



(21) 申请号 201980060588.0

(22) 申请日 2019.08.29

(65) 同一申请的已公布的文献号
申请公布号 CN 112703499 A

(43) 申请公布日 2021.04.23

(30) 优先权数据
16/135,260 2018.09.19 US

(85) PCT国际申请进入国家阶段日
2021.03.16

(86) PCT国际申请的申请数据
PCT/EP2019/073066 2019.08.29

(87) PCT国际申请的公布数据
W02020/057930 EN 2020.03.26

(73) 专利权人 国际商业机器公司
地址 美国纽约阿芒克

(72) 发明人 R.K.拉曼 K.R.瓦什尼 R.瓦库林
M.辛德 S.L.雷米 E.皮萨达基
N.K.博尔

(74) 专利代理机构 北京市柳沈律师事务所
11105

专利代理师 邸万奎

(51) Int.Cl.
G06F 21/64 (2006.01)
H04L 9/32 (2006.01)
G06N 20/00 (2006.01)

(56) 对比文件
CN 105706086 A, 2016.06.22
JP 2001202391 A, 2001.07.27
JP H10105054 A, 1998.04.24
US 2012222034 A1, 2012.08.30
US 2014281743 A1, 2014.09.18
US 2016299834 A1, 2016.10.13
US 2018152297 A1, 2018.05.31
US 2018205552 A1, 2018.07.19
WO 2017145006 A1, 2017.08.31
WO 2018020944 A1, 2018.02.01

审查员 邢延恒

权利要求书3页 说明书21页 附图20页

(54) 发明名称

用于计算以及可信验证的分布式平台

(57) 摘要

示例操作可以包括以下各项中的一项或多项:获得模拟的数据、识别模拟数据内的检查点,基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的演变状态,以及将所生成的顺序数据结构传送至区块链网络的节点以便包括在数据块的散列链接的链内的一个或多个数据块中。



1. 一种计算系统,包括:

处理器,被配置为接收模拟的数据,识别所述模拟的数据内的检查点,并且基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识相对于所述顺序数据结构中的先前数据结构所述模拟的状态的演变;以及

网络接口,被配置为将所述顺序数据结构传输至区块链网络的节点以便包括在数据块的散列链接的链内的一个或多个数据块中。

2. 如权利要求1所述的计算系统,其中,由所述处理器基于与所述模拟的所述接收到的数据内的相应数据结构对应的检查点的频率来自适应地确定每个数据结构的宽度。

3. 如权利要求1所述的计算系统,其中,所述模拟包括模型的迭代模拟和包括一组输入和输出对的非迭代模拟中的一个。

4. 如权利要求3所述的计算系统,其中,所述处理器进一步被配置为:将来自所述模拟的连续计算状态之中的迭代ID存储在每个数据结构内,并且迭代ID标识与相应数据结构相关联的所述迭代模拟的相应迭代。

5. 如权利要求1所述的计算系统,其中,所述网络接口被进一步配置成从区块链对等节点接收消息,所述消息指示与所述生成的顺序数据结构当中的数据结构相关联的模拟数据的状态已经无效。

6. 如权利要求5所述的计算系统,其中,所述处理器被进一步配置成细化包括在所述被无效的数据结构之中的所述模拟数据的状态以生成更新的数据结构,并控制所述网络接口以将所述更新的数据结构传送给所述区块链对等节点以供验证。

7. 如权利要求1所述的计算系统,其中,所述处理器进一步被配置为模拟一个或多个模型以生成所述模拟数据。

8. 一种方法,包括:

获得模拟的数据;

识别所述模拟数据内的检查点;

基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的状态的演变;以及

将所生成的顺序数据结构传输至区块链网络的节点以便包括在数据块的散列链接的链内的一个或多个数据块中。

9. 如权利要求8所述的方法,其中,基于所述模拟的接收的数据内的相应数据结构对应的检查点的频率来自适应地确定每个数据结构的宽度。

10. 如权利要求8所述的方法,其中,所述模拟包括训练的模型的迭代模拟和处理的一组输入和输出对的非迭代模拟中的一个。

11. 如权利要求8所述的方法,进一步包括将来自所述模拟的连续计算状态之中的迭代ID存储在每个数据结构内,其中迭代ID标识与相应数据结构相关联的迭代模拟的相应迭代。

12. 如权利要求8所述的方法,还包括:从区块链对等节点接收消息,所述消息指示与所述生成的顺序数据结构当中的数据结构相关联的模拟数据的状态已经无效。

13. 如权利要求12所述的方法,进一步包括细化包括在所述被无效的数据结构之中的所述模拟数据的状态以生成更新的数据结构,并控将所述更新的数据结构传送给所述区块链

对等节点以供验证。

14. 如权利要求8的方法,进一步包括执行所述模拟以生成所述模拟的数据。

15. 一种包括指令的非瞬态计算机可读介质,所述指令在由处理器读取时使所述处理器执行一种方法,所述方法包括:

获得模拟的数据;

识别所述模拟数据内的检查点;

基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的状态的演变;以及

将所生成的顺序数据结构传输至区块链网络的节点以便包括在数据块的散列链接的链内的一个或多个数据块中。

16. 一种计算系统,包括:

网络接口,被配置为从迭代模拟的顺序数据结构中接收存储所述迭代模拟的状态数据的数据结构;以及

处理器,被配置为基于来自先前数据结构的先前验证的状态数据来生成所述数据结构的状态数据的本地计算,并且确定所述状态数据的本地计算与所接收的数据结构内的所述状态数据之间的相似性,

其中,响应于确定所述相似性在预定阈值内,所述处理器还被配置为控制所述网络接口以向区块链网络发送所述数据结构的认可,以包括在数据块的散列链接的链当中的数据块内。

17. 如权利要求16所述的计算系统,其中,所述迭代模拟包括模型的深度学习模拟,并且所述顺序数据结构存储所述模型的状态随迭代的变化。

18. 如权利要求16所述的计算系统,其中,所述预定阈值标识所述状态数据的所述本地计算与所述接收到的数据结构内的所述状态数据之间的可接受偏差水平。

19. 如权利要求16所述的计算系统,其中,响应于确定所述相似度在所述预定阈值之外,所述处理器进一步被配置成控制所述网络接口以将指示所述状态数据无效的消息传输至提交所述数据结构的客户端节点。

20. 如权利要求19所述的计算系统,其中,所述网络接口还被配置为接收包括细化的状态数据的更新的数据结构,并且所述处理器被配置为确定所述状态数据的位置计算与所述更新的数据结构内的所述细化的状态数据之间的相似性。

21. 一种方法,包括:

从迭代模拟的顺序数据结构中接收数据结构,所述数据结构存储所述迭代模拟的状态数据;

基于来自先前数据结构的先前验证的状态数据来生成所述数据结构的状态数据的本地计算;

确定所述状态数据的本地计算与所述接收到的数据结构内的所述状态数据之间的相似性;以及

响应于确定所述相似性在预定阈值内,向区块链网络发送所述数据结构的认可,以包括在数据块的散列链接的链当中的数据块内。

22. 如权利要求21所述的方法,其中,所述迭代模拟包括被训练的模型的深度学习模

拟,并且所述顺序数据结构存储所述模型的状态随迭代的变化。

23.如权利要求21所述的方法,其中,所述预定阈值标识所述状态数据的所述本地计算与所述接收到的数据结构内的所述状态数据之间的可接受偏差水平。

24.如权利要求21所述的方法,进一步包括:响应于确定所述相似度在所述预定阈值之外,将指示所述状态数据无效的消息传输至提交所述数据结构的客户端节点。

25.如权利要求24所述的方法,进一步包括:接收包括细化的状态数据的更新的数据结构;以及确定所述状态数据的位置计算与所述更新的数据结构内的所述细化的状态数据之间的相似性。

用于计算以及可信验证的分布式平台

技术领域

[0001] 本申请总体上涉及一种模拟和学习系统,并且更具体地涉及一种去中心化数据库,如区块链,其中,进化的计算状态存储在区块链上,其中,多个节点在对计算的可信确认和验证中共享。

背景技术

[0002] 集中式数据库在一个位置处在一个单个数据库(例如,数据库服务器)中存储和维护数据。这个位置通常是中央计算机,例如,台式中央处理单元(CPU)、服务器CPU、或大型计算机。存储在集中式数据库上的信息通常可从多个不同的点访问。多个用户或客户端工作站可以例如基于客户端/服务器配置同时在集中式数据库上工作。由于其单个位置,集中式数据库易于管理、维护和控制,尤其是出于安全的目的。在集中式数据库内,数据完整性被最大化并且数据冗余被最小化,因为所有数据的单个存储位置还意味着给定的数据集仅具有一个主要记录。这有助于保持数据尽可能准确和一致,并且增强数据可靠性。

[0003] 然而,集中式数据库存在显著的缺点。例如,集中式数据库具有单一故障点。具体地,如果没有容错设置并且发生硬件故障,则数据库内的所有数据丢失并且所有用户的工作被中断。此外,集中式数据库高度依赖于网络连接性。结果,互联网连接越慢,每个数据库访问所需的时间量越长。另一个缺点是,当集中式数据库由于单个位置而经历高业务量时,出现瓶颈。此外,集中式数据库提供了对数据的有限访问,因为该数据库仅维护该数据的一个副本。结果,多个用户可能无法在不产生诸如重写所存储的数据之类的问题的情况下同时访问同一条数据。此外,因为中央数据库系统具有最小的数据冗余到没有数据冗余,所以如果一组数据意外丢失,除了通过手动操作难以从备份磁盘存储器检索它。

[0004] 诸如区块链系统的去中心化数据库提供能够解决集中式数据库的缺点的存储系统。在区块链系统中,多个对等节点存储分布式分类账。客户端与对等节点交互以获得对区块链的访问。对等节点可以由具有不同兴趣的不同实体控制,并且因此不是关于彼此信任的实体。此外,在区块链中没有中央机构。因此,为了以可信方式将数据添加到分布式分类账或在分布式分类账上改变数据,必须发生对等节点的共识。共识提供了一种在不信任对等节点的区块链系统中实现信任的方式。

[0005] 同时,计算和学习系统越来越多地被部署在各种环境中,以便理解、推断、并且最重要的是针对具有大规模隐含的场景以基于策略的规模做出决策。例如,模拟环境可以用于理解疾病的传播,以及不同控制机制在不同地理和人口统计环境中的影响。类似地,深度学习可用于学习系统的底层状态并日益部署在诸如自动驾驶汽车、分析、供应链等日常应用中。

[0006] 作为另一个示例,与诸如流行病学和气象学等领域相关联的计算可能花费数周的运行时间。例如,疟疾数据科学家(MDS)可以使用两种或更多种计算模型代理(例如,OpenMalaria和流行病学建模软件(EMOD))来运行实验以模拟疾病扩散。该模型运行许多不同的模拟以比较模型结构或参数的变化的影响。实验的目标是提出经验证的策略和干预策略以根除疾病。

[0007] 这些基于大模拟的系统通常需要在多个不同方之间共享输入和输出以便在低资源和社交好的上下文中进行及时决策。然而,这些计算和推断中的许多(例如,深度神经网络的训练等)由个体且常常独立的代理在本地进行,从而导致不受信任的交互系统的系统。考虑到此类系统所具有的影响的规模,重要的是由单独代理推断的结果是可信且可转移的。此外,认可可能是错误的、对手的或计算缓慢的。因此,就认可每个帧的时间而言,确认可能是昂贵的。例如,如果认可者是离散者,则获得认可可能被过度延迟。因此,系统中的实体能够信任由单独代理获得的结论的有效性,并且有效地执行这种有效性,以将它们作为组用于进一步的研究和开发是必要的。透明度还保证了在这样的系统中的起源,其中重要的是能够识别错误的来源。

发明内容

[0008] 一个示例性实施例可以提供一种计算系统,该计算系统包括被配置成进行以下各项中的一项或多项的一个或多个处理器:接收模拟的数据,识别所述模拟数据内的检查点,并且基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的状态的演变,以及网络接口,所述网络接口被配置为将所述顺序数据结构传输至区块链网络的节点以便包括在数据块的散列链接的链内的一个或多个数据块中。

[0009] 另一个示例性实施例可以提供一种方法,该方法包括以下各项中的一项或多项:获得模拟的数据,识别所述模拟数据内的检查点,基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的状态的演变,以及将所生成的顺序数据结构传送至区块链网络的节点以便包括在数据块的经散列链接的链内的一个或多个数据块中。

[0010] 另一个示例性实施例可以提供一种包括指令的非瞬态计算机可读介质,当由处理器读取时,使所述处理器执行以下各项中的一项或多项:获得模拟的数据,识别所述模拟数据内的检查点,基于所识别的检查点生成多个顺序数据结构,其中每个数据结构标识所述模拟相对于所述顺序数据结构中的先前数据结构的状态的演变,以及将所生成的顺序数据结构传送至区块链网络的节点以便包括在数据块的经散列链接的链内的一个或多个数据块中。

[0011] 另一个示例性实施例可以提供一种计算系统,该计算系统包括以下中的一个或多个:网络接口,该网络接口被配置为从迭代模拟的顺序数据结构中接收存储迭代模拟的状态数据的数据结构,以及处理器,其被配置为进行以下操作中的一个或多个:基于来自先前数据结构的先前验证的状态数据来生成所述数据结构的状态数据的本地计算,以及确定状态数据的所述本地计算与所接收的数据结构内的所述状态数据之间的相似性,其中,响应于确定所述相似性在预定阈值内,所述处理器还可以被配置为控制所述网络接口以向区块链网络发送所述数据结构的认可,以包括在数据块的散列链接的链当中的数据块内。

