基于改进 PBFT 算法防御区块链中 sybil 攻击的研究

赖英旭[1,2,3]，薄尊旭[1]，刘静[1,4]

（1. 北京工业大学信息学部，北京 100124；2. 信息保障技术重点实验室，北京 100072；
3. 智能感知与自主控制教育部工程研究中心，北京 100124；4. 西安电子科技大学陕西省网络与系统安全重点实验室，陕西 西安 710071）

摘 要：针对 sybil 攻击对区块链技术有极大危害的问题，在联盟链中对 PBFT 算法进行改进，以防御 sybil 攻击。首先，借鉴基于权益证明的共识算法思想，通过建立信誉模型，根据各节点共识过程中的行为计算节点的信誉值，并依据信誉值的大小赋予节点不同的话语权；然后，在 PBFT 算法中加入了 pre-commit 阶段来减少节点间通信的次数。形式化分析推理和安全性测试表明，改进的 PBFT 算法不仅可以有效防御区块链中的 sybil 攻击，而且使区块链系统性能在 TPS 和区块生成时延方面有明显提高。

关键词：区块链；sybil 攻击；PBFT 算法；信誉模型；话语权

中图分类号：TP309

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2020170

Research on sybil attack in defense blockchain based on improved PBFT algorithm

LAI Yingxu[1,2,3], BO Zunxu[1], LIU Jing[1,4]

1. Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China
2. Science and Technology on Information Assurance Laboratory, Beijing 100072, China
3. Engineering Research Center of Intelligent Perception and Autonomous Control, Ministry of Education, Beijing 100124, China
4. Shaanxi Key Laboratory of Network and System Security, Xidian University, Xi'an 710071, China

Abstract: Aiming at the problem that sybil attack has great harm to block chain technology, a method to improve the PBFT algorithm in the alliance chain to defend against sybil attacks was proposed. Firstly, using the idea of consensus algorithm based on proof of rights and interests, a reputation model was established, the reputation value of each node accorded to the behavior of each node in the consensus process was calculated, and different discourse rights accorded to the size of the reputation value was given. Then pre-commit phase was added to the PBFT algorithm to reduce the number of communication between nodes. The solution through formal analysis and reasoning and security testing shows that the improved PBFT algorithm can not only effectively defend against sybil attacks in the blockchain, but also make the performance of the blockchain system in terms of TPS and block generation delay.

Key words: blockchain, sybil attack, PBFT algorithm, reputation model, discourse right

1 引言

自 2008 年比特币问世以来，电子加密货币已成为当今社会的热点话题。作为电子加密货币底层支撑技术的区块链也进入了大众视野并得到了广泛关注。目前，区块链被应用在金融、物联网和贸

收稿日期：2020−04−19；**修回日期**：2020−07−10

基金项目：北京市自然科学基金—海淀原始创新联合基金资助项目（No.19L2020）；信息保障技术重点实验室基金资助项目（No.614211204031117）；陕西省网络与系统安全重点实验室开放课题基金资助项目（No.NSSOF1900105）；工业和信息化部 2018 年工业互联网创新发展工程基金资助项目

Foundation Items: Beijing Municipal Natural Science Foundation (No.19L2020), Foundation of Science and Technology on Information Assurance Laboratory (No.614211204031117), Foundation of Shaanxi Key Laboratory of Network and System Security (No.NSSOF1900105), Industrial Internet Innovation and Development Project of the Ministry of Industry and Information Technology of China in 2018

易管理等多种场景中，这对区块链的安全性提出了更高的要求，要使区块链真正得到实际应用，解决其安全性是首要条件[1]。已经出现的智能合约代码漏洞[2]、自私挖矿[3]、Eclipse 攻击[4]等安全问题对区块链的存在与发展带来了危害。

区块链的网络架构一般采用点对点（P2P, peer-to-peer）网络架构，所有节点的地位都是对等的，并且以拓扑结构相互连通，节点之间采用中继转发的模式进行通信[5]。由于区块链采用 P2P 的网络架构，因此更容易遭受 sybil 攻击，这种攻击发生在恶意节点伪造多个虚假节点身份加入区块链网络的过程中。例如，共识节点进行投票的过程中，sybil 节点为了谋取利益故意投票反对正确的共识结果，从而妨碍区块链中的节点达成共识。此外，sybil 攻击也可以破坏数据的冗余机制，恶意节点通过伪造多个虚假节点，将之前需要备份到多个节点的数据备份到了同一个恶意节点，严重破坏了分布式存储系统的安全性和可靠性。

目前，区块链中的 sybil 攻击通常配合 Eclipse 攻击共同发起。当区块链中的正常节点受到攻击时，无论是发出的还是接收的请求信息都可能被 sybil 节点截获并进行篡改，如果正常节点收到的信息是被 sybil 节点篡改过的，就无法进行正常的共识过程，sybil 攻击对区块链技术有极大的危害。本文针对防御区块链中的 sybil 攻击开展研究，主要贡献如下。

1) 针对 sybil 攻击的特点，在区块链网络中建立信誉模型来计算各共识节点的信誉值，借鉴权益证明（PoS, proof of stack）引入币龄的概念，通过币龄大小的关系降低了计算机进行 Hash 计算的难度[6]。将共识节点的信誉值与共识节点的投票权重相对应，各节点的信誉值不同，其拥有的话语权也不同，在共识过程中各节点根据投票权重来达成共识。

2) 对实用拜占庭容错（PBFT, practical Byzantine fault tolerance）算法进行改进，根据共识节点的信誉值选出信誉值最高的当前节点作为主节点 S_p。共识协议中增加了 pre-commit 阶段，不仅保证了节点间仍能正常达成共识，而且减少了节点间通信的次数。

3) 对改进的共识协议进行形式化证明，验证共识协议的安全性，通过实验证明改进的共识协议仍是安全的。

2　相关研究

2.1　sybil 攻击

Douceur 等[7]在 P2P 网络系统的应用中提出 sybil 攻击的存在，认为如果网络中没有一个集中式认证机构，很可能会出现 sybil 攻击。现阶段，大型 P2P 系统在处理远程攻击或者系统故障时，大多采用数据冗余措施来防御这些安全威胁，然而如果有一个恶意节点伪造出多个节点身份，同时对外宣称都是正常节点，那么这个恶意节点将会破坏系统的数据冗余机制。

如图 1 所示，sybil 攻击是指攻击者通过创建多个虚假身份加入 P2P 网络，并使用这些 sybil 节点来获得不成比例的影响，从而为自身谋取不正当的利益。sybil 攻击不仅会破坏对等网络中资源共享的安全性、消耗正常节点的计算资源，严重时甚至会控制整个网络系统，造成更大的危害。

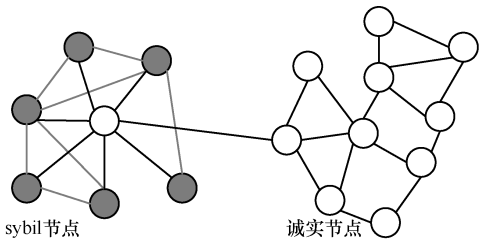

图 1　sybil 攻击

当前针对 sybil 攻击防御和检测的方法主要分为 4 类：基于图的方法、基于机器学习的方法、手动验证和传统防御方法[8]。其中，基于图的方法[9-12]是利用社交网络信息来识别 sybil 节点，但这些方法都依赖 sybil 节点和正常节点的连接是有限的这一假设，因此这类方法的扩展性很差。基于机器学习的方法[13-15]是从节点的行为和日志中提取特征来区分正常节点和 sybil 节点，但这类方法是根据已知 sybil 节点的行为来检测未知 sybil 节点，因此 sybil 节点可以通过改变不利于自身的行为来轻松地逃避检测。手动验证的方法是给经过人工验证的社交网络账户设置特别标志，拥有这样标志的用户发布的内容最有可能是真实的，但这种方法不仅验证效率低，也不适用于拥有众多对等节点的 P2P 网络。传统防御 sybil 攻击的方法是通过引入可信任的权威认证机构来对加入网络的节点进行认证，这种方法同样不适用于没有集中式认证机构的分布式系统，并且 Fabric1.0 版本中采用 CA（certificate

authority）作为证书的颁发机构，节点之间通信使用的公私钥由 CA 提供，如果 CA 服务出现故障，那么整个系统都会出现问题，因此对于去中心化的区块链系统来说扩展性差。

2.2 实用拜占庭容错算法

PBFT 算法由 Castro 和 Liskov 提出[16]，解决了 BFT 算法效率差的问题，并且算法的复杂度由指数级降为多项式级，使其可以应用于需要处理大量事件但吞吐量不大的系统。PBFT 作为经典的 BFT 状态机副本复制算法和基于投票的主流算法，同时保证了安全性和活性。

