计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210500038

面向物联网的 PBFT 优化共识算法

刘 炜[1,2] 阮敏捷[1] 佘 维[1,2] 张志鸿[3] 田 钊[1]
1 郑州大学软件学院 郑州 450000
2 郑州大学互联网医疗与健康服务河南省协同创新中心 郑州 450000
3 郑州大学信息工程学院 郑州 450000
(wliu@zzu.edu.cn)

摘 要 面对大量的物联网事务,高效的共识算法是区块链技术应用于物联网的关键。物联网设备大多以无线通信的方式接入互联网,基于此,文中构建了一种大规模无线密集型网络场景。针对该场景下实用拜占庭容错算法网络通信开销过高、共识时延较长、吞吐量较低的问题,提出了一种基于聚类的实用拜占庭容错算法。首先依据位置特征对节点进行聚类,形成一个多中心层次化的网络结构;其次将共识任务进行分解,在底层和上层网络中分别进行共识,以减少共识所需的通信量;最后引入动态信誉模型评估节点的可信度,减少异常节点的参与,提高系统的安全性和可靠性。实验结果表明,基于聚类的实用拜占庭容错算法能够有效减少通信开销和共识时延,并提高吞吐量。

关键词:物联网;区块链;PBFT;聚类;动态信誉模型
中图法分类号 TP302

PBFT Optimized Consensus Algorithm for Internet of Things

LIU Wei[1,2],RUAN Min-jie[1],SHE Wei[1,2],ZHANG Zhi-hong[3] and TIAN Zhao[1]
1 School of Software,Zhengzhou University,Zhengzhou 450000,China
2 Collaborative Innovation Center of Internet Medical and Health Services,Zhengzhou University,Zhengzhou 450000,China
3 School of Information Engineering,Zhengzhou University,Zhengzhou 450000,China

Abstract Faced with a large number of IoT transactions,efficient consensus algorithm plays a key role in the application of blockchain technology into IoT. In this paper,according to the problems of long consensus time delay and low throughput in practical Byzantine fault tolerant algorithm (PBFT),we propose a practical Byzantine fault tolerant algorithm based on clustering (C-PBFT). Firstly,the nodes are clustered according to location features to form a network structure with multiple centers and layers. Then,consensus tasks are divided to conduct consensus in bottom and top network,thereby reducing the communication cost needed by consensus. Finally,credibility of dynamic credit model evaluation node is introduced to reduce the participation of abnormal nodes and increase the security and reliability of the system. Experimental results show that the C-PBFT algorithm can effectively reduce communication overhead,consensus delay and improve throughput.

Keywords Internet of things,Blockchain,PBFT,Clustering,Dynamic credit model

1 引言

近年来,各种电子设备的迅速普及和无线通信技术的快速发展极大地推动了物联网技术的进步。物联网正在为智能家居[1]、智慧城市[2]、交通管理[3]、能源管理[4]、工业自动化[5]等多个应用领域提供服务,但同时其也面临着低延迟和数据安全性的关键挑战[6],尤其在一些实时物联网系统中,要求信息及时正确地在设备间传输处理。区块链技术虽然通过不可

到稿日期:2021-05-08 返修日期:2021-07-27
基金项目:河南省高校科技创新人才支持计划(21HASTIT031);河南省重大公益专项(201300210300);河南省高等学校青年骨干教师培养计划(2019GGJS018);河南省重点研发与推广专项(212102310039,212102310554);河南省高等学校重点科研项目(20A520035);中国铁路北京局集团有限公司科技研究开发计划重大课题(2021AY03)
This work was supported by the Program for Science & Technology Innovation Talents in Universities of Henan Province(21HASTIT031),Major Public Welfare Project of Henan Province(201300210300),Training Plan for Young Backbone Teachers of Colleges and Universities in Henan Province(2019GGJS018),Scientific and Technological Research Project in Henan Province(212102310039,212102310554),Key Scientific Research Project of Colleges and Universities in Henan Province(20A520035) and Major Project of Science and Technology Research and Development Plan of China Railway Beijing Group Co. Ltd.(2021AY03).
通信作者:田钊(tianzhao@zzu.edu.cn)

篡改的分布式账本和基于加密算法的信息交换给物联网的数据安全问题带来了解决方案[7-8]，但难以实现物联网系统的实时共识，这是由于传统的区块链共识机制通常需要消耗大量的计算资源或通信资源来完成共识过程，而大多数物联网设备内存和存储容量很小且计算能力有限，难以进行密集型计算。此外，这些设备多通过低功耗无线通信接入互联网，系统的通信效率易受周围环境影响[9]。这些问题使得物联网设备难以承受高耗能的区块链共识工作。因此，面对大量的物联网事务，设计低耗能、高效率的共识机制是将区块链技术应用于物联网的关键[10-11]。

目前区块链中常用的共识机制有工作量证明机制(Proof of Work，PoW)、权益证明机制(Proof of Stack，PoS)、授权股份证明机制(Delegated Proof of Stake，DPoS)、实用拜占庭容错机制(Practical Byzantine Fault Tolerant，PBFT)，其余的共识机制大多由这 4 种机制派生而来。表 1 列出了 4 种算法的通信开销、计算开销、容错性、吞吐量、响应时间、去中心化程度和应用平台。PoW 通过算力竞争，使系统拥有 50% 的容错率，但物联网系统中存在大量低功耗设备，难以进行密集型计算，因此 PoW 并不适用于物联网系统[12]。PoS 虽然缓解了 PoW 高耗能的问题，但同时也带来了币龄累积攻击、贿赂攻击的问题。此外，由于 PoW 和 PoS 出块节点的不确定性，区块链会产生分叉，从而降低系统性能，因此同样难以适用于物联网系统。相比 PoW 和 PoS，DPoS 虽然提高了吞吐量，但去中心化程度不足，面对分布广泛的物联网设备，难以选择出具有代表性的节点。相比 PoW 等证明类算法，PBFT 算法的吞吐量可达数千 TPS，响应时间为秒级[13]，被认为是适合物联网系统的共识算法[14]。但在有 N 个节点的网络中，PBFT 完成一轮共识需要 N 个节点向全网广播消息两次。当系统中节点增多时，节点间的通信量会急剧增加，给网络带宽带来巨大压力，导致系统性能迅速下降，因此 PBFT 难以适用于大规模网络环境。

