软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, [doi: 10.13328/j.cnki.jos.007391]

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

基于多父链辅助工作量证明共识机制的后量子区块链系统 *

王御天, 贾舒瑞, 陈铭希, 董怡帆, 杨亚芳

(复旦大学 计算机科学技术学院,上海 200433)

通讯作者: 王御天, E-mail: 23110240140@m.fudan.edu.cn

摘 要: 随着量子计算机的发展,对于以传统椭圆曲线数字签名为基石的公链会造成颠覆性安全问题,常见解决方案是将后量子数字签名算法应用到区块链系统中.对于采用工作量证明共识机制的区块链公链,支持算力也是公链安全的重要基石,如何节省能源且最大化算力支持是一个重要研究方向.本文提出一种算力多元化且应用自主可控后量子签名的后量子区块链系统.

Dilithium 签名方案是美国 NIST 所推荐的首选和通用后量子签名标准,其安全性基于 power-of-two 分圆环上的 MLWE 和 MSIS 问题.但是,正如比特币区块链虽然最初采用 EC-DSA 标准签名算法,但并没有采用美国 NIST 所规定的椭圆曲线一样,power-of-two 分圆环丰富的代数结构为公链所基于的后量子数字签名的长远安全带来较大风险和不确定性.素阶数域是一种代数结构更少、更为保守和安全的后量子格基密码技术路线.在本文中,我们采用基于素阶数域的后量子数字签名 Dilithium 变体: Dilithium-Prime,来作为后量子区块链系统的签名算法以提供高置信度的交易签署后量子安全.

为提供多元化的算力以最大化后量子公链的算力支持,并解决目前矿池和矿工收入不断减少的困境,我们引入一种基于多父链辅助工作量证明共识机制,可以请求所有采用 Sha256 和 Scrypt 哈希计算的算力来辅助共识而不额外增加现有矿工和矿池的工作量, 在增加了后量子区块链的算力来源的同时也提高了现有矿池和矿工的算力利用率. 同时提出适配这种多父链辅助工作量共识机制的区块和交易结构和难度调整算法,针对不同量级的算力,稳定出块比例和出块时间,并可有效应对算力突增突减等极端情况攻击以保持系统的健壮性.

关键词: 区块链;后量子密码;共识机制;辅助工作量证明

中图法分类号:

中文引用格式: 王御天,贾舒瑞,陈铭希,董怡帆,杨亚芳. 基于多父链辅助工作量证明共识机制的后量子区块链系统. 软件学报. http://www.jos.org.cn/1000-9825/7391.htm

英文引用格式: Wang YT, Jia SR, Chen MX, Dong YF, Yang YF. Post-quantum Blockchain System Based on Multi-parent Chain Auxiliary Proof-of-work Consensus Mechanism. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7391.htm

Post-quantum Blockchain System Based on Multi-parent Chain Auxiliary Proof-of-work Consensus Mechanism

WANG Yu-Tian, JIA Shu-Rui, CHEN Ming-Xi, DONG Yi-Fan, YANG Ya-Fang

(School of Computer Science, Fudan University, Shanghai 200433, China)

Abstract: With the development of quantum computers, the public blockchain based on traditional elliptic curve digital signature will face disruptive security issues. The common solution is to apply post-quantum digital signature algorithms to blockchain systems. For the public blockchain adopting proof-of-work consensus mechanism, supporting computing power is also an important cornerstone of public

* 基金项目: 国家重点研发计划基金资助项目（2022YFB2701601）; 上海市协同创新基金资助项目（XTCX-KJ-2023-54）; 上海市科委区块链关键技术攻关专项基金资助项目（23511100300）

收稿时间: 2024-06-30; 修改时间: 2024-09-05; 采用时间: 2024-12-30; jos 在线出版时间: 2025-01-20

blockchain security. How to save energy and maximize computing power support is an important research direction. This article proposes a post-quantum blockchain system with diversified computing power and independent post-quantum signature.

The Dilithium signature scheme is the preferred and universal post-quantum signature standard recommended by the NIST, and its security is based on the MLWE and MSIS problems on the power-of-two division ring. However, just as the Bitcoin blockchain initially adopted the EC-DSA standard signature algorithm but did not adopt the elliptic curve specified by the US NIST, the rich algebraic structure of the power-of-two cyclotomic rings poses greater risks and uncertainties for the long-term security of the post-quantum digital signatures on which the public blockchain is based. Large-Galois-group prime-degree prime-ideal field is a more conservative and secure post-quantum lattice-based cryptographic technology route with fewer algebraic structures. In this article, we adopt a Dilithium variant based on large-Galois-group prime-degree prime-ideal field: Dilithium-Prime, as the signature algorithm for the post-quantum blockchain system to provide high-confidence transaction signing post-quantum security.

To provide diversified computing power to maximize the computing power support of the post-quantum public blockchain and address the current dilemma of declining mining pool and miner income, we introduce a multi-parent chain auxiliary proof-of-work consensus mechanism that can request all computing power using Sha256 and Scrypt hash calculations to assist in consensus without adding additional work to existing miners and mining pools. This increases the source of computing power for the post-quantum blockchain and also improves the utilization rate of existing mining pools and miners. At the same time, we propose a block and transaction structure and difficulty adjustment algorithm adapted to this multi-parent chain auxiliary proof-of-work consensus mechanism, which can stabilize the block production ratio and block production time for different magnitudes of computing power, and effectively responding to extreme cases such as sudden increases or decreases in computing power to maintain the robustness of the system.

Key words: blockchain; post-quantum cryptography; consensus mechanism; auxiliary proof-of-work

区块链技术最早由中本聪在 2008 年提出[1],以数字签名算法、共识机制等为基础,具有去中心化、不可篡改等性质,广泛应用于数字资产管理、跨境支付、数据共享等方面.其中,比特币作为目前认可程度最高的公链,有数量众多的用户和一定的应用场景.同时,也出现了采用比特币相同的架构与共识算法的莱特币等公链.

以比特币为代表的这类区块链公链应用椭圆曲线数字签名算法,在交易过程中用户通过私钥签名并提交数据和签名到链上的方式,有效保证区块链数据的安全性和正确性.但是,随着量子计算机的出现,椭圆曲线数字签名算法等传统密码的安全性不能在量子时代得到保证,进而对区块链系统的安全性会产生严重影响.

此外,比特币等区块链系统中采用工作量证明（Proof-of-Work）共识机制[1],算法核心思想是计算满足一定难度的哈希值,计算成功的节点具有出块权.工作量证明共识机制具有去中心化程度高等优点,但也有利用率低、消耗能源多等缺点.随着比特币在 2024 年 4 月的第四次减半,大部分现存的矿池和矿工的收益率大幅下降,迫切希望在不额外增加算力投入的前提下实现收入的多元化.在量子计算机出现后,也需要更多的算力提高区块链的安全性.

同时,在共识中需要通过难度调整算法使得区块平均出块时间尽可能维持在目标值,以此保证区块链系统的稳定性.而难度调整算法因为调整周期过长,难以对全网算力的突然变化及时做出调整,算力的突增突减都会对系统的稳定性产生影响.

针对上述背景和问题,本文在比特币类的采用工作量证明共识算法和 UTXO 脚本的公链基础上设计了使用后量子数字签名算法的支持多种算力共识的区块链系统,主要具有以下三点贡献:

(1)设计并实现多父链辅助工作量证明共识机制(Mul-AuxPoW),该共识机制使其它在工作量证明中采用 Sha256 和 Scrypt 哈希算法的公链作为父链提供算力支持.优势是复用父链为了共识消耗的能源,子链本身并不需要额外的算力和能源消耗,同时也可以解决现存矿池和矿工收益率日益减少的困境,使得他们在不额外增加算力和能源消耗的前提下实现更多收入.

(2)基于多父链辅助工作量证明共识机制设计了难度调整算法,算法支持不同算力来源的父链出块数保持在一定比例以激励小算力.算法通过对于出块边界值的计算,更加灵活调整难度,有效应对算力突增突减的情况.

(3)在区块链中应用基于素阶数域的高效数字签名方案：Dilithium prime 算法[6],相比于基于 power-of-two 分圆环的 Dilithium 后量子签名标准,其代数结构更少、置信度更高.在算法应用中,对区块链的交易脚本扩容,适配后量子数字签名算法.

本文第 1 节介绍工作量证明共识机制以及后量子签名在区块链系统应用的相关工作.第 2 节介绍架构以及相关实现细节.第 3 节对多父链辅助工作量证明共识机制(Mul-AuxPoW)以及难度调整算法的安全性与正确性进行理论分析以及效率分析.第 4 节对系统进行仿真实验和评估.第 5 节总结全文.

1 相关工作

1.1 后量子签名算法在区块链中应用

1994 年,Shor 算法提出[2],可以在多项式时间解决离散对数问题和大整数分解问题,而这对应的是,对于量子计算机,目前的密码算法将受到安全性挑战.2016 年,美国国家标准技术研究所进行后量子密码征集项目[3].其中密码方案包含基于格、基于编码、基于哈希等方案.其中基于格的后量子密码算法,包含 CRYSTALS-Dilithium[4]、Falcon[5]等数字签名算法.