[0012] 另一个示例性实施例可以提供一种方法,该方法包括以下操作中的一个或多个:从迭代模拟的顺序数据结构中接收数据结构,所述数据结构存储所述迭代模拟的状态数据,基于来自先前数据结构的先前验证的状态数据生成所述数据结构的状态数据的本地计算,确定所述状态数据的所述本地计算与所接收到的数据结构内的所述状态数据之间的

相似性,以及响应于确定所述相似性在预定阈值内,向区块链网络发送所述数据结构的认可,以包括在数据块的散列链接的链当中的数据块内。

附图说明

- [0013] 图1是示出根据示例性实施例的用于验证和存储模拟数据的区块链网络的示意图。
- [0014] 图2A是示出根据示例性实施例的用于资产共享场景的对等节点区块链架构配置的示意图。
- [0015] 图2B是示出根据示例性实施例的区块链网络的节点之间的通信序列的示意图。
- [0016] 图3是示出根据示例性实施例的被许可的区块链网络的示意图。
- [0017] 图4A是示出根据示例性实施例的客户端节点生成模拟数据帧的过程的示意图。
- [0018] 图4B是示出根据示例性实施例的证实节点验证模拟数据帧的过程的示意图。
- [0019] 图4C是示出根据示例性实施例的排序节点将模拟数据帧安排在块中的过程的图。
- [0020] 图5A是示出了根据示例性实施例的生成模拟数据帧以存储在区块链上的方法的示意图。
- [0021] 图5B是示出根据示例性实施例的对模拟数据帧进行认可的方法的示意图。
- [0022] 图5C是示出根据示例性实施例的分配用于并行认可的模拟数据的连续帧的方法的示意图。
- [0023] 图5D是示出根据示例性实施例的作为并行认可处理的结果排序模拟数据的方法的示意图。
- [0024] 图5E是示出根据示例性实施方式的压缩模拟内容的方法的示意图。
- [0025] 图5F是示出根据示例性实施例的对模拟内容进行认可的方法的示意图。
- [0026] 图6A是示出根据示例性实施例的根据本文描述的一个或多个操作被配置为对区块链执行不同操作的物理基础设施的示意图。
- [0027] 图6B是示出根据示例性实施例的在合约方与被配置为在区块链上实施智能合约条款的中介服务器之间的智能合约配置的示意图。
- [0028] 图6C是示出根据示例性实施例的在合约方与被配置为在区块链上实施智能合约条款的中介服务器之间的智能合约配置的示意图。
- [0029] 图6D是示出根据示例性实施例的另一示例性基于区块链的智能合约系统的示意图。
- [0030] 图7A是示出根据示例性实施例的新块被添加到区块链分类账的过程的示意图。
- [0031] 图7B是示出根据示例性实施例的用于区块链的数据块结构的内容的示意图。
- [0032] 图8是示出根据示例性实施例中的被配置为一个或多个的示例性计算机系统的示意图。

具体实施方式

[0033] 将容易理解的是,如在本文中的附图中一般地描述和展示的即时部件可以以多种多样的不同配置来安排和设计。因此,如附图所示,对方法,设备,非暂时性计算机可读介质和系统中的至少一个的实施例的以下详细描述并非旨在限制所要求保护的本申请的范围,而仅仅是代表所选实施例。

[0034] 在整个说明书中描述的即时特征、结构或特性可以在一个或多个实施例中以任何

合适的方式组合。例如,在整个说明书中,短语“示例性实施例”,“一些实施例”或其他类似语言的使用是指以下事实:结合该实施例描述的特定特征、结构或特性可以包括在以下至少一个实施例中。因此,在整个说明书中,短语“示例性实施例”,“在一些实施例中”,“在其他实施例中”或其他类似语言的出现不一定全都指同一组实施例,并且所描述的特征、结构或特性在一个或多个实施例中,可以以任何合适的方式组合。

[0035] 另外,尽管在实施例的描述中可能已经使用术语“消息”,但是本申请可以应用于许多类型的网络数据,诸如分组、帧、数据报等。术语“消息”还包括分组、帧、数据报及其任何等同形式。此外,尽管在示例性实施例中可以描绘某些类型的消息和信令,但是它们不限于特定类型的消息,并且本申请不限于特定类型的信令。

[0036] 大规模计算实验可以用于通过建立机器学习模型、深度学习系统、预测分析等来做出策略级决策。这些计算通常源自多个独立组织的个体研究和开发的融合。然而,这些组织不一定彼此信任。此外,每个组织通常维护其自己的模型和数据的本地版本,其可在不同代理之间创建模型中的移位的数据。示例性实施例提供方法、系统、非瞬态计算机可读介质、设备和/或网络,其提供支持分布式计算环境的区块链网络,该区块链网络在可以与不同组织和实体相关联的节点之间建立信任。本文所述的系统还建立起源,使得每个实体能够跟踪决策源。该系统还提供可计量性和透明度,其对于多代理系统(诸如分布式计算和系统上下文的系统)是关键的。系统通过保证一致的本地计算,被同意为正确的,以及通过经由记录的审核跟踪计算一致性来验证计算来确保对计算的验证。

[0037] 去中心化数据库是包括彼此通信的多个节点的分布式存储系统。区块链是去中心化数据库的示例,其包括类似于能够在相互不信任方之间维护记录的分布式分类账的仅附加不可变数据结构。不可信方在本文中被称作对等体或对等节点。每个对等体维护数据库记录的副本,并且没有单个对等体能够修改数据库记录而没有在分布式对等体之间达到一致性。例如,对等体可执行一致协议以验证区块链存储事务,将存储事务分组成块,并在块上构建散列链。如有必要,该过程通过对存储事务进行排序而形成分类账以便一致。在公共的或无权限的区块链中,任何人都可以在没有特定身份的情况下参与。公共区块链通常涉及本地加密货币并且基于工作证明(PoW)使用一致性。另一方面,许可的区块链数据库提供可以保护实体组之间的互动的系统,所述实体组可以共享共同的目标但是可能不完全彼此信任,诸如交换资金、商品、信息等的商家。

[0038] 区块链操作针对去中心化存储方案定制的任意可编程逻辑并且被称为“智能合约”或“链码。”在一些情况下,可存在用于管理功能和称为系统链的参数的专用链。智能合约是可信的分布式应用,其利用区块链数据库的防篡改属性和节点之间的被称为认可或认可策略的底层协定。通常,区块链事务在提交给区块链之前通常必须被“认可”,而未认可的事务被忽略。典型的认可策略允许链码以认可所需的一组对等节点的形式指定交易的认可者。当客户端向认可策略中指定的对等体发送交易时,执行该交易以验证交易。在验证之后,交易进入排序阶段,其中使用一致协议来产生分组成块的认可交易的排序序列。

[0039] 节点是区块链系统的通信实体。在不同类型的多个节点可以在同一物理服务器上运行的意义上,“节点”可以执行逻辑功能。节点被分组在信任域中并且与以不同方式控制它们的逻辑实体相关联。节点可以包括不同类型,诸如向认可者(例如对等体)提交事务调用的客户端或提交客户端节点,并且向订购服务(例如订购节点)广播事务提议。另一类型

的节点是可接收客户端提交的事务、提交事务并维护区块链事务的分类账的状态和副本的对等节点。对等体还可以具有认可者的角色,尽管这不是必须的。订购业务节点或订购者是为所有节点运行通信业务的节点,并且其实现递送保证,诸如在提交事务和修改区块链的世界状态时向系统中的每个对等节点广播,其为初始区块链事务的另一名称,其通常包括控制和设置信息。

[0040] 分类账是区块链的所有状态转换的排序的、防篡改的记录。状态转换可由参与方(例如,客户端节点、排序节点、认可方节点、对等节点等)提交的区块链调用(即,事务)引起。事务可导致一组资产键-值对作为一个或多个操作数被提交给分类账,诸如创建、更新、删除等。分类账包括区块链(也称为链),其用于在块中存储不可变的、排序的记录。分类账还包括维护区块链的当前状态的状态数据库。每个通道通常有一个分类账。每个对等节点维护它们作为其成员的每个通道的分类账的副本。

[0041] 链是被结构化为散列链接的块的事务日志,并且每个块包含N个事务的序列,其中,N等于或大于1。块头部包括块事务的散列,以及先前块头部的散列。以此方式,分类账上的所有事务可被排序并加密地链接在一起。因而,不可能在不破坏散列链接的情况下篡改分类账数据。最近添加的区块链块的散列表示之前已经到来的链上的每个事务,使得可以确保所有对等节点处于一致和可信状态。链可存储在对等节点文件系统(即,本地、附加存储、云等)上,从而有效地支持区块链工作负荷的仅附加性质。

[0042] 不可变分类账的当前状态表示包括在链事务日志中的所有键的最新值。因为当前状态表示通道已知的最新键值,所以有时将其称为世界状态。链码调用针对分类账的当前状态数据执行事务。为了使这些链码交互有效,可以在状态数据库中存储密钥的最新值。状态数据库可以简单地是链的事务日志中的索引视图,因此可以在任何时间从链重新生成状态数据库。状态数据库可在对等节点启动时和在事务被接受之前自动恢复(或如果需要的话生成)。

[0043] 本文所描述和描绘的即时解决方案的一些益处包括将迭代模拟的经验证的状态存储在迭代帧中。即,状态由客户端通过迭代来计算,在帧中压缩,并且由认可者验证。在验证时,这些帧被顺序地添加到区块链。在枚举实验(给定输入的输出的非迭代评估)的情况下,区块链可存储输入-输出对的经验证的帧。随着系统在迭代上演进,系统的状态可被存储在区块链上的迭代帧中。帧由开始处的检查点来表征并且自适应地定义以包括状态,只要更新(差异)在定义的容差阈值内。可适当地选择帧的最大大小,且每一帧可包含与所述大小所允许的一样多的状态。此外,由认可者执行的验证被并行化,使得帧可被并行验证。非迭代实验也可以被并行化,而迭代模拟由客户端顺序地运行。

[0044] 此外,示例性实施例还提供用于开销减少的成帧和压缩框架。可基于模拟是迭代的还是非迭代的来不同地压缩每一帧。该系统还提供无效状态(其未被认可)的连续细化,以校正参数不可知设计的近似误差。对于非迭代模拟,可以执行用于枚举(非迭代)模拟的基于最小生成树(MST)的帧构造以用于压缩。该系统还提供用于验证时间减少的编码的基于计算的任务分配,以及使用概率验证保证用于具有外部随机性的模拟的验证。

[0045] 区块链与传统数据库的不同之处在于区块链不是中央存储装置,而是去中心化的、不可变和安全存储装置,其中节点必须共享存储装置中的记录的改变。区块链中固有的并且帮助实现区块链的一些特性包括但不限于本文进一步描述的不可变分类账、智能合

约、安全、隐私、去中心化、共识、认可、可及性等,本文将进一步描述。根据各个方面。由于不可变性,安全性,去中心化,共识,认可和可访问性是区块链固有和独特的,因此可以通过分布式平台实现计算过程的验证状态。

[0046] 具体地,因为经验证的状态的记录是不可变的,所以针对计算的一致性的模拟的验证转换成检查区块链上的散列链的一致性。这比运行整个模拟容易得多,并且因此使得验证易于保证。可计量性可由区块链保证,因为不仅跟踪计算的来源,而且识别计算中的错误及其来源是可行的。这在计算系统中建立起源。

[0047] 该系统进一步提供来自计算、认可和存储中的对手的安全性,因为状态一旦被验证,就可以通过参考区块链在任何稍后的时间点被信任。这允许跨模拟结果的使用实例转移的信任。此外,因为验证、计算和存储的整个过程是去中心化的,所以不存在破坏模拟结果的中央攻击点。由于过程的并行性,分布式本质还允许计算成本和时间的减少。

[0048] 对等体之间的确认一致性对于实现信任是关键的,因为至少足够分数的对等体将计算接受为正确以及还有存储在分类账的副本上的数据跨对等体是最新的是必要的。此外,网络中的独立认可者对计算步骤的认可允许在验证处理中的信任,因为网络中的认可者的随机指派暗示了通过冗余来否定对手对等体的影响,并且网络的整体性质恰当地起作用。因为区块链分类账是分布式的,并且在所有对等体之间共享,所以希望访问模拟的结果或从迭代中得出推断的任何对等体能够通过参考该分类账的它们的本地副本来容易地做到这一点。

[0049] 示例性实施例提供了优于传统数据库的许多益处。例如,系统的设计需要能够以一致的方式在网络中的对等端之间共享、是不可变的并且能够以分布式方式更新的数据结构。所有这些都精确地转换成区块链。另外,区块链还允许转换成模拟有效性的可验证性的简单一致性检查。然而另一个数据结构提供这些特征在技术上可行,不存在提供所有上述要求的单个统一机制。在这个意义上,该区块链被证明是最佳选择。

[0050] 区块链可结合压缩模式使用,所述压缩模式首先执行经验证的状态的新颖压缩以允许实施方式的可缩放性。在没有压缩模式的情况下,存储和通信开销变得过大,从而妨碍了设计向大规模模拟和大型网络的缩放。此外,这里定义的验证过程在与可应用于密码货币和智能合约的认可的的概念相比较时是新的,因为它基于函数的重新计算以测试报告偏差,与诸如检查资源可用性之类的东西。