安全性：在恶意节点数量不超过 $\frac{N-1}{3}$ 的情况下，副本节点的复制服务能够满足线性一致性。

活性：所有客户端都会收到针对他们请求的回复，使用弱同步假设保证在一定时间内传递消息规避 FLP（Fischer, Lynch, Patterson）不可能性的结果。

在 PBFT 算法中，共识节点发送的消息都必须由节点进行签名，其他节点以此验证消息的真伪。该算法的共识过程主要分为 3 个阶段：预准备、准备和提交。如果收到超过 $\frac{1}{3}$ 不同节点的同意消息，则提交的交易信息是有效的。在使用 PBFT 算法作为共识算法的区块链网络中，N 个节点可以包含 f 个拜占庭恶意节点，其中 $f \leqslant \frac{N-1}{3}$。也就是说，$N-f \geqslant 2f+1$，因此 PBFT 算法要达成共识，需要至少 $2f+1$ 个节点将共识信息添加到分布式账本中。

由于 PBFT 算法是分布式系统中达成共识一致性的一种性能良好的解决方案，增强了分布式环境下数据和系统服务的可用性[17]，因此 PBFT 算法在区块链中得到了广泛的应用。王海勇等[18]提出在 PBFT 算法中引入投票机制，以监督生产节点诚实工作。Wang 等[19]提出了基于信用授权的拜占庭容错（CDBFT, credit-delegated Byzantine fault tolerance）算法，通过投票奖惩和信用评估共识过程中每个节点的行为。文献[20]提出了一种高性能、可扩展的拜占庭容错（HSBFT, high performance and scalable Byzantine fault tolerance）算法，在正常情况下可将算法的复杂度降到 $O(n)$。在 Fabric 项目 0.6 版本和 Honey badger[21]项目中都使用 PBFT 作为核心的共识算法。然而，上述研究在防御恶意节点方面作用有限，缺乏对不同参与节点分配不同话语权的考虑。

在 PBFT 共识算法下关于防御 sybil 攻击的研究中，Alex 等[22]将信誉系统与分布式一致性协议相结合，引入声誉模块 ReCon 放置在 PBFT 等各种共识协议上，对 sybil 攻击有很强的抵抗能力。闵新平等[23]提出了许可链多中心动态共识机制，并且设计了一种多主节点的 PBFT 算法（MPBFT, multi primary node Byzantine fault tolerance），通过安全性分析与证明，许可链多中心动态共识机制可保证交易不可篡改，同时预防 sybil 攻击。Zhang 等[24]利用区块链的不可篡改性提出了一种 Hash-oriented 的 PBFT 算法，旨在减少共识确认的时延，安全性分析表明，其能够有效防御区块链中的双花攻击和 sybil 攻击。

2.3 信誉模型

在当前的研究工作中，P2P 网络中已经实现了各种不同的信誉系统。实现电子商务等的信誉系统需要可靠的中央服务器，用来记录用户的历史行为并进行信誉评级。而有些信誉系统则使用分布式数据库，网络中所有节点的地位都是对等的并且都能实时更新本地副本，使用这种信誉系统的节点只记录与其发生信息交互的对等节点的信誉值。

Janbi 等[25]提出了一种对分布式排名的信誉系统 DRank 进行结构优化的方法，提高了 DRank 的性能，但容易受恶意节点共谋攻击的影响。Sarah 等[26]对 AuthenticPeer 进行改进，通过防止不受信任的文件传播来增加信誉，并减少恶意节点的集体欺骗行为。Gupta 等[27]在 P2P 网络上实现了集中式服务器模型，确保网络中的所有节点都可以访问最新数据且不要求网络中的所有节点使用信誉服务，但该方案的前提是网络中不存在恶意节点，所以未能解决恶意节点可能串通来增加其自身信誉值以便从系统中获利的可能性。黄建华等[28]提出了基于信任度证明的共识机制（PoT, proof of trust），解决了权益证明机制中存在的易受贿赂攻击、币龄累积攻击，以及工作量证明机制中存在的自私挖矿等问题，但是对区块链中存在的 sybil 攻击没有提出很好的解决方案。

2.4 网络安全协议的形式化证明与分析

在网络安全协议的形式化证明与分析中，安全协议的分析方法主要包括模态逻辑技术、定理证明和模型检测技术[29]。其中，模态逻辑技术是一种比较重要的安全协议分析方法。模态逻辑用命题假设

和推理规则来表示主体对消息的知识或信仰，运用推理规则可以从已知的知识和信仰推导出新的知识和信仰[30]。

在总结和完善 BAN（Burrows, Abadi and Needham）逻辑和其他类 BAN 逻辑缺点和不足的基础上发展出了 SVO（Syverson, Van Orschot）逻辑，它的出现也标志着模态逻辑在安全协议的分析方法中走向成熟。SVO 逻辑不仅拥有规范的模态理论语义和极其出色的扩展能力，还建立了完备的理论模型和详细的计算模型，消除了在理解模态逻辑形式化表达式含义的过程中可能存在的歧义，通过规范的理论语义可以更好地理解协议消息所要表达的真正含义。

3　共识算法的改进

在本文改进的 PBFT 算法中，通过共识节点的可信状态选出一个信誉值最高的节点作为主节点 S_p。由于在 PBFT 共识过程中，有 2 个阶段需要传输的网络消息数为 $O(n^2)$，造成了很大的网络开销，因此在改进的共识算法中增加了 pre-commit 阶段来减少节点间通信的次数。每轮共识过程中主节点 S_p 会生成一个新区块并广播到所有的共识节点，同时参与共识的节点依据信誉模型计算节点的信誉值，依据信誉值为共识节点赋予不同的话语权。在达成共识的投票过程中，各节点拥有不同的投票权重，信誉值高的节点拥有更大的投票权重，而恶意节点由于信誉值低拥有很小的投票权重甚至没有投票权重，因此可以杜绝恶意节点对投票结果的干扰。本文的全局变量参数如表 1 所示。

表 1　　全局变量参数

变量参数	符号
共识节点的数量	N
主节点	S_p
节点信誉值	R
共识轮次	t
可信状态	TS
节点的信誉值总和	R_v
信誉阈值	$R_{\text{threshold}}$

3.1　系统模型

本文假设在区块链系统中有 N 个共识节点 $S=\{S_0,S_1,\cdots,S_{N-1}\}$，每轮共识过程都存在一个主节点 S_p，所有共识节点把接收的事务信息先缓存到本地，同时主节点 S_p 将客户端发来的有效交易事务打包到一个区块中。如果新区块被大多数共识节点验证通过，则认为它是有效的，该区块将被添加到区块链中。如果没有被大多数共识节点验证通过，那么这个区块被舍弃。在区块链系统中对 PBFT 算法进行改进，将共识节点的投票权重与所拥有的信誉值相对应，每个共识节点都维护更新一个共识节点信息表（CNIL, consensus node information table），如表 2 所示。

表 2　　共识节点信息

节点序号	邻接节点集合	信誉值	可信状态	节点 ID
0	{1, 2, 3}	0.5	3	0xxx
1	{0, 2, 3}	0.5	3	1xxx
2	{0, 1, 3}	0.5	3	2xxx
3	{0, 1, 2}	0.5	3	3xxx

CNIL 包含当前区块链系统中与本节点进行过信息交互的邻居节点的集合、信誉值、可信状态以及节点的 ID，其中每个节点随机选择邻居节点建立连接[31]。及时更新和维护共识节点的信息列表非常重要，关系到节点的信誉值和节点拥有的不同话语权，将所有节点的信息列表的初始值设为相同值。

3.2　建立信誉模型

信誉模型用来计算每个节点的信誉值，而信誉值由共识过程中节点的行为决定。信誉模型作为共识协议的一部分，可以在每个参与共识的节点中执行，并且信誉值是独立计算的，可以与共识消息进行同步广播。在改进的共识算法中，设定的信誉值 R 是 0～1 的实数，信誉值越大，可信度越高。对于系统新添加的共识节点，其初始信誉值都设为 0.5。由于主节点和其他副本节点在共识过程中的行为不同，本文分别讨论了 2 种情况。

