计算机科学
COMPUTER SCIENCE

基于动态分组的重要性共识优化算法

王冬，肖冰冰，金晨光，李政，李笑若，祝丙南

引用本文

王冬，肖冰冰，金晨光，李政，李笑若，祝丙南基于动态分组的重要性共识优化算法[J]．计算机科学，2022，49(12)：362-367.

WANG Dong，XIAO Bing-bing，JIN Chen-guang，LI Zheng，LI Xiao-ruo，ZHU Bing-nan. Consensus Optimization Algorithm for Proof of Importance Based on Dynamic Grouping [J]. Computer Science，2022，49(12)：362-367.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.211100282

基于动态分组的重要性共识优化算法

王　冬[1,2]　肖冰冰[1,2]　金晨光[1]　李　政[1,2]　李笑若[1]　祝丙南[1]
1 河南大学软件学院　河南 开封 475001
2 河南省智能网络理论与关键技术国际联合实验室　河南 开封 475001
(Juliawdd@qq.com)

摘　要　权益证明共识算法(PoS)虽然具有不需要花费算力的优势,然而由于权益越高的节点获得记账权的可能性越大,因此记账节点具有很强的确定性且容易富者愈富,一旦权益最高的节点无法正常记账出块,其余节点仍要重新竞争记账权,此时系统停滞的概率急剧增大。针对这两个缺陷,提出了一种基于动态分组的重要性共识优化算法(DPoI)。首先,算法引入重要性评估方案,依据节点活跃度、交易占比、寻找随机数的时间和信誉度计算每轮中节点的重要性分数 iValue;然后,利用斐波那契数列将 iValue 相近的节点动态分组,组内借鉴 DPoS 投票策略排名充当备选节点,形成灾备方案,从而有效避免系统停滞;最后,设计了二进制指数退避算法来快速剔除系统中的恶意节点,从而有效增强了区块链系统的安全性和稳定性。实验结果表明,DPoI 出块的速度约为 PoI 的 6 倍,大大加快了出块速度。当恶意节点占比达到 70%时,二进制指数退避算法仍能有效剔除恶意节点,系统的可靠性得到了充分保障。

关键词:区块链;动态分组;重要性证明;信誉度;DPoS
中图法分类号　TP309

Consensus Optimization Algorithm for Proof of Importance Based on Dynamic Grouping

WANG Dong[1,2],XIAO Bing-bing[1,2],JIN Chen-guang[1],LI Zheng[1,2],LI Xiao-ruo[1] and ZHU Bing-nan[1]
1 School of Software,Henan University,Kaifeng,Henan 475001,China
2 Henan International Joint Laboratory of Intelligent Network Theory and Key Technology,Kaifeng,Henan 475001,China

Abstract　Proof of stake consensus algorithm(PoS) has the advantage of not requiring arithmetic power. However,the higher the equity of the node,the higher the probability of obtaining the bookkeeping rights,resulting in a very deterministic bookkeeping node and makes it easy for the rich to get richer. Once the node with the highest equity fails to book the block properly,the rest of the nodes still have to compete for the bookkeeping rights again. The probability of system stagnation increases dramatically at this point. To address these two shortcomings,a consensus optimization algorithm for proof of importance based on dynamic grouping(DPoI) is proposed. The algorithm introduces an importance assessment scheme,which calculates the importance score iValue of nodes in each round based on node activity,transaction share,time to find random numbers and reputation. Then,the nodes with similar iValue are dynamically grouped using Fibonacci series. Within the group,the DPoS voting strategy ranking is borrowed to act as an alternative node,thus forming a disaster recovery scheme to effectively avoid system stagnation. Finally,a binary exponential backoff algorithm is designed to quickly remove malicious nodes from the system,thus effectively enhancing the security and stability of the blockchain system. Experimental results show that the speed of DPoI block-out is about 6 times faster than PoI,which significantly improves the block-out speed. When the percentage of malicious nodes reaches 70%,the binary exponential backoff algorithm can still effectively reject malicious nodes,and the reliability of the system is fully guaranteed.

