's schem

符合指定策略, 如果满足条件, 在投票阶段可以对其进行投票, 并把投票包含在新挖的区块里。投票期过后统计票数, 如果同意票超过一定阈值, 则对相应区块进行物理修改或删除, 同时修正链连接对应的纠正块。方案中的验证者需要同时验证和维护两条区块链, 当顺序遍历区块链时, 如果在原始数据链上发现区块被修改, 就切换到修正链上读取相应位置的纠正块数据。该方案和 Deuber 的方案都允许任何人提出对区块的编辑, 而且是基于投票机制的块级编辑, 同时也像 Reparo 协议一样需要辅助链, 但是该方案是把修改的相关数据存储在和主链具有相同创世块的辅助链上, 不能兼容现有区块链。

Zhang 等人基于使用秘密共享的容错陷门恢复机制的可编辑区块链, 增加了备份陷门碎片的方法, 提出一个可信工业物联网的数据管理方案[73]。该方案使用了双链架构将数据管理与工业系统中的其他交易区分开来, 其中一条是包含普通交易的原始区块链, 另一条是包含所有陷门管理交易的监管区块链, 从而最大程度减少原始区块链系统中工业物联网设备的负担。为了满足工业场景的各种功能, 方案共有三阶段: 在设置阶段, 由系统管理者设置相关参数并将陷门碎片分发给不同节点; 在第二阶段, 陷门碎片持有者通过监管区块链共同生成变色龙哈希的公钥; 在数据管理阶段中, 系统会定期检查链上数据是否正确, 若检测到错误数据, 则监管区块链上的节点会联合决定是否同意执行编辑, 之后, 陷门持有者将在链上发布碎片并合成陷门, 剩余节点使用智能合约来验证发布陷门的正确性, 以实现可靠的问责制, 编辑者将依据编辑请求使用陷门对相应信息进行编辑。如果某些碎片不正确导致无法正确恢复陷门, 那么将由其余的陷门持有者公布备份碎片, 以恢复完整的陷门。由于持有陷门的用户在修改操作提交给一个中心执行者时, 需要验证区块链是否被修改过以及是否被正确修改, 这会增加额外的开销。除了上述讨论的双链结构, 还有其他多链

结构的可编辑方案, 如 Kuperberg 等人提出支持剪枝、数据滚动等技术的一个新架构[67]。与侧链和状态通道不同, 该架构主要使用子链和上下文链树, 其中上下文是指交易类型和涉及实体之间的关系, 比如个体转给个体的交易、组织转给组织的交易等。该方案的思想和 Puddu 方案[11]相似, 主要区别在于后者存在时间窗口的限制, 不能删除或修改已在链上稳定的单个交易或区块, 而 Kuperberg 的方案能够删除, 但不能重写。

4 可编辑区块链的应用方案

许多应用都涉及实体间的信任问题, 区块链则能够实现去中心化, 搭建无交集实体之间的信任桥梁, 而可编辑操作又赋予区块链对数据的增、删、改功能, 拓展了区块链的使用范围, 使许多应用可以从可编辑方案中受益。目前, 除了上文讨论的可编辑区块链方案外, 一些研究也运用了可编辑区块链技术来满足多元异构的应用场景需求, 本节将依据不同的场景对这些应用进行分类讨论, 探索其应用价值。

4.1 可编辑区块链和身份管理的结合

如今的身份管理系统, 除了使用传统的口令密码和物理证明, 越来越多的生物识别技术已融入行径追踪、交易支付等不同领域的系统中, 包括指纹识别、人脸识别等。在现有系统里, 为了能对用户的身份进行有效识别或认证, 通常把原始的指纹特征图像集中存储在数据库中。然而, 这种方式可能存在数据被恶意敌手篡改的问题, 为了降低这种中心化带来的风险, 需要在一定条件下允许对数据执行编辑操作, Zhu 等人提出了一种基于可编辑区块链的指纹识别系统[16]。该系统利用变色龙哈希算法对指纹进行哈希计算, 并将其存储于区块链中, 具体来说, 继续使用传统的哈希算法计算区块头, 而区块体则采用变色龙哈希算法。其中把变色龙哈希函数的陷门给予管理员, 他可以更新指纹识别系统中的用户数据以及对区块体中的信息进行编辑, 从而在不改变区块链结构的基础上, 实现对用户指纹数据的删除或修改。虽然该系统成功的把区块链和生物特征识别技术进行了结合, 但是系统是部署在私有链上, 变色龙哈希函数的陷门只由管理员管理, 存在编辑权中心化的问题。其次, 方案中的每个区块只存储一个用户的指纹哈希, 效率较低。可以考虑将陷门碎片分发给多个可信管理员, 以及考虑每个区块能够存储多个用户指纹哈希的多中心化方案。