[0051] 示例性实施例还改变数据可如何被存储在区块链的块结构内。例如,相关的区块链可以存储诸如事务(属性的转移)和智能定约(要执行的指令集)的信息。同时,在示例性实施例中,所存储的数据是计算过程的经验证的状态。即,每个块由一个或多个帧组成,所述一个或多个帧包括计算状态的压缩检查点和从帧中的连续迭代的增量编码获得的量化状态更新。可以存储的附加元数据可以包括执行模拟的客户端的ID、验证状态的认可者、迭代索引和用于执行模拟的其他无关信息。通过将计算的经验证的状态存储在区块链的数据块内,计算的经验证的状态可通过块的散列链接的链被附加到不可变的分类账,并且可被用来验证计算的后续迭代以及执行压缩。

[0052] 由在本文描述的示例性实施例提供了不同技术优点。例如,实施例通过向并行验证状态的认可者的子集报告模拟的状态来提供用于大规模迭代模拟的分布式验证框架。另外,提供压缩模式,其识别检查点、构造模拟状态的帧,且执行增量编码和晶格向量量化以

在分派之前并行地压缩所述帧以用于验证。此外,提供了区块链分类账,用于存储来自使用排序者检查认可一致性的模拟的经验证的、压缩的状态帧。并且,在枚举实验中提供了一种基于MST的帧构造方法,用于创建状态树进行压缩。此外,提供编码的基于计算的认可分配以减少用于状态确认的计算时间。

[0053] 图1示出根据示例性实施例的用于验证和存储模拟数据的区块链网络100。参照图1,区块链网络100包括客户端节点110,该客户端节点可执行用于训练模型的大规模计算或者可从另一个系统或多个系统接收大规模计算数据。大规模计算可以生成用于训练模型/数据或用于计算非迭代过程的模拟数据。基于模拟数据内的检查点,模拟数据可被转换成数据结构,诸如帧、数据点等,并且数据结构可被压缩以减少开销。

[0054] 可以使用检查点来识别下一帧等。检查点可由客户端节点基于迭代数据、非迭代输入/输出对等来确定。随着模型状态的演变,帧可包括连续的数据帧。区块链网络100还包括认可节点120、130和140的多个子集,这些子集可彼此互斥,并且可用于并行验证模拟状态数据的帧或其他数据结构。当数据帧由认可节点子集120、130和/或140之一验证时,该帧可被发送到排序器150以包括在区块链160(在本文中也称为块的散列链接链)之中的数据块内。

[0055] 模拟可以是受益于多个计算节点的实验。作为非限制性实例,疟疾数据科学家(MDS)可以使用一种或多种计算模型代理(如OpenMalaria或流行病学建模软件(EMOD))来运行实验以模拟疾病的传播。在该示例中,模型运行许多不同的模拟以比较模型结构或参数中的变化的影响。实验的目标是提出经验证的策略和干预策略以根除疾病。本文的系统可以用于在模拟的计算结果中建立可计量性和透明度。

[0056] 再次参考图1,由认可节点120、130和140的子集执行分布式验证。用于认可的本地操作的重新计算的一致性可以由子集和报告的计算生成。一旦被子集验证,经验证的数据帧可存储在区块链160上并且在区块链网络上的所有节点之间共享,所述区块链网络可包括客户端节点110、认可节点(子集120-140)、其他对等节点等。区块链160在连续的检查点之间创建经验证的帧的共享的、不可变的、仅附加记录。此外,可使用有损压缩来压缩每一帧内的状态数据,借此减少存储成本和计算开销。

[0057] 客户端节点110还可以被称为执行计算的代理。客户端节点110可将模型的状态、数据等以帧的形式存储在检查点处。所述帧可包含帧内的压缩状态更新(差异)和先前帧内的状态数据。该帧可以是作为连续帧生成的较大帧组的一部分。这些帧可由认可节点子集120、130和40内的认可者(其他代理)通过重新计算来验证。经验证的帧可被添加到在区块链网络的认可者、对等体、客户端等之间共享的区块链160。

[0058] 系统的状态随着模拟的迭代演进为 $X_{t+1} = f(X_t, \theta_t)$

[0059] $X_t \in \mathbb{R}^d$ 系统状态向量

[0060] $\theta_t \in \mathbb{R}^{d'}$ 是(可能是随机的)信息的共享源

[0061] $f: \mathbb{R}^d \times \mathbb{R}^{d'} \rightarrow \mathbb{R}^d$ 原子操作,所有对等体共享

[0062] 模拟的有效性 \iff 中间状态的有效性 $\{X_1, X_2, \dots\}$

[0063] 假设 $f(\cdot)$ 是L-Lipschitz连续的,即

[0064] $\|f(x_1) - f(x_2)\| \leq L \|x_1 - x_2\|$

[0065] 模拟可以是状态的迭代演变 $X_{t+1} = f(X_t, \theta_t)$ 。客户端节点110可报告检查点处 $t \in \{T_1, T_2, \dots\}$ 的状态 X_t 。针对检查点之间的迭代, $t \in (T_1, T_2)$, 报告经压缩的更新 $\widetilde{\Delta X}_t$ 。作为响应, 节点的告知子集可基于客户端节点110所报告的与接收到的帧进行比较的先前经验证的状态来重新计算数据帧。此外, 排序器150可以顺序方式将经验证的帧添加到区块链160。

[0066] 此外, 客户端节点110可使用有损压缩 (类似于视频编码过程) 来报告状态更新 (差异), 对于小的 ϵ , $\|\widetilde{\Delta X}_t - \Delta X_t\| \leq \epsilon$ 。这里, 帧宽度 (检查点频率) 可被自适应地确定, 使得对于帧中的所有时隙, $\|\Delta X_t\| \leq \Delta_{\text{quant}}$, 并且帧中的迭代的数量可被最大数量限制。可使用类似于Lempel-Ziv的有损压缩器将当前状态存储在帧的开始 (检查点) 处。基于压缩率, 通信和存储的成本不同。如果模拟是混沌模拟, 则系统可能需要更小的帧和更大的压缩率, 然而, 实施例不限于此。

[0067] 在图1的网络100中, 实施认可并行化, 因为单独的帧可以由不相关的认可者子集并行验证。在此, 每个认可者仅验证从模拟生成的总帧的一部分, 从而显著地减少计算开销。此外, 认可者不需要等待具有大通信延迟或不良计算能力的对等体, 并且可能集中于他们曾被指派认可的帧。当认可子集不能验证数据帧时, 该认可子集可通知客户端节点110。如果确信, 则客户端节点110可在不重新计算的情况下使用连续细化来传达报告更新。否则, 客户端节点110可重新计算帧并将其发送到子集以供再次认可。

[0068] 为了确定帧认可, 认可对等节点可确定用于模拟的可接受的偏离余量 Δ_{marg} 。因此, 如果由客户端节点110报告的模型/模拟的状态在重新计算的状态的可接受偏差界限内, 则计算可仅被声明为有效的。对等体的子集认可帧中的原子计算并在一致时添加到区块链。此外, 可以自适应地减小裕度, 并且适当地更新压缩方案以获得更接近模拟终止 (或状态收敛) 的更严格的要求。

[0069] 排序器150可从认可节点120、130和140的不同子集接收针对不同帧的认可。认可器150可在区块链160上生成或以其他方式初始化数据块以存储。这里, 认可可在一致时顺序地排序帧以添加到区块链160。所存储的帧是不可变的, 并且可用于由认可节点验证检查点和帧。帧中的状态更新也可以由排序器150进行子采样, 并且以较少的存储成本进行存储。

[0070] 在一些实施例中, 计算涉及迭代计算, 如深度学习模型的细化、机器学习算法等。作为另一实例, 实验可涉及非迭代模拟, 其包含使用原子计算块且对每一输入产生的输出的大组输入参数。在此实例中, 模拟的状态可为输入-输出对, 其不提供使压缩困难的到帧中的自然分组。

[0071] 然而, 帧可以经由最小生成树 (MST) 来构建并且随后被压缩。此处, 可执行构建有效增量编码和状态更新的压缩报告所必需的状态更新的帧。在这个实例中, 可以利用成对距离矩阵, $\mathcal{W} = [\|Z_i - Z_j\|]_{i,j}$, 其中, $Z_i = (X_i, Y_i)$ 。客户端节点110可以生成加权图 $G([n], \mathcal{W})$ 的MST。阈值边由 Δ_{quant} 加权重, 并且修剪树以每个最多包括M个节点。此外, 每个树可对应于帧, 并且可包括用树的根创建检查点, 并沿着树的边缘传送经量化的状态更新。此外, 可以使用与针对迭代数据所描述的相同的压缩方案。

[0072] 在非迭代模拟数据的示例中, 客户端节点110可以报告将状态与协商子集相关的

压缩帧和MST结构。作为响应,认可节点可解压缩状态 $(\tilde{X}_i, \tilde{Y}_i)$ 。认可者可以从所报告的输入 $\hat{Y}_i = f(\tilde{X}_i)$ 重新计算输出,并且如果 $\|\hat{Y}_i - \tilde{Y}_i\| \leq \Delta_{\text{marg}}$,则确认帧中的状态报告。此外,如果所有状态都是有效的,则认可者可以验证帧。当计算(输入)的数量大时,帧构造、本地确认和认可并行化降低成本并确保计算有效性。

[0073] 模拟通常对随机性 (θ_t) 的外部源(即, $X_{t+1} = f(X_t, \theta_t)$)敏感。在没有访问随机性的情况下,各个认可者不能重新计算报告的状态。一种可能的解决方案,在每次迭代时存储 θ_t ,这产生了非常高的存储成本。作为另一个实例,通过测试报告的状态与由独立认可者重新计算的状态的平均值的偏差来验证。这里,如果随机性源相同,对于表现良好的功能,认可者沿着预期路径共同地表现

[0074] 本文的描述假设认可者诚实、并且用于及时验证的计算同质性。然而,如果认可者是错误的、对抗的或计算缓慢的,则就认可每个帧的时间而言,验证可能是昂贵的。例如,如果认可者是离散者,则获得认可可能被过度延迟。对于使用非信任对等体的分布式信任,需要通过分配用于每个状态确认的多个认可者的确认中的冗余。因而,可以根据系统中的对抗对等体的数量来定义认可策略(诸如至少必要的 τ 验证)。

[0075] 在一些实施例中,使用用于分配将由帧中的认可者验证的状态的编码计算可以减少由每个认可者执行的计算的数量。在这样做时,系统减少了在阈值认可策略下验证的时间,但是使用更多的认可者。

[0076] 本文描述的系统为传统的基于模拟的系统提供了多个优点。例如,可计量性被保证,因为环境保证在设计和用于跟踪多代理系统中的本地计算的方法中的起源。透明度是通过以常规审计和经由区块链的本地计算验证的形式的透明度通过在节点之间建立的信任来创建的。可以根据状态演进来适配框架设计、认可和验证。这里,通过改变验证裕度,可以适当地改变信任要求。平台使用相当通用的元件,并且可以使用各种各样的设计参数和算法中的任何一种来实现。此外,该设计对于模拟的细节是不可知的并且可以被实现,只要模拟可分解成可再现的原子计算即可。该系统需要相当简单的构建块来创建平台,该平台可以从现有的压缩和区块链技术适当地开发。此外,通过存储系统状态的中间评估,平台保证可靠的数据和模型共享以及协作研究。

[0077] 图2A示出根据示例实施方式的区块链架构配置200。参考图2A,区块链架构200可以包括某些区块链元件,例如,一组区块链节点202。区块链节点202可以包括一个或多个节点204-210(仅以示例描绘了四个节点)。这些节点参与多个活动,诸如区块链事务添加(例如,模拟数据帧等)和验证过程(一致性)。区块链节点204-210中的一个或多个可基于认可策略认可事务,并且可为架构200中的所有区块链节点提供排序服务。区块链节点可发起区块链认证并寻求写入存储在区块链层216中的区块链不可变分类账,其副本也可存储在构成的物理基础结构214上。区块链配置可包括链接到应用编程接口(API) 222以访问和执行所存储的程序/应用程序代码220(例如,区块链、智能合约等)的一个或多个应用程序224,所存储的程序/应用程序代码220可根据参与者寻找的定制配置来创建并且可维持他们自己的状态,控制它们自己的资产,并且接收外部信息。这可被部署为事务并经由附加到分布式分类账来安装在所有区块链节点204-210上。

[0078] 区块链基础或平台212可以包括不同层的区块链数据、服务(例如,密码信任服务、虚拟执行环境等)和底层物理计算机基础设施,其可以用于接收和存储新事务并且提供对寻求访问数据条目的审核者的访问。区块链层216可暴露接口,该接口提供对处理程序代码和占用物理基础设施214所必需的虚拟执行环境的访问。密码信任服务218可以用于验证诸如资产交换事务之类的事务并保持信息私密。

[0079] 图2A的区块链架构配置可以经由区块链平台212所暴露的一个或多个接口和所提供的服务来处理 and 执行程序/应用程序代码220。代码220可控制区块链资产。例如,代码220可以存储和转移数据,并且可以由节点204-210以智能合约和关联链码的形式与条件或就其执行的其他代码元素来执行。作为非限制性实例,可以创建智能定约以执行就改变、更新等的提醒、更新和/或其他通知。智能合约本身可以用于识别与分类账的授权和访问要求和使用相关联的规则。例如,读取集合226可由包括在区块链层216中的一个或多个处理实体(例如,虚拟机)来处理。写入集合228可以包括对键值的改变。物理基础结构214可用于检索本文所述的任何数据或信息。

[0080] 在链码内,可以经由高级应用和编程语言创建智能合约,并且然后将其写到区块链中的块。智能合约可以包括向区块链(例如,区块链对等体的分布式网络)注册、存储和/或复制的可执行代码。事务是智能合约代码的执行,其可以响应于满足与智能合约相关联的条件来执行。智能合约的执行可以触发对数字区块链分类账的状态的可信修改。由智能合约执行引起的对区块链分类账的修改可通过一个或多个一致协议贯穿区块链对等体的分布式网络被自动复制。

