

一种基于拓扑结构及分配机制设计的多子块激励共识机制



刘 帅^{1,2} 甘国华³ 刘明熹⁴ 房 勇^{1,2} 汪寿阳^{1,2}

1 中国科学院数学与系统科学研究院 北京 100190

2 中国科学院大学经济与管理学院 北京 100190

3 北京科技大学计算机与通信工程学院 北京 100083

4 中国科学院科技战略咨询研究院 北京 100190

(13220132323@163.com)

摘 要 对经典的 PoW 共识机制进行改进,改变了矿工所挖出区块接入主链的条件和收益分配策略,从而提出了一种改进共识机制。与 PoW 不同,在该改进共识机制中,首个生成的由 N 个子区块相连的子链将被整体接入主链,从而用更为复杂的网状结构来代替原有的单一链结构;改进了传统共识机制的收益分配策略,将分配策略分为 3 个步骤,以期提高算力小的矿工的预期收益,从而激励算力小的矿工积极参与挖矿,提升区块链的安全性。此外,该改进共识机制引入的网状结构,使矿工有了更多的挖矿策略选择。文中分别讨论了挖矿策略的选择、恶意矿工分拆算力、合谋等对区块链的安全与效能产生的影响。最后,通过设置多种市场情景对改进算法进行了仿真实验,分析在各种市场特征下各类矿工的挖矿收益,也就是挖矿策略的种类、选择各挖矿策略的矿工算力之比、算力分布的极差、大小矿工的划分标准等参数对矿工收益的影响效果,以为矿工后期选择挖矿策略提供指导。

关键词: 区块链;共识;矿工激励;挖矿策略

中图法分类号 TP301.6

Multi-subblock Incentive Consensus Mechanism Based on Topology and Distribution Mechanism

LIU Shuai^{1,2}, GAN Guo-hua³, LIU Ming-xi⁴, FANG Yong^{1,2} and WANG Shou-yang^{1,2}

1 Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China

2 School of Economics and Management, University of Chinese Academy of Sciences, Beijing 100190, China

3 School of Computer & Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China

4 Institutes of Science and Development, Chinese Academy of Sciences, Beijing 100190, China

Abstract First of all, this paper proposes an modified consensus mechanism based on the classic PoW (Proof of Work) consensus mechanism by changing the condition of take the miners' blocks into blockchain and income distribution strategy. To be specific, on the one hand, according to the rule of this modified consensus mechanism, the first generated sub-chain made up of N sub-blocks will be integrated into the main chain, which is different from PoW, the modified consensus mechanism replaces the simple single chain structure of PoW with a more complex network structure; on the other hand, the modified consensus mechanism improves on the revenue distribution strategy of the traditional consensus mechanism, its distribution strategy is divided into three steps in order to improve the expected earnings of miners with low computational power, so as to encourage those miners with low computational power to actively participate in mining and supervise the safety of blockchain. In addition, the network structure introduced by the modified consensus mechanism enables miners to have more strategies of mining. This paper also discusses the influence of selecting different mining strategies, splitting calculation power of malicious miners and collusion on the safety and efficiency of the block chain. Finally, a variety of market scenarios are set up to simulate the improved algorithm so as to analyze the mining benefits of various miners under different market characteristics, which are hoping to guide miners.

Keywords Blockchain, Consensus, Inspire miner, Mining strategy

收稿日期:2020-02-03 返修日期:2020-05-07 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家自然科学基金(71631008);大数据与区块链实验室项目

This work was supported by the National Natural Science Foundation of China (71631008) and Big Data and Blockchain Lab Project.

通信作者:房勇(yfang@amss.ac.cn)

1 引言

随着比特币的出现,一系列加密数字货币应运而生。区块链^[1]是这些加密数字货币的底层技术,其去中心化和分布式计算等特征改变了传统技术的中心化模式,引起了各国政府及学术界的广泛关注。从本质上来讲,区块链是一个去中心化的分布式账本数据库,它运用时间戳、Merkle 树形结构、不对称密钥加密算法、共识算法和奖励机制等技术,实现了基于 P2P 网络的去中心化的信用交易。与中心化系统相比,区块链可让节点在无需相互信任的前提下,保证分布式系统实现去中心化信用的点对点交易、协调与协作,从而具备了完全公开、不可篡改、防止多重支付及不依赖第三方等优点,成为了一种解决中心化机构普遍存在的高成本、低效率和数据存储不安全等问题的优选方案。

作为区块链的核心,共识机制的目标是促使所有诚实节点保存一致的区块链视图,即需要同时满足一致性和有效性。其中,一致性是指所有诚实节点保存的区块链前缀部分完全相同;而有效性是指由某诚实节点发布的信息终将被其他所有诚实节点记录在自己的区块中^[2]。一个好的共识机制能够平衡并有效解决区块链的安全性、扩展性、性能效率和能耗代价等问题^[3]。目前的共识机制主要有:PoW (Proof of Work)^[1],PoS (Proof of Stake)^[4],DPoS (Delegated Proof of Stake)^[5],PoA (Proof of Activity)^[6],PoSV (Proof of Stake Velocity)^[7],PoET (Proof of Elapsed Time)^[8],PoRs (Proofs of Retrievability)^[9],UTXO (Unspent Transaction Output)^[10],BFT (Byzantine Fault Tolerance)^[11],PBFT (Practical Byzantine Fault Tolerance)^[12],SBFT (Speculative BFT)^[13],RBFT (Redundant BFT)^[14],HoneyBadgerBFT^[15],dBFT (delegated BFT)等。

在现有共识机制中,有一类需要矿工通过竞争某一种资源来获得记账权,如 PoW, PoS, PoA, PoSV, PoSpace 和 PoET 等。针对经典的 PoW 共识机制存在的问题,人们进行了一系列的改进。例如,GHOST-PoW 机制^[16]试图解决共识效率低下、确认时间较长的问题;Bitcoin-NG 算法^[17]致力于解决系统延迟问题,提高系统的吞吐量;Inclusive 共识机制^[18]充分利用所有游离区块,改善了目前共识算法偏爱较为集中的节点的缺点;PoSpace 共识机制将 PoW 机制中的算力资源替换为磁盘空间资源,更节能;2015 年提出的 SCP 机制是 PoW 和 BFT 算法的结合;2016 年提出的 PeerCensus 机制是 PoW 和 PBFT 算法的结合;ZD 策略^[19]试图解决矿工们互相攻击时导致所有矿工收益均下降的问题,该算法建立了两个被采矿者之间的迭代博弈模型来缓解采矿者的困境;PoA 算法将交易写入新区块的手续费分配给挖矿成功的矿工和 N 个在线的持币者,通过奖励参与度高的持币者来刺激并提高挖矿的参与度,解决了部分持币者未必想参与记账的问题,同时试图减少交易报酬增加带来的网络攻击。

然而,目前大多数改进的算法仍是单一区块依次接入主链的运行机制,未改变单一的链式筛选结构,而是通过增加协议(包括安全协议^[20]、算法协议^[21]、惩罚协议^[22-24]等)、合约、监督机制或区分节点功能来提高区块链的性能和效率。此外,目前还没有解决“资源集中问题”的公认方案。为了更方

便地展示本文改进的共识机制,接下来以 PoW 算法作为通过竞争某一种资源获得记账权的共识机制的代表,设计一种改进的共识机制来解决上述两个问题。对于 PoW 算法,“资源集中问题”就是“算力集中问题”,是指算力大的矿工每次都大概率获得记账权,从而导致算力小的矿工相继退出竞争或丧失积极性。如果算力小的矿工相继退出系统,缺少了小矿工的参与和监督,那么区块链的记账权将更为集中地掌握在少数人的手里,从而减弱了系统的去中心化能力,甚至朝着中心化的方向发展。简洁起见,以下将算力小的矿工统一简称为“小矿工”,将算力大的矿工简称为“大矿工”。

