



(12) 发明专利

(10) 授权公告号 CN 110012100 B

(45) 授权公告日 2021.04.27

(21) 申请号 201910278978.7

(22) 申请日 2019.04.09

(65) 同一申请的已公布的文献号

申请公布号 CN 110012100 A

(43) 申请公布日 2019.07.12

(73) 专利权人 杭州秘猿科技有限公司

地址 310013 浙江省杭州市西湖区文三路
478号华星时代广场A座1301

(72) 发明人 高亦民 万涔涔 俞波

(74) 专利代理机构 北京德崇智捷知识产权代理
有限公司 11467

代理人 董柏雷

(51) Int. Cl.

H04L 29/08 (2006.01)

(56) 对比文件

CN 109255713 A, 2019.01.22

CN 109360100 A, 2019.02.19

CN 109150598 A, 2019.01.04

US 2018365691 A1, 2018.12.20

傲慢. 区块链共识算法 PBFT (拜占庭容错)、
PAXOS、RAFT 简述.《CSDN 博客》.2017,刘肖飞. 基于动态授权的拜占庭容错共识算
法的区块链性能改进研究.《中国优秀硕士学位
论文全文数据库 信息科技辑》.2018,

审查员 张翔

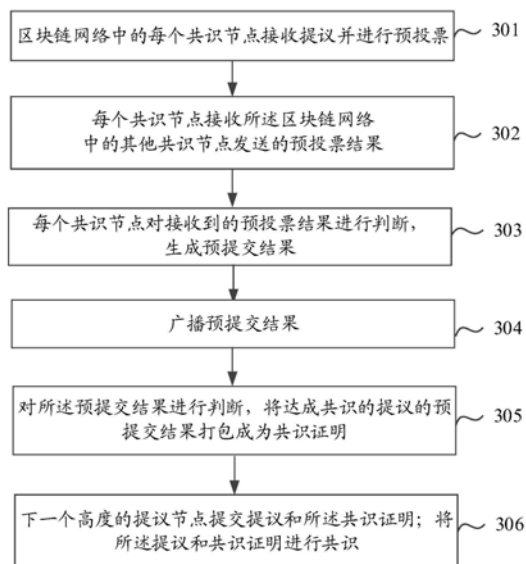
权利要求书2页 说明书8页 附图3页

(54) 发明名称

一种带宽优化的区块链共识方法、装置及电
子设备

(57) 摘要

本申请实施例公开了一种带宽优化的区块链共识方法、装置及电子设备,应用于包括多个区块链节点的区块链网络,所述方法包括:所述区块链网络中的每个共识节点接收提议并进行预投票;所述每个共识节点接收所述区块链网络中的其他共识节点发送的预投票结果;所述每个共识节点对接收到的预投票结果进行判断,生成预提交结果;广播所述预提交结果;所述每个共识节点对所述预提交结果进行判断,将达成共识的提议的预提交结果打包成为共识证明;下一个高度的提议节点提交提议和所述共识证明;将所述提议和所述共识证明进行下一轮的共识。本申请实施例中提供的方法,优化了带宽资源,避免了带宽资源的浪费,提高了带宽利用率。



1. 一种带宽优化的区块链共识方法,应用于包括多个区块链节点的区块链网络,所述方法包括:

所述区块链网络中的每个共识节点接收提议并进行预投票;

所述每个共识节点接收所述区块链网络中的其他共识节点发送的预投票结果;

所述每个共识节点对接收到的预投票结果进行判断,生成预提交结果;广播所述预提交结果;

所述每个共识节点对所述预提交结果进行判断,将达成一致的提议的预提交结果打包成为共识证明;

下一个高度的提议节点提交提议和所述共识证明;

将所述提议和所述共识证明进行新一轮的共识,从而生成共识结果。

2. 根据权利要求1所述的方法,其特征在于,所述每个共识节点对所述预提交结果进行判断,将达成一致的提议的预提交结果打包成为共识证明,包括:

所述每个共识节点接收到的所述预提交结果与所述区块链网络中的所有共识节点的比值大于或等于第一阈值,将达成一致的提议的预提交结果打包成为共识证明。

3. 根据权利要求1或2所述的方法,其特征在于,所述每个共识节点对接收到的预投票进行判断,生成预提交结果,包括:

所述每个共识节点接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于第二阈值,生成所述预提交结果。

4. 根据权利要求2所述的方法,其特征在于,所述第一阈值为2/3。

5. 根据权利要求3所述的方法,其特征在于,所述第二阈值为2/3。

6. 一种带宽优化的区块链共识装置,应用于包括多个区块链节点的区块链网络,所述装置包括:

收发模块,用于所述区块链网络中的每个共识节点接收提议和接收所述区块链网络中的其他共识节点发送的预投票结果;

处理模块,用于所述区块链网络中的每个共识节点根据提议进行预投票,以及对所述收发模块接收到的预投票结果进行判断,根据所述收发模块接收到的预投票结果生成预提交结果;

所述收发模块,还用于广播所述预提交结果;