[0081] 智能合约可以以键-值对的格式将数据写入区块链。此外,智能合约代码可以读取存储在区块链中的值并且在应用操作中使用它们。智能合约代码可以将不同逻辑操作的输出写入到区块链中。该代码可用于在虚拟机或其他计算平台中创建临时数据结构。写入到区块链的数据可以是公共的和/或可以被加密并维持为私有的。由智能合约使用/生成的临时数据由所提供的执行环境保存在存储器中,然后一旦区块链所需的数据被识别就被删除。

[0082] 链码可包括智能合约的代码解释,具有附加特征。如本文所述,链码可以是部署在计算网络上的程序代码,其中链码在一致性过程期间由链验证器一起执行和验证。链码接收散列并从区块链检索与通过使用先前存储的特征提取器创建的数据模板相关联的散列。如果散列标识符的散列和从所存储的标识符模板数据创建的散列匹配,则链码向所请求的服务发送授权密钥。链码可以将与密码细节相关联的数据写入区块链。

[0083] 图2B示出根据示例实施方式的区块链网络的节点之间的通信序列250的示例。在图2B的示例中,客户端节点260执行计算以生成存储在帧中的模拟数据。而且,模拟数据帧分布在多个认可对等节点271、272和273之间。当认可对等节点271、272和273验证这些帧时,这些帧被转移到排序节点274,排序节点274将被认可的帧排序成一个或多个块,并将经排序的块传送给区块链网络的节点以供存储在跨节点复制的分布式分类账上。

[0084] 参照图2B,在281中,客户端节点260将第一帧N传输至认可对等节点的第一子集中包括的认可对等节点271。同样,在282中,客户端节点260将第二帧N+1传输至包括在认可对等节点的第二子集中的认可对等节点272。此外,在283中,客户端节点260将第三帧N+2发送到对等节点273。帧N、N+1、N+2等可由客户端节点260或将模拟数据提供给客户端节点260的

另一系统生成。每个帧可包括在由帧N、N+1和N+2中的每个帧表示的每个步骤或阶段被模拟的模型/数据的状态。作为响应,客户端节点260可识别模拟数据内的检查点,压缩数据,并基于识别的检查点生成数据帧N、N+1、N+2等。

[0085] 在284中,认可对等节点271尝试验证帧N。同样,在285中,认可对等节点272尝试验证帧N+1,并且在286中,认可对等节点273尝试验证帧N+2。这里,可以并行执行284、285和286中的验证。为了执行认可过程,每个认可对等节点271-273可识别先前的模拟的验证的状态,重新计算模拟以生成模型的本地状态,并将本地计算的状态与从客户端节点260接收的相应帧中包括的状态进行比较。响应于所计算的状态在与所接收状态在可接受偏差范围内,该认可对等节点确定该帧是有效的,并且认可该帧。

[0086] 在该示例中,认可对等节点272和认可对等节点273分别确定帧N+1和N+2是有效的。因而,在289和290中,认可对等节点272和273向排序节点274传送认可。同时,认可对等节点271基于帧N在对应于帧N的检查点处模型的状态的本地计算来确定帧N是无效的。这里,认可对等节点271确定本地计算状态与帧N中包括的接收状态之间的差大于可接受偏差范围。因而,在288,认可对等节点271拒绝状态报告并向客户端节点260发送通知。作为响应,在一些情况下,客户端节点260可在无需重新计算的情况下细化模拟,或者它可重新计算模拟的状态。在289,客户端节点260基于更新/细化状态将帧N重新提交给认可对等节点271。作为响应,在291,认可对等节点271在289尝试验证接收到的经更新状态,并确定验证帧N。因而,在292中,认可对等节点271向排序节点274发送帧N的认可。

[0087] 响应于接收所有连续帧N、N+1、N+2等的认可,排序节点274基于原始帧的时间戳对帧进行排序。例如,排序节点274可将帧存储在队列中等。此外,在293中,包括帧N、N+1和N+2的经排序的帧数据可被存储在数据块中,并且在294中,被传送到认可节点271、272和273,使得在295中,包括经排序的帧数据的数据块可被提交到分布式分类账。这里,数据块可被提交到分布式分类账的复制副本(例如,区块链、世界状态DB等),其在认可对等节点271-273(以及可能的其他节点,诸如客户端节点260、排序节点274和未示出的其他节点)之间被分布和共享。

[0088] 图3示出了许可的区块链网络300的示例,该许可的区块链网络具有分布式、去中心化的对等架构和管理用户角色和许可的证书权威机构318的特征。在该示例中,区块链用户302可向被许可的区块链网络310提交事务。在该示例中,事务可以是部署、调用或查询,并且可以通过利用SDK的客户端应用、直接通过REST API等来提交。可信商业网络可以提供对诸如审核者(例如,美国股票市场中的安全和交换委员会)之类的监管机构系统314的访问。同时,节点的区块链网络运营商系统308管理成员许可,诸如将调节系统310登记为“审核者”并且将区块链用户302登记为“客户端”。审核者可仅限于查询分类账,而客户端可被授权部署、调用和查询特定类型的链码。

[0089] 区块链开发者系统316写入链码和客户端应用。区块链开发者系统316可通过REST接口将链直接部署到网络。为了将来自传统数据源330的凭证包括在链码中,开发者系统316可使用带外连接来访问数据。在该示例中,区块链用户302通过对等节点312连接到网络。在进行任何事务之前,对等节点312从证书颁发机构318检索用户的登记和事务证书。在一些情况下,区块链用户必须拥有这些数字证书以便在被许可的区块链网络310上进行交易。同时,可能要求尝试驱动链的用户验证他们在传统数据源330上的凭证。为了确认用户

的授权,链码可以使用通过传统处理平台320到该数据的带外连接。

[0090] 图4A示出根据示例性实施例的客户端节点410生成模拟的状态的帧412的过程400A。参照图4A,客户端节点410可执行深度学习模型、大规模非迭代实验等,其生成多个连续帧412。可基于由客户端节点410在模拟数据内识别的检查点413生成每个帧412。此外,帧412可以被压缩以生成压缩的数据帧414。

[0091] 根据不同实施例,模拟可以被分解成更简单、易于验证的步骤/迭代,这可以是跨网络中的对等体执行分布式验证的组成部分。特别地,分解允许多个帧在独立的认可之间被并行验证,允许验证时间要求的减少。因为分解允许验证与模拟过程一起执行,所以不仅是所执行的计算的同时验证,而且创建了信任保证或信任点,该信任保证或信任点可转移到稍后的时间点,在该时间点模拟可能被引用以用于后续帧的认可。

[0092] 模型的状态数据的步骤序列可一起分组成帧并被附加到区块链。然而,此不一定限制实施方式以为每一帧添加单独块。在不失一般性的情况下,可以使用诸如Merkle树结构的想法来对单个块内的多个帧进行分组,只要按(迭代的)顺序添加帧即可。类似于视频压缩,可以使用增量编码、连续细化和/或向量量化来压缩模拟数据帧,以减少存储需要的数据量和执行通信的开销。模拟或计算可以是迭代或非迭代的。例如,迭代情况考虑构造帧并执行增量编码的简单方式。同时,在非迭代模拟的情况下,可以使用基于最小生成树的策略来根据接近度对状态进行分组,从而允许有效的压缩。

[0093] 根据各种实施例,检查点413可在多个不同场景下生成或确定。例如,如果帧包括预设的最大数量的状态,则可以创建新的帧(其中第一状态为检查点)以便确保每个认可者最多验证每个帧中的固定数量的状态。作为另一示例,当连续状态偏离超过所选最大值的量级时,可以创建检查点。即,当状态相差很大程度时,创建检查点。

[0094] 在生成压缩帧414之后,各个压缩帧414可分布在认可对等节点的多个子集中以供验证。这里,认可对等节点的子集可包括至少一个认可对等节点。通过跨多个子集分配认可,可并行执行认可过程(包括解压缩、重新计算、偏离确定等),从而显著减少处理时间。此外,每个认可节点不必计算/验证每个帧。相反,在从区块链检索可信验证数据的同时,认可节点可仅认可帧的一部分。

[0095] 图4B示出根据示例性实施例的证实压缩的模拟数据帧422的认可节点420的过程400B。在此实例中,压缩数据帧可包含模拟的状态信息。该状态是继续计算的迭代所需的中间值。例如,在流行病学模拟的情况下,该状态可以包括描述疾病扩散的一组值。在训练深度神经网络的情况下,例如,该状态可以是应用于神经网络的节点的所有权重的集合等。在一些实施例中,训练数据可能已经预先传输到所有对等体。

[0096] 在认可过程424期间,每个认可对等节点可解压缩帧422以揭示从客户端提供的模拟的状态数据。认可对等节点420可从区块链检索模拟的先前状态。这里,先前状态可以是已被成功验证并被提交给区块链的最新状态,该先前状态可被本地存储在认可对等节点420上。认可对等节点420可重新计算对应于被压缩的帧422中的被接收状态的模拟的当前状态,并确定本地生成的状态是否足够类似于被压缩的帧422中的状态以验证用于认可的帧422。如果状态值相差超过预定阈值,则认可对等节点420可向客户端节点发送指示认可被拒绝的通知。作为响应,客户端节点可细化帧422的状态,且在基于细化更新状态之后重新提交帧422。可重复此过程直到帧422经验证为止。同时,一旦被验证,经验证的帧426可被

提交至排序节点以被包括在区块链上。

[0097] 图4C示出根据示例性实施例的排序节点430以块布置模拟数据帧的过程400C。参照图4C,多个帧由认可对等体的多个子集验证,并提供给排序节点430。作为响应,排序节点430可基于时间戳或当帧由客户端生成时添加的帧内的其他信息,在队列432内以连续顺序(或连续顺序,诸如1、2、3等)排列帧。此外,所安排的帧可在一个或多个数据块434内排序,这些数据块可被发送到区块链网络的节点并被提交给在节点之间共享的分布式分类账。

[0098] 图5A示出根据示例性实施例的生成模拟数据帧以存储在区块链上的方法510。例如,方法510可由区块链的对等节点(诸如客户端节点等)来执行。作为另一示例,该方法可由具有处理器和存储的任何计算设备(诸如服务器、数据库、云平台、多个设备等)来执行。参照图5A,在511中,方法可包括获得模拟的数据。例如,可以执行诸如迭代计算或非迭代计算的模拟,并且可以生成模拟数据。在一些实施例中,模拟可由向对等节点传送模拟数据的另一设备等执行。

[0099] 在512中,该方法可包括识别模拟数据内的检查点,并且在513中,该方法可包括基于识别的检查点生成模拟数据的多个连续帧,其中,每个帧标识相对于连续帧中的先前帧的模拟的演变状态。例如,可以基于模拟的迭代、非迭代模拟的输入/输出对等来识别检查点。例如,如果模拟数据的帧包括预设的最大数量的状态,则创建新的帧(其中第一状态为检查点)以便确保每个认可者最多验证每个帧中的固定数量的状态。作为另一示例,当连续状态偏离超过所选最大值的量级时,可以创建检查点。即,当状态相差很大程度时,创建检查点。检查点可用于标识计算数据内的前一帧的结束和下一连续帧的开始。在一些实施例中,可以进一步基于计算类型(例如,迭代、非迭代等)来压缩帧。

[0100] 在一些实施例中,可以基于对应于模拟的接收数据内的相应帧的检查点的频率来自适应地确定每个帧的宽度。在一些实施例中,模拟可包括被训练的模型的迭代模拟之一,其中每个迭代进一步基于一个或多个算法、神经网络等来细化模型的状态。作为另一示例,模拟可包括非迭代模拟,其包括正被处理的输入和输出对的集合。

[0101] 在514中,该方法可包括将所生成的连续帧传输至区块链网络的节点以用于包括在数据块的散列链接的链内的一个或多个数据块中。例如,每个帧可被传送给区块链网络内的节点的不同的认可对等组或子集。在一些实施例中,该方法可以进一步包括在来自连续生成的连续帧中的每个帧内存储迭代ID,其中,迭代ID标识与相应帧相关联的迭代模拟的相应迭代。在一些实施例中,该方法可进一步包括从区块链对等节点接收消息,该消息指示与所生成的连续帧中的帧相关联的状态已被无效化。作为响应,该方法可进一步包括重新计算和细化被无效的帧的状态以生成更新的帧以及将该更新的帧传送给区块链对等节点以供验证中的一者或多者。

[0102] 图5B示出根据示例性实施例的认可模拟数据帧的方法520。例如,方法520可由区块链网络的一个或多个认可节点来执行。节点可包括计算系统,诸如服务器、数据库、云平台等。参照图5B,在521中,方法可包括从迭代模拟的连续数据帧中接收数据帧。这里,数据帧可以包括迭代模拟的状态数据,其包括模型(算法等)的当前状态的标识、训练数据等。在一些实施例中,可使用压缩方案来压缩数据帧。在一些实施例中,可以对数据帧进行解压缩以揭示压缩之前的模拟的状态。

[0103] 在522中,该方法可以包括基于来自前一帧的之前验证的状态数据来生成该数据

帧的状态数据的本地计算。这里,先前验证的状态数据可被存储在区块链的块内的先前帧中。在523中,该方法可包括确定状态数据的本地计算与接收到的数据帧内的状态数据之间的相似性。响应于确定相似性在预定阈值内,在524中,该方法可包括向区块链网络发送数据帧的认可,该认可包括在数据块的散列链接的链中的数据块内。