表 1 共识算法对比

Table 1 Comparison between consensus

共识	通信开销	计算开销	容错性	吞吐量	响应时间	去中心化程度	是否分叉	应用平台
PoW	低	高	1/2	约等于 7 TPS	10 min	高	是	Bitcoin
PoS	低	中等	1/2	大于等于 25 TPS	1 min	高	是	Peercoin
DPoS	低	低	1/2	大于等于 300 TPS	约等于 3 s	中等	否	EOS
PBFT	高	低	1/3	数千 TPS	秒级	高	否	Hyperledger

由于物联网设备间基于互联网和无线网络的融合进行信息交换，因此本文构建了一种大规模无线密集型网络场景，针对该场景下 PBFT 网络通信开销过高、共识时延较长、吞吐量较低的问题，提出了一种基于聚类的实用拜占庭容错算法(Practical Byzantine Fault Tolerant Algorithm Based on Clustering，C-PBFT)。该算法将共识任务进行分解，减少了通信资源的开销和共识时间的消耗，并引入动态信誉模型选择出可信节点参与共识，降低了异常节点对共识效率的影响。

本文第 2 节介绍了关于 PBFT 算法优化的相关工作；第 3 节对构建的网络场景和 C-PBFT 共识机制进行了详细描述；第 4 节给出了理论分析和实验结果；最后总结本文。

2 相关工作

PBFT 是一种基于状态机副本复制的算法，用于解决分布式系统中状态机副本一致性的问题[15]，可在失效节点不超过$(N-1)/3$ 的情况下保证共识的安全性和一致性。PBFT 算法由一致性协议、检查点协议、视图转换协议组成，算法中节点角色分为客户端、主节点和从节点 3 类。在主节点接收到客户端请求之后，执行一致性协议。若执行过程中主节点发生故障或共识超时，从节点的超时机制将触发视图转换协议，更换主节点，生成新的视图，并在新视图下继续进行共识工作以维持系统活性。PBFT 算法的工作原理如图 1 所示，共识过程主要分为 3 个阶段，在 prepare 和 commit 阶段，节点需要向全网广播消息，因此网络中会产生较大的通信量。

目前，众多学者对 PBFT 算法进行了研究，文献[16]评估了具有 PBFT 共识的 Hyperledger Fabric v0.6 的性能，结果表明，拜占庭容错一致性算法提供了合适的吞吐量，但其性能会随着设备数量的增加显著下降。在区块链网络中，默认节点间的连接以及节点自身的资源是良好的且足够支持所使用的共识算法，而物联网设备通常计算和通信资源受限，若将 PBFT 高效应用到节点数量众多且低功耗的物联网网络中，则需要对其进行优化。目前针对 PBFT 算法的优化主要从控制参与共识的节点规模和改进共识结构两个方面进行研究。

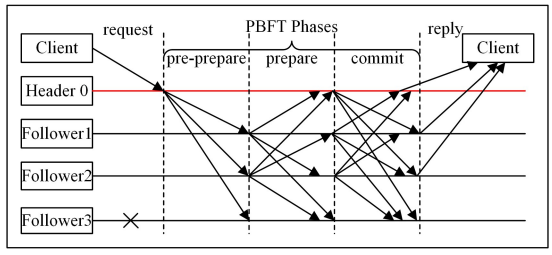


图 1 PBFT 通信过程

Fig. 1 PBFT communication process

在控制节点规模方面，文献[17]提出了一种基于位置的可扩展共识协议 G-PBFT。由于大多数物联网系统依赖于固定设备进行数据收集和处理，通常位置固定的设备相比移动设备有更强的计算能力。基于此，G-PBFT 利用固定设备的地理信息达成共识，选择位置相对固定的节点作为背书人参与 PBFT 共识，减少验证和记录事务的开销。文献[18]提出了一种适用于车联网的驾驶证明算法，它使用车辆行驶信息对矿工进行选择，使 PBFT 共识适用于公共交通网络，并引入节点服务标准来检测和消除恶意节点。文献[19]提出信用授权拜占庭共识机制，使用信誉值扩展节点属性，使信誉良好的节点有更大的机会获得出块权，同时减小异常节点参与共识的概率。文献[20]提出适用于动态网络的 EPBFT，使用可验证

随机函数(VRF)选取部分节点参与共识，并在网络状态良好的情况下执行简化的一致性协议，从而减少 PBFT 的通信量。

在改进共识结构方面，文献[21]提出了一种可扩展的多层 PBFT 共识机制以降低单层 PBFT 共识的通信量，并分析了通信量降低到最小时每一层的节点数。文献[22-23]基于分层思想，将区块链网络结构规划为树形拓扑类型，由信任度较高的节点担任树的根节点以及主节点，从底层节点自下向上进行区块的验证。文献[24]提出基于 K-medoids 的改进 PBFT 共识机制，利用 K-medoids 根据特征对参与共识的节点进行聚类，将改进后的 PBFT 算法用于聚类后的分层模型，以减少共识需要的通信次数，提高共识效率。文献[25]提出基于信任的动态分组拜占庭容错算法，该算法基于分组思想，依据信任值将节点分为若干组，将共识任务分配到各个组，从而减少网络中的通信量，提高共识效率。