3.2.1　S_i 为主节点

对于主节点而言，在 t 轮共识过程中如果有新区块产生，那么主节点的信誉值会增加，并且随着共识轮次越来越多，信誉值增长的速度会越来越慢，但最大值不会超过 1。如果没有新区块产生，那么主节点的信誉值会下降，下降的速度由系数 x 的值决定。如果主节点向其他节点发送了不同的节点信息列表，那么主节点的信誉值会降为 0，并将其从当前的共识节点信息列表中删除。设 $R_i(t)$ 为区块链中第 t 轮共识后节点 S_i 的信誉值，则 $R_i(t+1)$ 为

$$R_i(t+1)=\begin{cases}\min\left(1,R_i(t)\left(1+\dfrac{1}{t+1}\right)\right),\text{有新区块产生}\\ xR_i(t),\text{没有新区块产生}\\ 0,\text{向不同节点发送不同的信息列表}\end{cases}\tag{1}$$

3.2.2 S_i 为副本节点

对于副本节点而言，在 t 轮共识的过程中如果向其他节点发送了相同的信息列表并且投票结果与最终结果一致，则副本节点的信誉值会缓慢增加，但不会超过 1。如果副本节点在本轮没有参与共识过程，则其信誉值会降低，下降的速度由 x 的取值决定。如果副本节点参与了共识过程，但是投票结果与最终结果不一致，则其信誉值会降低，下降速度由系数 y 的取值决定。虽然 x 和 y 的值都决定了信誉值降低的速度，但根据节点行为的不同，需要以不同的速度来降低节点的信誉值。如果检测到同一共识节点发送了不同的信息列表，则该节点将被视为 sybil 节点，其信誉值降为 0，并将其从当前的共识节点信息列表中删除，如式(2)所示。

$$R_i(t+1)=\begin{cases}\min\left(1,R_i(t)\left(1+\dfrac{1}{t+1}\right)\right),\text{节点发送相同的信息}\\ \text{列表并同意多数}\\ xR_i(t),\text{节点在本轮没有发送信息}\\ yR_i(t),\text{节点在本轮不同意多数}\\ 0,\text{向不同节点发送不同的信息列表}\end{cases}\tag{2}$$

其中，$0<x<1$，$0<y<x<1$

在上述信誉模型中，所有正常参与共识的节点的信誉值都是缓慢增加的。随着系统持续运行，系统的话语权将更多集中于正确的共识节点。此外，可以考虑在系统的实际应用中加入激励机制，正常工作的节点拥有更高的信誉值、更多的话语权，会获得更多实质性的奖励。如果节点的信誉值过低，则获得很少的奖励，甚至会将其从当前的共识节点信息列表中删除。

3.3 主节点更新算法

PBFT 算法通过 $p=v \bmod |R|$ 来进行主节点更换，其中 v 为视图编号。改进的 PBFT 算法根据节点的信誉值来进行主节点的更换，节点的信誉值越高，在主节点更新的过程中就有越大的概率当选为主节点，如式(3)所示。

$$\forall S_i,S_j\in S:R_{(S_i)}\geqslant R_{(S_j)}\Rightarrow P_{(S_i|D)}\geqslant P_{(S_j|D)}\tag{3}$$

其中，S 为共识节点集合；P 为共识节点当选为主节点的概率；D 为指数分布，在 C2C（consumer to consumer）网络和大多数社交网络中的信誉分布都是指数分布[32]。指数分布的概率密度函数 $F(x)$如式(4)所示。

$$F(x)=\begin{cases}\lambda e^{-\lambda x},x\geqslant 0\\ 0\quad,\quad x<0\end{cases}\tag{4}$$

为了保证信誉值越高的节点有越大的概率当选为主节点，根据文献[22]中模拟实验，当设置最小恶意节点的概率 α_1=0.05 时，共识节点 N={5 000, 10 000, 20 000, 30 000}分别运行 10 000 轮后达成共识的成功率最高。本文不仅考虑共识节点的可信状态 μ，同时确保指数分布截断在区间(0,N]，因此取 $\lambda=\dfrac{-\ln(0.05\mu^2)}{N}$，$\mu$=1,2,3 分别对应共识节点的前 3 种可信状态，当共识节点的信誉值低于初始设定值 0.5 时，则其不在更换主节点的考虑范围内。共识节点的可信状态 TS=1，则 μ=1 时，有

$$\int_0^N F(x)\mathrm{d}x=1-e^{-\lambda x}\,|_0^N=0.95\tag{5}$$

也就是说，可信状态 TS=1 的共识节点有 95%的概率当选为主节点。依次类推，TS=2 和 TS=3 的共识节点分别有 80%和 55%的概率当选为主节点。本文使用指数分布，信誉值较高的节点有较大的概率当选主节点，而信誉值较低的节点当选主节点的概率较小。如果有多个节点的信誉值相等，则优先选择没有当选过的节点作为主节点。

共识节点的可信状态由信誉值 R 决定，因此可将共识节点分为 6 种可信状态，分别为良好节点、正常节点、初始节点、异常节点、错误节点和恶意节点，如表 3 所示。

表 3 节点可信状态

节点分类	R 范围	TS
良好节点	$(\alpha,1]$	1
正常节点	$(0.5,\alpha]$	2
初始节点	0.5	3
异常节点	$[\beta,0.5)$	4
错误节点	$(0,\beta)$	5
恶意节点	0	6

在本文设计的信誉模型中，分析了节点作为主

节点和副本节点的不同行为特征的评价标准。在共识过程中，当主节点被认为是恶意节点时，其信誉值会直接降为 0。这时需要在参与的共识节点中重新选举主节点。更新主节点的基本原则是，节点的信誉值越高越容易当选主节点。主节点的更新流程如图 2 所示。

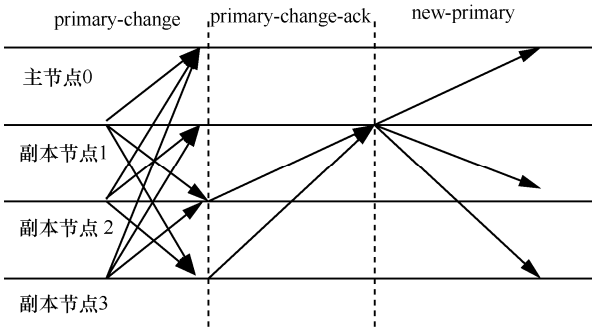

图 2　主节点的更新流程

变更主节点算法的具体步骤如下。

步骤 1　共识过程中，当主节点的信誉值低于设定的阈值时，各副本节点需要选取当前节点中信誉值最大的节点作为主节点，副本节点向其他节点广播 primary-change 消息的内容为<primary-change, S_m, $R_{\max}$, CNIL$_i$, i>，其中 S_m 为新当选主节点的序号，$R_{\max}$ 为当前节点的最大信誉值，i 为副本节点序号。

步骤 2　其他副本节点收集并计算是否有 $2f$ 个不同副本节点（不包括其自身）发送更新主节点为 S_m 的 priamry-change 消息，如果是，则副本节点向 S_m 发送 primary-change-ack 消息，其内容为<primary-change-ack, S_m, i, ack>，并执行步骤 3；否则直接结束。

步骤 3　新当选的主节点 S_m 向其他副本节点发送 new-primary 消息的内容为<new-primary, S_m, V, CNILN>，其中 V 为 primary-change 消息集合。

3.4　改进共识算法

在共识算法的改进中，通过计算各节点的信誉值为各节点分配不同的话语权。根据各共识节点提供的信息列表评估共识过程中每个节点的信誉值，不仅可以检测出其中的恶意节点，而且可以将检测出的恶意节点从共识节点信息表清除。良好节点的信誉值会逐渐累加，在共识过程中的话语权也逐渐增加，而恶意节点的影响会逐渐减少。共识过程中，节点 i 更新共识状态的条件是向其发送消息的共识节点的信誉值总和 R_v 足够大。其他节点信誉值总和 R_v 的计算方式如下。

假设区块链网络是一个有向网络 $G(\varepsilon, E)$，其中 ε 为 n 个节点集合，E 为有向链路加权集合，可以用来预测共识节点之间协商的可能结果。

在区块链网络中，R_v 由与其进行信息交互的节点信誉值共同决定，即节点 i 在 t 轮共识中收到的信誉值为

$$R_v(t) = R_i(t) + \sum_{j=1}^{\varepsilon}[R_j(t) - R_i(t)]\left(\boldsymbol{\phi}^{\mathrm{T}}\right)_{ij} \tag{6}$$

其中，$R_i(t)$ 为节点 i 在 t 轮共识时的信誉值，矩阵 $\boldsymbol{\phi} = (\phi_{ij}) \in \boldsymbol{R}^{\varepsilon\times\varepsilon}$ 由网络拓扑及链路上的信誉值构成，$\boldsymbol{\phi}^{\mathrm{T}}$ 为其转置矩阵。

