软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(12):3909−3922 [doi: 10.13328/j.cnki.jos.005894]

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

可修改的区块链方案*

任艳丽[1], 徐丹婷[1], 张新鹏[1], 谷大武[2]

[1](上海大学 通信与信息工程学院,上海 200444)

[2](上海交通大学 电子信息与电气工程学院,上海 200240)

通讯作者: 任艳丽, E-mail: renyanli@shu.edu.cn

摘 要: 随着区块链的迅速发展,上链数据不仅包括金融交易数据,还包括科技、文化、政治等多类数据.而在现有的区块链系统中,数据一旦上链便无法更改,可能会面临失效数据无法删除、错误数据无法修改等问题.因此,特定条件下可修改的区块链方案具有广阔的应用前景.在 POSpace(proof of space)共识机制下,基于陷门单向函数和新型区块链结构,提出了可修改的区块链方案.只要超过阈值数的节点同意,便可实现区块数据的合法修改,否则不能进行修改.除修改数据外,其余区块数据保持不变,全网节点仍可按原始验证方式对数据合法性进行验证.仿真实验表明:只要选定合适的阈值,所提方案中,区块生成与数据修改的效率均很高,数据的修改并不改变区块之间的链接关系,具有现实可操作性.

关键词: 区块链;可修改;陷门单向函数;空间证明;数据安全

中图法分类号: TP309

中文引用格式: 任艳丽,徐丹婷,张新鹏,谷大武.可修改的区块链方案.软件学报,2020,31(12):3909−3922. http://www.jos.org.cn/1000-9825/5894.htm

英文引用格式: Ren YL, Xu DT, Zhang XP, Gu DW. Scheme of revisable blockchain. Ruan Jian Xue Bao/Journal of Software, 2020,31(12):3909−3922 (in Chinese). http://www.jos.org.cn/1000-9825/5894.htm

Scheme of Revisable Blockchain

REN Yan-Li[1], XU Dan-Ting[1], ZHANG Xin-Peng[1], GU Da-Wu[2]

[1](School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China)

[2](School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: With the rapid development of the blockchain, the data on the chain not only include financial data, but also have data of technology, culture, politics, and so on. However, the data will not be revised once it is packaged on the existing blockchain system, which has the problem that the invalid data cannot be deleted and the wrong data cannot be modified. Therefore, a revisable blockchain under certain conditions has broad application prospects. Under the POSpace (proof of space) consensus mechanism, a revisable blockchain scheme is proposed based on the trapdoor one-way function and a new blockchain structure. In this scheme, the nodes can execute the revision operation as long as their number exceeds the threshold. Otherwise, the revision operation cannot be executed. Except for the revised data, the remaining data on the blocks keeps unchanged, and all of the nodes on the network can still verify the validity of data by using the original verification ways. The experiments show that block generation and data revision all have high efficiency as long as the threshold is selected appropriately, and the link relationship between the blocks will not be changed after the data revision, and the scheme has practical operability.

* 基金项目: 国家自然科学基金(U1736120, 61525203, U1636206); 上海市自然科学基金(20ZR1419700); 国家重点研发计划(2020YFC1523004)

Foundation item: National Natural Science Foundation of China (U1736120, 61572309, 61525203, U1636206); Beijing Municipal Natural Science Foundation (20ZR1419700); National Key Research and Development Program of China (2020YFC1523004)

收稿时间: 2019-03-15; 修改时间: 2019-09-06; 采用时间: 2019-09-27; jos 在线出版时间: 2019-12-05

CNKI 网络优先出版: 2019-12-05 14:54:55, http://kns.cnki.net/kcms/detail/11.2560.TP.20191205.1454.002.html

Key words: blockchain; revisable; trapdoor one-way function; proof of space; data security; data security

近年来,区块链技术得到学术界和企业界的广泛关注.从以比特币[1]为代表的加密货币系统——区块链1.0,到以太坊为代表的智能合约——区块链2.0,再到现在,区块链应用走出金融,走向社会多行业,进入区块链3.0时代.去中心化和上链数据不可篡改的特性,使其有别于传统的中心化系统,为金融、科技甚至政治等领域的数据存储提供了新的模式[2].

在区块链系统中,共识机制为各节点制定信任标准,构建激励机制,使得各节点就系统状态达成共识,共同维护系统健康运行[3].目前,主流的共识机制有基于工作量证明的POW(proof of work)共识机制[1,4]、基于权益证明的POS(proof of stake)共识机制[5−7]、基于授权股权证明的DPOS(delegated proof of stake)共识机制[8,9]等.现有的区块链系统基本以POW与POS共识机制为主,但POW基于算力竞争,挖矿电力损耗巨大,不利于节能环保.POS基于权益竞争,信用机制不牢固,大量低成本货币被分配于开发者,若利益驱使其大量抛售货币,将不利于系统维护. Park等人基于POSpace(proof of space)共识机制提出SpaceMint区块链系统[10],节点基于可循环利用的本地空间竞争挖矿,计算代价低,避免了算力竞争而造成的资源浪费.同时,安全的证明机制保证其信用牢固性,实现了对安全性与资源效率的兼顾.另外,SpaceMint 系统采用新型链式结构.传统的区块链利用哈希函数将区块头、区块体以及前后区块链接起来,以哈希函数的不可碰撞性保证数据安全.而SpaceMint通过加入签名子块,打破区块头和区块体的直接链接关系,构建了新型的区块链接结构,详见第1.3节.

现有的区块链系统中,数据一旦上链便无法更改.而随着区块链的发展,除金融数据外的科技、文化甚至政治等多类社会数据也将上链,失效数据无法删除、错误数据无法修改等问题将变得更加突出.例如:一个公司宣布破产,其交易数据失效,不再具备永久存储的价值,无法及时删除将造成存储资源浪费;或者某些上链的经典技术参数、公式在若干年后被推翻,需做出修正;又或者恶意、负面的舆论数据由于别有用心的挖矿者而上链,区块链的公开性将导致其不断扩散传播,造成恶劣影响.及时的数据修正在现有的区块链系统中均无法完成,可能造成巨大的经济损失和恶劣的社会影响.因此,上链数据无法更改将在一定程度上限制了区块链的进一步发展,而在特定条件下实现可修改的区块链,具有广阔的应用前景.

在可修改区块链研究中,爱哲森公司基于带陷门的变色龙哈希函数[11],在已知陷门时,可对区块数据进行修改.但该方案中,陷门被交于可信中心,修改权被中心化,数据安全面临威胁.Li 等人[12]对可修改区块链技术进行了改进,利用秘密共享方案,将陷门分配给系统各节点,当且仅当阈值节点联合时方可恢复陷门,实现数据修改.但上述方案均基于变色龙哈希函数实现区块数据修改,而现有的区块链系统大多基于传统的抗碰撞哈希函数,所以上述方案应用价值不高.Ren 等人[13]基于抗碰撞哈希函数,提出了可删除区块链系统,利用安全的门限环签名方案,在超过门限值的节点同意时,可对单个区块的全部交易数据进行删除.但该方案无法删除指定交易数据,也无法进行数据修改.本文使用抗碰撞哈希函数,基于POSpace共识机制实现可修改的区块链方案,在绝大多数节点同意后,可对指定交易数据进行修改或删除,具有重要的应用价值.

