

基于信誉权重与分组的 PBFT 共识算法优化

卢文凯¹, 唐郑熠^{1,2}, 连至助², 曹宝连¹

(1. 福建理工大学 计算机科学与数学学院, 福建 福州 350118;

2. 闽江学院 福建省信息处理与智能控制重点实验室, 福建 福州 350118)

摘要: 实用拜占庭容错(PBFT)算法在联盟链中应用广泛,但其消耗的时间会随着节点数量的增加而非线性增长,极大限制了联盟链共识过程的覆盖面。针对该问题,提出一种改进算法 GWPBFT(group-weighted PBFT),采用分组策略将节点划分为若干分组,通过并行的组内局部共识提高共识效率,从而支持大规模的节点共识;通过信誉权重机制,使得信誉较高的节点在共识过程中拥有更大的决策权重,增强了系统在不可靠环境下的安全性。实验结果表明,在节点数为175时,GWPBFT相比PBFT算法降低了92.85%的延迟、92.56%通信开销。在拜占庭节点比例达到80%的极端环境下,GWPBFT仍能够确保共识过程的高可靠性和稳定性。

关键词: 区块链;实用拜占庭容错;分组策略;信誉权重机制

中图分类号: TP399

文献标志码: A

文章编号: 2097-3853(2025)06-0564-07

PBFT consensus algorithm optimization based on reputation weight and grouping

LU Wenkai¹, TANG Zhengyi^{1,2}, LIAN Zhizhu², CAO Baolian¹

(1.School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou 350118, China;

2. Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou 350118, China)

Abstract: Practical Byzantine fault tolerance (PBFT) algorithm is widely applied in consortium blockchains. However, its time consumption grows non-linearly with the increase in the number of nodes, which greatly limits the coverage of the consensus process in consortium blockchains. To address this issue, an improved algorithm called GWPBFT (group-weighted PBFT) is proposed. This algorithm adopts a grouping strategy, dividing the nodes into several groups. Through parallel local consensus within each group, the consensus efficiency is improved, thus enabling support for large-scale node consensus. The GWPBFT algorithm also introduces a reputation weight mechanism, where nodes with higher reputation have greater decision-making weight in the consensus process, thereby enhancing the system's security in unreliable environments. Experimental results show that, with 175 nodes, GWPBFT reduces latency by 92.85% and communication overhead by 92.56% compared with the PBFT algorithm. Even in extreme environments where Byzantine nodes account for 80% of the total, GWPBFT still ensures high reliability and stability of the consensus process.

Keywords: blockchain; practical Byzantine fault tolerance; grouping strategy; reputation weight mechanism

区块链^[1]由于其去中心化、不可篡改、透明、安全和匿名等特性^[2]在金融服务、能源交易、供应链管理、物联网等领域显示出巨大潜力^[3]。区块链本质上是一种分布式数据库,其核心目标是

确保系统中所有节点的一致性^[4]。因此,共识机制成为系统有效运行的关键要素^[5]。目前,公有链的主流共识算法有工作量证明(PoW)^[1]和权益证明(PoS)^[6]两种算法。PoW算法存在资源浪

收稿日期:2024-12-05

基金项目:福建省自然科学基金资助项目(2022J01933);闽江学院福建省信息处理与智能控制重点实验室开放课题(MJUKF-IPIC202403)

第一作者简介:卢文凯(1999—),男,福建南平人,硕士研究生,研究方向:区块链技术。

费问题^[7],而PoS算法^[8]却可能导致权利集中化^[9]。私有链中常用的RAFT算法^[10]虽然效率较高,但无法防止恶意节点。在联盟链应用中,PBFT(practical Byzantine fault tolerance)算法因为无需依赖代币,同时具备拜占庭容错能力,被视为理想的共识算法之一。然而,PBFT通信开销较高^[11]、面对大规模节点网络时表现出适应性不足^[12]、不可靠环境下的安全性不足^[13]等问题。

