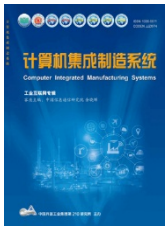

计算机集成制造系统
Computer Integrated Manufacturing Systems
ISSN 1006-5911,CN 11-5946/TP

《计算机集成制造系统》网络首发论文

题目：云制造服务场景下基于 QoS 值的改进 PBFT 算法
作者：伍星，范玉顺，郜振锋
收稿日期：2021-05-19
网络首发日期：2021-08-24
引用格式：伍星，范玉顺，郜振锋．云制造服务场景下基于 QoS 值的改进 PBFT 算法．计算机集成制造系统.
https://kns.cnki.net/kcms/detail/11.5946.TP.20210824.1146.002.html

网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

网络首发时间：2021-08-24 15:50:03
网络首发地址：https://kns.cnki.net/kcms/detail/11.5946.TP.20210824.1146.002.html

云制造服务场景下基于 QoS 值的改进 PBFT 算法

伍 星[1]，范玉顺[1+]，郜振锋[2]
(1. 清华大学 自动化系，北京信息科学与技术国家研究中心，北京 100084;
2. 深信服科技股份有限公司，广东 深圳 518071）

摘 要：在云制造服务场景下，区块链技术作为一种建立供需方信任桥梁的有效工具，得到了广泛的关注和应用。共识算法是区块链的核心技术，然而现有的共识算法，如实用拜占庭容错算法（PBFT），依然存在着消耗大以及延时高等缺点。针对这些缺点，提出了一种基于 QoS 值的改进 PBFT 算法 Q-PBFT，首先根据 QoS 值进行共识节点的筛选，然后将 PBFT 算法的三阶段协议优化为二阶段协议，从而在满足安全性的前提下提高通信效率。一系列理论和实验的分析，证明了本算法的有效性。
关键词：云制造服务；区块链；服务质量；共识算法；实用拜占庭容错算法
中图分类号：TP399 **文献标识码**：A

Improved PBFT algorithm based on QoS value in cloud manufacturing service scenario

WU Xing[1]，*FAN Yushun*[1+]，*GAO Zhenfeng*[2]
(1. Department of Automation, Tsinghua University, Beijing 100084, China;
2. Sangfor Technologies Inc., Shenzhen, 518071, China)

Abstract: In the cloud manufacturing service scenario, blockchain technology, as an effective tool to build a bridge of trust between service providers and consumers, has received extensive attention and witnessed a large number of applications. Consensus algorithm is the core technology of blockchain. However, existing consensus algorithms, such as Practical Byzantine Fault Tolerance (PBFT), still bear the disadvantages such as high consumption and high latency. An improved PBFT algorithm Q-PBFT based on QoS value is proposed in this paper. Firstly, consensus nodes are selected according to the QoS value, and then the three-phase protocol of the PBFT algorithm is optimized into a two-phase protocol, so as to improve communication effectiveness under the premise of satisfying security. A series of theoretical and experimental analyses have proved the effectiveness of this algorithm.
Keywords: Cloud manufacturing service; Blockchain; QoS; Consensus algorithm; Practical Byzantine Fault Tolerance

收稿日期：2021-05-19；修订日期：2021-08-16。Received 19 May 2021; accepted 16 Aug. 2021.
基金项目：国家重点研发计划资助项目（No.2018YFB1402500）。Foundation item：Project supported by the National Key Research and Development Program,China (No.2018YFB1402500).

0 引言

近年来，随着互联网、云计算等技术的飞速发展，以及面向服务架构（Service-Oriented Architecture, SOA）[1]、业务即服务（Business as a Service, BaaS）[2]、软件即服务(Software as a Service, SaaS)[3]等理念的兴起，制造型企业正经历着从传统制造业向云制造服务业的深刻转型，所承担的角色也从产品提供者转换为了服务提供者[4,5]。

在云制造模式下，制造型企业通过虚拟化技术将制造资源（如云服务器、软件、数据）和制造能力（如计算能力、设计能力和生产能力）封装成服务的形式发布在云制造服务平台上。该平台是一个包含了海量、异构、多层次制造服务信息的云共享资源池，用户可以从中查找、使用和集成制造服务以满足自身的制造需求，实现制造资源的按需分配和使用。

由于云制造服务平台承载着服务信息发布、服务资源匹配和服务系统管理等重要职责，因此建立一个安全的、多方可信的云制造服务平台是吸引制造企业和用户参与，促进云制造行业繁荣发展的关键。然而，传统的中心化云制造服务平台通常存在着安全防护不足的漏洞以及信任机制缺失的风险。近年来，一些学者提出了将区块链技术引入云制造服务平台的方法，试图通过利用区块链分布式、防篡改等特性来解决云制造服务平台中的安全问题和信任危机。

区块链本质上是部署在众多节点上的一种分布式数据库，为了保证这些节点上数据的正确性和一致性，共识算法发挥了重要的作用[6]。主流的共识算法包括应用于公有链的工作量证明（Proof of Work, PoW）[7]，股权权益证明（Proof of Stake, PoS）以及应用于联盟链的实用拜占庭容错（Practical Byzantine Fault Tolerance, PBFT）[8]。现有的共识算法通常都会带来显著的资源消耗，带宽开销和网络延迟，难以满足云制造服务平台对于高效性的要求。

服务质量（Quality of Service，QoS）是衡量云制造服务性能的重要指标，同样的，也可以用 QoS 值来反映区块链节点的性能。在云制造服务场景下，借鉴 PBFT 算法的基本思想，本文提出了基于 QoS 值的改进 PBFT 算法 Q-PBFT。Q-PBFT 算法由共识节点筛选算法和一致性协议改进算法两部分组成。在共识节点筛选算法中，首先基于区块链节点的 QoS 值筛选出具有高 QoS 值的节点构成候选共识集群，然后提出了集群的动态调整和轮换策略，在共识过程中识别拜占庭节点，移除拜占庭节点并允许高 QoS 值新节点的接入，以此提高系统的动态性和健壮性。在一致性协议改进算法中，将传统 PBFT 算法中的三阶段广播协议优化为二阶段协议，通过减少通信复杂度提高共识效率，并设计了广播消息的数据格式，使得算法能够识别拜占庭节点。Q-PBFT 算法天然地契合云制造服务场景，本文也通过理论分析和实验分析表明 Q-PBFT 算法能够在满足安全性的前提下，减少带宽消耗、降低共识时延，从而提升基于区块链的云制造服务平台的性能。