本文基于文献[10]中的区块结构和陷门单向函数,提出了可修改的区块链方案.通过引入机动因子,重构区块签名子块,在数据失效或错误时,只要超过阈值数节点同意,便可实现区块数据的合法修改;同时保持区块链接不变,除修改数据,其余数据不变,全网仍可按原始验证方式对数据合法性进行验证.而且我们设置了特定的阈值,使得修改操作必须经过绝大多数节点同意,恶意的非法修改无法完成,保证了数据的安全.

1 基础知识

本节将介绍一些基本概念,包括陷门单向函数、Grinding Blocks攻击及基于空间证明的区块链结构.

1.1 陷门单向函数

单向函数是密码学领域的一个重要概念,可简单描述为:正向求解容易,但反向求解不可实现的函数.陷门单向函数[14,15]不同于单向函数的“不可逆”,它是“陷门”可逆的.在陷门未知时,它等同于普通单向函数;当陷门已知时,求逆便计算上可实现.

定义 1(**陷门单向函数**). 陷门单向函数 $f(x)$是指满足以下性质的函数:

(1) 对于属于 f 定义域的任意 x,可在多项式时间内计算得到 $y=f(x)$;

(2) 对于属于 f 值域的任意 y,无法在多项式时间内求出 x,使得 $x=f^{-1}(y)$;

(3) 当 k 已知时,对于属于 f 值域的任何 y,可在多项式时间内计算出 $x=f_k^{-1}(y)$,其中,k 为该陷门单向函数的陷门.

1.2 Grinding Blocks攻击[10,16]

传统的区块结构分为区块头与区块体两部分:区块体存储交易信息,区块头包含版本号、父区块哈希、交易 Merkle Tree 根、时间戳、难度目标和 *Nonce*.挖矿挑战在于:寻找合适的随机数 *Nonce*,使区块头哈希结果小于难度目标.因此,挖矿挑战与交易信息相关.为了利益最大化,矿工可能将交易信息拆分成多份,以形成不同挖矿挑战,在同一时间寻找多个 *Nonce*,以打包形成不同区块,提高挖矿成功概率.此类行为没有遵守"将挖矿期间产生的交易全部打包于区块"的原则,影响系统效率,也可能进一步导致自私挖矿与双花攻击.此类行为称为 Grinding Blocks 攻击,如图 1 所示.

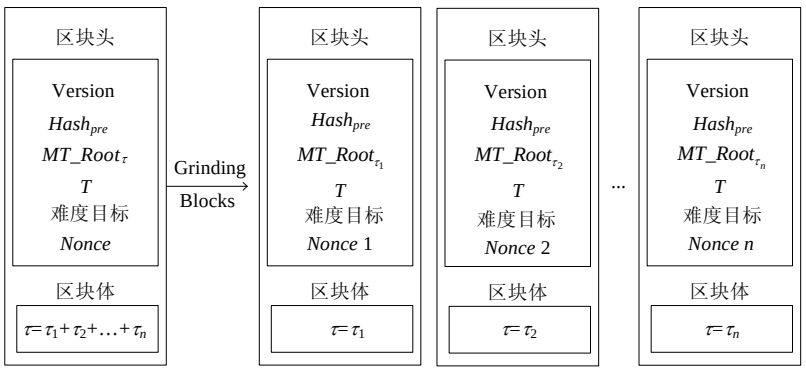


Fig.1 Schematic diagram of Grinding Blocks attack

图 1 Grinding Blocks 攻击示意图

在图 1 中,Version,$Hash_{pre}$,MT_Root_{τ},T,*Nonce* 和 τ依次表示区块头中的版本号、父区块哈希、交易 Merkle Tree 根、时间戳、矿工用以完成挖矿挑战的随机数和交易信息.

"Grinding Blocks"攻击在 POW 共识下不常见,因为 POW 共识机制基于算力挖矿,完成一次挖矿挑战需要耗费巨大算力,矿工在同一时间进行多个挑战的计算难度巨大.但在同时进行多个挖矿挑战难度较小的共识机制下,如第 1.3 节中介绍的基于空间证明的区块链中,"Grinding Blocks"攻击将变得相对容易.

1.3 基于空间证明的区块链[10,13]

最近,Park 等人构建了以"空间"为可信度衡量标准的区块链系统 SpaceMint,其共识机制为 POSpace.不同于 POW 和 POS,POSpace 将拥有最大存储空间的节点视为最可信节点,由其完成系统相关操作,获取奖励.各节点向全网证明自己空间大小、竞争成为记账者的过程,便是 POSpace 共识机制下的挖矿过程.因此,安全可靠的空间大小衡量标准是 POSpace 共识机制的关键.

POSpace 共识机制使用"pebble game",通过节点对有向无环图的构造速度来衡量空间大小.当时间一定时,节点空间越大,对有向无环图的构造越快;反之亦然.具体而言,POSpace 共识机制要求各节点在本地存储一有向无环图 $G=(V,E)$,其中,$V=\{v_1,v_2,\ldots,v_N\}$为顶点集合,N 为顶点个数,E 为有向边集合.有向无环图示例如图 2 所示.

为了突出有向无环图的结构,每个顶点 i 将被赋予特定的标签值,记为 l_i:

$$l_i = Hash(\mu,i,l_{p_1},l_{p_2},...,l_{p_t}),i=1,2,\ldots,N,$$

其中,i 为顶点序号,μ为与挖矿节点公钥关联的随机数,$l_{p_1},l_{p_2},...,l_{p_t}$ 为顶点 i 的所有母节点的标签值.

共识机制要求矿工存储有向无环图顶点信息,若矿工空间足够,便可全部存储;反之,矿工只能存储部分顶点标签值,未存储部分依靠后续计算得到,以时间换取空间.每次竞争时,系统随机要求矿工返回相关顶点标签值,显然,存储空间越大,矿工结果返回效率越高,赢取记账权概率越高.挖矿的具体实现过程见文献[10].这便形成了基于空间证明的共识机制.

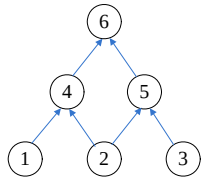


Fig.2 Directed acyclic structure diagram

图 2 有向无环结构图

同时,POSpace 对挖矿算力要求下降,矿工在同一时间完成多个挖矿挑战变得容易.因此,如第 1.2 节所述,在传统区块结构下,实行 POSpace 机制将使得 Grinding Blocks 攻击突出.为了对抗该攻击,SpaceMint 构建了全新的区块链结构.

如图 3 所示,新型区块 i 包含 3 个部分:证明子块φ_i、签名子块σ_i和交易子块τ_i.其中,证明子块φ_i和交易子块τ_i相当于传统区块结构中的区块头和区块体,而新增的签名子块σ_i则用于打破区块头和区块体之间的链接,使证明子块与交易子块不相连.因此,挖矿挑战与交易本身无关,矿工无法通过拆分交易以产生不同的挖矿挑战,从而增加挖矿成功概率,防止了 Grinding Blocks 攻击对系统的伤害.3 个子块的具体内容如下.