上述针对 PBFT 的优化研究在控制节点规模方面使用少部分节点代替全网节点，在一定程度上减少了通信量，提高了算法的效率，但在大规模网络环境下，若仅选择部分节点参与共识，系统将有中心化的风险。在改进共识结构方面，通常基于分组或分层的思想，将共识任务划分到各组或各层，从而减少 PBFT 共识的通信量，但在分组或分层时大多未能考虑到节点间的距离，而在物联网中，节点间的距离通常会影响节点间信息传输的时延。本文基于节点间的距离对节点进行分簇，构成一个多中心的双层网络结构，在底层和上层网络中分别进行共识，同时引入动态信誉模型，依据信誉值对节点赋予不同的权限。

3　C-PBFT 算法

3.1　场景描述

本文构建了一种大规模无线密集型网络场景，它由平面上随机分布的 N 个全节点和若干事务节点组成。每个全节点通过无线信道与最近的无线接入点相连，N 个全节点之间组成完全连通的无线网络，此场景的网络拓扑如图 2 所示。

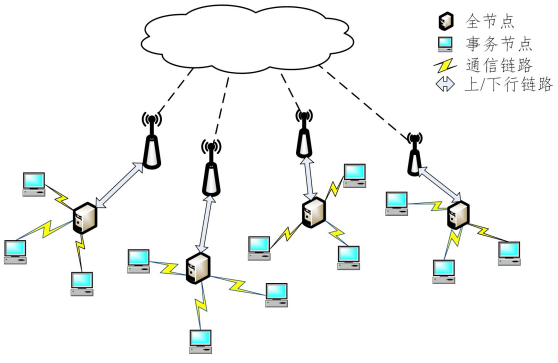


图 2　无线网络拓扑示意图

Fig. 2　Wireless network topology diagram

(1)全节点。全节点是性能较好的节点，具有区块链的全部功能，对网络中的交易进行处理，参与区块的共识过程，在本地同步完整的区块链。

(2)事务节点。事务节点可看作传统低功耗物联网设备，等同于轻量级客户端，根据系统的功能需求向全节点发送交易请求，存储部分与自身相关的数据。

事务节点根据不同的应用程序向全节点提交交易，如来自环境的感知数据、设备的位置信息等。交易由尽可能多的全节点接收，全节点将接收到的交易信息缓存到本地；出块节点打包一段时间内网络中的交易并组织成新区块，全节点基于信息交换对新区块进行验证和共享。若每轮共识结束生成一个有效区块，则全节点将区块中的交易信息从本地缓存池删除，并将新区块同步到本地，若生成一个无效区块，则丢弃该区块。一旦在区块链网络中达成共识，交易就被永久记录在区块链中。

对于上述网络场景，本文假设如下条件：

(1)系统通过配置可信的证书颁发机构(Certificate Authority，CA)感知节点的加入与退出，即支持节点数量的动态变化。

(2)使用数字签名技术来保证节点间信息安全可信地传输，拜占庭节点不能破解散列函数以及伪造签名。

(3)在无线网络覆盖范围内，故障节点不会干扰其他节点的信息传输。

3.2　算法概述

针对上述大规模密集无线网络场景，本文提出了一种基于聚类的实用拜占庭容错算法。C-PBFT 共识算法的过程分为 3 个阶段，其流程如图 3 所示。

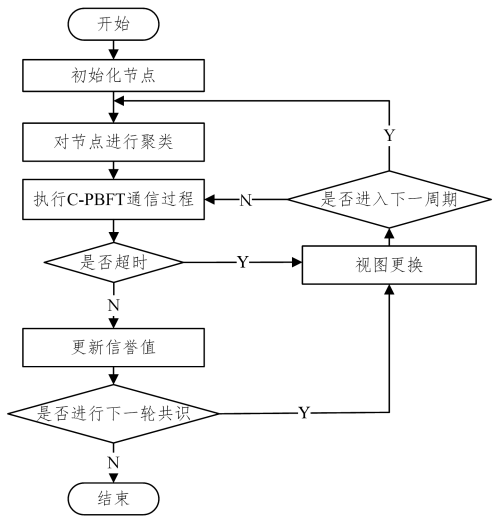


图 3　C-PBFT 共识流程

Fig. 3　C-PBFT consensus process

(1)准备阶段。完成为节点分配密钥、设置初始信誉值等初始化工作，获取节点的无线网络坐标，根据无线网络坐标对参与共识的节点聚类，将其划分为 k 个节点簇。

(2)共识执行阶段。节点分别在簇内和簇间基于信息交换对新区块进行验证和投票，将验证并投票通过的区块同步到本地。

(3)信誉更新阶段。根据节点在此次共识过程中的行为进行信誉值的更新。

一轮共识完成或共识超时则进行视图更换以进入新一轮的共识，在进行 k 轮共识之后，进入下一周期，重新对节点进行聚类。

3.3　准备阶段

在准备阶段首先完成为节点分配密钥、设置初始信誉值等一系列初始化工作，之后对节点进行聚类，形成多个节点

簇。在物联网中，可根据无线网络定位技术获取物体的空间位置信息，并将其映射到二维空间中，从而获取节点的无线网络坐标(Wireless Network Coordinates，WNC)，节点 i 的坐标表示为(x_i, y_i)，节点 i, j 之间的距离表示为 $d(i,j)$。

$$d(i,j)=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2} \tag{1}$$

节点簇的中心表示为 μ_t，即簇中所有样本特征的均值。

$$\mu_t=\frac{1}{|C_t|}\sum_{n\in C_t} n \tag{2}$$

K-means 算法在执行聚类任务时具有快速简单且易于实现的优点，本文基于 K-means 算法的思想将节点划分为 k 个非空簇。由于 PBFT 算法至少需要 4 个节点才能执行，因此需要控制每个子集群中节点的数量。此外，PBFT 算法的通信量受节点数量的影响较大，为防止某个节点簇因节点数过多而显著增大通信量，在此规定，每个集群中节点数最多为 $\lceil N/k \rceil$。而 K-means 原算法聚类时不能控制每个簇的节点数量，并对初始聚类中心和异常点敏感，为使 K-means 更适用于 PBFT，在此对其聚类过程进行改进，节点的聚类过程如算法 1 所示。

