提升区块链多轮 PBFT 验证效率的节点分配方案

贾志鹏 李志淮 孙 晴 王 森
(大连海事大学信息科学技术学院 辽宁 大连 116026)

摘 要 为解决区块链分片技术导致的单个分片失效的问题,多轮 PBFT(Practical Byzantine Fault Tolerance protocol)验证方案被提出,但其节点随机分配过程会导致交易验证有效性降低,通过对这一问题进行分析,同时对目前主要分片项目的节点分配方案进行研究比较,提出一种基于节点评价的节点分配方案。该方案通过对区块链系统内节点进行评分,可对疑似拜占庭节点进行标记,并根据身份标记实现加入和退出分片的操作,实现了分片内拜占庭节点比例的明显降低,提高了验证效率。通过四组对比实验,说明基于节点评价的节点分配方案有更高的验证效率,明显提升了系统吞吐量。

关键词 区块链 分片技术 节点分配 多轮验证 PBFT

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2024.07.036

NODE ALLOCATION SCHEME FOR IMPROVING THE EFFICIENCY OF MULTIPLE ROUNDS OF PBFT VERIFICATION IN BLOCKCHAIN

Jia Zhipeng Li Zhihuai Sun Qing Wang Sen
(*School of Information Science and Technology, Dalian Maritime University, Dalian 116026, Liaoning, China*)

Abstract In order to solve the problem of single shard failure caused by sharding technology in blockchain, a multi-round PBFT verification scheme is proposed. However, the random allocation process of nodes will reduce the effectiveness of transaction verification. By analyzing this problem and comparing the current node allocation schemes of major sharding projects, a node allocation scheme based on node evaluation is proposed. This scheme marked suspected Byzantine nodes by scoring the nodes in the blockchain system, and realized the operation of joining and exiting the shard according to the identity mark, which realized a significant reduction in the proportion of Byzantine nodes in the shard and improved the verification efficiency. Through four sets of comparative experiments, it shows that the node allocation scheme based on node evaluation has higher verification efficiency and significantly improves system throughput.

Keywords Blockchain Sharding technology Node allocation Multiple rounds verification PBFT

0 引 言

2008 年,中本聪在其发表的奠基性文章《比特币:一种点对点的电子现金系统》中首次提出了比特币(Bitcoin)一词[1],区块链作为比特币的底层技术,开始走进人们的视野,并在近几年受到了广泛关注。区块链实质是一个去中心化的账本,简单来说是对分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。区块链凭借其去中心化、不可篡改、不可伪造、公开透明、可验证等特点[2],在多个领域被广泛应用。然而,由于区块链自身的限制,区块链每秒能处理的交易有限,在被大规模应用时,区块链会产生拥堵,处理交易的效率低下。因此,区块链需解决其容量问题,目前已有的解决方案包括侧链[3]、分片技术[4]、DAG 技术[5]、状态通道[6]等。

侧链是相对于主链而言的,它是平行于主链的一条区块链,主链和侧链之间通过双向锚定建立连接,实现主链与侧链之间价值的双向转移。DAG 技术将区块链的链式结构改变为有向无环图结构,以非线性、并

收稿日期:2020-12-28。贾志鹏,硕士生,主研领域:区块链技术。李志淮,教授。孙晴,硕士生。王森,硕士生。

发的方式处理交易,提高扩展性,但 DAG 技术存在诸多挑战,无法达成最终一致性。状态通道是一个建立在区块链外的点对点价值转移通道,交易双方可在区块链外进行交易,然后将交易的最终状态广播到区块链,这一技术解决了高频、小额支付场景中手续费过高的问题,但不适用于所有场景。

分片技术的核心思想是“分而治之”,其最初是传统数据库中的概念,它将大型数据库分成更小、更快、更容易管理的部分。在区块链系统中,分片是指将区块链网络划分成若干能够独立处理事务的较小网络,并将任务分配到不同分片,所有分片可以并行处理不同的事务,以此提升区块链性能[7]。分片技术又分为网络分片、交易分片、状态分片[8]。其中网络分片是指将全网节点划分到不同分片,交易分片是指将全网交易划分到不同分片中进行验证和打包,状态分片是指将完整的状态信息分别存储到各个分片当中。分片技术因其能同时克服性能和扩展性问题,是目前最有效的区块链扩容解决方案。

在区块链中,共识机制是一个重要的问题,它可以使互不信任的节点达成正确的、一致的共识。实用拜占庭共识协议[9](PBFT)具有强一致性的特点,它作为确定性的拜占庭容错算法,被众多项目和文献研究并采用[10],在区块链公链系统中,由于参与共识的节点数量较多,采用 PBFT 算法会造成极大的通信复杂度,所以公链多采用 PoW 和 PoS 共识算法。而分片技术将计算和存储工作分散到对等网络,每个分片中节点数量较少,采用 PBFT 算法的复杂度降低,且由于其强一致性,使得目前多数实现了分片技术的区块链项目采用 PBFT 算法作为分片内共识算法,在这一环境下,需保证分片内拜占庭节点比例低于 1/3 才能正常工作。本文将考虑以 PBFT 作为片内共识算法环境下的分片效率问题。

1　问题描述与分析

1.1　问题描述

分片技术提升了区块链网络的性能,提高了系统吞吐量,同时显著减少通信、计算和存储开销,使区块链得以应用到大型系统中。但与此同时,分片技术为了提高区块链系统的整体性能,降低了单个分片的安全性[11]。

在采用 PBFT 共识协议的区块链系统中,若系统内共有 L 个节点,算法在拜占庭节点数量不超过$(L-1)/3$,即拜占庭节点比例不超过 1/3 的情况下,可同时保证系统安全性和活性。假设在一区块链系统中有 L 个节点,其中有 b 个拜占庭节点,则系统中总体拜占庭节点比例 $f=b/L(0\leqslant f<1/3)$,则系统总体可正常进行事务处理。采用分片技术后,将 L 个节点均匀分配到 k 个分片中,每个分片中的节点数为 $S(S=L/k)$,每个分片的拜占庭节点比例为 f'_i,其中 $i(1\leqslant i\leqslant k)$ 为分片编号。由于在节点分配过程中,拜占庭节点的分配并不均匀,可能会导致部分分片拜占庭节点比例 $f'_i\geqslant 1/3$,使得该分片无法达成共识进行正常工作,从而使得该分片失效,单个分片的安全性遭到破坏。