所述处理模块,还用于所述每个共识节点对所述预提交结果进行判断,将达成一致的提议的预提交结果打包成为共识证明;下一个高度的提议节点提交提议和所述共识证明;将所述提议和所述共识证明进行新一轮的共识,从而生成共识结果。

7. 根据权利要求6所述的装置,其特征在于,所述处理模块,具体用于所述收发模块接收到的所述预提交结果与所述区块链网络中的所有共识节点的比值大于或等于第一阈值,将达成一致的提议的预提交结果打包成为共识证明。

8. 根据权利要求6或7所述的装置,其特征在于,所述处理模块,具体用于所述收发模块接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于第二阈值,生成所述预提交结果。

9. 一种电子设备,包括:存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,所述计算机程序被所述处理器执行如权利要求1-5中任一项所述的方法。

10. 一种计算机存储介质, 包括指令, 用于使计算机执行如权利要求1-5中任一项所述的方法。

一种带宽优化的区块链共识方法、装置及电子设备

技术领域

[0001] 本申请涉及网络技术领域,尤其涉及一种带宽优化的区块链共识方法、装置及电子设备。

背景技术

[0002] 区块链是一种新的分布式技术,是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。采用区块链技术架构的网络可视为区块链网络,该区块链网络中包含多个区块链节点,任一区块链节点可对应至少一个区块链,任一区块链可包含至少一个区块。

[0003] 在区块链技术中,共识算法是区块链网络中实现不同区块链节点之间建立信任、获取权益的重要方法。目前采用拜占庭容错 (Byzantine Fault Tolerance, BFT) 这类共识算法的区块链中,区块链底层平台先对交易完成共识,共识完成后才会去完成交易的执行计算。在这类共识算法中,一般会采用多轮投票,在多轮投票的过程中,由于网络等原因,虽然所有共识节点都收到了超过2/3的对一个提议的预提交投票,但是每个共识节点可能收到的预提交投票是不同的。因此对超过一定比例的对共识成功的提议的预投票也需要共识。在当前共识方法中,大多是将所有的预提交结果的数据打包成为共识证明,在下一轮的提议节点一同提交进行新一轮共识。这导致了带宽资源的浪费。

[0004] 由此,亟需找到一种区块链中优化带宽的区块链共识方案,以克服上述问题。

发明内容

[0005] 本申请实施例提供一种带宽优化的区块链共识方法、装置及电子设备,用以解决现有技术的拜占庭容错这类共识算法中带宽资源浪费的问题。

[0006] 为了解决上述技术问题,本申请实施例采用下述技术方案:

[0007] 第一方面,提供了一种带宽优化的区块链共识方法,应用于包括多个区块链节点的区块链网络,所述方法包括:

[0008] 所述区块链网络中的每个共识节点接收提议并进行预投票;

[0009] 所述每个共识节点接收所述区块链网络中的其他共识节点发送的预投票结果;

[0010] 所述每个共识节点对接收到的预投票结果进行判断,生成预提交结果;

[0011] 广播所述预提交结果;

[0012] 所述每个共识节点对所述预提交结果进行判断,将达成一致的提议的预提交结果打包成为共识证明;

[0013] 下一个高度的提议节点提交提议和所述共识证明;

[0014] 将所述提议和所述共识证明进行下一轮的共识。

[0015] 可选的,所述每个共识节点对所述预提交结果进行判断,将达成一致的提议的预提交结果打包成为共识证明,包括:

[0016] 所述每个共识节点接收到的所述预提交结果与所述区块链网络中的所有共识节

点的比值大于或等于第一阈值,将达成共识的提议的预提交结果打包成为共识证明。

[0017] 在一种可能的实现方式中,所述每个共识节点对接收到的预投票进行判断,生成预提交结果,包括:

[0018] 所述每个共识节点接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于第二阈值,生成所述预提交结果。

[0019] 可选的,所述第一阈值为2/3。

[0020] 可选的,所述第二阈值为2/3。

[0021] 第二方面,提供了一种带宽优化的区块链共识装置,应用于包括多个区块链节点的区块链网络,所述装置包括:

[0022] 收发模块,用于所述区块链网络中的每个共识节点接收提议和接收所述区块链网络中的其他共识节点发送的预投票结果;

[0023] 处理模块,用于所述区块链网络中的每个共识节点根据提议进行预投票,以及对所述收发模块接收到的预投票结果进行判断,根据所述收发模块接收到的预投票结果生成预提交结果;

[0024] 所述收发模块,还用于广播所述预提交结果;

[0025] 所述处理模块,还用于所述每个共识节点对所述预提交结果进行判断,将达成共识的提议的预提交结果打包成为共识证明;下一个高度的提议节点提交提议和所述共识证明;将所述提议和所述共识证明进行下一轮的共识。

[0026] 在一种可能的实现方式中,所述处理模块,具体用于所述收发模块接收到的所述预提交结果与所述区块链网络中的所有共识节点的比值大于或等于第一阈值,将达成共识的提议的预提交结果打包成为共识证明。

[0027] 在一种可能的实现方式中,所述处理模块,具体用于所述收发模块接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于第二阈值,生成所述预提交结果。

