

基于 Gossip 协议的信任收集共识算法研究

张奇文 王志强 张逸谦

深圳大学计算机与软件学院 深圳 518060



摘要 共识算法是构筑区块链信任特性的基础。如何保证共识算法的高效和稳定一直是研究领域的热点。Gossip 协议因其高效性和可扩展性,被广泛用作共识算法底层框架。传统 Gossip 协议节点之间的通信方式呈随机性,使得共识时间稳定性不够,并且由于不能预测共识时间,无法应用在强一致性场合中。为解决 Gossip 协议中稳定性不够和最终共识的问题,提出一种基于 Gossip 协议的信任收集共识算法。节点通过评估邻近节点的信息度选择通信节点,消息在通信过程中收集信任值,直至消息所收集的信任值大于全网临界受信阈值时,认为消息确认为达成共识。同时,利用时间退化因子控制节点信息度,防止过热产生,维持网络负载均衡。实验表明,CCG 算法与传统 Gossip 和 Random Gossip 算法相比,具有高稳定性、高效率等优点。

关键词: 共识机制;Gossip 协议;节点信息度;信任收集

中图分类号 TP302.8

Trust Collection Consensus Algorithm Based on Gossip Protocol

ZHANG Qi-wen, WANG Zhi-qiang and ZHANG Yi-qian

School of Computer and Software, Shenzhen University, Shenzhen 518060, China

Abstract The consensus algorithm is the basis for constructing the trust characteristics of the blockchain. How to ensure its efficiency and stability has been a hot topic in the research field. The Gossip protocol is widely used as the underlying framework of consensus algorithms because of its efficiency and scalability. However, the communication methods between the traditional Gossip protocol nodes are random, which makes the stability of consensus time insufficient, and because the consensus time cannot be predicted, it cannot be applied in occasions with strong consistency. In order to solve the problem of insufficient stability and final consensus in Gossip protocol, a trust collection consensus algorithm based on Gossip protocol is proposed. The node selects the communication node by evaluating the information degree of the neighboring node, and the message collects the trust value in the communication process, the message is not considered to be in consensus until the threshold is greater than the critical threshold of the whole network. At the same time, the time degradation factor is used to control the node information degree, to prevent the occurrence of hot spots and maintain network load balancing. Experiments show that the CCG algorithm has the advantages of high stability and efficiency compared with the traditional and Random Gossip algorithms.

Keywords Consensus mechanism, Gossip protocol, Node information degree, Information collection

1 引言

共识机制作为区块链的关键技术之一,作用是规范区块链账户的权益归属和维持账本数据的一致性。共识机制本质是分布式一致性问题,但两者的关注点不同。分布式一致性问题关注的是如何在节点失效时达成共识;而共识机制则关注系统中信任达成以及恶意攻击等问题。

共识算法可分为概率一致性算法和绝对一致性算法。概率一致性算法是一种允许数据在某个时间点存在不一致的共识算法。常见的概率一致性算法包括工作量证明算法(PoW)、权益证明算法(PoS)和委托权益证明算法(DPoS),这3种算法可以较好地适用于实际商用场景,但有高能耗、共识速率慢和依赖代币的缺陷^[1-2]。Rajendra 等通过在各种难度下使用不同的共识算法,缩短了区块创建时长^[3],虽然算法获

得了较高的吞吐量,却增加了集中运算的负担。Kiayias 等提出权益证明并拥有安全保证的区块链协议,其相较于物理资源证明的区块链协议效率更高^[4],但由于权益过大使得网络流动性降低。绝对一致性算法是通过牺牲部分可用性来保持数据绝对一致的共识算法。拜占庭容错算法是一种典型的绝对一致性算法,它能够保证总节点数为 $3f+1$ 的情况下容忍最多 f 个拜占庭节点^[5-6],但是该算法效率较慢且不具有动态扩展的特性。Gossip 算法主要应用在分布式数据库系统中的节点之间的数据同步,由于其高效、扩展性强,可以增加拜占庭容错算法的动态扩展性^[7-8]。Boyd 等提出了随机式 Gossip 算法,其平均共识时间取决于双随机矩阵的第二大特征值^[9],加快了共识效率,但是具有消息延迟和冗余的缺点。Loizou 等将随机 Kaczmarz 算法与 Gossip 算法结合,解决了 Gossip 算法的加速和近似问题^[10-11],但无法解决 Gossip 算法最终一