图 1 中,笑脸代表正常参与工作的节点,鬼脸代表拜占庭节点。可以看出,该系统在分片前的拜占庭节点比例 $f=13/40<1/3$,但在进行分片后,分片一和分片四的拜占庭节点比例超过了 1/3,导致分片失效,同时对系统安全性造成威胁。

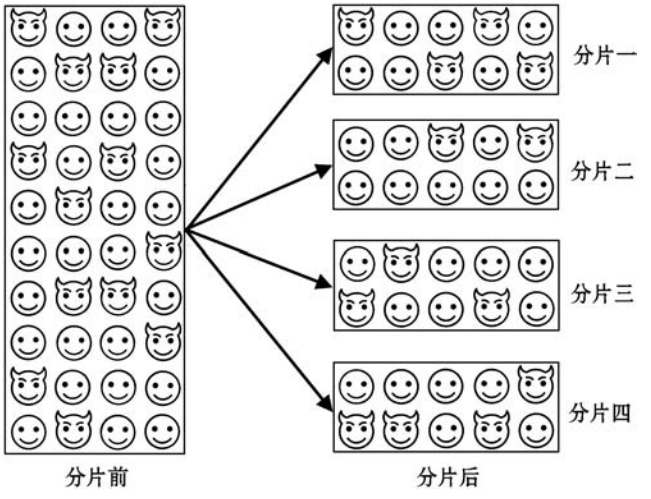

图 1　系统分片后单个分片失效示意图

1.2　单个分片失效的定量研究

假设一个系统中有 L 个节点,每个分片内包含 S 个节点,系统总拜占庭节点比例为 f。则分片的分配可认为是从 L 个节点中选择 S 个节点,令 X 表示选择的 S 个节点中拜占庭节点的数量,则 X 服从超几何分布,记为 $X\sim H(S,f\cdot L,L)$。当 $X\geqslant S/3$ 时,分片中拜占庭节点比例会高于 1/3,导致分片失效,可通过式(1)表示单个分片失效的概率。

$$P\left(X\geqslant\frac{S}{3}\right)=\sum_{x=\frac{S}{3}}^{S}\frac{\binom{f\cdot L}{x}\cdot\binom{(1-f)\cdot L}{S-x}}{\binom{L}{S}}\tag{1}$$

对式(1)进行模拟运算,可讨论在不同系统规模 L、不同分片规模 S 和不同系统拜占庭节点比例 f 的条件下,单个分片失效的概率。如表 1 列出了当系统规模为 600 时,不同分片规模和拜占庭节点比例下单个分片失效的概率。

表 1 分片规模和拜占庭节点比例对单个分片失效概率的影响

S	$f=0.1$	$f=0.2$	$f=0.3$
30	2.91E-4	0.06	0.41
60	9.81E-8	7.37E-3	0.32
90	7.34E-12	8.40E-4	0.26
120	5.22E-17	7.51E-5	0.22
150	6.44E-24	4.73E-6	0.17

由表 1 可见,在拜占庭节点比例 f 一定的情况下,分片规模 S 越大,分片失效的概率越小,且在 $f=0.1$、$S\geqslant 90$ 时,单个分片失效的概率已经小于千万分之一,可认为是极低概率的事件。同时可发现,在系统拜占庭节点比例较高的环境下,单个分片保持着较高的失效概率,这会极大影响系统效率和安全性。

1.3 分片中节点分配方案的研究

在采用分片技术的区块链系统中,每个分片中的节点需要进行动态的调整,保证节点不会长期处在同一分片内,以此满足系统对安全性的保证。如何确保节点分配的公平性是一个重要问题,同时在这一过程中还需保证重分配后的分片内正常节点占大多数,如在采用 PBFT 共识协议的系统中,需保证分片内拜占庭节点比例要低于 1/3。

在目前主流的项目中,分片的节点重分配问题得到了广泛讨论和研究。通过对现有方案的研究,总结目前分片内节点分配算法主要有以下三种:

1) 静态分配模式。在这一模式下,一个分片中的节点是固定不变的,节点的加入和退出由分片中所有节点进行验证,只有获得许可的节点才可加入分片,分片内节点不会进行周期性的更替。这一模式更适用于节点间彼此熟知、相互信任的场景,如在联盟链环境下。而目前分片技术多应用于公链环境下,节点身份不确定,节点间彼此不信任,所以静态分配模式并未得到广泛应用,同时在静态分配模式下由于节点长期处于同一分片中,可能会出现节点间彼此勾结攻击分片的情况,对系统安全性造成隐患[12]。

2) 滚动模式。滚动模式将一个分片视为一个滑动窗口,每进行一次节点的更新,窗口就滑动一次,处在窗口外的节点退出当前分片,同时有新的节点填补窗口,即加入分片。一般情况下,窗口中节点按加入分片的时间先后进行排列,即每次退出的节点为最先加入该分片的节点。滚动模式一般又分为单滚动模式与多滚动模式,单滚动模式要求窗口每次滑动一个单位,即转入/转出一个节点,而多滚动模式要求滑动多个单位,即每次转入转出多个节点。

OmniLedger[13] 则采用了多滚动模式作为其节点重分配的方案,在一个分片中,OmniLedger 将每次转出的节点数量设置为 $k=\log(n/m)$,其中:n 为节点总数;m 为分片数量。通过计算一个种子来设置分片中节点的序列,在等待一段时间后,若节点准备好,它会发送一个声明到要转入分片的分片领导者,后者会将其转入该分片。而 RapidChain[14] 则结合了随机性算法和多滚动思想,每一个想加入分片的节点都需要计算一个工作量证明(Proof of Work,PoW)问题,由参考委员会验证问题的答案,然后使用有限布谷鸟原则(Bounded Cuckoo Rule)把每一个节点随机地分配到每个分片中。同时参考委员会把分片分成两类,分别是活跃成员占大多数的活跃分片和不活跃成员占大多数的消极分片。当新节点加入时,参考委员会把节点随机加入到某个活跃分片中,并把该分片中的固定数量节点随机加入到不同的消极分片中。