如今的用户都希望获得更丰富的网络服务, 而不会泄露个人隐私信息。在无线移动网络场景下, 用户的身份和对应密钥通常是由网络运营商或服务商生成、授权和管理的, 这些个人信息的集中化存储给社会带来了一些信任和安全问题。在此背景下, Xu 等人提出了一种基于区块链的无线移动网络身份管理和认证方案[74], 允许用户控制自己的身份信息, 生成自己的主权身份和公私钥对, 并由区块链记录相应的身份和公钥信息。方案主要采用了基于变色龙哈希的可编辑区块链技术来删除链上非法用户的信息, 同时服务商可以通过在区块链上查询用户的主权身份和公钥, 以验证用户的身份。将可编辑技术与身份管理系统进行融合, 不仅能够减少中心实体干预的成本, 还能促进多元异构系统间信息的整合, 更加切合现实需求。

4.2 可编辑区块链和物联网的结合

可编辑技术可以有效减少物联网中传感器数据的存储, 已有许多工作提出或正在进行基于可编辑技术的物联网系统构造, 包括上文[12,46,65,73]的方案, 这些可编辑方案大多适用于私有链或联盟链。在可编辑私有链中, 物联网数据的编辑权限仅授予可信实体, 通常是核心机构, 而对数据的访问权可能是有限的, 也可能是公开的。在这种场景下, 涉及的陷门可授予权威机构, 数据编辑权的分配和管理较为简单。在可编辑联盟链中, 编辑共识通常是由一组实体组成的联盟共同达成。此时, 可在这些实体间共享陷门碎片或编辑权, 并使用安全多方计算或其他技术实现编辑操作。

为了解决存储空间不断增长以及合理删除数据的问题, 结合物联网场景更适合使用轻量级区块链, Chan 等人引入了称为 LiTiChain 的可扩展轻量级架构[75], 这是一个具有有限生命周期的区块链, 生命周期已过期的交易和区块可以安全地从链中移除, 包括交易值。LiTiChain 由代表生命周期到期顺序的树 EOG 和代表区块创建顺序的线性图 AOG 组成, 前者确保连通性, 后者增加区块高度。如果后续区块引用了被删除的区块, 为了保证剩余区块的可验证性, 过期区块可能仍需保留在链中, 这会产生额外的存储成本。因为物联网传感器的资源有限, 需要对其数据进行剪枝、压缩, 这样不仅能减少数据传输和存储空间的消耗, 还能保护真实数据不被泄露, 所以 Chang 等人提出了有界误差剪枝的数据隐私保护方案[76]。该方案主要通过区块链的智能合约和分布式文件系统提供给用户不同的数据资源, 同时根据不同需求, 允许数据拥有者售卖他们经过有界误差剪枝处理的数据。

面对多元异构的大数据场景, 人工智能和物联网(IoT)的结合促进了大数据的传输和处理, 其中利用区块链来管理海量数据和 IoT 设备成为了一个研究热点, 但区块链的不可篡改性阻碍其进一步发展。虽然许多方案能够实现对区块链的可编辑操作, 但是这些方案往往不能及时、自动地纠正错误, 缺乏智能处理。为支持区块链能够智能地执行编辑操作, Huang 等人提出了自主可编辑的区块链概念, 并提出一个带有临时陷门的可撤销变色龙哈希方案[77], 利用该方案, 自主可编辑的区块链将作为物联网的一个智能信任层。为防止滥用陷门, 方案使用临时陷门且其效用会定时到期, 虽然引入额外的计算来实现可编辑操作, 但能以离线的方式自动执行这些操作。