算法 1

输入：数据集 V，聚类数 k，最大迭代次数 M

输出：簇划分 $C=\{C_1, C_2, \cdots, C_k\}$

1. 根据节点的无线网络坐标，采用均匀取样技术，从数据集 V 中选择出代表节点分布特征的子集，从子集中选择信誉值排序前 k 位的节点作为初始中心节点，即$\{n_{C_1}, n_{C_2}, \cdots, n_{C_k}\}$。
2. 将 C 初始化为 $C_t=\emptyset$，$t=1,2,\cdots,k$。
3. 对于 $i=1,2,\cdots,N$，根据式(1)计算每个节点与 k 个初始中心节点的距离 $d(i, n_{C_t})$。
4. 根据距离最小原则，加入距离最近的簇，若簇内节点数量已达到最大值$\lceil N/k \rceil$，则加入距离次之的簇，重复这一步，直到加入某一簇内，此时 $C_t=C_t \cup \{n_i\}$。
5. 对于 $t=1,2,3,\cdots,k$，根据式(2)对 C_t 中所有样本点重新计算中心。若新计算的中心与原中心距离小于给定的阈值，则算法收敛；若新计算出的中心与原中心距离大于给定的阈值，则重复步骤(3)、步骤(4)，直到算法收敛。
6. 输出 $C=\{C_1, C_2, \cdots, C_k\}$。

对节点进行聚类后会形成 k 个节点簇，在此称其为子集群。各子集群中信誉值最高的节点为子集群的主节点，主节点形成主集群。子集群构成的网络称为底层网络，主集群构成的网络称为上层网络，由此构成一个多中心层次化的网络结构。聚类后的网络结构如图 4 所示。

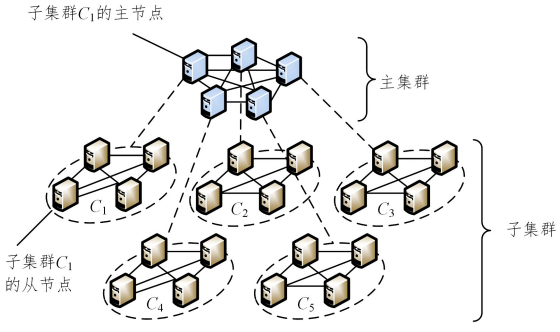


图 4 聚类后的双层网络结构

Fig. 4 Structure of double layer network after clustering

3.4 共识执行阶段

节点通过聚类被分为 k 个子集群，每个集群中信誉值最高的节点为主节点，其余节点为从节点。k 个主节点组成主集群，主集群中的节点轮流对网络中的交易进行打包，负责打包交易组织区块的节点称为出块节点。整个共识过程分为集群内共识、集群间共识和区块同步 3 个阶段。

(1)出块节点对一段时间内网络中的交易进行打包，然后将其发送给其余集群的主节点，在各集群内进行 PBFT 共识。

(2)在集群内共识完成后，由各个子集群的主节点进行 PBFT 共识，各主节点将集群间的共识结果返回给子集群内的从节点。

(3)根据返回结果，各个节点对本地数据进行更新。

为便于表述，共识过程中相关符号的说明如表 2 所列，节点间的通信过程如图 5 所示。

表 2 相关符号说明

Table 2 Explanation of related symbols

符号	定义
N	参与共识的总节点数量
k	节点聚类数
t	子集群编号
i	节点在共识网络中的编号
f	全网内拜占庭节点的最大数量
f_k	主集群内拜占庭节点的最大数量
f_t	子集群 C_t 内拜占庭节点的最大数量
v	节点当前的视图编号
h	当前区块高度
t	时间戳
b	一段时间内系统中交易打包成的区块
$D(b)$	区块 b 的数字摘要
n_i	共识网络中第 i 个节点
n_{C_t}	子集群 C_t 的主节点
$n_{C_{(t,j)}}$	子集群 C_t 中编号为 j 的从节点
$\langle M\rangle\sigma_i$	节点 n_i 对消息 M 的数字签名

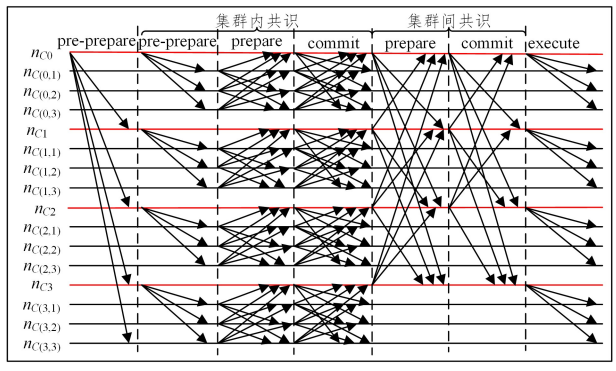


图 5 C-PBFT 通信过程

Fig. 5 C-PBFT communication process

共识过程如下：出块节点打包一段时间内网络中的交易组织成区块，并组装预准备消息发送至各个集群的主节点，消息格式为$\langle\langle G\text{-}PRE\text{-}PREPARE, v, h, t, D(b)\rangle\sigma_i, b\rangle$，其中 $G\text{-}PRE\text{-}PREPARE$ 为此消息的标识，v 为视图编号，h 为当前区块的高度，t 为时间戳，$D(b)$为区块 b 的摘要，σ_i 为此节点对消息的签名，b 为区块。