(1) Hash 子块$\varphi_i=Hash(i,\zeta_\varphi,(p_{ki},\gamma_i,c_i,a_i))$,具体包含:当前区块序号 i、记账者对前一区块的 Hash 子块φ_{i-1}的签名ζ_φ、记账者在竞争记账权时给出的承诺证明以及空间证明$(p_{ki},\gamma_i,c_i,a_i)$;

(2) 签名子块$\sigma_i=\{i,\zeta_\tau,\zeta_\sigma\}$,具体包含:当前区块序号 i、记账者对当前区块的交易子块τ_i的签名ζ_τ、记账者对前一区块的签名子块σ_{i-1}的签名ζ_σ;

(3) 交易子块$\tau_i=\{i,ctx\}$,具体包含当前区块序号 i、交易信息列表 ctx.

上述区块结构在保证数据安全的前提下,打破了证明子块φ_i与交易子块τ_i的直接链接关系,能更好地对抗“Grinding Blocks”攻击,也为可修改区块链提供了可能性.

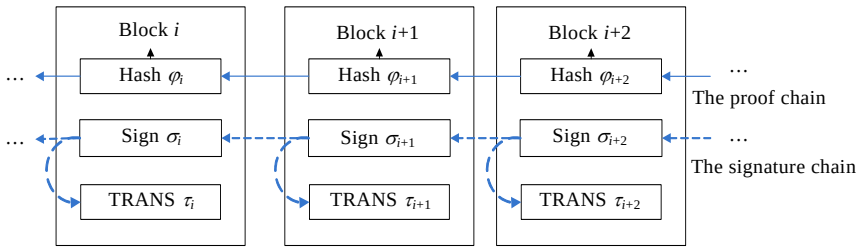


Fig.3 Blockchain structure of SpaceMint

图 3 SpaceMint 区块链结构

2 可修改的区块链

我们基于 SpaceMint 提出的区块链结构,使用陷门单向函数提出了可修改的区块链.在超过阈值数的节点同意时,可对失效或错误的上链交易数据进行合法修改,具体阈值设定会在下文给出分析.修改前后,区块链接结构不变,也不影响区块其他数据的使用和验证.而且由于阈值的设定,所有修改操作将代表系统意志,不合法的恶意修改无法完成.本节首先给出可修改的区块链结构,然后对修改原理及安全性进行分析.

2.1 可修改的区块链结构

基于 SpaceMint 的区块结构,我们提出可修改的区块链方案.如图 4 所示,其本质是重构签名子块σ_i,在记账者对交易的签名ζ_τ中加入机动因子 G_i,将σ_i变更为$\sigma_{i,G}$,而证明子块φ_i与交易子块τ_i保持不变.

图 5 对 SpaceMint 区块与可修改区块结构进行对比.原始签名子块$\sigma_i=\{i,\zeta_\tau,\zeta_\sigma\}$,其内容分别为当前区块序号 i、记账者对当前区块的交易子块τ_i的签名ζ_τ和记账者对前一区块的签名子块σ_{i-1}的签名ζ_σ.

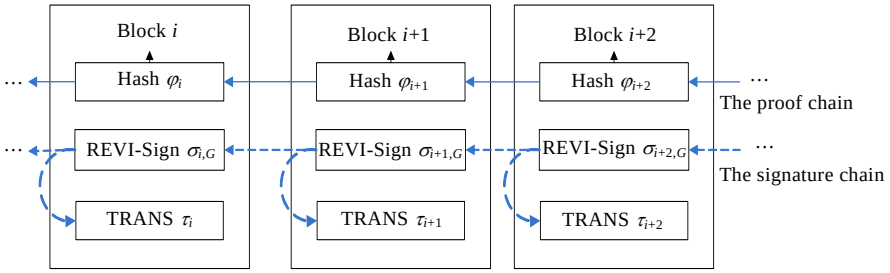


Fig.4 Revisable blockchain structure

图 4 可修改的区块链结构

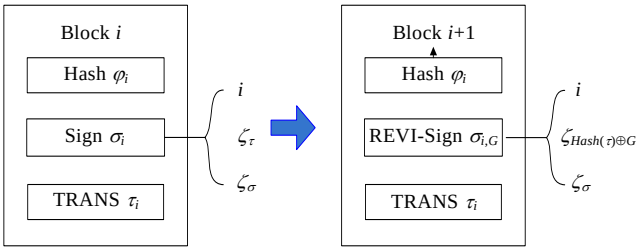


Fig.5 Structure comparison of SpaceMint blockchain and revisable blockchain

图 5 SpaceMint 区块与可修改区块结构对比

在可修改区块结构中,签名子块的内容变更为$\sigma_{i,G}=\{i,\zeta_{Hash(\tau)\oplus G},\zeta_\sigma\}$,其内容分别为:当前区块序号 i、记账者对交易子块τ_i的哈希值与机动因子 G_i异或结果的签名$\zeta_{Hash(\tau)\oplus G}$和记账者对前一区块签名子块$\sigma_{i-1,G}$的签名$\zeta_\sigma$.

我们称 G_i为签名子块$\sigma_{i,G}$对应的机动因子,由陷门单向函数构成,具体如下:

$$G_i: G_i(x_i^1,x_i^2,...,x_i^Q)=g_{P_i^1}(x_i^1)\|g_{P_i^2}(x_i^2)\|...\|g_{P_i^Q}(x_i^Q).$$

其中,Q=N×阈值比例,N 为系统总节点数,阈值比例设置将在第 2.2 节进行说明.另外,$g_{P_i^j}(j=1,2,...,Q)$表示基于矿工 P_i^j的陷门单向映射,$P_i^j(j=1,2,...,Q)$为区块 i 挖矿排名前 Q 位的各个矿工,$x_i^j(j=1,2,...,Q)$为矿工 P_i^j的专属随机数,初始化时由系统分配,作为公开参数被各节点记录在本地空间.超过阈值(N×阈值比例)节点同意时,随机数才能被修改.

SpaceMint 区块链系统使用签名子块承诺交易合法性,交易信息作为签名消息,与签名子块一一对应,交易信息改变,签名子块随之改变,区块的前后链接便被打破,因此交易数据无法修改.而本文提出的可修改区块链,在记账者对交易信息τ的签名中引入机动因子 G,签名消息变为 $Hash(\tau)\oplus G$,即使交易改变,亦可重构机动因子得到 G',保证签名消息与签名结果不变,实现除交易子块外,其余区块数据,即签名子块、证明子块和前后区块的所有信息均不变.在改变交易数据时,保证区块链结构完好,实现区块交易数据可修改.

在图 6 中,τ和τ'、MT_Root_τ和$MT_Root'_\tau$、ζ_τ和ζ'_τ、σ和σ'、G 和 G'、$\zeta_{Hash(\tau)\oplus G}$和$\zeta'_{Hash(\tau)\oplus G}$以及$\sigma_{i,G}$和$\sigma'_{i,G}$依次代表区块 i 交易修改前后的交易信息、交易信息的 Merkle Tree 根、记账者对交易信息的签名、传统签名子块信息、可修改区块链的机动因子、记账者关于交易信息哈希值与机动因子异或结果的签名以及可修改区块

链的签名子块信息.

