NE-Gossip:基于邻居节点评估机制的Gossip广播算法

黄春霖,何 晋,周 迅

(成都信息工程大学 区块链产业学院,四川 成都 610225)

摘 要: 与传统中心化架构的系统相比,去中心化的区块链网络吞吐量仍处于较低水平。为此,许多研究从共识机制、分片机制等对区块链的可扩展性进行了探索。然而,随着吞吐量的提升,区块链系统对底层消息广播网络效率的要求也在不断提高。由此,着眼于底层P2P消息广播网络,在Gossip的算法中引入邻居节点评估机制,提出NE-Gossip广播算法,根据评估结果在消息转发时选择具有更好消息转发能力的节点。实验结果表明,在同样的广播冗余度下,NE-Gossip算法在广播覆盖率上具有更好的表现。

关键词: 广播算法;P2P网络;Gossip;转发节点选择;区块链

DOI: 10.11907/rjdk.232159 **开放科学(资源服务)标识码(OSID):**

中图分类号: TP311 **文献标识码:** A **文章编号:** 1672-7800(2025)001-0078-08

NE-Gossip: Gossip Broadcasting Algorithm with Neighbor Evaluation Mechanism

HUANG Chunlin, HE Jin, ZHOU Xun

(*College of Blockchain Industry, Chengdu University of Information Technology, Chengdu* 610225, *China*)

Abstract: Compared to traditional centralized systems, the throughput of decentralized blockchain networks remains at a relatively low level. As a result, many studies have explored the scalability of blockchain through consensus mechanisms, sharding, and more. However, as throughput increases, the requirements for the efficiency of the underlying message broadcasting network in blockchain systems continue to rise. This study focuses on the underlying P2P message broadcasting network and introduces a neighbor evaluation mechanism into the Gossip algorithm, proposing the NE-Gossip broadcasting algorithm. Based on the evaluation results, NE-Gossip selects nodes with better message forwarding capabilities during message relay. Experimental results indicate that under the same broadcast redundancy, the NE-Gossip algorithm demonstrates better performance in terms of broadcast coverage rate.

Key Words: broadcast algorithm; P2P network; Gossip; relay node selection; blockchain

0 引言

区块链技术的核心是解决去中心化场景下的信任问题,区块链是一种能够在复杂环境下建立信任的支撑技术[1]。网络中的节点通过验证交易和区块的合法性、独立性,根据最长链原则处理分叉即可保证自己的账本状态和网络中诚实节点的账本状态趋于一致。区块链的概念起源于比特币,比特币首次实现了去中心化的货币交易系统[2]。以太坊利用区块链技术实现了通用应用的去中心化,用户可以通过在以太坊上部署智能合约以构建分布式服务[3]。

总体而言,区块链网络是一个分布式的概率状态机[4]。对于状态机,确定的指令序列与确定的状态可以得到确定的输出。区块链中的交易可以看成输入指令,而每个区块则是一系列指令的有序集合。每个区块都有父区块哈希,这保证了区块链中区块的有序性。因此,拥有同样区块链表的节点有着相同状态。在该状态机之上是具

收稿日期: 2024-01-26

扫描二维码阅读全文:

基金项目: 科技部创新方法工作专项(2017IM030100)

作者简介: 黄春霖(1996-),男,成都信息工程大学区块链产业学院硕士研究生,研究方向为区块链、P2P网络;何晋(1978-),男,成都信息工程大学区块链产业学院教授、硕士生导师,研究方向为智能治理、共识机制;周迅(1994-),男,成都信息工程大学区块链产业学院系统架构师,研究方向为分布式存储。本文通讯作者:何晋。

体应用，它可以是一个如同比特币的现金系统，也可以是像以太坊那样任意形式的去中心化平台及应用。当一个区块被某节点接收确认后，区块中的交易被执行，该节点则由当前状态迁移到下一个状态。在比特币网络中，为了获得出块奖励，每个节点都在竞争打包下一个区块的资格，在一个广播时延内存在多个节点在同一区块高度找到下一个区块的客观情况。当在同一区块高度接收到多个合法的区块时，如何保证网络中的绝大多数节点选择同样的区块是共识算法所要解决的问题。然而，无论采用什么共识算法，保证其能正确工作的前提都是能够在一个合理的时间内通过底层的P2P广播网络获得交易与区块数据。P2P广播网络为节点收集和广播交易与数据、管理邻居节点列表，并保证网络的连通性。因此，区块链系统大致可以分为3层：应用层、共识层和P2P广播网络层。

如何提升区块链系统的吞吐量是区块链领域中备受关注的研究热点。单纯地提高区块的容量或是增加出块频率并不是理想方案，如果仅仅增加区块的容量，这会使区块的发送时延增加，而更大的区块广播时延意味着更高的分叉率[5]。在不降低区块广播时延的前提下提高出块频率同样也会导致更高的分叉率[5]。提高区块容量或是增加出块频率对于区块链的吞吐量提升有限，要想取得重大进展，必须从根本上重新考虑技术方法[6]。许多研究从不同角度对区块链系统进行了创新和优化，比如采用了新的区块组织结构[7-9]或实现了账本状态分片[10-11]，以实现更高的吞吐量，其中Conflux[7]和OHIE[9]可以实现5 000 TPS的吞吐量。然而，实现高TPS还需要对底层P2P网络提出更高的要求，高效的交易传播对于区块链系统保持高吞吐量至关重要[12]。同时，过量的带宽占用是比特币社区试图解决的首要网络问题之一[13]。区块链网络是P2P网络的新型通信范式，对区块链应用和系统的安全可信与性能至关重要[14]。因此，用于数据广播的P2P广播网络层值得高度重视和深入研究。