本文首次基于 PoW 算法及拓扑学理论提出了一种间接激励小矿工的改进共识机制,其可在一定程度上解决“算力集中问题”,激励小矿工持续参与到区块链系统并乐于竞争记账权。一方面,本文提出的共识算法通过引入由多个区块生成的可并列竞争的子链形成复杂的网状拓扑结构,一改单一的链式筛选结构,提高了小矿工竞争成功的概率;另一方面,该机制对各子链中的各个矿工竞争记账权的资格进行约束,从而大大增加了小矿工获得记账权的机会。同时,为激励小矿工,本文在分配机制上也做了改进。有些共识机制扩大了分配收益的矿工群体^[25],有些挑选幸运者进行利益分配^[26],但本文主要借鉴 PoA^[6]共识机制的想法,将矿工收入分为确定收入和不确定收入,并将确定收入的分配机制分为初次分配、再分配和二次再分配 3 个步骤,在分配机制上给予小矿工更多的优惠。

本文设计的共识机制主要通过改进拓扑结构和分配机制来解决资源集中的问题,因此适用于 PoW, PoS, PoA, PoSV, PoSpace 和 PoET 等任一通过竞争某一种资源来获得记账权的共识机制,但不适用于各类拜占庭共识机制或 DPoS 等一致性强且不出现分叉的共识机制。

本文第 1 节简单介绍了区块链技术、常用的共识机制及两者间的关系,引出问题,陈述创新点;第 2 节详细介绍本文提出的共识机制;第 3 节讨论本文提出的共识机制性能;第 4 节进行仿真实验,比较不同情况下各矿工的收益,以期指导矿工得到优质挖矿策略;第 5 节对所提算法进行展望;最后总结全文。

2 基于 PoW 算法的多子块激励共识机制

针对目前 PoW 算法算力较为集中,且大矿工每次都大概率获得记账权而导致小矿工相继退出竞争或丧失积极性的问题,多子块激励共识机制意在激励小矿工,加强监督。

2.1 术语定义

定义 1(子链) 在已确定的主链下,选择最终连接到主链的子链过程中,会产生多个子链,这些子链的首个区块均连接到同一子区块(主链上最后一条子链的最后一个子区块)上,即这些子链首个区块均成功求解的是同一个区块留下的难题,而后逐步由多个区块连接成的链条。子链的结构可以很复杂,图 1 为参数 $N=4$ (参数的定义见 2.3 节)时可能出现的子链网状图,其中共 9 条子链,第一层的单一节点 O 为上一个竞争成功的子链的最后一个子区块,最终首个生成 4 个子区块的子链用虚线表示,它为成功上链的子链。

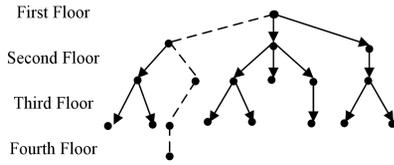


图1 挖矿过程中子区块构成的网状示例图

Fig. 1 Sample network of sub-blocks

定义 2(子区块(子块)) 竞争成功并连接到各子链的区块,如图 1 中的各个黑色节点。

定义 3(第 n 层节点) 各子链中依次连接到该子链的第 n 个子块构成的集合,特别地,第 N 层有且只有一个节点。

2.2 基于 PoW 算法的多子块激励共识机制的主要思路

基于 PoW 算法的多子块激励共识机制遵循 PoW 算法的原理,为获得记账权,各矿工必须充分利用各自计算机的算力来竞争求解一个 SHA256 数学难题,即搜索一个使区块头中所有数据的哈希值小于某个既定目标值的随机数,该问题易于验证。每次成功解决问题的矿工,都要立即向全网广播其挖矿成功的消息,包括:其挖矿成功的内容、留下的区块头、子链上排在他前面的区块名称。

多子块激励共识机制与传统 PoW 算法的不同点在于:不是仅选取一个区块连接到主链中,而是选取最先生成的由 N ($N > 1$) 个子区块相连的子链连接到主链中;为了增加小矿工获得记账权的机会,在每次生成接入主链的子链的过程中,对每个子链中矿工竞争记账权的资格都做了约束,要求各个子链中的所有子区块对应的矿工两两不可相同,即每个子链中的子区块两两不可由同一矿工生成的,且同一矿工生成的区块也不可出现在不同的子链(包括此次成功连接到主链中的子链和未接入主链中的各条子链)中;主链的选取采用 GHOST^[16] 方法,以避免一个矿工恶意隐藏大量子链并一次性公布的现象;确定收入的分配机制会增加小矿工的期望收益,从而激励小矿工参与挖矿,监督网络安全。

多子块激励共识机制共包括两大部分:(1)生成接入主链的由多个子区块组成的子链;(2)给各矿工分配收入。第一步,生成接入主链的子链:首先生成各子链的第一个区块,这些区块求解的问题为上一步连接到主链的子链的最后一个区块留下的难题,其求解难题后要立即广播,竞争记账权;而后各子链进行 $(N-1)$ 步筛选子区块的过程。在每一步中,各矿工只要不违背“各个子链中的所有子块两两不由同一矿工生成”的规则,就可选择接在任一子链后挖矿,成功求解难题后立即向全体矿工广播,立即进行验证。验证成功并最先求解的矿工生成的子块可连接到他选择的子链上。最先满足有 N 个子块按序相连的子链连接到主链中。第二步,分配收入。为激励小矿工,设计的分配机制主要思路为:初次分配的收入与矿工算力成反比,能在一定程度上给予小矿工更多的奖励;对于收入未达到平均水平的小矿工,通过再分配机制给予一定的补偿,对于收入达到平均水平的小矿工,对提取他们收入进入再分配池的比例低于大矿工的提取比例,也给予小矿工一定的保护;对未达到平均水平的小矿工的补偿有上限,即累积确定收入不得超过平均收入,因此再分配后仍未分配出去的资金仍按比例退回给那些收入超平均水平的矿工们。综合以上考虑,收入分配过程可概括为:将矿工收入分为

确定和不确定收入,确定收入分为初次分配、再分配和二次再分配共 3 步,初次分配收入与矿工算力、挖矿时间成反比;对收入未达均值水平的小矿工,通过再分配机制给予一定补偿,对收入达均值水平的矿工,将超均值的部分按一定比例提取至再分配池,小矿工提取比例小于大矿工;对未达平均收入的小矿工补偿后累积确定收入不得超过均值,因此再分配后剩余的资金仍按比例退回给那些超平均收入水平的矿工。各矿工的不确定收入为按照其计入区块的每一笔交易信息中的交易金额的某一百分比 $r\%$ 计提手续费。多子块激励共识机制的流程如图 2 所示。

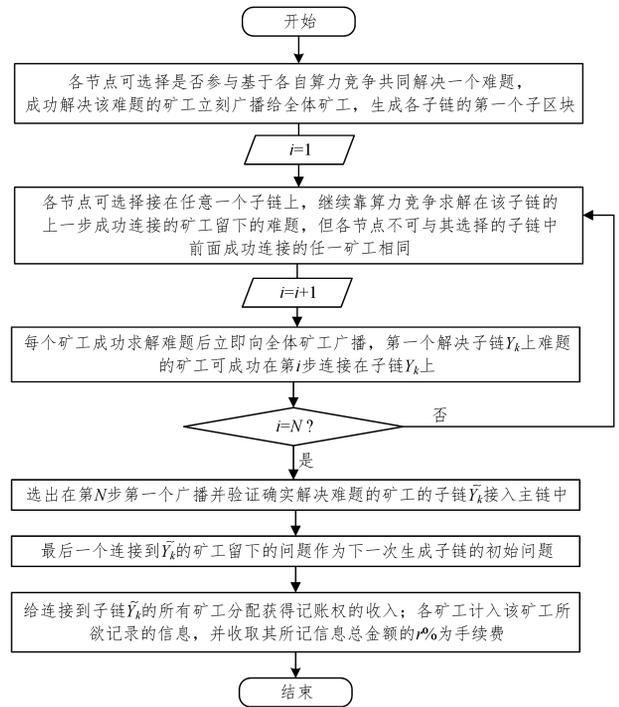


图2 多子块激励共识机制流程图

Fig. 2 Multi-subblock incentive consensus mechanism flow chart

2.3 基于 PoW 算法的多子块激励共识机制的详细运行步骤

(1) 选取接入主链的多个子块生成的子链,该选取过程可划分为 5 个小步骤。

1) 据上一成功连接到主链上的子链最后一个区块留下的区块头,各矿工可选择计算 SHA256 难题,竞争记账权。成功求解的矿工共 M 个,记为 $\{F_1, F_2, \dots, F_M\}$ (F_j 表示第 j 步生成子块的过程中第 j 个成功挖矿的矿工,其挖矿用时 t_j),子链的网状图共含 Q 个子链 Y_1, Y_2, \dots, Y_Q 。