许多研究引入分组机制以降低系统开销、提升系统的扩展性。例如,文献[14]利用C4.5决策树算法对节点根据其历史行为进行信用评估并分组,但在处理大规模网络时可能面临训练效率不足和复杂度较高的问题。文献[15]基于特征信用将节点划分为多个组,仅由代理节点与主节点通信;但是模型对代理节点的依赖性较强,系统安全性与稳定性存在隐患。文献[16]提出了评分和分组策略,将节点分为共识节点和候选节点,只有共识节点才参与共识流程,虽然显著提升了共识效率,但可能使部分节点长期处于候选状态,降低其参与积极性。此外,还有一些研究通过优化节点选择机制筛选高质量节点参与共识,以提高系统的安全性和效率。例如,文献[17]通过对节点性能的全面评估结合随机选举策略,优先筛选高性能节点参与共识。文献[18]基于拉格朗日插值生成的随机种子优化主节点集合的选举,并结合委托节点机制,减少了参与共识的节点数量。文献[19]通过信誉评分机制和监督节点机制减少了实际参与共识的节点数量,但长期选取高质量节点可能导致其他节点被排除在共识之外,从而削弱系统的公平性和整体节点的参与积极性。

综上,本课题基于PBFT提出GWPBFT(group-weighted PBFT)算法,引入决策权重的共识机制,通过信誉分配决策权重,增强系统的安全性和容错能力;采用分组策略在保证全网节点参与共识的同时减少通信开销,提高系统的效率和扩展性。

1 PBFT一致性协议

PBFT算法旨在解决分布式系统中的一致性问题,特别是针对拜占庭将军问题(Byzantine generals problem)。通过其独特的容错机制,系统即使存在部分节点行为异常,整体依然能够维持一致性和可靠

性。该算法的一致性协议通常分为请求、预准备、准备、确认和回复5个阶段,如图1所示。

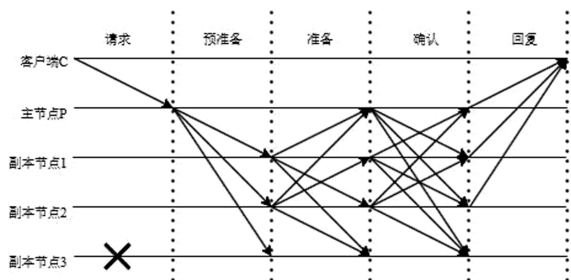


图1 PBFT一致性协议

Fig.1 PBFT consistency protocol

2 GWPBFT算法

GWPBFT算法在初始阶段配置了 n 个参与共识的节点,并为每个节点设定初始信誉分,根据预定的权重分配策略确定每个节点的个人决策权重,并据此进行节点分组。决策权重较高的节点被分配至优先级较高的组别,以增强其在共识过程中的影响力。完成分组后,系统进入共识阶段,执行GWPBFT一致性协议。随后,系统会更新各节点的信誉分,同时记录其活跃度和贡献度数据。算法检查是否达到预定的共识轮次 T ;若未达到,则继续执行共识协议;若已达到,则进一步检查是否到达系统的运行终止时间。若未达到终止时间,则重新分配个人权重并更新分组,继续下一轮共识过程;若达到终止时间,则流程结束。

2.1 信誉机制

节点的信誉分是基于其活跃度、贡献度和任务达成度等指标来评估的。

节点的活跃度越高,所获得的信誉分相应也越高。活跃度计算如式(1)所示。

$$A(x) = \sum_{t_x \in T} e^{-\lambda(t-t_x)} \quad (1)$$

其中, $A(x)$ 为节点 x 的活跃度, t_x 为节点 x 参与共识的时间点, T 为时间范围, t 是当前时间, λ 是衰减系数,用于调整时间的影响强度。

