

文章编号:1007-5321(2020)06-0140-07

DOI:10.13190/j.jbupt.2020-206

无线网络中区块链共识算法的开销分析

曹 宾^{1,2}, 聂凯君^{1,2}, 彭木根^{1,2}, 周治中³, 张 磊⁴

(1. 北京邮电大学 网络与交换技术国家重点实验室, 北京 100876;

2. 北京邮电大学 信息与通信工程学院, 北京 100876;

3. 中电科网络空间安全研究院有限公司, 成都 610041; 4. 格拉斯哥大学 工程学院, 格拉斯哥 G12 8QQ)

摘要: 选取工作量证明(PoW)和实用拜占庭容错(PBFT)作为公/私链代表,对比分析了两者在无线网络中的系统资源消耗,为区块链类型的选择提供合理评估。首先,建立公平统一的网络模型和区块链标准流程;然后,考虑无线网络传输失败导致的区块丢失,推导分析了相应的PoW分叉和PBFT视图更换概率;最后,分析了无线网络规模对PoW和PBFT的通信开销和算力开销的影响。仿真结果表明,PBFT的算力开销远小于PoW,但PBFT的通信开销受节点规模的影响较大,可扩展性较差;PoW的通信开销受节点规模的影响相对平缓,可扩展性相对较好。

关键词: 区块链; 共识算法; 系统开销; 工作量证明; 实用拜占庭容错

中图分类号: TN92; TP311.13

文献标志码: A

Overhead Analysis of Blockchain Consensus Algorithm in Wireless Networks

CAO Bin^{1,2}, NIE Kai-jun^{1,2}, PENG Mu-gen^{1,2}, ZHOU Zhi-zhong³, ZHANG Lei⁴

(1. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China;

3. China Electrics Technology Cyber Security Company Limited, Chengdu 610041, China;

4. School of Engineering, University of Glasgow, Glasgow G12 8QQ, The United Kingdom)

Abstract: In order to provide a reasonable evaluation for the choice of blockchain type, proof of work (PoW) and practical Byzantine fault tolerance (PBFT) are treated as the benchmark of typical public and private chains for blockchain selection evaluation, and the overhead of blockchain in wireless networks is studied. Firstly, a fair network system and standard blockchain procedure have been provided, and then, considering the impact of block loss generated by transmission failure, the forking probability in PoW and view change probability in PBFT have been analyzed. Moreover, how network scale affects the communication and computation overhead in PoW and PBFT has been also investigated. Finally, the experimental results show that the computational overhead of PBFT is much smaller than that of PoW, the communication overhead of PBFT is greatly affected by the scale of the node, and thus the scalability of PBFT is poor. In contrast, the overhead of PoW communication is affected by the network scale linearly, and the scalability is much better compared with PBFT.

Key words: blockchain; consensus algorithm; system overhead; proof of work; practical Byzantine fault tolerance

收稿日期: 2020-10-10

基金项目: 国家自然科学基金项目(61701059); 中央高校基本科研业务费专项项目; 四川省国际科技创新合作/港澳台科技创新合作项目(2019YFH0163); 四川省科技厅重要研究开发项目(2018JZ0071)

作者简介: 曹 宾(1983—), 男, 副教授, 博士生导师, E-mail: caobin@bupt.edu.cn.

区块链是一种在对等网络 (P2P, peer to peer) 中构建信任并达成节点共识的分布式记账技术, 而数字货币的领头者比特币更是证明了该技术的成功. 近年来, 区块链技术发展迅速, 被广泛应用在金融、物流、医疗、教育等领域, 成为关注的热点.

根据准入机制, 区块链可以划分为公链、联盟链和私链三类^[1]. 私链较联盟链而言, 中心化程度更高, 但从广义上来讲, 联盟链也可以划分在私链范畴 (后续提及的私链指广义上的私链, 即包括联盟链在内). 目前, 对于公链和私链的研究工作主要集中在性能和安全性等方面, 缺乏开销方面的研究, 尤其是在越来越受关注的无线网络中. 因此, 笔者旨在研究无线网络中的区块链开销问题.