2) 各矿工可在第 1) 步中挖矿成功的矿工留下的 M 个区块头中任选一个(但矿工 F_j 不可继续挖矿)接在其后继续计算 SHA256 数学难题,竞争记账权。例如,成功求解的矿工记为 $\{X_1^Y, X_2^Y, X_3^Y, \dots, X_i^Y\}$ 表示第 $i > 1$ 步生成子块的过程中,第 i 层节点中第 j 个成功挖矿的矿工,其挖矿用时为 t_j ,其所生成的子区块是接在子链 Y_k 上的第 i 个子区块)。

3) 从第二次挖矿开始,每个子链在该步骤最多只有一个成功连接在该子链上的区块,以最先求解上一步成功连接在该子链上的区块留下的数学问题的区块为准。

4) 各矿工可在上一步挖矿成功的矿工留下的区块头中任选一个接在其后继续计算 SHA256 难题,竞争记账权。第 3) 步成功求解的矿工记为 $\{X_1^Y, X_2^Y, X_3^Y, \dots\}$ 。

5)以此类推,直到某个子链成为首个生成了由 N 个子区块按序连接在一起的区块链。每次最先成功生成 N 个子区块连接且满足其他全部要求的子链,被视为挖矿成功的子链,连接到主链中,获得相应的区块记账权和奖励。

(2)给成功竞得记账权的矿工分配奖励,各矿工收到的奖励分为两部分:确定收入和不确定收入。

针对(1)中被选取连接到主链上的子链,生成其中各子区块的矿工算力比从大到小分别是 P_1, P_2, \dots, P_N , 他们每个人成功解决问题并验证成功所用的时间即成功挖矿并连接在子链上的间隔时间依次为 t_1, t_2, \dots, t_N 。区分是否为算力小的矿工的标准设为 p , $P_i > p$ 时,认定为非算力小的矿工,不予补偿; $P_i \leq p$ 时,认定为算力小的矿工。标准 p 可根据整个网络需求(如网络的挖矿难度或挖矿效率等)进行人为的调整,并选取某个固定值或者 P_1, P_2, \dots, P_N 的某个分位数。

确定收入的分配包含初次分配、再分配和二次再分配共 3 步,共分配 NA 报酬,因有 N 个子区块,故每个区块确定收入的均值为 A 。为激励各矿工挖矿的积极性,各矿工初次分配的收入与所用挖矿时间成反比,反映这一关系的函数定义为 $f(t)$ 。为激励算力小的矿工积极参与,初次分配的收入与算力的大小成反比,反映这一关系的函数定义为 $g(P)$ 。算力比为 P_i 且挖矿所耗时间为 t_i 的矿工初次分配(Primary Distribution)的收入用变量 $pd(t_i, P_i)$ 表示。

$$(1) \text{初次分配, } pd(t_i, P_i) = \frac{NA}{\sum_{i=1}^N f(t_i)g(P_i)} f(t_i)g(P_i)。$$

(2)若 $P_i \leq p$ 且 $pd(t_i, P_i) \geq A$, 即是小矿工且初次分配收入大于平均收益,则需要进行再分配,在 $(pd(t_i, P_i) - A)$ 部分提取 k_1 存入再分配池。

(3)若 $P_i > p$ 且 $pd(t_i, P_i) < A$, 即是大矿工且初次分配收入小于平均收益,则不进行再分配,该矿工的最终收入为 $pd(t_i, P_i)$, 不需要提取该矿工收入存入再分配池。

(4)若 $P_i \leq p$ 且 $pd(t_i, P_i) < A$, 即是小矿工且初次分配收益小于平均收益,则需要进行再分配,但不需要提取该矿工的初次分配收入存入再分配池。

(5)若 $P_i > p$ 且 $pd(t_i, P_i) \geq A$, 即是大矿工且初次分配收益大于平均收益,则矿工的收入需要进行再分配,在 $(pd(t_i, P_i) - A)$ 部分提取 k_2 (满足 $k_2 > k_1$) 存入再分配池。

(6)汇总再分配池中总共提取的金额 S 。

(7) $P_i \leq p$ 且 $pd(t_i, P_i) < A$ 的所有矿工算力之和为 \tilde{P} 。

(8)若 $P_i \leq p$ 且 $pd(t_i, P_i) < A$, 则该矿工会受到再分配收入 $\frac{S}{P}P_i$ 。若 $pd(t_i, P_i) + \frac{S}{P}P_i > A$, 则将 $(pd(t_i, P_i) + \frac{S}{P}P_i - A)$ 收回,计入二次再分配池;若该矿工的累积收入

$$pd(t_i, P_i) + \frac{S}{P}P_i < A, \text{ 则最终收入为 } pd(t_i, P_i) + \frac{S}{P}P_i。$$

(9)二次再分配池中总共收回的金额为 \tilde{S} 。

(10)将 \tilde{S} 按照再分配池中各矿工被提取的金额比例发回各被提取的矿工。

每个矿工不确定收入的计算方式为:按照其计入区块的每一笔交易信息中的交易金额的某一百分比 $r\%$ 来计提手续费。

3 算法性能

本节将从多子块激励共识机制的全网特征改进和各矿工挖矿之间的博弈这两个角度对算法性能进行分析。下文分析中均假设:(1)挖矿成功后,向全网发布挖矿成功并验证的时间忽略不计;(2)各个矿工的算力不随着时间的变化而变化。

首先分析基于 PoW 算法的多子块激励共识机制对全网特征进行的改进。

3.1 降低恶意节点收益,以保障区块链的安全

首先估计各矿工确定收益的期望值。假设:(1)只有一条子链,且该子链下无分叉;(2)任一矿工 X 的算力为 P_0 , 除此之外,矿池中还有 K 个矿工,它们的算力分别是 $P_1 \dots P_K$, 满足 $\sum_{i=0}^K P_i = a (a \leq 1)$, 若所有潜在具有挖矿能力的矿工均参与挖矿,则 $a=1$;(3)各矿工均老实挖矿,使用且仅使用自己全部的算力进行挖矿,不存在分拆自己的算力或多个矿工合谋的问题;(4)各矿工挖矿的策略很简单,即紧接着当前子链进行挖矿,上一个子区块未挖矿成功,就继续使用全部算力在当前时刻进行挖矿,不存在等待进入某一阶段才进行挖矿的预谋。

在以上假设下,矿工 X 成功挖取第 i 个子区块的概率用 $P_i(P_0)$ 表示,则 $P_i(P_0)$ 的表达式为:

$$P_i(P_0) = \frac{P_0 \prod_{j=1}^{i-1} (1 - P_j(P_0))}{a - \sum_{j=1}^{i-1} P_j} \quad (i > 1)$$

特别地, $P_1(P_0) = \frac{P_0}{a}$ 。其中, \tilde{P}_i 为挖矿成功的子链中除了矿工 X 外成功挖取第 i 个子区块的矿工对应的算力。