1 背景介绍

本章将对本研究所涉及的研究背景、重要概念给出详细描述，以便于后续讨论。

1.1 云制造服务平台和区块链

一个典型的云制造服务平台主要包含了三类实体，分别为制造服务提供者、制造服务以及服务使用者。其中制造服务提供者将制造资源和能力封装成制造服务的形式发布在平台。每个制造服务包含了一些服务基本信息，如服务名称、服务标签、服务描述、服务发布时间、服务价格、服务质量等。服务使用者可以根据自身的制造需求，从平台上选择符合需求的一个或多个服务。云制造服务平台通常能够提供用户需求和服务资源的自动匹配功能，匹配完成后，用户可以按需购买。服务的交易信息，如服务购买数量、服务交易时间等将记录在云制造服务平台上[9]。

近年来，一些学者通过通过引入区块链技术，建立了基于区块链的云制造服务平台，Wang 等建立了云制造服务交易信息的多链数据存储架构，通过定制智能合约实现了制造服务的安全匹配与可信交易[10]。Xu 等通过构建分层扩展式区块链，扩展了不同资源能力的制造型企业在云服务平台中的高效区块操作，同时确保了端到端的数据安全[11]。Gao 等在基于区块链的云服务平台中，借鉴比特币的概念，设计了自定义通证“Stoken”来担当价值传递的媒介，并围绕“Stoken”设计了一系列激励机制以促进云服务平台的繁荣[12]。

利用区块链搭建云制造服务平台，就可以通过区块链来记录服务的基本信息和服务的交易信息，通过区块链独特的分布式链式存储结构和密码学技术保证这些数据的真实性。数据一旦上链就不可删除和修改，可以避免因为恶意攻击而造成的数据篡改，有效提高云制造服务平台的信誉。同时区块链还可以保证交易的全过程可追溯，当服务提供者和使用者之间出现交易纠纷时，这些可溯源的信息就将成为有力的证据。

区块链根据应用场景的不同可以分为公有链和联盟链[13]。在公有链中，任何节点都可以自由加入和退出网络，节点完全不可信，数据的可信性由复杂的共识算法如 PoW 来保证。在联盟链中，节点背后都有与之对应的实体组织机构，由这些组织机构组成联盟，共同维护区块链平台的安全运转。云制造服务业的场景与联盟链的应用场景高度吻合，云制造服务业中的各个企业，可以构成一个云制造服务联盟。联盟中的制造型企业在提供云制造服务的同时，以节点的形式接入到区块链网络中，由企业为节点背书，共同维护云制造服务平台的安全运转。本文所要优化的 PBFT 算法也正是一种广泛应用于联盟链中的共识算法。

1.2 共识节点和记账节点

区块链节点根据其在网络中所承担的职责不同，主要分为两类：共识节点和记账节点[14]。共识节点根据共识协议达成的一致性结果，产生数据区块，并保证区块信息的一致性和正确性。记账节点进一步验证产生的区块信息的合法性，并将通过验证的区块记录在本地账本。区块能否有效产生是决定区块链网络能否健康运行的关键，因此云制造服务平台需要选择实力较强、信誉良好的企业参与维护共识节点，联盟中的其它企业负责

维护记账节点。共识节点和记账节点之间在一定的条件下可以相互转换。由于共识节点通常需要根据已有区块链内容验证交易信息的合法性，因此本文提到的共识节点同时也承担着记账功能。

1.3 PBFT 算法

PBFT 算法[8]主要包含了三种角色：客户端、主节点、从节点。客户端是消息的发送方，需要记录在云制造服务平台中的所有信息，比如发布的服务信息、服务交易信息等统称为消息，由客户端发送到共识节点集群请求共识。主节点和从节点都是共识节点，相互配合完成共识过程。传统 PBFT 算法的一致性协议（共识过程）主要包括以下三个通信阶段：预准备阶段、准备阶段和确认阶段：

（1）预准备阶段：主节点接收来自客户端发来的消息请求 m，为其分配编号 r，生成预准备消息，然后再广播到所有从节点，广播的消息格式为 $\langle PREPREPARE, v, r, m, d(m) \rangle$，其中 $d(m)$ 是消息的摘要（如哈希编码），v 是视图编号。

（2）准备阶段：节点检查预准备消息的内容，如果同意，则进入准备阶段并向其他所有节点广播准备消息，准备消息格式为 $\langle PREPARE, v, r, m, d(m), n_i \rangle$，其中 n_i 表示第 i 个区块链节点。

（3）确认阶段：当节点接收到 $2f+1$ 个准备消息后，f 表示拜占庭节点的个数，首先检查消息的正确，然后进入确认阶段：节点向其他所有节点广播确认消息，确认消息格式为 $\langle COMMIT, v, r, m, d(m), n_i \rangle$。当节点收到 $2f+1$ 个确认消息后，生成区块，向客户端返回生成区块的信息。

PBFT 算法能够解决拜占庭将军问题，在即使存在一定数量拜占庭节点（恶意节点）的情况下，依然能够在分布式节点间达成一致性共识。然而，该算法需要所有节点都参与到共识过程中，并且其共识流程依赖于三阶段协议这种高强度网络通信协议，因此共识集群的规模成为了限制 PBFT 算法性能的主要因素。当系统中共识节点的数量增加时，算法的通信代价（带宽消耗和共识时延）都会大大增加。

针对此问题，研究学者提出了很多 PBFT 改进算法，通过减少共识节点的规模来提升 PBFT 算法的效率。Zhu 等提出了基于距离的 DS-PBFT 算法，根据地理距离进行分组共识，在缩小共识节点规模的基础上缩短共识节点间的通信距离[15]。Wang 等将投票证明（Proof of Vote, PoV）与 PBFT 结合，将网络中的节点划分成具有不同职责的节点，由投票节点投票产生共识节点，并监督共识节点诚实可靠地生产数据区块[16]。Lao 等提出了应用于物联网环境的 G-PBFT 算法，选择若干个忠实可靠的、强健的物联网设备作为共识节点，利用物联网设备防止 Sybil 攻击[17]。

以上研究，分别从不同角度减少了共识集群的规模，在一定程度上提升了 PBFT 算法的共识效率。但是将这些共识算法应用在基于区块链的云制造服务平台存在着局限性，比如在云环境下，难以根据地理距离进行分

