浙江科技学院学报，第35卷第1期，2023年2月
Journal of Zhejiang University of Science and Technology
Vol. 35 No. 1, Feb. 2023

doi：10.3969/j.issn.1671-8798.2023.01.007

基于通信时延分组的改进实用拜占庭容错算法

邵　楠，陈善圣，陈　宁
（浙江科技学院 机械与能源工程学院，杭州 310023）

摘　要：【目的】为解决实用拜占庭容错算法（practical Byzantine fault-tolerant algorithm，PBFT）通信复杂度高、共识时延高等不足，提出节点间通信时延分组的改进实用拜占庭容错算法（grouping PBFT，GPBFT）。**【方法】**首先将区块链系统节点进行分组，依据最少网络通信次数确定分组数；然后计算各组节点间平均通信时延进行组内节点筛选，确定组内节点数；最后以共识成功率、失败率和节点历史行为评估参数为变量计算节点信誉值，监督节点共识行为，减少异常节点的参与。**【结果】**通过基于Hyperledger Fabric平台的区块链系统进行仿真试验，结果表明：与PBFT相比，GPBFT平均时延降低57.86%、平均吞吐量提高55.04%，通信复杂度数量级由平方级降低为对数级。**【结论】**GPBFT可满足多节点场景下区块链复杂通信的高时效性需求，解决了行业区块链系统大规模节点的需求问题。

关键词：区块链；共识算法；实用拜占庭容错算法；信誉评分
中图分类号：TP393　　**文献标志码：**A　　**文章编号：**1671-8798(2023)01-0048-07

Improved practical Byzantine fault-tolerant algorithm based on communication delay grouping

SHAO Nan, CHEN Shansheng, CHEN Ning
(School of Mechanical and Energy Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, Zhejiang, China)

Abstract: [**Objective**] In response to the shortcomings of high communication complexity and high consensus delay that plague the practical Byzantine fault-tolerant algorithm (PBFT), an improved practical Byzantine fault-tolerant algorithm (grouping PBFT, GPBFT) was proposed for grouping inter-node communication delay. [**Method**] First, blockchain system nodes were grouped, and the number of groups was determined according to the minimum number of

收稿日期：2022-02-28
基金项目：国家重点研发计划重点专项项目(2019YFE0126100)；浙江省“一带一路”国际科技合作项目(2019C04025)
通信作者：陈　宁(1975—　)，男，陕西省汉中人，教授，博士，主要从事智能交通系统和智能网联汽车研究。E-mail：neilching@163.com。

network communication; then, the average communication delay between each group of nodes was calculated to screen the nodes in the group and determine the number of nodes in the group; finally, the consensus success rate, the failure rate and the historical behavior evaluation parameters of nodes were used as variables to calculate the reputation values of nodes, supervise the consensus behavior of nodes, and reduce the participation of abnormal nodes. [**Result**] Through the simulation test of the blockchain system based on the Hyperledger Fabric platform, the results show that compared with PBFT, the average delay of GPBFT decreases by 57.86%, the average throughput increases by 55.04%, and the order of magnitude in communication complexity decreases from the square to logarithmic. [**Conclusion**] GPBFT can meet the high timeliness requirements of complex blockchain communication in multi-node scenarios, and fulfill the needs of large-scale nodes of the industry blockchain system.

Keywords: blockchain; consensus algorithm; practical Byzantine fault-tolerant algorithm; reputation rating

共识算法已成为区块链技术向数字金融、车联网等领域深层渗透研究的热点和难点[1]。实用拜占庭容错算法(practical Byzantine fault-tolerant algorithm，PBFT)由 Miguel 和 Barbara 首次提出，因其启动节点数量少、共识效率和容错率高而在私有链和联盟链中得到大量应用[2]。PBFT 是一种基于状态机复制的共识算法，在分布式系统的不同节点中进行副本复制，每个状态机的副本都保存了服务的状态和所实现的操作，其原理是让每个收到消息的节点都去询问其他节点收到的消息内容，在发生错误的节点比例不超过节点总数 1/3 的情况下仍可保证系统正常运行[3]。