在工业物联网(IIoT)的场景下, 同样也需要使用可控的区块链, 因为如果发现区块链数据被篡改, 为了使数据恢复到正常状态, 需要对区块链历史进行编辑。然而现有的可编辑区块链方案思想不能直接应用到工业物联网中, 所以 Huang 等人构建了一个可编辑的联盟区块链[78], 并提出一个门限变色龙哈希方案和可问责的变色龙签名方案。前者实现了变色龙哈希的多中心化设计, 后者实现了可以用更新的签名验证编辑内容的功能。构建的方案允许一组授权传感器重写区块链数据, 符合具有较低算力的工业物联网设备要求。

随着近几年智能电网的发展, 分布式能源交易也迅速发展, 然而利用区块链技术的方案基本是建立在半去中心化的环境上, 用户的隐私不能得到很好的保护, 对此 Yang 等人提出了一个基于联盟区块链点对点的能源交易系统[79], 该系统使用 CP-ABE 实现对用户的隐私保护, 使用人工神经网络来预测能源的生成, 同时还使用了基于变色龙哈希函数的可编辑区块链技术, 让数据拥有者能够修改链上存储的敏感信息。在区块链和物联网结合的基础上, 采用可编辑技术能够丰富当前已实现的功能, 更加贴合物联网轻量级设备的特性, 加速数据处理和转化。

4.3 可编辑区块链和外包技术的结合

在可编辑区块链方案中, 特别是基于变色龙哈希函数的方案, 会涉及一些复杂且高负荷的计算, 比如椭圆曲线中的配对操作。然而现实中大多数用户只拥有算力较小的轻量级设备, 为了减少普通用户的计算负荷, Guo 等人提出了把基于策略的变色龙哈希和离线技术相结合的方案[80]。该方案把基于策略的变色龙哈希涉及的复杂计算外包给云, 采用在线和离线的双方式对区块链进行编辑, 简单来说就是把高计算量的操作移到离线阶段进行预计算, 到在线阶段时就能够减小计算所需要的算力。其中外包计算是可审计的, 用户可以验证计算结果是否正确, 且可以控制用户使用外包的次数, 该功能契合用户付费给云计算, 以获得相应计算次数的场景。

通过分析目前运用可编辑区块链技术的应用方案, 可以观察到: 一方面, 当前的应用大多只局限于使用基于变色龙哈希函数的可编辑区块链技术, 把陷门授予单个实体, 导致编辑权限相对中心化。另一方面, 已有的应用方案大多基于联盟链或私有链, 很少基于公有链场景。在可编辑公有链中, 编辑权不受任何一方控制, 是完全去中心化的, 各方能随时参与数据的读和写操作, 编辑共识通常是以投票的方式来达成。此外, 现有可编辑区块链的方案很多是基于诸如比特币、以太坊等无许可链设计的, 所以今后可以针对无许可区块链的不同场景进行探索和研究。

5 开放的研究问题

基于上文讨论的可编辑区块链方案和我们目前的观察分析, 本节主要分享一些目前可编辑区块链方案有待完善或尚未解决的开放研究问题, 以及我们对如何解决这些问题的看法。

5.1 可编辑区块链与分片、跨链的结合

随着区块链用户的增多, 存储空间越来越大, 较低的吞吐量和较高的交易确认时延等问题限制了区块链的许多应用发展, 与区块链密切相关的拓展性问题很快引起广泛关注[81], 对这方面的研究也成为近几年的一大热点。当前可拓展区块链方案主要分为链上和链下, 链上包括改进区块结构、共识算法、分片、DAG 等, 链下包括状态通道、侧链和跨链等。可以发现, 可编辑区块链和可拓展区块链方案都是为了打破区块链技术应用到现实场景的限制。然而, 目前还没有方案把可编辑区块链和可拓展性技术进行结合, 特别是跨链和分片技术。如果两者可以进行结合, 那么就能够实现一条链上的节点修改另一条链上的数据, 或是一个分片上节点修改另一个分片数据。此外, 使用分片的思想不仅能够显著增加普通交易的吞吐量, 还能提高待编辑交易的处理速度。例如可以考虑使用一个主分片来统一汇总各分片处理后的编辑交易。使用跨链技术, 不仅能够打破同构或多元异构链间的数据孤岛问题, 实现链间的价值转移, 增强区块链的互操作性, 而且还特别符合近期火热的元宇宙愿景的思想[82]。所以我们认为将跨链、分片和可编辑区块链方案进行结合是一个有待探索、