1 相关工作

对于公链共识算法的研究, 主要集中在性能和安全性上. Eyal 等^[2-3] 提出了工作量证明 (PoW, proof of work) 的改进算法. Shahsvari 等^[4] 分析了网络参数与分叉概率之间的关系. 对于私链共识算法的研究, 性能问题是其研究的重点. Zhang 等^[5-6] 提出了基于实用拜占庭容错 (PBFT, practical byzantine fault tolerance) 的改进算法. 上述的研究工作都是基于有线网络中的区块链共识研究, 随着区块链的发展以及 5G 时代的到来, 无线网络中的区块链共识研究开始受到关注. Jiang 等^[7] 提出了一种在无线网络中用于实时物联网应用的无许可拜占庭共识协议. Xu 等^[8] 研究了存在恶意干扰情况下无线区块链网络的安全性能.

尽管目前一些学者开展了无线网络中区块链共识的研究工作, 但是相对于有线网络中的研究工作很少. 而无线网络中的区块链共识开销问题, 至今尚未得到关注. 在有线网络中, 针对恶意攻击造成的分叉和视图更换进行了很多研究, 然而在没有恶意攻击的情况下, 在无线网络中传输失败、分叉和视图更换也有可能发生, 会对区块链的共识开销造成

影响. 这些问题在此前的工作中尚未充分考虑, 这是研究的出发点.

2 网络模型以及共识流程

2.1 区块链流程

假设网络中共有 N 个诚实节点, 随机分布在一定一跳通信范围内^[7], 即任意 2 个节点之间的消息可以直达. 节点之间通过无线信道传输数据, 并且数据只发送 1 次, 失败后不再重传 (重传次数为 0, 后续分析可以根据实际情况扩展).

区块链标准流程如图 1 所示. 每当交易到达, 节点立刻将交易广播并对来自其他节点的交易进行验证收集, 全网节点的交易到达率为 λ . 全网节点执行 PoW/PBFT 共识算法, 当共识达成之后, 节点将新生成的区块写入本地区块链.

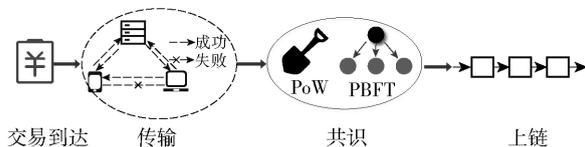


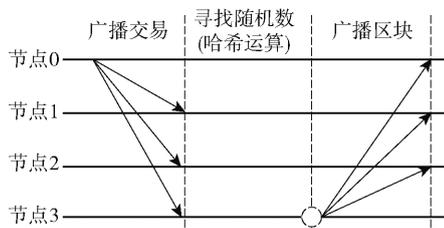
图 1 区块链标准流程

2.2 PoW 共识流程

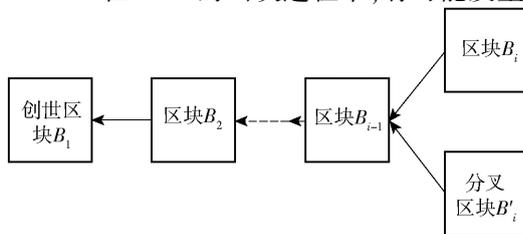
在 PoW 中, 假设 N 个节点算力均相同, 且为 r_1 , 哈希算法的目标难度值为 D . PoW 区块链网络的正常出块流程^[9] 如图 2(a) 所示, 包括以下 3 个步骤:

- 1) 节点将自身交易广播, 并且对其他新交易进行验证并收集, 全网节点同时参与计算, 寻找满足要求的随机数;
- 2) 某个节点率先找到符合难度值要求的随机数, 获得新区块的记账权, 并且立刻将该区块向全网广播;
- 3) 其他节点接收到新区块后, 验证区块中的交易和随机数的有效性, 然后将该区块加入本地区块链, 开始下一个区块的构建.

在 PoW 的出块过程中, 有可能发生分叉. 当某



(a) PoW 正常出块流程



(b) PoW 分叉

图 2 PoW 正常出块与分叉

个节点率先找到正确的随机数,向其他节点广播新区块时,由于节点间消息可能传输失败,一些节点并没有收到新区块的广播,这些节点会继续当前块的工作,从而产生意外的分叉情况,如图2(b)所示。

2.3 PBFT 共识流程

在 PBFT 中,共有 N 个节点 ($N = 3f + 1$, f 为可容忍的恶意节点数),包括一个主节点和 $N - 1$ 个副本节点,假设节点算力均相同,且为 r_2 。PBFT 区块链网络的正常出块流程^[10]如图3(a)所示,包括以下5个步骤。