人们对 PBFT 进行了大量的研究。He 等[4]提出平均主义实用拜占庭容错算法，优化 PBFT 中选择主节点的过程，从而使链中的每个节点都是平等且高效的，提高了数据备份和验证的效率，优化了区块链的共识过程。Gueta 等[5]为区块链提出了一种可扩展的分散信任基础设施，解决了可扩展性和分散性两大挑战，同时支持 Ethereum[6]和智能合约[7]的执行。Li 等[8]针对 PBFT 不适合动态网络的问题提出改进的可扩展 PBFT 算法，可以根据系统的网络环境采取不同的步骤来达成共识，该算法使用可验证随机函数来选择共识节点，简化节点协议和视图更改协议，以减少通信开销和共识所需时间。涂园超等[9]提出基于信誉投票的 PBFT 改进方案，根据节点划分评估机制动态地选取可靠节点参与共识，并根据节点状态转移机制转换节点的角色，提高了系统共识的安全性与稳定性。唐宏等[10]在 PBFT 中引入节点基础配置评分及信誉评分机制，依据信誉评分机制计算节点可靠性，并根据节点可靠性来选取领导节点与共识节点，以减少参与共识的节点数。上述研究均基于共识流程的优化与节点数量的调整，因此 PBFT 算法仍存在随节点数增加，系统共识时延增大、吞吐量下降，同时通信复杂度呈平方级增长的问题。对此，本研究提出一种基于节点间通信时延分组的实用拜占庭容错算法(grouping PBFT，GPBFT)。通过将节点间通信时延较小的节点进行分组，从而在减少共识节点数量的基础上降低共识时延、提升吞吐量并降低通信复杂度的数量级；同时设计节点信誉评分机制，有效降低恶意节点参与共识过程的概率，提高系统共识的稳定性与安全性。

1　GPBFT 概述

GPBFT 由领导节点(leader node，LN)、备份节点(replica node，RN)和客户端构成，GPBFT 框架如图 1 所示。在 GPBFT 中，LN 是小组内主节点，其作用为组内共识请求的广播与验证消息的收集；RN 是参与组内共识的节点，负责接收与验证由 LN 和其他 RN 发送的消息；客户端是共识的请求者，不参与共识的具体过程。

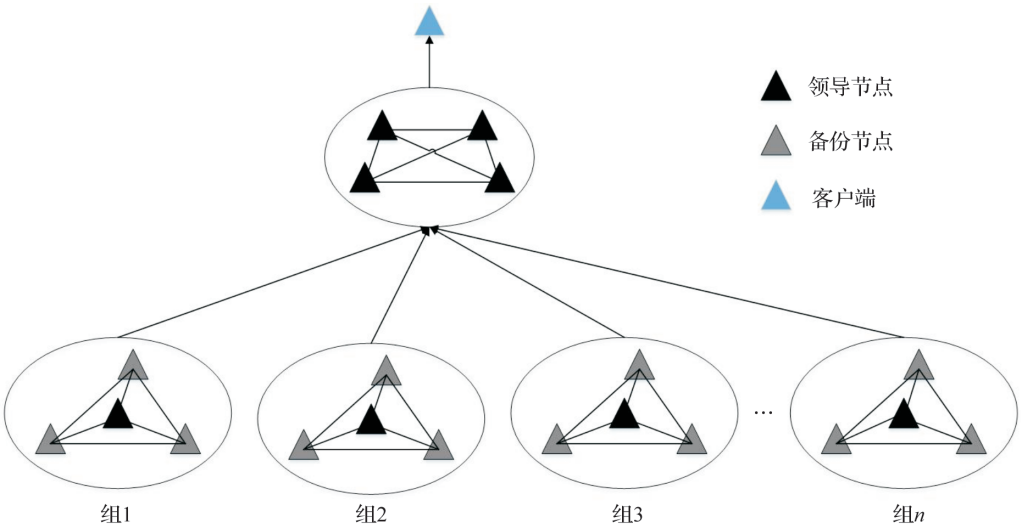

图 1 GPBFT 框架

Fig. 1 GPBFT framework

GPBFT 流程如图 2 所示。GPBFT 主要包含两部分：第一部分为组内共识，首先区块链系统提供节点集合$\{Z\}$，依据最少网络通信次数原则确定分组数 m，然后计算各组内节点间平均通信时延，据此选出组内低于平均时延的节点并确定各组内节点数，同时依据组内最低通信时延与节点信誉评分机制选出各组内 LN，最后由组内 LN 与 RN 构成共识节点进行组内共识；第二部分为全局 LN 共识，完成组内共识后，各组的 LN 组成 LN 小组，进行全局 LN 共识，对于产生共识请求的具体组号，则由发起共识请求的客户端位置决定。