相较于隐私保护、共识算法等区块链领域的热门研究方向，国内外研究者对P2P广播网络层的关注较少。在区块链P2P广播网络层相关研究中，大多从网络拓扑结构、传输数据角度进行探索与改进。这些研究中，在节点接收数据进行转发的过程中，转发节点的选择是随机的，这是因为无法或难以对路由表中的节点进行转发能力上的区分。本文在Gossip的算法中引入邻居节点评估机制，使得网络节点能够区分其邻居节点之间的优劣，并提出NE-Gossip广播算法，为P2P广播算法研究提供一种新的思路。

1　相关工作

1.1　区块链网络中基于洪泛法的广播算法

如图1所示，在比特币网络中的两个节点间传输消息时，发送方在验证消息的正确性后，为了避免向接收方发送一个重复的消息，需要先向接收方发送一个inv消息。inv消息是一组交易或区块数据的哈希，接收方如果没有接收过对应的数据，会通过getdata消息向发送方说明需要的数据信息，发送方再传输完整数据。

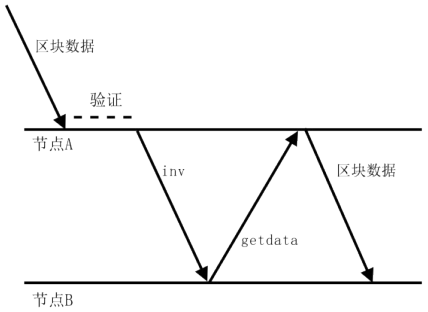


Fig. 1　A message transmission process between two nodes in the Bitcoin network

图1　比特币网络中节点间的一次消息传输流程

比特币网络中的节点在收到消息后，会转发给除发送方外的所有节点，通过这种方式可以将消息广播到全网[15]。为了使数据能够更快地让大部分节点接收到数据本身，假设一个节点有n个邻居节点，以太坊会随机向$\sqrt{n}$个节点发送完整数据，另外$n-\sqrt{n}$个节点发送inv消息。选择不同的节点转发完整数据可能会有不同的效果，因为网络中节点之间存在网络状况差异是客观事实，但以太坊的节点选择是随机进行的。

Gossip算法的工作原理较为直观，网络中的节点每接收到一条新消息时，从自己的邻居节点中随机选择一定数量的后继转发节点，并向它们转发这条消息[16]。因此，Gossip算法通过提高冗余度提升消息到达更多节点的概率。Gossip广播算法的广播覆盖率即使在网络中无任何节点发生异常的情况下也难以确保能达到百分之百，除非采用洪泛法要求每个节点转发给所有的邻居节点。对于每条广播消息，除进行广播初始化的源节点外，其他节点只需要对消息进行一次接收，其余的消息传输均为冗余传输。对于有N个节点的P2P网络，虽然相比于基于洪泛法的广播算法，Gossip只向部分邻居节点进行了消息转发，但消息传输次数仍然是远大于$N-1$。接收消息后的消息转发可以看作是将后续的消息转发任务交给了被选中为转发节点的部分邻居节点。因此，如果消息广播任务由于随机性被分配到了网络状况较差的节点，可能会对最终的消息广播效果产生影响。

1.2　紧缩块协议

区块中的节点在进行区块传输时，会完整地包含区块中所有交易的完整数据，因而区块传输比交易传输会占用更大量的带宽。由于区块被挖出的时间比交易的平均传输时间长得多，接收方很有可能已经收到了区块中的大部分交易，因而区块中这部分交易的完整数据传输并不必要。出于对节省区块数据带宽占用的考虑，紧缩块协议（Compact Block Protocol）被提出[17]。如图2所示，传输紧

缩块时，发送方将一些预估接收方已经拥有的交易通过6字节的短ID替代原始交易数据。在紧缩块中，完整的交易数据被称为预填充交易。

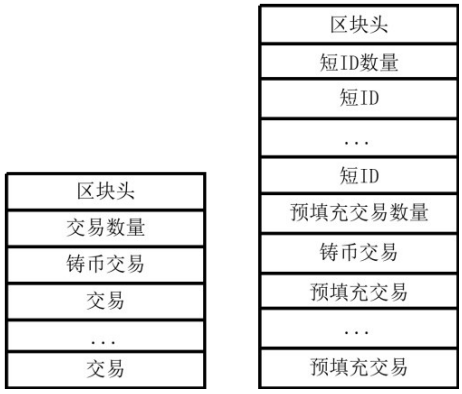


Fig. 2 Bitcoin block message structure and compact block message structure

图2 比特币区块消息结构与紧缩块消息结构

然而，发送方并不能明确地知道接收方接收到了哪些交易数据，因而预填充交易列表中的交易数据仅为发送方的猜测。当接收方缺失短ID对应的数据时，仍然需要与发送方进行额外的数据交互，需要保证较高的交易发送量以减少交易池中交易数据的缺失[13]。