为了统一和标准化度量标准,对活跃度 $A(x)$ 进行归一化的处理,如式(2)所示。

$$A_{\text{norm}}(x) = \frac{A(x) - \min(A)}{\max(A) - \min(A)} \quad (2)$$

其中, $A_{\text{norm}}(x)$ 为节点 x 归一化后的活跃度,范围在 $[0, 1]$, $\max(A)$ 为所有共识节点中最大的活跃度, $A(x)$ 代表节点 x 归一化前的活跃度, $\min(A)$

代表所有共识节点中最小的活跃度。

节点的贡献度越高,所获得的信誉分也相应越高,其计算如式(3)所示。

$$C(x) = \frac{\sum_{j=1}^{N(x)} w_j \times c_{x,j}}{W_x} \quad (3)$$

其中, $C(x)$ 为节点 x 的贡献度, $N(x)$ 是节点 x 完成的任务总数; w_j 是第 j 个任务的权重,反映任务的重要性或难度; $c_{x,j}$ 为 1 时,表示节点 x 在第 j 个任务完成,否则为 0; W_x 是参与任务权重的总和。

任务达成度指的是节点是否按照预期准确地达成共识。若节点按预期成功完成共识,则应获得 ΔS_{reward} 信誉分,否则扣除 ΔS_{punish} 。

节点的信誉分会受节点行为的影响,其计算如式(4)所示。

$$R_{\text{new}}(x) = \begin{cases} R_{\text{old}}(x) + \Delta S_{\text{reward}} + \alpha \times A_{\text{norm}}(x) + \beta \times C(x), & \text{if 共识成功} \\ R_{\text{old}}(x) - \Delta S_{\text{punish}}, & \text{else} \end{cases} \quad (4)$$

其中, $R_{\text{old}}(x)$ 和 $R_{\text{new}}(x)$ 分别为节点 x 更新前后的信誉分, α 为用于调节节点活跃度影响程度的系数, β 为用于调节节点贡献度影响程度的系数。

2.2 个人决策权重分配

为了合理分配决策权重,采用斐波那契数列确定每个小组的成员数量,数列的不同数字(如 1,1,2,3,5,8,13 等)对应不同的成员数量。在确定每个小组的成员数量后,依据节点的信誉评分从高到低依次将其分配到各个小组中。评分较高的节点将被分配到成员数量较少的小组中,确保了更具信誉的小组在决策过程中发挥更大的影响力,同时优化了权重分配的合理性。该分组方法仅用于决策权重的分配,旨在提高决策过程的可靠性,而与共识协议中的分组策略无直接关联。

为了进一步增强信誉分较高的节点的决定权,算法采用衰减函数来调控斐波那契数列中每个小组的权重,确保分组数量较少的小组获得更大的权重分配,提升其在决策过程中的影响力。衰减函数如式(5)所示。