Keywords　Blockchain,Dynamic grouping,Proof of Importance,Credit,DPoS

1　引言

区块链本质上是一种分布式数据库,具有去中心化、不可篡改、可追溯、多方共同维护等特点[1]。它利用数字加密、共识算法、分布式存储、P2P 协议等多种技术,维护全网数据的一致性和有效性[2-4]。应用区块链技术可以在无信任基础的

到稿日期:2021-11-29　返修日期:2022-05-27
基金项目:国家自然科学基金面上项目(61872125);河南省自然科学基金(192102210271);基于鲲鹏平台的国产操作系统研究与示范(201300210400),2020 年度河南省重大科技专项
This work was supported by the National Natural Science Foundation of China(61872125),Henan Natural Science Foundation(192102210271),Research and Demonstration of Domestic Operating System Based on Kunpeng Platform(201300210400) and Major Science and Technology Special Projects in Henan Province in 2020.
通信作者:肖冰冰(xiaobingkf@qq.com)

多方间不通过第三方机构就实现可信、对等的价值传输。由于 P2P 网络中不同节点之间的网络传输速率不同，存在较高的网络延迟，因此各个节点所收到的信息存在一定的差异。在基于区块链技术的应用中，最核心的问题就是如何在去中心的前提下保证数据的一致性。

共识算法是区块链的核心技术[5-6]，旨在解决拜占庭将军问题，使各节点在没有中心管理机构的情况下遵循一定的规则实现自治，达成最终一致性。共识算法的优劣直接影响着区块链应用系统的安全和性能。比特币中基于算力的 PoW 机制[7]具有完整的数学证明，它不依赖节点的数量达成共识，成为了对抗女巫攻击[8]的主要手段。然而，以 PoW 为基础的加密货币可能会遭受双重支付攻击。为降低风险，交易的完成通常需要 6 个确认区块，这使得攻击者可以发起很多低价值交易来冲击网络，让拒绝服务攻击（DoS）[9]成为可能。2015 年 7 月就发生过一次针对比特币网络的洪泛攻击[10]。另外，PoW 资源浪费严重，若节点将彼此的算力联合组成矿池，易出现算力中心化和 51%攻击等问题[11]。针对 PoW 中的问题，Hanke 在 2012 年提出了权益证明共识算法（Proof of Stake，PoS）[12]，该算法依据权益来决定节点获得记账权的概率，缩短了共识达成的时间，并减少了资源的浪费，但会出现无成本权益攻击（Nothing of Stake）[13]，产生分叉并导致富者愈富和权益粉碎攻击[14-15]。

随着区块链技术的进一步发展，新的共识机制层出不穷。新经济运动（New Economy Movement，NEM）提出了独特的重要性证明算法（Proof of Importance，PoI）[16]，节点所持权益不再是重要性的主要因素。但因各种环境的影响，节点无法保证持续稳定在线，可能会出现重要性最高的节点因故障离线而无法正常打包记账的问题。在 PoI 中也存在如下风险：1）重要性排名靠后的节点几乎没有获得记账权的概率，不积极参与全网广播；2）任何一个节点都可以参与到共识中，存在某节点故意发起多笔无效交易以进行权益粉碎攻击。DPoS 也存在恶意节点给自己投票的情况，从而出现腐败贿赂的现象[17]。一旦恶意节点在区块生产链中只是为了获得更多的利益而没有能力去生产区块，则会导致区块生产能力降低，所有节点的收益都将受到损害。基于重要性证明（PoI）的区块链共识机制需要计算节点的重要性分数，并由重要性分数最高的节点获得新区块的记账权。PoI 共识算法规定[16]，当记账节点无法正常出块时，系统中其他节点将始终处于挂起等待状态，由此系统陷入停滞，无法继续运行。本文提出的基于动态分组的重要性共识优化算法（Dynamic Grouping Based Proof of Importance，DPoI）为 PoI 在无法正常出块时提供了一种灾备方案，其利用动态分组的方法能够有效解决 PoI 存在的系统陷入停滞的问题。

本文针对 PoI 存在的记账节点故障的问题进行改进，提出了一种基于动态分组的重要性共识优化协议（DPoI）。本文的主要贡献如下：

（1）引入 Ltime，iTrade，aValue 和 Credit 这 4 个因素来动态评估节点的重要性。设计了节点信誉度评估方案，并且讨论了在不同场景下的信誉评估函数，以便激励节点正常出块。