滚动模式中的单滚动模式由于每次只更新分片中的一个节点,无法明显快速地降低分片中拜占庭节点比例,在实际应用中仍会导致验证轮数的持续增加[15]。同时在滚动模式中,分片中节点按序排列并进行更替,对节点的选择没有目的性,无法确保更新后的分片能够有效降低拜占庭节点比例。

3) 全更新模式,又称随机数模式。这一模式需要随机数的参与,通过随机数决定每个节点将会被分配到哪个分片中,一般会对分片中所有节点进行重分配,并以此保证节点分配的公平性。在现有的项目中,以太坊和 Elastico 便使用了该种模式。

以太坊作为领先的区块链项目,目前拥有 2 000 亿美元的市值。在以太坊 2.0 中,采用了 RANDAO + VDF 的模式来进行节点的随机分配,首先通过 RANDAO 方式每经过一个 epoch 产生一次随机数,但由于 RANDAO 运行过程中其最后一次操作可被操控,于是引入可验证延迟函数(Verifiable Delay Functions,VDF)来保证最终结果的不可操纵、可验证和不可预测性,由此得来的结果将作为随机数种子,用来为分片选定下一组验证节点。Elastico[16] 作为第一个为公有链设计分片方案的项目,它同样通过产生随机源来对节点进行分配。其随机源生成分为两个阶段,在第一个阶段,分片中所有节点选择一个随机字符串并将其哈希值发送到分片中其他节点,分片中所有节点会对该哈希值集合进行验证并全网广播。在第二个阶段,分片中所有节点将各自的随机字符串全网广播进行验证,系统中的恶意节点将会被剔除,而经过验证的随机字符串将作为随机源用于下一个 epoch。

全更新模式由于其使用随机数作为节点分配的条件,保证了节点分配的随机性和公平性,被现在大多数项目使用,但其存在若干会影响其效率的问题:(1) 随

机数的选择需要初始化设置来选取主节点和对密钥进行收集验证,会耗费一定的时间和资源,且需要额外的节点间通信消耗[17]。(2) 在节点重新分配的过程中,由于所有节点都要参与重分配,此时的系统无法进行交易的验证,会出现系统暂停工作的情况,降低验证率,影响系统效率。(3) 由于完全依靠随机数进行节点重分配,可能会出现重分配后的大多数分片仍然无法达成共识,难以保证重分配后的分片拜占庭节点比例低于 1/3,在一些情况下会增加验证的轮数,并对系统安全造成新的威胁。(4) 由于每次重分配后每个节点都会以很大的概率分配到全新的分片中,此时分片需要一个初始化过程,例如分片内节点需要认证彼此的身份、同步分片的状态等,这使得重分配后的分片无法立即对交易进行验证,这个问题在实现了状态分片的项目中更加明显。节点分配方案对比如表 2 所示。

表 2　节点分配方案对比

指标	静态分配	单滚动	多滚动	随机数
置换数量	不确定	一个节点	多个节点	全部节点
节点选择	系统选择	按序选择	按序选择	随机选择
准入原则	受信任节点准入	任何节点准入	任何节点准入	任何节点准入
安全性	安全	不安全	不安全	不安全
效率	低	低	高	中

1.4　多轮 PBFT 验证方案的提出与分析

针对由分片区块链单个分片安全性降低带来的处理效率问题,文献[18]提出了多轮 PBFT 验证(Multiple Rounds of PBFT Verification,MRPV)方案,且作者在其后续研究中引入了适应多轮方案的节点随机分配算法。MRPV 方案的思想是当某分片中一笔交易在一次验证中未达成共识时,不是直接丢弃该笔交易,而是调用节点随机分配算法,分配一组新的节点对该交易进行新一轮的共识验证,直到该交易被成功验证打包或达到轮数上限后放弃该笔交易。通过对交易进行多次验证,提高了对交易验证的成功率,进而提高区块链效率。

在该方案中,利用 RandHerd 算法作为随机性算法来对节点进行随机分配。RandHerd 基于联合签名和 Schnorr 门限签名,通过初始化设置和随机数生成两个阶段在每一个固定时间间隔产生随机数,满足节点分布的不可预测、可验证、无偏倚等特性,保证了节点无法预测到自己将会被分配到哪个分片。然而全更新模式带来的问题在很大程度上限制了 MRPV 方案的交易验证效率。同时,多轮验证思想对单个分片失效概率的优化效果不明显。

2　基于节点评价的节点分配方案

2.1　方案设计原理与思想

针对第 1 节中的分析,本文应用多滚动模式思想,提出一种适应于多轮 PBFT 验证的基于节点评价的节点分配方案。通过每次更新分片中多个节点,能够对分片内拜占庭节点比例进行快速调整,同时能解决基本多滚动模式“先进先出”原则导致的对轮换节点的选择没有目的性的问题,可以明显降低分片内的拜占庭节点比例,同时对每次轮换节点的数量加以限制,在节点重分配和新节点同步分片信息时分片内原有节点仍可以对交易进行验证,有效提高验证效率。

该方案的主要思想是:为系统中每个节点附加一个分值,该分值会根据节点在分片中的工作表现进行增加或削减。当一个分片在一轮 PBFT 验证中达成共识时,对该分片中所有节点进行分值的增加,若分片未能达成共识,则认为该分片拜占庭节点比例高于三分之一,对该分片中所有节点进行分值的削减。在多个轮次验证后,系统中的拜占庭节点和正常节点的平均分值会出现明显差异,并以此作为根据,可认为分值越低的节点其身份就越倾向于拜占庭节点,基于这一假设,在进行分片更新时,选择分片中分值较低的那一部分节点,认为是疑似拜占庭节点,将其转出当前分片。同时假设系统中存在一等待序列,已加入系统但不在分片中进行验证工作的节点在该序列中按节点分值排序。转出分片的节点将进入等待序列,等待再次进入分片工作,而在进行分片节点转入时,会优先选择等待序列中分值较高的那一部分节点。