组共识；同时上述算法虽然在能耗、吞吐量、延时等方面有所改进，但仍然存在着进一步的优化空间。本文正是在云制造服务场景下，针对上述算法所作的进一步优化。

2 QoS 值计算

服务质量（Quality of Service, QoS）指标通常被用来衡量云制造服务的性能，QoS 指标包含了多个维度的信息，比如服务的可用性、可靠性、吞吐量、响应延时、信誉等。类似的，可以用 QoS 值来衡量区块链节点的性能。云服务的 QoS 值通常体现了发布该服务的云制造企业的综合实力，当服务具有低响应延时和高吞吐量时，企业部署的区块链节点通常也具有较高的信息处理能力；当云服务具有较高的信誉度，区块链节点也不容易成为拜占庭节点。因此，当一个云制造企业发布云服务并接入到云制造平台时，就可以用其发布服务的 QoS 值来表示其所承担区块链节点的 QoS 值。具有高 QoS 值的区块链节点具有较高的稳定性、信息处理能力和可信度。

由于 QoS 指标包含了多个维度的信息，有的指标是正指标（值越高越好，如吞吐量、信誉等），有的指标是负指标（值越低越好，如响应延时等）。因此，基于 QoS 值筛选区块链共识节点，就需要融合多个维度的 QoS 信息，建立一个统一的归一化的 QoS 值计算函数，其计算方法如下：

$$U(n_i)=\sum_{j=1}^{K}\left(I_a(q_j)\frac{q_j(n_i)-Q_j^{\min}}{Q_j^{\max}-Q_j^{\min}}+I_b(q_j)\frac{Q_j^{\max}-q_j(n_i)}{Q_j^{\max}-Q_j^{\min}}\right)\times w_j, i\in[1,N]$$
$$\sum_{j=1}^{K}w_j=1 \tag{1}$$

式中 n_i 表示第 i 个区块链节点，区块链节点总数为 N； w_j 表示第 j 项指标的权重值，一共有 K 项指标； $q_j(n_i)$ 表示 n_i 第 j 个维度的 QoS 值；$Q_j^{\max}$ 与 $Q_j^{\min}$ 分别表示所有节点在第 j 个维度的最大 QoS 值与最小 QoS 值； $I_a(q_j)$ 与 $I_b(q_j)$ 为示性函数，当 q_j 为正指标时， $I_a(q_j)=1$， $I_b(q_j)=0$，当 q_j 为负指标时， $I_a(q_j)=0$， $I_b(q_j)=1$。

除了上述提到的常规 QoS 指标，本文将在 QoS 评价体系中新增一个维度的指标：共识贡献度 λ，用来衡量一个区块链节点对消息达成共识这一过程所作出的贡献，其计算方法如下：

$$\lambda=\alpha_1(1-e^{-\beta_1 N_T})-\alpha_2(e^{\beta_2 N_F}-1) \tag{2}$$

其中 $\lambda_1=\alpha_1(1-e^{-\beta_1 N_T})$ 表示正贡献度部分： N_T 为节点做出正确共识判断的数量， α_1 为奖励因子， β_1 控制正贡献度曲线的增长速度； $\lambda_2=\alpha_2(e^{\beta_2 N_F}-1)$ 表示负贡献度部分： N_F 为错误共识判断的数量， a_2 为惩罚因子， β_2 控制负贡献度曲线的增长速度。PBFT 算法本质上是一种少数服从多数的共识算法，在对一个消息的共识过程中，可以认为多数节点的判断即为正确判断。令 $\alpha_1=1$， $a_2=1$ 两个部分的共识贡献度曲线分别如图 1、图 2 所示。

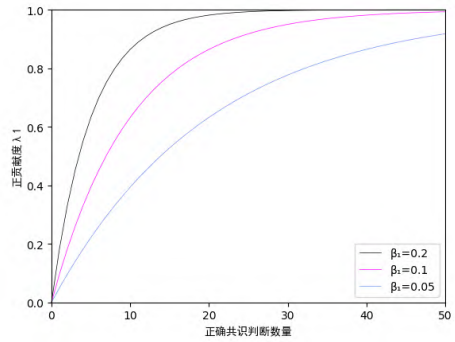

图 1 正共识贡献度曲线

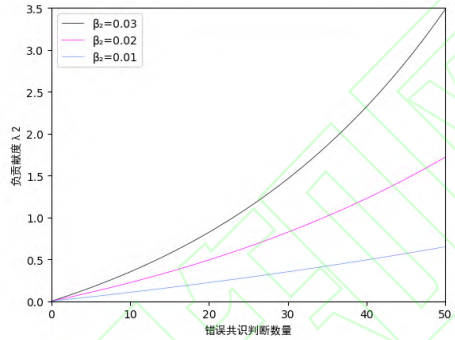

图 2 负共识贡献度曲线

从图 1 可以看到，节点正贡献度的积累速度会随正确共识判断数量的增加而减缓，通过此方法降低了新节点加入的门槛；反过来，从图 2 可以看到，节点受到的惩罚会随错误共识判断数量的增加而加重，因此，多次作恶的节点再无可能得到信任。从图 1、图 2 也可以看到，通过增大 β_1、β_2，能够增大贡献度曲线的变化趋势。

共识贡献度是一种正指标，可以代入公式（1）与常规共识指标融合，用于计算统一的 QoS 值。融合后区块链节点 n_i 的 QoS 值计算公式为：

$$U(n_i)=\sum_{j=1}^{K}\left(I_a(q_j)\frac{q_j(n_i)-Q_j^{\min}}{Q_j^{\max}-Q_j^{\min}}+I_b(q_j)\frac{Q_j^{\max}-q_j(n_i)}{Q_j^{\max}-Q_j^{\min}}\right)\times w_j'+w_\lambda\cdot\frac{\lambda(n_i)-\lambda^{\min}}{\lambda^{\max}-\lambda^{\min}}$$
$$\sum_{j=1}^{K}w_j'+w_\lambda=1 \qquad (3)$$

式中，w_λ 是共识贡献度指标的权重，$\lambda(n_i)$ 是区块链节点 n_i 的共识贡献度，$\lambda^{\max}$ 与 $\lambda^{\min}$ 分别代表所有节点的共识贡献度中的最大值与最小值。由于增加了新的指标，为了保证权重的归一化，需要对常规 QoS 指标原始的权重值进行调整。融合后常规 QoS 指标新权重 w_j' 的计算公式如下：

$$w_j'=w_j\cdot(1-w_\lambda),\ j\in[1,K] \qquad (4)$$

在基于区块链的云制造服务系统中，企业及其所承担的区块链节点的 QoS 值高低是消费者是否选择该企业服务

的重要依据，因此这些信息需要以公开透明的形式存储在区块链平台上以保证其真实性。利用这些信息，就可以通过调用智能合约来获取 $Q_j^{\max}$ 、 $Q_j^{\min}$ 、 $\lambda^{\max}$ 、 $\lambda^{\min}$ ，从而完成公式（3）的计算。