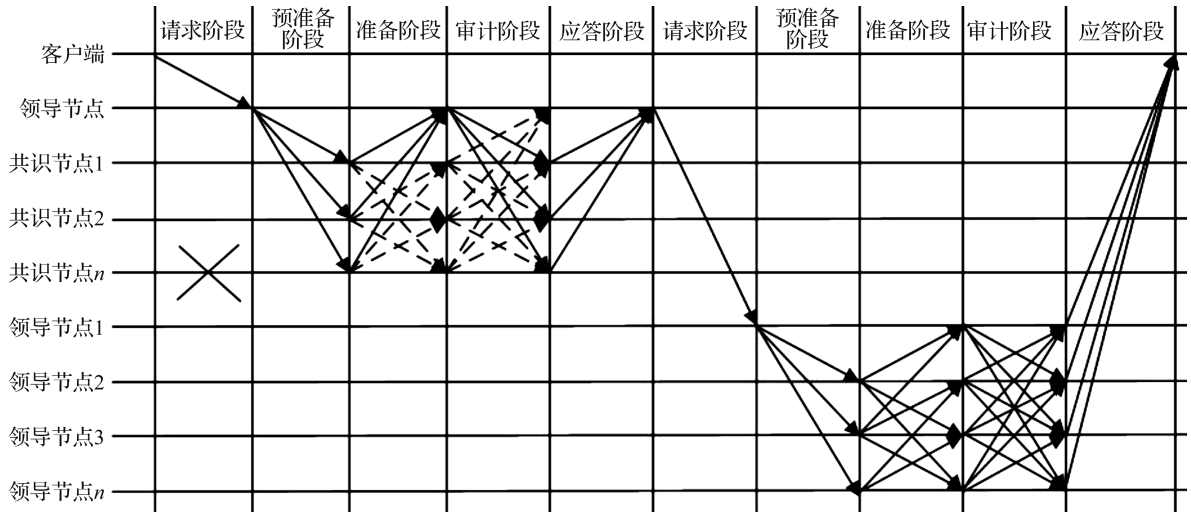

图 2 GPBFT 流程

Fig. 2 GPBFT flow

共识流程具体如下：

1）组内请求阶段。在完成节点分组与组内 LN 选择后，由客户端发起共识请求，进行小组内的共识。

2）组内预准备阶段。在预准备阶段，所有组内节点都必须接收由该组 LN 发出的请求消息，并给每个共识节点广播事务的执行顺序。LN 对需要放置在复制节点中的新区块中的多个事务进行排序，并将它们存储在列表中，然后向整个组内广播该列表。

3）组内准备阶段。各节点收到交易列表后，对交易的完整性和合法性进行验证和审计。将审计结果添加到每个节点的数字签名中，并广播给其他非 LN。

4）组内审计阶段。RN 接收并总结来自其他 RN 的审计结果，并将它们与自己的审计结果进行比

较，然后向其他节点广播确认消息。

5）组内应答阶段。当 LN 和 RN 都收到一定数量的相同审计结果时，RN 将其记录的审计结果反馈给组内 LN。若组内共识结果在审计后无误，则由该组内的 LN 向其他小组的 LN 进行广播，申请 LN 共识。

6）全局 LN 共识。在完成组内共识后，组内 LN 向全局的 LN 发起共识，共识过程与组内共识流程一致。若存在异常节点，且异常节点 f 与全局节点 Z 之间满足 $f \leqslant (Z-1)/3$ 时，则采用少数服从多数的基本原则，不影响共识结果。达成共识后，各节点一致同意将新区块加入区块链。