与其他类型相比,基于格的后量子数字签名,例如 CRYSTALS-Dilithium,速度一般强于其它类型后量子数字签名,但是空间开销有所增大.董怡帆等人改进 Dilithium 方案[6],提出了基于素阶数域的数字签名方案,记为 Dilithium-Prime.与 Dilithium 相比, Dilithium-Prime 具有更小的内存开销,更强的签名效率,同时安全性也有所提高.

对于后量子签名算法在区块链中的应用也有一些探索.例如,对于基于格的后量子签名算法, Torres 等人设计了一种基于格的环签名并应用到门罗币中[7];Arielcoin 在基于 PoW 的公链系统中应用了 CRYSTALS-Dilithium 签名算法[8],主要使用了 Dilithium3 参数;Hcash 采用 DAG（有向无环图）的分布式结构[9],PoW 和 PoS 混合,使用基于环 LWE 的公钥加密方案.

在基于其他的的后量子签名算法中,也有一些尝试应用在共识机制中.例如 ABC 链应用 Rainbow 签名算法[10],使用 PoW 共识机制,但是基于多变量的 NP-hard 问题;QRL 使用基于散列的前向安全签名方案[11],使用 WOTS+作为主要构建块;Mochimo 同样使用 XMSS+[12],又提出并验证一次性签名的 WOTS+变体.

1.2 工作量证明共识机制

区块链中的共识机制用于参与节点以何种方式对区块数据达成一致[13],即选举出块者、生成区块、同步区块的流程.在公链系统中,比特币(BTC)、莱特币(LTC)等使用工作量证明(PoW)共识机制,PoW 共识机制利用单向哈希函数计算,当节点计算满足一定难度的哈希值就获得出块权.其中 nounce 是遍历用来满足公式的值,Blockheader Hash 是区块头的哈希值,Target 是目标值,与难度成反比.

$$Hash(nounce+BlockheaderHash)<Target$$

随着发展,PoW 共识机制因为具有去中心化强、安全程度高等优点,被广泛应用.BTC 和 BCH 等在共识机制中使用 Sha256 哈希函数,算力自 2012 年 10TH/s 增加到如今 500EH/s 左右,占据着目前绝大多数的算力.LTC 和 Dogecoin 等在共识机制中使用 Scrypt 哈希函数,算力目前达到 1PH/s,此外,为了增加算力的应用率,2014 年 LTC 进行硬分叉,支持辅助工作量证明共识机制(AuxPoW,Auxiliary Proof of Work).

辅助工作量证明共识机制[14]在 PoW 的基础上进行改进,达到同时进行两条区块链共同共识的效果,其中需要进行辅助工作量共识的区块链称为子链,帮助子链进行共识的区块链称为父链.但是目前的辅助工作量证明共识机制只支持相同的哈希函数,例如 LTC 和 Dogecoin 都使用 Scrypt 哈希函数.

1.3 难度调整算法

在工作量证明共识机制中,由于算力随时发生变化,为了维持出块时间保持稳定,需要在一定周期内调整难度,一般认为 Target 越大,计算出结果更容易,难度更小,反之 Target 越小,难度更大.在比特币中,采用以下公式调整难度[15],其中 actualTimespan 表示一定出块数的实际出块时间,PoWTargetTimespan 表示一定出块数的目标出

块时间.

$$newTarget = prevTarget \times \frac{actualTimespan}{powTargetTimespan}$$

同时限制每次难度调整值倍数范围在 4 以内,也就是新的难度值满足以下不等式:

$$0.25 * prevTarget \le newTarget \le 4 * prevTarget$$

难度调整算法往往具有滞后性,需要在一定时间周期内进行调整,而算力变化非常快,常见的有跳矿攻击(pool-hopping attack)[16],即共识时矿工选择利益更高的区块链进行共识,而当该链利益不足时则共识另外一条链,所以出块算力并不稳定.最基本的攻击方式有两种:第一种是在一定周期内期增加算力,使 actualTimespan 减少,难度公式会增加难度,由于难度增加后可能导致利益不足,进行跳矿攻击,在之后周期算力减少,而难度增加会使得这个周期时间增加,影响区块链出块效率.第二种是在一定周期内减少算力,使得 actualTimespan 增加,难度公式会减少难度,由于难度减少后,进行跳矿攻击,在之后周期算力增加,攻击者获得额外收益且难度减少会使得这个时间减少,降低区块链稳定性.

2 区块链系统设计

本节从 4 方面介绍系统细节,分别是系统架构,多父链辅助工作量证明共识机制、难度调整算法和后量子签名算法应用.在系统架构方面,通过样例介绍系统中区块的生成、共识和难度调整的流程.在多父链辅助工作量证明共识机制部分,从矿池、父链与子链三方交互的共识流程进行介绍.在难度调整部分,将从算法流程、算法应用进行介绍.在后量子签名算法应用过程中,将从密钥生成、交易签名、区块扩容等方面介绍修改流程.

2.1 系统架构

基于多父链辅助工作量证明共识机制的后量子签名区块链和目前常见区块链系统架构基本相同,如图 1 所示,共包含存储层、网络层、共识层、协议层和应用层,从下至上具体含义为:

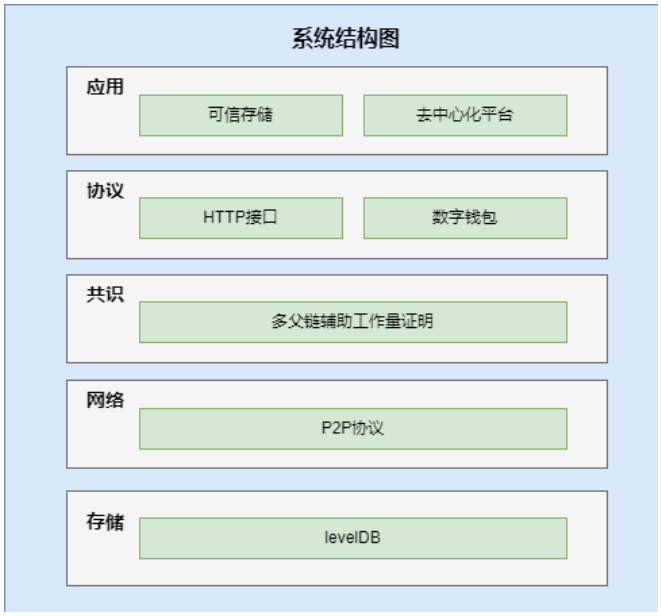


图 1 系统架构图

存储层:重新定义了区块数据结构,在底层使用 levelDB 数据库,存储了采用由 Sha256 哈希函数和 Scrypt 哈希函数共识的区块和 Dilithium-Prime 后量子数字签名算法的交易.

网络层：使用 Peer-to-Peer（P2P）协议,节点之间使用点对点的方式直接通信,不需要中心化的服务器,P2P 协议提供了节点之间的连接和消息传递机制.

共识层：采用多父链辅助工作量证明共识机制.对于多父链的辅助工作量证明共识机制,以使用 PoW 区块链系统算力较大的两类哈希函数作为父链,对两类父链设置不同难度,根据我们设定的一些高效、合理、公平的原则,用一种全新的难度调整算法自动调整对应父链的难度.

协议层：用于接收交易和返回结果的 HTTP 接口,也有数字钱包等用于区块链系统使用、升级的模块.

应用层：一般是指在区块链系统中的应用,包含可信存储等多种功能.

2.2 多父链辅助工作量证明共识机制

共识机制首先遵循“最长链协议”,与其他使用 PoW 共识机制的区块链类似,核心思想是每个节点总是选择并尝试扩展代表最多工作量证明的区块链.同时为了避免分叉,设置目标区块出块时间且增加区块大小,尽可能减少分叉.如果发生分叉,将根据最长链协议,选择区块高度最大的链.