3 Q-PBFT 共识算法

在传统 PBFT 共识算法的基础上，Q-PBFT 算法主要做了以下两个方面的工作：基于 QoS 值的共识节点筛选和一致性协议改进。本节将从这两个方面分别进行介绍。值得注意的是，共识节点筛选算法和一致性协议改进算法这两个模块的功能是相辅相成不可割裂的。共识节点筛选算法关注的问题是：如何基于已经确定的 QoS 值，在每个阶段建立共识集群。而一致性协议改进算法，则是基于已经建立的共识集群，执行共识算法，解决区块生成、QoS 值更新等问题。更新的 QoS 值也将反过来成为下阶段共识节点筛选的依据。

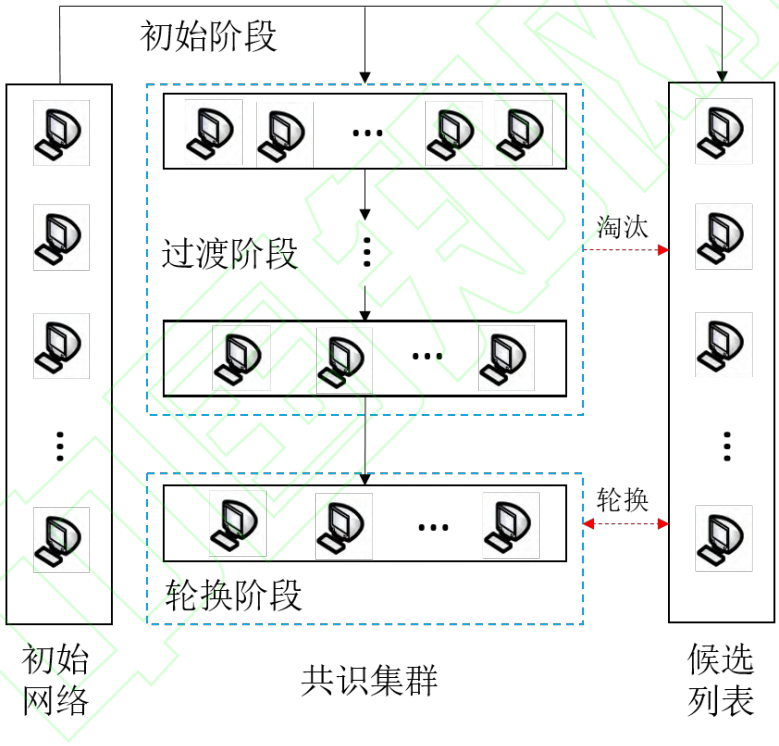

图 3 Q-PBFT 共识节点筛选算法

3.1 共识节点筛选

假设区块链节点总数为 N，拜占庭节点数量为 f， $N>3f+1$。通过筛选后，共识节点集群的规模为 C。基于 QoS 值的共识节点筛选算法主要包含了三个阶段：初始阶段、淘汰阶段、轮换阶段。算法框架如图 3 所示。

（1）初始阶段

在实际的网络中，如果直接从 N 个区块链节点中选择 C 个节点作为共识节点，即使 $N>3f+1$，也难以保证 $C>3f'+1$, f' 为最终共识集群中拜占庭节点的数量。即使节点具有较高 QoS 值，也难以完全保证其在共识

过程中的表现。因此，应该避免一刀切的筛选方式，本文提出的正是一种分段筛选方案：在初始阶段，对 N 个区块链节点按 QoS 值的大小进行降序排序，取前 P 个节点构成候选共识集群，$C \le P \le N$。P 的大小取决于云服务制造联盟中企业的信誉程度，若整体信誉度较高，则可以取一个较小的 P 值。

（2）过渡阶段

在该阶段，对长度为 P 的候选共识集群进行进一步的筛选，将共识集群的规模缩小到目标长度 C。本文采取了一种平稳的过渡策略：定义共识周期 T，在一个周期中，共识集群可以完成 m 轮共识。一个周期结束后，重新计算集群中所有节点的 QoS 值，淘汰综合表现最差的 d 个共识节点，因此从过渡阶段开始到结束，系统完成的共识轮数：

$$M = \left\lceil \frac{P-C}{d} \right\rceil \times m \tag{5}$$

式中 $\lceil \bullet \rceil$ 表示向上取整计算。

（3）轮换阶段

当过渡过程稳定后，共识算法进入第三个阶段——轮换阶段，并将长期维持在该阶段。在轮换阶段，所有非共识节点都将被加入候选列表。轮换阶段就是在共识集群和候选列表之间建立动态平衡：在一个共识周期 T 后，淘汰共识表现最差的 d 个共识节点，候选列表中 QoS 值最高的 d 个节点将“晋级”到共识集群。通过轮换策略，可以提高网络的动态性和健壮性，当一个正常的节点成为拜占庭节点后，将会在轮换阶段被发现并被及时移出网络（自动进入候选列表）；一个新的具有高 QoS 值的节点也有机会成为共识节点。

共识节点筛选算法的三个阶段都是基于 QoS 值对节点进行筛选。该 QoS 值是综合考虑了吞吐量、响应延时、共识贡献度等多项指标的一个统一加权值。由于每个阶段所处的时期不同以及筛选目标不同，因此每个阶段计算 QoS 值时对各种指标应当考虑不同的加权系数。在初始阶段，由于共识过程还未开展，无法考虑共识贡献度，因此完全基于常规 QoS 指标对节点进行初步筛选。在过渡阶段，候选共识集群都是具有较高 QoS 值的节点，因此主要考察共识贡献度，基于共识表现淘汰节点。在轮换阶段，网络进入长期稳定运行状态，共识集群中的节点存在成为故障节点（常规 QoS 值降低）和拜占庭节点（共识贡献度降低）的风险，因此对于共识集群中的节点，应该综合考虑常规 QoS 指标和共识贡献度；而对于候选列表中的节点，由于其未参与现阶段共识过程，因此主要考察常规 QoS 指标，部分候选节点的历史共识表现也将成为参考。各种指标的权重在不同阶段中的权重系数如表 1 所示，其中常规 QoS 指标权重为吞吐量、响应延时等多种常规指标权重之和。