又“各子链中的所有子两两不由同一矿工生成”,所以矿工 X 成功挖取第 i 和 j ($i \neq j$) 个子块是两两互斥事件。矿工 X 挖矿成功的概率 $P(P_0) = \sum_{i=1}^N P_i(P_0)$; 初次分配矿工 X 的收入为

$$pd(t_0, P_0) = \frac{NA}{f(t_0)g(P_0) + \sum_{i=1}^N f(t_i)g(P_i)} f(t_0)g(P_0),$$

其中 t_0 为矿工 X 挖矿所需时间。

若矿工 X 为大矿工,其初次分配收益小于各矿工的平均收益时,其收入就是初次分配收益;其初次分配收益大于各矿工的平均收益时,需要抽取一部分收益分配给算力小的矿工,因此其收益减小,也即大矿工的最终收益不会高于初次分配的收益。大矿工 X 初次分配收入的数学期望为挖矿成功的概率和收益的乘积,不超过 $pd(t_0, P_0) * P(P_0)$, 也就是大矿工最终收益的数学期望不超过 $pd(t_0, P_0) * P(P_0)$ 。由于这样的收益是给予矿工 N 次挖矿机会得到的,因此平均每次挖矿的期望收益为 $pd(t_0, P_0) * P(P_0) / N$ 。

若矿工 X 为小矿工,其初次分配收益大于平均收益时,须抽取部分收益进入再分配池,故其收益减小;其初次分配收益小于平均收益时,会得到再分配收入,但不会超过均值 A , 即小矿工的最终收益不超过 $\max(A, \text{初次分配收益})$ 。

综合以上两种情况可知,各矿工最终收入不超过 $\max(A, \text{初次分配收益 } pd(t_0, P_0))$ 。矿工 X 最终固定收益的数学期望不超过 $\max\{A, pd(t_0, P_0)\} * P(P_0)$, 平均每次挖矿的期望收益不超过 $\max\{A, pd(t_0, P_0)\} * P(P_0) / N$ 。

经典的 PoW 算法下, 矿工挖矿成功的概率为 $\frac{P_0}{a}$, 收益为 A , 获得收益的数学期望值为 $\frac{P_0}{a}A$ 。

由于不确定收入无法控制, 为达到本文避免算力集中导致的“51%进攻”问题, 主要通过确定收入进行约束来激励小矿工的监督热情, 抑制大矿工进行恶意攻击。对于算力小的矿工, 只要其安分挖矿, 不恶意合谋, 就可以由参数 p (大小矿工的区分标准) 来使其恶意进攻网络的能力不会太强; 对于大矿工, 可在设置参数 $p, f(x)$ 和 $g(x)$ 的函数形式时, 要求其满足大矿工的确定收益的期望值小于经典 PoW 算法的收益期望值, 即设置参数 $p, f(x)$ 和 $g(x)$, 使得对任意的参数 a, P_i 和 $P_0 > p$, 满足 $pd(t_0, P_0) * P(P_0) / N < \frac{P_0}{a}A$ 恒成立。

对大矿工做出“ $pd(t_0, P_0) * P(P_0) / N < \frac{P_0}{a}A$ ”的要求主要考虑到: 在改进的算法中, 每个区块确定收益的均值与经典 PoW 算法的相等, 均为 A , 也就是不论采用改进算法还是经典算法, 大矿工和小矿工从每个区块中获得的期望之和均为 A , 从而推导出“小矿工在改进算法中获得的确定收益期望值 = A - 大矿工改进算法中获得的确定收益期望值”“小矿工在经典 PoW 算法的收益期望值 = A - 大矿工经典算法中获得的收益期望值”。若大矿工在改进算法中获得的确定收益期望值小于在经典 PoW 算法中的收益期望值, 那么小矿工在改进算法中获得的确定收益期望值大于在经典 PoW 算法中的期望收益, 即小矿工的期望收益在改进算法中有所提升。

3.2 各矿工的挖矿策略

首先分析各矿工的挖矿策略, 除了第 N 层节点外, 每出现一个新的子区块后, 各矿工会有很多种挖矿策略。仿照经典的 Axelrod 实验^[27]曾提出的 9 种经典两矿工博弈策略, 可设计出如下 3 种策略。

策略 1 沿着目前最长的子链继续挖矿。

策略 2 沿着目前长度排名第二的子链挖矿。

策略 3 在最后一步进行挖矿, 即挖掘第 N 层节点。

当全网各个矿工当前在哪条子链的哪个区块下挖矿的信息及各矿工的算力公开可获得时, 还可设计如下两个策略。

策略 4 多条最长子链时, 选择其中沿着该子链挖矿的矿工算力之和最小的子链挖矿; 最长子链唯一时, 沿该链挖矿;

策略 5 沿着某个子链挖矿的矿工算力之和最小, 该矿工沿着该子链挖矿。

具体而言, 针对图 3 中的子链网状图, 共有 8 条子链。假设子链 1—子链 8 中延各子链挖矿的算力之和最小的子链为子链 3 和子链 5 (子链 3 和子链 5 两者算力之和相等), 则各策略的选择为: 策略 1—子链 1—子链 4、子链 6—子链 8 中之一; 策略 2—子链 5; 策略 4—子链 3; 策略 5—子链 3 或子链 5。大矿工可能并不一定会和大家集中算力挖掘最长的子链, 而是可能采取策略 2 进行挖矿, 避免竞争, 并最终提早实现子链生成 N 个子区块。在选择接在哪个第 $N-1$ 层节点后继续挖掘第 N 层子区块时, 策略 1 和策略 3 是一致的; 但接在

第 $i (i < N-1)$ 层节点后选择挖矿策略时, 选择策略 1 的节点将会挖矿来争夺下一层节点的机会, 而选择策略 3 的节点将不进行挖矿, 等待最后一层节点的争夺。两种策略在某一时刻对应的可选择集合或许会有重合。

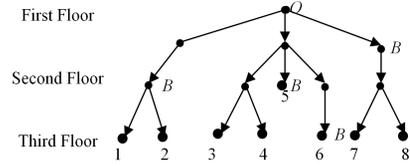


图 3 子链网状图

Fig. 3 Sub-chains network diagram

据全网各矿工选择策略的差异, 可分为如下几种情形。

情景 1 全网所有矿工都选择某一相同策略。如全网所有矿工都选择策略 1, 则此时子链 5 将无矿工在其后继续挖矿, 子链 5 被舍弃。可选子链变少, 此时导致算力集中。

情景 2 全网各矿工的策略都是固定不变的, 不随时间而变化, 但它们并不一定都是相同策略, 选择策略 1 至策略 5 的矿工算力之比为 c_1, c_2, c_3, c_4, c_5 。极端地, 采用某类策略的矿工占比极大时, 易出现情景 1, 造成算力集中。但当某一策略对应的可选子链较多 (如策略 1 对应着子链 1—子链 4、子链 6—子链 8, 共 7 条子链) 时, 能分散算力, 因此子链构成的网状图越复杂, 越易分散算力, 保障区块链的稳定和安全。

情景 3 全网各矿工选择的策略会随时间随机变化, 每个矿工会根据当时的挖矿网络状况进行一定的分析, 而后选择最适宜的挖矿策略。各个矿工根据自己的效用最优原则实时选择挖矿策略, 效用函数的形式、影响效用的因子等都会影响策略的选择。由于这涉及到各个矿工的行为和心理, 暂时无法进行模拟仿真, 因此本文暂不做讨论, 将其作为接下来将要进一步深度挖掘的研究方向。

3.3 矿工恶意分拆算力

首先不考虑复杂情形, 假设某矿工 X 将算力分拆为两部分, 分别记为 P_{01} 和 P_{02} , 满足 $P_{01} + P_{02} = P_0$, 则此时矿工 X 挖矿成功的情况包括两种: 拥有算力 P_{01} 的“假矿工 1”挖矿成功 (记成功的概率为 $P(P_{01})$); 拥有算力 P_{02} 的“假矿工 2”挖矿成功 (记成功的概率为 $P(P_{02})$)。综上, 矿工 X 挖矿成功的概率为 $P(P_{01}) + P(P_{02}) - P(P_{01})P(P_{02}) \leq P(P_{01}) + P(P_{02}) = P(P_{01} + P_{02}) = P(P_0)$, 其中 $P(x)$ 的表达式同 3.1 节。