[0028] 第三方面,提供了一种电子设备,包括:存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,所述计算机程序被所述处理器执行第一方面或第一方面任意一种实现方式所述的方法。

[0029] 第四方面,提供了一种计算机存储介质,包括指令,用于使计算机执行第一方面或第一方面任意一种实现方式所述的方法。

[0030] 第五方面,本申请提供一种芯片,所述芯片与存储器相连,用于读取并执行所述存储器中存储的软件程序,以实现第一方面或第一方面任意一种实现方式所述的方法。

[0031] 本申请实施例采用的上述至少一个技术方案能够达到以下有益效果:

[0032] 本申请实施例中提供的技术方案,对一个提议的成功共识,要达到或超过一定比例的预投票和达到或超过一定比例的预提交才可以达成共识,共识证明中只包含对共识成功的提议的预提交,有效地优化了带宽资源,避免了带宽资源的浪费,提高了带宽利用率。

附图说明

[0033] 为了更清楚地说明本说明书实施例或现有技术中的技术方案,下面将对实施例或

现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本说明书实施例中记载的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0034] 图1为本申请实施例提供的一种区块链共识算法的示意图;

[0035] 图2为本申请实施例提供的另一种区块链共识算法的示意图;

[0036] 图3为本申请实施例提供的一种带宽优化的区块链共识方法流程图;

[0037] 图4为本申请实施例提供的一种电子设备的结构示意图;

[0038] 图5为本申请实施例提供的一种带宽优化的区块链共识装置结构示意图。

具体实施方式

[0039] 为使本说明书实施例的目的、技术方案和优点更加清楚,下面将结合本说明书具体实施例及相应的附图对本说明书实施例的技术方案进行清楚、完整地描述。显然,所描述的实施例仅是本说明书一部分实施例,而不是全部的实施例。基于本说明书中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本说明书实施例保护的范围。

[0040] 以下结合附图,详细说明本说明书各实施例提供的技术方案。

[0041] 区块链本质上是一个去中心化的数据库,是一连串使用密码学方法产生相关联的数据块,每一个数据块中包含了一段时间内全网交易的信息。不同的区块链,有着不同的内涵和功能。区块链中最重要的就是共识算法,共识算法中包括分布式一致性算法,分布式一致性算法一般可以分为BFT算法和非BFT算法。目前主流的拜占庭容错算法包括传统的实用拜占庭容错(Practical Byzantine Fault Tolerance, PBFT)算法、tendermint算法等。

[0042] 基于拜占庭将军的问题,一致性的确保主要分为三个阶段:预准备(pre-prepare)、准备(prepare)和确认(commit)。其示意图如图1所示。

[0043] 在图1中,C为发送请求端节点,0、1、2、3为服务端节点,3为宕机的服务端节点,具体步骤如下:

[0044] 请求(Request):请求端节点C发送请求到任意一节点,在本申请的实施例中为0。

[0045] 预准备(Pre-Prepare):服务端节点0收到节点C的请求后进行广播,扩散至节点1、2、3。

[0046] 准备(Prepare):节点1、2、3收到节点0的广播后记录并再次广播。节点1向节点0、2、3广播,节点2向节点0、1、3广播,节点3因为宕机无法广播。

[0047] 确认(Commit):节点0、1、2、3在准备阶段,若收到超过一定数量的相同请求,则进入确认阶段,广播确认请求。

[0048] 答复(Reply):节点0、1、2、3在确认阶段,若收到超过一定数量的相同请求,则对节点C进行反馈。

[0049] 由此可以看出,PBFT共识算法其实是一种状态机副本复制算法,即服务作为状态机进行建模,状态机在分布式系统的不同节点进行副本复制。每个状态机的副本都保存了服务的状态,同时也实现了服务的操作。

[0050] 拜占庭容错算法中的tendermint算法是一个易于理解、大部分操作为异步的BFT共识协议,其针对PBFT算法做了优化,只需要有两轮投票即可达成共识。图2为本申请实施

例提供的一种tendermint共识算法的示意图。

[0051] 协议中的参与者叫着“验证人”(validator)。他们轮流对交易区块进行提议,并对这些区块进行投票。区块会被提交到链上,每一个块占据一个“高度”(height)。提交块可能会失败,如果失败,协议就会开始下一轮的提交,并且一个新的验证人会继续提交那个高度的区块。要想成功提交一个块,需要有两个阶段的投票:“预投票”(prevote)和“预确认”(precommit)。在同一轮提交中,只有超过2/3的验证人对同一个块进行了预提交,这个块才能被提交到链上。

[0052] 如图1所示,验证人做的事情就像是在跳波卡舞。超过2/3的验证人对同一个块进行了预投票,叫做一个“波卡”(polka)。每一个预提交都必须被同一轮中的一个波卡所证明。由于一些原因,验证人可能在提交一个块时失败:当前提议者可能离线了,或者网络非常慢。Tendermint共识算法允许他们证实一个验证人应该被跳过。在进行下一轮的投票前,验证人会等待一小段时间从提议者那里接收一个完整的提议块。这种对于超时的依赖,使得Tendermint共识算法成为了一个弱同步协议,而非一个异步协议。但是,协议的剩余部分都是异步的,只有在接收到超过2/3的验证人集合时,验证人才会采取下一步操作。Tendermint共识算法能够简化的一个原因就是它使用了同样的机制来提交一个块和跳过直接进入下一轮。