表 1　QoS 值计算权重系数表

共识阶段	常规 QoS 指标	共识贡献度
初始阶段	1	0
过渡阶段	0.2	0.8

轮换阶段（共识集群）	0.5	0.5
轮换阶段（候选列表）	0.8	0.2

3.2 一致性协议改进

传统 PBFT 算法主节点是通过如下公式确定的：

$$p = v \bmod N \tag{6}$$

第 p 个节点会被选为主节点，这是一种轮流坐庄的选举方式，选举结果具有较大的随意性，因此选举产生的主节点很有可能是恶意节点，具有一定的安全隐患。若主节点是恶意节点，则会触发视图切换协议，频繁的视图切换会增加系统开销。本文提出的一致性协议改进算法首先优化主节点选举方案，不再按照编号选择主节点，而是依照 QoS 值选择主节点。在基于 QoS 值筛选出的共识节点集群中，进一步选择 QoS 值最大的节点作为主节点，其他节点均为副节点。

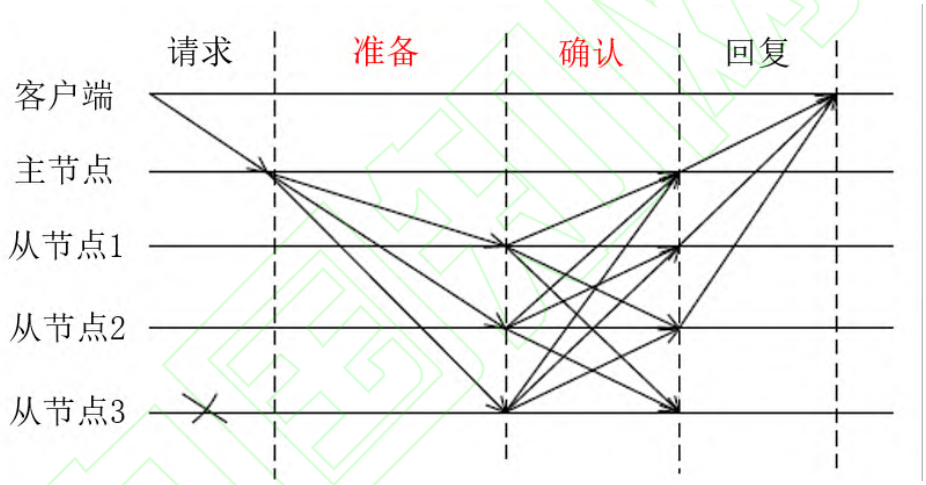

图 4 Q-PBFT 一致性协议算法流程示意图

传统的 PBFT 算法通过三阶段协议来保证同一个视图 v、同一个编号 r 下消息 m 的一致性和正确性。假设 m 为正确消息，m' 为错误消息，若一个节点是拜占庭节点，一方面该节点可能将正确消息篡改为对自己有利的错误消息，我们称这类恶意攻击为正确性攻击；另一方面，该节点可能发送不同的消息 m，m' 给不同的节点，我们称这类恶意攻击为一致性攻击。PBFT 算法可以确保网络中非拜占庭节点处理的是消息<m,v,n>，而非<m’,v,n>。在云制造服务交易平台中，部分企业可能会发起正确性攻击以获得不正当利益，比如篡改交易消息，将消费者对服务的差评修改为好评；或伪造交易消息，生成虚假的订单等。但企业却难以从一致性攻击中获得利益。因此在云制造服务场景下，本文假设恶意节点只会发起正确性攻击，而不会发起一致性攻击。基于该前提，本文将对传统 PBFT 算法中的三阶段协议进行简化，提出二阶段协议，以优化共识流程。优化后的一致性协议算法流程示意图如图 4 所示。设某一共识阶段参与共识的区块链节点数为 N'，在区块链场景下算法主要流程如下：

请求阶段：主节点接收来自客户端的消息请求 m，检查消息的合法性，检查内容包括：消息的格式是否符合系

统要求，消息的内容是否与区块链已有内容冲突等。若检查通过，则为消息编号，排序，并将该消息加入到待共识消息列表 M_0， M_0 的内容为 $\{j, m_j, d(m_j)\}$。当列表长度或等待时间达到设定阈值，进入准备阶段。

准备阶段：主节点为消息列表 M_0 生成准备消息，准备消息的格式为 $\langle PREPARE, v, h, M_0, D(M_0), Sign(n_0)\rangle$，其中 h 为待生成区块的高度， $D(M_0)$ 为消息列表 M_0 的摘要（梅克尔树哈希值）， $Sign(n_0)$ 为主节点的签名。将该准备消息广播到所有节点，进入确认阶段。

确认阶段：副节点 n_i 接收到准备消息，检查准备消息的合法性，检查内容包括：数字签名的合法性，视图编号 v，待生成区块高度 h，消息列表 M_0 及其摘要，并逐条检查消息的合法性，将每条消息及检查结果添加到消息列表 M_1， M_1 的内容为 $\langle M_{11}, M_{12}, n_i\rangle$，其中 M_{11} 为检查通过的消息列表， M_{12} 为检查不通过的消息列表。生成确认消息，确认消息的格式为 $\langle COMMIT, v, h, M_1, D(M_1), Sign(n_i)\rangle$， $Sign(n_i)$ 为副本节点 n_i 的签名。副节点将确认消息广播到所有节点。

回复阶段：当某节点 n_i 接收到 $2f+1$ 个来自其他节点的确认消息后，首先检查确认消息的合法性，然后对确认消息中的消息列表 M_1 中的消息进行逐条统计，将统计结果添加到消息列表 M_2， M_2 的格式为 $\{j, m_j, d(m_j), M_{2T}, M_{2F}\}$。其中 M_{2T} 是同意消息 m_j 的节点的集合， M_{2F} 是不同意消息 m_j 的节点的集合。若 M_{2T} 的长度大于 M_{2F}，则将消息 m_j 写进区块，最终生成的区块为 $block_h$。该节点 n_i 生成回复消息，回复消息的格式为 $\langle REPLY, v, h, block_h, D(block_h), M_2, D(M_2), Sign(n_i)\rangle$。所有副节点将回复消息反馈给客户端，当客户端收到至少 $f+1$ 个包含相同区块信息（ $block_h$ 、 $D(block_h)$ ）的回复消息后，即确定 $block_h$ 为 h 高度上的区块，包含该区块信息的回复为有效回复。客户端在网络中广播区块信息，所有记账节点将最新的区块加入到区块链中。同时，客户端根据收到的有效回复中的信息 M_2，评判各个节点在共识过程中的表现，如是否做出假阳性共识判断（对不合法的消息检查通过）或假阴性共识判断（对合法的消息检查不通过），根据共识表现更新节点的共识贡献度，从而更新节点的 QoS 值。