(1)集群内共识阶段。各主节点接收到出块节点的消息后，对其进行验证，若验证通过，则发起本地子集群内的 PBFT 共识，共识分为 pre-prepare，prepare，commit 这 3 步。

步骤 1　子集群主节点向集群中从节点发送〈〈*L-PREPREPARE*，v，h，t，$D(b)$〉σ_i，b〉。

步骤 2　集群内从节点接收到来自主节点的预准备消息之后对其进行验证。若验证通过则向集群内其余节点广播准备消息，消息格式为〈〈*L-PREPARE*，v，h，$D(b)$，i〉σ_i〉，同时节点会接收到来自集群内其他从节点的准备消息，若超过 $2f_t+1$ 个准备消息则通过验证，进入 commit 阶段。

步骤 3　集群内从节点向集群内其他节点发送确认消息，消息格式为〈〈*L-COMMIT*，v，h，$D(b)$，i〉σ_i〉，同时节点会接收到来自集群内其他从节点的确认消息，若收到 $2f_t+1$ 条确认消息并且通过验证，则子集群内共识阶段完成。

(2)集群间共识阶段。子集群内主节点在进行本地共识后，代表本地集群中所有节点进行集群间的共识，共识过程分为 prepare，commit 两步。

步骤 1　各个子集群的主节点向其他子集群的主节点发送准备消息，消息格式为〈〈*G-PREPARE*，v，h，$D(b)$〉σ_i，b〉。同时各主节点接收来自其他主节点的准备消息，若收到 $2f_k+1$ 个准备消息且验证通过，则进入 commit 阶段。

步骤 2　各个子集群的主节点向其他子集群的主节点发送确认消息，消息格式为〈〈*G-COMMIT*，v，h，$D(b)$，i〉σ_i〉。当各节点收到 $2f_k+1$ 个有效的确认消息时，表示集群间共识阶段完成。

(3)区块同步阶段。在完成集群间共识后，各主节点向其子集群从节点发送执行消息，各个集群中的节点同步区块，实现分布式系统中数据的最终一致。

在出块节点向主节点发送预准备消息以及主节点向从节点发送预准备消息后，收到预准备消息的节点需要验证的内容有以下几点：

1)v 与当前自身视图编号是否一致；

2)h 是否为当前区块高度；

3)$D(b)$与区块 b 的摘要是否一致；

4)区块 b 中的交易是否遵循交易格式、交易的时间戳是否有效、交易的脚本能否正确执行；

5)区块 b 中默克尔根的值是否正确、区块头中存储的哈希值与前一区块的哈希值是否一致。

若验证通过，则认为消息有效，节点执行区块，并缓存执行结果。由于在收到预准备消息后已经验证区块和交易的有效性，因此在集群内和集群间共识的 prepare 和 commit 阶段只需验证 1)－3)。

3.5　信誉更新阶段

在一轮共识完成之后，根据动态信誉模型对信誉值 R 进行更新，信誉值是表示节点可信度的一种方式，信誉值数值越大，代表节点可信度越高。对于初始共识节点，信誉值统一设置为 r，信誉取值在[0，1]之间。

C-PBFT 中，动态信誉模型包括信誉奖惩、信誉状态设置、信誉重置与恢复 3 部分。模型实现算法如算法 2 所示。首先根据节点行为对其信誉值进行增减；其次设置其信誉状态；最后对信誉值过高的节点进行信誉重置，对信誉值过低的节点进行恢复。本节将对此模型进行详细介绍。

(1)信誉奖惩指根据节点在共识过程中的行为对其信誉值进行动态增减。设 $R_i(t)$为节点 i 在第 t 轮共识的信誉值，则节点在 $t+1$ 轮共识的信誉值 $R_i(t+1)$的计算式如下：

$$R_i(t+1)=\begin{cases}R_i(t)+\alpha(1-R_i(t)), & \text{节点行为正常}\\ \beta R_i(t), & \text{节点行为异常}\\ R_i(t)\mathrm{e}^{-\lambda\Delta b}, & \text{节点离线}\\ 0, & \text{拜占庭节点}\end{cases}\tag{3}$$

若出块节点在 $t+1$ 轮打包有效区块并领导全网节点完成共识，主节点领导从节点成功完成一轮共识，从节点参与共识且最终同步结果与大多数节点一致，则 $R_i(t+1)=R_i(t)+\alpha(1-R_i(t))$。系数 $\alpha\in(0,1)$，用于控制信誉值的增长速度，其取值根据系统具体的应用需求设置。当 α 固定不变时，$R(t)$数值越大，$R_i(t+1)$的增长速度越缓慢，最终趋于 1。

若出块节点在 $t+1$ 轮打包无效区块或未能领导全网节点完成共识，主节点未能领导其从节点成功完成一轮共识，从节点最终同步结果与大多数节点不一致，则这些节点的信誉值呈线性下降。$R_i(t+1)=\beta R_i(t)$，其中 $\beta\in(0,1)$，为惩罚系数，用于控制信誉值下降的速度，具体取值根据系统的应用需求设置。