1) 节点(指主节点和副本节点在内的所有节

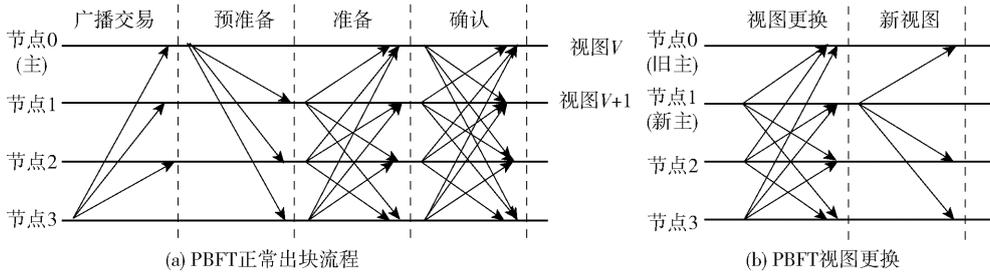


图3 PBFT 正常出块及视图更换

如果 PBFT 共识过程未能正常出块,就会执行视图更换协议。由于传输失败,使得确认阶段中节点接收不到 $2f + 1$ 以上条确认消息,从而可能触发视图更换协议。记当前视图编号为 v ,视图更换^[10]流程如图3(b)所示,包括以下3个步骤:

1) 副本节点认为主节点存在问题,向其他节点广播一条视图更换消息,当前活跃的编号最小的节点成为新的主节点;

2) 新的主节点收到 $2f + 1$ (包括主节点) 条有效的视图更换消息后,视图进入 $v + 1$ 状态,并向其他副本节点广播一条新视图消息;

3) 副本节点接收到新的主节点的新视图消息,验证其有效性,将视图切换到 $v + 1$ 状态。

当视图更换发生时,PBFT 的系统开销也会有所增加。

3 共识开销分析

3.1 节点传输成功概率

在无线网络中,传输失败可能会引起发生区块分叉或者视图更换,从而增加网络的系统开销。因此在进行开销分析之前,需要先对节点广播时的传输成功概率进行分析。

假设节点服从密度为 γ 的二维泊松点过程。随

机选择一个节点作为发送节点,以其为圆心,接收节点分布于半径为 R 的区域内。根据二维泊松点分布的性质,发送节点到接收节点之间的距离 r 的概率密度函数为

2) 主节点从交易池中取出,交易打包成新的区块,将区块广播至全网节点。

3) 副本节点接收到区块后,验证区块中交易的有效性,然后将区块的哈希摘要广播至全网。

4) 当节点收集到 $2f$ 条其他节点发来的摘要都和自己相同时,则向全网广播一条确认消息。

5) 当节点收到 $2f + 1$ (包括节点本身) 条相同的确认消息后,就会正式将新区块及其包含的交易提交到本地的区块链。

机选择一个节点作为发送节点,以其为圆心,接收节点分布于半径为 R 的区域内。根据二维泊松点分布的性质,发送节点到接收节点之间的距离 r 的概率密度函数为

$$f(r) = \frac{d(r^2/R^2)}{dr} = \frac{2r}{R^2} \quad (1)$$

设定节点传输的信道为瑞利信道,根据无线通信中小尺度衰落的特性,接收节点处的信噪比可以表示为

$$g = \frac{Shr^{-\alpha}}{\sigma^2} \quad (2)$$

其中: S 为节点的发射功率; h 代表瑞利衰落中非负的功率增益随机变量,服从指数为 1 的负指数分布; α 为路径损耗指数; σ^2 为干扰噪声功率。

设定系统的信噪比阈值为 Z ,根据泊松点过程性质^[11],节点传输的平均成功概率 P_s 为

$$P_s = \int_0^R P\{g > Z\} f(r) dr = \frac{2\pi\gamma}{N} \int_0^{\sqrt{N/(m\gamma)}} \exp\left\{-\frac{\sigma^2 r^\alpha Z}{S}\right\} r dr \quad (3)$$