有前景的方向。

5.2 可编辑区块链与隐私保护

现如今在信息技术、数字经济高速发展的时代, 个人信息数据成为了各行业关注的焦点, 同时也提出了许多和信息安全相关的概念。越来越多的用户会选择注重隐私保护的应用服务, 包括在使用服务期间不能泄露用户的真实身份、具体的交易内容等敏感数据。为加速区块链应用的落地, 需要完善可编辑区块链的隐私保护功能。即区块链的可编辑功能不能泄露编辑请求者和编辑者的身份, 且在全匿名环境下, 区块链的可编辑功能还不能泄露编辑操作涉及的交易内容和交易双方的身份。经过对现有可编辑区块链方案的分析总结, 发现在大部分方案中, 用户的身份和交易内容都是透明公开的, 用户的隐私会在编辑过程中被泄露。而只有少部分方案考虑加入了隐私保护技术[40,43-44,46,49,51-52]。为了缓解这个问题, 有必要提出一个保护编辑请求者和编辑者身份隐私的可编辑区块链系统, 如[44]使用动态群签名方案实现编辑者的身份匿名功能, 并在必要时由群管理员揭示编辑者的身份以实现问责功能。此外, 目前还没有基于全匿名环境下的可编辑区块链方案, 如在 Zcash、Monero 匿名环境里编辑交易或区块数据, 以保护交易双方的身份和交易内容不被泄露。为了防止匿名环境下诸如洗钱的违法行为发生, 在设计隐私保护的可编辑区块链方案时还需要权衡匿名和问责两方面, 可以考虑采用零知识证明技术或可追踪的匿名技术, 如群签名和环签名等。

5.3 旧数据遗留问题

通常, 在对旧数据进行编辑后, 每个节点都会对被编辑的交易或区块进行本地更新, 但是有可能存在恶意节点故意不更新数据, 本地保留旧数据备份的情况。如果在编辑区块链后, 不诚实节点还保留着编辑前的旧数据, 那么虽然这些旧数据在区块链上无效, 只是处于“离链”状态, 但是这样的行为依然会侵害到数据拥有者的隐私或利益。例如, 节点 A 发现自己的一些隐私信息被存储在区块链上, 在请求删除这些数据时, 恶意节点 B 可能拒绝执行删除操作或是本地保留一份副本, 这样节点 A 的隐私信息还是没有被彻底删除。因为不易察觉节点不执行编辑或本地保留旧数据的行为, 目前还没有方案能够实现强制所有节点更新被编辑的交易或区块。

6 总结

区块链技术的不可篡改性是一把双刃剑, 它既保证了区块链数据的安全, 又阻碍了区块链在某些领域的进一步应用发展, 可编辑区块链的目的就是在这两者利弊之间做出权衡, 使区块链技术更好地和不同场景下的服务进行融合。本文结合提出的编辑统一流程和六种不同分类, 从编辑共识、编辑处理和编辑证明三个重要环节系统地分析了当前先进的可编辑区块链方案, 分别总结了每个方案的核心思想、新颖之处和优缺点, 并对几个基于可编辑区块链技术的典型应用方案、密码学原语、共识机制等进行了讨论, 最后指出了目前方案还需要完善或有待解决的问题, 为相关研究人员提供新见解和未来探索的新方向。

参考文献

[1] Tanwar S, Parekh K, Evans R. Blockchain-Based Electronic Healthcare Record System for Healthcare 4.0 Applications[J]. *Journal of Information Security and Applications*, 2020, 50: 102407.

[2] Hackius N, Petersen M. Blockchain in logistics and supply chain: trick or treat?[C]. *Proceedings of the Hamburg International Conference of Logistics*. 2017, 23: 3-18.

[3] Hou H. The Application of Blockchain Technology in E-Government in China[C]. *2017 26th International Conference on Computer Communication and Networks*, 2017: 1-4.

[4] Tsankov P, Dan A, Drachsler-Cohen D, et al. Securify: Practical Security Analysis of Smart Contracts[C]. *The 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018: 67-82.

[5] Matzutt R, Hiller J, Henze M, et al. A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin[C]. *22nd International Conference on Financial Cryptography and Data Security.* 2018: 420-438.

[6] Schwerin S. Blockchain and Privacy Protection in the Case of the European General Data Protection Regulation (GDPR): A Delphi Study[J]. *The Journal of British Blockchain Association*, 2018, 1(1): 1-77.