1.3 基于地理位置的广播算法

文献[18]指出，现有的广播拓扑结构由于忽略了节点间的地理位置，存在一些不必要的高时延传输路径，并提出了基于地理位置的P2P覆盖网络FRING。FRING将地理位置临近的节点相互建立邻居关系，再从各区域中选出代表节点再次在区域间建立连接，重复这个过程形成一个递归的多层环状结构。类似地，BlockP2P利用K-Means算法将地理位置临近的全节点进行聚集，再在各集合中选出路由节点相互连接以保证整个网络的连通性，轻节点连接到临近的全节点，由此形成了一个三层结构[19]。然而，无论是FRING的递归环状结构，还是BlockP2P的三层结构，所在层数越高节点的故障对网络带来的影响越严重。

2 NE-Gossip广播算法

相关工作中的方法均未对转发节点选择策略进行研究。邻居节点评估(Neighbor Evaluation，NE)机制是一种动态评估邻居节点转发能力的方法，并根据评估值在进行消息转发时更大概率地选择到具有更强转发能力的节点。我们将这种机制引入Gossip提出NE-Gossip广播算法。

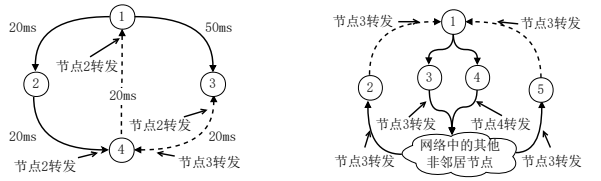
2.1 广播算法中的转发节点选择问题

P2P网络中节点的连接质量和网络条件在客观上存在差异，因而节点在接收到一个新数据后，后继转发节点的不同选择也可能会对广播效率带来不同的影响。对于一个去中心化无结构化的P2P网络，节点对于网络的感知来自且仅来自于与有限数量节点的数据交互。节点从数据交互的过程中能够得到的信息为：①邻居节点的在线情况，可以根据该信息将离线的节点移除或替换掉；②接收到的消息是冗余消息还是新消息，该信息决定了数据应该被忽略还是接收验证并转发；③接收到的消息来自于哪一个邻居节点，该信息决定了应该转发到哪些节点。

理想情况下，希望节点在进行数据转发时能够选择到网络状况更好的节点，以达到更快或更可靠地将消息广播到网络中的目的，但对于邻居节点质量的评估不易。即使能够获取到邻居节点的带宽、链路时延等网络指标，这些指标越高并不代表该节点能够更快地进行消息转发。一方面，由于网络状况可能动态发生变化，节点可能因为数据传输任务较多或计算任务繁忙而导致数据转发能力下降甚至丢包；另一方面，节点网络状况较好也无法保证该节点的其他邻居也具有较好的网络状况，当该节点将消息转发到其邻居节点后，并不能确保其他邻居节点也能及时地转发消息。

2.2 邻居节点评估机制

网络中每个节点都有自己独立的路由表，节点与路由表中记录的节点直接通信，这些节点即邻居节点。邻居关系是逻辑上的，与具体的地理位置无关。在广播源节点的视角下，当它向部分邻居节点初始化一条消息时，如果网络是连通的，它会从其他的部分邻居节点处接收到它初始化的这条消息。将前者记作转发节点，后者记作反馈节点。图3中节点之间有边相连即代表建立了邻居关系，实线代表此次广播中一次有效的数据传输，虚线代表冗余传输，箭头为数据传输的方向。图3(a)的边上标记了传输路径上的时延，源节点1将其中两个邻居节点，即节点2和节点3在本轮广播中选作转发节点，初始化一条广播消息后，会从另外一个邻居节点4，即反馈节点接收到这条消息。在现有的广播算法中，传输回源节点的消息往往被视作为无用的冗余消息被忽略。虽然来自节点4的这条消息是冗余的，但节点1可以通过这条冗余消息了解到节点2或节点3将消息转发到了网络中这个实事，该冗余反馈正是使得NE机制能够实现的关键。



(a) Redundant feedback in a four nodes network

(a) 四节点网络中的冗余反馈

(b) Competition among relay nodes during a broadcast

(b) 转发节点间形成的竞争

Fig. 3 Redundancy feedback phenomenon in networks

图3 网络中的冗余反馈现象

此外，可以观察到从反馈节点4转发到源节点1的消息来自于转发节点2，这体现了相比于转发节点3，节点2

能够更快地将消息转发到节点4。但对于此轮广播中的源节点1，由于发送给转发节点的消息都是一致的，故并无法从接收到的消息中推断出节点2具有更高转发能力的信息。因此，需要在消息中附加能够表明消息转发节点的字段relay。源节点在向不同的转发节点初始化数据时，则需要在relay字段填充不同的信息，比如用自己公钥加密的转发节点的ID。