以此概率作为广播时的节点传输成功概率。

3.2 PoW 非恶意分叉

在 PoW 共识过程中,假设某个节点 k 率先找到了正确的随机数,获得新区块的记账权,并且向其他

节点广播新区块. 其中有 $M (M \leq N-1, M = N-1$ 时不发生分叉) 个节点传输成功, 这些节点验证区块的有效性后立即放弃当前区块的工作, 并基于新区块寻找下一个区块, 将这 M 个传输成功的节点和节点 k 定义成集合 $Y = \{1, 2, \dots, M, M+1\}$. 剩下 $N-M-1$ 个节点由于传输失败, 会继续寻找当前区块, 将这些传输失败的节点定义成集合 $Z = \{1, 2, \dots, N-M-1\}$. 集合 Y 中的节点和集合 Z 中的节点开始竞争寻找随机数, 当集合 Z 中的节点先找到符合难度值要求的随机数并生成区块广播后, 就会产生分叉. 记集合 Y 产生区块的时间为 T_Y , 集合 Z 产生区块的时间为 T_Z . 根据目标难度值 D 和节点的算力, 可以得出集合生成区块的时间为^[12]

$$T_Z = \frac{D}{(N-M-1)r_1}, M \neq N-1 \quad (4)$$

对于任意的 M 值, 集合 Z 中的节点先生成区块的概率可以表示为

$$P\{T_Z < T_Y\} = P\{T = T_Z\} = \frac{(N-M-1)r_1}{Nr_1} = \frac{N-M-1}{N} \quad (5)$$

根据节点传输成功的概率 P_s , 在遍历完 M 个所有取值的情况后, 可以推导出 PoW 发生分叉的概率 P_f , 其表达式为

$$P_f = \sum_{M=0}^{N-2} C_{N-1}^M P_s^M (1-P_s)^{N-M-1} P\{T_Z < T_Y\} = \frac{N-1}{N} (1-P_s) \quad (6)$$

3.3 PBFT 非恶意视图更换

在 PBFT 共识过程中, 当确认阶段接收到 $2f+1$ 以上个确认消息的节点个数少于 $2f+1$, 就会触发视图更换协议^[10], 视图从 v 更换到 $v+1$.

PBFT 不发生视图更换须满足预准备阶段成功接收消息的节点数目不少于 $2/3$, 准备、确认 2 个阶段中成功接收 $2f$ 以上个节点消息的节点个数超过 $2/3$. 因此, PBFT 不发生视图更换的概率 P_v 可以表示为

$$P_v = P\{L_1 \geq 2f, L_2 \geq 2f+1, L_3 \geq 2f+1\} \quad (7)$$

其中: L_1 为 PBFT 预准备阶段接收到消息的节点个数, L_2 和 L_3 分别为准备、确认 2 个阶段中成功接收 $2f$ 以上个节点消息的节点个数.

根据前面推导的节点传输成功概率 P_s , 可以得到准备、确认 2 个阶段中单个节点接受 $2f$ 以上个消息的概率 P_{s_2} 和 P_{s_3} 分别为

$$P_{s_2} = \sum_{M_2=2f}^{L_1} C_{L_1}^{M_2} P_s^{M_2} (1-P_s)^{L_1-M_2} \quad (8)$$

$$P_{s_3} = \sum_{M_3=2f}^{L_2} C_{L_2}^{M_3} P_s^{M_3} (1-P_s)^{L_2-M_3} \quad (9)$$

其中 M_2 和 M_3 分别为准备、确认 3 个阶段中单个节点成功接收到的消息个数.

因此, PBFT 发生视图更换的概率 P_v 为

$$P_v = 1 - P_{\bar{v}} = 1 - \sum_{L_1=2f}^{3f} \left\{ C_{3f}^{L_1} P_s^{L_1} (1-P_s)^{3f-L_1} \sum_{L_2=2f+1}^{3f+1} \left\{ C_{3f+1}^{L_2} P_{s_2}^{L_2} (1-P_{s_2})^{3f+1-L_2} \sum_{L_3=2f+1}^{3f+1} C_{3f+1}^{L_3} P_{s_3}^{L_3} (1-P_{s_3})^{3f+1-L_3} \right\} \right\} \quad (10)$$

3.4 系统开销分析

基于对非恶意情况下 PoW 发生分叉以及 PBFT 发生视图更换的分析, 接下来将进一步研究非恶意情况下 PoW 和 PBFT 的系统开销, 包括通信开销和算力开销.