[7] Ateniese G, Magri B, Venturi D, et al. Redactable Blockchain - or - Rewriting History in Bitcoin and Friends[C]. *2017 IEEE European Symposium on Security and Privacy*, 2017: 111-126.

[8] Deuber D, Magri B, Thyagarajan S A K. Redactable Blockchain in the Permissionless Setting[C]. *2019 IEEE Symposium on Security and Privacy*, 2019: 124-138.

[9] Tian Y G, Liu B W, Li Y J, et al. Accountable Fine-Grained Blockchain Rewriting in the Permissionless Setting[EB/OL]. 2021: arXiv: 2104.13543. https://arxiv.org/abs/2104.13543.

[10] Politou E, Casino F, Alepis E, et al. Blockchain Mutability: Challenges and Proposed Solutions[J]. *IEEE Transactions on Emerging Topics in Computing*, 2021, 9(4): 1972-1986.

[11] Puddu I, Dmitrienko A, Capkun S. μchain: how to forget without hard forks. IACR Cryptology ePrint archive: report 2017/106 [Online], Available: http://eprint.iacr.org/2017/106, January 1, 2019.

[12] Dorri A, Kanhere S S, Jurdak R. MOF-BC: A Memory Optimized

and Flexible Blockchain for Large Scale Networks[J]. *Future Generation Computer Systems*, 2019, 92: 357-373.

[13] Meiklejohn S. Top Ten Obstacles along Distributed Ledgers Path to Adoption[J]. *IEEE Security & Privacy*, 2018, 16(4): 13-19.

[14] Yuan Y, Wang F Y. Editable Blockchain: Models, Techniques and Methods[J]. *Acta Automatica Sinica*, 2020, 46(5): 831-846.
(袁勇, 王飞跃. 可编辑区块链: 模型、技术与方法[J]. *自动化学报*, 2020, 46(5): 831-846.)

[15] Zhang D, Le J Q, Lei X Y, et al. Exploring the Redaction Mechanisms of Mutable Blockchains: A Comprehensive Survey[J]. *International Journal of Intelligent Systems*, 2021, 36(9): 5051-5084.

[16] Zhu Y Y, Li S, Feng G R, et al. Fingerprint Recognition System Based on Editable Blockchain[J]. *Journal of Applied Sciences*, 2021, 39(2): 330-337.
(朱艳艳, 李晟, 冯国瑞, 等. 基于可编辑区块链的指纹识别系统[J]. *应用科学学报*, 2021, 39(2): 330-337.)

[17] Ateniese G, Medeiros B D. Identity-based Chameleon Hash and Applications[C]. *2004 International Conference on Financial Cryptography,* 2004: 164-180.

[18] H. Krawczyk and T. Rabin, Chameleon Hashing and Signatures. IACR Cryptol. ePrint archive: report 1998/101[Online]. Available: https://eprint.iacr.org/1998/010, Mar 17, 1998.

[19] Chen X, F Zhang, Kim K. Chameleon hashing without key exposure[C]. *7th International Conference on Information Security*, 2004: 87-98.

[20] Camenisch J, Derler D, Krenn S, et al. Chameleon-Hashes with Ephemeral Trapdoors: And Applications to Invisible Sanitizable Signatures[C]. *Public-Key Cryptography–PKC 2017: 20th IACR International Conference on Practice and Theory in Public-Key Cryptography*, 2017: 152-182.

[21] Shamir A. How to share a secret[J]. *Communications of the ACM*, 1979, 22(11): 612-613.

[22] Yao A C. Protocols for Secure Computations[C]. *23rd Annual Symposium on Foundations of Computer Science*, 2008: 160-164.

[23] Harn L. Group-oriented (t,n) threshold digital signature scheme and digital multisignature[J]. *IEE Proceedings-Computers and Digital Techniques*, 1994, 141(5):307-313.

[24] Boneh D, Gentry C, Lynn B, et al. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps[C]. *The 22nd international conference on Theory and applications of cryptographic techniques*, 2003: 416-432.

[25] Chaum D, Eugène van Heyst. Group Signatures[C]. *Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques*, 1991: 257-265.

[26] Rivest R L, Shamir A, Tauman Y. How to Leak a Secret[C]. *The 7th International Conference on the Theory and Application of Cryptology and Information Security*: *Advances in Cryptology*, 2001: 552-565.