GPBFT 继承 PBFT 的基本共识流程，保留系统存在恶意节点时的容错率，因而提高了系统的安全性。GPBFT 考虑到实际节点数量激增导致的高时延与节点信誉问题，对 PBFT 进行了如下改进：1）对系统全局节点进行分组，将 PBFT 的全局共识模式改进为小组内共识，从而大大减少了参与共识的节点数量，提高了共识效率；2）将节点间通信时延较小的节点进行归类分组，从而在减少共识数量的基础上降低共识时延；3）提出了节点信誉评分机制，依据节点信誉评分来进行 LN 选择，从而减小了恶意节点参与共识的概率。

2　GPBFT 实现

2.1　基于节点间通信时延的节点分组算法

随着实际网络与节点性能的变化，节点间通信时延情况会出现波动，影响共识效率。对此，本研究提出一种基于节点间通信时延的分组算法，通过计算节点间通信时延，将低于平均通信时延的节点归为一组，有效降低了共识过程中的通信消耗。节点分组原理如图 3 所示。

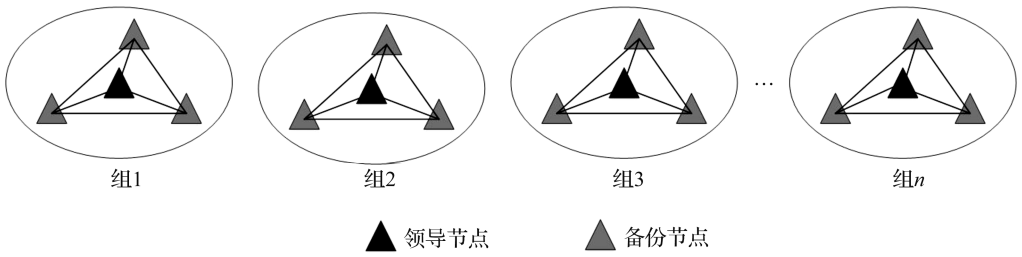

图 3　节点分组原理

Fig. 3　Node grouping principle

下面介绍分组算法具体流程。

首先，计算初始节点集合$\{Z\}$中的节点间通信时延。通信时延包括发送时延、传播时延、处理时延和排队时延[11]。在此处将上述 4 种时延相加作为系统节点的总时延，记为 D。那么，两节点之间的时延可计算如下：

$$D = D_j - D_i。\tag{1}$$

式(1)中：D_j 为系统节点数量为 n 的区块链网络中节点 i 接收到节点 j 响应的时间；D_i 为节点 i 发送的请求时间。为了判断节点在系统全网中通信时延的相对大小，我们提出节点平均通信时延 $\overline{D}$，计算如下：

$$\overline{D} = \frac{\sum_{j=i+1}^{n} D}{n-1}。\tag{2}$$

然后，预选组内 LN。由计算得到的节点平均通信时延来统计各组中通信时延最小的节点，得到组内 LN。

最后，筛选组内 RN。完成组内 LN 统计后，对小组内节点间时延进行升序排序，将通信时延大于 $\overline{D}$ 的节点移出组外，以保证组内节点的通信时延处于较低水平。移出组外的节点会与其他小组节点进行通信时延计算，若通信时延小于组内平均值，则加入该小组内；若通信时延仍然大于其他小组平均值，则节点无法加入该组，该节点会与其他类似节点进行组队。