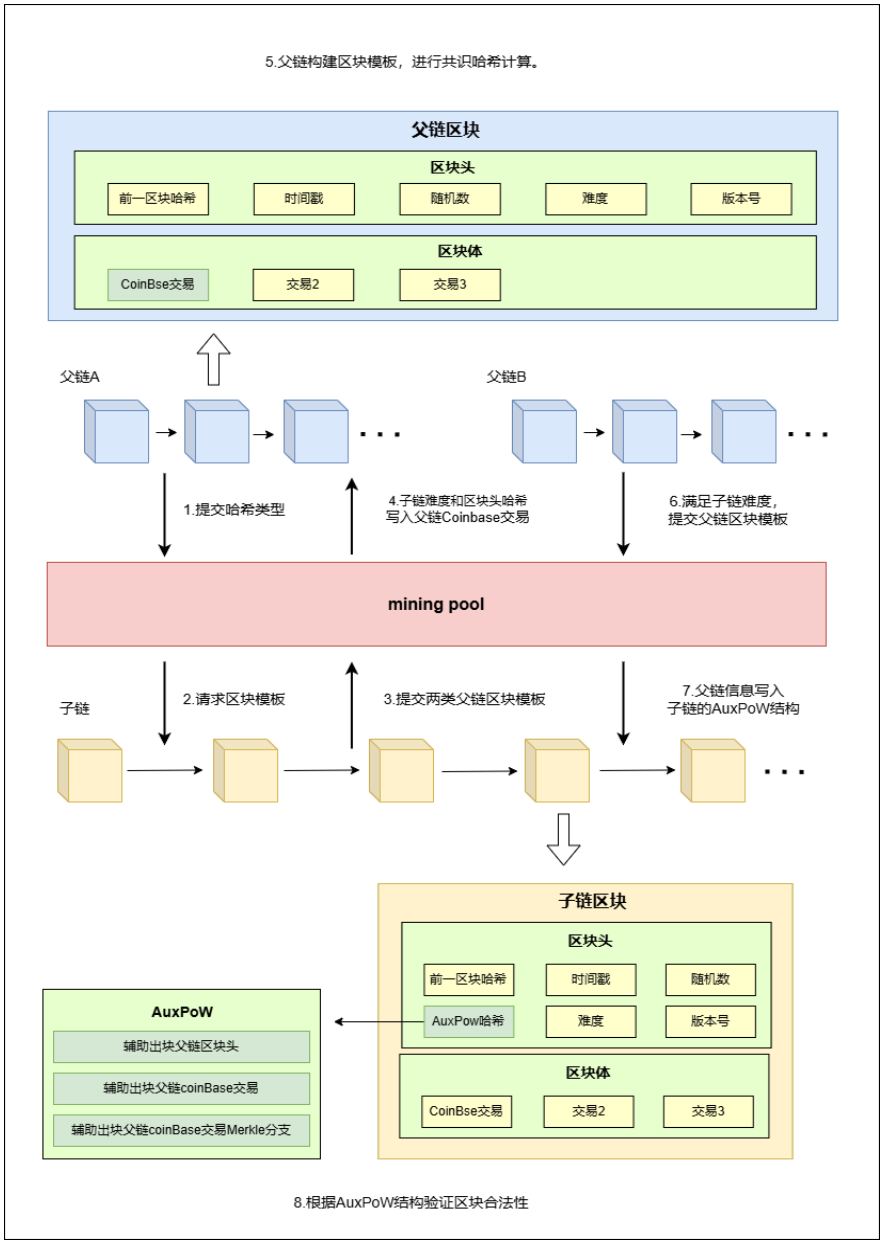
多父链辅助工作量证明算法(Multi-parent Chain Auxiliary Proof-of-work Consensus Algorithm,简称 Mul-AuxPoW)可以使所有在共识机制中采用 Scrypt 和 Sha256 哈希算法的区块链作为父链,为子链提供算力辅助支持计算.因为目前共识机制主要算力集中在矿池中,为了适应这一特点,设计了多父链、矿池和子链模型,具体流程如图 2 所示：

算法共包含四个阶段,第一阶段是请求多父链辅助共识.在共识机制中采用 Scrypt 和 Sha256 哈希算法的父链将类型提供给矿池,矿池根据父链的类型向子链请求包含不同难度的区块模板,矿池将对应类型父链的子链区块模板里的哈希值和难度写入父链的 CoinBase 交易(区块的铸币交易,一般不包含区块链的交易信息),构建父链的区块模板进行共识.

第二阶段是辅助共识阶段.父链构建区块模板后,此时父链有两个难度值,第一个难度值是父链区块头中用于父链出块的难度值,第二个难度值是 CoinBase 交易里用于子链辅助公式的难度值.父链共识节点会在共识的哈希计算中计算两个难度值是否满足.如果满足子链难度值,向子链提交工作;如果满足父链本身难度要求,父链节点拥有出块权,再继续构建出块模板.

图 2　多父链辅助工作量证明共识机制流程图

第三阶段是提交共识阶段.矿池将满足子链难度要求的父链区块或区块模板提供给子链,包含以下信息:



父链区块头、CoinBase 交易的默克尔分支和 CoinBase 交易字段.子链修改了数据结构,保存信息到子链区块头 hsahAuxPoW 字段指向的 CAuxPoW 字段.

第四阶段是验证工作阶段.子链保存父链三部分信息后,对信息进行存储和验证：（1）验证 CAuxPoW 字段的 CoinBaseTx 中的子链区块头哈希是否与子链提交给矿池的区块头哈希和难度值相同.这一步用于验证子链提交的信息是否与父链中的 CoinBase 一致.（2）验证 CAuxPoW 字段的 CoinBaseTx 和对应的 merkleBranch 计算出块父链区块交易默克尔树根哈希是否与 hashBlockHeader 中的根哈希值相同.这一步用于验证存储正确信息的 CoinBase 交易是否在出块父链中.（3）根据 CAuxPoW 字段的 hashBlockHeader 中的 nVersion 字段判断该工作来自哪类父链,选择对应提交的难度值对 hashBlockHeader 进行对应父链的哈希验证,判断是否满足难度要求.

2.3 难度调整算法

目前算力主要都集中在 BTC、BCH 等使用 Sha256 哈希函数以及 LTC、DOGE 等使用 Scrypt 哈希函数的区块链中.但是两类链的算力差距仍比较大且比较集中在一些成熟的区块链中, 为了满足吸引算力小的区块链的加入,同时不降低大算力链的积极性,并根据实际情况需要子链满足以下三点需求：

（1）同其他使用工作量证明共识机制的区块链相同,难度调整算法需要稳定子链在ET时间内出块数恒定在φ左右,（ET与φ为常数）.

（2）为了保护相对算力较小的父链 B,算力较大的父链 A（使用 Sha256 哈希函数类父链）与父链 B（使用 Scrypt 哈希函数类父链）出块比例基本保持在α上下（α为常数）.

（3）抵抗算力突然变化对出块时间稳定性的影响.

根据这三点需求设计了难度调整共识机制,设两种类型的父链根据各自在周期内的出块数x_n、y_n和时间,计算两类父链第 n 轮难度调整周期的难度值H_{A_n}、H_{B_n}. 在比特币难度调整算法的基础上,增加了两个系数因子分别表示算力变化对难度的影响和两类父链出块比例.

首先根据临近ω个周期中,父链A 出块的数量$x_{n-\omega+1}\cdots x_{n-1}$、$x_n$、父链 B 出块数$y_{n-\omega+1}\cdots y_{n-1}$、$y_n$,计算各自出块变化的加权平均值$x_A, y_B$.为反映两类链算力的变化,使用指数是$L$ 类链第 $n-1$个周期与第 n 个周期出块差值和L 类链前ω周期出块的变化值的加权平均比值的比,底数是δ_T,用T_{A_n}, T_{B_n} 表示,即：

$$T_{A_n} = \delta_T^{\frac{x_{n-1}-x_n}{x_A}}$$

$$T_{B_n} = \delta_T^{\frac{y_{n-1}-y_n}{y_B}}$$

其中越新的出块越能反映算力的变化,其周期差占比越大.若$x_A = 0$,则$x_A = x_n - x_{n-1}$,同时,当x_A 且$|x_n - x_{n-1}|$均较小时,即 A 链若干周期出块稳定时,会通过算法使得T_{A_n}趋于 1,降低该参数的影响程度,y_B, T_{B_n}类似.

接下来再统计在目标的φ个区块中,两类父链辅助子链出块数目x_n、y_n.为了出块比例保持稳定,有两种方式进行控制.对于父链 A,首先可以比较二者比值$\frac{x_n}{y_n}$与目标出块比例α的关系,即计算$\alpha/(x_n/y_n)$.为了防止该值波动过大影响稳定性,设定边界$[\mu,\xi]$,即计算$max\left[min\left(\alpha\cdot\frac{y_n}{x_n},\ \mu\right),\xi\right]$.对于父链 B,同理即为$max\left[min\left(\frac{x_n}{\alpha\cdot y_n},\ \mu\right),\xi\right]$.

其次计算x_n与目标值φ的比值与目标比值$\frac{\alpha}{\alpha+1}$的比例关系,即$(\alpha/\alpha+1)/(x_n/\varphi)$.为了防止该值波动过大影响稳定性,设定边界$[\mu,\xi]$,即$max\left[min\left(\frac{\varphi\cdot\alpha}{x_n(\alpha+1)},\ \mu\right),\xi\right]$,但是可以发现$\frac{\varphi\cdot\alpha}{x_n(\alpha+1)}$最小不会超过$\frac{\alpha}{(\alpha+1)}$,因此可以简化为$\min\left(\frac{\varphi\cdot\alpha}{x_n(\alpha+1)},\mu\right)$.对于父链 B,同理即为$max\left[min\left(\frac{\varphi}{y_n(\alpha+1)},\mu\right),\xi\right]*\gamma$.在算法中两种方式结合,并为之设定了加权系数$\beta$、$\gamma$,且$\beta+\gamma=1$.

最后类似比特币难度调整方式计算φ个区块出块的实际出块时间AT与目标出块时间ET的比值,即$\frac{AT}{ET}$.此时还需要根据结果进行判断,当$\frac{AT}{ET} > 1.05$和$\frac{AT}{ET} < 0.95$时,可以认为出块时间波动较大, 此时稳定出块时间更为重要,需要使新增因子的乘积与1对比,使难度调整的方向不发生变化.当$0.95 < \frac{AT}{ET} < 1.05$时,可以认为出块时间比较稳定,此时不需要控制新增因子的结果.