[27] Sahai A, Waters B. Fuzzy Identity-Based Encryption[C]. *The 24th annual international conference on Theory and Applications of Cryptographic Techniques*, 2005: 457-473.

[28] Micali S, Rabin M, Vadhan S. Verifiable Random Functions[C]. *40th Annual Symposium on Foundations of Computer Science*, 2002: 120-130.

[29] Boneh D, Bonneau J, Benedikt Bünz, et al. Verifiable Delay Functions[C]. *38th Annual International Cryptology Conference*, 2018: 757-788.

[30] Badruddoja S, Dantu R, He Y Y, et al. Making Smart Contracts Smarter[C]. *2021 IEEE International Conference on Blockchain and Cryptocurrency*, 2021: 1-3.

[31] Krug J, Peterson J. Sidecoin: A Snapshot Mechanism for Bootstrapping a Blockchain[EB/OL]. 2015: arXiv: 1501.01039. https:// arxiv.org/abs/1501.01039.

[32] E. Palm, O. Schelén, U. Bodin. Selective Blockchain Transaction Pruning and State Derivability[C]. *2018 Crypto Valley Conference on Blockchain Technology*, 2018: 31-40.

[33] van Dijk M, Gentry C, Halevi S, et al. Fully Homomorphic Encryption over the Integers[C]. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2010: 24-43.

[34] Bünz B, Bootle J, Boneh D, et al. Bulletproofs: Short Proofs for Confidential Transactions and more[C]. *2018 IEEE Symposium on Security and Privacy*, 2018: 315-334.

[35] Garay J, Kiayias A, Leonardos N. The Bitcoin Backbone Protocol: Analysis and Applications[C]. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2015: 281-310.

[36] R Pass, L Seeman, A Shelat. Analysis of the Blockchain Protocol in Asynchronous Networks[C]. *36th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2017: 643-673.

[37] Lee N Y, Yang J H, Onik M M H, et al. Modifiable Public Blockchains Using Truncated Hashing and Sidechains[J]. *IEEE Access*, 2019, 7: 173571-173582.

[38] Jia Y X, Sun S F, Zhang Y, et al. Redactable Blockchain Supporting Supervision and Self-Management[C]. *The 2021 ACM Asia Conference on Computer and Communications Security*, 2021: 844-858.

[39] Ashritha K, Sindhu M, Lakshmy K V. Redactable Blockchain using Enhanced Chameleon Hash Function[C]. *2019 5th International Conference on Advanced Computing & Communication Systems*, 2019: 323-328.

[40] Tian Y G, Li N, Li Y J, et al. Policy-Based Chameleon Hash for Blockchain Rewriting with Black-Box Accountability[C]. *ACSAC'20*: *Annual Computer Security Applications Conference*, 2020: 813-828.

[41] Ateniese G, de Medeiros B. On the Key Exposure Problem in Chameleon Hashes[J]. *Lecture Notes in Computer Science*, 2005, 3352: 165-179.

[42] Wu C H, Ke L S, Du Y S. Quantum Resistant Key-Exposure Free Chameleon Hash and Applications in Redactable Blockchain[J]. *Information Sciences*, 2021, 548: 438-449.

[43] Huang K, Zhang X S, Mu Y, et al. Scalable and Redactable Blockchain with Update and Anonymity[J]. *Information Sciences*, 2021, 546: 25-41.

[44] Panwar G, Vishwanathan R, Misra S. ReTRACe: Revocable and Traceable Blockchain Rewrites using Attribute-based

Cryptosystems[C]. *The 26th ACM Symposium on Access Control Models and Technologies. ACM*, 2021: 103-114.

[45] Xu S M, Ning J T, Ma J H, et al. K-Time Modifiable and Epoch-Based Redactable Blockchain[J]. *IEEE Transactions on Information Forensics and Security*, 2021, 16: 4507-4520.

[46] Ren Y L, Cai X J, Hu M Q. Privacy-Preserving Redactable Blockchain for Internet of Things[J]. *Security and Communication Networks*, 2021, 2021(6): 1-12.

[47] Li X Y, Xu J, Yin L Y, et al. Escaping from consensus: Instantly redactable blockchain protocols in permissionless setting[J]. IEEE Transactions on Dependable and Secure Computing, 2022(01): 1-20.