[0104] 在一些实施例中,迭代模拟可以包括经由神经网络训练的模型的深度学习模拟等。此外,连续帧存储模型状态随迭代的变化。在一些实施例中,预定阈值可标识状态数据的本地计算与接收到的数据帧内的状态数据之间的可接受偏差水平。在一些实施例中,该方法还可包括响应于确定相似度在预定阈值之外,向提交数据帧的客户端节点传送指示状态数据无效的消息。在一些实施例中,该方法可以进一步包括:接收包括经细化的状态数据的经更新的数据帧;以及确定该状态数据的位置计算与该经更新的数据帧内的经细化的状态数据之间的相似性。

[0105] 图5C示出根据示例性实施例的指派用于并行认可的模拟数据的连续帧的方法530。例如,方法530可由区块链网络的客户端节点来执行。节点可包括计算系统,诸如服务器、数据库、云平台等。参照图5C,在531中,该方法可包括基于预定义的检查点生成迭代模拟的多个连续数据点。例如,每个数据点可以包括数据的帧或窗口,其存储关于连续数据点中的先前数据点的迭代模拟的演变状态的标识。这里,模拟可以是深度学习模拟,诸如经由神经网络等训练的机器学习算法。数据点可以被压缩以减少计算开销。

[0106] 在532中,该方法可包括:将用于验证多个连续数据点中的第一数据点内的状态数据的区块链请求传送到区块链网络的认可节点的第一子集,并且在532中,该方法还可包括将用于验证多个连续数据点中的第二数据点内的状态数据的区块链请求传送到与区块链网络的用于并行认可第一数据点和第二数据点的第一认可节点子集互斥的第二认可节点子集。客户节点可通过只使用认可对等节点的子集来认可帧,并使用不同的认可节点的子集来认可不同的帧,来减少认可对等节点的计算。

[0107] 在一些实施例中,基于迭代模拟的第一迭代生成第一数据点的状态数据,并且第二数据点的状态数据基于迭代模拟的后续迭代。在一些实施例中,第二数据点存储第一数据点的迭代模拟的状态与第二数据点处的迭代模拟的状态之间的差。在一些实施例中,该方法还包括接收来自第二认可节点子集的消息,该消息指示第二数据点的状态数据无效。

[0108] 在一些实施例中,该方法可以进一步包括细化被无效的第二数据点的状态数据以生成第二数据点的更新的状态数据,以及将更新的状态数据传送给认可节点的第二子集以供验证。在一些实施例中,该方法还可以包括在连续数据点中的每个数据点内存储迭代ID,其中迭代ID标识与相应数据点相关联的迭代模拟的相应迭代。在一些实施例中,该方法可以进一步包括执行迭代模拟以生成多个连续数据点。

[0109] 图5D示出根据示例性实施例的对作为并行认可处理结果的模拟数据排序的方法540。例如,方法540可以由区块链网络的排序节点来执行。节点可包括计算系统,诸如服务器、数据库、云平台等。参照图5D,在541中,方法可包括经由第一对等节点子集接收,通过迭代模拟产生的多个连续数据点中的第一数据点的状态数据的验证,以及在542,经由与第一对等节点子集互斥的第二对等节点子集来接收,所述迭代模拟的所述多个连续数据点中的第二数据点的状态数据的验证。这里,第一数据点和第二数据点可以由包括排序节点的相同区块链网络的认可节点的不同子集并行地验证或以其他方式认可。

[0110] 在532中,该方法可包括生成包括第一数据点和第二数据点的一个或多个数据块,该第一数据点和该第二数据点包括经验证的状态数据,并且在533中,该方法可包括将该一个或多个数据块传输至区块链网络内的对等节点以用于在数据块的散列链接的链之间进行存储。例如,排序节点可将数据块传输到区块链网络中的对等节点用于存储在由每个对等节点保持的区块链的本地副本上。

[0111] 在一些实施例中,该方法可以进一步包括基于包括在第一数据点和第二数据点中的时间戳将第一数据点和第二数据点安排在队列内。在此实例中,产生所述一个或多个数据块可包含基于所述队列内的所述第一数据点和所述第二数据点的位置对所述一个或一个以上数据块内的所述第一数据点和所述第二数据点进行排序。在一些实施例中,每次验证可以指示相应数据点的状态数据在与预定义阈值的可接受偏差范围内。在一些实施例中,第一和第二数据点可以各自包括迭代ID,其标记与相应数据点相关联的迭代模拟的相应迭代。

[0112] 图5E展示了根据示例性实施例的压缩模拟内容的方法550。例如,方法550可由生成和/或接收计算数据的区块链网络的客户端节点来执行。节点可包括计算系统,诸如服务器、数据库、云平台、多个系统等。参照图5E,在551中,方法可包括生成存储模拟内容的数据帧。这里,数据帧可以是包括连续生成的计算模型的状态数据和诸如深度学习模型、大规模模拟等的数据的多个数据帧中的一个。

[0113] 在552中,该方法可以包括基于存储在另一个数据帧中的先前模拟内容来压缩该数据帧内的模拟内容以生成压缩的数据帧。例如,模拟内容可以是来自迭代模拟的状态数据或来自非迭代模拟的状态数据。如果模拟是具有多个输入/输出对的非迭代模拟,则压缩可以包括基于基于输入/输出对的接近度的最小生成树(MST)过程来压缩非迭代模拟内容。作为另一示例,如果模拟是具有迭代演进的状态的迭代模拟,则压缩可以包括基于与迭代模拟的先前迭代的状态的差来压缩迭代模拟内容。先前状态可由节点识别且用于在两个帧之间插入状态数据的增量,而非包含整个状态数据。

[0114] 在553中,该方法可包括经由区块链请求将经压缩的数据帧传输至区块链网络的一个或多个认可对等节点以将经压缩的数据帧包括在区块链网络的散列链接的区块链内。例如,区块链请求可包括认可事务以存储在区块链上的请求。在一些实施例中,数据帧可包括基于模拟内容内的一个或多个预定义检查点的自适应大小。在一些实施例中,压缩数据帧可以包括关于模拟内容的先前压缩数据帧的状态更新。

[0115] 图5F展示了根据示例性实施例的认可经压缩的模拟内容的方法560。例如,方法560可由区块链网络内的认可对等节点或认可可对等节点的子集来执行。参照图5F,在561中,所述方法可以包括接收包括模拟内容的数据帧,所述模拟内容已经基于存储在先前数据帧中的先前模拟内容被压缩。在一些实施例中,压缩内容可被解压缩。在562中,该方法可以包括从数据块的哈希链接的链中提取存储在先前数据帧中的先前模拟内容。可以从区块链中提取先前的模拟内容,该区块链在块中存储模拟的先前验证的状态数据。

[0116] 在563,该方法可以包括基于所提取的先前模拟内容生成本地模拟内容。这里,可以通过使用由节点存储的模拟的先前验证的状态重新计算来生成本地模拟内容,以确定状态的估计。在564中,该方法可以包括基于接收的压缩的模拟内容和本地模拟内容之间的差别来确定是否认可接收的数据帧以包括在块的散列链接的链中。该确定可以基于接收的模

拟内容的状态数据和本地生成的模拟内容的状态数据之间的相似性。如果状态数据在预定义的阈值偏差内,则可以授权认可。在565中,响应于确定认可接收到的数据帧,该方法可包括向区块链网络中的对等节点发送认可。

[0117] 在某些实施例中,所接收的压缩的模拟内容可以存储在数据帧中并且可以包括来自具有多个输入/输出对的非迭代模拟的内容。在该示例中,可以基于MST过程来压缩非迭代模拟内容,该MST过程基于输入/输出对的紧密度。在一些实施例中,所接收的压缩模拟内容可以存储在数据帧中,并且可以包括来自迭代模拟的内容,所述迭代模拟具有迭代演进的状态。在一些实施例中,可以基于与迭代模拟的先前迭代的状态的差来压缩迭代模拟内容。

[0118] 图6A示出了示例物理基础设施,该物理基础设施被配置成用于根据示例性实施例的这些示例操作中的一个或多个在区块链上执行不同操作。参见图6A,示例配置600A包括具有区块链620和智能合约640的物理基础设施610,其可以执行任何示例性实施例中包括的任何操作步骤612。步骤/操作612可包含一个或多个流程图和/或逻辑图中所描述或描绘的步骤中的一或多者。步骤可以表示从驻留在计算机系统配置的物理基础设施610上的一个或多个智能合约640和/或区块链620写入或读取的输出或写入信息。数据可以从执行的智能合约640和/或区块链620输出。物理基础设施610可包括一个或多个计算机、服务器、处理器、存储器和/或无线通信设备。

[0119] 图6B展示了根据示例性实施例的在订约方和被配置成用于在区块链上实施智能合约条款的中介服务器之间的示例智能定约配置。参见图6B,配置650B可以表示由显式地标识一个或多个用户设备652和/或656的智能合约630驱动的通信会话、资产传输会话或流程或过程。智能合约执行的执行、操作和结果可以由服务器654管理。智能合约640的内容可以要求作为智能合约交易各方的实体652和656中的一个或多个进行数字签名。

[0120] 图6C示出了根据示例性实施例的在订约方和被配置成用于在区块链上实施智能定约条款的中介服务器之间的示例智能定约配置。参见图6C,配置650可以表示通信会话、资产传输会话或由显式地标识一个或多个用户设备652和/或656的智能合约630驱动的过程或过程。智能合约执行的执行、操作和结果可以由服务器654管理。智能合约630的内容可能需要作为智能合约交易各方的实体652和656中的一个或多个实体的数字签名。智能合约执行的结果可以作为区块链事务写入区块链620。在该示例中,智能合约630驻留在可以驻留在一个或多个计算机、服务器、处理器、存储器和/或无线通信设备上的区块链620上。

[0121] 图6D示出了根据示例性实施例的用于访问区块链的逻辑和数据的通用接口。参考图6D的示例,应用编程接口(API)网关662提供用于访问区块链逻辑(例如,智能合约630或其他链码)和数据(例如,分布式分类账等)的通用接口。在该示例中,API网关662是用于通过将一个或多个实体652和656连接到区块链对等体(即,服务器654)来对区块链执行事务(调用、查询等)的通用接口。服务器654是保持世界状态的副本和允许客户端652和656查询关于世界状态的数据以及将事务提交到区块链网络中的分布式分类账的区块链网络对等体组件,其中,取决于智能合约630和认可策略,认可对等体将运行智能合约630。

[0122] 图7A示出根据示例性实施例的被添加到分布式分类账720的新块730的过程700,并且图7B示出根据示例性实施例的用于区块链的块结构730的内容。参考图7A,客户端(未示出)可以向区块链节点711、712和/或713提交事务。客户端可以是任何源接收的指令以

在区块链上执行活动。作为示例,客户端可以是代表请求者(诸如设备、人或实体)行动以提议针对区块链的事务的(基于SDK的)应用程序。多个区块链对等体(例如,区块链节点711、712和713)可维护区块链网络的状态和分布式分类账720的副本。

[0123] 在区块链网络中可存在不同类型的区块链节点/对等体,包括模拟和认可客户端所提议的事务的认可对等体和向分布式分类账720验证认可、验证事务和提交事务的提交对等体。在该示例中,区块链节点711、712和713可以执行认可方节点、提交者节点或两者的角色。

[0124] 分布式分类账720包括区块链722和状态数据库724(当前世界状态),该区块链以块存储不可变的经排序的记录,该状态数据库维持区块链722的当前状态(键值)。一个分布式分类账720可以存在于每个通道,并且每个对等体为每个通道维护分布式分类账720的其自己的副本,每个对等体是每个通道的成员。区块链722是事务日志,被结构化为散列链接的块,其中每个块包含N个事务的序列。块(例如,块730)可以包括如图7B中所示的各种部件。可以通过在当前块的块报头内添加先前块的报头的散列来生成块的链接(由图7A中的箭头示出)。以此方式,区块链722上的所有事务被排序并且加密地链接在一起,从而防止篡改区块链数据而不破坏散列链接。此外,由于链接,区块链722中的最新块表示在其之前已经到来的每个事务。区块链722可被存储在对等文件系统(本地或附连存储)上,该对等文件系统支持仅附加的区块链工作负荷。

[0125] 区块链722和分布式分类账720的当前状态可以存储在状态数据库724中。这里,当前状态数据表示曾经包括在区块链722的链事务日志中的所有键的最新值。链码调用针对状态数据库724中的当前状态执行事务。为了使这些链码交互有效,可以将所有键的最新值存储在状态数据库724中。状态数据库724可包括区块链722的事务日志中的索引视图,并且因此可在任何时间从该链重新生成。在接受事务之前,状态数据库724可在对等体启动时自动恢复(或在需要时生成)。

[0126] 认可节点接收来自客户端的事务并且基于模拟结果来认可该事务(例如,模拟状态的变化等)。认可节点可以持有模拟事务提议的智能合约。认可事务所需的节点取决于可在链码内指定的认可策略。认可策略的示例为“大多数认可对等体必须认可事务。”不同的信道可具有不同的认可策略。客户端应用程序将认可的事务转发到排序服务710。

[0127] 排序服务710接受认可的事务,将它们排序成块,并且将块递送给提交对等体。例如,当达到事务的阈值、定时器超时或另一条件时,排序服务710可以发起新的块。排序服务710可基于本文描述的时间戳协定过程,诸如基于来自区块链节点711-713的时间戳的加权平均值来计算每个事务的最终时间戳等,来操作。在图7A的示例中,区块链节点712是已经接收到新数据块730以存储在区块链722上的提交对等体。