综上,具体难度调整公式如下:

如果 $\frac{AT}{ET} > 1.05$:

$$H_{An+1} = H_{An} \cdot \frac{AT}{ET} \cdot \max\left\{\left\{\max\left[\min\left(\alpha \cdot \frac{y_n}{x_n}, \mu\right), \xi\right] * \beta + \min\left(\frac{\varphi * \alpha}{x_n(\alpha+1)}, \mu\right) * \gamma\right\} * T_{A_n}, 1\right\}$$

$$H_{Bn+1} = H_{Bn} \cdot \frac{AT}{ET} \cdot \max\left\{\left\{max\left[\min\left(\frac{x_n}{\alpha \cdot y_n}, \mu\right), \xi\right] * \beta + \max\left[\min\left(\frac{\varphi}{y_n(\alpha+1)}, \mu\right), \xi\right] * \gamma\right\} * T_{B_n}, 1\right\}$$

如果 $0.95 < \frac{AT}{ET} < 1.05$:

$$H_{An+1} = H_{An} \cdot \frac{AT}{ET} \cdot \left\{\max\left[\min\left(\alpha \cdot \frac{y_n}{x_n}, \mu\right), \xi\right] * \beta + \min\left(\frac{\varphi * \alpha}{x_n(\alpha+1)}, \mu\right) * \gamma\right\} * T_{A_n} .$$

$$H_{Bn+1} = H_{Bn} \cdot \frac{AT}{ET} \cdot \left\{max\left[\min\left(\frac{x_n}{\alpha \cdot y_n}, \mu\right), \xi\right] * \beta + \max\left[\min\left(\frac{\varphi}{y_n(\alpha+1)}, \mu\right), \xi\right] * \gamma\right\} * T_{B_n}$$

如果 $\frac{AT}{ET} < 0.95$:

$$H_{An+1} = H_{An} \cdot \frac{AT}{ET} \cdot \min\left\{\left\{\max\left[\min\left(\alpha \cdot \frac{y_n}{x_n}, \mu\right), \xi\right] * \beta + \min\left(\frac{\varphi * \alpha}{x_n(\alpha+1)}, \mu\right) * \gamma\right\} * T_{A_n}, 1\right\}$$

$$H_{Bn+1} = H_{Bn} \cdot \frac{AT}{ET} \cdot \min\left\{\left\{max\left[\min\left(\frac{x_n}{\alpha \cdot y_n}, \mu\right), \xi\right] * \beta + \max\left[\min\left(\frac{\varphi}{y_n(\alpha+1)}, \mu\right), \xi\right] * \gamma\right\} * T_{B_n}, 1\right\}$$

在上面算法基础上,还需要保证$H_{A_{n+1}}$与H_{A_n}、$H_{B_{n+1}}$与H_{B_n}的比例在[1/5,5]之间,为难度调整增加上下边界.为了防止A, B类链算力突增突减攻击与组合攻击,根据最近 ω 周期,各链出块数少于$B_{low_{limit}}, A_{low_{limit}}$的次数计数$c_{less}, c_{more}$,设定一些特殊情况:

(1) 如果 $y_n \le B_{low_{limit}}$,此时说明小算力父链 B 辅助出块数量较少,原因可能是父链 B 的算力明显减少或者父链 A 的算力明显增加.此时如果$\frac{AT}{ET} \ge 1$可以认为整体算力在减少,说明是父链 B 在减少算力.如果连续多个周期都出现这样的情况,可以认为父链 B 在进行算力攻击,此时对父链 B 的难度调整设置更加平滑,取消新增因子更强烈的影响,以此可以减少在该特殊情况下发动算力攻击的收益.除了此种攻击外,为了防止新增因子对难度调整的影响过强,减少难度调整的幅度.因此,需要对$H_{B_{n+1}}$进行调整,公式如下:

如果 $\frac{AT}{ET} \geq 1$ 且 $c_{lessB} \geq \omega - 1$,则: $H_{Bn+1} = H_{Bn} \cdot \log_{\delta}(\delta + \frac{AT}{ET})$

否则: $H_{Bn+1} = H_{Bn} \cdot \log_{\delta}(\delta + \frac{AT}{ET} \cdot \left\{ max\left[\min\left(\frac{x_n}{\alpha \cdot y_n}, \nu \right), \sigma \right] * \beta + \max\left[\min\left(\frac{\varphi}{y_n(\alpha+1)}, \mu \right), \xi \right] * \gamma \right\} * T_{B_n})$

(2) 如果 $y_n \geq A_{low_{limit}}$,与上同理,说明父链 A 的算力明显减少或者父链 B 的算力明显增加,为了防止算力攻击,因此对$H_{A_{n+1}}$进行调整,公式如下:

如果 $\frac{AT}{ET} \geq 1$ 且 $c_{moreB} \geq \omega - 1$,则: $H_{An+1} = H_{An} \cdot \log_{\delta}(\delta + \frac{AT}{ET})$

否则: $H_{A_{n+1}} = H_{A_n} \cdot \log_{\delta}(\delta + \frac{AT}{ET} \cdot \left\{ \max\left[\min\left(\alpha * \frac{y_n}{x_n}, \mu \right), \xi \right] * \beta + \min\left(\frac{\varphi * \alpha}{x_n(\alpha+1)}, \mu \right) * \gamma \right\} * T_{A_n})$

此外,提供两组参数,如表 1 所示.其中第一组参数周期出块数为 720,每个块的目标出块时间 2 分钟,相对周期时间较短,难度调整更加敏捷,推荐用于快速进行难度调整的区块链和测试使用.第二组参数周期出块数为 2016,每个块的目标出块为 5 分钟,相对周期较长,与目前常见的应用 PoW 共识机制的公链参数类似,推荐应用于更为成熟的公链应用.

表 1 难度调整算法参数表

参数	目标周期出块数φ	父链出块目标比例α	因子上界μ	因子下界ξ	加权系数β	加权系数γ
1	720	10	3	0.33	0.4	0.6
2	2016	10	3	0.33	0.5	0.5
参数	**目标时间$ET(min)$**	**出块记录周期ω**	**底数δ**	**底数δ_A**	**父链 A 出块数边界$A_{low_{limit}}$**	**父链 B 出块数边界$B_{low_{limit}}$**
1	1440	4	10	1.2	320	5
2	10080	3	10	1.2	1516	6

2.4 应用基于素阶数域的后量子数字签名算法

基于 PoW 共识机制的区块链系统中,采用基于 UTXO 的脚本形式构建交易,本文采用 P2PKH（Pay-to-Public-Key-Hash）的脚本[18],交易脚本内容如图 3 所示.本文在区块链的交易中应用 Dilithium-Prime 后量子数字签名方案[6].签名尺寸和安全参数如表 2 所示,本文中选择第III组参数.

Dilithium-Prime 方案是在 Fiat-Shamir with Aborts 技术的基础上提出的基于素阶数域的数字签名方案.当前基于格的签名方案,例如 CRYSTALS-Dilithium,使用的底层代数结构是 power-of-two 分圆环. Power-of-two 分圆环中存在大量子域、环同态等代数结构,且具有小 Galois 群,这三个特性导致了它容易受到 Campbell-Grove-Shepherd、Biasse-Song、Cramer-Ducas-Wesolowski、S-unit 等针对性攻击.而 Dilithium-Prime 使用的底层代数结构是 OpenSSH 等协议所采用的事实标准 NTRU-Prime[21,22]所引入的素阶数域$\mathbb{Z}_q[x]/(x^n - x - 1)$,其中$n$和$q$为素数,且$x^n - x - 1$是$\mathbb{Z}_q[x]$中的不可约多项式.代数结构的单一性有效抵抗 Cramer-Ducas-Wesolowski 攻击、S-unit 攻击和对偶攻击等对分圆环的针对性攻击,此外素阶数域由于原始域的子域数量极少和自同态数量少可以抵抗子域攻击和自同态攻击.因此,具有大 Galois 群的素阶数域因其代数结构的大幅减少被认为在理论上能提供更强的安全保障.同时,从长远考虑,在密码系统构造中去除不必要的代数结构也具有广泛的共识.关于素阶数域相对于 power-of-two 分圆环的安全优势参见[21,22,6].

Dilithium-Prime 方案效率也进行了优化.通过使用种子ρ来代替矩阵$\mathbf{A}$、使用了一系列算法分离并提取$\mathbb{Z}_q$中元素的高位和低位,来进一步压缩公钥尺寸；针对素阶数域上无法使用传统 NTT 技术,多项式乘法效率低的问题,Dilithium-Prime 方案设计了素阶数域上的扩展 NTT 算法,同时针对签名方案的特殊多项式乘法设计了效率更高的小多项式乘法算法.

基于格的后量子数字签名方案具有较高的计算效率和较好的可扩展性,Dilithium-Prime 方案相对于 CRYSTALS-Dilithium,由于采用素阶数域底层结构,具有更鲁棒的安全性且在同等安全程度下,空间开销更小,签名效率更高,更接近目前基于椭圆曲线的数字签名方案.因此,在区块链的交易中应用 Dilithium-Prime 后量子数字签名方案.方案包含密钥生成算法 Dilithium-Prime.KeyGen,签名算法 Dilithium-Prime.Sign 和验证算法 Dilithium-Prime.Verify 三个部分,具体内容如算法 1-算法 3 所示.

算法 1. 密钥生成算法 Dilithium-Prime.KeyGen.