分拆后挖矿成功的概率虽然不超过分拆前挖矿成功的概率, 但矿工并不一定会进行恶意分拆。因为分拆后, 大矿工可能被分拆为两个或多个小矿工, 从而获得更多的分配收益, 收益的期望值为两者相乘的结果, 因此并不一定变小。

3.4 矿工合谋

矿工的合谋按照大小矿工共分为 3 类: 大矿工和大矿工合谋, 小矿工和大矿工合谋, 小矿工和小矿工合谋。本文设计的共识算法本身就有抑制大矿工收益过高的作用, 因此大矿工和大矿工进行合谋形成更大算力的矿工, 并不能带来更多的收益, 只有大矿工与小矿工进行合谋时, 才可能获得更多的收益。例如, 大矿工分给小矿工一些算力, 但这样的分配使得小矿工吸收大矿工算力后的算力之和仍然维持在标准 p 内,

其仍维持小矿工的身份去挖矿并获得激励,由于小矿工的算力增大,其挖矿成功的概率就会提升,但其仍可获得再分配收益,因此收益的期望值会提高,小矿工收益提高的部分与大矿工共同进行“分赃”;小矿工将算力集中起来,提高挖矿成功的概率,提高收益的期望值,而后进行“分赃”。

4 多子块激励共识机制的仿真测试

4.1 仿真测试的参数、假设和实验场景

本节将设计仿真实验对改进的共识机制进行测试。为了简化实验,并遵循 2.3 节设定各个参数的主要思想,做以下假设。

假设 1 所有潜在具有挖矿能力的矿工均参与到挖矿中,也就是 3.1 节提出的参数 $a=1$,各个矿工均诚实挖矿,不存在合谋或分拆算力等情况。

假设 2 仅有一个子链,不考虑多个子链的情况,最先生成连续 3 个子区块的子连接入主链,也就是 $N=3$ 。

假设 3 所有具有潜在挖矿能力的矿工共 10 个。

假设 4 与目前已实际应用的共识机制一样,求解问题的难度会随时间调整,将生成区块的时间维持在某一个固定值(设为 t_{plan})附近,各个矿工挖矿所用时间的期望值为 t_{plan} ,此

处不妨设 $t_{\text{plan}}=10 \text{ min}$,此为目前比特币生成一个新区块的平均时间,每个取得记账权的区块确定收入的平均值 $A=1$ 。

假设 5 同 2.3 节的思想,为了激励各矿工挖矿的积极性,各矿工初次分配的收入与其所用挖矿时间成反比,反映这一关系的函数设为:

$$f(t) = I(t \leq \frac{t_{\text{plan}}}{2}) (3 + \frac{t_{\text{plan}}}{2} - t) + I(t > \frac{t_{\text{plan}}}{2}) (1 + \frac{t_{\text{plan}}}{t})$$

其中, $I(x) = \begin{cases} 1, & \text{如果事件 } x \text{ 为真} \\ 0, & \text{如果事件 } x \text{ 为假} \end{cases}$ 为示性函数。

为激励算力小的矿工继续积极参与挖矿,各矿工初次分配的收入与各矿工算力成反比,反映这一关系的函数设为:

$$g(P) = \frac{6.6}{P}$$

再分配的提取比例 $k_2=50\%$, $k_1=25\%$ 。

假设 6 挖掘各个子区块时,矿工挖矿成功的概率与其算力占该时刻正在挖矿的所有矿工算力之比呈正线性关系。

针对矿工们可选择的 3.2 节中提到的挖矿策略种类、选择各挖矿策略的矿工算力之比、矿工们的算力分布的极差的大小、大小矿工的划分标准 p 等参数,本文设置如表 1 所列的 16 种实验,并将每个实验重复 200 次。

表 1 各实验场景

Table 1 Each experimental scene

实验编号	策略选择	设置	$p/\%$
实验 1		所有矿工算力均相等,极差为 0	10
实验 2	挖矿策略	有一个算力较大的矿工且其算力为 34%,一个算力很小的矿工且其算力为 2%,其他矿工算力均	5
实验 3	仅有策略 1	相等,极差大	10
实验 4			5
实验 5		所有矿工的算力均相等,极差为 0	10
实验 6	选择策略 1 和 3	有一个算力较大的矿工且其算力为 34%,一个算力很小的矿工且其算力为 2%,其他矿工算力均	5
实验 7	的算力比为 1:1	相等,极差大	10
实验 8			5
实验 9		所有矿工的算力均相等,极差为 0	10
实验 10	选择策略 1 和 3	有一个算力较大的矿工且其算力为 32%,一个算力很小的矿工且其算力为 4%,其他矿工算力均	5
实验 11	的算力比为 4:1	相等,极差大	10
实验 12			5
实验 13		所有矿工的算力均相等,极差为 0	10
实验 14	选择策略 1 和 3	有一个算力较大的矿工且其算力为 32%,一个算力很小的矿工且其算力为 4%,其他矿工算力均	5
实验 15	的算力比为 1:4	相等,极差大	10
实验 16			5

按照大小矿工数量来看,实验 1,5,9,13 中均为小矿工,实验 2,6,10,14 中均为大矿工,实验 3,7,11,15 中有 9 个小矿工,实验 4,8,12,16 中有 9 个大矿工。按照选取策略来看,实验 1-4 的情景为所有矿工均采取 3.2 节中提到的策略 1;实验 5-8 的情景为选择挖矿策略 1 或策略 3 的矿工算力之和均为 50%;实验 9-12 的情景为选择挖矿策略 1 和策略 3 的矿工算力之和分别为 80% 和 20%;实验 13-16 的情景为选择挖矿策略 1 和策略 3 的矿工算力之和分别为 20% 和 80%。按照算力比来看,实验 1,2,5,6,9,10,13,14 这 8 个实验中,所有矿工的算力均为 10%;实验 3,4,7,8 中有一个算力为 34% 的大矿工,一个算力为 2% 小矿工,其他矿工算力均为 8%;实验 11,12,15,16 中有一个算力为 32% 的大矿工,一个算力为 4% 小矿工,其他矿工算力均为 8%。特别地,32% 算力的大矿工在实验 11 中选择策略 1,实验 15 中选择策略

3;4% 算力的小矿工在实验 12 中选择策略 3,在实验 16 中选择策略 1。

4.2 仿真测试的结果分析

本文选用算力最大和算力最小的矿工作为算力占比很大和算力占比极小的矿工代表,试图探索影响他们收益的原因并期望为不同算力的矿工在不同情景下选择挖矿策略提供指导。

接下来将分析挖矿策略的种类、选择各挖矿策略的矿工算力之比、矿工们算力分布的极差的大小、大小矿工的划分标准 p 中的某一个市场参数在其他 3 个参数纷繁多变的情况下对矿工平均收益的影响。

通过比较实验 1-实验 4 和实验 5-实验 16,也就是考虑了矿工们的算力分布的极差、大小矿工的划分标准 p 、选择各挖矿策略的矿工算力之比这 3 个参数多样变化的情况(见表

2)后,据表 2 可以看出:矿工采取的挖矿策略种类越大,小矿工的收益就越大。据此,应鼓励矿工们采用多样化的投资策略,进而激励小矿工参与并保证网络的监督与安全。

表 2 实验 1—实验 4 与实验 5—实验 16 的比较

Table 2 Comparison of experiment 1—experiment 4 with experiment 5—experiment 16

实验	大矿工平均收入	小矿工平均收入
实验 1—实验 4	0.91	1.11
实验 5—实验 16	0.90	1.12

在策略 1 和策略 3 中,又该如何选择呢? 通过比较实验 5—实验 16 中采取不同策略的矿工的收益,也即综合考虑选择各挖矿策略的矿工算力之比、矿工们的算力分布的极差大小和大小矿工的划分标准 p 这 3 个参数多样设置的种种情形(见表 3)后,可以发现:当挖矿策略不单一时,采用策略 1 的矿工的收益会更高。因此,如果矿工没有明确的策略选择意向,且不了解市场其他方面的信息,选择策略 1 更保守。