定义 T_1 为 PoW 正常生成一个区块的平均时间, T_2 为 PBFT 正常生成一个区块的平均时间, T_3 为 PBFT 发生视图更换的时间. 对于 PoW, 生成一个区块的时间包括: ① 打包生成区块时间; ② 寻找随机数时间; ③ 广播时延; ④ 验证区块时间. 考虑到 PoW 寻找随机数占绝大部分时间 (数量级 ② 为分级、③ 为秒级、①④ 为毫秒级), 可以忽略其他时间, 因此, $T_1 \approx T_d$, T_d 是寻找随机数的平均时间. 根据节点算力和目标难度值 D , 可以得出生成区块时间^[12] $T_1 = T_d = D/Nr_1$.

对于 PBFT, 生成一个区块的时间包括: ① 打包生成区块时间; ② 广播时延; ③ 验证区块时间. T_c 为 PBFT 验证区块时间, T_b 为广播一次时延, 考虑到 PBFT 广播占大部分时间 (数量级 ② 为秒级、①③ 为毫秒级), 忽略其他时间, 因此 T_2 可以表示成 $T_2 = 3T_b$. 而 PBFT 发生视图更换时, 进行了 2 次广播, 因此 $T_3 = 2T_b$.

1) 通信开销

采用生成一个有效区块所需要的平均通信次数来衡量 PoW 和 PBFT 的通信开销.

在 PoW 中, 记发生分叉时的通信次数为 C_f , 不发生分叉时的通信次数为 C_i , 那么一个有效区块内的通信次数可以表示为

$$C_{PoW} = (1 - P_f)C_{\bar{f}} + P_f C_f \quad (11)$$

PoW 不发生分叉时, T_1 内生成一个有效区块, 这段时间内交易的到达数为 λT_1 , 则不发生分叉时的通信次数为

$$C_{\bar{f}} = \lambda T_1 (N - 1) + N - 1 \quad (12)$$

PoW 发生分叉时, 先在 T_1 内生成一个区块, 之后的 T_z 内生成一个分叉块, 则发生分叉时的通信次数为

$$C_f = C_{\bar{f}} + \lambda T_z (N - 1) + N - 1 \quad (13)$$

因为 $T_z = D / (N - M - 1) r_1$, 可得到 T_z 与 T_1 的关系, 即 $T_z = N T_1 / (N - M - 1)$, $M \neq N - 1$. 根据分叉概率 P_f , 可以推导出 PoW 的通信次数 C_{PoW} 为

$$C_{PoW} = (1 - P_f) C_{\bar{f}} + P_f C_f = \lambda T_1 (N - 1) (2 - P_s^{N-1}) + \frac{(N - 1)^2}{N} (1 - P_s) + N - 1 \quad (14)$$

在 PBFT 中, 记不发生视图更换时的通信次数为 $C_{\bar{v}}$, 从执行 PBFT 到发生视图更换期间的通信次数为 C_v . 假设在生成一个有效区块之前, PBFT 连续发生了 m 次视图更换, 则 PBFT 生成一个有效区块内的通信次数 C_{PBFT} 为

$$C_{PBFT} = \begin{cases} \sum_{m=0}^{\infty} P_v^m (1 - P_v) (m C_v + C_{\bar{v}}), & P_v \neq 1 \\ \infty, & P_v = 1 \end{cases} \quad (15)$$

PBFT 不发生视图更换时, T_2 内产生一个有效区块, 这段时间内交易的到达数为 λT_2 , 则不发生视图更换时的通信次数为

$$C_{\bar{v}} = \lambda T_2 (N - 1) + 2N(N - 1) \quad (16)$$

当 PBFT 发生视图更换时, 先在 T_2 内执行 PBFT 协议, 由于传输失败, 继续在 T_3 内执行视图更换协议. T_3 内, 交易到达数为 λT_3 , 增加通信次数为 $\lambda T_3 (N - 1)$. 记执行视图更换协议的通信次数为 C_v , 那么 C_v 可以表示为

$$C_v = C_{\bar{v}} + \lambda T_3 (N - 1) + C_+ \quad (17)$$

执行视图更换协议的通信次数执行 PBFT 协议的程序有关. L_1 是 PBFT 预准备阶段接收到消息的节点个数, L_2, L_3 分别是准备、确认 2 个阶段中成功接收 $2f$ 以上个节点消息的节点个数. PBFT 预准备阶段中 L_1 小于 $2f$ 、或者准备、确认阶段中任何一个阶段中的 L_2, L_3 小于 $2f + 1$, 都会触发视图更换, 而 L_3 的值直接决定视图更换协议的通信次数. 视图更换协议的通信次数与 L_3 的关系为