输入: 安全参数1^λ

输出: 签名公私钥对(pk, sk)

$\zeta \leftarrow \{0,1\}^{256}$

$(\rho, \rho', K) \in \{0,1\}^{256} \times \{0,1\}^{512} \times \{0,1\}^{256} := \mathrm{H}(\zeta)$

$\mathbf{A} \in \mathcal{R}_q^{k\times l} := \mathrm{ExpandA}(\rho)$

$(\boldsymbol{s_1}, \boldsymbol{s_2}) \in \mathcal{S}_\eta^l \times \mathcal{S}_\eta^k := \mathrm{ExpandS}(\rho')$

$\boldsymbol{t} := \mathbf{A}\boldsymbol{s_1} + \boldsymbol{s_2}$

$(\boldsymbol{t_1}, \boldsymbol{t_0}) := \mathrm{Power2Round}_q(\boldsymbol{t}, d)$

$tr \in \{0,1\}^{256} := \mathrm{H}(\rho||\boldsymbol{t_1})$

RETURN $(pk := (\rho, \boldsymbol{t_1}), sk := (\rho, K, tr, \boldsymbol{s_1}, \boldsymbol{s_2}, \boldsymbol{t_0}))$

算法 2. 签名算法 Dilithium-Prime.Sign.

输入: 私钥sk,消息M

输出: 签名σ

$\mathbf{A} \in \mathcal{R}_q^{k\times l} := \mathrm{ExpandA}(\rho)$

$\mu \in \{0,1\}^{512} := \mathrm{H}(tr||M)$

$\kappa := 0,\ (\boldsymbol{z}, \boldsymbol{h}) := \perp$

$\rho' \in \{0,1\}^{512} := \mathrm{H}(K||\mu)$ //算法的随机性版本中$\rho' \leftarrow \{0,1\}^{512}$

WHILE $(\boldsymbol{z}, \boldsymbol{h}) := \perp$ **DO**

  $\boldsymbol{y} \in \tilde{\mathcal{S}}_{\gamma_1}^l := \mathrm{ExpandMask}(\rho', \kappa)$

  $\boldsymbol{w} := \mathbf{A}\boldsymbol{y}$

  $\boldsymbol{w_1} := \mathrm{HighBits}_q(\boldsymbol{w}, 2\gamma_2)$

  $\tilde{c} \in \{0,1\}^{256} := \mathrm{H}(\mu||\boldsymbol{w_1})$

  $c \in \mathcal{B}_\tau := \mathrm{SampleInBall}(\tilde{c})$

  $\boldsymbol{z} := \boldsymbol{y} + c\boldsymbol{s_1}$

  $\boldsymbol{r_0} := \mathrm{LowBits}_q(\boldsymbol{w} - c\boldsymbol{s_2}, 2\gamma_2)$

  IF $\|\boldsymbol{z}\|_\infty \geq \gamma_1 - \beta$ **or** $\|\boldsymbol{r_0}\|_\infty \geq \gamma_2 - \beta$

    $(\boldsymbol{z}, \boldsymbol{h}) := \perp$

  ELSE

    $\boldsymbol{h} := \mathrm{MakeHint}_q(-c\boldsymbol{t_0}, \boldsymbol{w} - c\boldsymbol{s_2} + c\boldsymbol{t_0}, 2\gamma_2)$

    IF $\|c\boldsymbol{t_0}\|_\infty \geq \gamma_2$ **or** $the\ \#\ of\ 1's\ in\ \boldsymbol{h}\ is\ greater\ than\ \omega$

      $(\boldsymbol{z}, \boldsymbol{h}) := \perp$

  $\kappa := \kappa + l$

20 **RETURN** $\sigma = (\tilde{c}, \boldsymbol{z}, \boldsymbol{h})$

算法 3. 验证算法 Dilithium-Prime.Verify.

输入: 公钥pk,消息M,签名$\sigma = (\tilde{c}, \boldsymbol{z}, \boldsymbol{h})$

输出: 验证通过(输出 1)或验证不通过(输出 0)

01 $\mathbf{A} \in \mathcal{R}_q^{k\times l} := \mathrm{ExpandA}(\rho)$

02 $\mu \in \{0,1\}^{512} := \mathrm{H}(\mathrm{H}(\rho||\boldsymbol{t_1})||M)$

03 $c := \mathrm{SampleInBall}(\tilde{c})$

04 $\boldsymbol{w_1'} := \mathrm{UseHint}_q(\boldsymbol{h}, \mathbf{A}\boldsymbol{z} - c\boldsymbol{t_1} \cdot 2^d, 2\gamma_2)$

05 **IF** $\|\boldsymbol{z}\|_\infty < \gamma_1 - \beta$ **and** $\tilde{c} = \mathrm{H}(\mu||\boldsymbol{w_1'})$ **and** $\#\ of\ 1's\ in\ \boldsymbol{h}\ is \leq\ \omega$

06 **RETURN** 1

07 **ELSE**

08 **RETURN** 0

表 2 Dilithium-Prime 推荐参数集($n = 251$)

安全等级	II	III	V
模数q	7681537	7681537	7681537
矩阵维数 (k,l)	(4,4)	(6,5)	(8,7)
密钥范围η	2	2	2
拒绝采样次数	3.49	2.96	6.10
公钥尺寸	1288	1916	2544
私钥尺寸	2504	3605	4801
签名尺寸	2504	3233	4511
LWE 困难度	(121,110)	(162,146)	(247,224)
SIS 强困难度	(110,100)	(169,153)	(243,220)
SIS 弱困难度	(120,109)	(184,167)	(263,238)

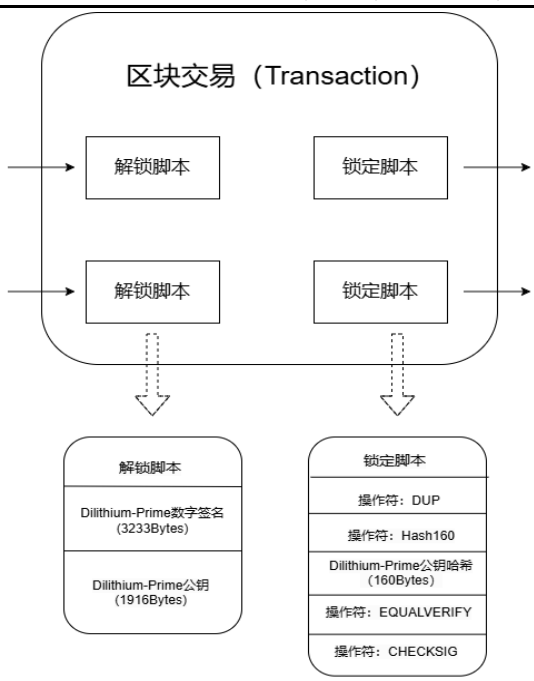


图 3 区块交易脚本结构图

下面举例说明应用过程：

节点 A 与 B 进行交易 Tx 时,由于采用 Dilithium-Prime 数字签名算法,不再具备应用椭圆曲线性质的密钥衍生功能.A 在解锁脚本中生成 Dilithium-Prime 签名sig_a,将生成的签名sig_a,和公钥pk_a写入解锁脚本中.在为交易添加解锁脚本的过程中,会验证脚本的合法性.交易生成后,会广播给区块链的其它节点,各个节点收到交易后会对签名sig_a进行验证,并将交易 Tx 加入交易池中用于构建区块模板.经过多父链辅助工作量证明算法后,生成区块 Block.区块广播其它给节点,其它节点在收到区块后,会对 Tx 进行验证,验证成功后,存储区块 Block,进行下一个区块构建和共识.

值得一提的是,如图 3 所示,与基于椭圆曲线的 Schnorr 签名[17]等相比,签名长度明显增加,因此交易脚本长度变长,为了增加区块处理交易效率,建议将区块空间扩容.

3 理论分析

3.1 Mul-AuxPoW安全性分析

多父链辅助工作量证明共识机制在 PoW 的基础上增加了算力的来源,支持所有在共识机制中使用 Sha256 和 Scrypt 哈希函数的区块链作为父链提供算力来源.对于共识机制将分析 AuxPoW 面对女巫攻击、双花攻击等常见攻击时的表现.

(1) 51%算力攻击

51%算力攻击[19]是指控制超过至少 51%的算力,以达到控制区块链网络,进而修改区块交易的攻击.而在多父链辅助工作量证明共识机制中,算力来源包括子链本身的算力与其它提供辅助共识的父链,相比较 PoW 共识机制明显增加了算力来源,攻击者控制超过至少 51%算力的可能性更加困难,而 PoW 共识机制抵抗 51%算力攻击已经被证明[20].

(2) 双花攻击

双花攻击是指攻击者花费已经花费过的代币.常见方式是通过自私挖矿的方式进行双花,方法是将区块链进行分叉,挖到区块后不广播,等产生一条长度大于主网的分支后,再将其中的交易代币进行双花.在多父链辅助工作量证明共识机制中,增加了算力来源,而要使攻击者算力强于主网需要投入的算力明显增加,进而降低分叉的可能性.此外与比特币等类似,采用 UTXO、时间戳、多次确认等方式抵抗双花攻击.