如果 R_v 受到多个共识节点的影响，那么节点 i 收到的信誉值就是作用于 i 上所有影响的总和，如式(7)所示。

$$\Delta R_i(t) = \sum_{j=1}^{\varepsilon}[R_j(t) - R_i(t)]\phi_{ij} \tag{7}$$

状态方程可表示为 $\Delta R(t) = LR(t)$，其中 $\boldsymbol{L}$ 是 $\boldsymbol{\phi}^{\mathrm{T}}$ 的 Laplacian 矩阵。因此，更新规则为

$$R(t+1) = R(t) + \boldsymbol{L}R(t) = (\boldsymbol{I} + \boldsymbol{L})R(t) \tag{8}$$

其中，矩阵 $\boldsymbol{T} = \boldsymbol{I} + \boldsymbol{L} = (T_{ij}) \in \boldsymbol{R}^{\varepsilon\times\varepsilon}$，其中，$T_{ij}$ 表示节点 i 受节点 j 的影响，$T_{ij} \in R$。当前节点的 R_v 受与其信息交互的共识节点的影响，使共识节点的 R_v 随着时间的推移而发生动态变化。

收到的其他节点 R_v 应不小于设定的阈值 $R_{\text{threshold}}$，如式(9)所示。

$$R_{\text{threshold}} = \frac{1}{N}\left(2\left\lfloor\frac{N-1}{3}\right\rfloor + 1\right) \tag{9}$$

改进的 PBFT 算法共有 6 个阶段，其中最主要是以下 4 个阶段：预准备、准备、预提交和提交。改进的 PBFT 算法中引入了信誉模型，通过各节点共识过程中的行为对节点进行信誉评估，检测区块链中的 sybil 节点，工作流程如图 3 所示，具体步骤如算法 1 所示。

算法 1　改进 PBFT 算法

输入　交易 tx

输出　共识结果

1) 客户端 c 发起交易 tx，并将交易广播到主节点 0。主节点 0 收到发送的交易 tx，首先验证 tx 是否有效，若无效直接将其删除；若有效则将 tx 打包到区块中，并根据区块体内的信息生成区块头 B_{head}。

2) 主节点 0 广播预准备（pre-prepare）消息到

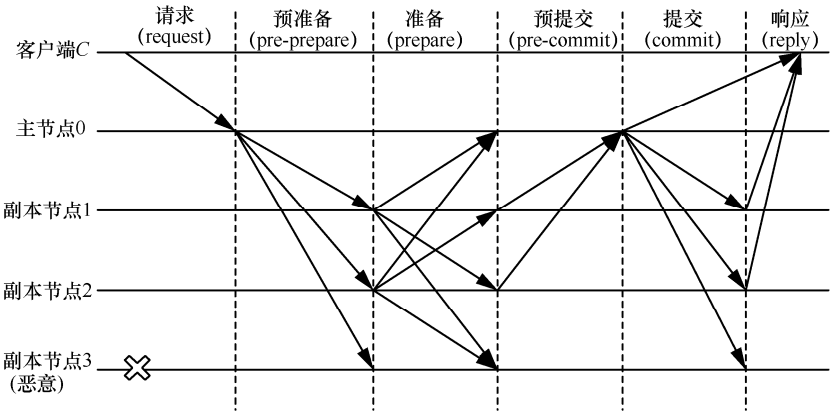

图 3 改进 PBFT 算法工作流程

各副本节点，内容为

<<PRE-PREPARE, h, d, t, P_0, $\mathrm{CNIL}_0>_{\sigma 0}$, B_{head}>

其中，h 是当前新区块的高度，d 是区块头 B_{head} 的摘要，t 是当前的时间戳，P_0 是当前主节点 0 的 ID，CNIL_0 是主节点 0 的共识节点信息表。

3) 副本节点 1,2 收到主节点 0 发送的预准备消息，首先验证新区块的有效性，验证通过则分别向其他节点发送 prepare 消息，内容为

<< PREPARE, h, d, t, P_i, $\mathrm{CNIL}_i>_{\sigma i}$, B_{head}>

4) 副本节点 1,2 收到其他副本节点发送的 prepare 消息，发送消息的节点拥有不同的信誉值。首先副本节点计算当前向其发送消息的节点的 R_v，如果 $R_v \geqslant R_{\mathrm{thershold}}$ 则在本地更新交易信息的共识状态并发送 pre-commit 信息，内容为

<<PRE-COMMIT, h, d, t, $P_i>_{\sigma i}$, $<\mathrm{CNIL}>_{\sigma i}$>

5) 主节点 0 将收到的副本节点发送的 pre-commit 信息进行比较，根据信誉模型计算当前每个节点的信誉值，更新本地的共识节点信息列表，同时将共识结果反馈给客户端和所有的副本节点，发送 commit 消息，内容为

<<COMMIT, h, d, t, $P_0>_{\sigma 0}$, $<\mathrm{CNIL}>_{\sigma 0}$>

6) 完成提交状态后，区块链中的副本节点更新本地的共识节点信息表，并将共识结果反馈给客户端，准备下一轮的共识过程。

在节点进行共识过程的不同阶段需要分别验证交易和新区块的有效性，验证的主要内容如下。

1) 验证交易的有效性

在进行共识的预准备阶段，主节点需要验证收到客户端发送的交易 tx 的有效性。验证 tx 的格式是否正确和时间戳是否有效，验证事务中的脚本是否可以正确的执行，如果通过验证则 tx 是有效的。

2) 验证新区块的有效性

当副本节点收到主节点生成的新区块时，需要验证新区块的有效性。验证区块头中的 Merkle 根是否正确和区块头中是否引用前一区块的哈希值，同时验证区块中的交易是否有效，如果通过验证则新区块则是有效的。

4 共识协议的形式化分析验证

共识协议的形式化分析验证主要分为共识协议的安全性分析和对协议的安全性测试。

4.1 共识协议的安全性分析

共识协议的安全性分析主要分为共识协议的理想化描述、初始化假设、设定安全目标和对协议进行理论分析 4 个部分。

4.1.1 共识协议的理想化描述

使用 SVO 逻辑对改进的 PBFT 算法的共识协议进行安全分析。首先需要对该协议的工作流程进行符合 SVO 逻辑的理想化描述。描述结果如下。

1) 客户端 c 向主节点 0 发送交易 tx，交易信息中带有时间戳和客户端标识。

$c \rightarrow 0$:<REQUEST, tx, t, c>

2) 主节点 0 对 tx 进行验证，验证通过后向其他副本节点发送带有自己签名的预准备消息，该消息中包含当前区块的高度、区块头的哈希摘要、主节点的 ID 以及共识节点信息表。

$0 \rightarrow \{1,2,3\}$:

<< PRE-PREPARE, h, d, t, P_0, $\mathrm{CNIL}_0>_{\sigma 0}$, B_{head}>

3) 副本节点 1, 2, 3 收到预准备消息验证之后向其他节点发送各自签名的准备消息，将接收到的

其他共识节点信息表存储到本地。

$1\rightarrow\{0,2,3\}$:

<< PRE-PREPARE, h, d, t, P_1, CNIL_1>$_{\sigma 1}$, B_{head}>

$2\rightarrow\{0,1,3\}$:

<< PRE-PREPARE, h, d, t, P_2, CNIL_2>$_{\sigma 2}$, B_{head}>

$3\rightarrow\{0,1,2\}$:

<< PRE-PREPARE, h, d, t, P_3, CNIL_3>$_{\sigma 3}$, B_{head}>

4) 各共识节点收集其他节点的准备消息，如果 $R_v \geqslant R_{\mathrm{thershold}}$，则在本地更新交易信息的共识状态，并发送 pre-commit 信息。

$1\rightarrow 0$:<< PRE-COMMIT, h, d, t, P_1>$_{\sigma 1}$,<CNIL>$_{\sigma 1}$>

$2\rightarrow 0$:<< PRE-COMMIT, h, d, t, P_2>$_{\sigma 2}$,<CNIL>$_{\sigma 2}$>

$3\rightarrow 0$:<< PRE-COMMIT, h, d, t, P_3>$_{\sigma 3}$,<CNIL>$_{\sigma 3}$>

$0\rightarrow\{1,2,3\}$:<< COMMIT, h, d, t, P_0>$_{\sigma 0}$,<CNIL>$_{\sigma 0}$>

5) 主节点和各副本节点向客户端 c 发送响应消息，将达成的共识结果返回给客户端。

$0\rightarrow c$: <REPLY, t, c, P_0, r>

$\{1, 2\}\rightarrow c$: <REPLY, t, c, P_i, r>

4.1.2 协议的初始化假设

使用 SVO 逻辑对共识协议进行推理分析的过程中，首先进行初始假设，即定义共识协议中每个节点在初始时刻拥有的知识和信仰。初始化假设是进行推理证明的第一步，根据共识协议流程，设定如下假设。