若节点长时间离线，不参与共识，则信誉值随时间逐渐衰减。λ 为衰减因子，Δb 为节点最后一次参加共识时区块高度与当前区块高度之差。

若节点向不同的节点发送不一致的消息，则视为拜占庭节点，其信誉值降为 0，禁止参与共识。

(2)节点的信誉状态 *RS* 由信誉值 R 决定，是赋予节点不同权限的依据。如表 3 所列，在此定义 4 种信誉状态，a，r，b 为状态变更的阈值，它们的取值根据网络中节点的信誉值分布和系统的安全需求设置。

表 3　节点信誉状态

Table 3　Node reputation status

R 范围	信誉状态	权限
$[a,1]$	excellent	主节点、从节点
$[r,a)$	normal	主节点、从节点
$[b,r)$	abnormal	从节点
$[0,b)$	error	禁止参与共识

(3)信誉重置与恢复指对信誉值过高的节点进行信誉重置，对信誉值过低的节点进行信誉恢复。当节点信誉值高于 m 时，在下一周期开始时将节点信誉值重置为 r，防止节点因信誉值过高而产生中心化的趋势，其中 $m\in(a,1)$，其取值根据系统具体应用需求确定。当节点信誉值低于 b 而被禁止参与共识时，在下一周期将节点信誉值恢复至 b。信誉的重置与恢复既能防止高信誉值节点权力中心化，又能保证低信誉值节点的积极性。

算法 2　动态信誉模型

输入(R_i，N_i，α，β)

输出(R_i，N_i)

```
1. for i∈N
2.    if complete this round of consensus
3.       Ri←Ri+α(1−Ri)
4.    else if Malicious behavior
5.          Ri←0
6.       else Ri←βRi
```

```
7.  if Ri<b
8.      Ni. TS←error
9.      Ri←b recover in the next cycle
10.     else if Ri<r
11.     Ni. TS←abnormal
12.     else if Ri<a
13.         Ni. TS←normal
14. else
15. Ni. TS←excellent
16. if Ri>m
17.     Ri←r reset at next cycle
18. return(Ri, Ni)
19. end for
```

4 实验分析

4.1 实验环境

为了评估 C-PBFT 的性能，使用 go 语言对 C-PBFT 和 PBFT 流程进行模拟仿真，实验环境为 Intel I7-4790 CPU 和 8GB 内存，操作系统为 64 位 Win 10，go 语言版本为 1.14.4，实验结果用 Matlab 进行处理分析。

实验中节点随机分布，不具有移动性。实验分别测试了总节点数量为 40，50，60，70，80，90，100，以及 k 为 4，7，10 时两种算法的时延和吞吐量。为了减小误差，每个实验重复 20 次，取平均值作为最终结果。本节从通信开销、共识时间延迟、吞吐量 3 个方面对 C-PBFT 和 PBFT 进行对比。

4.2 通信开销分析

由于 PBFT 是基于信息交换实现的，而信息交换会消耗通信资源，因此通信开销是一项关系到算法效率的关键指标。为验证改进后的算法是否减小了通信开销，可对比 PBFT 和 C-PBFT 算法完成一次共识所需的通信次数，PBFT 和 C-PBFT 共识过程所需通信次数如表 4 所列。

表 4 PBFT 和 C-PBFT 的通信次数

Table 4 Communication times of PBFT and C-PBFT

	PBFT	C-PBFT	
		集群内共识	集群间共识
pre-prepare	$N-1$	$N-k$	$k-1$
prepare	$(N-1)^2$	$k\left(\frac{N}{k}-1\right)^2$	$(k-1)^2$
commit	$N(N-1)$	$N\left(\frac{N}{k}-1\right)$	$k(k-1)$
总计	$2N(N-1)$	$2N\left(\frac{N}{k}-1\right)+2k(k-1)$	

假设系统中参与共识的节点总数为 N，PBFT 算法中节点在 prepare 和 commit 阶段均需在全网范围内进行广播，此时每个节点需要的通信次数为 $N-1$。令完成一轮 PBFT 共识所需的通信次数为 X，可得：

$$X=2N(N-1) \tag{4}$$

若将 N 个节点分为 k 个子集群，设每个子集群中的节点个数为 N/k，运行 C-PBFT 算法，所需通信次数的分析如下。

在集群内共识阶段，子集群中主节点向从节点发送预准备消息，此过程的通信次数为 $(N/k)-1$。从节点收到预准备消息并验证，验证结果为真之后向集群内除自己外的所有节点发送准备消息，此过程的通信次数为 $(N/k-1)^2$。从节点接收来自集群内其他节点的准备消息并验证，若验证结果为真，则向集群内除自己外的所有节点发送确认消息，此过程的通信次数为 $N/k\,(N/k-1)^2$。由于共识网络内存在 k 个子集群，记集群内共识阶段网络内的通信次数为 W，可得：

$$W=2N\left(\frac{N}{k}-1\right) \tag{5}$$

集群间共识阶段由 k 个主节点进行 PBFT 共识，根据 PBFT 共识过程所需的通信次数为 $2N(N-1)$ 可得集群间共识阶段所需的通信次数为 $2k(k-1)$。C-PBFT 共识算法完成一轮共识所需的共识次数为两阶段所需通信次数之和，记为 Y，综上可得：

$$Y=2N\left(\frac{N}{k}-1\right)+2k(k-1) \tag{6}$$

由式(4)和式(6)可得到两种算法单次共识通信次数比 Z 为：

$$Z=\frac{2N(N-1)}{2N\left(\frac{N}{k}-1\right)+2k(k-1)} \tag{7}$$

PBFT 和 C-PBFT 算法的通信次数比值曲面图如图 6 所示，当节点数 N 不变、k 等于 1 时，C-PBFT 与 PBFT 通信过程相同，Z 为 1。随着 k 值的增大，Z 随之增大，这是由于子集群数量增加，每个子集群内的节点数减少，C-PBFT 在小范围内共识，有效减少了共识所需的通信次数。当 k 增大至极值点，Z 达到最大值，之后开始下降，此时由于 k 值过大，子集群过多，在进行集群间共识时，所需要的通信次数显著增多。当 k 等于 N 时，C-PBFT 再次与 PBFT 通信过程相同，Z 再次为 1。虽然 k 的取值会影响通信次数的比值，但整体上 C-PBFT 所需的通信次数远少于 PBFT。