$$C_+ = (N - L_3)(N - 1) + N - 1 = 3f(3f + 2 - L_3) \quad (18)$$

当 L_1 小于 $2f$ 或者 L_2 小于 $2f + 1$, PBFT 确认阶段任意一个节点接收到的消息数一定小于 $2f + 1$, 即 L_3 一定等于 0. 而当 L_1 不小于 $2f$ 并且 L_2 不小于 $2f + 1$ 时, L_3 的取值可以从 0 到 $2f$. 因此执行视图更换协议的通信次数为

$$C_+ = P(L_1 < 2f + 1) 3f(3f + 2) + P(L_2 < 2f + 1 | L_1 \geq 2f + 1) 3f(3f + 2) + P(L_3 < 2f + 1 | L_1 \geq 2f + 1, L_2 \geq 2f + 1) 3f(3f + 2 - L_3) \quad (19)$$

因此, C_v 可以表示成式 (20).

$$C_v = C_{\bar{v}} + \lambda T_3 (N - 1) + C_+ = \lambda (T_2 + T_3) (N - 1) + 2N(N - 1) + C_+ = 3f\lambda (T_2 + T_3) + 6f(3f + 1) + 3f(3f + 2) \times \left(1 - \sum_{L_1=2f}^{3f} \left\{ \sum_{L_2=2f+1}^{3f+1} \left\{ \sum_{L_3=2f+1}^{3f+1} C_{3f+1}^{L_3} P_{s_3}^{L_3} (1 - P_{s_3})^{3f+1-L_3} \right\} \right\} - \sum_{L_1=2f}^{3f} \left\{ \sum_{L_2=2f+1}^{3f+1} \left\{ \sum_{L_3=0}^{2f} C_{3f+1}^{L_3} P_{s_3}^{L_3} (1 - P_{s_3})^{3f+1-L_3} 3fL_3 \right\} \right\} \right) \quad (20)$$

计算出 C_v 和 $C_{\bar{v}}$ 后, 可以将式 (15) 简化为

$$C_{PBFT} = \begin{cases} P_v / (1 - P_v) C_v + C_{\bar{v}}, & P_v \neq 1 \\ \infty, & P_v = 1 \end{cases} \quad (21)$$

从而得出 PBFT 生成一个有效区块所需要的通信次数.

2) 算力开销

采用生成一个有效区块所需要的平均哈希次数来衡量 PoW 和 PBFT 的算力开销.

在 PoW 中, 算力的消耗是用来寻找符合难度值要求的随机数, 这一过程的时间近似为出块的时间. PoW 不发生分叉时, T_1 内的哈希次数为 $N r_1 T_1$, 当发生分叉时, $T_1 + T_z$ 内的哈希次数为 $N r_1 (T_1 + T_z)$. 根据分叉概率 P_f , 生成一个有效区块所需要的哈希次数 H_{PoW} 为

$$H_{PoW} = (1 - P_f) N r_1 T_1 + P_f N r_1 (T_1 + T_z) = N r_1 T_1 (2 - P_s^{N-1}) \quad (22)$$

在 PBFT 中,算力的消耗是对区块的验证,这一时间占比很小,只占区块生成时间的一部分. PBFT 发生视图更换时,并没有对区块的验证. 因此 PBFT 算力的消耗考虑 PBFT 运行时,对于区块中交易的验证,这一时间为 T_c . 从而可以计算出生成一个有效区块所需要的哈希次数 H_{PBFT} 为

$$H_{PBFT} = \begin{cases} \sum_{m=0}^{\infty} P_v^m (1 - P_v) (m + 1) N r_2 T_c, & P_v \neq 1 \\ \infty, & P_v = 1 \end{cases} \quad (23)$$

简化后可以表示为

$$H_{PBFT} = \begin{cases} 1/(1 - P_v) N r_2 T_c, & P_v \neq 1 \\ \infty, & P_v = 1 \end{cases} \quad (24)$$

4 仿真与分析部分

对 PoW 分叉概率和 PBFT 视图更换概率以及 PoW 和 PBFT 的通信开销和算力开销进行了评估. 一些关键参数的设定如下:全网节点的交易到达率 $\lambda = 4\text{TPS}$ (TPS, transactions per second), 传输时延 $T_b = 0.5\text{ s}$, PBFT 交易验证时间 $T_c = 5\text{ ms}$, PBFT 节点算力 $r_2 = 1\text{ MH/s}$, PoW 节点算力 $r_1 = 1\text{ MH/s}$, 当节点个数为 100 时, PoW 的难度值 $D = 2.0\text{ GH}$.