[0128] 排序服务710可以由订购者集群组成。排序服务710不处理事务、智能合约或维护共享分类账。相反,排序服务710可接受认可的事务,确定事务的最终时间戳,并基于最终时间戳指定那些事务被提交给分布式分类账720的顺序。区块链式网络的架构可以被设计成使得‘排序’(例如,Solo、Kafka、BFT等)的具体实现变成可插拔组件。

[0129] 事务以一致的顺序被写入到分布式分类账720。事务的顺序被建立以确保对状态数据库724的更新在它们被提交给网络时是有效的。与其中通过解决密码拼图或挖掘来进行排序的密码区块链系统(例如,Bitcoin等)不同,在该示例中,分布式分类账720的各方可

选择最适合该网络的排序机制,诸如按时间顺序的排序。

[0130] 当排序服务710初始化新块730时,可以向提交对等体(例如,区块链节点711、712和713)广播新块730。作为响应,每个提交对等体通过检查以确保读取集合和写入集合仍然匹配状态数据库724中的当前世界状态来验证新块730内的事务。具体地,提交对等体可以确定在认可者模拟事务时存在的读取数据是否与状态数据库724中的当前世界状态相同。当提交对等体验证该事务时,该事务被写到分布式分类账720上的区块链722,并且状态数据库724用来自读-写集合的写数据来更新。如果事务失败,即,如果提交对等体发现读写集合与状态数据库724中的当前世界状态不匹配,则排序到块中的事务将仍然被包括在该块中,但是它将被标记为无效,并且状态数据库724将不被更新。

[0131] 参照图7B,存储在分布式分类账720的区块链722上的块730(也被称为数据块)可以包括多个数据段,如块头部732、块数据734和块元数据736。应当理解,各种所描绘的块及其内容(诸如图7B中所示的块730及其内容)仅用于示例的目的,并且不旨在限制示例性实施例的范围。在一些情况下,块头部732和块元数据736两者可以小于存储事务数据的块数据734,然而这不是必须的。块730可将N个事务(例如,100、500、1000、2000、3000等)的事务信息存储在块数据734内。根据各实施例,每一事务可包括帧数据735,该帧数据735包括已由认可对等节点验证的模拟(例如,迭代、非迭代等)的状态的改变。

[0132] 元数据736可以包括生成数据帧的客户端节点/工作者的标识,认可该帧的认可节点、从其中取得状态数据的模拟的迭代(例如,迭代索引等),指向被运行以生成该帧的模拟实例的环境ID,等等。除了用于每个事务的有效/无效指示符之外,元数据可以被存储在块元数据区段736中。所存储的元数据对于重新验证存储在数据块中的信息的指示符是有用的。

[0133] 传统区块链存储信息,如事务(传递属性)和智能定约(有待执行的指令集)。在本实施例中,所存储的数据还可以包括计算过程的经验证的状态。即,每个块可以包括帧,该帧包括从帧中的连续迭代的增量编码获得的压缩检查点和量化状态更新。要存储的附加元数据可包括执行模拟的客户端的ID、验证状态的认可者、迭代索引和用于执行模拟的其他无关信息。

[0134] 块730还可以包括块头部732内的到前一个块(例如,在图7A中的区块链722上)的链接。具体而言,块头部732可包括先前块的头部的散列。块头部732还可包括唯一块编号、当前块730的块数据734的散列等等。块730的块编号可以是唯一的,并且以从零开始的递增/连续的顺序分配。区块链中的第一个块可被称为起源块,其包括关于区块链、其成员、存储在其中的数据等的信息。

[0135] 块数据734可以存储在块730内记录的每个事务的事务信息。例如,存储在块数据734内的事务数据可包括事务的类型中的一个或多个,版本、时间戳(例如,最终计算的时间戳等)、分布式分类账720的通道ID,事务ID、时期、有效载荷可见性、链码路径(部署tx)、链码名称,链码版本、输入(链码和功能)、客户端(创建者)标识(诸如公钥和证书),客户端的签名、认可者的身份、认可者签名、提议散列、链码事件,响应状态、命名空间、读取集(事务所读取的密钥和版本的列表等),写入集合(键和值的列表等)、开始键、结束键,键列表、Merkel树查询摘要等。可为N个交易中的每个交易存储交易数据。

[0136] 块元数据736可以存储元数据的多个字段(例如,作为字节阵列等)。元数据字段可

以包括关于块创建的签名、对最后配置块的引用、标识块内的有效和无效事务的事务过滤器、对块进行排序的排序服务的最后保留偏移等等。签名、最后一个配置块和排序器元数据可以由排序服务710添加。同时,块的提交节点(诸如区块链节点712)可基于认可策略、读/写集合的验证等添加有效性/无效性信息。事务过滤器可包括大小等于块数据734中的事务的数量的字节阵列和标识事务是否有效/无效的验证码。

[0137] 以上实施例可以用硬件、由处理器执行的计算机程序、固件、或以上的组合来实现。计算机程序可以体现在诸如存储介质的计算机可读介质上。例如,计算机程序可以驻留在随机存取存储器(“RAM”)、闪存、只读存储器(“ROM”)、可擦除可编程只读存储器(“EPROM”)、电可擦除可编程只读存储器(“EEPROM”)、寄存器、硬盘、可移动盘、致密盘只读存储器(“CD-ROM”)、或本领域已知的任何其他形式的存储介质中。

[0138] 示例性存储介质可以耦合至处理器,从而使得处理器可以从存储介质读取信息和向存储介质写入信息。在替代方案中,存储介质可集成到处理器。处理器和存储介质可以驻留在专用集成电路(“ASIC”)中。在替代方案中,处理器和存储介质可作为离散组件驻留。例如,图8示出示例计算机系统架构800,其可表示或集成在任何上述组件等中。

[0139] 图8不旨在对在此描述的本申请的实施例的用途或功能的范围提出任何限制。无论如何,计算节点800能够被实现和/或执行上文阐述的功能中的任何功能。例如,计算节点800可以执行关于图5A-5F所示出和描述的方法510-560中的任何方法。

[0140] 在计算节点800中,存在计算机系统/服务器802,该计算机系统/服务器可与众多其他通用或专用计算系统环境或配置一起操作。可以适合于与计算机系统/服务器802一起使用的众所周知的计算系统、环境和/或配置的示例包括,但不限于个人计算机系统、服务器计算机系统、瘦客户机、厚客户机,手持式或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品,网络PC、小型计算机系统、大型计算机系统和包括任何上述系统或设备的分布式云计算环境,等等。

[0141] 可以在由计算机系统执行的计算机系统可执行指令(如程序模块)的一般上下文中描述计算机系统/服务器802。一般而言,程序模块可包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、逻辑、数据结构等。计算机系统/服务器802可以在分布式云计算环境中实现,其中任务由通过通信网络链接的远程处理设备来执行。在分布式云计算环境中,程序模块可位于包括存储器存储设备的本地和远程计算机系统存储介质两者中。

[0142] 如图8所示,云计算节点800中的计算机系统/服务器802以通用计算设备的形式示出。计算机系统/服务器802的组件可以包括但不限于一个或多个处理器或处理单元804、系统存储器806和将包括系统存储器806的不同系统组件耦合到处理器804的总线。

[0143] 总线表示若干类型的总线结构中的任一种总线结构中的一种或多种,包括存储器总线或存储器控制器、外围总线、加速图形端口、以及使用各种总线架构中的任一种的处理器或局部总线。作为示例而非限制,此类架构包括工业标准架构(ISA)总线、微通道架构(MCA)总线、增强型ISA(EISA)总线、视频电子标准协会(VESA)局部总线和外围组件互连(PCI)总线。

[0144] 计算机系统/服务器802通常包括各种计算机系统可读介质。这样的介质可以是可由计算机系统/服务器802访问的任何可用介质,并且它包括易失性和非易失性介质、可移

动和不可移动介质两者。在一个实施例中,系统存储器806实现其他附图的流程图。系统存储器806可包括易失性存储器形式的计算机系统可读介质,诸如随机存取存储器(RAM) 810和/或高速缓存存储器812。计算机系统/服务器802还可以包括其他可移动/不可移动、易失性/非易失性计算机系统存储介质。仅作为示例,存储系统814可以被提供用于从不可移动、非易失性磁介质(未示出,并且通常被称为“硬盘驱动器”)读取和向其写入。尽管未示出,可以提供用于从可移除的非易失性磁盘(例如,“软盘”)读取和向其写入的磁盘驱动器以及用于从可移除的非易失性光盘(如CD-ROM、DVD-ROM或其他光学介质)读取或向其写入的光盘驱动器。在这样的实例中,每一个都可以通过一个或多个数据介质接口连接到总线。如下面将进一步描绘和描述的,存储器806可以包括至少一个程序产品,该程序产品具有被配置为执行应用的不同实施例的功能的一组(例如,至少一个)程序模块。

[0145] 具有一组(至少一个)程序模块818的程序/实用程序816以及操作系统、一个或多个应用程序、其他程序模块和程序数据可以通过示例而非限制的方式存储在存储器806中。操作系统、一个或多个应用程序、其他程序模块和程序数据中的每一者或其某一组合可包含联网环境的实施例。程序模块818通常执行本文所述的应用程序的不同实施例的功能和/或方法。

[0146] 如本领域技术人员将认识到的,本申请的各方面可以具体化为系统、方法或计算机程序产品。相应地,本申请的多个方面可以采取完全硬件实施例、完全软件实施例(包括固件、驻留软件、微代码等)或组合软件和硬件方面的实施例的形式,这些实施例在此可以全部统称为“电路”、“模块”或“系统。”此外,本申请的多个方面可以采取体现在一个或多个计算机可读介质中的计算机程序产品的形式,该计算机可读介质具有体现在其上的计算机可读程序代码。

[0147] 计算机系统/服务器802还可以与一个或多个外部设备820,如键盘、定点设备、显示器822等;使得用户能够与计算机系统/服务器802交互的一个或多个设备;和/或使计算机系统/服务器802能够与一个或多个其他计算设备通信的任何设备(例如,网卡、调制解调器等)通信。这样的通信可以经由I/O接口824发生。此外,计算机系统/服务器802可以经由网络适配器826与诸如局域网(LAN)、通用广域网(WAN)和/或公共网络(例如,互联网)之类的一个或多个网络通信。如所描绘的,网络适配器826经由总线与计算机系统/服务器802的其他组件通信。应当理解,虽然未示出,但是其他硬件和/或软件组件可以与计算机系统/服务器802结合使用。示例包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动器阵列、RAID系统、磁带驱动器和数据归档存储系统等。

[0148] 尽管系统、方法和非瞬态计算机可读介质中的至少一者的示例性实施例已经在附图中展示并且在以上详细描述中进行了描述,应当理解,本申请不限于所公开的实施例,而是能够进行如所附权利要求所阐述和定义的许多重新布置,修改和替换。例如,各个附图的系统的能力可以由本文描述的模块或组件中的一个或多个或以分布式架构来执行,并且可以包括发射机、接收机或两者的对。例如,由各个模块执行的全部或部分功能可以由这些模块中的一个或多个来执行。此外,本文描述的功能可以在模块或组件内部或外部的各种时间并且与各种事件相关地执行。而且,可以经由以下各项中的至少一项在各个模块之间发送各种模块之间发送的信息:数据网络、互联网、语音网络、互联网协议网络、无线设备、有线设备和/或通过多种协议。而且,由任何模块发送或接收的消息可以直接和/或经由一个

或多个其他模块发送或接收。

[0149] 本领域技术人员将认识到,“系统”可以具体化为个人计算机、服务器、控制台、个人数字助理(PDA)、蜂窝电话、平板计算设备、智能电话或任何其他合适的计算设备、或设备的组合。呈现如由“系统”执行的上述功能不旨在以任何方式限制本申请的范围,而是旨在提供许多实施例的一个实例。实际上,本文公开的方法、系统和装置可以按照与计算技术一致的本地化和分布式形式来实现。

[0150] 应注意的是,本说明书中所描述的一些系统特征已经被呈现为模块,以便更具体地强调其实现方式独立性。例如,模块可以被实现为包括定制的超大规模集成(VLSI)电路或门阵列、现成的半导体(诸如逻辑芯片、晶体管或其他分立组件)的硬件电路。模块还可以在诸如现场可编程门阵列、可编程阵列逻辑、可编程逻辑设备、图形处理单元等的可编程硬件设备中实现。

[0151] 模块还可以至少部分地在软件中实现以便由不同类型的处理器执行。所标识的可执行代码单元可以例如包括计算机指令的一个或多个物理或逻辑块,其可以例如被组织为对象、过程或函数。然而,所识别模块的可执行文件不需要在物理上位于一起,而是可以包括存储在不同位置中的不同指令,当逻辑上结合在一起时,所述不同指令包括模块并且实现模块的所述目的。进一步,模块可存储在计算机可读介质上,所述计算机可读介质可为(例如)硬盘驱动器、快闪存储器装置、随机存取存储器(RAM)、磁带或用于存储数据的任何其他此类介质。

[0152] 实际上,可执行代码的模块可以是单个指令或许多指令,并且甚至可以分布在若干不同代码段上、不同程序之间、和跨若干存储器设备。类似地,操作数据可以在本文中在模块内被识别和示出,并且可以以任何合适的形式体现和被组织在任何合适类型的数据结构内。操作数据可以作为单个数据集收集或者可以分布在不同位置(包括不同的存储设备)上,并且可以至少部分地仅作为系统或网络上的电子信号存在。

[0153] 将容易理解的是,本申请的部件(如在此的附图中总体上描述和展示的)可以被安排和设计成各种各样不同的配置。由此,实施例的详细描述不旨在限制所要求保护的申请的范围,而仅代表本申请的所选实施例。