2.1.1　GPBFT 网络通信次数分析

假设系统节点总数 Z 分为 $m(m \geqslant 3)$ 组，各组中有 $n+1(n \geqslant 3)$ 个节点，则系统单次共识通信次数计算

如下：

$$S_1=m(n+1)^2+m^2。\tag{3}$$

PBFT 完成一次共识，系统通信次数计算如下：

$$S_2=2Z^2-Z-1。\tag{4}$$

将 GPBFT 与 PBFT 通信次数进行比较，得

$$S_1-S_2=m(n+1)^2+m^2-2Z^2+Z+1。$$

由于 $Z=m(n+1)$，得

$$S_1-S_2=\left(\frac{1}{m}-2\right)Z^2+Z+m^2+1<0。$$

可见，GPBFT 的通信复杂度小于 PBFT。

2.2.2 GPBFT 网络通信最少次数计算

在分析最小通信次数时，同样假设系统节点总数 Z 分为 m（$m\geqslant 3$）组。将 $m=\frac{Z}{n+1}$ 代入式(3)，计算得

$$S_1=Z(n+1)+\frac{Z^2}{(n+1)^2}。\tag{5}$$

在式(5)中对 n 分别进行一阶和二阶求导，得：

$$\frac{\partial S_1}{\partial n}=Z-2Z^2\ \frac{1}{(n+1)^3};\tag{6}$$

$$\frac{\partial^2 S_1}{\partial n^2}=6Z^2\ \frac{1}{(n+1)^4}。\tag{7}$$

根据式(6)和式(7)求出极值点，即可在确保通信次数最低的情况下获得最佳小组数 m 与组内节点数。

2.2 节点信誉评估机制

节点信誉评估机制对参与共识的节点行为进行评估，以区分高信誉值节点与恶意行为节点。首先，给刚进入系统的节点 i（节点参与共识的轮数 $t=0$）赋予初始信誉值 $r(i,0)=0.5$；然后，引入共识错误率 f_i、共识成功率 s_i 和历史行为评估参数 a_i。

2.2.1 共识错误率

共识错误率 f 的设置是为了避免错误率较高的节点成为 LN，该参数对节点的信誉值有负面作用，定义如下：

$$f=\frac{f_i}{t_{\text{false}}},f\in[0,1]。\tag{8}$$

式(5)中：f_i 为节点 i 生成有效块失败的次数；t_{false} 为节点生成块失败的总次数。对于新节点或从未生成块的节点，其错误率 $f=0$。

2.2.2 共识成功率

共识成功率 s 衡量了节点成功参与生成区块的能力，优先选择成功率较高的节点作为 LN 或 RN，这样可以提高共识速度，定义如下：

$$s=\frac{s_i}{t_{\text{total}}},s\in[0,1]。\tag{9}$$

式(6)中：s_i 为节点 i 成功创建块的次数；t_{total} 为节点生成块的总数。对于新节点或从未生成块的节点，其成功率 $s=0$。

2.2.3 历史行为评估参数

历史行为评估参数 a_i 体现节点参与共识过程中的稳定性，能判断节点加入区块链网络后的发展情况，从而促使节点做出良好的行为以维持较好的历史行为评估参数，定义如下：

$$a_i = \frac{1}{n}\sum_{t=1}^{n}\frac{r(i,n)}{r(i,t)} - 1。 \tag{10}$$

式(10)中：$r(i,n)$为节点 i 在当前时刻的信誉值；$r(i,t)$为节点 i 在 t 时刻的信誉值。对于新节点或从未生成块的节点，其 $a_i=0$。

2.2.4　节点信誉值

通过初始信誉值、错误率、成功率和历史行为评估参数来确定节点最终的信誉值 $R(i,t)$。在信誉值的计算过程中，各项参数对总体信誉值影响的程度是不同的，本研究重点在节点错误率与成功率，以激励节点能正常运行。参考文献[12]，定义各因素所占权重如下：

$$R(i,t) = r(i,0) - \frac{2}{5}f + \frac{2}{5}s + \frac{1}{5}a_i = \frac{2f_i}{5t_{\text{false}}} - \frac{2s_i}{5t_{\text{total}}} + \frac{1}{5n}\sum_{t=1}^{n}\frac{r(i,n)}{r(i,t)} - \frac{3}{10}。$$

2.3　LN 选择与共识行为监管

各组完成 LN 预选后，由信誉评分机制计算各组预选 LN 的信誉评分。在首次分组共识前，默认节点信誉评分 $R(i,0)=0.5$。随着系统共识次数的增加，若节点信誉值 $R(i,t)<0.5$，则预选 LN 或组内 RN 被禁止参与共识过程，组内 LN 重新选择；当节点信誉值 $R(i,t)\geqslant 0.5$ 时，预选 LN 成为组内主节点，RN 允许参与共识。在选择 LN 时，若两节点的信誉值 $R(i,t)$相同，则优先选择通信时延较小的节点。