此外,为了方便方案设计,设置每隔一定轮次为一个 epoch,在一个 epoch 结束时,主要进行两项工作:节点分值的更新和新节点的加入。为避免节点分值急剧下降,在每个 epoch 结束后,以当前系统中节点的最高分为基准,对系统内所有节点进行分值重设。此外,新节点加入和系统内节点退出也在 epoch 结束后进行,同时设置新加入系统的节点分值为当前系统内节点的平均分。

通过以上设计,实现了目的性地转入和转出分片中的节点,将分片中的疑似拜占庭节点转出分片,同时将等待队列中身份较好的节点转入分片继续工作,可以快速明显地调整分片的拜占庭节点比例,以此提高系统的验证效率。

2.2　方案基本流程

采用基于节点评价的节点分配方案的多轮 PBFT 验证流程如下:

1）系统初始化:分配验证节点到分片中;根据交易发送者的地址将交易分配到特定分片。

2）分片中的节点对分配到该分片的交易进行 PBFT 共识验证。

3）若分片达成共识,交易验证成功,则将交易打包并添加到区块,该分片内所有节点分值增加,并将分片内分值较低的部分节点转出到等待序列,同时将等待序列中分值较高的等量节点转入当前分片。

4）若未达成共识,即该笔交易当前轮次未验证成功,则对当前分片内节点进行分值削减以及分片内节点的更新,同时将该交易验证轮数加一并判断是否达到了轮数上限,若达到上限,则放弃该笔交易,若未达到上限,则将该交易分配到新的分片中进行下一轮的验证。

5）重复步骤 2）- 步骤 4)进行多个轮次验证,并判断是否完成了一个 epoch,若已完成,则进行节点分值的更新和节点的加入和退出。

需注意,在步骤 3)中,分片达成共识同样需要轮换节点,这是因为若分片共识成功且不更新分片会出现如下两个问题:1）分片中的拜占庭节点会因为一次偶然的分配而一直留在正常工作的分片中,其分值无法被削减,身份倾向无法体现;2）随着验证轮数的增加,同一分片中的节点熟悉彼此后,可能会互相勾结攻击分片。

图 2 是对一笔交易进行验证的流程,它表示了一笔交易从开始验证到验证完成或丢弃的生命周期。

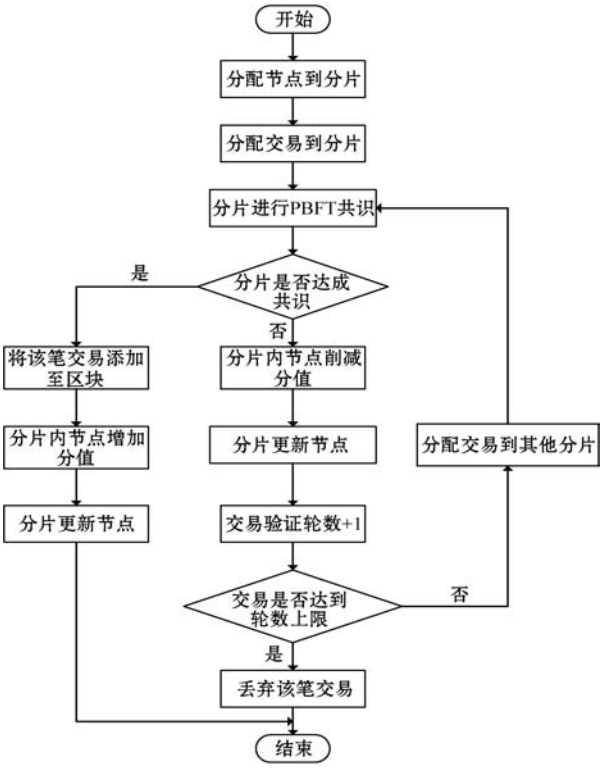

图 2 一笔交易的验证流程

2.3 验证节点分值变化

本方案中,节点分值为设计的核心,节点的分值体现了节点的身份倾向,并且作为分片进行节点轮换时的主要依据。拜占庭节点和正常节点间分值的差异越明显就越能体现其身份。图 3 和图 4 分别为一次实验中第二个 epoch 和第四个 epoch 内拜占庭节点与正常节点的平均分值变化。此处设置分片共识失败后每个节点削减 5 分,共识成功后增加 1 分,每 32 轮验证为一个 epoch。