PoW 分叉概率随节点传输成功概率的变化曲线如图 4 所示. 由图 4 可知,随着节点传输成功概率的增加, PoW 分叉概率线性减小,并且节点数量越多,传输成功概率与分叉概率的相关性越强. 而分叉概率本质上就是未接收到记账节点广播的节点算力之和与节点总算力之和的比. 当节点传输成功概率增加时,未接收到记账节点广播的节点算力和降低, PoW 分叉概率减小.

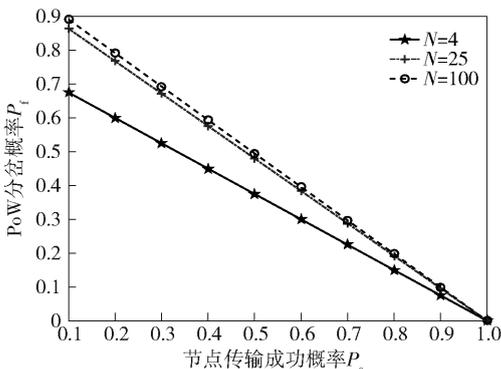


图 4 不同节点传输成功概率下的 PoW 分叉概率

图 5 所示为 PBFT 视图更换概率随节点传输成功概率的变化趋势. 由图 5 可知,随着节点传输成

功概率的增加, PBFT 视图更换概率先保持不变(概率为 1),再减小到 0 后保持不变. 这与 PBFT 三阶段中需要 2/3 以上的节点确认才能进入下一个阶段相关. 当节点成功传输概率较低时, PBFT 三阶段未能正常进行,发生视图更换;当节点成功传输概率较高时, PBFT 三阶段正常进行,不发生视图更换.

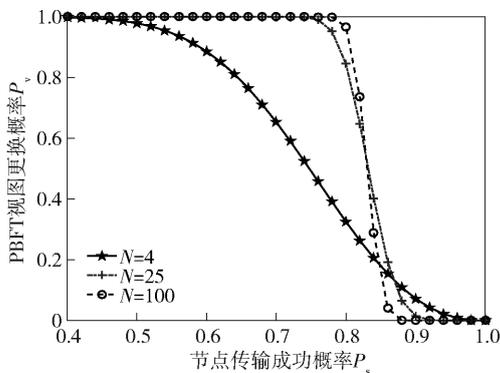


图 5 不同节点传输成功概率下的 PBFT 视图更换概率

图 6 所示为 PoW 和 PBFT 的通信开销随节点个数的变化曲线图. 由图 6 可知, PoW 的通信次数与节点数目成正比; PBFT 的通信次数与节点数目的二次方成正比. 当节点数目较小时, PoW 的通信开销大于 PBFT, 当节点数目不断增大, PBFT 的通信开销将远远大于 PoW. 在节点传输成功概率为 0 和不为 0 两种情况下,在通信开销相同时 PBFT 的节点数目较 PoW 有所增加.

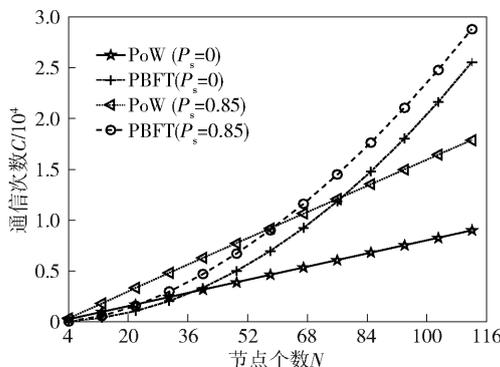


图 6 不同节点数量下的通信开销

图 7 所示为 PoW 和 PBFT 的算力开销随节点个数的变化曲线图. 由图 7 可知, PoW 的算力开销与节点数目成正比, PBFT 的算力开销远小于 PoW 的算力开销. 实际上,从式(24)可知, PBFT 的算力开销与节点数目也是成正比的,但 PBFT 的算力消耗相对 PoW 来说很小,几乎可以忽略.