本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家科技支撑计划(2014BAH28F05)

This work was supported by the National Science and Technology Support Program Funding Project (2014BAH28F05).

通信作者:张奇文(18688789334@163.com)

致性的问题。Silvestre 等通过交换节点间局部估计值来得到不保守集值状态估计,在非故障环境中实现有限时间共识^[12],但无法保证数据的强一致性。综上所述,Gossip 算法具有传播速度快、共识效率高特点,但存在消息延迟和冗余,将带来最终一致性和资源浪费等问题,使其无法较好地在实际应用场景中运行。

本文提出基于 Gossip 协议的信任收集共识算法 CCG (Confidence Collection Gossip),以节点信息度作为参考选择传播节点,设置信任阈值作为共识结束标志,同时对节点信息度设置时间退化,防止网络热点出现,保证网络负载均衡。消息在传播过程中收集信任值,直到消息所收集的信任值达到信任阈值时确认为达成共识。阈值设定使最终共识时间可以被估测,解决了 Gossip 算法最终一致性的缺陷。实验表明,本文提出的 CCG 算法具有共识速度快以及稳定性强等优势。

2 通信模型

2.1 通信模式

Gossip 协议是某节点与其邻近节点进行数据交换时数据更新的一种通信方式,利用节点本地计算资源可以有效减少网络中传输的数据量,达到节能和减少带宽的作用。Gossip 协议是一种去中心化的通信方式,各节点只需维护自身邻居节点视图,而不用保存全局网络的信息。原始 Gossip 算法的交互模式有以下 3 种。

(1) Pull 模式:节点 A 将信息包 $Gossip\langle Version, Key, Value \rangle$ 发送给节点 B,节点 B 根据信息包中的内容同步新数据,保留旧数据。

(2) Push 模式:节点 A 仅发送 $Gossip\langle Version, Key \rangle$ 给节点 B,节点 B 根据版本号,将本地比 A 版本号高的数据打包发送给 A,节点 A 接收并更新本地数据。

(3) Pull/Push 模式:上述两种模式的结合。

传统 Gossip 协议是在一个有界网络中运行,每个节点随机与其他节点发生通信,最终所有节点的状态达成一致。因为节点选择的随机性,所以节点间可能发生重复通信,使得算法冗余度较大。因此,对节点的通信模式进行如下改进:

(1) 节点选择。通过选择概率矩阵 $V_i = [v_1, v_2, \dots, v_n]$ 选择传播节点,节点概率矩阵由邻居节点决定。

(2) 消息传播。在传播消息过程中,根据所经过的节点属性收集信任值。当信任值超过阈值时,达成共识。

2.2 节点信息度

节点信息度是衡量节点信息量的一个评价指标。节点信息度越大,表示节点的信息量越多,与其发生通信时获得的收益越大,这种收益体现在消息传播的更快、更有效。设复杂网络图 $G = \{S, L\}$ 是一个无向连通图, S 表示网络中节点的集合, L 表示节点之间边的集合。节点 t 周围邻居节点数记为 S_t ,单位时间内发生的信息交互次数记为 K_t ,节点 t 的单位信息价值记为 φ_t ,则邻居节点的信息价值可以用集合表示为 $\{\varphi_1, \varphi_2, \dots, \varphi_n\}$,节点 t 的单位信息价值的计算公式为:

$$\varphi_t = \frac{S_t K_t}{S_t + K_t} \quad (1)$$

根据空间的自相关性理论,认为网络内相邻的节点之间具有潜在相关性,因此引入多阶加权的方式来计算节点信息度^[13],节点信息度与其相邻 m 阶邻居节点有关,节点 t 的信息度函数 $I(t)$ 为:

$$I(t) = \alpha \varphi_t + \beta \sum_{i \in \omega^1} \varphi_i + \beta^2 \sum_{i \in \omega^2} \varphi_i + \dots + \beta^m \sum_{i \in \omega^m} \varphi_i \quad (2)$$