$$d_i = d_{\text{max}} - k \times \ln(2i + 1) \quad (5)$$

其中, d_i 为第 i 个小组的决策权重, d_{max} 表示最大的小组决策权重, i 为小组编号, k 为权重衰减系数。

在确定了小组的决策权重后,需要计算每个成员的个人决策权重,如式(6)所示。

$$p_{ij} = \frac{d_i}{s_i} \quad (6)$$

其中, p_{ij} 表示第 i 小组中第 j 个成员的个人决策权重, d_i 为第 i 个小组的决策权重, s_i 为第 i 小组的成员人数。

2.3 共识协议分组策略

为了支持 PBFT 共识算法在各个小组内的高效执行,共识节点根据个人决策权重分配到不同的小组,权重较大的节点会进入前排小组,从而确保这些小组具备更强的决策能力。

分组的总数为节点总数的平方根,如式(7)所示。

$$m = \lfloor \sqrt{n} \rfloor \quad (7)$$

其中, m 为分组总数, n 为节点总数。

确定每组的基本节点数,如式(8)所示。

$$G = \frac{n}{m} \quad (8)$$

其中, G 表示每组的基本节点数。

如果分组后存在剩余的节点,将其平均分配给前 m 个小组,共识协议分组策略确保所有节点被平等地考虑在内。

2.4 一致性协议

改进后的一致性协议和 PBFT 的一致性协议步骤一致,但通过分组策略在组内进行预准备、准备、确认,可以减少大量的通信开销,如图 2 所示。

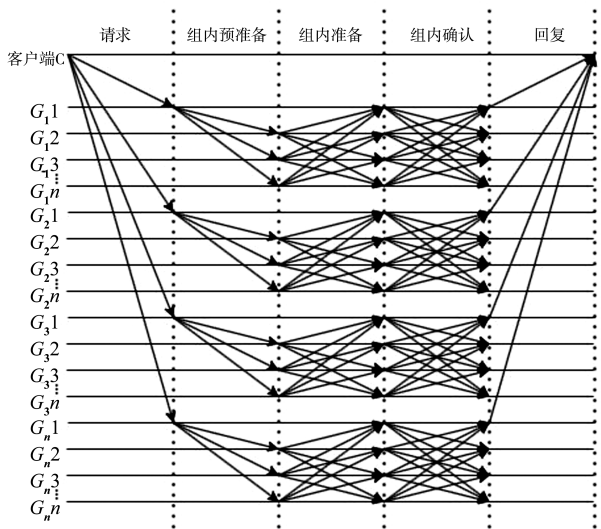


图 2 GWPBFT 一致性协议

Fig.2 GWPBFT consistency protocol

(1)在请求阶段,客户端向各组的主节点发送提案的请求。

(2)在组内预准备阶段,各组主节点验证客户端提案有效性后,生成并广播预准备消息至组内副本节点。副本节点验证消息签名和提案后,若无误则进入组内准备阶段。

(3)在组内准备阶段,组内的副本节点对提案生成准备消息,并将其广播至其他组内节点。当组内节点至少收到 $2f$ 条正确的准备消息时(其中 f 表示组内最大容忍的拜占庭节点数),则进入组内确认阶段。

(4)在组内确认阶段,节点生成确认消息,并将其广播出去。当组内主节点收到至少 $2f$ 条正确的确认消息时,则进入回复阶段。

(5)在回复阶段,各组主节点向客户端发送回复消息。客户端收集多组回复后,当累计决策权重总和达到或超过全局总决策权重的 $2/3$ 时,即视为提案共识完成。

2.5 GWPBFT 算法时间复杂度分析

GWPBFT 算法在执行共识协议分组策略时,使用快速排序(时间复杂度为 $O(n \log n)$)对节点按信誉分进行排序,之后将节点分配到各个小组的时间复杂度为 $O(n)$ 。因此,节点分组的总时间复杂度为 $O(n \log n)$ 。

在每轮共识中,每个小组进行决策权重的分配,分配基于斐波那契数列。斐波那契数列使用矩阵快速幂方法,其计算时间复杂度为 $O(\log n)$ 。每个小组进行投票的复杂度为 $O(G^2)$,而有 m 个小组,总的共识阶段时间复杂度为 $O(n^{3/2})$,计算过程如式(9)所示。

$$O(m \times G^2) + O(\log n) = O\left(\sqrt{n} \times \left(\frac{n}{\sqrt{n}}\right)^2\right) + O(\log n) \approx O(n^{3/2}) \quad (9)$$

在信誉分更新阶段,每轮共识结束后,需要根据节点的行为更新其信誉分。该阶段涉及对所有节点的活跃度、贡献度和任务达成度进行计算。每个节点的信誉更新涉及对活跃度、贡献度等多项指标的计算,每个节点的时间复杂度为 $O(1)$ 。对于 n 个节点,总的复杂度为 $O(n)$ 。