表 3 实验 5—实验 16 的比较

Table 3 Comparison of experiment 5—experiment 16

策略选择	矿工平均收益
策略 1	1.01
策略 3	0.96

通过比较实验 5—实验 8 和实验 9—实验 16(见表 4),可以发现:当挖矿策略不单一时,不论矿工们的算力分布的极差大小和大小矿工的划分标准 p 如何变化,选择各挖矿策略的矿工算力之比越接近 1,则选择不同策略的矿工收益之差越小,矿工之间的收益分配越均衡,因此建议各个矿工选择挖矿策略时不要集中于某一个策略。

表 4 实验 5—实验 8 和与实验 9—实验 16 的比较

Table 4 Comparison of experiment 5—experiment 8 with experiment 9—experiment 16

实验	选择策略 1 和策略 3 的矿工平均收入之差
实验 5—实验 8	0.01
实验 9—实验 16	0.11

进一步,若真发生了选择某一挖矿策略的算力较为集中的情景,矿工们该如何选择挖矿策略呢? 比较实验 9—实验 12 与实验 13—实验 16,也就是综合考虑大小矿工的算力极差大小和大小矿工的划分标准 p 这 2 个参数多样设置的种种情形(见表 5)后,可以发现:在矿工们挖矿策略的种类不单一时,采用某一策略的矿工的算力占比越大,采用该策略的矿工的收益就越小。

表 5 实验 9—实验 12 与实验 13—实验 16 的比较

Table 5 Comparison of experiment 9—experiment 12 with experiment 13—experiment 16

实验	选择策略 1 的矿工平均收入	选择策略 3 的矿工平均收入
实验 9—实验 12	0.98	1.19
实验 13—实验 16	1.08	0.84

通过比较实验 $4x-3$ 与实验 $4x-1(x=1,2,3,4)$,综合

考虑挖矿策略的种类和选择各挖矿策略的矿工算力之比这 2 个参数多样设置的情况(见表 6)后,可以发现:大小矿工的划分标准 p 越大(也就是小矿工的人数多),大小矿工们的算力分布的极差就越大,小矿工的收益亦越大。

表 6 实验 $4x-3$ 与实验 $4x-1(x=1,2,3,4)$ 的比较

Table 6 Comparison of experiment $4x-3$ with experiment $4x-1(x=1,2,3,4)$

实验	实验 $4x-1$ 中小矿工平均收入与实验 $4x-3$ 中小矿工平均收入之差
实验 1 和实验 3	0.22
实验 5 和实验 7	0.20
实验 9 和实验 11	0.24
实验 13 和实验 15	0.10

通过比较实验 $4x-2$ 和实验 $4x(x=1,2,3,4)$,综合考虑挖矿策略的种类和选择各挖矿策略的矿工算力之比这 2 个参数多样设置的情况(见表 7)后,可以发现:大小矿工的划分标准 p 越小(也就是小矿工的人数少),大小矿工算力的极差越大,大矿工的收益亦越小。

表 7 实验 $4x-2$ 与实验 $4x(x=1,2,3,4)$ 的比较

Table 7 Comparison of experiment $4x-2$ with experiment $4x(x=1,2,3,4)$

实验	实验 $4x$ 中大矿工平均收入与实验 $4x-2$ 中的大矿工平均收入之差
实验 2 和实验 4	-0.027
实验 6 和实验 8	-0.029
实验 10 和实验 12	-0.013
实验 14 和实验 16	-0.127

通过比较实验 $2x$ 与实验 $2x-1(x=2,4,6,8)$,也即综合考虑挖矿策略是否单一、选择各挖矿策略的矿工算力之比是否均衡等多种情况(见表 8)后,可以发现:大小矿工的划分标准 p 的值越大,也就是全网小矿工的占比越多,小矿工的收益就越小,并且大矿工的收益也越小。因此,为保障大小矿工的收益均提高和共识机制的稳定持续运行,应该给 p 设置较小的参数值。

表 8 实验 $2x-1$ 与实验 $2x(x=2,4,6,8)$ 的比较

Table 8 Comparison of experiment $2x-1$ with experiment $2x$

实验	实验 $2x-1$ 中大矿工平均收入与实验 $2x$ 中大矿工平均收入之差	实验 $2x-1$ 中小矿工平均收入与实验 $2x$ 中小矿工平均收入之差
实验 3 和实验 4	-0.666	-0.984
实验 7 和实验 8	-0.664	-0.916
实验 11 和实验 12	-0.658	-0.601
实验 15 和实验 16	-0.592	-0.552

实验 7 或实验 8 对算力的约束设置只包括算力最大和最小的两个矿工选择不同挖矿策略的 2 种情形。当算力最大的矿工选择策略 1 而算力最小的矿工选择策略 3 时,设为细分实验 7.1 和 8.1;当算力最大的矿工选择策略 3 而算力最小的矿工选择策略 1 时,设为细分实验 7.2 和 8.2。

为了分析算力最大和最小的两个矿工群体是否选择相同挖矿策略对收益的影响,本文又新设计了 2 个实验:算力最大 34% 和最小 2% 的矿工可以选择相同的挖矿策略(此时还有一个 6% 算力的矿工,其他 7 个矿工的算力均为 8%),当算力

最大和最小的矿工均选择策略 1 时,设为细分实验 7.3 和 8.3;当算力最大和最小的矿工均选择策略 3 时,设为细分实验 7.4 和 8.4。这 8 个实验的关系如表 9 所列。

表 9 各实验场景的关系

Table 9 Relationship of each experimental scene

	算力最小矿工 选择策略 1	算力最小矿工 选择策略 3
算力最大矿工 选择策略 1	实验 7.3 或实验 8.3	实验 7.1 或实验 8.1
算力最大矿工 选择策略 3	实验 7.2 或实验 8.2	实验 7.4 或实验 8.4

比较实验 7.1、实验 7.2、实验 8.1、实验 8.2 中与 34% 算力的大矿工采取相同策略的其他 8% 算力的矿工平均收益和与 2% 算力的小矿工采取相同策略的其他 8% 算力的矿工平均收益(见表 10),可以发现:当挖矿策略种类、选择各挖矿策略的矿工算力之比、矿工们的算力分布的极差的大小和大小矿工的划分标准 p 这 4 个参数均相同,且算力最大和最小的两个矿工选择不同挖矿策略时,其他矿工选择与算力最大的矿工相同的挖矿策略时平均收益小,而选择与算力最小的矿工相同的挖矿策略时平均收益大。

表 10 实验 7.1、实验 7.2、实验 8.1、实验 8.2 的比较

Table 10 Comparison of experiment 7.1, experiment 7.2, experiment 8.1 and experiment 8.2

实验	与 34% 算力的大矿工采取 相同策略的其他 8% 算力的 矿工的平均收益	与 2% 算力的小矿工 采取相同策略的其他 8% 算力的矿工的平均收益
实验 7.1	1.31	1.33
实验 7.2	0.98	1.08
实验 8.1	1.30	1.33
实验 8.2	0.99	1.08

比较实验 7.1 和实验 7.2(或实验 8.1 与实验 8.2),当挖矿策略种类、选择各挖矿策略的矿工算力之比、矿工们的算力分布的极差的大小和大小矿工的划分标准 p 这 4 个参数均相同,且最大和最小算力的矿工挖矿策略不同时(见表 11),算力最大的矿工选择策略 1 时,大矿工的收益较大,算力最小的矿工选择策略 3 时,小矿工的收益较大。