其中, m 阶邻居节点集合记为 ω^m , α 和 β 表示 m 阶邻居节点的权重调节系数,满足 $k \cdot \beta > \alpha > \beta$ 且 $1 > \beta > 0$,其中 k 为节点平均度。在网络中,邻近节点的选择影响共识时间的长短,导向性传播方式有助于提高传播效率并加快共识速度。

2.3 时间退化因子

理论上,节点信息度的增长是无限制的。当某一节点的信息度过高时,与该节点相关的通信节点会不断增多,引发过热现象,造成通信延时和功耗增大。因此,引入时间退化因子对节点信息度进行抑制,防止过热节点产生。同时,合理的时间退化因子可以引导网络资源分配,即信息度高的节点经过抑制而无法一直参与通信,低信息度节点则有更多机会参与共识过程。

时间退化因子主要由节点的邻居节点数和单位时间内信息交互次数决定。其中,邻居节点数越多的节点拥有的信息度越高,节点退化速率越高。单位时间信息交互次数越多的节点,信息交互效率越高,节点退化速率越低。这一设置是为了激励节点在网络内的活跃性,鼓励节点之间积极通信。因此,时间退化因子的计算公式为:

$$\gamma_t = \epsilon \frac{S_t}{K_t} - \epsilon^2 \quad (3)$$

其中, ϵ 是退化速率控制系数。

综合式(2)和式(3)可以得出时间退化后的节点信息度计算公式为:

$$I'(t) = I(t) - \gamma_t \quad (4)$$

通过实时调整退化速率系数,可以保证节点信息度维持在一个合理范围,在激励节点参与通信的同时,保证节点不会过多地承担通信压力。

3 CCG 算法

CCG 算法可分为两个阶段:邻近节点的选择阶段和信任收集阶段。CCG 算法的流程如图 1 所示。

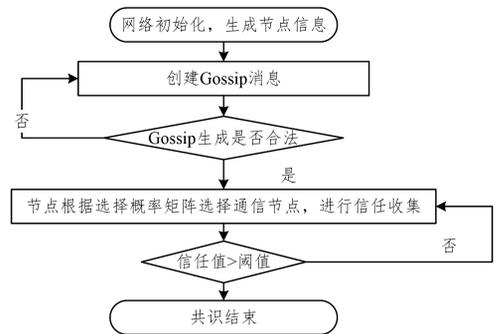


图 1 CCG 算法的流程图

Fig. 1 Flow chart of CCG algorithm

在网络初始化后,节点生成 Gossip 消息并判断消息是否合法,然后节点根据选择概率矩阵选择一个邻近节点发送消息。在传播消息过程中,收集经过节点的信任值,当其超过阈值时,共识完成。

3.1 邻近节点的选择阶段

设邻近节点信息度集合为 $\{I_1, I_2, \dots, I_n\}$, $I_N = \sum_{i \in n} I_i$,可以得到节点 i 的选择概率为 $\frac{I_i}{I_N}$,则节点选择邻近节点的概率矩阵为 $\left[\frac{I_1}{I_N}, \frac{I_2}{I_N}, \dots, \frac{I_n}{I_N} \right]$,其过程如图 2 所示。