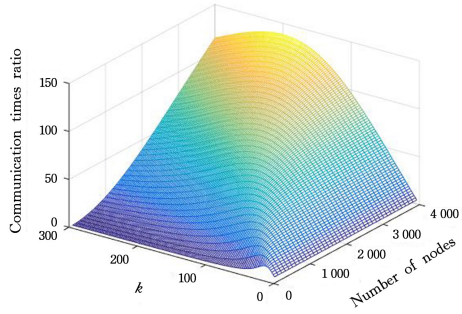


图 6 通信次数比值曲面图

Fig. 6 Comparison curve of communication time

由 PBFT 算法完成一轮共识所需的通信次数为 $2N(N-1)$，可得其时间复杂度为 $O(N^2)$。由 C-PBFT 完成一轮共识所需的通信次数为 $2N(N/k-1)+2k(k-1)$，可得其时间复杂度为 $O(m^2)$，其中 $m=\max\{N/\sqrt{k},k\}$。当 k 为 1 或 N 时，C-PBFT 的时间复杂度与 PBFT 均为 $O(N^2)$；当 $1<k<N$ 时，虽然 C-PBFT 的时间复杂度依然维持在平方级，但由于 $m<N$，因此 $O(m^2)<O(N^2)$。综上，C-PBFT 的时间复杂度小于 PBFT。

4.3 时延分析

共识时延指交易从提交到完成所经历的时间，是衡量区块链性能的重要指标，时延越短，表示交易确认的速度越快，共识的效率就越高。本文通过共识时延对比两种算法的效

率，实验中时间延迟测试的是交易从提交到客户端收到足够多回复之间的时间。

PBFT 算法时延随节点数的增加而急剧增长，这是由其 $O(N^2)$ 的时间复杂度造成的。在 prepare 和 commit 阶段，所有节点向全网节点广播消息，需要消耗大量时间。相比之下，C-PBFT 由于将全网范围内的共识划分为 k 个子集群内和子集群间的共识，大大减少了通信次数，C-PBFT 算法时延随节点数的增加而缓慢增长。时延测试结果如图 7 所示，节点数相同，$k=4,7,10$ 时，时延依次减小，结合图 6 可得，在 k 达到极值点前，k 值越大，整个共识过程所需的通信次数越少，此时 k 值越大，对应的时延越短。

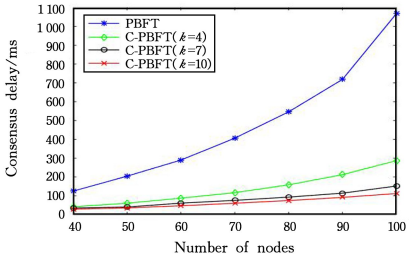


图 7　PBFT 和 C-PBFT 的时延对比

Fig. 7　Comparison of time delay between PBFT and C-PBFT

4.4　吞吐量分析

区块链系统的吞吐量指单位时间内处理的事务数量，吞吐量的大小反映了系统处理事务能力的高低，计算式为：

$$TPS=Trade_{\Delta t}/\Delta t \tag{8}$$

其中，$Trade_{\Delta t}$ 为 Δt 时间内系统处理的事务数量，Δt 为响应时间。

PBFT 算法的吞吐量受节点数影响较大。吞吐量测试结果如图 8 所示，当节点数大于 70 时，吞吐量明显下降，这是因为共识期间通信量的急剧增加给网络带宽带来了压力，增加了共识所需的时间。C-PBFT 的吞吐量随节点数增加基本稳定，这是由于将全网节点分为了若干个子集群。首先在子集群内进行小范围共识，之后在集群间的共识仅需 k 个主节点参与，因此极大地降低了通信与计算开销，使得 C-PBFT 在节点数量较多的情况下也能保持较高的吞吐量。当节点数不变，$k=4,7,10$ 时，C-PBFT 的吞吐量依次增大，结合图 6 可知，k 达到极值点前，共识所需的通信次数随 k 的增大而减小，此时 k 值越大，对应的吞吐量就越大。

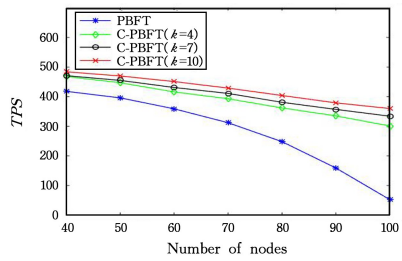


图 8　PBFT 和 C-PBFT 的吞吐量对比

Fig. 8　Throughput comparison between PBFT and C-PBFT

结束语　本文针对 PBFT 算法在大规模无线密集型网络场景下网络通信开销过高、共识时延较长、吞吐量较低的问题，提出了一种基于聚类的实用拜占庭容错共识算法。C-PBFT 依据节点空间位置特征进行聚类，将共识任务分解到各个节点簇，降低了网络中的通信开销，同时依据节点行为对节点信誉进行评估，减少了异常节点的参与。实验结果表明，C-PBFT 在通信开销、共识时延和吞吐量方面均优于 PBFT 算法。

然而，C-PBFT 算法仍存在一些不足，在聚类过程中虽然可以控制每个集群的节点数量，但集群内节点间的距离未能达到最优解。在接下来的工作中，将对共识节点的聚类特征和聚类方法进行研究，以达到更高的共识效率，同时对动态信誉模型进行完善，以提高系统的安全性。

参 考 文 献

[1] KIM U H, KIM J H. A Stabilized Feedback Episodic Memory (SF-EM) and Home Service Provision Framework for Robot and IoT Collaboration[J]. IEEE Transactions on Cybernetics, 2020, 50(5): 2110-2123.