(3)数据交互中恶意修改信息

可以发现在 Mul-AuxPoW 共识过程中主要存在父链与矿池、子链与矿池的两类交互.在交互过程中,主要是矿池将子链的区块头哈希与难度值提交给父链,父链将计算结果通过矿池返回给子链.对于这两类交互的信息如果被恶意篡改,对于将信息提交给父链,只会引起父链辅助共识失败,父链算力依然保证父链共识,没有造成资源浪费.对于父链将计算结果返回给子链 ,首先子链会检查子链区块头哈希和难度值与父链中写入是否一致,如果恶意修改数据后仍想保证哈希结果的正确值,则需要与正在进行共识的同级别的算力.

3.2 难度调整算法安全性分析

对于比特币等采用难度调整算法,常见具有两种算力攻击:一种是跳矿攻击,在多个周期不断减少算力,总体增加出块时间,进而降低难度,再在下一个周期增加算力.通过这种方式,敌手可以利用在不同的公链之中不断跳转算力,将算力提供到已经减少过算力的公链中,此时该公链难度已经降低,更加容易获得额外利益,此后根据这种方式将算力跳转到其他公链,获得超额利益,同时也会明显减少该周期出块时间,降低稳定性.另一种反之,在多个周期不断增加算力,总体减少出块时间,进而增加难度,再在下一个周期减少算力,大幅度增加出块时间,严重时可以使区块链暂停出块,虽然这种方式很难对敌手产生利益,但会对区块链的稳定程度造成严重影响.

而对于 Mul-AuxPoW 的难度调整算法,有两类父链可以提供算力来源,在这两种攻击中,又可以分为算力较大的父链 A 和算力相对较低的父链 B 分别进行算力的突增突减.下面对可能出现的四种攻击进行分析：

（1）父链 A 持续减少算力再突然增加算力.这种攻击主要目的是使父链 A 获得更多利益,但在难度调整算法中,希望父链 A 和 B 保持一定比例,在参数中建议父链 A 的出块比例一般远大于父链 B.所以在父链 A 这种攻击中,持续减少算力会持续减少之前的出块奖励,而获取在一个周期相对较低的额外收益.

（2）父链 A 持续增加算力再突然减少算力.父链 A 持续增加算力,使得父链 A 出块数增加且总体出块时间减少,两种结果都需要增加对父链 A 的难度.为了避免这种攻击,当发现出现这两种情况时,尤其是使得父链 B 出块数小于参数$B_{low_{limit}}$但不是父链 B 自己减少算力使出块数减少到 0 时,会明显减少父链 B 的难度增加程度.

（3）父链 B 持续减少算力再突然增加算力.这种攻击主要目的是父链 B 不断减少出块数,不能维持出块比例,使得难度不断降低,再突然增加算力,会明显增加额外奖励.为了避免这种攻击,在父链出块 B 不断减少且出块时间增加时,需要控制父链 B 的难度降低程度,使得父链 B 降低难度的收益不断降低.

（4）父链 B 持续增加算力再突然减少算力.由于父链 A 占据大多数的出块数量,对于出块时间影响较大,所以父链 B 的这种数量攻击并不有效,反而会明显降低父链 B 的收益.

3.3 Mul-AuxPoW正确性分析

多父链辅助工作量证明共识机制如 4.2 所示,在共识前为两类父链构建了不同难度的区块模板,并根据不同的父链类型将不同的难度 D 与子链区块头哈希值 H 写入对应父链的 CoinBase 交易中. CoinBase 交易作为区块的一笔交易,利用 Merkle 树的性质, CoinBase 的内容发生修改,形成的 Merkle 树的 Root 也会发生修改,进而父链的区块头哈希也会改变.而原本父链使用 PoW 共识机制中,公式如下：

$$Hash(nounce+Block\ header\ Hash)<Target$$

其中 Block header Hash 值发生改变,那么遍历哈希的结果也不会相同,所以 H 作为父链 CoinBase 交易中的一部分,与父链 Block header Hash 形成一一对应关系,而 H 作为子链工作量证明的一部分,可以理解为在父链的工作量证明共识中也会进行对应难度 D 的计算,进而满足工作量证明.

而满足难度D的父链区块模板也将保存在子链区块的新结构部分中,具体验证内容见2.2的第四阶段,利用 Merkle 树的性质,验证 H 是否存在在 CoinBase 交易中,再根据 CoinBase 所在 Merkle 树分支,利用 SPV 方法计算 Merkle Root 值和区块头哈希是否正确.最后再验证工作量证明是否满足难度,也即子链工作量证明计算满足难度,成功出块.

3.4 难度调整算法正确性分析

难度调整算法总体可以分为 4 个部分,第一部分是控制出块时间稳定,通过计算$\frac{AT}{ET}$,即子链φ个区块的实际出块时间和期望出块时间的比值,并根据值的大小将公式分为 3 个部分.如果大于 1.05,说明实际出块时间略大于目标出块时间,总体难度需要增加以控制总体时间减少；如果小于 1.05 并大于 0.95,说明实际出块时间与目标出块时间基本接近,此时应平衡出块比例更重要；如果小于 0.95,说明实际出块时间略小于目标出块时间,总体难度需要减少以控制总体时间增加.

第二部分是控制两类父链出块比例可以稳定,计算两个值.对于相对算力较大的父链 A 第一个是计算$\alpha \cdot \frac{y_n}{x_n}$,用较小算力的父链 B 和相对算力较大的父链 A 的出块比值,与父链 A 和父链 B 目标出块比值α作比较,如果乘积为 1,说明实际比例和目标值相同,不需要再额外调整难度.如果大于 1,说明父链 B 出块数增多,需要对父链 A 额外减少难度、对父链 B 增加难度,但是变化值需要设定边界,设为$[\mu,\xi]$.但是这个值是由比值构成,当实际比例与目标值相差较多时,变化程度增加更大,因此对于父链 A 还需要计算周期出块数φ与父链 A 出块数的比值,再与目标出块比例做一个比较；父链 B 同理.两部分的值进行加权平均,得出的值为综合计算两类父链出块比例与目标值的关系系数,比例系数为β和γ.

第三部分计算 T_{A_n}和T_{B_n},其中对于父链 A,指数$\frac{x_{n-1}-x_n}{x_A}$部分表示 第 n-1 个周期与第 n 个周期出块差值和 A、B 近若干周期出块变化值的加权平均的比值,当本阶段出块比上一阶段多时,难度会适当变大,反之,难度会变小,同时变化程度会被近几个阶段出块变化情况限制和影响.若近几个阶段变化程度都较小,最新阶段的变化更可能是网络不稳定,故这部分参数会比近几个阶段变化程度大的时候更明显,若近几个阶段变化程度大,最近阶段的变化可能只是因为难度处于调整至稳定的状态,波动较大,要减少这部分参数的影响.y_B同$\mathrm{x_A}$.底数δ_T为参数.

第二部分与第三部分作为调整系数,分别控制出块比例和控制出块波动程度.两部分的乘积需要根据第一部分计算的比例大小再控制比例.这是因为,假设第一部分值大于 1.05,说明为了控制减少出块时间需要减少难度,而第二部分与第三部分的乘积不能小于 1,否则可能将使得三部分乘积后值小于 1,最后增加难度；假设第一部分值再 0.95 与 1.05 之间,说明时间基本稳定,需要重点控制父链出块比例与控制出块波动,因此不需要再设置边界；假设第一部分值小于 0.95,说明为了控制增加出块时间需要增加难度,而第二部分与第三部分的乘积不能大于 1,否则可能将使得三部分成绩后值大于 1,最后减少难度.同时这也表明,在难度调整时,首先需要控制出块时间尽可能保持稳定,再保证出块比例与波动稳定,与比特币等难度调整算法的思想契合.

第四部分设定一些根据算法特性,设定特殊情况.当$y_n \le B_{low_{limit}}$时,只根据这个条件,理论来说要降低父链 B 的难度.有两种出现$y_n \le B_{low_{limit}}$的情况,一是父链A算力突增（或者上一阶段父链 A 难度下降）,针对这种情况应该增加父链 A 难度,父链B难度应降低,但不能降低过多.二是父链B发动算力突减攻击,降低本阶段出块,目的是下一阶段降低父链 B 难度,或者增加父链 A 难度,使得父链 B 下阶段多出块,得到更多的收益,这种情况是最可能发生的情况,应该减少父链 B 难度降低的幅度,但是不会增加或者保持父链 B 难度,防止其他的情况发生.但是当$c_{less} \ge 1$时,可以判断是父链 B 发动多轮算力突减攻击,若不是的话,父链 A 难度第一次调整时难度会增加,父链 B 难度小幅降低,若父链 B 出块还达不到 5 块以上,说明父链 B 算力多个阶段处于不健康状态,再度降低其难度减少幅度,使得父链 B 发动攻击收益较少,成本更大,从而保证系统安全性.对于当发生这种情况时,仍增加父链 A 难度,是对父链 A 不公平的情况,首先这样设置使得,父链 B 发动攻击即使成功,出块的单位时间仍被延长,单位时间收益仍不乐观,其次父链 A 有监督的作用,矿工是逐利的,发生这种情况会涌入父链 B 进行挖矿,那么下一周期,难度就会调整至合适的大小,其次防止了情况一发生的可能.其他的情况有,父链 A、B 同时撤销部分算力但父链 B 撤销的更多,父链 B 上一阶段难度增加的较大,导致本轮出块较少或者其他概率性原因导致出块较少,对于此种情况,我们会慢慢降低算力,在一轮调整后,父链 B 出块便会超过$B_{low_{limit}}$,会根据其他的相应公式对父链 B 难度迅速调整.$y_n \ge A_{low_{limit}}$的情况类似.