上述一致性协议算法流程的主要执行步骤如图 5 所示。

Algorithm 1 一致性协议算法流程执行图

请求阶段：
while 消息列表长度 M_0 或等待时间小于设定阈值 **do**
  主节点 n_0 接收来自客户端的请求消息 m，检查消息合法性。若检查通过，为消息编号，排序，并加入到待共识消息列表 M_0。
end while
准备阶段：
主节点为 M_0 生成准备消息 $< PREPARE, v, h, M_0, D(M_0), Sign(n_0) >$。
主节点将准备消息广播到参与共识算法的节点。
确认阶段：
for $0 < i < N'$ **do**
  副节点 n_i 接收到准备消息，检查准备消息的合法性
  if 检查通过 **then**
    对 M_0 中的消息逐条检查，生成检查完毕的消息列表 M_1。
    生成确认消息 $< COMMIT, v, h, M_1, D(M_1), Sign(n_i) >$。
    将确认消息广播到所有节点。
  end if
end for
回复阶段：
for $0 \leq i < N'$ **do**
  if 节点 n_i 接收到 $2f+1$ 个来自其他节点的确认消息并检查通过 **then**
    逐条检查 M_1 中的所有消息，生成统计消息列表 M_2，生成区块 $block_h$。
    生成回复消息 $< REPLY, v, h, block_h, D(block_h), M_2, D(M_2), Sign(n_i) >$。
  end if
end for
if 客户端收到至少 $f+1$ 个包含相同区块 $block_h$ 的回复消息 **then**
  确定 $block_h$ 为 h 高度上的区块。
  评判各个节点的共识表现，更新节点的 QoS 值。
end if

图 5 一致性协议算法流程执行图

4 分析

本节将从理论和实验两个方面对 Q-PBFT 算法的性能进行分析，PBFT 算法和 S-PBFT 算法将作为对照方法纳入比较。其中 S-PBFT 算法是指从 N 个区块链节点中选择 C 个节点进行共识的一类方法的统称。

4.1 理论分析

4.1.1 容错性分析

对于 PBFT 算法，假设网络中有 f 个拜占庭节点，则网络需要从 $N-f$ 个节点的回复中做出共识判断。由于网络延迟的原因，拜占庭节点的回复可能先于诚实节点到达，则在最极端的情况下，$N-f$ 个回复包含了 f 个拜占庭节点的回复，因此需要保证诚实节点的回复个数大于拜占庭节点的回复个数，即 $N-f-f>f$。所以 PBFT 算法最多能容纳 $f=\lfloor (N-1)/3 \rfloor$ 个拜占庭节点，$\lfloor \bullet \rfloor$ 表示向下取整计算。

Q-PBFT 算法、S-PBFT 算法本质上与 PBFT 算法一样，都是基于少数服从多数原则的算法。类似的分析，在最极端的情况下，需要保证 $C>3f'$，f' 为筛选后共识集群中拜占庭节点的个数。在初始节点集合满足 $N>3f$

的前提下，S-PBFT 算法在筛选上由于条件限制，难以保证筛选后的集合仍然满足容错性的要求。而 Q-PBFT 算法基于 QoS 值筛选共识节点，节点的安全性有一定保障，并且通过淘汰阶段和轮换阶段淘汰拜占庭节点，因此可以有效地保证筛选后集合的容错性。

4.1.2 安全性分析

本小节讨论在共识节点是拜占庭节点的前提下，系统的安全性，以及系统能否识别出拜占庭节点。

若主节点是拜占庭节点，则其可能将错误的消息 m' 加入消息列表，诚实的副节点 n_i 在确认阶段通过检查消息的合法性发现了错误消息 m',并将该检查结果加入到了 M_{12} 列表。副节点 n_i 收到 $2f+1$ 个来自其他节点的确认消息，发现至少在 $f+1$ 个确认消息中，错误消息 m' 在 M_{12} 列表，因此 n_i 会生成不包含消息 m' 的 $block_h$。回复阶段，客户端收到了 $f+1$ 个不包含 m' 的 $block_h$，即有 $f+1$ 个节点对消息 m' 作出了检查不通过的判断，便可以判断是主节点作恶。这时就会触发视图切换协议，立即更新节点的共识贡献度并重新选举主节点。

若副节点 n_i 是拜占庭节点，则其可能在确认阶段，对正确的消息 m 做出错误的共识（如对消息 m 检查不通过，或替换成伪造的消息 m' 并检查通过）。由于错误的共识最多有 f 个，根据诚实的 $f+1$ 个节点的共识判断情况，错误的共识结果不会被加入到区块中，并且错误的共识判断将会被记录，因此可以判断 n_i 是拜占庭节点。副节点 n_i 也可能直接将消息 m' 加入到区块中，该错误的区块在回复阶段会被客户端发现，并且包含该错误区块的回复消息也将作为无效消息处理。在一个共识周期完成后，拜占庭节点由于做出了错误的共识判断就可能会被轮换。

综上分析，无论主节点作恶还是副节点作恶，本文提出的 Q-PBFT 算法中的一致性协议改进算法都能够保证系统的安全运行，并且可以及时识别共识集群中出现的拜占庭节点，通过淘汰和轮换策略进一步提高系统的安全性。

4.1.3 通信次数分析

在传统的 PBFT 算法中，所有节点都将参与共识过程，并且在单次共识过程中，需要完成一次单节点全网广播和两次全节点全网广播，因此单次共识过程总通信次数如下：

$$count_{PBFT} = N-1+(N-1)(N-1)+N(N-1) = 2N^2-2N \tag{7}$$

S-PBFT 算法减少了共识集群的规模，因此单次共识过程总通信次数为：

$$count_{S-PBFT} = 2C^2-2C \tag{8}$$

本文提出的 Q-PBFT 算法，基于 QoS 值的筛选，通过多阶段策略动态地减小共识集群的规模；并且对一致性协议过程进行了简化，在单次共识中，减少了一次全网广播。设共识过程某一阶段的共识节点数目为 N'，则单次共识过程总通信次数如下：

$$count_{Q-PBFT} = N'-1+(N'-1)(N'-1) = N'^2 - N' \tag{9}$$

当共识过程进入轮换稳定阶段后，设 h' 为当前区块高度，则网络中单次共识的平均通信次数为：

$$\overline{count_{Q-PBFT}} = (1-\frac{m\left\lfloor \frac{P-C}{d} \right\rfloor}{h'})(C^2-C) + \frac{\sum_{k=0}^{\left\lfloor \frac{P-C}{d} \right\rfloor}\left[(P-dk)^2 - (P-dk)\right]}{h'} \tag{10}$$