扩展到具有一定数量的P2P网络中时，反馈节点向源节点的冗余转发可以视作对转发节点的投票，而转发节点之间则形成了竞争反馈节点选票的局面。在图3(b)中，由转发节点3转发的消息在网络中传播得更快，比节点4转发的消息更快地到达了反馈节点2和5。由于节点2和5无法获知节点1是否接收过这条消息，因而对节点1分别做了一次冗余转发。节点1在接收到这两条冗余消息后，便能推断出此轮广播中节点3及其后续的网络具有更好的转发能力。由于节点间邻居关系的建立是随机的，当节点具有一定数量的邻居节点后，反馈节点对转发节点的投票能够较好地反应各转发节点及其后继网络的消息转发能力。

当广播消息中的relay字段对于源节点而言变得可识别后，源节点对邻居节点转发能力的评估成为可能。需要说明的是，反馈节点无法感知自己的身份，反馈节点只是和网络中的其他节点一样进行了一次转发流程。并且，转发节点同样也无法感知自己的身份，因为relay字段仅源节点能识别，这避免了引入新字段后为广播算法带来额外的安全问题。

2.2.1　邻居节点转发能力评估

节点在进行消息转发时，能够选择到转发能力更好的后继节点的前提是节点的转发能力能够用相互可比较的数值加以体现，因而需要对节点转发能力进行定量。当某邻居节点相较于其他邻居节点对节点与网络的交互起到正向作用时，该邻居节点转发能力的评估数值应该得到提升。增加评估数值的行为包括：①向该节点转发了一条从未收到过的消息；②行使了一次反馈节点的职责，即向源节点冗余转发了一条由该节点广播的消息；③作为转发节点转发到网络中的消息抢先到达了某些反馈节点，并被反馈节点冗余转发给了该源节点。

$$S = n_m \times S_m + n_f \times S_f + n_r \times S_r \tag{1}$$

某邻居节点总的评估数值即3个事件发生的次数乘上发生一次应该增加的分值，其中m、f、r分别为行为①、②、③对应的下标。当识别到某邻居节点离线时，它在转发表中的信息将被丢弃，这也意味着其积累的分数也会丢失。各行为增加多少数值可以作为系统参数在具体实现时进行设定。

2.2.2　转发节点的选择策略与广播算法

当节点与其邻居节点进行交互后，邻居节点会得到一定的评估数值。转发节点的选取机制应该更大概率地选择具有高评估数值的节点作为广播时的转发节点，但应该保证低评估数值的节点也具有一定概率被选中，避免新建立邻居关系的节点无法被选中的情况。因此，不根据节点的评估数值直接决定其概率，而是由其在路由表中评估数值的降序排序决定，如表1所示。

Table 1　Grouping and weighting of nodes in routing tables

表1　路由表中节点的分组及权重

组	1	2		3			
排序	1	2	3	4	5	6	7
权重	4	2	2	1	1	1	1

排序为i的节点将被分到第$\lfloor log_2 i \rfloor + 1$组中，节点路由表的大小应满足$2^t - 1$(如7、15、31)。若当前共有$R$组，则第$r$组每个节点的权重为$2^{R-r}$。排序为$i$的节点权重记为$S_i$，则其被选为转发节点的概率为其权重在所有未被选中节点权重之和中的占比，即：

$$P_i = \frac{S_i}{\sum_{k=1}^{n} S_k - \sum_{k=1}^{m} S'_k} \tag{2}$$

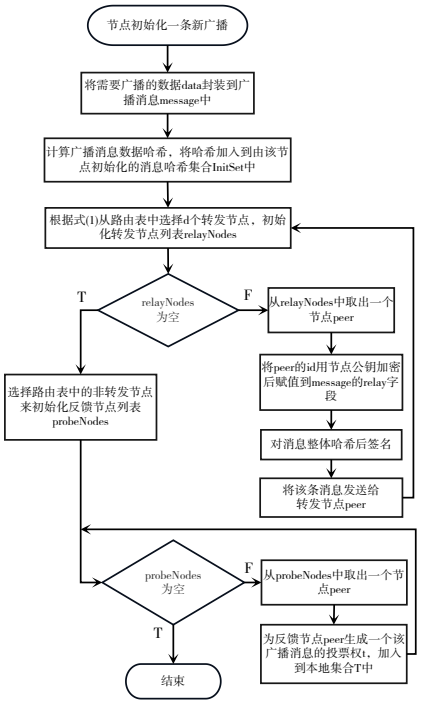
其中，n为路由表中的节点数量，S'表示已经被选中节点的集合，m为目前已经选中的转发节点数量。节点在收到一个数据包并进行转发时，选择后继转发节点时也遵循相同的策略。广播算法即节点将初始化一条广播消息，或接收到一条广播消息后的执行步骤，NE-Gossip广播算法流程如图4所示。