表 11 实验 7.1(实验 8.1)与实验 7.2(实验 8.2)的比较

Table 11 Comparison of experiment 7.1(experiment 8.1) with experiment 7.2(experiment 8.2)

实验	大矿工的收益	小矿工的收益
实验 7.1	0.31	1.33
实验 7.2	0.30	1.10
实验 8.1	0.98	2.26
实验 8.2	0.96	2.07

比较实验 7.3、实验 7.4、实验 8.3、实验 8.4 中与算力最大和最小的两个矿工群体选择相同、不同挖矿策略的矿工的平均收益(见表 12),可以发现:当挖矿策略种类、选择各挖矿策略的矿工算力之比、矿工们算力分布的极差的大小和大小矿工的划分标准 p 这 4 个参数均相同,且算力最大和最小的两个矿工选择相同挖矿策略时,其他矿工选择与算力最大、最小的矿工相同的挖矿策略时能收益。

表 12 实验 7.3、实验 7.4、实验 8.3、实验 8.4 的比较

Table 12 Comparison of experiment 7.3, experiment 7.4, experiment 8.3 and experiment 8.4

实验	与算力最大和最小的两个矿工 群体选择相同挖矿策略的其他 矿工(6%和 8%算力)的平均收益	与算力最大和最小的两个 矿工群体选择不同挖矿 策略的矿工的平均收益
实验 7.3	1.34	1.19
实验 7.4	1.06	1.13
实验 8.3	1.32	1.17
实验 8.4	1.10	1.09

比较实验 7.3 和实验 7.4(或实验 8.3 与实验 8.4),当挖矿策略种类、选择各挖矿策略的矿工算力之比、矿工们算力分布的极差的大小和大小矿工的划分标准 p 这 4 个参数均相同,且最大和最小算力的矿工挖矿策略相同时(见表 13),算力最大和最小的矿工选择策略 3 时,大矿工的收益较大,而算力最大和最小的矿工选择策略 1 时,小矿工的收益较大。

表 13 实验 7.3(实验 8.3)与实验 7.4(实验 8.4)的比较

Table 13 Comparison of experiment 7.3(experiment 8.3) with experiment 7.4(experiment 8.4)

实验	大矿工的收益	小矿工的收益
实验 7.3	0.26	1.33
实验 7.4	0.32	1.14
实验 8.3	0.92	2.20
实验 8.4	0.99	2.01

比较实验 7.1、实验 7.2 和实验 7.3、实验 7.4(实验 8.1、实验 8.2 和实验 8.3 实验 8.4),综合考虑了小矿工占比较大和较小的多种情况(见表 14)后发现:当选择策略 1 和策略 3 的矿工算力均等且矿工们算力分布的极差较大时;最大和最小算力的矿工挖矿策略相同,有利于小矿工提升平均收益,最大和最小算力的矿工群体选择的挖矿策略不同,有利于大矿工提升平均收益。

表 14 实验 7.1、实验 7.2(实验 8.1、实验 8.2)与实验 7.3、实验 7.4(实验 8.3、实验 8.4)的比较

Table 14 Comparison of experiment 7.1 and experiment 7.2 (experiment 8.1 and experiment 8.2) with experiment 7.3 and experiment 7.4 (experiment 8.3 and experiment 8.4)

实验	大矿工的收益	小矿工的收益
实验 7.1 和实验 7.2	0.31	1.20
实验 7.3 和实验 7.4	0.28	1.23
实验 8.1 和实验 8.2	0.97	2.12
实验 8.3 和实验 8.4	0.96	2.19

4.3 与 PoW 相较,改进的共识机制能激励小矿工

本节将对小矿工在 PoW 算法和改进共识算法中的收入,证明不论实验场景如何设置,改进的算法均能实现“小矿工在改进共识算法中的收益大于在 PoW 算法中获得的收益”,从而说明改进的共识机制有激励小矿工持续参与,在一定程度上解决“算力集中问题”的效果。

在 PoW 算法的分配机制下,不论大矿工还是小矿工,其挖矿成功后得到的收入均为 A ,而他们挖矿成功的概率与其算力成正比。由于期望收益 = 挖矿成功取得收入 * 挖矿成功的概率 = $A * \text{算力}$,因此大小矿工每次挖矿的期望收益都与其算力成正比。另一方面,在 PoW 算法下,每次的挖矿事件

是相互独立的,又根据 4.1 节中的假设 2,仅有一个子链,最先生成连续 3 个子区块的子接入主链,也就是 $N=3$,因此在 PoW 算法下生成这样一个 3 个区块相连的子链,各个矿工的期望收益 $= 3 * A * \text{算力}$ 。又根据 4.1 节中的假设 4,设每个取得记账权的区块确定收益的平均值 $A=1$,为了对比 PoW 算法和改进算法,将 PoW 算法中每次挖矿成功后获得的收益缩小一定程度后也不妨假设 $A=1$,那么在 PoW 算法中获得的期望收益 $= 3 * \text{算力}$ 。接下来比较各个实验场景下的各次重复实验中,小矿工在改进的共识机制中获得的收益是否大于 PoW 算法中获得的期望收益即 $3 * \text{算力}$,统计结果如表 15 所列。

表 15 小矿工在改进机制与 PoW 算法中的期望收入的比较

Table 15 Comparison of expected earnings of miners in improved mechanism and PoW algorithm

	改进共识机制中获得的收益 $>$ PoW 算法中获得的期望收益的次数	改进共识机制中获得的收益 \leq PoW 算法中获得的期望收益的实验次数
小矿工	5867	0

从表 15 中可以看出,不论哪个实验场景下的哪次重复实验,小矿工在改进的共识机制中获得的收入均大于 PoW 算法中获得的期望收入,从而证明改进的共识机制能激励小矿工并在一定程度上解决“算力集中问题”。

5 多子块激励共识机制的展望

基于 PoW 算法的多子块激励共识机制中仍有很多限制和不足,这些限制可以在不妨碍算法最终实现和算法设计初衷的情况下进行放松。目前的算法要求“各个子链中的所有子区块对应的矿工两两不可相同”,可将其放松为“每个子链中的子区块两两不可由同一矿工生成的,但同一矿工生成的区块可出现在不同的子链中”,这些子链包括此次成功连接到主链中的子链和未接入主链中的各条子链,如图 3 中矿工 B 可出现在子链 1、子链 2、子链 5—子链 8 中。这样的条件放松虽然有可能导致矿工充分利用现有算力来增大挖矿成功的概率,但能够加快区块链的生成速度,提高算力的利用效率。

对于算力集中的问题,改进算法虽然在一定程度上缓解了该矛盾,但仍然可能出现所有矿工扎堆挖矿的情况。例如 3.2 节中的讨论,所有矿工都采用相同策略进行挖矿,而此时满足策略要求的最长链或算力最小链就仅有一条,此即为所有矿工都集中在一个子区块后挖矿的极端情形。对于这样的问题,可以在算法中加入限制:每个子区块后按照报道的先后顺序,仅允许 m 个矿工进行挖矿或仅允许不超过算力之和为 m 的矿工们在其后进行挖矿。

改进的算法中并未考虑监督机制,可以加入监督机制来提高区块链的安全性。一种方法是参考 PBFT 的检查点协议^[28],根据各节点以往的表现对每个节点的状态进行标记,将其分为良好、正常、异常和恶意节点 4 类,从而及时剔除恶意节点。良好节点的权限最高,多次产出的区块有效,可获得记账权并参与到检查点协议中对其余节点进行标记;正常节点的状态为初始状态,同样可获得记账权并参与到检查点协议中;异常节点是那些曾经产出过的无效区块,但产出的区块

个数并未超过标准的节点,它们仅有参与检查点协议的权限;恶意节点是产出过无效区块并且产出的区块个数超过标准的节点,它们没有任何权限。另一种方法是加入监督委员会,选出一些代表节点对整个网络进行监督,并对检举恶意节点并证实其检举的节点确为恶意节点的矿工进行激励。