随着共识过程的进行，$\overline{count_{Q-PBFT}}$ 将会无限趋近于 C^2-C，该值远小于 $count_{PBFT}$ 和 $count_{S-PBFT}$，因此 Q-PBFT 算法是三者之中通信代价最低的算法。

4.2 实验分析

4.2.1 数据集及实验环境介绍

WSDream[18]是做 Web 服务研究常用的数据集，其记录了 2699 个服务提供商向 339 个用户提供的 5825 个服务的 QoS 值，比如传输延迟等。本文首先对该数据集进行了清洗（如删除负值和异常较大值），然后统计了服务提供商提供服务的平均传输延迟。一个服务提供商代表一个云制造企业，本文用平均传输延迟作为云制造企业的常规 QoS 值标，该 QoS 值将作为筛选共识节点的标准。

本文配置了 5 台基于 i7-6700 处理器、32GB 内存、ubuntu18.04 操作系统的服务器，通过 python 对 PBFT、Q-PBFT 等算法进行了数学仿真。受限于实验环境，在不影响本文实验结论的前提下，本文从 2699 个云制造型企业中随机选择了 100 个企业进行实验，每个企业部署一个虚拟区块链节点，因此本实验在 5 台服务器上至多虚拟了 100 个节点。本文用 QoS 指标中的传输延迟加上节点间的实际通信时延，作为共识节点处理消息、传输消息的真实延迟，以此在实验中体现 QoS 值对区块链节点性能的影响。

实验中一些参数设置如下：初始候选共识集群规模 $P=50$，最终共识集群规模 $C=30$；一个共识周期包含共识次数 $m=5$，淘汰及轮换列表长度 $d=4$；主节点每隔 $\Delta t=0.1$s 发起一次共识请求。

4.2.2 共识时延分析

共识时延是指从主节点发起一次共识请求到整个共识过程完成所需要的时间，较低的共识时延意味着系统能更快地确认消息并具有更高的吞吐量，因此共识时延是衡量区块链性能的重要指标。其计算公式如下：

$$delay = t_{finish} - t_{req} \tag{11}$$

式中 t_{req} 表示发起某一共识过程的时刻，t_{finish} 表示该共识过程结束的时刻。

实验 1：设置区块链节点总数为 100。观察随着共识过程进行，Q-PBFT 算法共识时延的变化。

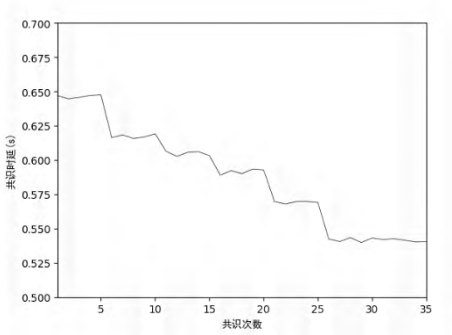

图 6 Q-PBFT 算法共识时延变化曲线

由图 6 可知，随着共识次数的增加，Q-PBFT 算法的共识时延在整体上呈现下降的趋势，并且最终稳定在 0.54s。在过渡阶段，由于共识节点淘汰过程发生在一个共识周期的结束阶段，因此如图 6 所示，一个共识周期内的共识时延不会有明显变化。只有在一个共识周期的结束阶段，共识集群规模减小，才会导致新周期内的共识时延降低。

实验 2：设置区块链节点的数目由 10 增加到 100，比较 PBFT 算法，S-PBFT 算法、Q-PBFT 算法、QPBFT_1 算法的平均共识时延。S-PBFT 算法随机筛选共识节点；Q-PBFT_1 算法是 Q-PBFT 算法的退化版本，只执行基于 QoS 值的共识节点筛选算法，而不执行优化的一致性协议。实验 2 的实验结果如图 7 所示，图 7 中横坐标下每个节点数目所对应的共识时延为多次共识过程的平均共识时延。

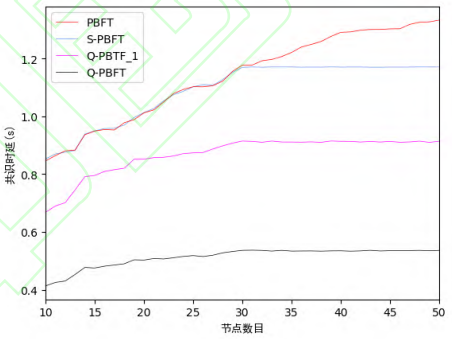

图 7 不同算法共识时延对比

由图 7 可知，当 $N \leq 30$ 时，四种算法的共识时延均随着区块链节点数目的增加而增加；当 $N > 30$ 时，PBFT 算法的共识时延随着区块链节点数目的增加而继续增加，而 S-PBFT 算法、Q-PBFT 算法、Q-PBFT_1 算法由于将共识集群的规模控制在了 30，因此共识时延没有显著变化。S-PBFT 算法中的共识节点是随机筛选的，部分节点可能具有较大的数据传输时延，因此其共识时延高于 Q-PBFT_1 算法。Q-PBFT 算法在 Q-PBFT_1 算法的基础上，又进一步优化了一致性协议。因此在各种节点数量下，Q-PBFT 算法都体现出了最优的共识性能。

5 结束语

近年来，基于区块链技术建立一个安全的、多方可信的云制造服务平台，成为了学者们研究的一个趋势。

共识算法作为区块链的核心技术，往往决定了区块链系统的性能。然而，以 PBFT 算法为例的现有的共识算法通常都会带来显著的资源消耗。服务质量（Quality of Service，QoS）是衡量云制造服务性能的重要指标，在云制造服务场景下，本文就提出了基于 QoS 值的改进 PBFT 算法 Q-PBFT，从共识节点筛选和一致性协议改进两个方面优化 PBFT 算法。在共识节点筛选算法中，首先基于区块链节点的 QoS 值筛选出具有高 QoS 值的节点构成候选共识集群，然后提出了集群的动态调整和轮换策略。在一致性协议改进算法中，将传统 PBFT 算法中的三阶段广播协议优化为二阶段协议，通过减少通信复杂度提高共识效率，并设计了广播消息的数据格式，使得算法能够识别拜占庭节点。理论分析和实验分析表明 Q-PBFT 算法能够在满足安全性的前提下，减少带宽消耗、降低共识时延，从而提升基于区块链的云制造服务平台的性能。

在未来的研究中，我们拟搭建基于区块链的云制造服务平台，将本文提出的 Q-PBFT 算法作为共识组件接入平台。然后围绕该平台，开展进一步的优化工作，如完善奖惩机制，提出面向用户隐私保护的数据存储和读取方法等，从而促进云制造服务生态的繁荣发展。

参考文献：

[1] Erl T. Service-Oriented Architecture: Concepts, Technology, and Design[M]. Prentice Hall PTR, 2005.