[0154] 本领域的普通技术人员将容易理解,可以用不同顺序的步骤和/或用与所公开的那些不同的配置中的硬件元件来实践以上内容。因此,尽管已基于这些优选实施例描述了本申请,但对于本领域的技术人员显而易见的是,某些修改、变化和替代构造将是显而易见的。

[0155] 虽然已经描述了本申请的优选实施例,但应理解的是,所描述的实施例仅是说明性的,并且当考虑对其的全范围的等效物和修改(例如,协议、硬件设备、软件平台等)时,本申请的范围仅由所附权利要求书限定。

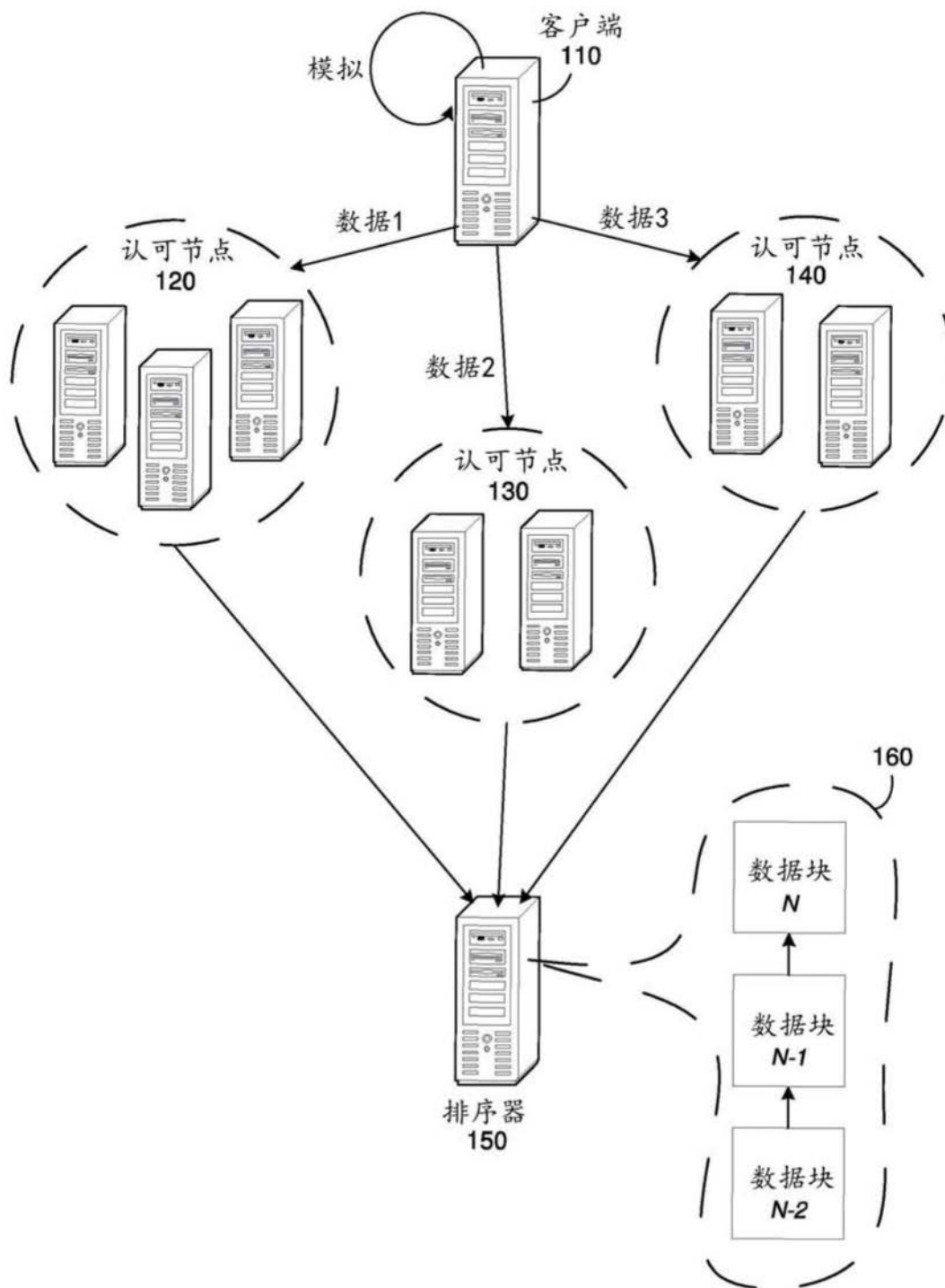
100

图1