[2] OGAWA K, KANAI K, NAKAMURA K, et al. IoT Device Virtualization for Efficient Resource Utilization in Smart City IoT Platform[C]// 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, 2019: 419-422.

[3] WU X, DUAN J, ZHONG M, et al. VNF Chain Placement for Large Scale IoT of Intelligent Transportation[J]. Sensors, 2020, 20(14): 3819.

[4] PAWAR P, VITTAL P K. Design and development of advanced smart energy management system integrated with IoT framework in smart grid environment[J]. Journal of Energy Storage, 2019, 25(Oct.): 100846. 1-100846. 13.

[5] KUMAR T, HARJULA E, EJAZ M, et al. BlockEdge: Blockchain-Edge Framework for Industrial IoT Networks[J]. IEEE Access, 2020, 8: 154166-154185.

[6] XU L D, HE W, LI S. Internet of Things in Industries: A Survey [J]. IEEE Transactions on Industrial Informatics, 2014, 10(4): 2233-2243.

[7] YU Y, DING Y, ZHAO Y, et al. LRCoin: Leakage-resilient Cryptocurrency Based on Bitcoin for Data Trading in IoT[J]. IEEE Internet of Things Journal, 2018, 6(3): 4702-4710.

[8] AMMI M, ALARABI S, BENKHELIFA E. Customized blockchain-based architecture for secure smart home for lightweight IoT[J]. Information Processing & Management, 2021, 58(3): 102482.

[9] SUN Y, ZHANG L, FENG G, et al. Blockchain-Enabled Wireless Internet of Things: Performance Analysis and Optimal Communication Node Deployment[J]. IEEE Internet of Things Journal, 2019, 6(3): 5791-5802.

[10] SI H, SUN C, LI Y, et al. IoT information sharing security mechanism based on blockchain technology[J]. Future Generation Computer Systems, 2019, 101(Dec.): 1028-1040.

[11] BISWAS S, SHARIF K, LI F, et al. PoBT: A Lightweight Consensus Algorithm for Scalable IoT Business Blockchain[J]. IEEE Internet of Things Journal, 2020, 7(3): 2343-2355.

[12] HUANG Y, ZHANG J, DUAN J, et al. Resource Allocation and Consensus on Edge Blockchain in Pervasive Edge Computing

Environments [C]// 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2019: 1476-1486.

[13] TIAN Z H, ZHAO J D. Overview of blockchain consensus mechanism for internet of things[J]. Journal of Computer Applications, 2021, 41(4): 917-929.

[14] SALIMITARI M, CHATTERJEE M, FALLAH Y. A Survey on Consensus Methods in Blockchain for Resource-constrained IoT Networks[J]. Internet of Things, 2020, 11: 1-19.

[15] CASTRO M, LISKOV B. Practical Byzantine fault tolerance [C]// Proceeding of the 3rd Symposium on Operating Systems Design and Implementation. New Orleans, USA: USENIX Association, 1999: 173-186.

[16] HAN R, GRAMOLI V, XU X. Evaluating Blockchains for IoT [C]// Ifip International Conference on New Technologies. 2018: 1-5.

[17] LAO L, DAI X, XIAO B, et al. G-PBFT: A Location-based and Scalable Consensus Protocol for IoT-Blockchain Applications [C]// 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS). IEEE, 2020: 664-673.

[18] KUDVA S, BADSHA S, SENGUPTA S, et al. Towards Secure and Practical Consensus for Blockchain based VANET[J]. Information Sciences, 2020, 545: 170-187.

[19] WANG F Y, CAI S S, LIN T C, et al. Study of Blockchains' s Consensus MechanismBased on Credit[J]. IEEE Access, 2019, 7: 10224-10231.

[20] LI Y, WANG Z, FAN J, et al. An Extensible Consensus Algorithm Based on PBFT[C]// 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC). 2019: 17-23.

[21] LI W, FENG C, ZHANG L, et al. A Scalable Multi-layer PBFT Consensus for Blockchain[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(5): 1146-1160.

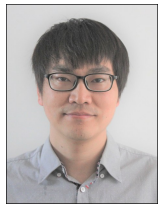
[22] BAO Z S, WANG K X, ZHANG W B. A Practical Byzantine Fault Tolerance Consensus Algorithm Based on Tree Topological Network [J]. Journal of Applied Sciences, 2020, 38(1): 34-50.

[23] DUAN J, LV X, LIU F. Hierarchical Consensus Optimization of Blockchain Based on Trust Delegation[J]. Computer Engineering, 2020, 46(10): 120-130, 136.

[24] CHEN Z H, LI Q. Improved PBFT Consensus Mechanism Based on K-medoids[J]. Computer Science, 2019, 46(12): 101-107.

[25] YU G, WU B, NIU X. Improved Blockchain Consensus Mechanism Based on PBFT Algorithm[C]// 2020 2nd International Conference on Advances in Computer Technology, Information Science and Communications (CTISC). 2020: 14-21.

LIU Wei, born in 1981, Ph.D, associate professor, is a member of China Computer Federation. His main research interests include blockchain, wireless mesh work and information security.



TIAN Zhao, born in 1985, Ph.D, lecturer. His main research interests include information security, blockchain and intelligent transportation.