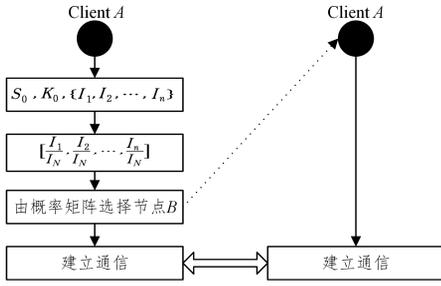


图2 通过选择概率矩阵选择通信节点

Fig. 2 Select communication node by selecting a probability matrix

选择通信节点的步骤如下:

- 1) Client A 初始化 S_0, K_0 以及周围邻居节点的信息度集合为 $\{I_1, I_2, \dots, I_n\}$;
- 2) 根据信息度集合 $\{I_1, I_2, \dots, I_n\}$ 计算选择概率矩阵 $V_i = [v_1, v_2, \dots, v_n]$;
- 3) Client A 以 v_B 概率选择到节点 B 建立通信。

3.2 信任收集阶段

信任收集机制的作用是判定 CCG 算法何时终止。在传播过程中, Gossip 消息每经过一个节点就获得一部分信任值, 这些信任值的大小由节点信息度决定。经过节点的信息度越高, 获得信任值越多。假设消息的实时信任值为 C , 传输经过节点 A (节点 A 的信息度为 I_A), 其信任增加值为:

$$\Delta C = I_A + \delta \sum_{i \in \omega^1} I_i \quad (5)$$

其中, δ 是信息收集系数, 用以调节信任收集速率。信任收集过程如图 3 所示, 消息由 A 点发出, 通过邻近节点选择算法, 由低信任节点向高信任节点传播, 直至消息收集的信任值达到阈值。信任收集机制解决了 Gossip 协议最终一致性的缺点, 使算法的共识时间可以被估测。传统 Gossip 协议中, 消息如同谣言一个接一个地传递至整个网络, 但是无法准确估测出消息传遍整个网络的具体时间点。通过信任收集机制根据所收集信任值的大小以及全网信息度信息, 便可以计算出该条消息达成共识所需经过的节点数, 进而估测最终的共识时间。

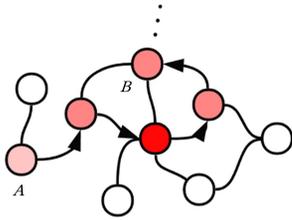


图3 信任收集机制

Fig. 3 Trust collection mechanism

3.3 算法分析

Gossip 算法的时间收敛性已经得到证明^[14-15], 因此本文仅对 CCG 算法的共识结束标志作论证分析。假设全网平均节点信息度为 \bar{I} , 全网信任值为 I_{ALL} , 全网总节点数为 N , Gossip 消息经过 n 个节点完成信任收集。设已收集的信任值为 $C_s = \sum_{i=1}^n C_i$:

$$C_s > \mu I_{ALL} \quad (6)$$

其中, C_i 为每一次传播所收集的信任值。

当满足式(6)时, 认为 Gossip 达成共识。已知要满足共识时间是可预测的, 则传播过程中消息经过最多有限个节点

即可确认达成共识。即:

$$\begin{aligned} \sum_{i=1}^n C_i &= \sum_{i=1}^n (I_i + \delta \sum_{j \in \omega^1} I_j) \\ &= \sum_{i=1}^n I_i + \delta \sum_{i=1}^n \sum_{j \in \omega^1} I_{i,j} \\ &= n \bar{I} + \delta \sum_{i=1}^n S_i \bar{I} \\ &= \bar{I} (n + \delta \sum_{i=1}^n S_i) \geq I_{ALL} = \mu N \bar{I} \end{aligned}$$

当经过的节点数满足 $n \geq \mu N - \delta \sum_{i=1}^n S_i$ 时, 可以预测该消息信任收集完成并达成共识。因为 μ 和 N 是常数, 节点的邻近节点数至少为 1, 在 $\sum_{i=1}^n S_i = n$ 时, Gossip 共识时间最长, 此时 $n \leq \frac{\mu N}{1 + \delta}$, 因此经过的节点数最多为 $\frac{\mu N}{1 + \delta}$, 证明 CCG 算法的共识时间是可以估测的。