1) 关于密钥的有效性

$$N_0 \mid\equiv \mathrm{PK}(N_0, K_{N0}) \wedge \mathrm{PK}(N_1, K_{N1}) \wedge \mathrm{PK}(N_2, K_{N2}) \wedge \mathrm{PK}(N_3, K_{N3})$$

$$N_1 \mid\equiv \mathrm{PK}(N_1, K_{N1})$$

$$N_2 \mid\equiv \mathrm{PK}(N_2, K_{N2})$$

$$N_3 \mid\equiv \mathrm{PK}(N_3, K_{N3})$$

2) 关于信息的发送

客户端 c 向主节点 0 发送交易 tx，并验证交易的格式和时间戳，以及事务的脚本能否正常执行，验证通过后向副本节点发送信息

$$N_0 \mid\sim (h, d, P_0, \mathrm{CNIL}_0)$$

副本节点收到主节点 0 发送的信息，验证新生成区块的 Merkel 根和指向前一区块的哈希值，验证通过后向其他节点发送消息

$$N_1 \mid\sim (h, d, P_1, \mathrm{CNIL}_1)$$

$$N_2 \mid\sim (h, d, P_2, \mathrm{CNIL}_2)$$

$$N_3 \mid\sim (h, d, P_3, \mathrm{CNIL}_3)$$

3) 关于信息的接收

各共识节点收集其他节点的准备消息，如果 $R_v \geqslant R_{\mathrm{thershold}}$，则在本地更新交易信息的共识状态。

主节点 0 收到的信息为

$$N_0 \triangleleft (h, d, P_1, \mathrm{CNIL}_1)$$

$$N_0 \triangleleft (h, d, P_2, \mathrm{CNIL}_2)$$

$$N_0 \triangleleft (h, d, P_3, \mathrm{CNIL}_3)$$

副本节点 1 收到的信息为

$$N_1 \triangleleft (h, d, P_0, \mathrm{CNIL}_0)$$

$$N_1 \triangleleft (h, d, P_2, \mathrm{CNIL}_2)$$

$$N_1 \triangleleft (h, d, P_3, \mathrm{CNIL}_3)$$

副本节点 2 收到的信息为

$$N_2 \triangleleft (h, d, P_0, \mathrm{CNIL}_0)$$

$$N_2 \triangleleft (h, d, P_1, \mathrm{CNIL}_1)$$

$$N_2 \triangleleft (h, d, P_3, \mathrm{CNIL}_3)$$

副本节点 3 收到的信息为

$$N_3 \triangleleft (h, d, P_0, \mathrm{CNIL}_0)$$

$$N_3 \triangleleft (h, d, P_1, \mathrm{CNIL}_1)$$

$$N_3 \triangleleft (h, d, P_2, \mathrm{CNIL}_2)$$

4) 关于信息的新鲜性

$$N_0 \mid\equiv \#(h, d)$$

$$N_1 \mid\equiv \#(\mathrm{CNIL}_1)$$

$$N_2 \mid\equiv \#(\mathrm{CNIL}_2)$$

$$N_0 \mid\equiv \#(\mathrm{CNIL}_3)$$

$$N_1 \mid\equiv \#(h, d, P_0, \mathrm{CNIL}_0)$$

$$N_2 \mid\equiv \#(h, d, P_0, \mathrm{CNIL}_0)$$

$$N_3 \mid\equiv \#(h, d, P_0, \mathrm{CNIL}_0)$$

4.1.3 协议的安全目标

由于改进的 PBFT 算法中加入了共识节点信息表，每轮共识之后都要根据各节点的行为计算信誉值并且更新 CNIL。因此共识协议需要达到的安全目标主要是各节点的能否收到其他节点发送的 CNIL，同时保证客户端发送的请求能够达成共识。共识协议的安全目标用 SVO 逻辑语言表达如下。

首先，确保每个节点都能收到其他节点发送的共识节点信息表。

$$N_1 \mid\equiv \#(h, d)$$

$$N_2 \mid\equiv \#(h, d)$$

$$N_3 \mid\equiv \#(h, d)$$

其次，确保每个节点都能收到客户端发送的请求

$$N_1 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_2, \mathrm{CNIL}_3)$$

$$N_2 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_1, \mathrm{CNIL}_3)$$

$$N_3 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_1, \mathrm{CNIL}_2)$$

4.1.4 协议分析

根据初始化假设，使用 SVO 逻辑对共识协议进行逻辑推理，验证共识协议是否达到预先设定的安全目标，以此分析共识协议的安全性。共识协议的分析过程如下。

由假设 c: $N_1 \triangleleft (h, d, P_0, \mathrm{CNIL}_0)$和接收公理 $P \triangleleft [X]_K \supset P \triangleleft X$，可得

$$N_1 \triangleleft (h, d)$$

又由信息拥有公理 $P \ni (X_1, \cdots, X_n) \supset P \ni X_i$，可得

$$N_1 \ni (h, d)$$

由假设 d: $N_1 | \equiv \#(h, d, P_0, \mathrm{CNIL}_0)$和信任公理 $P | \equiv \alpha \supset P | \equiv (P | \equiv \alpha)$，得

$$N_1 | \equiv \#(h, d) \tag{10}$$

同理，可得

$$N_2 | \equiv \#(h, d) \tag{11}$$

$$N_3 | \equiv \#(h, d) \tag{12}$$

由假设 c: $N_1 \triangleleft (h, d, P_0, \mathrm{CNIL}_0)$ 和接收公理，可得

$$N_1 \triangleleft (\mathrm{CNIL}_0)$$

同理，可得

$$N_1 \triangleleft (\mathrm{CNIL}_2)$$

$$N_1 \triangleleft (\mathrm{CNIL}_3)$$

综合上述分析，可得

$$N_1 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_2, \mathrm{CNIL}_3) \tag{13}$$

$$N_2 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_1, \mathrm{CNIL}_3) \tag{14}$$

$$N_3 \triangleleft (\mathrm{CNIL}_0, \mathrm{CNIL}_1, \mathrm{CNIL}_2) \tag{15}$$

4.2 协议的安全性测试

4.2.1 测试工具

AVISPA（automated validation of internet security protocols and application）是自动验证网络协议和应用程序是否安全的模型工具，其融合了 4 种不同的分析组件：动态模型检验器（OFMC, on-the-fly model-checker）、基于逻辑约束的攻击搜索器（CL-AtSe, constraint-logic-based attack searcher）、基于 SAT 的模型检验器（SATMC, SAT-based model-checker）和基于自动逼近的树自动机安全协议分析（TA4SP, tree automata based on automatic approximations for analysis of security protocol）[33]。

AVISPA 工具提供了一种模块化和富有表现力的高级协议规范语言 HLPSL（high-level protocol specification language），用于指定安全问题和数据结构。其集成了多种不同的检测后端，能够自动进行各种分析技术，从协议伪造（通过发现输入协议的攻击）到有限和无限数量会话的基于抽象的验证方法[34]。其架构如图 4 所示。