3　实验与结果分析

3.1　实验环境与基本参数设置

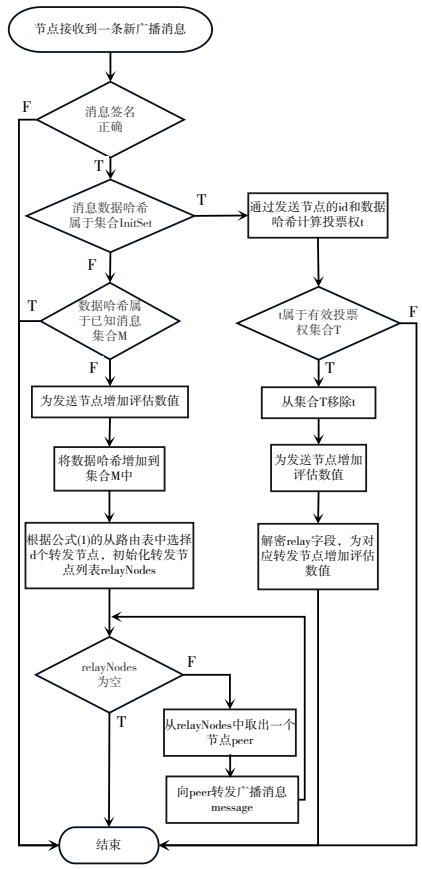
本文仿真环境的运行设备为搭载2.40 GHz的Intel Core i5-9300H CPU处理器和8.00 GB RAM的计算机。算法验证在IBS(Information Broadcasting Simulator)仿真环境中进行。IBS以Golang进行实现，它以消息为主体而不是节点为主体，因而可以在单机上模拟若干节点间的消息广播。IBS中，节点是一条条数据项，主要记录了其网络状态和路由表。每个数据包的数据结构中包含了数据ID、来源节点、目标节点、时间戳等信息，所有的数据包按其接收时间的时间戳被组织到了一个小根堆中。某条数据包转发的实现方式为：从小根堆堆顶取出一条数据包，从消息中记录的目标节点处获取它的后继广播节点列表，为每个后继节点创建一个新的数据包，通过计算其时延得出应该被接收的时间戳后插入到小根堆中。数据包的转发方式取决于具体广播算法的路由表组织方式，通过不断地从这个小根堆中获取时间戳最小的数据包并进行一次数据转发，数据将被按照对应的方式完成整个广播过程。

实验中的参数设置如下：当邻居节点表现出有利于该节点与网络的交互行为，即3种增加评估数值的行为时，其评估数值增加1。仿真网络中各节点路由表的大小设置为31，即每个节点随机地与31个其他节点建立邻居关系，广播消息的大小设为128 B。仿真环境中，每隔50 ms轮流



（a） New broadcast initialization

（a） 新广播初始化



（b） Message reception and relaying

（b） 消息接收与转发

Fig. 4 NE-Gossip broadcast algorithm flow

图4 NE-Gossip广播算法流程

向网络中一个正常运行的节点初始化一个新的广播消息，总共初始化10 000次广播。网络中节点数量为1 000，编号1至1 000。节点被随机分配到了4个不同的区域，并且也随机分配了4种不同的上行带宽，具体数值如表2和表3所示，地区间和地区内的时延设置如表4所示。在仿真实验中，上文3种行为评估数值的增加量均为1。

Table 2 The ratio of nodes in four different regions within the network

表2 网络中4个地区节点数量比例

区域	a	b	c	d
比例/%	30	10	40	20

Table 3 The ratio of nodes in four different upstream bandwidths within the network

表3 网络中4种上行带宽节点数量比例

上行带宽（BPS）	2^{19}（512 k）	2^{18}（256 k）	2^{10}（1 k）	2^{9}（512）
比例/%	30	10	40	20

Table 4 Network latency between and within the four regions

表4 网络中4个地区间及地区内的网络时延 μs

	a	b	c	d
a	10 000	200 000	250 000	250 000
b	200 000	3 000	100 000	100 000
c	250 000	100 000	7 000	200 000
d	250 000	100 000	200 000	8 000

将模拟两种网络环境对算法的广播覆盖率进行验证：

（1）仿真广播过程中每经过一个固定的间隔时间（仿真环境中的60 s）对网络进行一次扰动，但始终维持网络中任意时刻存在约半数的节点处于宕机状态。在这种环境下，覆盖率计算方式为：

$$\frac{\sum_{i=1}^{N} Received_i}{Round*N} \times 100\% \tag{3}$$

其中，N为网络中的节点数量，即1 000，$Received_i$为节点i接收到的消息数量，$Round$为广播次数，即10 000。

（2）在广播开始前，将网络中的一半节点设置为接收消息但不转发消息的恶意节点。在这种环境下，覆盖率的计算方式为：

$$\frac{\sum_{i=1}^{N_{honest}} Received_i}{Round*N_{honest}} \times 100\% \tag{4}$$

其中，N_{honest}为诚实节点数量，为500，$Received_i$为诚实节点i接收到的消息数量，$Round$同样为10 000。

此外，消息广播效果还将通过网络中各诚实节点接收到的消息数量进行更直观的体现。

3.2 实验结果与分析

3.2.1 算法在网络中半数节点宕机情况下的有效性

通过将节点宕机的概率与其序号线性相关使网络中的节点具有不同的连续运行时长。每次进行扰动时，节点1的宕机概率为0.001，节点500的概率为0.5，节点1 000的概率为1。因此，在整个模拟过程中，节点1大概率始终保

持在线,节点500每次扰动有一半的概率在线,一半的概率离线,而节点1 000则始终处于宕机状态。每次扰动节点宕机数量的平均预期值为:$\sum_{i=1}^{1\,000} P_i = 500.5$。