[2] C Matt. Software as a service[J]. Gpsolo, 2007, 24(3):28-31.

[3] Theobaldt L, Vervest P. Making Business Smart: How to Position for Business as a Service[M]. 2012.

[4] LI Bohu, ZHANG Lin, WANG Shilong, et al. Cloud manufacturing: a new service-oriented networked manufacturing model [J]. Computer Integrated Manufacturing Systems, 2010, 16(1): 1-16 (in Chinese). [李伯虎, 张霖, 王时龙, 等. 云制造——面向服务的网络化制造新模式[J]. 计算机集成制造系统, 2010, 16(1): 1-16.]

[5] YI Shuping, LIU Mi, WEN Peihan. Overview of cloud manufacturing service based on lifecycle theory [J]. Computer Integrated Manufacturing Systems, 2016, 22(4): 871-883 (in Chinese). [易树平, 刘觅, 温沛涵. 基于全生命周期的云制造服务研究综述[J]. 计算机集成制造系统, 2016, 22(4): 871-883.]

[6] Yuan Yong, Ni Xiaochun, Zeng Shuai, et al. Development Status and Prospect of Blockchain Consensus Algorithm[J]. Journal of Automation, 2018, 44(11):2011-2022 (in Chinese). [袁勇, 倪晓春, 曾帅, 等. 区块链共识算法的发展现状与展望[J]. 自动化学报, 2018, 44(11):2011-2022.]

[7] Li J, Wolf T . A One-Way Proof-of-Work Protocol to Protect Controllers in Software-Defined Networks[C]// Symposium on Architectures for Networking & Communications Systems. IEEE, 2016

[8] Miguel, Castro, Barbara, et al. Practical byzantine fault tolerance and proactive recovery[J]. ACM Transactions on Computer Systems, 2002.

[9] Hao Yushi, Fan Yushun. Cloud Manufacturing Service Recommendation based on Scenario Recognition[J]. Computer Integrated Manufacturing Systems, 2020(8):2007-2019(in Chinese). [郝予实, 范玉顺. 基于场景识别的云制造服务推荐[J]. 计算机集成制造系统, 2020(8):2007-2019.]

[10] Wang Qiang, Liu Changchun, Zhou Baoru. Trusted Transaction Method of Manufacturing Services Based on Blockchain[J]. Computer Integrated Manufacturing Systems, 2019(12): 3247-3257 (in Chinese). [王强, 刘长春,

周保茹. 基于区块链的制造服务可信交易方法[J]. 计算机集成制造系统, 2019(12): 3247-3257.]

[11] Xu Xuesong, Jin Yong, Zeng Zhi, et al. Hierarchical Lightweight High-throughput Blockchain for Industrial Internet Data Security[J]. Computer Integrated Manufacturing Systems, 2019, 25(12):3258-3266 (in Chinese). [徐雪松, 金泳, 曾智, 等. 应用于工业互联网数据安全的分层轻量级高通量区块链方法[J]. 计算机集成制造系统, 2019, 25(12):3258-3266.]

[12] Gao Z, Fan Y, Wu C, et al. DSES: A Blockchain-Powered Decentralized Service Eco-System[C]. 2018 IEEE 11th International Conference on Cloud Computing (CLOUD),2018, pp. 25-32.

[13] Shen Xin, Pei Qingqi, Liu Xuefeng. Survey of Block Chain[J]. Chinese Journal of Network and Information Security, 2016, 2(11): 11-20 (in Chinese). [沈鑫, 裴庆祺, 刘雪峰. 区块链技术综述[J]. 网络与信息安全学报, 2016, 2(11):P.11-20.]

[14] Thakkar P, Nathan S and Viswanathan B. Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform[C]. 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), 2018, pp. 264-276.

[15] Zhu Hai, Jin Yu. DS-PBFT: A Distance Based Consensus Algorithm for Blockchain[J/OL]. Journal of Chinese Computer Systems (in Chinese). [朱海,金瑜.DS-PBFT :一种基于距离的面向区块链的共识算法[J/OL].小型微型计算机系统.]

[16] Wang Haiyong, Guo Kaixuan, Pan Qiqing. Byzantine Fault Tolerance Consensus Algorithm Based on Voting Mechanism[J]. Journal of Computer Applications, 2019, 39(6):1766-1771 (in Chinese). [王海勇, 郭凯璇, 潘启青. 基于投票机制的拜占庭容错共识算法[J]. 计算机应用, 2019, 39(6):1766-1771.]

[17] Lao L, Dai X, Xiao B, et al. G-PBFT:a location-based and scalable consensus protocol for IoT-blockchain applications[C]. IEEE International Parallel and Distributed Processing Symposium(IPDPS),IEEE,2020:664-673.

[18] Zheng Z, M. R. Lyu. WS-DREAM: A distributed reliability assessment Mechanism for Web Services[C]. 2008 IEEE International Conference on Dependable Systems and Networks With FTCS and DCC (DSN), 2008, pp. 392-397.

作者简介：

伍 星（1995—），男，四川达州人，博士研究生，研究方向：面向服务计算、云制造服务系统、区块链、机器学习等，E-mail: wuxing17@mails.tsinghua.edu.cn；

+范玉顺（1962—），男，江苏扬州人，教授，博士，博士生导师，研究方向：工作流理论与技术、面向服务计算、企业建模与企业集成等，通讯作者，E-mail: fanyus@tsinghua.edu.cn；

郜振锋 （1991—），男，江苏南通人，博士研究生毕业，就职于深信服科技股份有限公司，研究方向：云计算、人工智能、区块链等，Email: gzf@sangfor.com.cn。