3.5 Mul-AuxPoW共识机制和交易效率分析

在共识时间方面,时间由构建区块模板时间t_v、共识哈希计算t_c、区块广播t_p、区块被各节点验证t_y构成.与 PoW 共识机制比较,Mul-AuxPoW 共识机制主要增加在t_v和t_y.在构建区块模板中,需要与父链、矿池进行交互构建区块模板；在验证区块中,需要对 AuxPoW 结构进行验证.但是在共识中,哈希计算t_c对时间影响较深,而其余时间相比较可以忽略不计.所以整体会增加共识时间,但是相比之下增加的时间可以忽略.需要说明的是,t_v和t_y中需要对交易的签名进行验证,需要的时间取决于签名数量和单次验签时间,这里主要考虑共识结构对于时间的影响.

空间方面,区块头中增加 32Bytes 的 AuxPoW 哈希指针.额外增加 CAuxPoW 结构,其中包含 32Bytes 的辅助出块父链区块头哈希、144Bytes 的辅助出块父链 CoinBase 交易,0~576Bytes 的辅助出块父链 CoinBase 交易 Merkle 分支,总体增加了 176~752Bytes 的空间.相比于比特币 1Mb 的区块大小,总体增加了 0.03%的空间大小.另外对于应用后量子区块链的区块链,如果选用表 2 中安全等级为**III**的参数,签名大小将提高到 3233Bytes,公钥大小为 1916Bytes,交易空间明显提高,为了提高区块链效率,将区块空间扩容至 30Mb.相比较增加 Mul-AuxPoW 共识机制应用而言,共识机制额外增加了 0.001%的空间大小.

下面将详细说明后量子签名对交易的影响,在时间方面,由于交易通过脚本形式生成,脚本中主要操作是生

成签名和填充公钥和公钥哈希、操作码等,其中签名时间主要影响交易,由于应用 Dilithium-Prime 签名,总体交易生成时间有所变长.

空间方面以图 3 所示的 P2PKH 脚本为例,解锁脚本由 3234Bytes 的签名和 1916Bytes 的公钥构成,锁定脚本由操作码和 160Bytes 的公钥哈希构成.相对于锁定脚本,解锁脚本的签名和公钥的对空间开销更大,相对于时间,应用 Dilithium-Prime 签名对于空间影响更大,明显增加了单笔交易空间.

3.6 常见PoW公链对比分析

本文中设计的区块链系统是在采用UXTO脚本架构和采用工作量证明共识算法的区块链基础上进行设计实现,代表为比特币和莱特币等公链,在功能上进行对比,如表 3 所示.

在共识算法方面,比特币采用哈希函数为 Sha256 的工作量证明共识算法,算力来源只有选择共识比特币的节点算力.莱特币与狗狗币采用哈希函数为 Scrypt 的辅助工作量证明共识算法,算力来源包括选择共识这两条公链的节点算力.本文设计公链采用多父链辅助工作量证明共识算法,该算法在工作量证明共识算法的基础上,利用哈希函数的性质,允许采用 Sha256 和 Scrypt 哈希函数和工作量证明共识算法的公链节点提供算力支持.明显增加了算力来源,进而提高了区块链的安全性以及算力的利用效率.

在难度调整方面,比特币和莱特币通过一定周期内出块时间与目标时间的比例描述算力的变化情况,进而调整难度.而本文设计的公链在此基础上增加了父链多个周期的出块数稳定性以及针对两类父链出块数目比例的因子调整难度,可以更好地适应多父链辅助工作量证明共识算法和算力变化.针对算力攻击,还设计了特殊情况下的难度调整公式,提高了系统出块地稳定性.

在签名算法方面,比特币和莱特币都采用椭圆曲线数字签名算法,面对量子计算机的不断突破,对于其安全性受到严重影响.本文设计的公链采用具备后量子安全的 Dilithium-Prime 算法并适配 UTXO 脚本设计,满足正常的功能需求,并明显提升了系统交易安全性.

表 3：与其他公链功能对比

区块链	共识算法	难度调整算法	签名算法
比特币	工作量证明共识算法	通过出块时间与目标时间比例调整难度	椭圆曲线数字签名算法
莱特币	辅助工作量证明共识算法	通过出块时间与目标时间比例调整难度	椭圆曲线数字签名算法
本文设计公链	多父链辅助工作量证明共识算法	通过出块时间与目标时间比例、父链出块数目比例、父链多个周期的出块数稳定性调整难度	后量子安全的 Dilithium-Prime 算法

4 实验验证

本文通过 C++语言实现了包含多父链辅助工作量证明共识机制、难度调整算法、在交易中应用 Dilithium-Prime 后量子签名算法的区块链系统,并搭建 5 个 4 核 16Gb 内存云服务器用于仿真实验.此外,通过 Python 脚本,利用 Docker 技术在每台云服务器平均构建 30 个节点用于分别搭建区块节点.相关实验参数设定如下：

(1) 难度调整算法选择表中第 1 组参数,周期设置较小便于系统测试.

(2) 根据目前算力实际环境,分配为运行 Sha256 哈希函数父链节点 140 个,运行 Scrypt 哈希函数父链节点 5 个,运行子链节点 5 个.

(3) 使用 C++语言模拟实现矿池交互流程,云服务器节点网络通信延迟约为 0.6ms,带宽为 3M.

本节对 Mul-AuxPoW 共识机制效率、难度调整算法的模拟攻击、应用后量子签名的交易效率开展实验,测试系统的稳定性和效率.

4.1 Mul-AuxPoW共识机制效率

（1） Mul-AuxPoW 共识机制速度

本实验通过模拟区块共识流程,进行 1000 次共识(不包含交易),计算各流程平均用时和比例.实验结果如表所示.可以观察共识哈希计算t_c占据共识流程的绝大多数时间,没有填充交易的区块模板时间t_v和区块验证时间t_y相比较而言可以忽略不计.

表 4：区块共识流程平均用时和比例

流程	区块模板时间t_v	共识哈希计算t_c	区块广播t_p	区块验证t_y
用时（ms）	10.52	120,000.00	0.21	0.58
比例	<0.01%	99.99%	<0.01%	<0.01%

（2） Mul-AuxPoW 共识机制存储

本实验测试区块大小的边界值,即只有 CoinBase 交易的区块与空间完全应用的区块头长度、CAuxPoW 模块大小以及整体区块大小.实验结果如表所示.可以观察到对于只有 CoinBase 交易的区块,区块头和 CAuXPoW 共识占据主要空间,随着交易的增多,直到空间区块结构占满,区块交易占据主要空间,新增的 CAuXPoW 共识空间开销和区块头开销可以忽略不计.

表 5：只有 CoinBase 交易的区块空间开销和比例

流程	区块头	CAuxPoW	区块
空间（Bytes）	80	176	301
占区块比例	26.58%	58.47%	100%

表 6：空间全部应用的区块空间开销和比例

流程	区块头	CAuxPoW	区块
空间（Bytes）	80	752	30000000
占区块比例	<0.01%	<0.01%	100%

4.2 难度调整算法攻击测试

本实验模拟两类父链进行算力攻击时区块链的稳定性.首先稳定模拟采用 Scrypt 哈希函数的父链 B 算力不变,采用 Sha256 哈希函数的父链 A 的算力在一段周期内先后大幅增加、大幅减少、小幅增加、小幅减少后,子链系统整体出块时间如图 4 所示,可以观察总体出块时间随算力变化先减少后增,根据变化程度可以快速调整到正常波动区间范围,在 2-4 个周期以内稳定出块时间.两类父链的出块数目如图 5 所示,父链 A 的出块数随着算力变化发生剧烈变化后,在 2-4 个周期以内可以稳定出块数目在目标值左右.

同理,稳定模拟采用 Sha256 哈希函数的父链 A 算力不变,采用 Scrypt 哈希函数的父链 B 算力在一段周期内先后大幅增加、大幅减少、小幅增加、小幅减少后,子链系统整体出块时间如图 6 所示,两类父链出块数如图 7 所示.