GWPBFT 算法的时间复杂度为 $O(n^{3/2})$,计算过程如式(10)所示。随着节点数量 n 的增加,GWPBFT 算法的时间复杂度呈现次线性增长,较 PBFT(时间复杂度为 $O(n^2)$)在大规模网络中有显著的性能提升。

$$O(n \log n) + O(n^{3/2}) + O(n) \approx O(n^{3/2}) \quad (10)$$

2.6 收敛性分析

信誉分的变化由节点的历史行为与当前行为共同决定。其中,历史行为指标反映了信誉分的长期稳定性;而当前行为的波动则可能对信誉分造成短期影响,并通过影响历史行为的累积进而作用于未来信誉分。因此,信誉分需要经过一定数量的共识后才趋于稳定。假设每个节点的初始信誉分相同,随着多轮共识的执行和信誉分的持续更新,各节点的信誉分将反映各自的长期行为特征。理论上,随着共识轮次的增加,节点历史行为的累积效应逐渐增强,当前行为的影响逐步减弱,从而有效降低信誉分的波动性,最终使节点的信誉分趋于稳定。

GWPBFT 算法中的决策权重是基于节点的信誉分动态调整的。由于信誉分的动态调整存在一定的延迟效应(即节点的行为需要经过一段时间后才能影响其决策权重),初期决策权重波动较大,但随时间推移,节点行为趋稳,决策权重逐渐平稳。GWPBFT 的收敛性依赖于节点信誉分的稳定过程,基于实际行为的更新机制使系统呈渐进式而非瞬时收敛。经多轮共识后,节点行为模式稳定,决策权重波动减小,系统最终达到稳定共识状态。

3 实验

本研究基于 Python 语言搭建并模拟了一个小型区块链系统,从延迟、通信开销和共识成功率等关键性能指标方面,将 GWPBFT 与 SGPBFT^[16]、G-PBFT^[20]和经典 PBFT 进行对比实验,对 GWPBFT 算法的性能进行验证和评估。实验环境为 Windows Server 2019 操作系统,采用 Intel(R) Xeon(R) Gold 6230 CPU @ 2.10 GHz 处理器,配备 320 GB 内存。

3.1 实验参数设置与分析

活跃度衰减系数 λ 影响节点活跃度的衰减速率,较大的衰减系数能够确保近期活跃度较高的节点能够得到更多的信誉分,从而增强其在后续共识中的影响力。本实验设将 λ 设为 10,该设置对性能的影响较小,以确保系统能够及时反映近期节点的活跃状态。

为了确保节点在共识过程中的诚实性,设计了一个以惩罚为主、奖励为辅的信誉机制,任务达成度惩罚分 ΔS_{punish} 设置为 5,任务达成度奖励分

ΔS_{reward} 设置为 1。通过较强的惩罚措施动态降低恶意节点的决策权重,确保节点未按预期完成任务时会承担较大的代价,从而提升系统的安全性。在信誉评分的计算中,活跃度系数 α 和贡献度系数 β 均设置为 0.5,确保节点的历史行为(活跃度和贡献度)与当前行为(共识达成度)在评分中同等重要。

3.2 共识延迟

与传统的全网广播共识模式相比,GWPBFT 的动态分组共识策略通过局部化通信减少了全网节点间的消息传递,避免了延迟的指数级增长。特别是在节点数量大规模扩展时,分组策略能够限制通信范围,保证延迟的平稳增长。

本研究分别在包含 75、100、125、150 和 175 个节点的实验环境对 GWPBFT、SGPBFT、G-PBFT 和 PBFT 4 种共识算法的延迟进行测试评估。每轮测试中,客户端发送 10 条请求消息,重复测试 10 次,最终将 10 次测试的平均值作为评估结果。实验结果如图 3 所示。