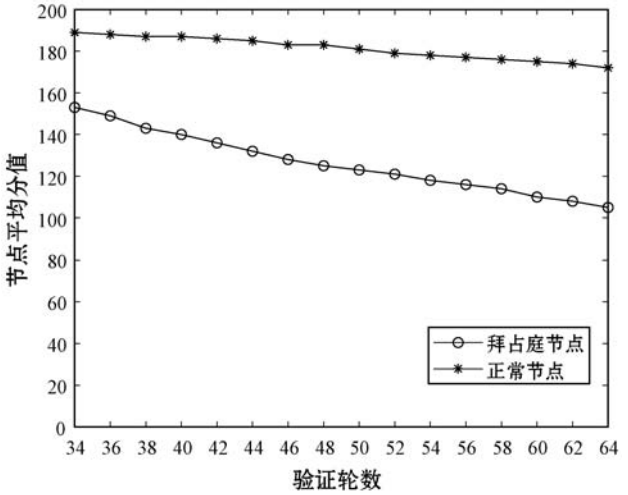

图 3 epoch2 中节点分值变化情况

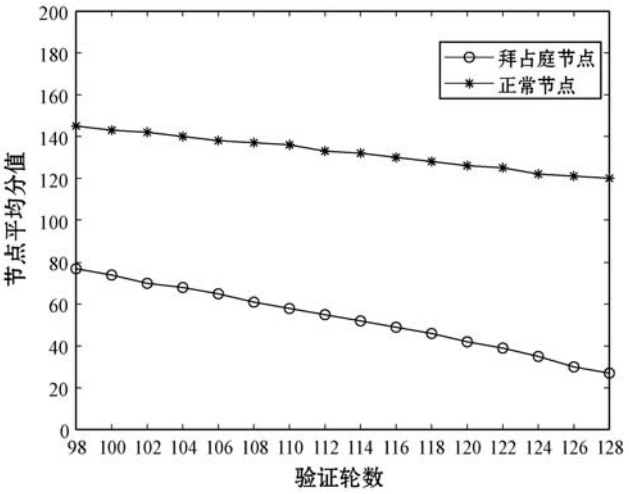

图 4 epoch4 中节点分值变化情况

通过图 3 和图 4 发现,两个 epoch 内正常节点和拜占庭节点的分值均呈下降趋势,但正常节点的平均分值明显高于拜占庭节点,且下降幅度较缓,则可认为分值越低的节点是疑似拜占庭节点的可能性越大。同时,在进行分值削减时,需注意如下问题:一旦某分片未达成共识,则对分片内所有节点进行分值的削减,无论其是拜占庭节点还是正常节点,这就会导致一些正常节点被连带而受到惩罚,但这种影响会随着节点参与验证的轮数增加而减弱,正常节点的分值不会因为少数次的削减而影响其整个验证过程。同时在经过多

个验证轮次后,系统中的拜占庭节点因为其处在未达成共识的分片的可能性更高,分值削减更频繁,则会因为分值极低,而处于等待队列的末端,几乎无法再次进入分片工作,即使偶然加入分片进行验证工作,无论是否影响到分片的正常共识验证,也会因为分片节点的轮换被立即转出到等待队列。

2.4　验证转出部分拜占庭节点比例

在本方案设计中,我们期望每次节点重分配时,分片转出的部分节点中拜占庭节点的比例越高越好。本文的方案通过节点分值辨识出疑似拜占庭节点,并将其置换出当前分片,能尽可能保证转出部分中拜占庭节点占多数。图 5 即为随着验证轮次的增加,未达成共识的分片置换出的部分节点中拜占庭节点占比的变化情况。此处设置每个分片包含 60 个节点,每次置换 15 个节点。

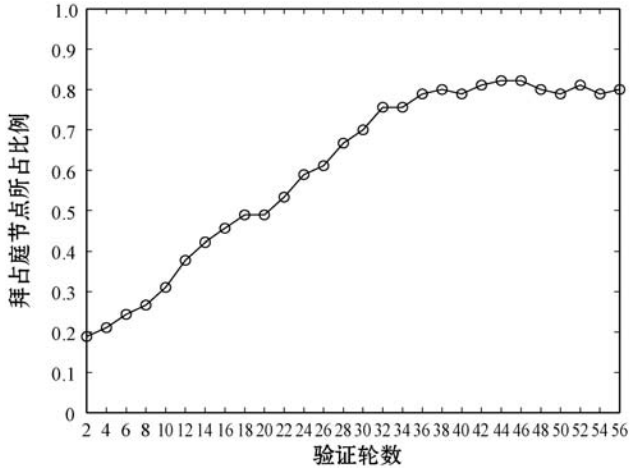

图 5　转出部分拜占庭节点比例变化情况

可以看出,整个趋势呈对数增长,在第一个 epoch 内,随着轮数的增加,转出的节点中拜占庭节点占比呈增长趋势,在第 30 轮后,该占比变化幅度减小,约保持在 80%。这是由于在前期节点间身份差异不明显,但随着验证轮次增加,节点身份有所差异,每次转出的节点能够包含更多的拜占庭节点,同时,在多个轮次后,分片间身份差异体现较为明显,分片每次都可以根据分值置换出拜占庭节点,但也存在部分拜占庭节点分值较高,在当前轮次无法被转出的情况,这就导致了后期该占比趋于平稳但未达到 100%。综上可表明系统中每一次节点置换都可以将较多拜占庭节点转出分片,有效降低分片内拜占庭节点比例。

2.5　交易验证轮数的研究

我们在方案中设置一笔交易被验证失败的次数高于某一限定值后就放弃该笔交易,对验证轮数上限的设置关系到系统整体的性能,若轮数上限设置过低,会导致大量交易被放弃,系统的交易验证率降低,若轮数上限设置过高,又会影响到一笔交易的处理延迟,并对系统的负荷带来压力。下面将详细分析对于验证轮数上限的设置。

由于一笔交易直到被验证成功才被打包到区块上,假设交易在第 X 次验证时被成功验证,则前 $X-1$ 次均验证失败,那么 X 服从几何分布,记为 $X \sim GE(p)$,其中 p 为验证成功的概率,即 $p = 1 -$ 分片失效概率。综上,可通过式(2)表示一笔交易在前 k 轮就能验证成功的概率。

$$P(X \leqslant k) = \sum_{x=1}^{k} p \cdot (1-p)^{x-1} \qquad (2)$$

对式(2)进行模拟运算,计算在不同系统规模 L、不同分片规模 S 和不同系统拜占庭节点比例 f 的条件下,一笔交易在 k 轮内能够验证成功的概率。因在实际环境中,系统拜占庭节点比例无法确定,故主要讨论在不同分片规模下,交易轮数的选择。表 2 列出了系统规模为 600、拜占庭节点比例 $f=0.3$ 时,不同分片规模条件下,交易能在 k 轮内验证成功的概率。

表 3　不同分片规模下 k 轮内验证成功概率

S	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
30	0.59	0.83	0.93	0.97	0.98
40	0.57	0.82	0.92	0.96	0.98
50	0.57	0.81	0.92	0.96	0.98
70	0.66	0.88	0.96	0.98	0.99
90	0.73	0.93	0.98	0.99	0.99