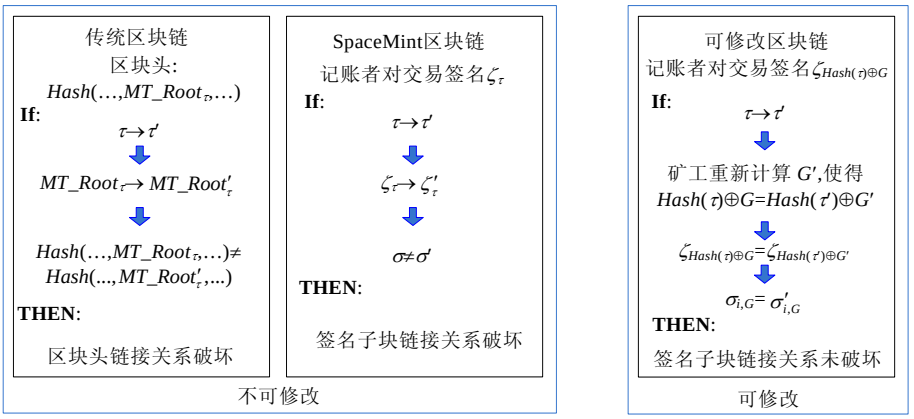


Fig.6 Comparison of transaction modification results under different block structures

图 6 不同区块结构下的交易修改结果对比

2.2 可修改的区块链原理

在可修改区块链中,超过阈值的节点同意时,可对上链数据进行符合系统利益的合法修改,修改操作不破坏区块链接结构.下面详细说明如何实现数据修改和保证修改操作合法.

2.2.1 如何实现数据修改

(1) 修改请求合法性验证

当区块链网络节点认为相关交易数据需要修改时,可向全网发送修改请求 *ReviseTx*,其合法性由区块 i 的修改群组验证.群组成员为区块 i 挖矿排名前 Q 位的各个矿工 $P_i^j(j=1,2,...,Q)$.当 Q 名矿工全部认可 *ReviseTx* 时,本次修改请求被视为合法,进入数据修改进程;否则请求将被驳回.通过合理的阈值设定(具体在第 2.2.2 节进行分析),修改群组意见可代表系统意志和利益,由修改群组进行修改请求的合法性验证是合理的.其中,$ReviseTx=\{id_{Block},reason,ctx_{old},ctx_{new}\}$,其内容依次为:修改交易所在的区块号、修改原因、待修改交易信息集合以及建议修改成的新交易信息集合.

(2) 数据修改操作实施

当修改请求合法时,区块 i 的修改群组即相应 Q 名矿工将实施数据修改操作.Q 名矿工将根据 *ReviseTx* 中的 ctx_{old} 和 ctx_{new},计算交易子块 τ_i 对应的 τ_i',即修改后的交易信息.接着,Q 名矿工重新计算一组新的随机数 $x_i'^1,x_i'^2,...,x_i'^Q$,得到全新的 G_i',使得 $Hash(\tau_i)\oplus G_i(x_i^1,x_i^2,...,x_i^Q)=Hash(\tau_i')\oplus G_i(x_i'^1,x_i'^2,...,x_i'^Q)$,保证签名子块 $\sigma_{i,G}$ 中的 $\zeta_{Hash(\tau)\oplus G}$ 不变,也即: $sign(Hash(\tau_i')\oplus G_i')=sign(Hash(\tau_i)\oplus G_i)=\zeta_{Hash(\tau)\oplus G}$.如此,交易数据改变并未引起签名子块的变化,无需对后续区块数据进行调整,维持了区块链接结构,是具有可操作性的区块数据修改.然后,Q 名矿工新随机数在系统更新.如第 2.1 节所述,$Q=N\times$阈值比例,因此,Q 名矿工满足随机数更新条件,更新合法;同时,矿工将交易子块 τ_i 变更为 τ_i',全网可随时验证数据的合法性.交易数据的合法修改正式完成.

(3) 系统状态更新

交易数据修改后,系统会进行相关状态更新,主要包括上述随机数及交易信息的更新;同时,为记录本次数据修改过程以供溯源,修改完成后,将生成一条对应的交易信息:

$$ctx=(revise,id_{Tx},t,id_{Block},reason,x_i^1,x_i^2,...,x_i^Q,x_i'^1,x_i'^2,...,x_i'^Q,ctx_{old},ctx_{new}).$$

其内容依次为:交易的类型标识符、交易号、生成时间、数据修改的区块号、修改原因、修改前、后相应矿工的专属随机数以及修改前、后的交易信息集合.特别地,ctx_{old} 和 ctx_{new} 为实际修改的交易集合,并非整个交易块数据,以便节约空间.若对数据的修改为删除操作,则 ctx_{new}=0.该交易将于后续挖矿中被打包上链.

数据修改总流程如图 7 所示.

- 第一,网络中出现区块 i 的交易数据修改请求,申请将交易子块 τ_i 变更为 τ_i';
- 第二,对应验证群组的矿工 $P_i^j(j=1,2,...,Q)$ 会对该次修改请求的合法性进行验证:请求若非法,则修改请求被驳回;反之,交易数据的修改正式进行.如第 2.2.1 节分析,为保证数据修改前后区块结构及内容不变,验证群组只能重新计算一组专属随机数作为机动因子 G_i 的入参,以配合交易数据的变化;
- 第三,验证群组在全网将自己重新计算的专属随机数进行更新,并正式将交易子块 τ_i 变更为 τ_i';
- 最后,用于记录该次修改操作的交易 *ctx* 生成并进入交易池.

至此,整个数据修改过程完成,全网可对修改后的数据进行验证.

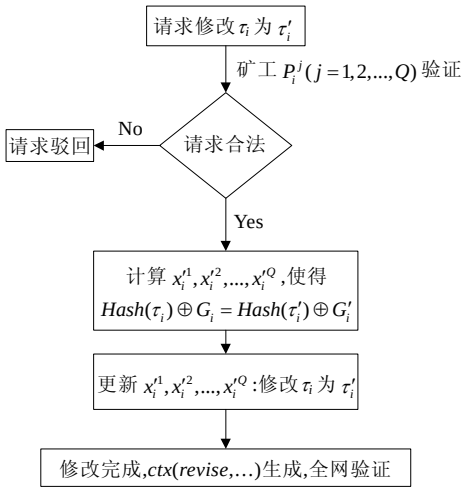


Fig.7 Process of data modification

图 7 数据修改流程

2.2.2 如何保证修改操作合法

“可修改”不会挑战区块链本身的安全性,它是对“不可篡改”造成的应用局限性的补充.因为修改操作必须代表系统意志与利益,也即合法才能被完成,否则修改后的区块数据无法通过全网验证,因此,“可修改”的存在仍可保证数据安全,具体由陷门单向函数特性及相关方案设计保证.

其核心思想如图 8 所示:首先,如第 2.2.1 节介绍,为保证数据修改前后区块 i 结构及内容不变,只能调整机动因子 G_i 的入参以配合交易数据 τ_i 的变化.而机动因子 G_i 由 Q 个陷门单向函数构成,陷门单向函数的性质决定只有拥有全部陷门,才能反向求逆,获取 G_i 的新入参,使 G_i 配合交易数据 τ_i 的变化做出正确调整.在本文中,陷门也即验证群组 Q 名矿工 $P_i^j(j=1,2,...,Q)$ 的私钥 $sk_i^j(j=1,2,...,Q)$.因此,只有 Q 名矿工同意数据修改(与方案数据修改条件一致),联合一起,才能在交易数据 τ_i 变为 τ_i' 时,利用各自私钥为机动因子计算全新入参 $x_i^j(j=1,2,...,Q)$,使得区块结构与内容不变的前提下,完成交易数据的改变.