[0053] 基于不到1/3的验证人是拜占庭节点的前提,Tendermint共识算法保证了永远都不会违背其安全性。也就是说,验证人永远不会在同一高度提交冲突块。为了达到这一点,它引入了一些“锁定”(locking)的规则,这些规则对流程图中的路径进行了模块化。一旦一个验证人预提交了一个块,它就被“锁定”在了那个块上。然后,它必须为被锁定的那个块进行预投票,并且只有在之后的轮中,有了那个块的一个波卡,它才能够解锁,并为一个新块进行预提交。

[0054] 对高度为h的块的共识可能需要多轮(round),每一轮又包括3步:提议(Propose),预投票(Prevote),预确认(Precommit)。当在某一轮达成共识(收到+2/3的Precommit投票)后,就会进入对下一个高度的共识,从第0轮开始。下面结合图2对tendermint共识算法进行一下简单介绍。

[0055] PoLC(Proof of Lock Change)在Tendermint共识算法中是一个非常重要的概念,其表示在某个特定的高度和轮数(height,round),对某个块或空块(nil)超过总结点2/3的Prevote投票集合,简单来说PoLC就是Prevote的投票集。

[0056] Propose步骤:节点轮流提出Proposal,当轮到某一节点提议时,该节点就会提出Proposal,当非提议节点(Proposer)接收到Proposal后,就会进入Prevote步骤,如果有Lock-Block,还需要收集所有针对Lock-Block的Prevote投票。

[0057] Prevote步骤:节点进入Prevote步骤后,如果自己有Lock-Block,并且有一个新的针对另外一个块的PoLC,并且满足 $LastLockRound < PoLC-Round < 当前Round$,则解锁Lock-Block。此时,如果自己仍然有Lock-Block,则对Lock-Block投Prevote,否则的话,投Proposal或者nil。当超时或者接收到+2/3的针对某个块或nil的投票时进入Precommit步骤。

[0058] Precommit步骤:如果当前有对某个块的PoLC,则将这个块锁住,并且将LastLockRound置为当前Round,并对这个块投Precommit;如果有针对nil的PoLC,则解锁并

且对nil投Precommit; 否则的话保持Lock-Block不变, 并投nil, 当节点收到有超过2/3的针对某一个块的Precommit投票时, 就可以将这个块commit, 并且进入针对下一个高度块的共识。

[0059] 在图1和图2这类的共识算法中, 都采用了多轮投票, 在多轮投票的过程中, 由于网络等原因, 虽然所有共识节点都收到了超过一定比例(例如2/3)的对一个提议的预提交投票, 但是每个共识节点可能收到的预提交投票是不同的。因此对超过一定比例的对共识成功的提议的预投票也需要共识。在当前共识算法中, 大多是将所有的预提交结果的数据打包成为共识证明(Proof), 在下一轮的提议节点一同提交进行新一轮共识。这导致了带宽资源的浪费。为了解决多轮投票过程中资源浪费、利用率低的问题, 本申请实施例提供了一种带宽优化的区块链共识方法, 解决了现有技术中带宽资源浪费的问题, 有效地优化了带宽资源, 提高了带宽资源的使用率。

[0060] 实施例一

[0061] 图3为本申请实施例提供的一种带宽优化的区块链共识方法流程图, 应用于包括多个区块链共识节点的区块链网络, 包括如下步骤:

[0062] S301、所述区块链网络中的每个共识节点接收提议并进行预投票。

[0063] S302、所述每个共识节点接收所述区块链网络中的其他共识节点发送的预投票结果。

[0064] S303、所述每个共识节点对接收到的预投票结果进行判断, 生成预提交结果。

[0065] 在本申请的一个实施方式中, 为了优化带宽, 提高带宽的利用率, 每个共识节点对接收到的预投票结果进行判断, 当接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于预定阈值, 才生成预提交结果。对于预定阈值的取值, 不同的共识算法会设定不同的预定阈值。例如, 在BFT算法中, 预定阈值可以为2/3。

[0066] S304、广播所述预提交结果。

[0067] 每个共识节点通过广播接收预提交结果。

[0068] S305、对所述预提交结果进行判断, 将达成一致的提议的预提交结果打包成为共识证明。

[0069] 在本申请的一个实施方式中, 为了提高带宽资源的利用率, 所述每个共识节点对所述预提交结果进行判断, 当预提交结果与所述区块链网络中的所有共识节点的比值大于或等于第一阈值, 才将达成一致的提议的预提交结果打包成为共识证明(proof)。与预定阈值相类似, 第一阈值的取值会因为不同的共识算法而设定不同的值。可选的, 第一阈值也可以为2/3。