由表 3 可见,随着分片规模的加大,一笔交易在 k 轮内被验证成功的概率越大。同时在不同分片规模下,大多数交易均能在 5 轮内被验证成功,在分片规模为 $S=90$ 时,3 轮内对一笔交易验证成功的概率就达到了 98%,表明系统能以较低的处理延迟确保更高的验证成功率。

3　实　验

3.1　实验的基本设置

为验证本文提出的基于节点评价的节点分配方案的有效性,将本文方案与原 MRPV 方案和 Elastico 方案进行对比实验,观察三种方案在交易验证率、平均验证轮数、单个分片失效率、系统吞吐量四个方面的性能。实验设置总结点数为 1 500,每 60 个节点为一个分片,每次置换节点数量为 15 个,节点初识分值为 200 分,共识失败扣 5 分,共识成功加 1 分。系统内节点可以动态加入和退出,每个 epoch 结束后系统新加

入节点 20 个,节点身份随机确定,分值设置为当前 epoch 系统内节点平均分,同时随机转出 20 个节点。

在进行实验时,需要设置以下假设条件:

(1) 交易随机、平均分配到每个分片进行验证。

(2) 不考虑由于节点性能导致的分片负载不均衡的问题。

(3) 认为系统内所有节点的拜占庭比例小于 1/3,整个系统可正常运行。

(4) 认为网络中节点间通信良好,延迟在可控范围内。若由于节点延迟导致分片共识失败,同样按分片拜占庭节点比例大于 1/3 处理。

3.2 实验设计与结果分析

3.2.1 交易验证率对比实验

交易的验证率表示一批交易中被成功进行验证的交易所占的比率。在 Elastico 方案中,一笔交易若未被成功验证则直接丢弃,而在多轮验证系统中,一笔交易可能需要进行多次验证才会成功。实验时向系统注入 25 000 笔交易,实验观察随着验证轮次的增加,该批交易中已验证交易占交易总数的比例。本实验设置若一笔交易经过 10 轮验证仍未成功则放弃该笔交易。进行多次实验后得到三种方案验证率结果如图 6 所示。

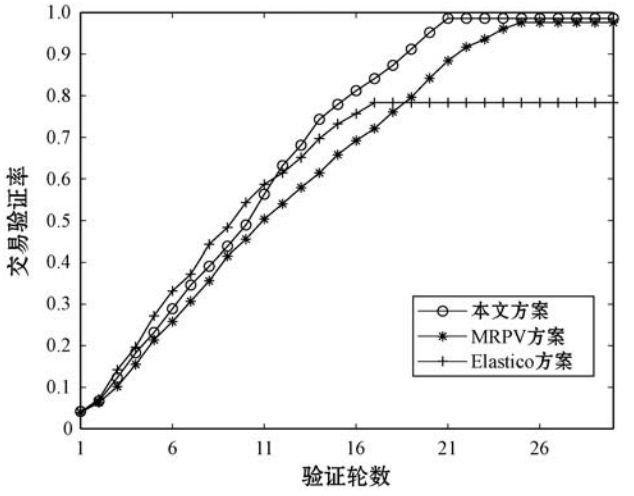

图 6 交易验证率对比

可以看出,在最终的交易验证成功率上,三种方案均未实现交易的 100% 验证,但两种采用了多轮验证思想的方案的验证率超过了 95%,而 Elastico 方案的验证成功率只有 78%,超过 1/5 的交易因一次验证未通过而被直接丢弃。在验证的速度方面,Elastico 方案最先验证完所有交易,而本文方案和 MRPV 方案对所有交易的验证较慢,这是由于后两者需要对前几轮未被成功验证的交易进行重新验证,可视为需验证的交易增多。在验证过程中,Elastico 在前几轮验证速度较快,但同时也丢弃了不少交易。在经过前几轮的验证后,本文方案下系统内拜占庭节点由于分值较低而有更少的机会参与分片的验证工作,分片的拜占庭节点比例降低,验证效率明显提高,交易能在当前轮次被成功验证,无须滞留在系统中造成负担。

通过对以上的分析可以看出,本文提出的节点分配方案通过分析节点行为标识节点身份进而有目的性地进行节点分配,让更多的正常节点参与验证,可以优化分片内拜占庭节点比例,从而提高分片验证交易的效率,同时可以提高交易验证的成功率。

3.2.2 平均验证轮数对比实验

验证轮数表示交易从第一次被验证到最终验证成功所经过的轮数,平均验证轮数体现的是当前系统进行交易验证的效率,平均验证轮数越低,表示该系统处理交易的速率越快,系统越稳定。系统验证交易的效率受到系统中拜占庭节点比例的影响,高拜占庭节点比例会导致分片有效性和安全性降低,从而使得系统验证速率下降。本实验通过设置不同的系统内拜占庭节点比例,观察原 MRPV 方案和本文方案随着拜占庭节点比例的增加,其平均验证轮数的变化情况。实验时向系统注入 25 000 笔交易,拜占庭节点比例 f 分别设置为 0.1、0.15、0.2、0.25、0.3,进行多次实验,分别统计每次实验中所有交易的平均验证轮数,实验结果如图 7 所示。