具体而言,我们选择 ECC 加密算法[17,18]作为陷门单向映射 $g_{P_i^j}$,本质上,即以矿工 P_i^j 的公钥对其专属随机数 x_i^j 进行加密,即 $g_{P_i^j}=E_{P_i^j}^{ECC}(x_i^j)(j=1,2,...,Q)$, P_i^j 的私钥即单向映射函数的陷门.由陷门单向函数的性质可知,任何节点都可以根据公开参数计算得到 G_i,但计算 $x_i^j=D_{P_i^j}^{ECC}(g_{P_i^j})(j=1,2,...,Q)$,只能由拥有私钥(陷门)的矿工 P_i^j 才能完成.换言之,如果要计算新的随机数 $x_i'^1,x_i'^2,...,x_i'^Q$,使得交易数据 τ_i 变为 τ_i' 后,

$$Hash(\tau_i)\oplus G_i(x_i^1,x_i^2,...,x_i^Q)=Hash(\tau_i')\oplus G_i(x_i'^1,x_i'^2,...,x_i'^Q)$$

仍然成立,只能由 Q 名矿工联合才可完成.其中,$Q=N\times$阈值比例,阈值比例的设置需要兼顾方案安全性及效率:一

方面,阈值比例越大,交易数据修改权由更多的节点掌控,方案安全性越高,但是计算时间会越长,方案效率越低;另一方面,阈值比例越小,计算时间会越短,方案效率越高,但是交易数据修改权由更少的节点掌控,方案安全性越低.本文通过实验测试,权衡安全性与效率两方面,选定阈值比例为 80%,具体实验过程在第 3.3 节展开.

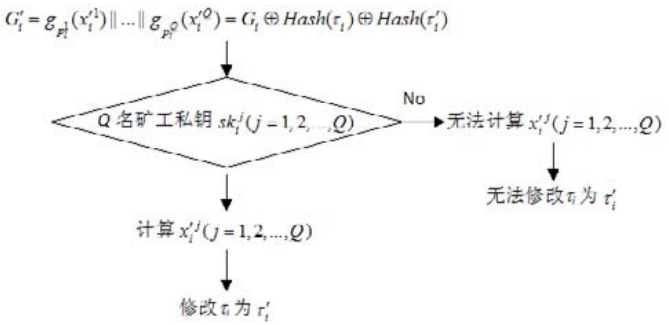


Fig.8 Q private keys (trap doors) ensure the validity of data modification

图 8 Q 个私钥(陷门)确保数据修改的合法性

2.3 数据修改安全性分析

可修改区块链实现了交易数据的合法修改,该操作不会破坏已有区块链结构,也不改变除修改数据外的其他区块数据.其安全性体现在修改操作的合法性,也即交易数据的修改操作是代表系统意志,符合系统利益的,恶意非法的修改操作无法完成.

为了说明可修改区块链方案的安全性,下面模拟恶意敌手在不符合系统利益时,企图将区块 i 的交易子块 τ_i 恶意修改为 τ_i',以谋取私利的过程.由方案设计可知,恶意攻击的条件为:修改区块的节点小于阈值 Q.如第 2.2 节分析,为了使修改后的数据被系统认可,数据必须符合原始区块链接结构.敌手可采取以下两种方法.

(1) 从当前区块 i 出发,向后依此调整每个签名子块数据,使 τ_i 修改为 τ_i' 后,前后区块链接关系仍然正确;

(2) 调整当前签名子块中的机动因子 G_i,使得交易字块变为 τ_i' 的同时,签名子块 $\sigma_{i,G}$ 保持不变.

对于方法(1),由于 τ_i 变为 τ_i',$\sigma_{i,G}$ 随之变为 $\sigma_{i,G}'$,因此,$\sigma_{i+1,G}$ 中 ζ_σ 需重新生成.以此类推,区块 i 后所有签名子块都需重新生成,从区块 i 开始后的整条链被颠覆,这需要耗费巨大的计算量,不具有现实操作性.

对于方法(2),为确保 $Hash(\tau_i) \oplus G_i(x_i^1, x_i^2, \ldots, x_i^Q) = Hash(\tau_i') \oplus G_i(x_i'^1, x_i'^2, \ldots, x_i'^Q)$ 成立,敌手需要对 Q 个陷门单向函数求逆,计算新的随机数组.由陷门单向函数的性质,敌手必须获取 Q 名矿工的陷门,才能完成陷门单向函数求逆,否则计算上不可行.而由恶意攻击条件可知,此时修改区块的节点数小于 Q,无法获取 Q 个陷门,修改操作无法完成.综上,如果要求修改的节点达不到规定的阈值,则不能进行区块数据的修改,可修改的区块链方案是安全有效的.

3 实验仿真

本节进行仿真实验,对 POSpace 共识下的挖矿、区块生成和交易数据修改过程进行模拟.挖矿节点用 Intel i5 Processor(2.4GHz,4G memory)模拟,并在 Visual Studio Ultimate 开发环境下,使用 C++语言编程实现.在区块生成与修改时,哈希函数、签名算法以及陷门单向函数分别使用 SHA-256,DSA-512 和 ECC-200 实现.

3.1 基于POSpace共识机制的记账权竞争

仿真实验基于图 9 的有向无环图,模拟基于 POSpace 共识机制的记账权竞争过程.

当前系统中序号 1~10 的矿工空间存储信息情况见表 1.

系统初始化时,矿工利用相关参数对有向无环图的各顶点标签值进行计算,并依据各自的存储空间大小进行最优存储,具体存储情况见表 1.无法进行全图存储的节点将利用已存储的顶点标签值计算未存储顶点的标

签值,实现对有向无环图的恢复.每一轮空间竞争中,系统将充当验证者向矿工充当的证明者发起挑战,记为$C(c_1,c_2,\ldots,c_k)$,也即顶点序号集合.证明者需返回C中各顶点对应的标签值.很显然,空间越大,存储的顶点越多,越能尽快返回结果.存储顶点数少的证明者只能牺牲时间,计算未存储的顶点标签值再返回.因此空间越大,竞争成功概率越高.以上便是基于空间竞争的挖矿过程,具体细节详见文献[10].本实验模拟多次竞争过程,记录不同挑战下各矿工的证明时间,部分结果如表 2 所示.