（2）根据节点的重要性大小，按排名依次将节点划分进具有斐波那契数列特性的小组[18]。组内通过 DPoS 投票策略投票选出负责记账的“替补节点”，为记账节点故障提供了一种灾备方案，增强了区块链系统的可靠性，实现了分组共识，提高了共识效率。

（3）引入二进制退避算法，作为节点作恶的剔除方式，用于及时处理恶意节点，提高区块链系统的安全性。

实验结果证明，DPoI 通过斐波那契数列分组和 DPoS 的投票方式，能够使 PoI 的平均出块时间由 60 s[16]缩短至 10s 左右，并且出块速率稳定；二进制退避算法使恶意节点能够被及时惩罚和剔除，增强了系统的鲁棒性和可靠性。

本文第 2 节介绍基于动态分组的重要性共识机制改进的相关工作；第 3 节介绍基于动态分组的 DPoI 改进与实现策略；第 4 节进行实验和分析；最后总结全文。

2　相关工作

2.1　权益证明共识算法（PoS）

PoS[19]把节点的币龄（Coin Age）作为竞争记账权的依据，节点的币龄越大，越容易获得记账权，产生区块。相比 PoW，PoS 是一种出块效率更高、更节约算力资源的共识算法，它使得区块链系统无需高昂的硬件和电力挖矿成本就能正常运行。

为解决 PoS 中需要全网节点参与共识导致共识时间过长这一问题，比特股（Bitshares）项目在 2013 年 8 月提出了代理权益证明共识算法（Delegated Proof-of-stake，DPoS）[20]。DPoS 由持币人投票选举若干代表节点参与共识，并由这些代表节点记账和验证。相比 PoS，DPoS 机制中竞争记账权的节点数大大减少，记账同步实现更快。因此 DPoS 具有能耗更低、出块速度更快的优点。

2.2　重要性证明共识算法（PoI）

PoI 根据交易量、活跃度等因素评估出每个节点的重要性分数，得分越高，获得记账权和代币奖励的机会就越大，平均每 60s 打包一个区块。由于 PoI 中部分因素周期性归零，而非 PoS 中所有因素持续不断地累积，因此有效解决了 PoS 中富者愈富的问题。

无论是依赖币龄的 PoS 还是依赖重要性分数的 PoI，都会导致记账节点的确定性过强，容易出现单点故障问题，严重时会导致系统崩溃。文中将重要性设定为一个可以动态调节的值，提高了记账节点的不可预测性，避免了富者愈富的问题，并为单点故障问题提供了灾备方案。

3　DPoI 共识算法设计

3.1　DPoI 重要性评估策略

DPoI 利用节点寻找随机数的时间，以及节点的活跃度、交易量和信誉值计算节点的重要性分数。

（1）引入 SHA256 哈希函数计算节点寻找随机数的时长，以增强算力较强节点的重要性。节点寻找随机数的时间越长，说明其算力越差，因此重要性越小。当某节点找到 Nonce 后，立即向全网广播。为降低哈希计算的难度，减少

寻找随机数所花费的算力，将上一个区块中的时间戳的后 4 位数字设置成哈希计算中所要求的随机数(Nonce)，将找到 Nonce 的时长占比 *Ltime* 作为评估重要性分数的一个因素。将前 80%的节点的 *Ltime* 记为该节点所花费时长占最后一名花费时长的比例；将后 20%的节点的 *Ltime* 记为 1。

(2)引入上一轮的活跃度和交易占比，以全面评估节点在本轮共识中的重要性。节点发起一笔交易以后，需立即向全网广播。在节点同意该交易的基础上，这些节点会将这笔交易再次向全网广播。在一个共识轮次中节点参与的交易量记为该节点的交易量，在一个共识轮次中全网总共的交易量记为总交易量。在每一轮共识中，将节点参与广播的次数占系统总广播次数的比例作为活跃度，记为 *aValue*；将节点参与交易量占系统总交易量的比例作为交易占比，记为 *iTrade*。