3　试验分析

3.1　试验设计

为分析 GPBFT 的性能，分别对吞吐量、共识时延和通信复杂度进行试验分析，通过与 PBFT 对比试验来分析 GPBFT 的共识效率与稳定性，环境为 Intel® CoreTM i7-7700HQ，内存为 16 GB。试验在区块链环境下进行，因此利用 Linux 系统建立基于 Hyperledger Fabric1.0 环境下的区块链系统，通过 Python3.9.2 实现了 PBFT 和 GPBFT 的运行。首先，为分析不同节点数量对 GPBFT 性能的影响，将模拟节点数分别定为 5、10、15、20 至 65(每 5 个节点取 1 个)，得到初始节点集合$\{Z\}$；然后，为模拟节点的通信时延 D，在得到初始节点集合后，利用随机算法得到节点集合的通信时延矩阵；最后，在虚拟机中开启多个进程，分别测试不同节点数量情况下 PBFT 与 GPBFT 共识时间，每个进程试验 20 次，取平均值后统计平均共识时延与吞吐量，并分析不同场景下的系统通信复杂度。

3.2　试验结果分析

3.2.1　吞吐量

吞吐量是衡量一致性算法性能的重要指标，表示单位时间内的交易数量。在吞吐量试验中，固定交易次数为 500 次，统计完成 500 次交易所用时间，然后计算两种算法的吞吐量，吞吐量对比如图 4 所示。

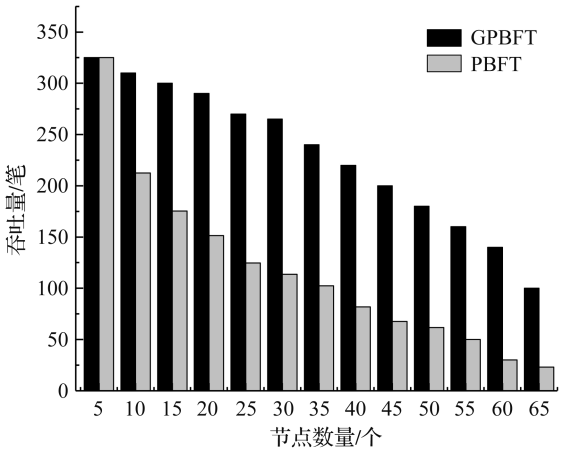

图 4　吞吐量对比

Fig. 4　Throughput comparison

由图 4 可知，在节点数量较少时，PBFT 和 GPBFT 两种共识算法的吞吐量都较高，随着节点数量的增加两者的吞吐量都在下降，但是 PBFT 的吞吐量下降速率较快，表明随着节点数量的增加，PBFT 的性能下滑严重。GPBFT 比 PBFT 平均吞吐量提高了 55.04%，在节点数为 35 和 65 时，吞吐量下降速率加快，原因是在节点数量较多情况下分组算法增加了共识组数，LN 共识数量的增加对共识效率产生了负影响，但吞吐量仍远高于 PBFT。

3.2.2　共识时延

共识时延是衡量共识算法性能的关键因素之一，时延越小表示共识算法完成一轮共识所用的时间越

短，共识效率更高。试验设置PBFT作为对照组，分别对两种共识算法进行20轮共识测试，统计共识时延并取平均值，共识时延对比如图5所示。

由图5可知，随着共识节点的增加，PBFT共识时延剧增，而GPBFT的共识时延增长较为平缓，GPBFT相比PBFT平均时延降低了57.86%，可见节点数量的增加对PBFT共识效率的影响远大于GPBFT。在分组算法将节点分组后，系统共识效率明显提高且系统更稳定。

3.2.3 通信复杂度

PBFT的通信复杂度已知为$O(Z^2)$，GPBFT的通信复杂度通过参考文献[13]计算而得：$C=mZ\log_m Z$。试验分别将节点总数Z与分组数m代入两种算法的通信复杂度中，通信复杂度对比如图6所示。在节点数量较少时，GPBFT通信复杂度与PBFT较接近，随着节点数量增加，GPBFT通信复杂度则远小于PBFT，且增幅较小，通信稳定。可见在节点数量较多时，GPBFT能大幅降低系统通信复杂度，提高系统效率。