[48] Xu J, Li X Y, Yin L, et al. Redactable Blockchain Protocols with Instant Redaction. IACR Cryptol. ePrint archive: report 2021/223[Online]. Available: https://eprint.iacr.org/2021/223, May 17, 2021.

[49] Ren Y L, Xu D T, Zhang X P, et al. Deletable Blockchain Based on Threshold Ring Signature[J]. *Journal on Communications*, 2019, 40(4): 71-82.
(任艳丽, 徐丹婷, 张新鹏, 等. 基于门限环签名的可删除区块链[J]. *通信学报*, 2019, 40(4): 71-82.)

[50] SS Al-Riyami, KG Paterson. Certificateless public key cryptography[C]. *9th International Conference on the Theory and Application of Cryptology and Information Security*, 2003: 452-473.

[51] Li K M, Zheng D, Guo R. An Anonymous Editable Blockchain Scheme Based on Certificateless Aggregate Signature[C]. *2021 3rd International Conference on Natural Language Processing*, 2021: 57-67.

[52] Cai X J, Ren Y L, Zhang X P. Privacy-Protected Deletable Blockchain[J]. *IEEE Access*, 2019, 8: 6060-6070.

[53] Li P L, Xu H X, Ma T J, et al. Research on Fault-Correcting Blockchain Technology[J]. *Journal of Cryptologic Reseatch*, 2018, 5(5): 501-509.
(李佩丽, 徐海霞, 马添军, 等. 可更改区块链技术研究[J]. *密码学报*, 2018, 5(5): 501-509.)

[54] Manevich Y, Barger A, Assa G. Redacting Transactions from Execute-Order-Validate Blockchains[C]. *2021 IEEE International Conference on Blockchain and Cryptocurrency*, 2021: 1-9.

[55] Rivest R L, Shamir A, Adleman L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems[J]. *Communications of the ACM*, 1983, 26(1): 96-99.

[56] Grigoriev D, Shpilrain V. RSA and Redactable Blockchains[J]. *International Journal of Computer Mathematics*: *Computer Systems Theory*, 2021, 6(1): 1-6.

[57] MS Dousti, A Küpü. Tri-op redactable blockchains with block modification, removal, and insertion. IACR Cryptology ePrint archive: report 2021/724[Online]. Available: https://eprint.iacr.org/2021/724, Jun 1, 2021.

[58] Martins F F, Matos D R, Pardal M L, et al. Recoverable Token: Recovering from Intrusions Against Digital Assets in Ethereum[C]. *2020 IEEE 19th International Symposium on Network Computing and Applications*, 2021: 1-9.

[59] Thyagarajan S A K, Bhat A, Magri B, et al. Reparo: Publicly verifiable layer to repair blockchains[C]. *Financial Cryptography and Data Security*, 2021: 37-56.

[60] Florian M, Henningsen S, Beaucamp S, et al. Erasing Data from Blockchain Nodes[C]. *2019 IEEE European Symposium on Security and Privacy Workshops*, 2019: 367-376.

[61] MS Dousti, A Küpü. Moderated Redactable Blockchains: A Definitional Framework with an Efficient Construct[M]. *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, 2020: 355-373.

[62] R Matzutt, Henze M, Ziegeldorf J H, et al. Thwarting Unwanted Blockchain Content Insertion[C]. *2018 IEEE International Conference on Cloud Engineering*, 2018: 364-370.

[63] Derler D, Kai S, Slamanig D, et al. Fine-Grained and Controlled Rewriting in Blockchains: Chameleon-Hashing Gone Attribute-Based[C]. *The 26th Annual Network and Distributed System Security Symposium*, 2019: 24-27.

[64] Sartori, Damiano. Redactable Blockchain: how to change the immutable and the consequences of doing so[D]. UNIVERSITY OF TWENTE STUDENT THESES, 2020.

[65] Niya S R, Willems J, Stiller B. On-Chain IoT Data Modification in Blockchains[EB/OL]. 2021: arXiv: 2103.10756. https://arxiv.org/abs/2103.10756.

[66] Hou H Y, Hao S D, Yuan J M, et al. Fine-Grained and Controllably Redactable Blockchain with Harmful Data Forced Removal[J]. *Security and Communication Networks*, 2021, 2021(1): 1-20.

[67] Kuperberg M. Towards Enabling Deletion in Append-only Blockchains to Support Data Growth Management and GDPR Compliance[C]. *2020 IEEE International Conference on Blockchain*, 2020: 393-400.