[0070] 需要说明的是, 本申请实施例中的第一阈值和预定阈值的取值, 会因为不同的共识算法而设定不同的取值, 并不局限于本申请实施例中所列举的数值。

[0071] S306、下一个高度的提议节点提交提议和所述共识证明; 将所述提议和所述共识证明进行下一轮的共识。

[0072] 因为对达到或者超过一定比例的对共识成功的提议的预投票也需要共识, 在该步骤中, 将预提交结果打包成为共识证明, 并由下一轮的下一个高度的提议节点连同提议一同提交进行新一轮的共识, 从而生成共识结果。

[0073] 在现有技术的共识方法中, 是将所有的预提交结果的数据打包生成共识证明。而

在本申请提供的方法中,每个共识节点对其预提交结果进行了判断,只将对达成共识的提议的预提交结果打包生成成为共识证明进行新一轮的共识。以BFT算法为例,对一个提议的成功共识,要达到或超过2/3(预定阈值)的预投票和达到或超过2/3(第一阈值)的预提交才可以达成共识,共识证明中只包含对共识成功的提议的预提交,与打包所有的预提交投票数据生成的共识证明进行共识,能够提高33.3%的带宽利用率。

[0074] 本申请实施例中提供的方法,优化了带宽资源,避免了带宽资源的浪费,提高了带宽利用率。

[0075] 实施例二

[0076] 下面参照图4详细介绍本说明书实施例的电子设备。请参考图4,在硬件层面,该电子设备包括处理器,可选地还包括内部总线、网络接口、存储器。其中,存储器可能包含内存,例如高速随机存取存储器(Random-Access Memory,RAM),也可能还包括非易失性存储器(Non-Volatile Memory),例如至少1个磁盘存储器等。当然,该电子设备还可能包括其他业务所需要的硬件。

[0077] 处理器、网络接口和存储器可以通过内部总线相互连接,该内部总线可以是工业标准体系结构(Industry Standard Architecture,ISA)总线、外设部件互连标准(Peripheral Component Interconnect,PCI)总线或扩展工业标准结构(Extended Industry Standard Architecture,EISA)总线等。所述总线可以分为地址总线、数据总线、控制总线等。为便于表示,图4中仅用一个双向箭头表示,但并不表示仅有一根总线或一种类型的总线。

[0078] 存储器,用于存放程序。具体地,程序可以包括程序代码,所述程序代码包括计算机操作指令。存储器可以包括内存和非易失性存储器,并向处理器提供指令和数据。

[0079] 处理器从非易失性存储器中读取对应的计算机程序到内存中然后运行,在逻辑层面上形成内容推荐装置。处理器,执行存储器所存放的程序,并具体用于执行前文所述服务器作为执行主体时所执行的方法操作。

[0080] 上述如本说明书实施例图3所示实施例揭示的方法可以应用于处理器中,或者由处理器实现。处理器可能是一种集成电路芯片,具有信号的处理能力。在实现过程中,上述方法的各步骤可以通过处理器中的硬件的集成逻辑电路或者软件形式的指令完成。上述的处理器可以是通用处理器,包括中央处理器(Central Processing Unit,CPU)、网络处理器(Network Processor,NP)等;还可以是数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)、现场可编程门阵列(Field-Programmable Gate Array,FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。可以实现或者执行本说明书实施例中的公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本说明书实施例所公开的方法的步骤可以直接体现为硬件译码处理器执行完成,或者用译码处理器中的硬件及软件模块组合执行完成。软件模块可以位于随机存储器,闪存、只读存储器,可编程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器,处理器读取存储器中的信息,结合其硬件完成上述方法的步骤。

[0081] 该电子设备还可执行图3的方法,并实现带宽优化的区块链共识装置在图3所示实

施例的功能,本说明书实施例在此不再赘述。

[0082] 当然,除了软件实现方式之外,本说明书实施例的电子设备并不排除其他实现方式,比如逻辑器件抑或软硬件结合的方式等等,也就是说以下处理流程的执行主体并不限定于各个逻辑单元,也可以是硬件或逻辑器件。

[0083] 实施例三

[0084] 本说明书实施例还提供一种计算机可读存储介质,所述计算机可读存储介质存储一个或多个程序,所述一个或多个程序当被包括多个应用程序的电子设备执行时,使得所述电子设备执行实施例一所述的方法。在此不再赘述。

[0085] 其中,所述的计算机可读存储介质,如只读存储器(Read-Only Memory,ROM)、随机存取存储器(Random Access Memory,RAM)、磁碟或者光盘等。

[0086] 实施例四

[0087] 参照图5所示,为本说明书实施例提供的带宽优化的区块链共识装置的结构示意图,该装置主要包括收发模块501和处理模块502。

[0088] 其中,收发模块501,用于所述区块链网络中的每个共识节点接收提议和接收所述区块链网络中的其他共识节点发送的预投票结果。

[0089] 处理模块502,用于所述区块链网络中的每个共识节点根据提议进行预投票,以及对所述收发模块接收到的预投票结果进行判断,根据所述收发模块接收到的预投票结果生成预提交结果。

[0090] 在本申请的一个实施方式中,为了优化带宽,提高带宽的利用率,当收发模块接收到的预投票结果与所述区块链网络中的所有共识节点的比值大于或等于预定阈值,才进行下一步的操作。对于预定阈值的取值,不同的共识算法会设定不同的预定阈值。例如,在BFT算法中,预定阈值可以为 $2/3$ 。