(3)将信誉度(*Credit*)引入到重要性评估方案中，以降低恶意节点获得记账的概率。节点的初始信誉值设置为 0.5。信誉度增长过快不利于合理地判断节点信誉的增长，为了避免这种情况，本文提出了对 logistic 回归模型产生的信誉度进行修正的算法，根据节点的历史行为，对在投票、记账和举报时的信誉度进行权重均衡。基于 logistic 回归模型提出 *Credit* 动态度量公式如下：

$$Credit=\frac{1}{1+e^{-(vCredit\cdot rCredit+aCredit)}} \tag{1}$$

其中，*vCredit*，*aCredit*，*rCredit* 是分别依据投票信誉函数、记账信誉函数和举报信誉函数计算出的投票信誉值、举报信誉值和记账信誉值。

(1)投票信誉函数

为提高节点投票的积极性，设立了投票信誉函数。根据每个共识轮次中参与投票节点的数量，计算节点的投票信誉值，参与投票的节点越多，*vCredit* 越小，对 *Credit* 的增加则有促进作用，以此来激励多数节点参与投票。在拜占庭容错算法(PBFT)中，至多可以容忍不超过系统全部节点数量 1/3 的拜占庭节点“背叛”，即如果超过 2/3 的节点正常，整个系统就可以正常工作。借鉴 PBFT 算法，本文规定，随着投票次数的减少，节点所获收益也越少。投票信誉的定义如下：

$$vCredit=\begin{cases}\sum_{i=1}^{k}X_i-\sum_{i=1}^{n}Y_i, & m\geqslant\frac{2}{3}n\\ (n-m)/n, & m<\frac{2}{3}n\end{cases} \tag{2}$$

其中，m 是当前轮次中参与投票的节点数；n 是当前轮次中总结点数；k 是从系统开始运行时已完成的共识轮数；X_i 是一个节点在第 i 轮共识中正常投票的次数，一个节点组内可进行多次投票，直到选出记账节点，组内轮流结束；Y_i 是该节点在第 i 轮共识中不投票的次数。

(2)记账信誉函数

为降低恶意节点成功创建区块的概率，设立了记账信誉函数以评估本轮的记账信誉值。将该节点在第 i 轮中是否成功记帐的行为，作为计算 $i+1$ 轮次中节点信誉值的影响因素。记账信誉的定义如下：

$$aCredit=\begin{cases}1, & \text{第 } i \text{ 轮次中非记账节点}\\ M/T_{i-1,i}, & \text{第 } i \text{ 轮次中记账节点}\end{cases} \tag{3}$$

其中，每个共识轮次产生一个区块；第 i 轮次中非记账节点的 *aCredit* 统一设置为 1；记账节点成功记账，M 的值为 1，否则为 0。$T_{i-1,i}$ 表示第 i 轮出块时间(单位：min)。

(3)举报信誉函数

为避免恶意节点在验证阶段联盟作恶，设立了举报函数奖励以举报节点和惩罚贿赂节点。本方案控制奖励力度大于贿赂力度的策略，以降低贿赂成功率。若节点发现贿赂节点并进行举报，系统将会增加举报节点的 *rCredit*，即举报奖励。相应地，贿赂节点也将受到处罚。验证节点举报信誉度量的表达式如下：

$$rCredit=\begin{cases}\alpha\cdot i, & \text{举报节点}\\ 0, & \text{不举报不贿赂节点}\\ -\beta\cdot m, & \text{贿赂节点}\end{cases} \tag{4}$$

其中，α 表示用户自定义的举报奖励力度；i 为节点历史举报次数；β 表示节点的贿赂惩罚力度；m 表示节点历史贿赂次数。β 的变化是基于贿赂节点的每个轮次 *Credit* 在 *iValue* 中的占比，β 具体变化的公式如下：

$$\beta=Credit/iValue \tag{5}$$

其中，*Credit* 表示该节点上轮共识中的信誉度；*iValue* 表示上轮共识中的重要性分数。

重要性评估流程：各节点按照当前设置的难度值通过哈希计算寻找 Nonce，当每个节点找到 Nonce 时立刻向全网广播，然后找到随机数的节点，并结合 *Ltime*，*aValue*，*iTrade* 和 *Credit* 这 4 个值计算出该节点的重要性分数 *iValue*。*iValue* 的计算式为：

$$iValue=aValue+iTrade+Credit-Ltime \tag{6}$$

Ltime 降低了算力较强的节点获得记账权的概率，增大了信誉值、交易量和活跃度的比重，动态评估了节点的重要性，强化了信誉值对记账权竞争的影响，有利于减少矿工节点作恶行为的发生。灾备方案的流程如图 1 所示。