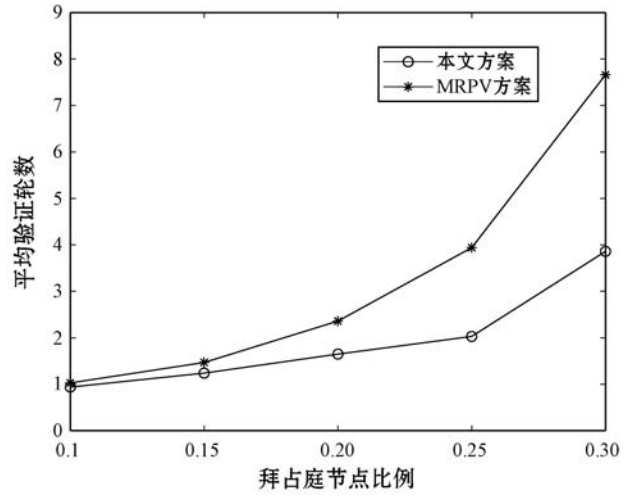

图 7 平均验证轮数对比

可以看出,MRPV 方案和本文方案的平均验证轮数都随着系统内拜占庭节点比例的增加而呈上升趋势,且这一上升趋势在拜占庭节点比例 $f>0.25$ 时更加明显。此外,两方案在不同的拜占庭节点比例下的平均验证轮数存在差异,在拜占庭节点比例 $f\leqslant 0.15$ 时,两方案的平均验证轮数差距并不明显,均在 1 轮左右。而当 $f\geqslant 0.25$ 时,本文方案的平均验证轮数能达到 MRPV 方案的一半,明显提升了交易验证速度,降低了延迟。综上可知,在拜占庭节点比例较低的情况下,两方案均能有效完成对交易的验证,而在高拜占庭节

点比例的环境下,本文方案表现出了明显的优势,平均验证轮数可降低为原方案的一半,这是由于本文方案可将拜占庭节点进行标记并降低其加入分片进行验证的概率,能够很好地平衡所有分片的拜占庭节点比例,从而使得交易能更快被验证,有效降低系统内交易的平均验证轮数。

3.2.3　单个分片失效率对比实验

在应用了分片技术的区块链系统中,一个分片是否能够对交易进行成功验证,极大地影响着系统的性能。一般情况下,当一个分片失效,我们认为该分片中拜占庭节点比例超过 1/3,导致分片内节点未能达成一致。一个分片的失效概率受到分片规模和区块链系统的整体拜占庭节点比例的影响。实验在系统总体拜占庭节点比例为 0.1、0.15、0.2、0.25、0.3 时,对本文方案、MRPV 方案和 Elastico 方案的单个分片失效率进行对比,计算在 2 000 次验证中,分片失效的次数占总次数的比例。实验结果如图 8 所示。

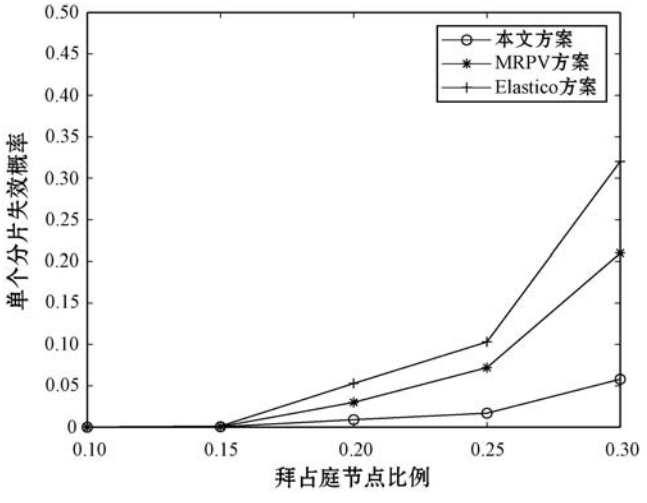

图 8　单个分片失效概率对比

可以看出,随着系统总拜占庭节点比例增加,单个分片失效率也增加,在总体拜占庭节点比例较小时,三种方案的分片失效率均接近于零,当总体拜占庭节点比例较大时,三种方案表现出明显差异。Elastico 方案的失效率增长幅度最为明显,在拜占庭节点比例 $f=0.3$ 时,其分片失效率超过了 0.3。MRPV 方案通过采用 RandHerd 作为其随机数生成方案,在一定程度上减少了拜占庭节点聚集在单一分片上的现象,但并未有效降低分片拜占庭节点比例。本文方案对分片内节点进行有目的的调整,减少分片中疑似拜占庭节点的数量。在系统趋于稳定时,能够及时将系统内拜占庭节点停止工作,能有效保证分片拜占庭节点比例低于 1/3,使得本方案分片失效率保持在较低水平。

3.2.4　系统吞吐量对比实验

在区块链技术日益发展的今天,区块链系统规模日益扩大,需处理的交易越来越多,能够在一定时间尽可能多地处理交易成为评价一个区块链系统的重要指标,系统吞吐量一般用系统每秒处理的交易数 TPS(Transaction Per Second)来表示,它体现的是系统处理交易的能力,TPS 越高,表示系统每秒能处理的交易越多,系统能够实现更大的规模。同时,区块链系统的吞吐量受到系统内拜占庭节点比例的影响。本节通过进行多次实验,观察在不同的系统拜占庭节点比例的条件下,原 MRPV 方案和本文方案平均 TPS 的变化情况,实验结果如图 9 所示。

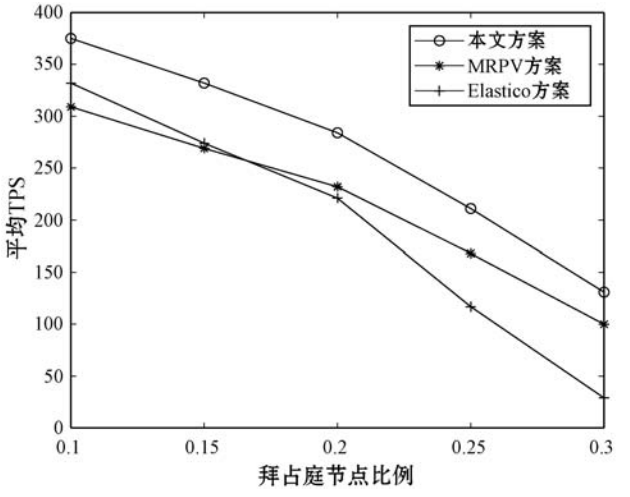

图 9　系统吞吐量对比

可以看出,拜占庭节点比例和系统平均 TPS 成反比,且在拜占庭节点比例越高时,平均 TPS 下降效果更加明显。通过对比可以看出,本文方案在平均 TPS 上要优于其余两方案,表明本文方案能够在每秒内处理更多交易,且在任意拜占庭节点比例下,这种优势都比较明显。造成这种优势的原因有二:(1) 本文方案能够使拜占庭节点参与交易验证的概率降低,使系统更稳定,提高了交易处理的效率。(2) 由于在进行节点重分配时,MRPV 方案对所有节点进行重新分配,在新的分片进行交易验证前,需进行分片的初始化,如节点间彼此确认信息,进行系统同步等工作,需要消耗一定时间。而本文方案在进行节点重分配时,通过选择合适的节点置换数量,能够实现分片内保留的节点仍可进行交易的验证,且在新加入节点进行同步时继续工作,使该分片持续性工作,减少了暂停系统的时间,使系统能在一定时间内处理更多的交易。综合以上两点,本文方案体现出了更高的平均 TPS,能够满足更大规模的系统实现。