[0091] 收发模块501,还用于广播所述预提交结果。

[0092] 处理模块502,还用于所述每个共识节点对所述预提交结果进行判断,将达成共识的提议的预提交结果打包成为共识证明;下一个高度的提议节点提交提议和所述共识证明;将所述提议和所述共识证明进行下一轮的共识。

[0093] 在本申请的一个实施方式中,为了提高带宽资源的利用率,当预提交结果与所述区块链网络中的所有共识节点的比值大于或等于第一阈值,才将达成共识的提议的预提交结果打包成为共识证明。与预定阈值相类似,第一阈值的取值会因为不同的共识算法而设定不同的值。可选的,第一阈值也可以为 $2/3$ 。

[0094] 需要说明的是,本申请实施例中的预定阈值和第一阈值的取值,会因为不同的共识算法而设定不同的取值,并不局限于本申请实施例中所列举的数值

[0095] 具体地,因为对达到或者超过一定比例的对共识成功的提议的预投票也需要共识,在该步骤中,处理模块将预提交结果打包生成共识证明,并由下一轮的提议节点连同提议一同提交进行新一轮的共识,从而生成共识结果。

[0096] 在现有技术的共识方法中,是将所有的预提交结果的数据打包生成共识证明。在本申请的实施方式中,处理模块对预提交结果进行了判断,只将对达成共识的提议的预提交结果打包生成成为共识证明进行新一轮的共识。以BFT算法为例,对一个提议的成功的共识,要达到或超过 $2/3$ (预定阈值)的预投票和达到或超过 $2/3$ (第一阈值)的预提交才可以达

成共识,共识证明中只包含对共识成功的提议的预提交,与打包所有的预提交投票数据生成的共识证明进行共识,能够提高33.3%的带宽利用率。

[0097] 本申请实施例中提供的装置,优化了带宽资源,避免了带宽资源的浪费,提高了带宽利用率。

[0098] 总之,以上所述仅为本说明书实施例的较佳实施例而已,并非用于限定本说明书实施例的保护范围。凡在本说明书实施例的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本说明书实施例的保护范围之内。

[0099] 上述实施例阐明的系统、装置、模块或单元,具体可以由计算机芯片或实体实现,或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的,计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[0100] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[0101] 还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[0102] 本说明书实施例中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