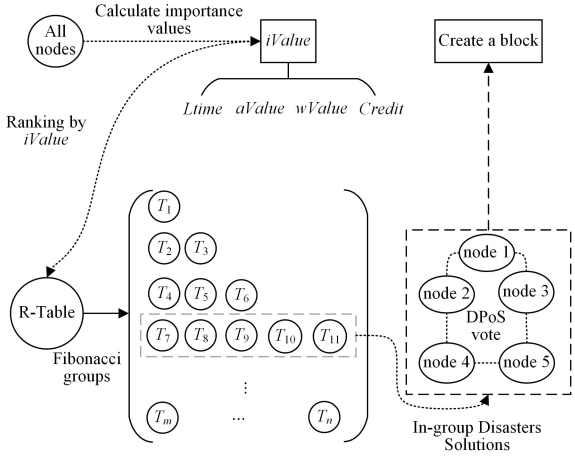


图 1 灾备方案示意图

Fig. 1 Diagram of disaster recovery solution

3.2 斐波那契分组策略

在第一轮的 DPoI 取得共识后，各节点在全网广播自己的重要性分数 *iValue*。系统根据节点的 *iValue* 由高到低排名，并记录到 R-Table 列表中。得票最高的节点记账时，系统通过设定的计时监测机制来监测记账节点是否超时，同时验证节点会负责对最后账本的验证，若超时或账本验证不通过，

则证明记账节点无法记账，其余节点通过 gossip 协议最终得知记账节点无法成功记账。如果仅根据节点的排名确定记账权，当首节点出现故障时，系统将停滞。为解决此问题，本文利用斐波那契分组提供一种灾备方案。去掉斐波那契数列中的第一项，斐波那契数列中的每一项的数值代表分组中的节点个数。例如，斐波那契数列中的第一项是 1，则将重要性排名第一的节点放进第一组；斐波那契数列中第二项是 2，则将重要性排名第二、第三的节点放进第二组；斐波那契数列中第三项是 3，则将重要性排名第四、第五、第六的节点放进第三组，以此类推。

在组内采取 DPoS 投票策略选拔出备选节点，任何一个 *Credit* 大于 0 的用户都可以参与到组内投票和竞选记账人这两个过程中。每隔一段时间，组内节点可以通过把票投向自己认可的节点来记账。如果被投票节点作恶，组内节点可以随时撤销对该节点的投票，每个节点投票的权重和自己的 *iValue* 成正比。投票和撤票可以随时进行，在投票选举结束后，组内得票最高的节点成为记账节点，若该节点无法成功记账，则得票次之的节点成为记账节点，以此类推，形成无法记账的灾备方案。若得票相同，*Credit* 高的节点先获得记账权；得票为 0，则失去本轮记账权。若本组内选出的备选节点都无法正常记账，便按照此方法在下一组中选出记账节点，直到节点成功记帐。由于第 i 组的备选节点记账失败后到第 $i+1$ 组投票时存在一个等待时间，为减少这个等待时延，在第 i 组的备选节点记账时，第 $i+1$ 组的备选节点也同时被投票选出。记账之后，指定 *Credit* 排名前 50%的节点为验证节点。重要性排名靠前的节点相对比较可靠，因此系统中设定固定值前 50%的节点为验证节点，以此来减少全部节点验证的时间消耗，提高验证效率。为降低恶意节点签署无效交易的概率，打包的区块需要所有验证节点都确认。