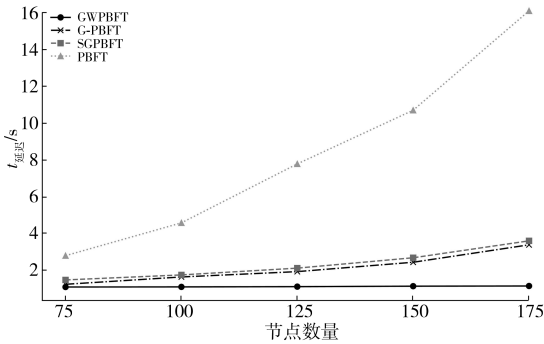


图 3 各算法延迟对比

Fig.3 Comparison of latency for different algorithms

从图 3 可见,在相同的节点数量下,GWPBFT 在共识延迟方面的表现显著优于 PBFT、SGPBFT 和 G-PBFT。随着节点数量的增加,PBFT 的延迟呈现急剧增长趋势,SGPBFT 和 G-PBFT 则表现为中等速率的增长,而 GWPBFT 的延迟增长则相对平稳。当节点数量达到 175 时,PBFT、SGPBFT、G-PBFT 和 GWPBFT 的延迟分别为 16.108、3.606、3.389 和 1.151 s。相较于 PBFT、SGPBFT 和 G-PBFT,GWPBFT 的延迟分别降低了 92.85%、68.08% 和 66.04%。

3.3 通信开销

通信开销指的是在共识过程中,网络中各节点为达成共识而产生的总通信量。GWPBFT 的动态分组共识策略显著减少了每个小组内的通信

量,有效降低了共识过程中的通信开销。该策略还避免了全网广播所导致的通信瓶颈,尤其在大规模节点环境下表现尤为突出。此外,GWPBFT 结合了信誉分机制,动态调整小组成员的决策权重,使得高信誉节点在共识过程中占据更大影响力,有效抑制了低信誉节点的干扰,减少了无效的通信与计算负担,从而进一步降低了系统的通信开销。

本研究在由 75、100、125、150 和 175 个节点组成的网络环境中分别对 GWPBFT、SGPBFT、G-PBFT 和 PBFT 4 种共识算法的通信开销表现进行了测试。实验结果如图 4 所示。

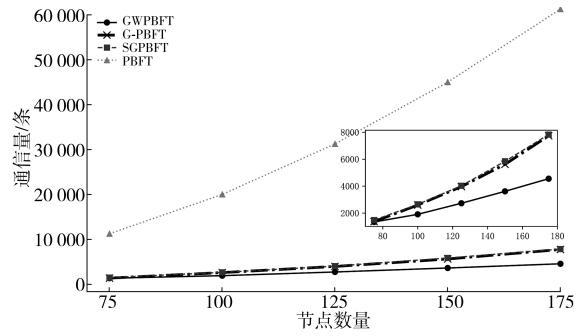


图 4 各算法通信开销对比

Fig.4 Comparison of communication overhead for different algorithms

从图 4 可见,在相同节点数量的情况下,GWPBFT 的通信开销明显较少。随着节点数量增加,PBFT 的通信开销呈现出快速且近似指数级的增长趋势;SGPBFT 的通信开销在节点数量不超过 100 时相对稳定,后增长速率明显加快;G-PBFT 在节点数量增加时,通信开销增长较为缓和,略优于 SGPBFT。相比之下,GWPBFT 的通信开销增长幅度则始终保持在较低水平,展现出较为平稳的增长趋势。与 PBFT、SGPBFT 和 G-PBFT 相比,当节点数量增加至 175 时,GWPBFT 的通信开销分别降低了 92.56%、41.81% 和 41.15%,体现出其在大规模节点下更高的网络通信效率。

3.4 共识成功率

共识成功率是指在共识过程中,网络节点成功达成一致的的概率,用以衡量共识算法在不同网络条件下实现有效共识的能力。GWPBFT 的决策权重策略使其在共识过程中能让诚实节点拥有更高的决策权重,从而能够有效抵御拜占庭节点