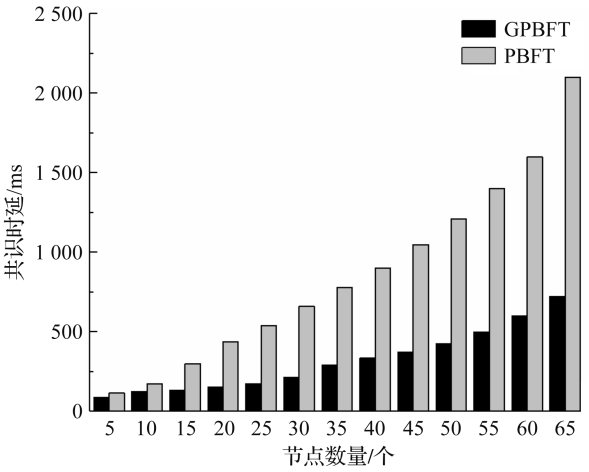

图5 共识时延对比

Fig. 5 Comparison of consensus delay

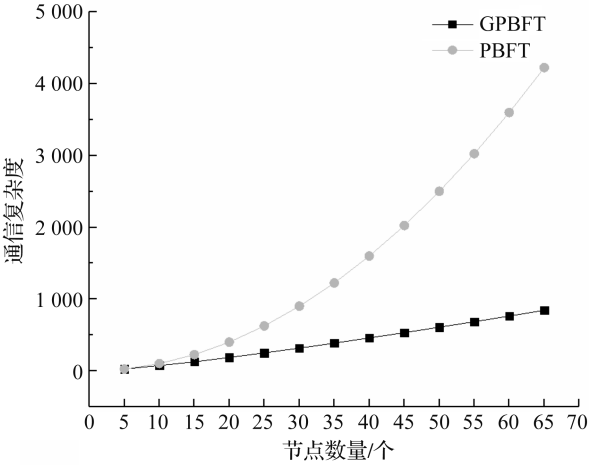

图6 通信复杂度对比

Fig. 6 Comparison of communication complexity

4 结 语

为解决PBFT效率低下的问题，本研究提出了GPBFT。首先，基于分组算法将系统中通信时延较低的节点分成一组，以降低节点间的通信时延；然后，选出各个小组中的LN并且建立节点监管机制，以减少LN的视图转换，从而降低算法的通信代价；最后，通过试验对比分析发现，GPBFT相对于PBFT在共识时延、通信复杂度等方面有较大的提升，从而有效提高了系统共识效率，解决了行业区块链系统大规模节点的需求问题。

目前，GPBFT仍处于理论试验阶段。下一步，我们将继续研究GPBFT在共识节点总数不确定的情况下，如何动态控制各组中节点数量大小，以达到更好的共识效率；并进一步研究在具体应用场景下，对不同特征进行分组后的共识效率，从而为GPBFT的实际工程应用打好理论基础。

参考文献：

[1] 袁勇，周涛，周傲英，等. 区块链技术：从数据智能到知识自动化[J]. 自动化学报，2017，43(9)：1486.

[2] CASTRO M，LISKOV B. Practical Byzantine fault tolerance[J]. ACM Transactions on Computer Systems，2002，20(4)：398.

[3] NGUYEN G T，KIM K. A survey about consensus algorithms used inblockchain[J]. Journal of Information Processing Systems，2018，14(1)：101.

[4] HE L，HOU Z. An improvement of consensus fault tolerant algorithm applied to alliance chain[C]//IEEE 9th International Conference on Electronics Information and Emergency Communication. Beijing：IEEE，2019：1.

[5] GUETA G G，ABRAHAM I，GROSSMAN S，et al. SBFT：a scalable decentralized trust infrastructure for blockchains[EB/OL]. (2018-04-04)[2022-01-28]. http://arxiv.org/pdf/1804.01626v/.pdf.

（下转第88页）