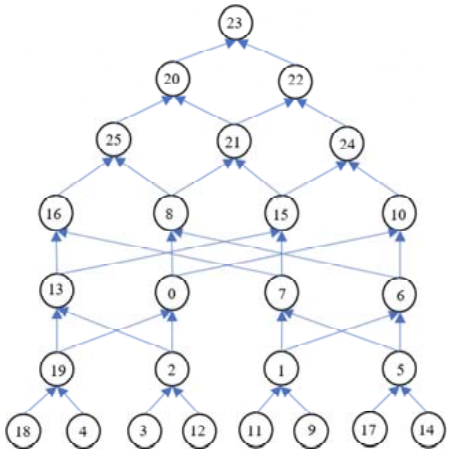


Fig.9 Directed graph of system

图 9 系统有向图

Table 1 Store information of the miner vertex in directed graph

表 1 矿工有向图顶点存储信息

矿工	存储有向图顶点号
1	18,4,3,12,11,9,17,14
2	18,4,3,12,11,9,17,14,19,2,1,5
3	18,4,3,12,11,9,17,14,19,2,1,5,6,7
4	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13
5	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13,15,16,8,10
6	18,4,3,12,11,9,17,14,19,2
7	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13,15,16
8	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13,15,16,8,10,25,21
9	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13,15,16,8,10,25,21,20
10	18,4,3,12,11,9,17,14,19,2,1,5,6,7,0,13,15,16,8,10,25,21,20,22,23

Table 2 Space proof time comparison of multiple miners

表 2 多名矿工空间证明时间比较

	C={20,7,21, 10,22}	C={19,15,22, 24,7,8,23, 0,20,5}	C={1,6,13,10,4, 0,25,12,11,22, 2,7,24,8,23}	C={18,4,3,2,1,5,13,0, 7,6,16,8,15,10,25, 21,24,20,22,23}	C={18,4,3,12,11,9,17,14,19, 2,1,5,13,0,7,6,16,8,15,10, 25,21, 24, 20, 22, 23}
1	2.521s	2.990s	3.131s	3.246s	3.377s
6	2.103s	2.417s	2.610s	2.747s	2.800s
2	1.157s	1.471s	1.607s	1.793s	1.803s
3	0.843s	1.124s	1.320s	1.504s	1.633s
4	0.510s	0.913s	1.007s	1.154s	1.243s
7	0.421s	0.680s	0.804s	1.000s	1.077s
5	0.253s	0.531s	0.660s	0.750s	0.857s
8	0.163s	0.430s	0.470s	0.720s	0.807s
9	0.127s	0.367s	0.390s	0.567s	0.687s
10	0.110s	0.307s	0.350s	0.443s	0.590s

由表 2 可以看出:本地空间越大、存储顶点数越多的证明者即矿工 10,返回证明结果的时间越短,成功挖矿概率越大.这符合 Pospace 共识机制下,区块记账权竞争的基本过程.

3.2 新区块的生成

本节模拟区块生成过程,在第 3.1 节模拟的空间竞争中胜出的矿工获得记账权,可发布相应区块.以区块 72 为例,其记账者为矿工 10,挖矿排名交易消息 ctx={*ranking*,90F2246A,A,9,8,5,7,4,3,2},Q/N=80%(80%的设置原因在第 3.3 节展开),N=10,Q=8.如第 2.1 节节介绍,区块 72 包含 φ_{72},$\sigma_{72,G}$ 和 τ_{72},其中,φ_{72},τ_{72} 分别为证明子块与交易字块,由实验测试得:

- φ_{72}=1c98228c2086fbd9d74b4645eba94d151d47fd9ecdcb0287729caf52b76906b8;
- τ_{72}={72,ctx},ctx 包含 25 条交易信息.

下面主要对新构建的$\sigma_{72,G}$ 生成过程进行介绍.$\sigma_{72,G}$={i,$\zeta_{(Hash(\tau)\oplus G)}$,$\zeta_{\sigma}$},$i$=72;

$\zeta_{(Hash(\tau)\oplus G)}$为记账者关于 $Hash(\tau)\oplus G$ 的 DSA 签名,其中,

$$G = G_{72}(x_{72}^1, x_{72}^2, ..., x_{72}^8) = g_{P_{72}^1}(x_{72}^1) \| g_{P_{72}^2}(x_{72}^2) \| ... \| g_{P_{72}^8}(x_{72}^8).$$

$P_{72}^1, P_{72}^2, ..., P_{72}^8$ 分别为矿工 10,9,8,5,7,4,3,2 的公钥,即挖矿排名前 8 位的矿工.实验中,$g_{P_{72}^j}(j=1,2,...,8)$ 使用 ECC-200 实现,当 $g_{P_{72}^j}(j=1,2,...,8)$ 输出长度为 200bit 时,G 的长为 200×8bit.为了使得每个 $g_{P_{72}^j}(j=1,2,...,8)$ 都参与 $Hash(\tau)\oplus G$ 结果的构建,防止 $Hash(\tau)\oplus G$ 结果被部分矿工掌控,取 $Hash(\tau)=Hash_{256}(\mu_1\|\tau_{72})\|...\|Hash_{256}(\mu_6\|\tau_{72})$,其中,$\mu_1,\mu_2,...,\mu_6$ 为随机数,剩余位补零,使得 $Hash(\tau)$与 G 长度相等.最终计算得:

$\zeta_{Hash(\tau_{72})\oplus G_{72}}$ =A2BF2998C76CD6415EEB8E102F4D72DEC0D38EC9;

ζ_{σ}则为记账者关于$\sigma_{71,G}$ 的 DSA 签名,ζ_{σ}=4B66BF0111FCD4841DD75621535870049130F182.

因此,

$\sigma_{72,G}$={72,A2BF2998C76CD6415EEB8E102F4D72DEC0D38EC9,
4B66BF0111FCD4841DD75621535870049130F182}.

随后,矿工 10 将以上数据打包成区块并发布,经过全网验证通过后,区块 72 正式上链,其余区块生成过程同理.图 10 为实验模拟生成的连续 3 个区块具体数据:

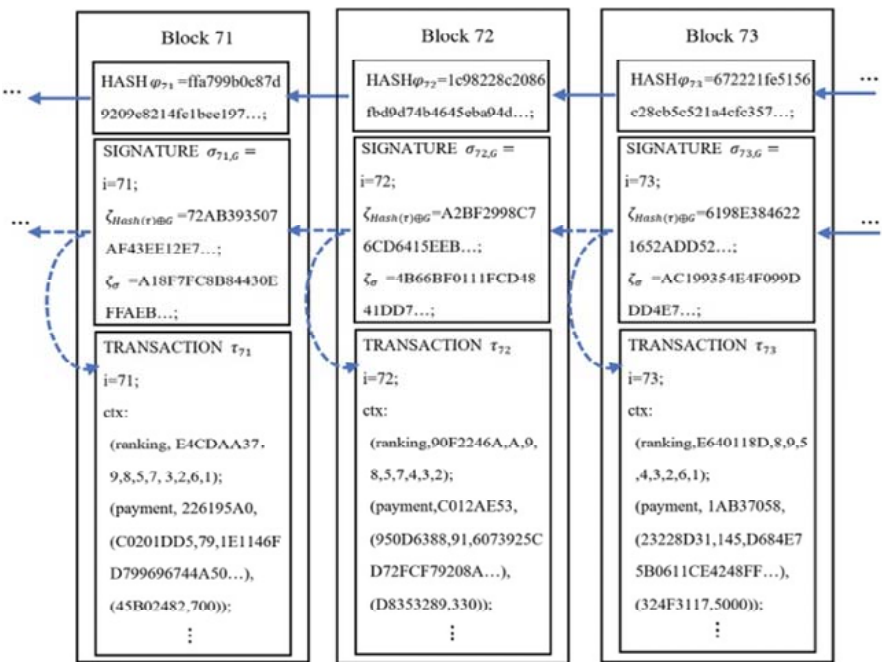


Fig.10 Original data of block 71~73

图 10 原始 71~73 区块数据

3.3 区块数据的修改

本节对上链交易数据的修改操作进行模拟.如图 10 所示,区块 72 上有一条交易:

“(payment,C012AE53,(950D6388,91,6073925CD72FCF79208A…),(D8353289,330))”.