由于随机性,每次网络扰动实际的宕机数在500.5上下浮动。Gossip广播算法冗余度为3的实验中各节点宕机次数如图5所示,可以观察到节点的宕机次数与其序号大致呈线性相关。在整个过程中,共有64次网络扰动,扰动0次代表节点始终在线,扰动64次代表节点始终离线。

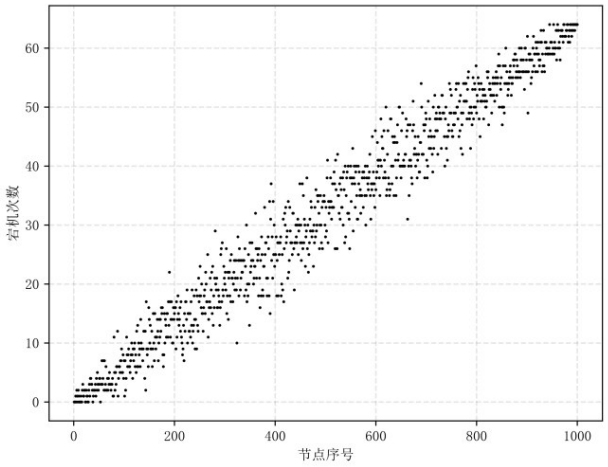


Fig. 5　Number of failures for each node in the network

图5　网络中各节点宕机次数

在第一种网络环境下,对冗余度分别为3、4、5、6时的广播覆盖率进行了模拟测试。按式(3)进行覆盖率计算,实验结果如图6所示,未被接收到的消息占比数值越低代表覆盖率越高。在各广播冗余度下,NE机制均使得广播算法具有正向提升作用。在4种冗余度下未接收消息数量分别降低了约0.96%、7.10%、5.40%、5.85%。各冗余度下网络中所有节点接受到的消息数量如图7所示,其中横坐标为节点按消息接收数量增序排序后的位次,纵坐标即对应位次的节点接收到的消息数量。

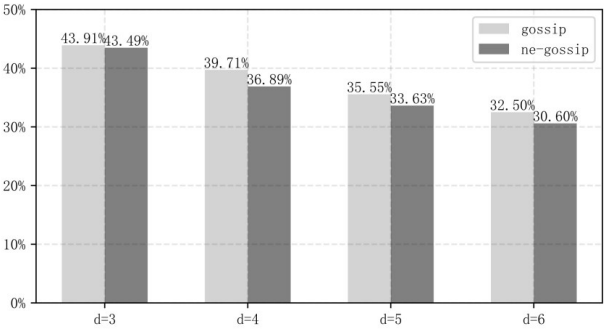
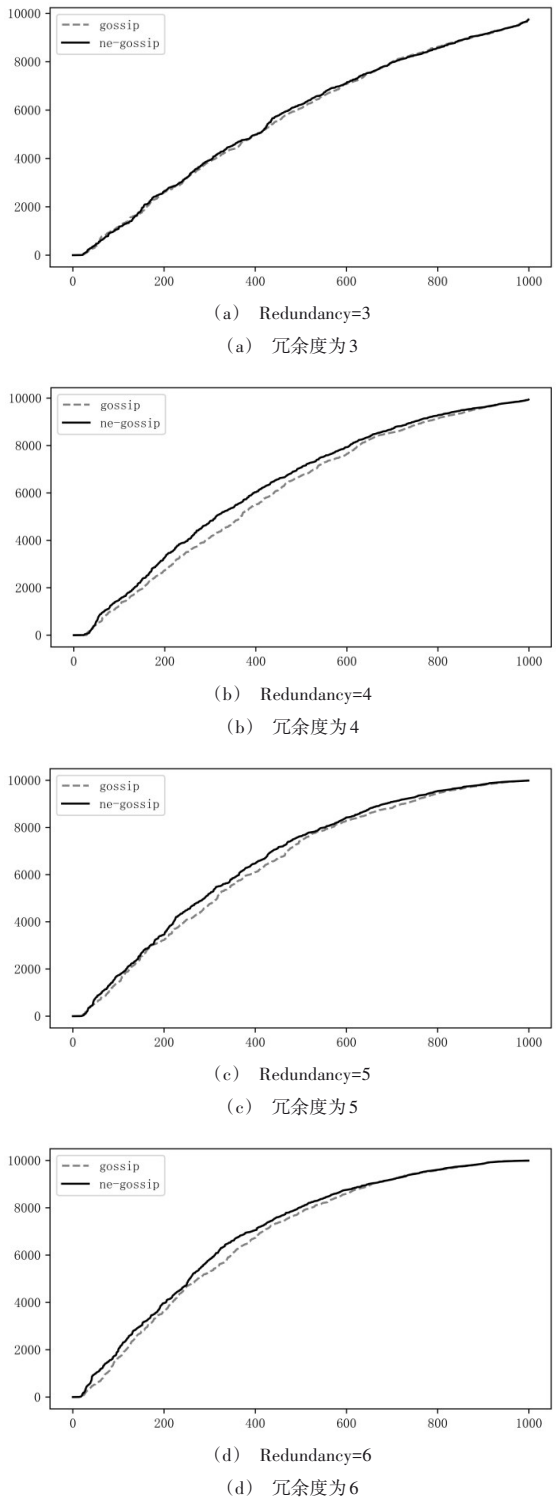