综上可以分析, PBFT 的算力开销始终远小于

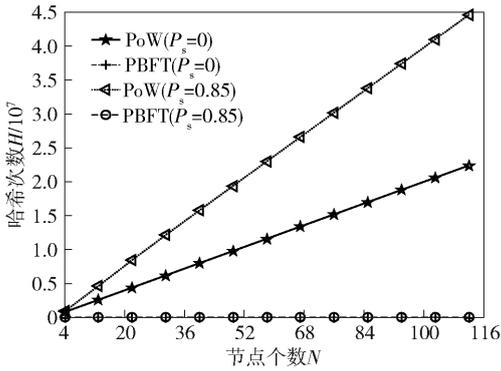


图 7 不同节点数量下的算力开销

PoW; 当节点数目小于 40 时, PBFT 的通信开销更小, 当节点数目在 40 ~ 60 之间, PoW 和 PBFT 通信开销相差不多, 当节点数目大于 60 时, PoW 的通信开销更小。因此, 在网络环境较为恶劣的情况下, 应当选择 PoW 作为共识算法, 而在网络环境良好的情况下, 应当根据网络节点规模的数目来灵活选择 PoW 或者 PBFT 作为共识算法。

5 结束语

选取 PoW 和 PBFT 作为公/私链代表并建立公平统一的网络模型和区块链标准流程, 分析对比了无线网络中 PoW 和 PBFT 的系统开销。结果表明, PBFT 的算力开销始终远小于 PoW, PBFT 通信开销受节点规模的影响较大, 可扩展性较差, PoW 通信开销受节点规模的影响相对平缓, 可扩展性相对较好。PBFT 通过选举主节点以达成共识, 而 PoW 通过消耗算力寻找随机数达成共识, 这 2 种不同的机制决定了 PBFT 比 PoW 对网络的通信质量要求更高, 同时可扩展性也会受到限制, 而 PoW 寻找随机数要消耗大量的算力, 相对 PBFT 可扩展性较好。

参考文献:

- [1] 曾诗钦, 霍如, 黄韬, 等. 区块链技术研究综述: 原理、进展与应用[J]. 通信学报, 2020, 41(1): 134-151.
Zeng Shiqin, Huo Ru, Huang Tao, et al. Overview of blockchain technology research: principle, progress and application[J]. Journal on Communications, 2020, 41(1): 134-151.
- [2] 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. Santa Clara: USENIX Association, 2016: 45-59.

- [3] Kokoris-Kogias E, Jovanovic P, Gailly N, et al. Enhancing bitcoin security and performance with strong consistency via collective signing[J]. Applied Mathematical Modelling, 2016, 37(8): 5723-5742.
- [4] Shahsvari Y, Zhang K, Talhi C. A theoretical model for fork analysis in the bitcoin network[C]//IEEE International Conference on Blockchain. Atlanta: IEEE Press, 2019: 237-244.
- [5] Zhang L, Li Q. Research on consensus efficiency based on practical byzantine fault tolerance[C]//2018 10th International Conference on Modelling, Identification and Control. Guiyang: IEEE Press, 2018: 1-6.
- [6] Jiang Y, Lian Z. High performance and scalable byzantine fault tolerance[C]//IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference. Chengdu: IEEE Press, 2019: 1195-1202.
- [7] Jiang Z, Cao Z, Krishnamachari B, et al. SENATE: a permissionless byzantine consensus protocol in wireless networks for real-time Internet-of-Things applications[J]. Internet of Things Journal, 2020, 7(7): 6576-6588.
- [8] Xu H, Zhang L, Liu Y, et al. Raft based wireless blockchain networks in the presence of malicious jamming[J]. IEEE Wireless Communications Letters, 2020, 9(6): 817-821.
- [9] 黄俊飞, 刘杰. 区块链技术研究综述[J]. 北京邮电大学学报, 2018, 41(2): 1-8.
Huang Junfei, Liu Jie. A review of blockchain technology research[J]. Journal of Beijing University of Posts and Telecommunications, 2018, 41(2): 1-8.
- [10] Astro M, Liskov B. Practical byzantine fault tolerance[C]//The Third Symposium on Operating Systems Design and Implementation. [S. l.]: USENIX Association, 1999: 173-186.
- [11] Andrews J G, Baccelli F, Ganti R K. A tractable approach to coverage and rate in cellular networks[J]. IEEE Transactions on Communications, 2011, 59(11): 3122-3134.
- [12] BitFury G. Proof of stake versus proof of work[R/OL]. (2015-09-13) [2020-12-20]. <https://bitfury.com/content/downloads/pos-vs-pow-1.0.2.pdf>.