假设后续有节点发现,曾经由于恶意节点发起双花攻击,区块 71~73 等所在的当前主链被另一条侧链颠覆,使系统在认可该笔交易后,又认可与该交易输入账户相同、输出账户不同的另一笔等额交易,使输入账户实现双花.为弥补该攻击造成的后果,可认为原本流入公钥为 D8353289 的受益方的金额又再次流向了另一个账户,也即该笔交易中受益方的实际收益并非 330,而是 0,该条上链交易数据需要修改.节点于 2019 年 3 月 13 日向全网广播一条交易修改请求:

ReviseTx={72,double spending,(payment,C012AE53,(950D6388,91,6073925CD72FCF79208A…),(D8353289,330)),(payment,C012AE53,(950D6388,91,6073925CD72FCF79208A…),(D8353289,0))}.

对应的矿工 10,9,8,5,7,4,3,2 收到该请求后,将对其合法性进行认证.发现合法,便根据 *ReviseTx* 和原始交易子块 τ_{72} 生成新交易子块 τ'_{72}.

进一步,根据 $Hash(\tau_{72})\oplus G_{72}(x_{72}^1,x_{72}^2,...,x_{72}^8)=Hash(\tau'_{72})\oplus G_{72}(x_{72}'^1,x_{72}'^2,...,x_{72}'^8)$ 求得 $G_{72}(x_{72}'^1,x_{72}'^2,...,x_{72}'^8)$.然后,8 名矿工使用各自私钥,做 ECC-200 解密计算:

$$x_{72}'^j=D_{P_{72}^j}^{ECC}(g'_{P_{72}^j})(j=1,2,...,8).$$

从而得到新的专属随机数: $x_{72}'^1,x_{72}'^2,...,x_{72}'^8$.

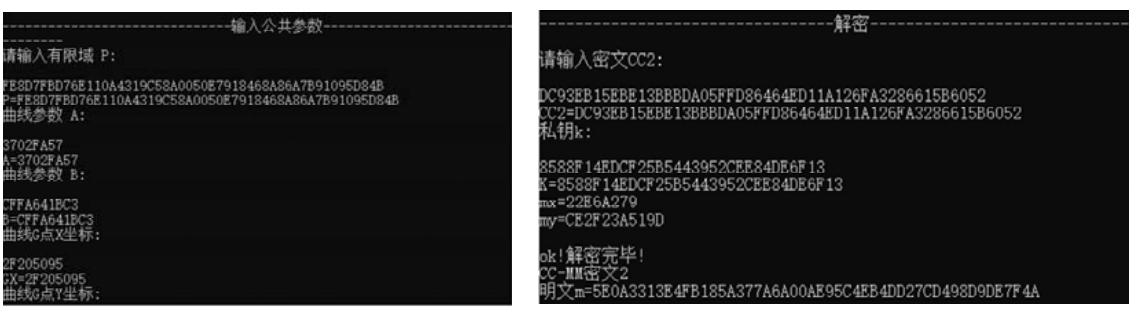
其中,

$$\tau'_{P_{72}^1}=\text{DC93EB15EBE13BBBDA05FFD86464ED11A126FA3286615B6052.}$$

矿工 10 利用私钥:973DBCD86EDE25F932599C1B79BC0953,对其进行 ECC-200 解密,求得:

$$x_{72}'^1=\text{5E0A3313E4FB185A377A6A00AE95C4EB4DD27CD498D9DE7F4A.}$$

编程运行如图 11 所示.



(a) Public parameters in ECC-200 (b) $x_{72}'^1$

Fig.11 Parameters calculation

图 11 参数获取

$x_{72}'^2,...,x_{72}'^8$ 求解同理,矿工随后在全网更新随机数.由陷门单向函数性质可知,同时拥有 8 名矿工的私钥才可完成.最后,8 名矿工将区块 72 的交易子块由 τ_{72} 变更为 τ'_{72},并生成相应的修改记录,作为一条溯源交易信息放入交易池,以供后续矿工打包上链,该交易消息为

ctx=(*revise*,531694D0,20190313,72,double spending,(15347EFD0B…,204D562128…,19A87E1029…,3959191C54…,2F75D24A42…,411BE1FE2D…,3651CB06D6…,2515C96F5…),(5E0A3313E4…,CE441CDD19…,C9E73B440E…,1F8B5086DF…,971E915B1A…,B3C28D3E58…,F2648076ED…,D569C22260…),(payment,C012AE53,(950D6388,91,6073925CD72FCF79208A…),(D8353289,330)),(payment,C012AE53,(950D6388,91,6073925CD72FCF79208A…),(D8353289,0))).

至此,整个交易数据修改过程全部完成,修改后区块数据如图 12,黑体部分为改变后的交易数据.验证可知:各区块间链接关系及区块结构不变的同时,交易数据已按照要求修改,合法的数据修改操作完成.

最后,我们对方案效率进行分析.如第 2.2 节所述,阈值比例的设置需权衡方案安全性及效率,阈值比例越高,修改权由更多节点掌控,安全性越高;但修改参与节点越多,耗时越久,方案效率越低.我们对不同阈值比例下的区块生成及修改耗时进行测试,结果如表 3 所示.其中,为使修改权由系统大多数节点控制,阈值比例应大于 50%,因此,合法阈值集合为:{60%,70%,80%,90%,100%}.

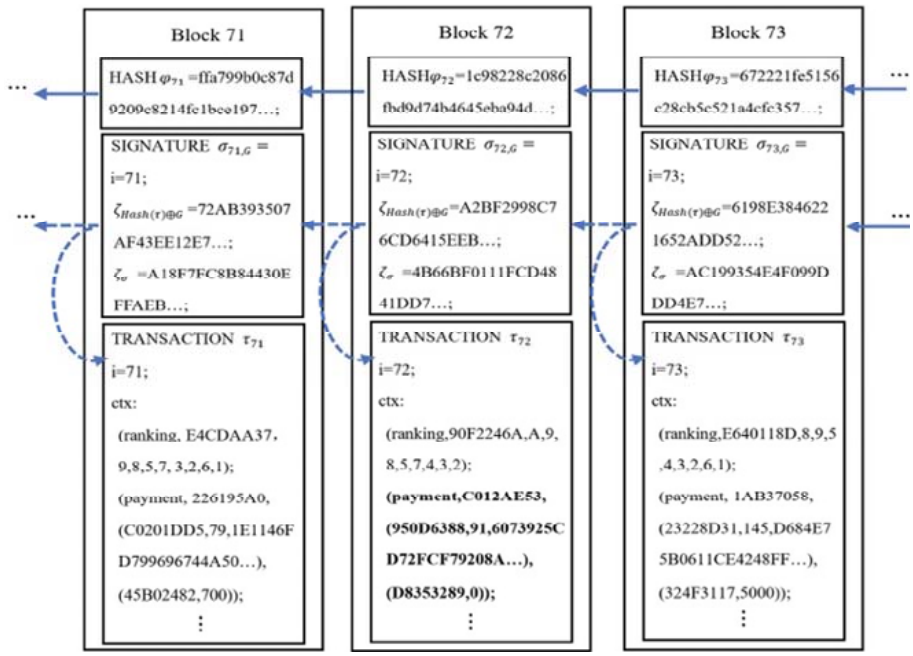


Fig.12 Data of block 71~73 after revision

图 12 数据修改后的 71~73 区块数据

Table 3 Time comparison of block generation and modification under different threshold ratio