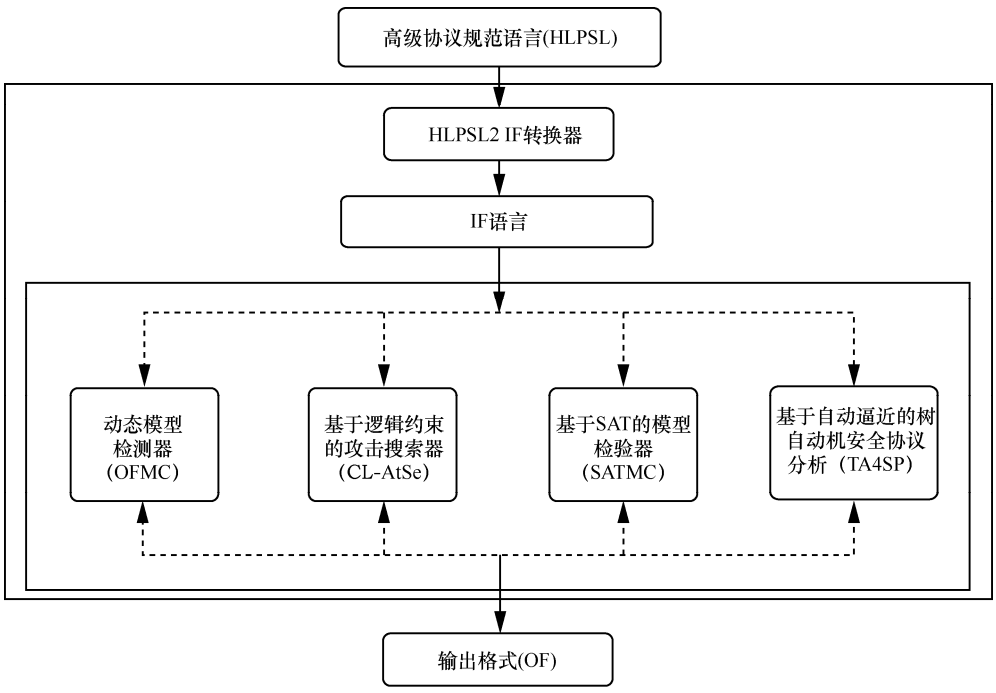

图 4 AVISPA 架构

4.2.2 基本角色

在共识协议中分别为每个参与者定义一个角色，即客户端 c、主节点 N_0、副本节点 N_1, N_2, N_3 的角色，如表 4 所示。

4.2.3 会话场景

本文根据改进的 PBFT 算法定义了 3 种会话场景来验证共识协议是否符合安全目标，如表 5 所示。场景 1 为正常的会话过程，其中包含所有合法的场景；场景 2 中主节点为 sybil 节点；场景 3 中副本节点为 sybil 节点。

4.2.4 安全目标

为了评估改进的 PBFT 算法共识协议的安全性，首先要明确协议需要达到的安全目标。AVISPA 工具中提供了不同的关键字表示安全目标。在本文的安全性测试中，用关键字描述如下。

1) 共识过程中的请求信息 Q 是由客户端 C 产生的，并且将此信息广播给主节点 N_0。

$$\text{secret}(Q, t, \{C, N_0\})$$

其中，t 是安全目标中的标识符。

2) 关键字 request 声明副本节点 $N_{\{1,2,3\}}$收到了主节点 N_0 发送的共识信息 Q，关键字 witness 声明主节点 N_0 向副本节点 $N_{\{1,2,3\}}$发送了共识信息 Q。

$$\text{request}(N_{\{1,2,3\}}, N_0, t, Q)$$
$$\text{witness}(N_0, N_{\{1,2,3\}}, t, Q)$$

为了对共识协议的安全性进行测试，本文定义了如下安全目标。

1) 在共识协议中，主节点 N_0 验证客户端 C 发送的请求信息的有效性，确保请求信息的交易格式和时间戳有效。其他副本节点 $N_{\{1,2,3\}}$需要验证主节点生成的新区块的有效性，验证区块头中的 Merkel 根是否正确，以及区块头是否引用前一区块的哈希值，以此判断主节点是否是 sybil 节点。用 HLSPL 描述如下。

$$N_0\text{: sercet(o, } t, c)$$
$$N_{\{1,2,3\}}\text{: request}(v, n, d, m)$$
$$\text{sercet}(v, n, d, i)$$

2) 在共识协议中，每一轮共识之后各节点之间需要同步共识节点信息表，防止 sybil 节点篡改共识信息，干扰正常的共识，保证各节点的信誉值能够同步进行更新。用 HLSPL 描述如下。

$$N_1\text{: request(CNIL}_0\text{, CNIL}_2\text{, CNIL}_3\text{)}$$
$$N_2\text{: request(CNIL}_0\text{, CNIL}_1\text{, CNIL}_3\text{)}$$
$$N_3\text{: request(CNIL}_0\text{, CNIL}_1\text{, CNIL}_2\text{)}$$

4.2.5 实验结果

在 AVISPA 中使用 OFMC 和 CL-AtSe 分析工具对改进 PBFT 算法的共识协议进行分析，实验结果如图 5 所示。

由图 5 可知，使用 OFMC 和 CL-AtSe 分析组件对改进 PBFT 算法的共识协议的分析结果是安全的，即 SUMMARY: SAFE，并且没有发现协议中存在缺陷。因为如果检测到协议中有缺陷，SUMMARY 字段会显示 UNSAFE，而 DETAILS 字段会提示 ATTACK_FOUND。

在分析过程中，改进 PBFT 算法的共识协议由 HLPSL 的高级协议规范语言转换为 IF 语言形式，保存在 PROTOCOL 字段给出的路径中，其文件名为 hlpslGenFile.if。图 5 中的 BACKEND 字段显示实验所用的分析工具的类型，STATISTICS 字段显

表 4　基本角色定义

基本角色	定义
c	role_ c(c,N_0,N_1,N_2,N_3:agent,Kc,Kn$_0$,Kn$_1$,Kn$_2$,Kn$_3$:public_key,SND,RCV:channel(dy))
N_0	role_N_0(N_0,N_1,N_2,N_3:agent,Kc,Kn$_0$,Kn$_1$,Kn$_2$,Kn$_3$:public_key,CNIL0:text,SND,RCV:channel(dy))
N_1	role_N_1(N_0,N_1,N_2,N_3:agent,Kc,Kn$_0$,Kn$_1$,Kn$_2$,Kn$_3$:public_key,CNIL1:text,SND, RCV:channel(dy))
N_2	role_N_2(N_0,N_1,N_2,N_3:agent,Kc,Kn$_0$,Kn$_1$,Kn$_2$,Kn$_3$:public_key,CNIL2:text,SND, RCV:channel(dy))
N_3	role_N_3(N_0,N_1,N_2,N_3:agent,Kc,Kn$_0$,Kn$_1$,Kn$_2$,Kn$_3$:public_key,CNIL3:text,SND, RCV:channel(dy))

表 5　会话场景定义

场景	定义
场景 1	session(c, N_0, N_1, N_2, N_3, CNIL$_0$, CNIL$_1$, CNIL$_2$, CNIL$_3$)
场景 2	session(c, i, N_1, N_2, N_3, CNIL$_1$, CNIL$_2$, CNIL$_3$)
场景 3	session(c, N_0, i , N_2, N_3, CNIL$_0$, CNIL$_2$, CNIL$_3$)

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL  /home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.40s
visitedNodes: 112 nodes
  depth: 14 plies
```

(a) OFMC 方法