基于重要性分数利用斐波那契原理对节点进行分组，重要性越大的节点将被分在靠前的分组中以优先处理，提高了成功记账的概率，进而提高了系统的稳定性和可靠性。当记账失败后，灾备方案能够立刻找到新的记账节点，进行记账的无缝衔接，有效避免了系统停滞。

3.3　计算恶意节点记账等待时延

在 PoI 机制中，故障节点及恶意节点获得记账权导致系统无法正常出块后，在下轮共识中获得记账权的概率变化不大。为降低恶意节点记账的概率，本文引入二进制指数类型退避策略，根据节点失败记账的历史行为给节点设置一个等待轮数。当节点获得记账权后，将等待一定轮数后再记账，等待轮数呈指数增长，从而增加恶意节点记账时间，降低恶意节点的记账信誉 *aCredit*。

退避算法的流程如图 2 所示。

退避算法的具体步骤如下：

(1)确定基本退避轮数，作为等待轮数 x。该轮次前的失败记账次数记录为等待轮数 x。

(2)从离散的整数集合[0,1,…,$2k-1$]中随机取一个数，记为 r，等待时延就是 r 倍的等待轮数。上面的参数 K 按公式 $K=Min$[失败记账次数,5]计算。当失败次数不超过 5 时，参数 K 等于失败次数；但当失败次数超过 5 时，K 将不再增大而是一直等于 5。

(3)当失败次数达到 5 次仍不能成功记帐时，则剔除该节点。

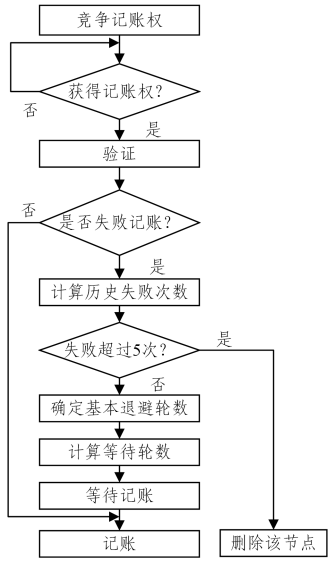


图 2　退避算法流程图

Fig. 2　Flow chart of avoidance algorithm

4　实验结果与分析

4.1　实验环境

实验模拟出 100 个节点对 DPoI 共识算法进行验证。初始时，节点的信誉值为 0.5。通过搭建验证模型，对系统出块的稳定性和安全性进行实验验证，分析 DPoI 算法中重要性动态评估策略、分组策略提供的灾备方案、恶意节点退避算法的有效性和可靠性。

4.2　出块稳定性分析

在 DPoI 中，依据重要性动态评估策略和分组策略，动态地选出记账节点和灾备节点，在系统中存在 50%恶意节点的情况下，通过 50 轮共识来分析 DPoI 的出块时间，其中每轮共识所花费的时间如图 3 所示。

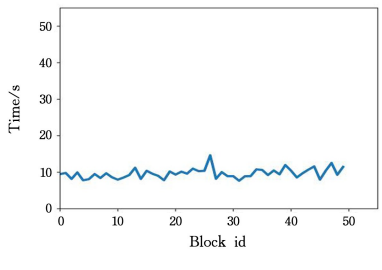


图 3　50 个区块的出块时间

Fig. 3　Out time of 50 blocks

从实验结果可以看出，图 3 是通过统计的方式记录下每创建一个区块的时间。在 PoI 共识机制下出块时间大约为 60 s 左右，而在 DPoI 共识机制下出块时间略有波动，但都稳定在 10 s 左右。这是因为在 DPoI 中，依据 *iValue* 排名按斐波那契分组策略分组后，系统将选出两组节点作为记账的灾备节点。当记账节点无法成功记账时，灾备节点将会立即替补，节点无须再重新计算 *iValue* 和分组，减少了时间的浪费。因此 DPoI 中的灾备方案能够有效保证系统

稳定出块，提高了系统的稳定性。

4.3 出块概率分析

通过斐波那契分组策略对节点进行分组，可使重要性排名高的节点被分配到组号靠前的小组中。为分析各组节点获得记账的概率，将 100 个节点按重要性分数排名，并依据分组策略分组后，统计前 10 组获得记账权的概率，如图 4 所示。