表 3 不同阈值比例下区块生成和修改耗时对比

阈值	60%	70%	80%	90%	100%
生成区块耗时(s)	3.643	3.742	3.841	3.940	4.039
修改区块耗时(s)	1.068 8	1.115 6	1.198 4	1.615 2	1.980 0
修改/生成耗时占比(%)	29.34	29.81	31.20	40.99	49.02

由表 3 可知:

- 不同阈值比例下,生成区块耗时两两大约相差 0.1s,占比小于 3%;
- 但修改区块耗时上,阈值比例在 60%~80%时,两两相差小于 0.1s,修改与生成区块的耗时占比较为接近,在 30%左右;
- 而阈值比例在 80%~100%时,两两耗时差大幅增加,约在 0.4s 左右,且修改与生成区块耗时占比超过 40%,对方案效率影响显著.

因此,当阈值超过 80%时,修改区块耗时明显增加,耗时占比超过 40%,方案效率不高;而阈值比例为 60%,70%时,方案效率与 80%相差不大,但安全性不及后者.

因此,权衡方案安全性与执行效率,最佳阈值比例设为 80%.

在表 4 中,我们给出了阈值比例为 80%时,区块生成及修改的具体耗时.由表 4 可知,区块生成和修改的平均

耗时分别为 3.8404s 和 1.1984s,区块修改耗时不超过区块生成耗时的 1/3,具有可操作性.

Table 4 Time of block generation and modification under the threshold of 80%

表 4 阈值比例为 80%时区块生成和修改耗时

耗时(s)	区块 1	区块 2	区块 3	区块 4	区块 5	平均
生成区块	3.826 2	3.833 6	3.841 4	3.852 4	3.848 5	3.840 4
修改区块	1.198 5	1.146 7	1.158 1	1.262 4	1.226 4	1.198 4

4 结 论

本文在 POSpace 共识机制下,基于陷门单向函数,提出了可修改的区块链方案.通过引入机动因子,重构区块的签名子块,在区块数据需要修改时,只要特定阈值数的节点同意,便可实现区块数据的合法修改,且不破坏区块的链接结构,全网仍可按原始验证方式对数据合法性进行验证.仿真实验表明:区块修改不超过区块生成耗时的 1/3,具有可操作性.同时,阈值的设定使得恶意的非法修改无法完成,保证了数据修改的安全性.因此,可修改的区块链可兼顾数据修改的安全性与执行效率,使区块链系统更加完善,适用性更强.

References:

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. 2009. https://bitcoin.org/bitcoin.pdf

[2] Li XQ, Jiang P, Chen T, Luo XP, Wen QY. A survey on the security of blockchain systems. Future Generation Computer Systems, 2017. [doi: 10.1016/j.future.2017.08.0200167-739X]

[3] Yuan Y, Wang FY. Blockchain: The state of the art and future trends. Acta Automatica Sinica, 2016,42(4):481−494 (in Chinese with English abstract).

[4] Garay J, Kiayias A, Leonardos N, *et al*. The Bitcoin backbone protocol: Analysis and applications. In: Proc. of the 34th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Springer, 2015. 281−310.

[5] King S, Nadal S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Self-Published Paper, 2012.19.

[6] Larimer D. Transactions as proof-of-stake. 2013. http://7fvhfe.com1.z0.glb.clouddn.com/@/wpcontent/uploads/2014/01/TransactionsAsProofOfStake10.pdf

[7] Aggelos K, Alexander R, Bernardo D, *et al*. Ouroboros: A provably secure proof-of-stake blockchain protocol. In: Proc. of the 37th Annual Int'l Cryptology Conf. Springer-Verlag, 2017. 357−388.

[8] BitShares Blockchain Foundation. The bitshares blockchain. 2014. https://github.com/bitshares-foundation/bitshares.foundation/blob/master/download/articles/BitSharesBlockchain.pdf

[9] Fabian S, Daniel L. Bitshares 2.0: Financial smart contract platform. 2015. https://www.weusecoins.com/assets/pdf/library/Bitshares%20Financial%20Platform.pdf

[10] Park S, Kwon A, Fuchsbauer G. SpaceMint: A cryptocurrency based on proofs of space. In: Proc. of the 22nd Int'l Conf. Springer, 2017.

[11] Krawczyk H, Rabin T. Chameleon hashing and signatures. US Patent 6108783, 2000-08-22.

[12] Li PL, Xu HX, Ma TJ. Research on fault-correcting blockchain technology. Journal of Cryptologic Research, 2018,5(5):501−509 (in Chinese with English abstract).

[13] Ren YL, Xu DT, Zhang XP, *et al*. Delegable blockchain based on an threshold ring signature scheme. Journal on Communications, 2019,40(4):71−82 (in Chinese with English abstract).

[14] Jacques P, Louis G. Trapdoor one-way permutations and multivariate polynomials. In: Proc. of the First Int'l Conf. on Information and Communications Security, Vol.1334. Springer-Verlag, 1997. 356−368.

[15] Diffie W, Hellman M. New direction in cryptography. IEEE Trans. on Information Theory, 1976,22(6):644−654.

[16] Eyal I, Sirer EG. Majority is not enough: Bitcoin mining is vulnerable. Communications of the ACM. ACM, 2018,61(7):95−102.

[17] Koblitz N. Elliptic curve cryptosystems. Mathematics of Computation, 1987,48:203−209.

[18] Gu ZL, Zheng SH, Yang YX. Modern Cryptography. 2nd ed., Beijing: Beijing University of Posts and Telecommunications Press, 2015. 190−207 (in Chinese).

附中文参考文献:

[3] 袁勇,王飞跃.区块链技术发展现状与展望.自动化学报,2016,42(4):481−494.

[12] 李佩丽,徐海霞,马添军.可更改区块链技术研究.密码学报,2018,5(5):501−509.

[13] 任艳丽,徐丹婷,张新鹏,等.基于门限环签名的可删除区块链.通信学报,2019,40(4):71−82.

[18] 谷利泽,郑世慧,杨义先.现代密码学教程.第 2 版,北京:北京邮电大学出版社,2015.190−207.

任艳丽(1982－),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为公钥密码学,可验证外包计算,区块链安全.

徐丹婷(1994－),女,硕士生,主要研究领域为可验证外包计算,区块链安全.

张新鹏(1975－),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为多媒体信息安全,信息隐藏,数字取证,图像处理.

谷大武(1970－),男,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为密码分析与设计,信息分析与密码工程,计算机安全体系结构.