```
SUMMARY
  SAFE
DETAILS  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
/home/span/span/testsuite/results/hlpslGenFile.if
GOAL
  As Specified
BACKEND
  CL-AtSe
STATISTICS
  Analysed  : 0 states
Reachable  : 0 states
  Translation: 0.22 seconds Computation: 0.00 seconds
```

(b) CL-AtSe 方法

图 5　测试结果摘要

示分析工具所运行的时间以及搜索的节点数。

5　改进 PBFT 算法的性能分析与评估

本文实现了一个简单的区块链系统，主要包括共识模块和生成区块模块两部分。其中，共识模块将达成共识的信息发送给生成区块模块，从而生成包含交易信息的区块链。共识模块中分别包含不同的共识算法，即 PBFT 和改进的 PBFT，来验证交易、生成和提交新块。本文实验在平台上模拟不同的共识节点进行共识的过程。实验平台参数如下：Intel Core i5-8265U，2.60 GHz，8 GB RAM。

从 3 个方面来评估改进 PBFT 算法的性能，分别为事务吞吐量、时延和算法的时间复杂度。本文实验设计的共识节点数量分别为 4、7、10、13 和 16，统计的块生成周期分别为 3 s、6 s、9 s、12 s、15 s 和 18 s。当系统运行稳定时，统计区块链系统的 TPS（transaction per second）和生成区块的时延。

5.1　TPS 的比较

在区块链系统中，TPS 是指系统每秒钟能够处理的事务数量。TPS 的大小直接反映了系统处理能力的高低。

$$\mathrm{TPS}=\frac{\text{事务的并发数}}{\text{平均响应时间}} \tag{16}$$

实验中分别设定不同数量的共识节点，不断向区块链系统中发送交易，待系统稳定后，每隔 3 s 统计一次区块链系统处理的交易数量，然后进行计算。图 6 对比了相同的实验平台下，PBFT 算法和改进的 PBFT 算法在不同数量共识节点下的 TPS。

由图 6 可知，当系统运行时间为 12 s 和 18 s 时，系统中的 TPS 基本保持在同一水平。当运行时间为 15 s 时，改进的 PBFT 算法的 TPS 达到最高，平均值约为 3 850，PBFT 算法的 TPS 平均值约为 2 760，改进的 PBFT 算法的 TPS 提高了 40%左右。与 PBFT 算法相比，改进的 PBFT 算法的 TPS 全面提高，并且随着共识节点数量的增多，TPS 下降的速度也在放缓。

5.2　生成区块时延的比较

生成区块的时延 T_{delay} 是指区块从生成到经过共识算法得到节点确认的过程。时延越短表示共识算法的执行时间越短，共识的效率也就越高。区块时延包括共识算法的执行时间 $T_{\mathrm{consensus}}$、信息的广播时间 $T_{\mathrm{broadcast}}$、区块中交易的打包时间 T_{package} 和区块的确认时间 T_{confirm}。

$$T_{\mathrm{delay}}=T_{\mathrm{consensus}}+T_{\mathrm{broadcast}}+T_{\mathrm{package}}+T_{\mathrm{confirm}} \tag{17}$$

区块时延主要由 $T_{\mathrm{consensus}}$ 和 $T_{\mathrm{broadcast}}$ 组成，而

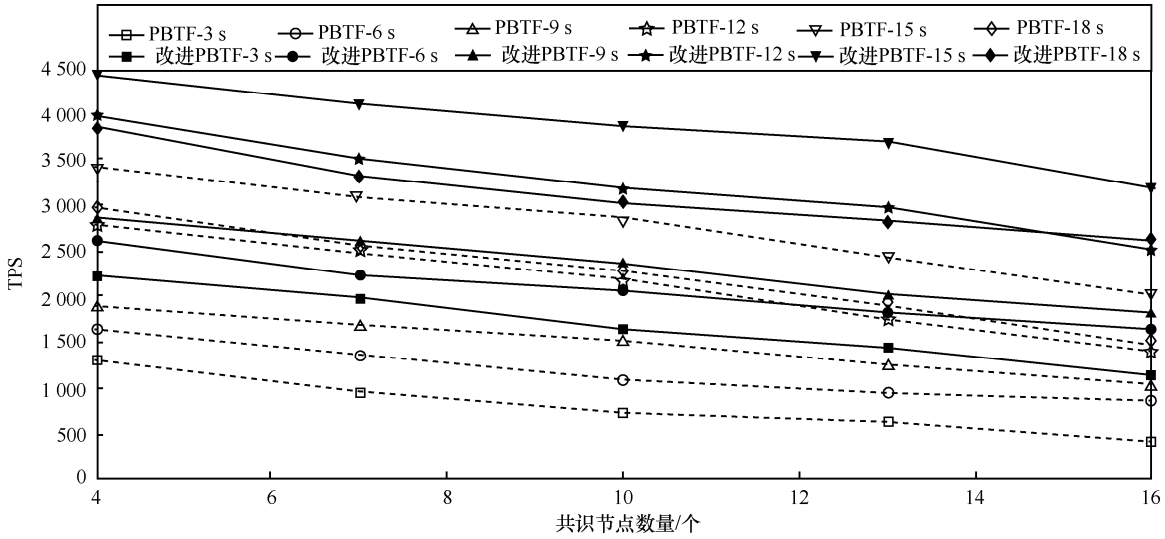

图 6　2 种算法不同数量共识节点的 TPS 对比

T_{package} 和 T_{confirm} 可以忽略。在相同的实验平台下，本节分别统计了 PBFT 算法和改进的 PBFT 算法在不同数量共识节点下生成区块的时延，如图 7 所示。

由图 7 可知，改进的 PBFT 算法中共识节点生成区块的时延比 PBFT 算法整体上减少了 10 ms 左右，并且随着共识节点数量的增多，共识节点生成区块的时延减少的效果更加明显，尤其在运行时间为 15 s 时，生成区块的时延减少最多。以上数据说明改进的 PBFT 算法在减少生成区块的时延方面也有很大的提升。

5.3　时间复杂度的比较

在 PBFT 算法的 3 个主要阶段中，在预准备阶段主节点广播 pre-prepare 消息给其他副本节点，通信次数为 $(n-1)$；在准备阶段，每个节点向其他节点广播 parepare 消息，总的通信次数为 $n(n-1)$；在提交阶段，每个节点向其他节点广播 commit 消息，通信次数也为 $n(n-1)$。PBFT 算法总通信次数为 $(2n^2-n-1)$，时间复杂度为 $O(n^2)$。

改进的 PBFT 算法在节点进行共识的过程中加入了对节点信誉值的计算，并且增加了 pre-commit 阶段，因此改进 PBFT 算法的总通信次数为 (n^2+2n-3)。虽然改进算法的时间复杂度仍维持在 $O(n^2)$，但减少了节点间通信的次数。

6　结束语

本文说明了 sybil 攻击对区块链的危害，分析

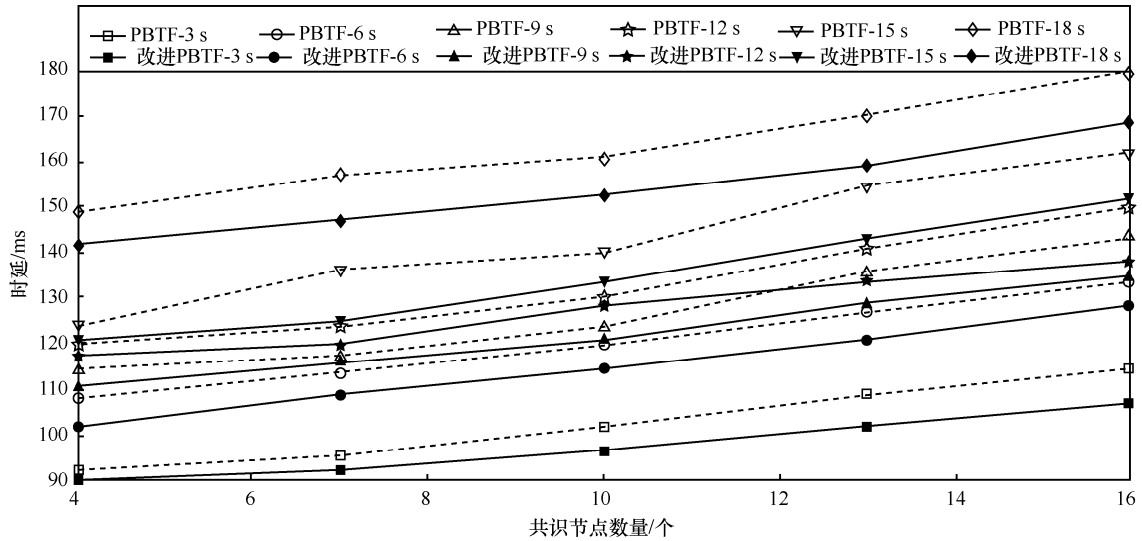

图 7　2 种算法不同数量共识节点生成区块的时延对比

了各种防御 sybil 攻击的方案的不足。本文提出了一种能有效防御区块链中 sybil 攻击的方法，借鉴基于权益证明的共识思想在 PBFT 算法中加入信誉模型，将共识节点的投票权重与所拥有的信誉值相对应，根据共识节点的信誉值为节点分配不同的权重，在达成共识的投票过程中，不同节点拥有不同的投票权重。通过对 PBFT 算法进行改进，根据共识节点的可信状态选择当前信誉值最高的节点作为主节点，同时在共识协议的过程中加入 pre-commit 阶段来减少节点间通信的次数。

通过设计的一个简单区块链系统开展实验，将改进的 PBFT 算法与原 PBFT 算法在 TPS、生成区块的时延和算法复杂度 3 个方面进行了对比，得出的主要结论如下。

1) 改进的 PBFT 算法在系统运行时间为 15 s 时，TPS 提高最多，整体上提高了 40%左右，并且随着共识节点数量的增多，TPS 下降的速度也在放缓。

2) 改进的 PBFT 算法在共识节点数量为 4、统计时间为 15 s 时生成区块的时延减少最多，相比 PBFT 算法生成区块的时延整体上减少了约 10 ms。

3) 改进的 PBFT 算法增加了 pre-commit 阶段，虽然时间复杂度仍维持在 $O(n^2)$，但减少了节点间通信的次数。

本文对改进的 PBFT 算法共识协议的安全性进行了形式化证明，通过理论推导以及实验证明改进的共识协议在通信过程中仍然是安全的。

本文方案不仅可以有效地防御区块链中的 sybil 攻击，而且使区块链的 TPS 和生成区块的时延的性能全面提升，因此本文方案可以满足大多数区块链系统的性能要求，同时可以保证系统的安全和性能的稳定。接下来的工作重点是进一步优化 PBFT 算法，降低 PBFT 算法的时间复杂度，在保证达成共识一致性的前提下减少节点间通信的次数。

参考文献：

[1] 徐蜜雪, 苑超, 王永娟, 等. 拟态区块链——区块链安全解决方案[J]. 软件学报, 2019, 30(6): 1681-1691.
XU M X, YUAN C, WANG Y J, et al. Mimic blockchain—solution to the security of blockchain[J]. Journal of Software, 2019, 30(6): 1681-1691.

[2] DELMOLINO K, ARNETT M, KOSBA A, et al. Step by step towards creating a safe smart contract: lessons and insights from a cryptocurrency lab[C]//Proceedings of International Conference on Financial Cryptography and Data Security. Berlin: Springer, 2016: 79-94.

[3] SAPIRSHTEIN A, SOMPOLINSKY Y, ZOHAR A. Optimal selfish mining strategies in bitcoin[C]//International Conference on Financial Cryptography and Data Security. Berlin: Springer, 2016: 515-532.

[4] ETHAN H, ALISON K, AVIV Z, et al. Eclipse attacks on bitcoin's peer-to-peer network[C]//Proceedings of the 24th USENIX Conference on Security Symposium. Berkeley: USENIX Association, 2015: 129-144.

[5] 祝烈煌, 高峰, 沈蒙, 等. 区块链隐私保护研究综述[J].计算机研究与发展, 2017, 54(10): 2170-2186.
ZHU L H, GAO F, SHEN M, et al. Survey on privacy preserving techniques for blockchain technology[J]. Journal of Computer Research and Development, 2017, 54(10): 2170-2186.

[6] 刘怡然, 柯俊明, 蒋瀚, 等. 基于沙普利值计算的区块链中 PoS 共识机制的改进[J]. 计算机研究与发展, 2018, 55(10): 2208-2218.
LIU Y R, KE J M, JIANG H, et al. Improvement of the PoS consensus mechanism in blockchain based on Shapley value[J]. Journal of Computer Research and Development, 2018, 55(10): 2208-2218.

[7] DOUCEUR J R. The sybil attack[C]//The First International Workshop on Peer-to-Peer Systems. Berlin: Springer, 2002: 251-260.

[8] AL-QURISHI M, AL-RAKHAMI M, ALAMRI A, et al. sybil defense techniques in online social networks: a survey [J]. IEEE Access, 2017, 5: 1200-1219.

[9] SHI L, YU S, LOU W, et al. sybilShield: an agent-aided social network-based sybil defense among multiple communities[C]// Proceedings of 32nd IEEE INFOCOM. Piscataway: IEEE Press, 2013: 1034-1042.