Fig. 6　The proportion of unreceived messages for the two algorithms at various redundancy levels

图6　各冗余度下两种算法未接收到的消息占比

3.2.1　算法在网络中半数节点拒绝转发消息情况下的有效性

在仿真实验开始前,将序号为偶数的节点设置为恶意节点,使得网络中一半的节点保持在线,但接收消息后不进行转发。按式(4)进行覆盖率计算,实验结果如图8所示。Gossip算法能够通过节点在线状态移除掉离线节点,因而路由表中往往保留着在线时长更长的节点,但无法获知到邻居节点是否为其转发了自己初始化的广播消息,因此相较于频繁宕机的情况,NE机制在存在一定比例不转



(a)　Redundancy=3

(a)　冗余度为3

(b)　Redundancy=4

(b)　冗余度为4

(c)　Redundancy=5

(c)　冗余度为5

(d)　Redundancy=6

(d)　冗余度为6

Fig. 7　The number of received messages for nodes at different redundancy levels

图7　各冗余度下节点接收消息数量情况

发消息的节点的网络环境中对算法覆盖率有明显提升。在冗余度分别为3、4、5、6时，未被接收到的消息数量降低了约64.07%、69.62%、68.78%、62.35%。

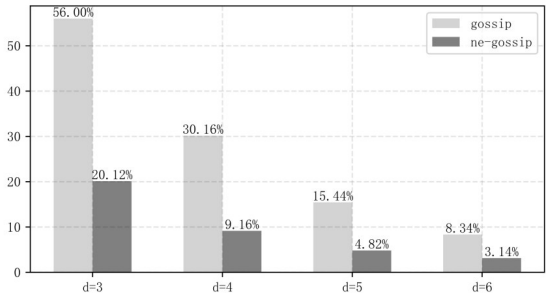


Fig. 8 The proportion of undelivered messages for the two algorithms at various redundancy levels