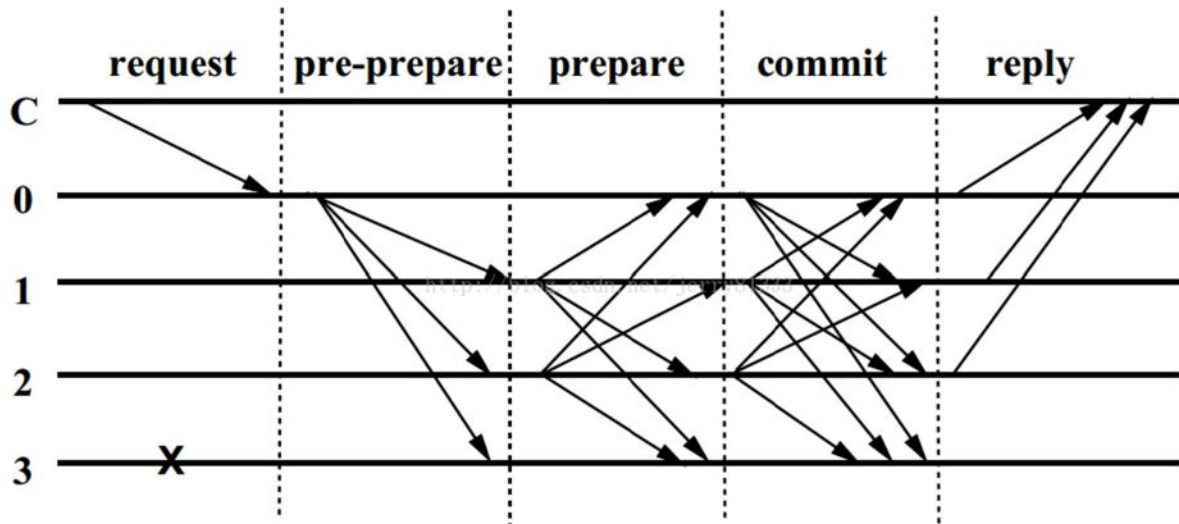


图1

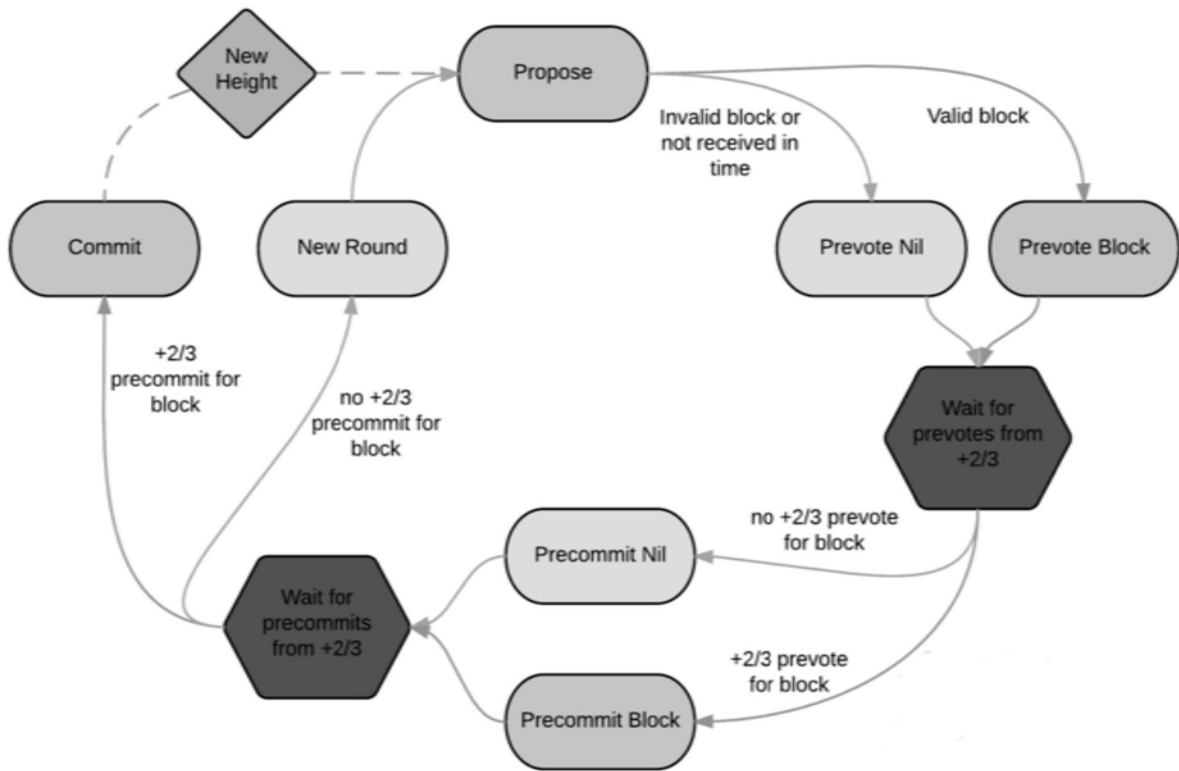


图2

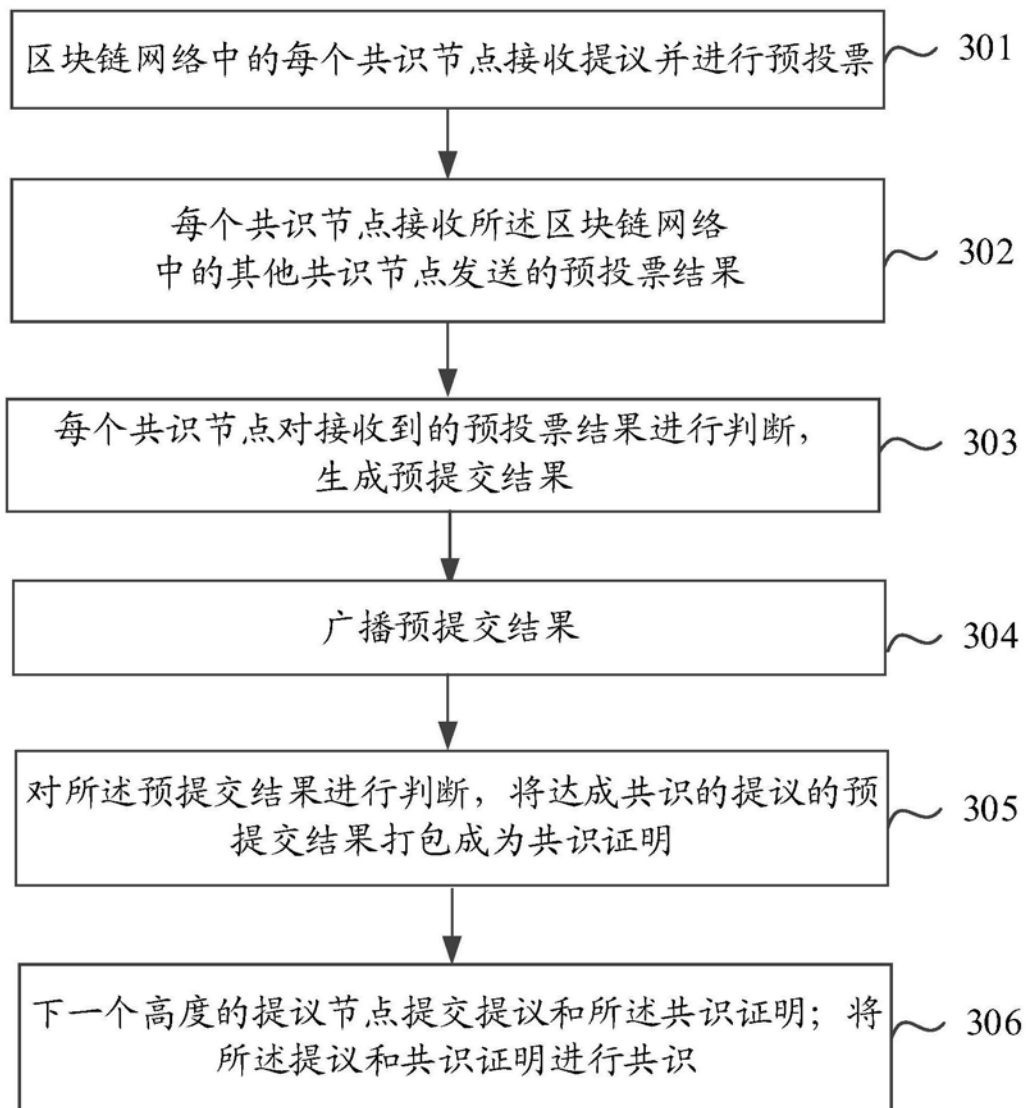