[68] Matzutt R, Kalde B, Pennekamp J, et al. How to Securely Prune Bitcoin's Blockchain[C]. *2020 IFIP Networking Conference*, 2020: 298-306.

[69] Matzutt R, Kalde B, Pennekamp J, et al. CoinPrune: Shrinking Bitcoin's Blockchain Retrospectively[J]. *IEEE Transactions on Network and Service Management*, 2021, 18(3): 3064-3078.

[70] Hillmann P, Knüpfer M, Heiland E, et al. Selective Deletion in a Blockchain[C]. *2020 IEEE 40th International Conference on Distributed Computing Systems*, 2020: 1249-1256.

[71] Daniel E, Tschorsch F. Towards Verifiable Mutability for Blockchains[EB/OL]. 2021: arXiv: 2106.15935. https://arxiv.org/abs/ 2106.15935.

[72] Marsalek A, Zefferer T. A Correctable Public Blockchain[C]. *2019 18th IEEE International Conference on Trust*, *Security and Privacy In Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering*, 2019: 554-561.

[73] Zhang C, Ni Z F, Xu Y, et al. A Trustworthy Industrial Data Management Scheme Based on Redactable Blockchain[J]. *Journal of Parallel and Distributed Computing*, 2021, 152: 167-176.

[74] Xu J, Xue K P, Tian H Y, et al. An Identity Management and Authentication Scheme Based on Redactable Blockchain for Mobile Networks[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(6): 6688-6698.

[75] Pyoung C K, Baek S J. Blockchain of Finite-Lifetime Blocks with Applications to Edge-Based IoT[J]. *IEEE Internet of Things*

Journal, 2020, 7(3): 2102-2116.

[76] Chang R I, Wei L C, Wang C H, et al. Blockchain for Bounded-Error-Pruned Content Protection[J]. *ICT Express*, 2021, 7(3): 295-299.

[77] Huang K, Zhang X S, Mu Y, et al. Achieving Intelligent Trust-Layer for IoT via Self-Redactable Blockchain[J]. *IEEE Transactions on Industrial Informatics*, 2019, PP(99): 1.

[78] Huang K, Zhang X S, Mu Y, et al. Building Redactable Consortium Blockchain for Industrial Internet-of-Things[J]. *IEEE Transactions on Industrial Informatics*, 2019, 15(6): 3670-3679.

[79] Yang W T, Guan Z T, Wu L F, et al. Autonomous and Privacy-Preserving Energy Trading Based on Redactable Blockchain in Smart Grid[C]. *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2021: 1-6.

[80] Guo L F, Wang Q L, Yau W C. Online/Offline Rewritable Blockchain with Auditable Outsourced Computation[J]. *IEEE Transactions on Cloud Computing*, 2023, 11(1): 499-514.

[81] Belchior R, Vasconcelos A, Guerreiro S, et al. A Survey on Blockchain Interoperability: Past, Present, and Future Trends[J]. *ACM Computing Surveys*, 2022, 54(8): 1-41.

[82] Jiang Y, Kang J, Niyato D, et al. Reliable distributed computing for metaverse: A hierarchical game-theoretic approach[J]. *IEEE Transactions on Vehicular Technology*, 2022, 72(1): 1084-1100.

罗彬 于 2020 年在青岛大学信息安全专业获得学士学位。现在暨南大学网络空间安全专业攻读硕士学位。研究领域为区块链、密码学。Email: xsbinluo@163.com

温金明 于 2015 年在麦吉尔大学应用数学专业获得哲学博士学位。现任暨南大学网络空间安全学院研究员, 国家级青年人才, 青年珠江学者。研究领域为格密码、区块链、稀疏学习。Email: jinmingwen1@163.com

吴永东 于 1997 年在中国科学院自动化所获得哲学博士学位。目前是暨南大学网络空间安全学院研究员, 珠江学者讲座教授, 网络安全检测与防护国家地方联合工程中心常务副主任。研究兴趣包括区块链应用、人工智能安全、通信安全等。Email: wuyd007@qq.com

陈洁 于 2013 年在南洋理工大学获得哲学博士学位。现任华东师范大学软件工程学院研究员。研究兴趣包括公钥密码、密码应用等。Email: s080001@e.ntu.edu.sg