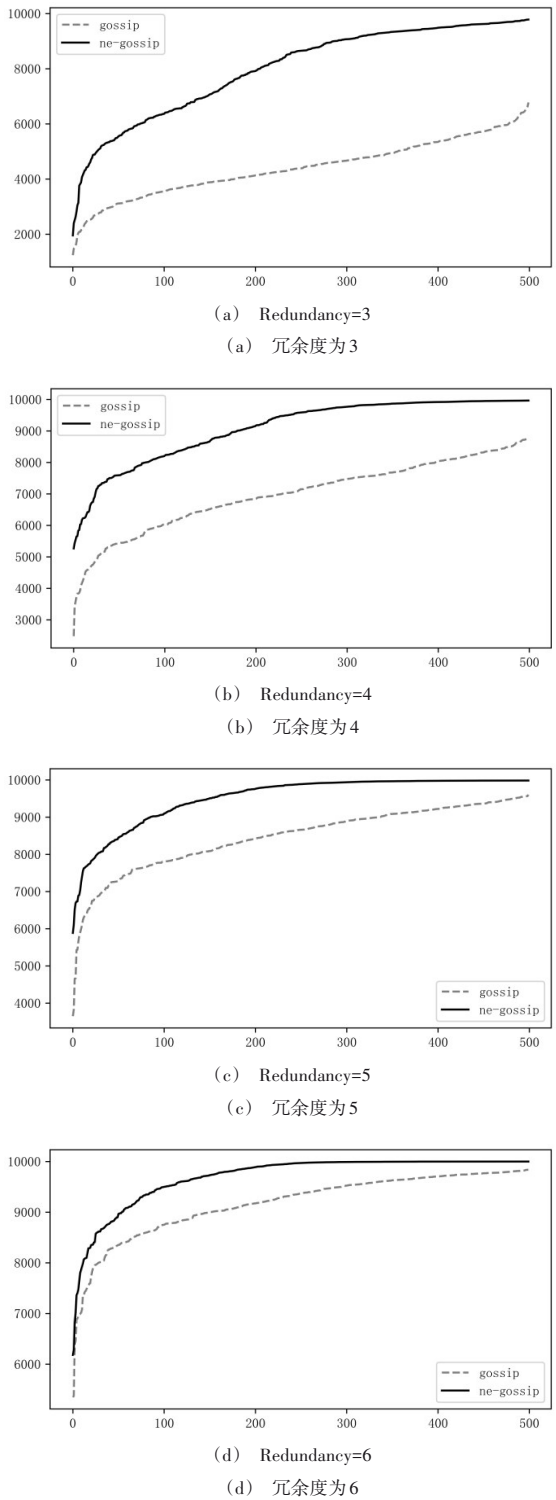
图8 各冗余度下两种算法未接收到的消息占比

各冗余度下网络中所有诚实节点接受到的消息数量如图9所示，其中横坐标为节点按消息接收数量增序排序后的位次，纵坐标即对应位次的节点接收到的消息数量。

4 结语

在Gossip算法中引入NE机制只需在广播消息中增加一个relay字段，并且该字段仅广播源节点能够解析，并不会为Gossip算法增加安全隐患。NE机制并非通过减少消息的传播量，而是将消息传播任务根据过去的表现分配给了更能胜任消息转发任务的节点，从而提升广播效率。根据实验结果，NE-Gossip算法相较于Gossip算法在广播覆盖率上有更好的表现。在网络中有一半的节点保持在线但恶意不转发消息时，广播覆盖率有显著提升，能够减少至少60%未被接收消息的数量；在网络中有大量节点频繁宕机，任意时刻仅有约一半的节点在线时，NE机制对Gossip在广播冗余度分别为3、4、5、6时均有正向提升作用，但这种提升不如存在恶意节点的环境下显著。Gossip算法中节点是否具有对应场景的判断力影响并促成了这种差别。节点能够根据邻居节点是否有响应判断在线情况，进而在路由表中移除掉线的邻居节点，在线时间更长的节点信息更容易存留在其他网络节点的路由表中。然而，节点完全无法判断邻居节点是否为其转发了消息，NE机制使得节点具有了这种判断力，因而在实验数据上表现出了明显差异。

高效的P2P广播算法是区块链系统能够正常运行的前提，相较于区块链的其他研究课题而言关注度较少，但其在区块链及其他分布式系统中非常重要。本文NE-Gossip广播算法对邻居节点的转发能力进行评估，并根据评估结果选择广播节点达到了更好的广播效果。相信，随着相关研究的不断深入，广播算法也能够得到进一步完善。



（a） Redundancy=3

（a） 冗余度为3

（b） Redundancy=4

（b） 冗余度为4

（c） Redundancy=5

（c） 冗余度为5

（d） Redundancy=6

（d） 冗余度为6

Fig. 9 The number of received messages for nodes at different redundancy levels

图9 各冗余度下节点接收消息数量情况

参考文献：

[1] ZENG S Q, HUO R, HUANG T, et al. Survey of blockchain: principle, progress and application[J]. Journal on Communications, 2020, 41(1): 134-151.
曾诗钦，霍如，黄韬，等. 区块链技术研究综述：原理、进展与应用[J].

通信学报,2020,41(1):134-151.

[2] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system[DB/OL]. https://bitcoin. org/en/bitcoin-paper,2008.

[3] BUTERIN V. A next-generation smart contract and decentralized application platform[EB/OL]. https://github. com/ethereum/wiki/blob/old-before-deleting-all-files-go-to-wiki-wiki-instead/old-whitepaper-for-histori - cal-reference. md.

[4] SAITO K, YAMADA H. What's so different about blockchain? —blockchain is a probabilistic state machine[C]//2016 IEEE 36th International Conference on Distributed Computing Systems Workshops (ICDCSW), 2016: 168-175.

[5] DECKER C, WATTENHOFER R. Information propagation in the bitcoin network[C]//Trento: IEEE Thirteenth International Conference on Peer-to-peer Computing,2013.

[6] CROMAN K, DECKER C, EYAL I, et al. On scaling decentralized blockchains[C]//Proceedings of International Conference on Financial Cryptography and Data Security,2016: 106-125.

[7] LI C, LI P, ZHOU D, et al. A scaling Nakamoto consensus to thousands of transactions per second[DB/OL]. https://arxiv. org/abs/1805. 03870v4, 2018.

[8] BAGARIA V, KANNAN S, TSE D, et al. Prism: deconstructing the blockchain to approach physical limits[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019: 585-602.

[9] YU H, NIKOLIĆ I, HOU R, et al. Ohie: blockchain scaling made simple[C]//2020 IEEE Symposium on Security and Privacy(SP),2020: 90-105.

[10] LUU L, NARAYANAN V, ZHENG C A. Secure sharding protocol for open blockchains[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security,2016: 17-30.

[11] ZAMANI M, MOVAHEDI M, RAYKOVA M. Rapidchain: scaling blockchain via full sharding[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security,2018: 931-948.

[12] HAN Y, LI C, LI P, et al. Shrec: bandwidth-efficient transaction relay in high-throughput blockchain systems[C]//Proceedings of the 11th ACM Symposium on Cloud Computing,2020:238-252.

[13] MIŠIĆ J, MIŠIĆ V B, CHANG X. On the benefits of compact blocks in Bitcoin[C]//Dublin: 2020 IEEE International Conference on Communications (ICC),2020.

[14] SI B R, XIAO J, LIU C Y, et al. Survey on blockchain network[J]. Journal of Software,2024,35(2):773-799.
司冰茹,肖江,刘存扬,等. 区块链网络综述[J]. 软件学报,2024,35(2):773-799.

[15] GARAY J, KIAYIAS A, LEONARDOS N. The Bitcoin backbone protocol: analysis and applications[C]//Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2015: 281-310.

[16] LEITÃO J, PEREIRA J, RODRIGUES L. Gossip-based broadcast[M]. New York: Springer,2010.

[17] CORALLO M. Compact block relay[EB/OL]. https://github. com/bitcoin/blob/ master/bip-0152. mediawiki.

[18] QIU H R, JI T, ZHAO S X, et al. A geography-based P2P overlay network for fast and robust blockchain systems[J]. IEEE Transactions on Services Computing,2023,16(3):1572-1588.

[19] HAO W, ZENG J, DAI X, et al. BlockP2P: enabling fast blockchain broadcast with scalable peer-to-peer network topology[C]//Proceedings of 14th International Conference on Green, Pervasive and Cloud Computing,2019:223-237.

(责任编辑:孙　娟)