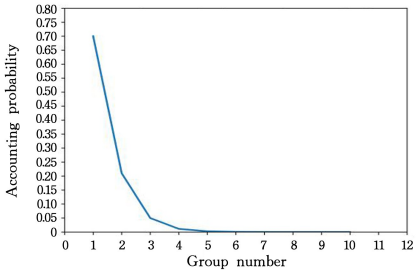


图 4 各组记账的概率

Fig. 4 Probability of each group's accounting

记账权概率计算式如下：

$$p_i = p(1-\bar{p}_{i-1}) \tag{7}$$

其中，p 表示节点成功记账的概率，p_i 表示第 i 组节点成功记账的概率，$\bar{p}_{i-1}$ 表示第 $i-1$ 组记账不成功的概率。

排名前十的节点获得记账权的概率远大于排名靠后的节点。DPoI 采用斐波那契分组策略将节点分组后，在保证各组中节点获得记账权的概率依然遵循重要性排名的同时，利用 DPoS 投票策略增加了节点获得记账权顺序的不确定性。排名靠前的节点获得记账权的概率依旧可以得到充分保障，系统中其他节点获得记账权的概率也会因其所在分组的位置而明显不同。因此，DPoI 共识算法中的分组策略能够在尊重节点重要性排名的同时，增加组内节点获得记账权的不可预测性，减少了系统中的贿赂攻击。

4.4 安全性分析

分别选取 30%，50%，70%的节点作为恶意节点，在实验初始将 100 个节点中编号后 30%的节点设定为恶意节点，它们正确或恶意参与区块的创建设定为随机。系统在经过 100 轮共识后，记录下恶意节点成功记账的次数和剩余的节点总数，如图 5 所示。系统中总节点数量变化如图 6 所示。图 5 是通过统计的方式，计算获得记账权的恶意节点并成功记账的次数。图 6 是在每轮共识后，通过统计的方式计算系统中剩余恶意节点的数量。根据图 5、图 6 综合分析恶意节点占比对正常记账的影响。

从图 5、图 6 可以看出，当恶意节点占比从 30%增加到 70%时，恶意节点成功记账的次数由 6 减少到 1，二进制退避算法也使得恶意节点被剔除的速度越来越快。经过 40 轮共识后，节点作恶被及时惩罚和剔除，恶意节点信誉值变得极低，后续获得记账权的概率极小，这恶意节点的剔除也开始趋于缓慢，系统开始趋于稳定。

因此，即使恶意节点获得了记账资格，但参与验证的节点验证不通过，恶意节点成功记账的概率也极小，这极大地降低了恶意节点通过联盟获得成功记账的可能性；当恶意节点想要通过联盟作恶时，恶意节点的增多会使得恶意节点的信誉值快速下降，节点的信誉值越低，节点恶意参与区块的创建将会得不偿失。

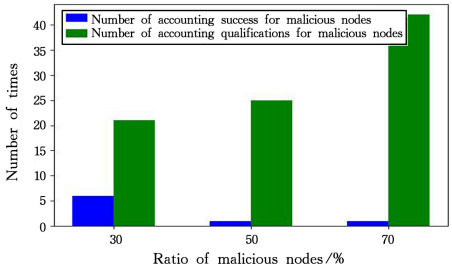


图 5 恶意节点获得记账权后记账成功的次数

Fig. 5 Number of successful bookkeeping after malicious section acquires bookkeeping rights

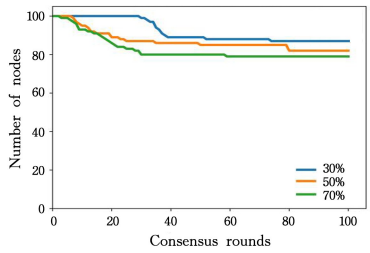


图 6 每轮共识后恶意节点个数

Fig. 6 Number of malicious nodes after each consensus round

结束语 共识算法作为区块链的核心，解决了分布式一致性的问题。通过对现有共识算法的研究和改进，本文提出了基于动态分组的重要性共识优化算法。该算法是针对重要性共识算法提出的灾备方案，并且通过实验对改进的共识机制的可行性和性能进行了分析和验证。实验结果证明，DPoI 通过不同的信誉函数机制解决了节点冷漠问题；斐波那契数列分组和 DPoS 投票策略则提高了出块速率的稳定性；二进制退避算法方案使恶意节点能够被及时惩罚和剔除，增强了系统的鲁棒性和可靠性。

下一步的工作是将包括加密算法在内的安全机制有效融入区块链系统，以提高其安全性。

参 考 文 献

[1] YUAN Y, WANG F Y. Blockchain: thestate of the art and future trends[J]. Acta Automatica Sinica, 2016, 42(4): 481-494.