4 实验结果与分析

4.1 实验平台及参数配置

实验部署在多台计算机上模拟网络通信, 以多线程方式模拟节点运行。在不考虑网络带宽波动和延时的情况下, 搭建原型算法模型, 将 CCG 算法与传统 Gossip 算法、Random Gossip 算法进行比较^[17-18], 通过对比最终共识时间以及共识时长波动来判断 3 种算法的效率和稳定性。

网络连接通过随机矩阵生成, 每个节点通过初始化的随机矩阵得到自己的邻近视图; 每个节点可以自由进入或者退出当前网络。采用的参数值如表 1 所列。

表1 参数设置

Table 1 Parameter settings

Parameter	Weight Reference Value
α	$0.8 (\beta < \alpha < \bar{k} \cdot \beta)$
β	$0.5 (0 < \beta < 1)$
δ	$0.3 (0 < \delta < 1)$
μ	$0.5 (0 < \mu < 1)$

注: α 和 β 是邻近节点权重调节系数, δ 是信息收集系数, 阈值 μ 是全网临界受信值

4.2 实验分析

实验主要从两个场景对算法性能进行对比。第一个场景是网络节点数增多时, 算法共识时长的对比; 第二个场景是节点数为 1000 时, 算法共识时长的对比。

(1) 假设每个节点在一个时钟周期内只能选择一个节点进行通信。由图 4 可以看出, 在不考虑网络延迟的情况下, 网络节点数由 1 增加到 1000 时, 3 种算法的共识时长都处于增长的趋势, 但是 CCG 算法在节点数不断增多时, 共识时间仍然保持较低水准, 波动更为稳定。

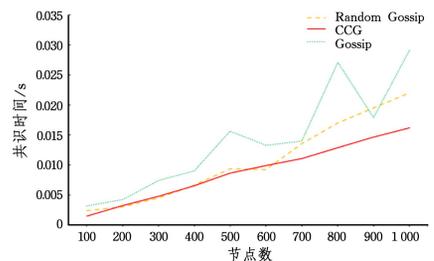


图4 共识时长变化

Fig. 4 Consensus duration change

实验表明, CCG 算法减少了无意义的通信, 加快了共识

时间,同时减少了网络资源的消耗。信任阈值的设定越大,Gossip消息共识过程越长,经过节点数越多。通过改变信任阈值,可以调节共识的安全程度和共识速度,表现为阈值设定越大,网络越安全,共识时间越长。

(2)假设网络节点数相同,网络节点数为1000,通过100次共识,观察3种算法共识时长的稳定性(每次实验中节点的邻近节点视图均一致)。如图5所示,原始Gossip和Random Gossip都表现出更长的共识时间。Random Gossip算法作为一种随机通信方法,其最终共识时长的波动幅度较大^[16]。这种随机通信方式使最终共识时间难以预测,无法应用在强一致性场景中。CCG算法表现最为稳定,共识时长波动较小。如表2所列,CCG算法相比于另外两种算法,平均共识时间更短,稳定性能更高。

表2 共识稳定性指标

Table 2 Consensus stability indicator
(单位:s)

	St. d	Var	Ave
CCG	0.121	0.0147	1.45
Random Gossip	0.154	0.0237	1.63
Gossip	0.471	0.2218	2.01

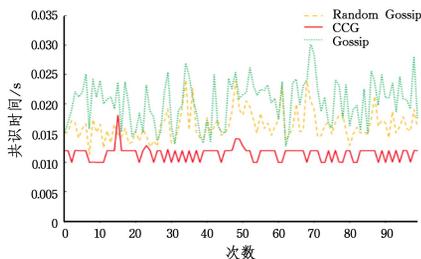


图5 稳定性变化

Fig. 5 Stability change

结束语 本文针对传统Gossip算法的低稳定性和最终一致性的问题,提出了通过信任收集的Gossip共识算法。该算法优化了通信过程中的节点选择方式,并利用消息的信任收集量作为共识标志,保证了共识过程的稳定性以及共识时间的可预测性。在节点数相同时,CCG算法的共识效率更高;在节点数增加时,CCG算法的稳定性能要优于传统Gossip算法。该算法拓宽了实际场景的应用范围,但增加了集中运算。因此,如何减少集中运算的负担将是下一步的研究方向。