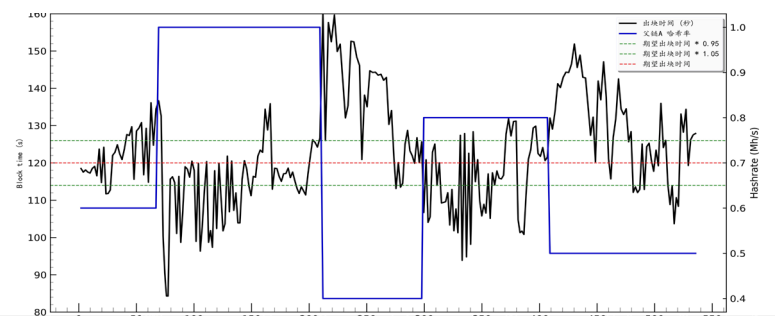


图 4　父链 A 算力攻击下的系统出块时间

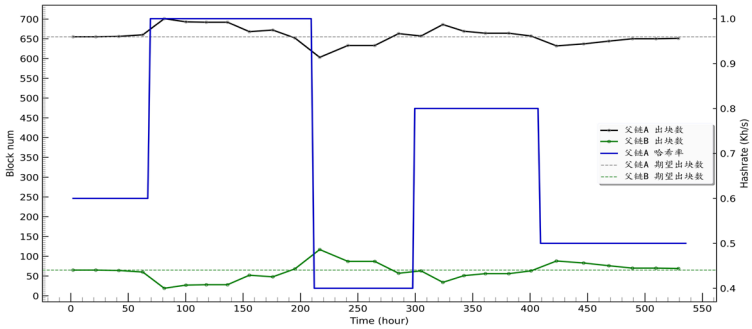
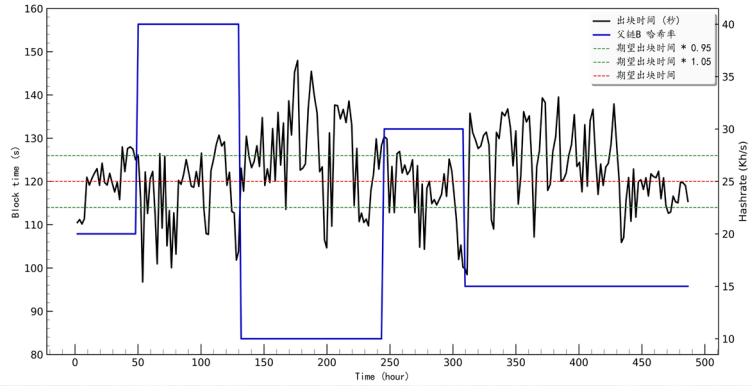


图 5　父链 A 算力攻击下的两类父链出块数



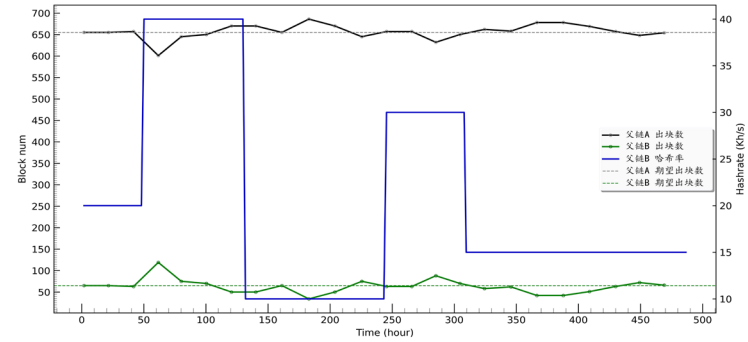


图 6　父链 B 算力攻击下的系统出块时间

图 7　父链 B 算力攻击下的两类父链出块数

4.3 应用后量子签名的交易效率实验验证

（1）　应用后量子签名的交易速度分析

本实验通过模拟交易流程,进行 1000 次交易,计算签名和交易平均用时和比例.实验结果如表所示.可以观察一次签名平均用时 0.5ms,占一次交易的比例为 28.8%.后量子签名 Dilithium-Prime 应用在时间上对交易影响较小,此外交易还会进行读写内存等消耗时间.

表 7：签名与交易平均用时和比例

流程	签名	交易
时间（ms）	0.55	19.09
比例	28.8%	100%

（2） 应用后量子签名的交易存储开销

本实验通过模拟交易,测试交易和签名的空间存储开销,实验结果如表所示.在交易中签名和公钥大小占据交易的大部分空间,操作符和公钥哈希等也会占据一定空间.可以看出后量子签名 Dilithium-Prime 应用在空间上对交易影响较大.

表 8：签名与交易空间开销与比例

流程	签名	公钥	交易
空间（Bytes）	3233	1916	5374
比例	60.2%	35.7%	100%

5 总 结

本文设计并实现一种基于多父链辅助工作量证明共识机制和应用基于素阶数域后量子签名算法的后量子区块链系统.系统采用多父链辅助工作量证明共识机制,算法通过子链-矿池-父链三方交互模型,使在共识机制中采用 Sha256 和 Scrypt 哈希函数的区块链作为父链,对子链进行辅助共识,共识机制增加了子链的算力来源,提高了现存矿池和矿工算力的利用率.针对这种共识机制,设计同时可以稳定两类父链出块比例和稳定出块时间的难度调整算法,并可以抵抗算力突增突减等攻击.在交易中采用基于素阶数域的 Dilithium-Prime 后量子数字签名,大幅减少后量子签名算法所基于底层数学问题的代数结构,增加了交易的后量子安全高可靠性.最后,给出了系统的理论分析和实验验证.

References:

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

[2] Shor P W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Review, 1999, 41(2): 303-332.

[3] NIST. PQC standardization process: Announcing four candidates to be standardized, plus fourth round candidates. (2022-07-05) [2023-07-31]. https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4.

[4] Bai S, Ducas L, Kiltz E, et al. CRYSTALS-Dilithium Algorithm Specifications and Supporting Documentation(Version 3.1). NIST PQC Round 3 Submission, 2020 [2023-07-31]. https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions.

[5] Prest T, Fouque P A, Hoffstein J, et al. Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU(Specification v1.2). NIST PQC Round 3 Submission, 2020 [2023-07-31]. https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions.

[6] Dong YF, Fang BY, Liang ZC, Zhao YL. An efficient lattice-based digital signature scheme over large-Galois-group prime-degree prime-ideal field. Ruan Jian Xue Bao/Journal of Software, 2021 (in Chinese). http://www.jos.org.cn/1000-9825/0000.htm.

[7] Alberto Torres W A, Steinfeld R, Sakzad A, et al. Post-quantum one-time linkable ring signature and application to ring confidential transactions in blockchain (lattice RingCT v1. 0).In:Information Security and Privacy: 23rd Australasian Conference, ACISP 2018, Wollongong, Australia:Springer International Publishing, 2018: 558-576.

[8] AyakaEna.Whitepaper draft.2022.https://github.com/ArielCoinOrg/docs/blob/main/whitepaper.md.

[9] HCASH Foundation.Hcash technical yellow paper.2018https://h.cash/themes/zh-cn/images/YellowPaper-1.01-Chinese-1.pdf.

[10] Ding J. A new proof of work for blockchain based on random multivariate quadratic equations.In:Applied Cryptography and Network Security Workshops: ACNS 2019 Satellite Workshops, SiMLA, Cloud S&P, AIBlock, and AIoTS.Bogota, Colombia: Springer International Publishing, 2019: 97-107.

[11] Jackalys. Quantum Resistant Ledger (QRL).2016.thttps://github.com/theQRL/Whitepaper/blob/master/QRL_whitepaper.pdf.

[12] M. Zweil.Mochimo: Post-quantum currency.2018.https://mochimo.org/assets/files/mochimo_wp_EN.pdf.

[13] LIU YZ, LIU JW, ZHANG ZY, XU TG, YU H. Overview on Blockchain Consensus Mechanisms. Journal of Cryptologic Research, 2019, 6(4): 395-432 (in Chinese). https://doi.org/10.13868/j.cnki.jcr.000311[10.13868/j.cnki.jcr.000311]

[14] Judmayer A, Zamyatin A, Stifter N, et al. Merged mining: Curse or cure.In:Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2017 International Workshops, DPM 2017 and CBT 2017, Oslo, Norway: Springer International Publishing, 2017: 316-333.

[15] Noda S, Okumura K, Hashimoto Y. An economic analysis of difficulty adjustment algorithms in proof-of-work blockchain systems. SSRN 3410460, 2019. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3410460.

[16] Cortesi E, Bruschi F, Secci S, et al. A new approach for Bitcoin pool-hopping detection. Computer Networks, 2022, 205: 108758.

[17] Schnorr C P. Efficient signature generation by smart cards. Journal of cryptology, 1991, 4: 161-174.

[18] Bistarelli S, Mercanti I, Santini F. An analysis of non-standard bitcoin transactions.In:2018 Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland: IEEE, 2018: 93-96.

[19] F. A. Aponte-Novoa, A. L. S. Orozco, R. Villanueva-Polanco and P. Wightman. The 51% Attack on Blockchains: A Mining Behavior Study—*IEEE Access*, vol. 9, pp. 140549-140564, 2021 .

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

[21] Bernstein D J, Chuengsatiansup C, Lange T, et al. NTRU prime: Reducing attack surface at low cost. In: Selected Areas in Cryptography–SAC 2017: 24th International Conference. Ottawa, ON, Canada: Springer International Publishing, 2018, 235-260.

[22] Bernstein D J, Brumley B, Chen M S, et al. NTRU Prime: Round 3. NIST PQC Round 3 Submission, 2020 [2023-07-31]. https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions.

附中文参考文献:

[6]董怡帆,方博越,梁志闯,赵运磊.素阶数域上的高效数字签名方案.软件学报. http://www.jos.org.cn/1000-9825/0000.htm

[13] 刘懿中, 刘建伟, 张宗洋, 徐同阁, 喻辉. 区块链共识机制研究综述. 密码学报, 2019, 6(4): 395-432 https://doi.org/10.13868/j.cnki.jcr.000311[10.13868/j.cnki.jcr.000311

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