4　结　语

本文针对分片技术带来的单个分片失效问题和多

轮验证思想中节点重分配方案存在的节点大规模置换带来的时间消耗、分片安全性无法保证等问题,提出一种基于节点评价的节点分配方案,该方案通过对区块链系统内节点进行评分,通过其工作行为进行分值的增加和削减,能够通过分值标定疑似拜占庭节点,从而有目的性地进行节点的置换,使系统内各分片的拜占庭节点比例有效降低,提高了系统的验证效率和安全性,同时该方案支持节点的自由加入和退出,保证了系统活性。本文最后通过实验进行验证,说明了基于节点评价的分配方案能够实现更高的验证成功率,有效降低交易验证的轮数,明显降低单个分片失效概率同时提高系统的吞吐量,较原 MRPV 方案有明显提升,在处理更大规模的交易请求时有明显优势。

但本文方案具有一定的局限性。目前大多数区块链分片项目基本实现了网络分片和交易分片,而对状态分片的实现还在探索中。本文方案在实现状态分片的环境下,需要考虑节点的性能是否能对所有分片的状态进行同步,一般节点只能同步若干个分片的状态,如何确保节点在分配到新的分片后能成功同步状态为目前的主要问题,也为下一步研究的重点。

参 考 文 献

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system [EB/OL]. (2008 - 08 - 22). http://bitcoin. org/bitcoin. pdf.

[2] Zakhary V, Amiri M, Maiyya S, et al. Towards global asset management in blockchain systems[EB]. arXiv:1905. 09359, 2019.

[3] Poon J, Buterin V. Plasma: Scalable autonomous smart contracts[EB/OL]. (2017 - 08 - 11)[2020 - 12 - 28]. http://plasma. io/plasma. pdf.

[4] Yu G, Wang X, Yu K, et al. Survey: Sharding in Blockchains[J]. IEEE Access, 2020, 8: 14155 - 14181.

[5] Wang Q. Improving the scalability of blockchain through DAG[C]//20th International Middleware Conference Doctoral Symposium, 2019: 34 - 35.

[6] Raiden foundation. Raiden network whitepaper[EB/OL]. (2018 - 05 - 11)[2020 - 12 - 28]. http://raiden. network.

[7] 潘晨,刘志强,刘振,等. 区块链可扩展性研究:问题与方法[J]. 计算机研究与发展,2018,55(10):2099 - 2110.

[8] Wang G, Shi Z, Nixon M, et al. SoK: Sharding on Blockchain[C]//1st ACM Conference on Advances in Financial Technologies, 2019: 41 - 61.

[9] Castro M, Liskov B. Practical byzantine fault tolerance [C]//3rd Symposium on Operating Systems Design and Implementation, 1999: 173 - 186.

[10] Sukhwani H, Martinez J, Chang X, et al. Performance modeling of PBFT consensus process for permissioned blockchain network (Hyperledger Fabric)[C]//2017 IEEE 36th Symposium on Reliable Distributed Systems, 2017: 253 - 255.

[11] Dang H, Dinh A, Loghin D, et al. Towards scaling blockchain systems via sharding[EB]. arXiv: 1804. 00399, 2018.

[12] Androulaki E, Barger A, Bortnikov V, et al. Hyperledger fabric: A distributed operating system for permissioned blockchains[EB]. arXiv: 1801. 10228, 2018.

[13] Kokoris-Kogias E, Jovanovic P, Gasser L, et al. OmniLedger: A secure, scale-out, decentralized ledger via sharding [C]//2018 IEEE Symposium on Security and Privacy, 2018: 583 - 598.

[14] Zamani M, Movahedi M, Raykova M. RapidChain: Scaling blockchain via full sharding[C]//2018 ACM SIGSAC Conference on Computer and Communications Security, 2018: 931 - 948.

[15] Kokoris-Kogias E, Jovanovic P, Gailly N, et al. Enhancing bitcoin security and performance with strong consistency via collective signing[C]//25th USENIX Conference on Security Symposium, 2016: 279 - 296.

[16] Luu L, Narayanan V, Zheng C, et al. A secure sharding protocol for open blockchains[C]//2016 ACM SIGSAC Conference on Computer and Communications Security, 2016: 17 - 30.

[17] Syta E, Jovanovic P, Kokoris-Kogias E, et al. Scalable bias-resistant distributed randomness[C]//2017 IEEE Symposium on Security and Privacy, 2017: 444 - 460.

[18] 王夫森,李志淮,田娜. 提升分片规模和有效性的多轮 PBFT 验证方案[J]. 计算机工程与应用,2020,56(24): 102 - 108.

(上接第 170 页)

[27] Ke Q H, Bennamoun M, An S J, et al. A new representation of skeleton sequences for 3D action recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition, 2017: 4570 - 4579.

[28] Li S, Li W Q, Cook C, et al. Independently recurrent neural network (IndRNN): Building a longer and deeper RNN [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2018: 5457 - 5466.

[29] Vemulapalli R, Arrate F, Chellappa R. Human action recognition by representing 3D skeletons as points in a lie group [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2014: 588 - 595.

[30] Huang Z W, Wan C D, Probst T, et al. Deep learning on lie groups for skeleton-based action recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition, 2017: 1243 - 1252.