的干扰,提升系统的共识效率和稳定性。特别是在极端情况下,GWPBFT 能够有效防止恶意节点的影响,使系统维持高效的共识过程,从而表现出优越的抗拜占庭特性。

本研究设置了一个包含 100 个节点的网络,其中,出错概率为 $[0, 0.05]$ 的节点被定义为可靠节点,出错概率为 $[0.5, 1.0]$ 的节点为拜占庭节点。实验通过逐步增加网络中拜占庭节点的比例来测试共识成功率,每次增加全网节点的 5%。重复 100 次实验,并取其平均值作为最终测试结果。对 GWPBFT、SGPBFT、G-PBFT 和 PBFT4 种共识算法在不同情境下的实验结果如图 5 所示。GWPBFT 算法引入的权重衰减系数 k 对决策权重的分配具有重要作用,实验通过设定不同的 k 值(10、20、30)验证数调整对共识成功率的影响。

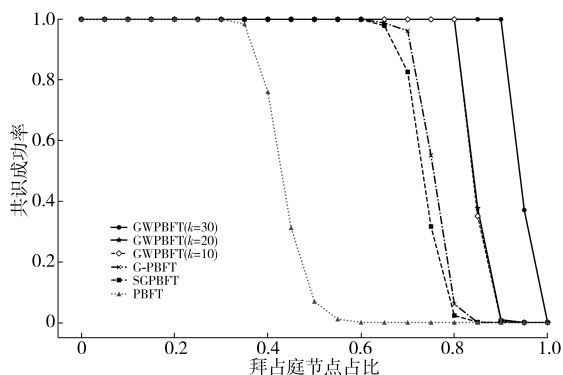


图 5 各算法的共识成功率对比

Fig.5 Comparison of consensus success rate for different algorithms

从图 5 可见,当拜占庭节点占全网节点 30% 以内时,各算法的共识成功率均表现稳定,几乎没有显著差异。然而,随着拜占庭节点比例的进一步增加,PBFT 的共识成功率最先开始显著下降,尤其是在拜占庭节点比例超过 50% 后迅速降至 0%,表现出该算法在高拜占庭节点比例下的低鲁棒性。当拜占庭节点比例增加至 65% 时,

SGPBFT、G-PBFT 和 GWPBFT 共识成功率分别为 97.9%、98.8% 和 100%,显示出较强的抗拜占庭容错能力。然而,在拜占庭节点占比超过 65% 后,SGPBFT 的共识成功率开始迅速下降。尽管 G-PBFT 略优于 SGPBFT,但随着拜占庭节点占比超过 70% 后,G-PBFT 的共识成功率也开始迅速下降。相较之下,GWPBFT(则在拜占庭节点占比达到 80% 之前始终保持稳定的共识成功率,直至拜占庭节点占比超过 80% 后,GWPBFT($k=10$ 和 $k=20$)才出现显著下降,而 GWPBFT($k=30$)在拜占庭节点占比超过 90% 后才开始明显下降。这表明 GWPBFT 在高拜占庭节点占比条件下具有更强的鲁棒性和稳定性;较大的 k 值($k=30$)使得系统在高比例拜占庭节点条件下表现出更强的鲁棒性;而较小的 k 值(如 $k=10$)虽然在低比例拜占庭节点条件下表现良好,但在高比例拜占庭节点时易受恶意节点影响。

4 结束语

本研究针对 PBFT 存在的一些局限性提出了一种基于分组和信誉权重策略的改进算法——GWPBFT。该算法通过提升信誉较高节点在共识过程中的决策权重,增强系统容忍拜占庭节点的能力,从而提高系统在不可靠环境下的安全性。为进一步提高效率,GWPBFT 引入了分组策略,在组内节点之间进行局部共识,从而有效减少了通信开销。实验结果表明,GWPBFT 在延迟、通信开销和共识成功率等方面均优于传统 PBFT 协议,展现了更高的性能和更强的鲁棒性。