[10] KRISHNAVENI S, KUMAR A V S. A survey on defense mechanism for sybil attacks in large social networks[J]. International Journal of Advanced Research in Computer Science, 2014, 24(12): 2492-2502.

[11] CAO Q, YANG X. Sybilfence: improving social-graph-based sybil defenses with user negative feedback[J]. arXiv Preprint, arXiv: 1304.3819, 2013.

[12] BOSHMAF Y , LOGOTHETIS D , SIGANOS G , et al. Integro: leveraging victim prediction for robust fake account detection in large scale OSNs[J]. Computers & Security, 2016, 61: 142-168.

[13] GAO P, GONG N Z, KULKARNI S, et al. SybilFrame: a defense-in-depth framework for structure-based sybil detection[J]. arXiv Preprint, arXiv:1503.02985, 2015.

[14] MISRA S, TAYEEN A S M, XU W. SybilExposer: an effective scheme to detect sybil communities in online social networks[C]//Proceedings of IEEE International Conference on Communications. Piscataway: IEEE Press, 2016:1-6.

[15] GONG N Z, FRANK M, MITTAL P. SybilBelief: a semi-supervised learning approach for structure-based sybil detection[J]. IEEE Transactions on Information Forensics and Security, 2014, 9(6):976-987.

[16] CASTRO M, LISKOV B. Practical Byzantine fault tolerance[C]// 3rd Symposium on Operating Systems Design and Implementation. Berkeley: USENIX Association, 1999: 173-186.

[17] 刘明达, 陈左宁, 拾以娟, 等. 区块链在数据安全领域的研究进展[J]. 计算机学报, 2020, 43(1): 1-28.
LIU M D, CHEN Z N, SHI Y J, et al. Research progress of blockchain in data security[J]. Chinese Journal of Computers, 2020, 43(1): 1-28.

[18] 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.
WANG H Y, GUO K X, PAN Q Q. Byzantine fault tolerance consensus algorithm based on voting mechanism[J]. Journal of Computer

Applications, 2019, 39(6):1766-1771

[19] WANG F Y, CAI S S, LIN T C , et al. Study of blockchains's consensus mechanism based on credit[J]. IEEE Access, 2019, 7: 10224-10231.

[20] JIANG Y, LIAN Z. High performance and scalable Byzantine fault tolerance[C]//IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference. Piscataway: IEEE Press, 2019: 1195-1202.

[21] MILLER A, XIA Y, CROMAN K, et al. The honey badger of BFT protocols[C]//Proceedings of 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2016: 31-42.

[22] ALEX B, DANIEL F. ReCon: sybil-resistant consensus from reputation[J]. Pervasive and Mobile Computing, 2020, 61: 1574-1192.

[23] 闵新平, 李庆忠, 孔兰菊, 等. 许可链多中心动态共识机制[J].计算机学报, 2018, 41(5): 1005-1020.
MIN X P, LI Q Z, KONG L J, et al. Permissioned blockchain dynamic consensus mechanism based multi-centers[J]. Chinese Journal of Computers, 2018, 41(5): 1005-1020.

[24] ZHANG X, LIU J, LI Y, et al. Blockchain based secure package delivery via ridesharing[C]//2019 11th International Conference on Wireless Communications and Signal Processing. Piscataway: IEEE Press, 2019: 1-6.

[25] JANBI N F, RADENKOVIC M. An enhanced Bayesian-based reputation system for P2P file sharing[C]//Proceedings of 2017 Computing Conference. Piscataway: IEEE Press, 2017:1247-1252.

[26] SARAH A, HEBA K, RASHA A, et al. Authenticpeer++: a trust management system for P2P networks[C]//Proceedings of 11th UKSIM-AMSS European Modelling Symposium on Computer Modeling and Simulation. Piscataway: IEEE Press, 2017: 191-196.

[27] GUPTA M, JUDGE P, AMMAR M. A reputation system for peer-to-peer networks[C]//Proceedings of International Workshop on Network & Operating Systems Support for Digital Audio & Video. New York: ACM Press, 2003: 144-152.

[28] 黄建华, 夏旭, 李忠诚, 等. 基于动态授权的信任度证明机制[J].软件学报, 2019,30(9): 2593-2607.
HUANG J H, XIA X, LI Z C, et al. Proof of trust: a new mechanism of trust degree based on dynamic authorization[J]. Journal of Software, 2019, 30(9): 2593-2607.

[29] 刘庆华, 周小燕. 安全协议的形式化分析方法[J].光盘技术, 2008(3):32-33.
LIU Q H, ZHOU X Y. The discusses of the formal methods for security protocol verification[J].CD Technology, 2008(3):32-33.

[30] 项俊龙, 陈传峰. 安全协议形式化验证方法综述[J]. 信息安全与通信保密, 2013(5): 52-54.
XIANG J L, CHEN C F. Overview on security protocol formal verification methods[J]. Information Security and Communications Privacy, 2013(5): 52-54.

[31] 海沫, 朱建明. 区块链网络最优传播路径和激励相结合的传播机制[J].计算机研究与发展, 2019, 56(6): 1205-1218.
HAI M, ZHU J M. A propagation mechanism combining an optimal propagation path and incentive in blockchain networks[J]. Journal of Computer Research and Development, 2019, 56(6): 1205-1218.

[32] LIN Z, LI D, HUANG W. Current security management & ethical[J]. Issues of Information Technology, 2003, PP: 249-266.

[33] 赖英旭, 刘岩, 刘静. 一种网络间可信连接协议[J].软件学报, 2019, 30(12): 3730-3749.
LAI Y X, LIU Y, LIU J. Trusted connection protocol between networks[J]. Journal of Software, 2019, 30(12): 3730-3749

[34] LUCA V. Automated security protocol analysis with the AVISPA tool[J]. Electronic Notes in Theoretical Computer Science. 2006, 155: 61-86.

[作者简介]

赖英旭（1973– ），女，辽宁抚顺人，博士，北京工业大学教授，主要研究方向为工业控制网络安全和软件定义网络安全等。

薄尊旭（1993– ），男，山东东营人，北京工业大学硕士生，主要研究方向为信息安全、区块链共识算法等。

刘静（1978– ），女，北京人，博士，北京工业大学助理研究员，主要研究方向为工业互联网安全、可信计算等。