图3

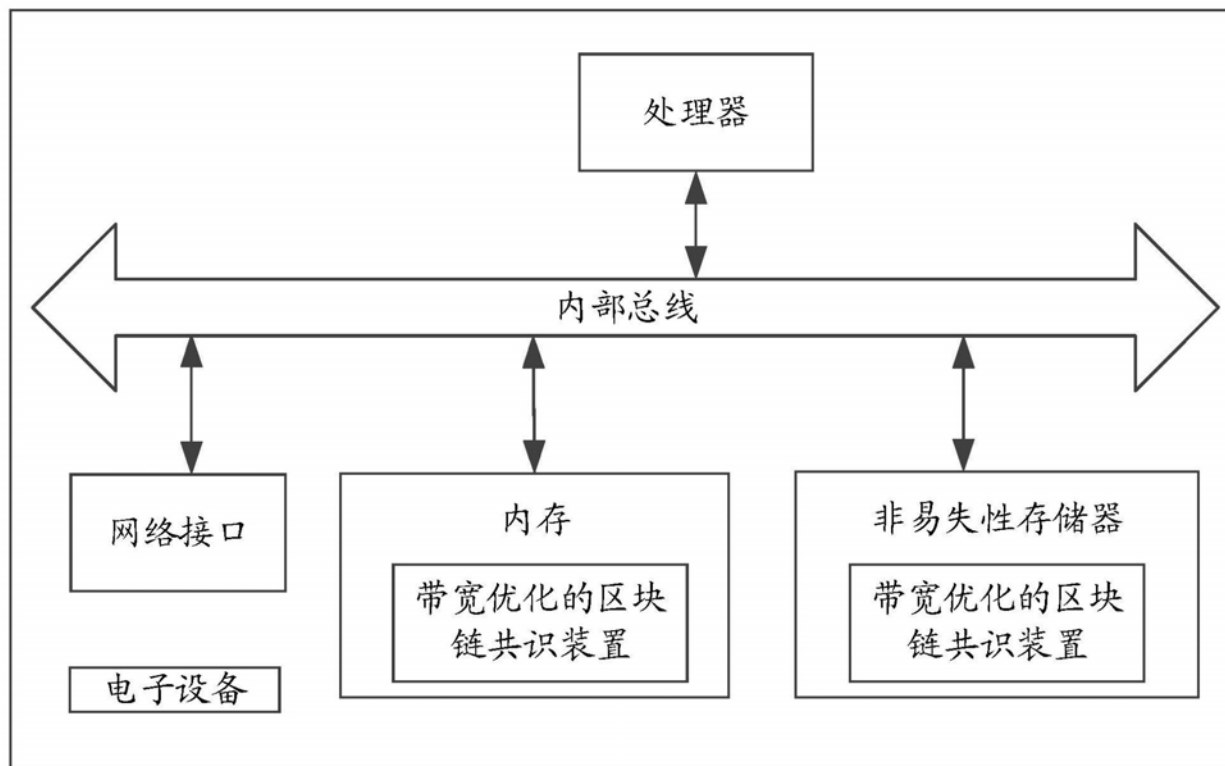


图4

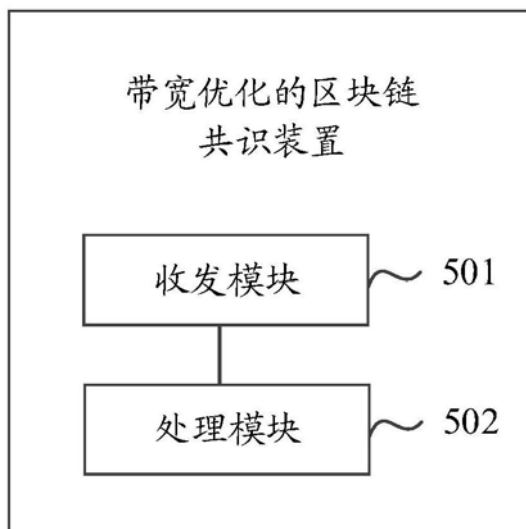


图5