3.1 节曾对参数 $p, f(x)$ 和 $g(x)$ 的设置进行讨论,当满足条件的参数组合并不唯一时,可以放松对参数 $p, f(x)$ 和 $g(x)$ 的设置要求,将这些参数设置成随着网络状况动态变化的函数,从而调节区块的生成速度、生成质量等。

由于对 3.2 节的挖矿策略尚无实际数据进行分析,因此第 4 节得出的各矿工的最佳博弈策略不一定符合现实。期望未来能够融入各矿工的心理和行为,设计出市场情景更符合真实市场的、市场情景更多样的仿真算法,以期进一步分析其他参数对博弈的影响并使分析结果更可靠。

结束语 本文在 PoW 算法的基础上改变了矿工所挖出区块接入主链的条件,创造性地用更为复杂的网状结构来代替原有的单一区块连接的结构。该共识机制改进了以往确定收益的分配策略来激励小矿工积极参与挖矿,抑制大矿工进行恶意攻击,同时可通过调节参数来达到以上目的,保障区块链的安全。所提机制虽然使得每次生成区块过程中耗费的算力增加,但每次生成的区块数量也成比例增加,因此对区块链的生成效率并未有太大影响。本文设计的基于 PoW 算法的多子区块激励的共识机制,是对目前区块链主流共识机制的一种改进,将对区块链共识机制的发展起到促进作用。本文还对各个矿工的挖矿策略、恶意矿工分拆算力和合谋等进行了分析,并对改进的算法进行了仿真实验,讨论了大小矿工和选择不同挖矿策略的矿工的期望收益受挖矿策略的种类、选择各挖矿策略的矿工算力之比、矿工们的算力分布的极差的大小、大小矿工的划分标准 p 等参数的影响,接下来将尝试对参数进行多重约束来解决分拆算力和合谋的问题。由于本文涉及的参数众多,各矿工挖矿策略的博弈与矿工的心理有关,期望后期收集调查数据做进一步分析改进。

参考文献

- [1] NAKAMOTO S. Bitcoin: A peer-to-peer electronic cash system [EB/OL]. (2008-11-01) [2018-09-30]. <https://bitcoin.org/bitcoin.pdf>.
- [2] LIU M X, GAN G H, CHENG Y K, et al. The development status and prospect of blockchain consensus mechanism[J]. Operations Research Transactions, 2020, 24(1): 23-39.
- [3] DU M X, MA X F, ZHANG Z, et al. A review on consensus algorithm of blockchain[C]//2017 IEEE International Conference on Systems, Man and Cybernetics (SMC). Banff, 2017: 2567
- [4] QuantumMechanic. Proof of stake [EB/OL]. (2011-07-11) [2018-09-30]. <https://bitcointalk.org/index.php?topic=27787>.
- [5] LARMER D, KASPER L, SCHUH F. BitShares 2.0: Financial smart contract platform [EB/OL]. (2015-11-01) [2018-09-30]. <http://docs.bitshares.eu/downloads/bitshares-financial-platform.pdf>.
- [6] BENTOV I, LEE C, MIZRAHI A, et al. Proof of activity: Extending bitcoin's proof of work via proof of stake [J]. ACM

- SIGMETRICS Performance Evaluation Review, 2014, 42(3): 34-37.
- [7] RENL. Proof of stake velocity; Building the social currency of the digital age [EB/OL]. (2018-04-10) [2018-09-30]. <https://assets.coss.io/documents/whitepapers/reddcoin.pdf>.
- [8] Intel. Proof of elapsed time [EB/OL]. (2016-12-16) [2018-09-30]. <https://intelledger.github.io/introduction.html>.
- [9] JUELS A, KALISKI B S. PORs: Proofs of retrievability for large files [C] // Proceedings of the 14th ACM Conference on Computer and Communications Security. Alexandria: ACM, 2007: 584-597.
- [10] MILLER A, JUELS A, SHI E, et al. Permacoin: Repurposing bitcoin work for long-term data preservation [C] // 2014 IEEE Symposium on Security and Privacy. IEEE Computer Society, 2014, 1: 475-490.
- [11] GILAD Y, HEMO R, MICALI S, et al. Algorand: Scaling byzantine agreements for cryptocurrencies [C] // Proceedings of the 26th Symposium on Operating Systems Principles. Shanghai, 2017: 51.
- [12] CASTRO M, LISKOV B. Practical byzantine fault tolerance [C] // Proceedings of the Third Symposium on Operating Systems Design and Implementation. New Orleans: ACM, 1999: 1-10.
- [13] KOTLA R. Zyzzyva: speculative byzantine fault tolerance [J]. ACM SIGOPS Operating Systems Review, 2007, 41(6): 45-58.
- [14] AUBLIN P L, MOKHTAR S B, QUEMA V. RBFT: Redundant byzantine fault tolerance [C] // 2013 IEEE 33rd International Conference on Distributed Computing Systems. Washington: IEEE Computer Society, 2013: 297-306.
- [15] MILLER A. The honey badger of BFT protocols [C] // Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications. Vienna: ACM, 2016: 31-42.
- [16] SOMPOLINSKY Y, ZOHAR A. Secure high-rate transaction processing in bitcoin [C] // International Conference on Financial Cryptography and Data Security. Heidelberg: Springer, 2015: 507-527.
- [17] EYAL I, GENCER A E, SIRER E G, et al. Bitcoin-NG: A scalable blockchain protocol [C] // Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation. 2016: 45-59.
- [18] SOMPOLINSKY Y, LEWENBERG Y, ZOHAR A. Inclusive block chain protocols [C] // International Conference on Financial Cryptography and Data Security. Heidelberg: Springer, 2015: 528-547.
- [19] ZHEN Y, YUE M, YU C Z, et al. Zero-determinant strategy for the algorithm optimize of blockchain PoW consensus [C] // 2017 36th Chinese Control Conference (CCC). IEEE, 2017: 1441-1446.
- [20] ZHANG R, PRENEEL B. Lay Down the Common Metrics: Evaluating Proof-of-Work Consensus Protocols' Security [C] // 2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019: 175-192.
- [21] KUMAR G, SAHA R, RAI M K, et al. Proof-of-Work Consensus Approach in Blockchain Technology for Cloud and Fog Computing Using Maximization-Factorization Statistics [J]. IEEE Internet of Things Journal, 2019, 6(4): 6835-6842.
- [22] BAHACK L. Theoretical Bitcoin attacks with less than half of the computational power (draft) [J]. arXiv:1312.7013, 2013.
- [23] LERNER S D. DECOR+HOP: A scalable blockchain protocol [EB/OL]. <https://scalingbitcoin.org/papers/DECOR-HOP.pdf>.
- [24] CAMACHO P, LERNER S D. DECOR+LAMI: A scalable blockchain protocol [EB/OL]. <https://scalingbitcoin.org/he/papers/DECOR-LAMI.pdf>.
- [25] PASS R, SHI E. Fruitchains: A fair blockchain [C] // Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC'17). ACM, 2017: 315-324.
- [26] RIZUN P R. Subchains: A technique to scale Bitcoin and improve the user experience [J/OL]. <https://www.ledgerjournal.org/ojs/index.php/ledger/article/view/40>.
- [27] ROBERT A, HAMILTON W D. The Evolution of Cooperation [J]. Science, 1981(211): 1390-1396.
- [28] HUANG Q B, AN Q W, SU H Q. Study and realization of an improved PBFT algorithm as an ethereum consensus mechanism [J]. Computer Applications and Software, 2017, 34(10): 288-293, 297.



LIU Shuai, born in 1993, postgraduate. Her main research interests include blockchain and financial engineering and risk management.



FANG Yong, born in 1974, Ph.D, research associate, Ph.D supervisor. His main research interests include financial engineering and risk management and operations management.