参考文献

[1] BENTOV I, LEE C, MIZRAHI A, et al. Proof of Activity; Extending Bitcoin's Proof of Work via Proof of Stake [J]. *Acm Sigmetrics Performance Evaluation Review*, 2014, 42(3): 34-37.

[2] LI W, ANDREINA S, BOHLI J M, et al. Securing proof-of-stake blockchain protocols [M] // *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, Cham, 2017: 297-315.

[3] RAJENDRA S A, JIM B. Optimal block time for proof of work blockchains [C] // *Twenty-Sixth European Conference on Information Systems*. 2018.

[4] KIAYIAS A, RUSSELL A, DAVID B, et al. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol [C] // *Annual International Cryptology Conference*. Springer, Cham, 2017.

[5] LAMPORT L, SHOSTAK R, PEASE R. The Byzantine Generals Problem [J]. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 1982, 4(3): 382-401.

[6] CASTRO M, LISKOV B. Practical Byzantine fault tolerance [C] // *OSDI*. 1999: 173-186.

[7] LIU D H, YIN G, WANG H M, et al. Overview of Gossip Algorithm in Distributed Environment [J]. *Computer Science*, 2010, 37(11): 24-28.

[8] ZHANG S J, CHAI J, CHEN Z H, et al. Byzantine Consensus Algorithm Based on Gossip Protocol [J]. *Computer Science*, 2018, 45(2): 20-24.

[9] BOYD S, GHOSH A, PRABHAKAR B, et al. Randomized gossip algorithms [J]. *IEEE Transactions on Information Theory*, 2006, 52(6): 2508-2530.

[10] LEE S, NEDIĆ A. Asynchronous Gossip-Based Random Projection Algorithms Over Networks [J]. *IEEE Transactions on Automatic Control*, 2013, 61(4): 953-968.

[11] LOIZOU N, RABBAT M, RICHTÁRIK P. Provably accelerated randomized gossip algorithms [C] // *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019: 7505-7509.

[12] SILVESTRE D, ROSA P, HESPANHA J P, et al. Stochastic and deterministic fault detection for randomized gossip algorithms [J]. *Automatica*, 2017, 78(Complete): 46-60.

[13] AYSAL T C, YILDIZ M E, SARWATE A D, et al. Broadcast Gossip Algorithms for Consensus [J]. *IEEE Transactions on Signal Processing*, 2009, 57(7): 2748-2761.

[14] USTEBAY D, ORESHKIN B N, COATES M J, et al. Greedy Gossip With Eavesdropping [J]. *IEEE Transactions on Signal Processing*, 2010, 58(7): 3765-3776.

[15] SARWATE A D, DIMAKIS A G. The Impact of Mobility on Gossip Algorithms [J]. *IEEE Transactions on Information Theory*, 2012, 58(3): 1731-1742.

[16] NEWPORT C, WEAVER A. Random Gossip Processes in Smartphone Peer-to-Peer Networks [J]. *arXiv:1902.02763*, 2019.

[17] HANZELY F, KONEČNÝ J, LOIZOU N, et al. Privacy preserving randomized gossip algorithms [J]. *arXiv:1706.07636*, 2017.

[18] TUNCER C, AYSAL M E, YILDIZ A D, et al. Broadcast gossip algorithms: Design and analysis for consensus [C] // *2008 47th IEEE Conference on Decision and Control*. IEEE, 2009.



ZHANG Qi-wen, born in 1993, master student. His main research interests include blockchain and education big data.