[2] XIE P, SHI W G. Digital cryptocurrency research: a literature review[J]. Financial Research, 2015, 415(1): 1-15.

[3] YUAN Y, NI X C, ZENG S, et al. The development status and outlook of blockchain consensus algorithm[J]. Journal of Automation, 2018, 44(11): 2011-2022.

[4] XU F, YANG G W, JU D P. Design of Distributed Storage System on Peer-to-Peer Structure based on Peer-to-Peer [J]. Journal of Software, 2004, 15(2): 268-277.

[5] NAKAMOTO S. Bitcoin: A Peer-to-Peer Electronic Cash System[EB/OL]. https://blog.csdn.net/yingkee/article/details/53888910.

[6] CHO H. ASIC-Resistance of Multi-Hash Proof-of-Work Mechanisms for Blockchain Consensus Protocols[J]. IEEE Access,

2018,6:66210-66222.

[7] VUKOLI M. The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication[C]// International Workshop on Open Problems in Network Security. Springer International Publishing,2016.

[8] WANG J. Research on resource management mechanisms in P2P systems [D]. Hefei: University of Science and Technology of China,2007.

[9] LI D Q. Denial of Service Attacks [M]. Beijing: Electronic Industry Press,2007.

[10] July 2015 flood attack[EB/OL]. Bitcoin Wiki. https://en.bitcoin.it/wiki/July_2015_flood_attack.

[11] LI W,ANDREINA S,BOHLI J M,et al. Securing Proof-of-Stake Blockchain Protocols[C]// European Symposium on Research in Computer Security International Workshop on Data Privacy Management Cryptocurrencies and Blockchain Technology. 2017.

[12] HANKE T. Asicboost-a speedup for bitcoin mining[J]. arXiv: 1604.00575,2016.

[13] HOU Y N. It Will Cost You Nothing to 'Kill' a Proof-of-Stake Crypto-Currency[J]. Social Science Electronic Publishing,2014,34(2):1038-1044.

[14] POELSTRA A. Distributed consensus from proof of stake is impossible[EB/OL]. https://download.wpsoftware.net/bitcoin/pos.pdf.

[15] BUTERIN V. On stake[EB/OL]. https://blog.ethereum.org/2014/07/05/stake/.

[16] BEIKVERDI A. NEM technical reference [EB/OL]. https://nem.io/wp-content/themes/nem/files/NEM_techRef.pdf.

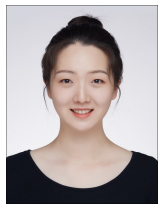
[17] FANG W,ZHANG W,PAN T,et al. Cyber Security in Blockchain: Threats and Countermeasures[J]. Journal of Cyber Security,2018,3(2):87-104.

[18] ZHOUC Z. The Fibonacci-Lucas sequence and its applications [M]. Changsha: Hunan Science and Technology Press,1993.

[19] YANG J,PAUDEL A,GOOI H B,et al. A Proof-of-Stake Public Blockchain-Based Pricing Scheme for Peer-to-Peer Energy Trading[J]. Applied Energy,2021,298:117154.

[20] LARIMER D. Delegated Proof-of-Stake(DPOS) [EB/OL]. https://bitshares.org/technology/delegated-proof-of-stake-consensus/2014.

WANG Dong, born in 1977, Ph.D, professor, is a member of China Computer Federation. Her main research interests include blockchain and its applications.



XIAO Bing-bing, born in 1997, postgraduate. Her main research interests include blockchain consensus algorithms and applications.

(责任编辑:杨雪敏)