然而,现阶段 GWPBFT 中的权重衰减系数 k 为静态设置,难以适应复杂多变的动态环境。未来研究可探索基于系统运行状态或网络环境的动态调整机制。例如,根据检测到的拜占庭节点比例或网络负载状态,自适应调整权重衰减系数,以在系统公平性与容错性之间实现动态平衡,从而进一步提升系统的适应性和实用性。

参考文献:

- [1] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system[EB/OL]. (2008-10-31) [2025-04-15]. <https://bitcoin.org/bitcoin.pdf>.
- [2] ZHENG Z B, XIE S A, DAI H N, et al. Blockchain challenges and opportunities: a survey[J]. International Journal of Web and Grid Services, 2018, 14(4): 352.
- [3] 张亮, 刘百祥, 张如意, 等. 区块链技术综述[J]. 计算机工程, 2019, 45(5): 1-12.
- [4] 刘昌平, 刘海, 李威. 一种区块链网络的 DDoS 攻击模型[J]. 计算机应用与软件, 2024, 41(11): 366-372, 392.

- [5] GRAMOLI V. From blockchain consensus back to Byzantine consensus[J]. *Future Generation Computer Systems*, 2020, 107:760–769.
- [6] 谢晴晴,董凡. 轻量级区块链技术综述[J]. *软件学报*, 2023, 34(1):33–49.
- [7] 张彭奕,宋杰. 区块链共识算法效能优化研究进展[J]. *计算机科学*, 2020, 47(12):296–303.
- [8] 谭朋柳,王润庶,曾文豪,等. 区块链共识算法综述[J]. *计算机科学*, 2023, 50(S1):691–702.
- [9] 何婧,豆天晨,陈琳,等. 基于 SVM 的 DPoS 共识机制改进[J]. *计算机应用研究*, 2024, 41(9):2598–2603.
- [10] 张学旺,罗欣悦,雷志滔,等. 基于节点识别的改进拜占庭容错共识算法[J]. *信息安全研究*, 2024, 10(9):818–823.
- [11] TANG S, WANG Z Q, JIANG J, et al. Improved PBFT algorithm for high-frequency trading scenarios of alliance blockchain [J]. *Scientific Reports*, 2022, 12(1):4426.
- [12] CHEN P Y, CHEN Y L, WANG X W, et al. A high-capacity slicing PBFT protocol based on reputation evaluation model [J]. *Wireless Networks*, 2024, 30(9):7469–7482.
- [13] 叶翰文,欧阳思源,包振强. 引入秘密共享的改进实用拜占庭容错算法[J]. *小型微型计算机系统*, 2023, 44(6):1146–1152.
- [14] ZHENG X D, FENG W L, HUANG M X, et al. Optimization of PBFT algorithm based on improved C4.5[J]. *Mathematical Problems in Engineering*, 2021, 2021(1):5542078.
- [15] WANG Y, ZHONG M L, CHENG T. Research on PBFT consensus algorithm for grouping based on feature trust[J]. *Scientific Reports*, 2022, 12(1):12515.
- [16] XU G Q, BAI H P, XING J, et al. SG-PBFT: a secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles[J]. *Journal of Parallel and Distributed Computing*, 2022, 164:1–11.
- [17] WU J Y, LIU Y, CAI J F, et al. Blockchain consensus mechanism for distributed energy transactions[J]. *Wireless Communications and Mobile Computing*, 2022, 2022(1):4314734.
- [18] WANG Z F, REN Y W, CAO Z Y, et al. LRBFT: improvement of practical Byzantine fault tolerance consensus protocol for blockchains based on Lagrange interpolation[J]. *Peer-to-Peer Networking and Applications*, 2023, 16(2):690–708.
- [19] HE F Y, FENG W L, ZHANG Y, et al. An improved Byzantine fault-tolerant algorithm based on reputation model[J]. *Electronics*, 2023, 12(9):2049.
- [20] 涂园超,陈玉玲,李涛,等. 基于信誉投票的 PBFT 改进方案[J]. *应用科学学报*, 2021, 39(1):79–89.

(责任编辑:方素华)