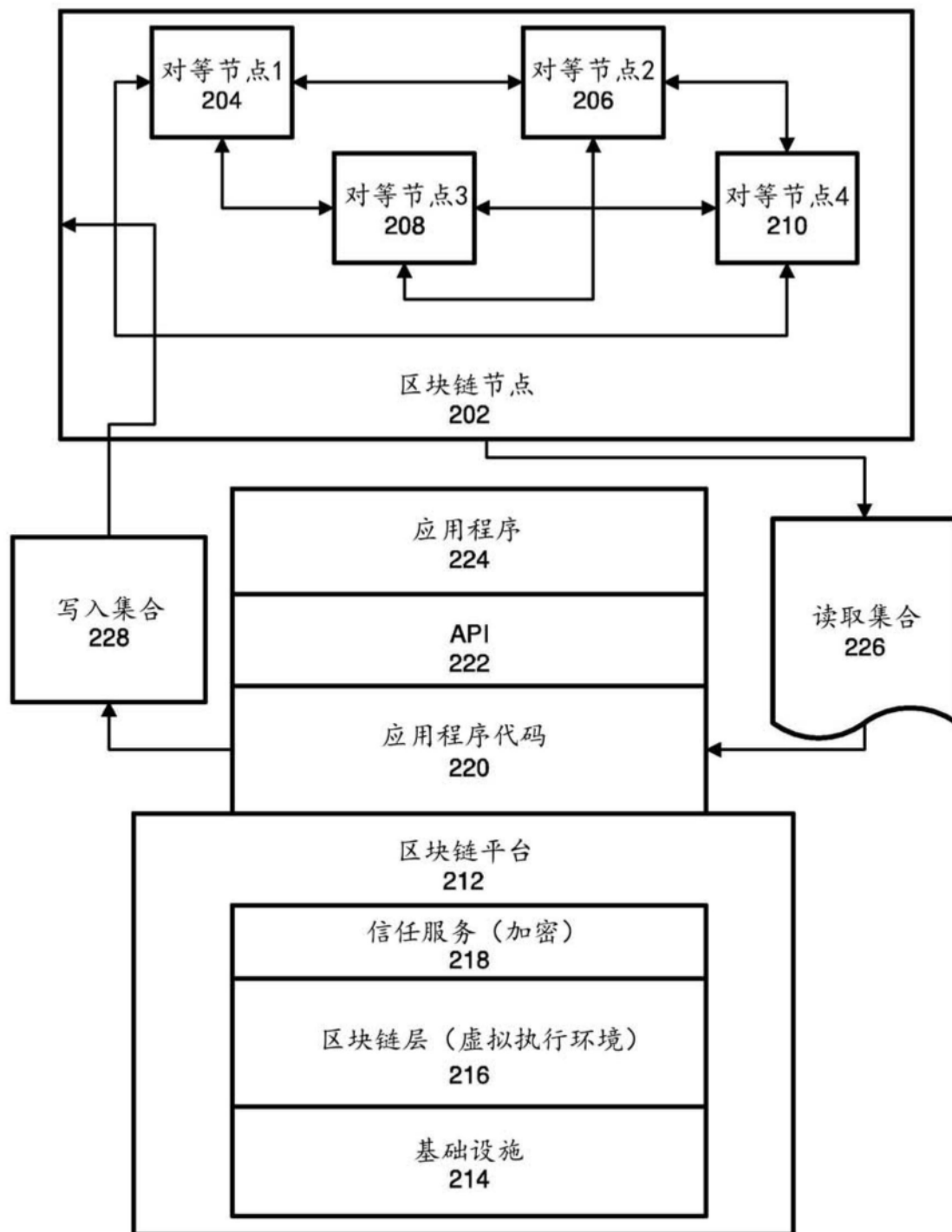
200

图2A

250

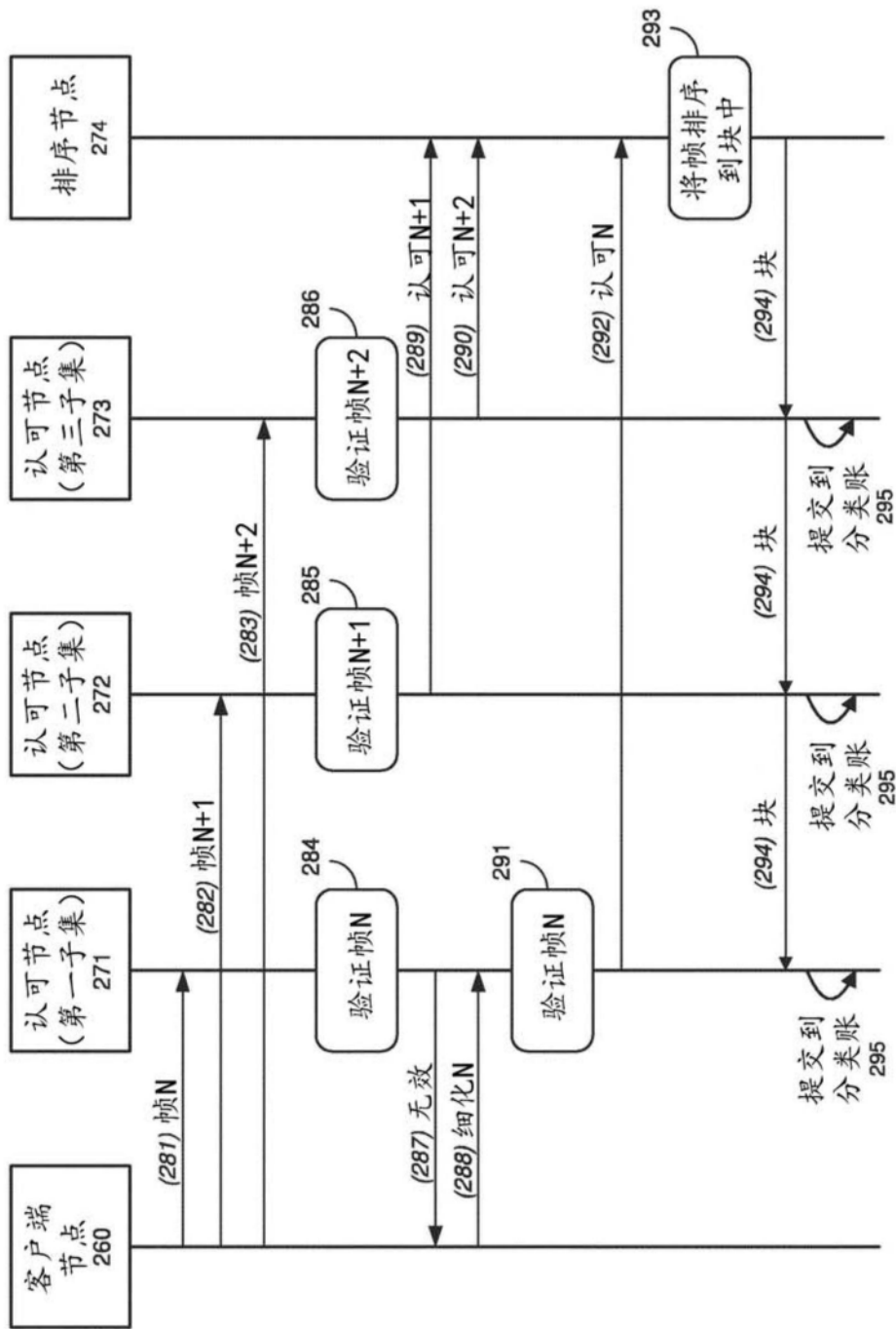


图2B

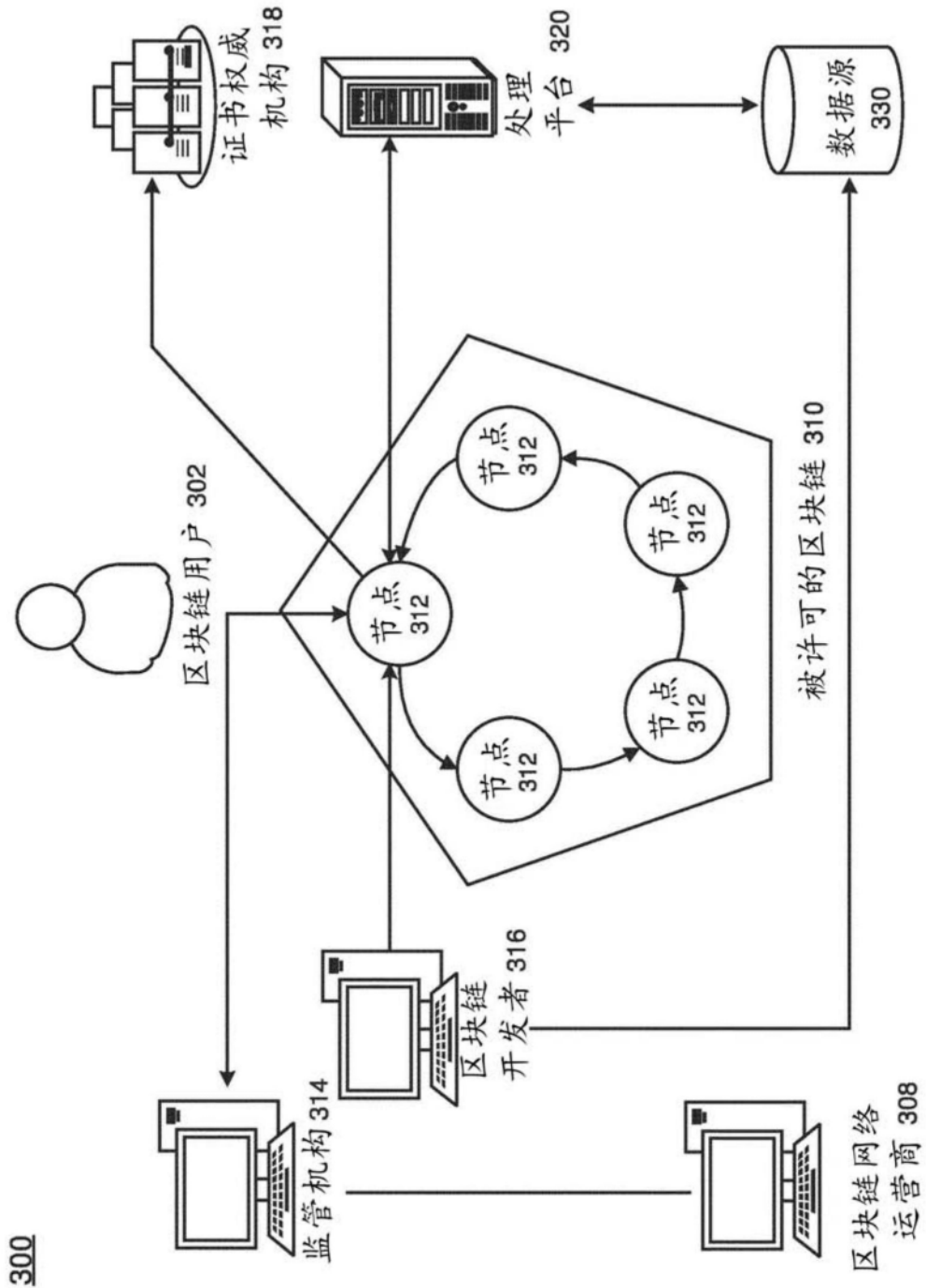


图3

400A

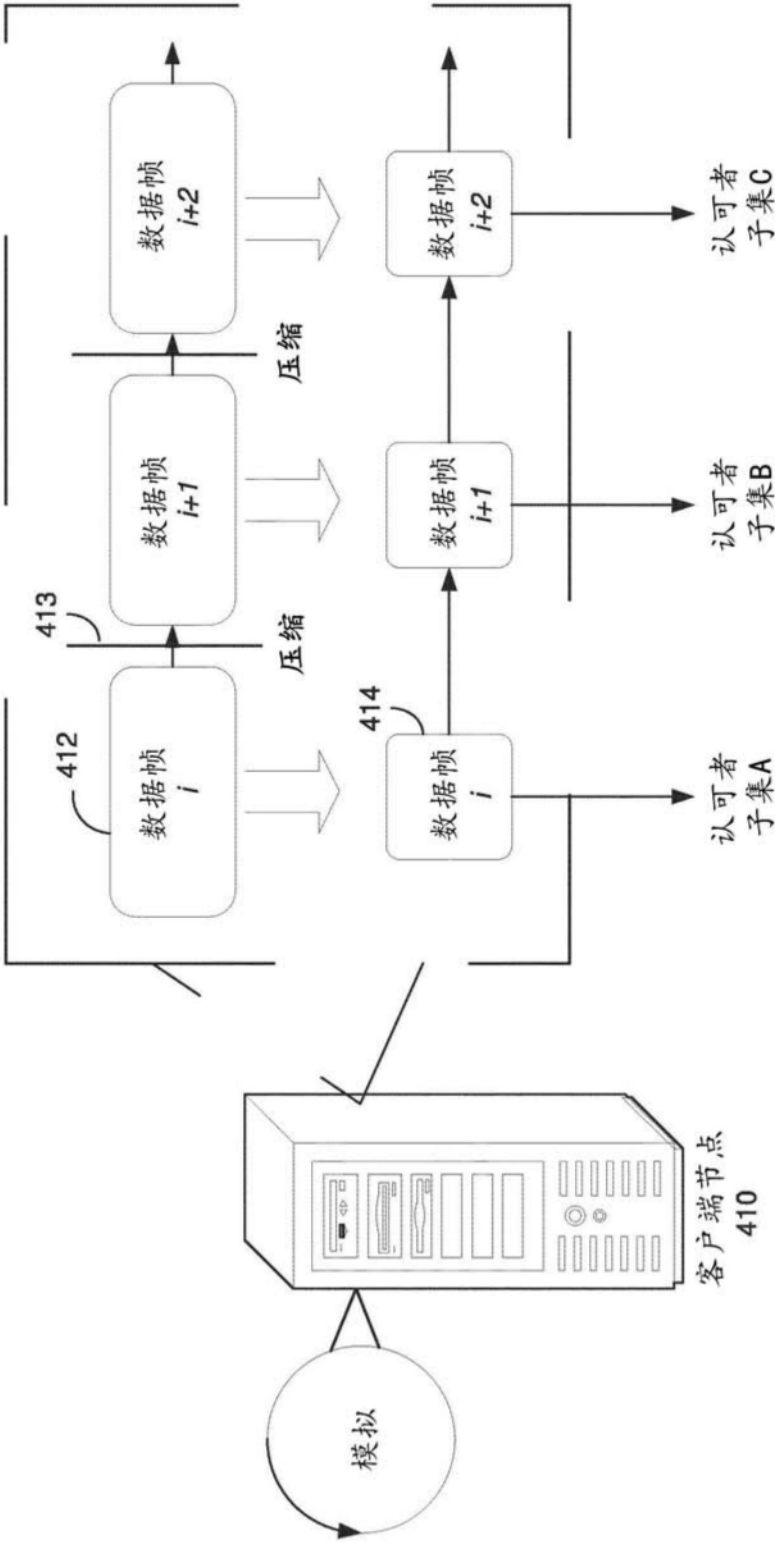


图4A

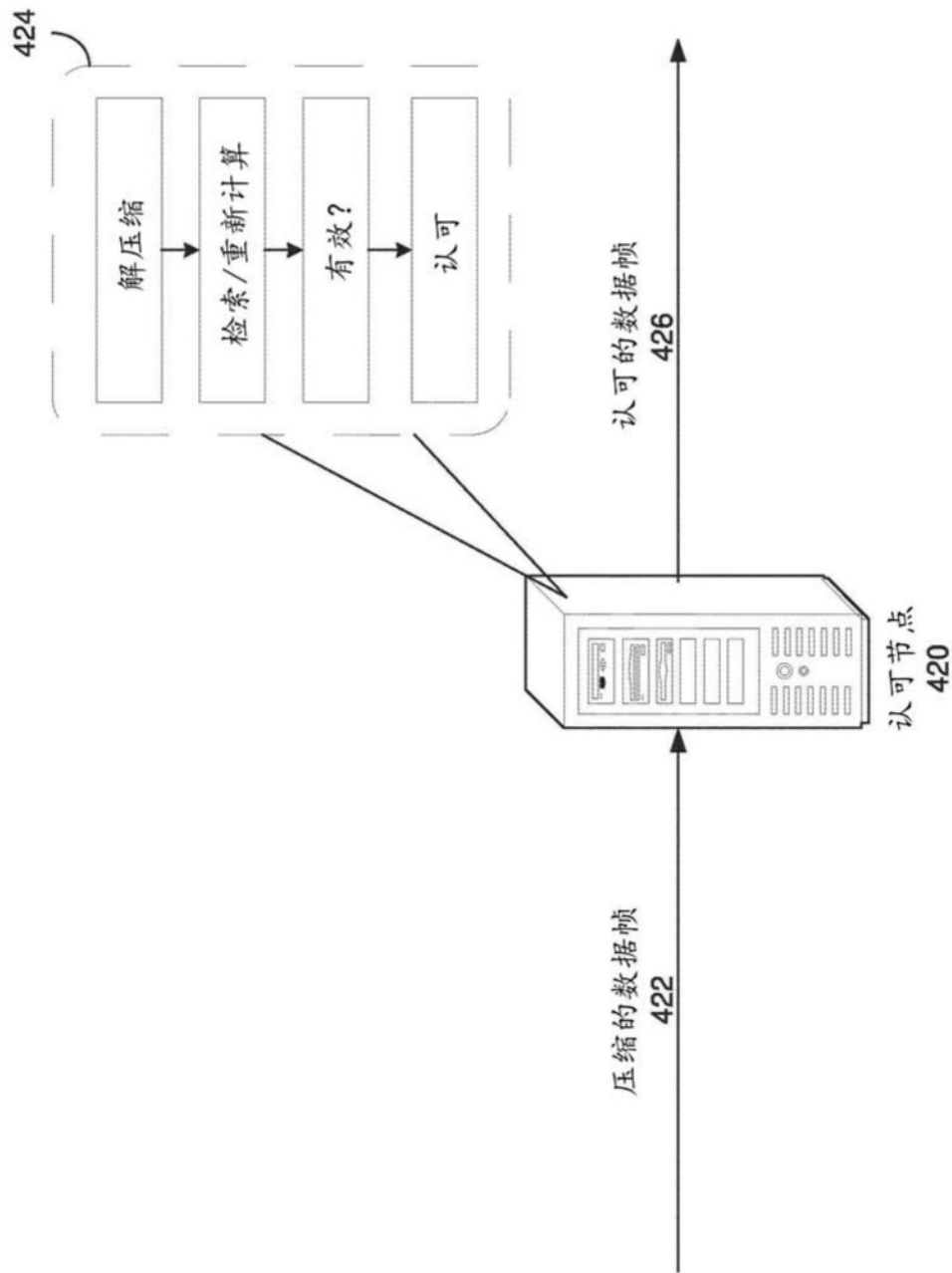
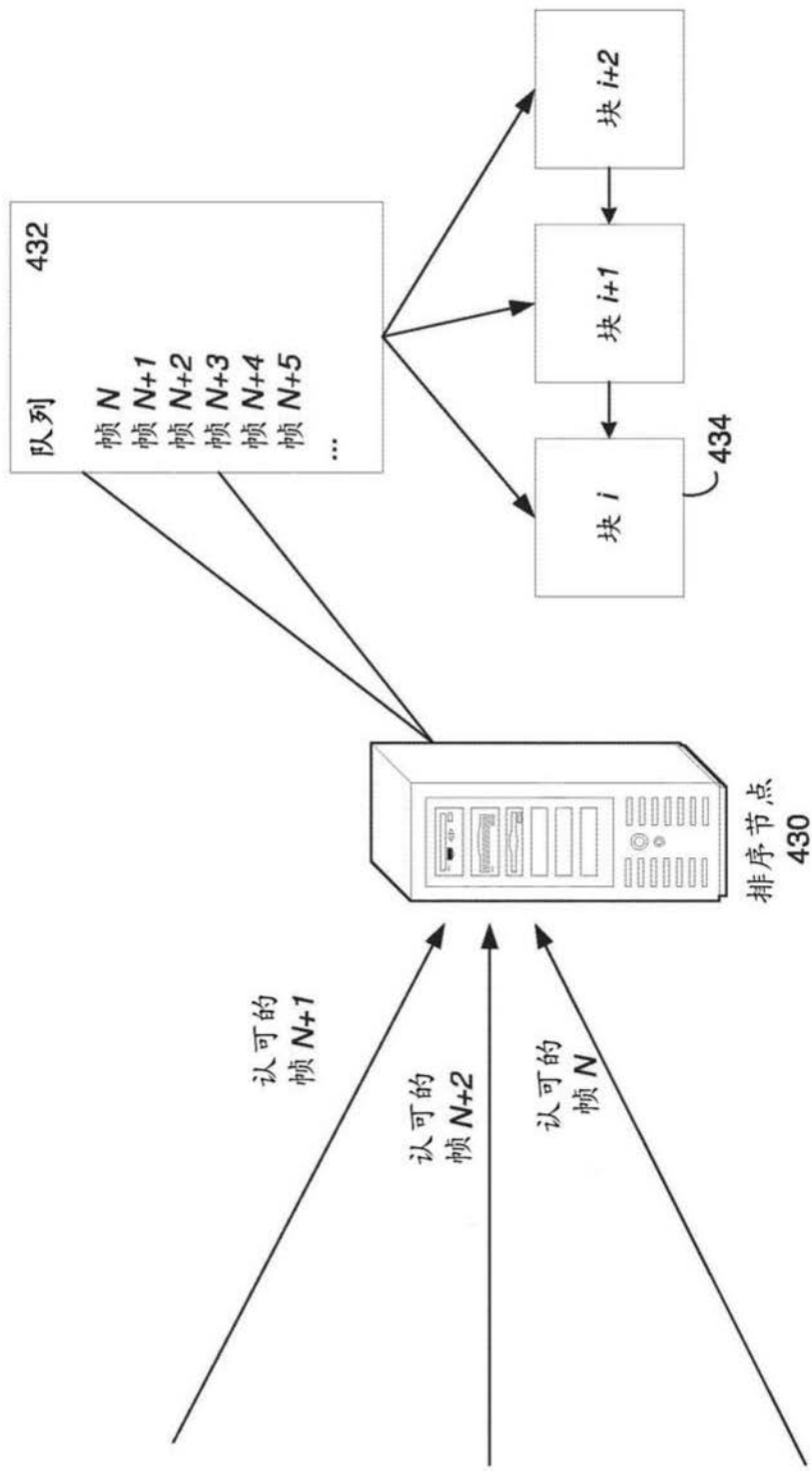


图4B



400C

图4C

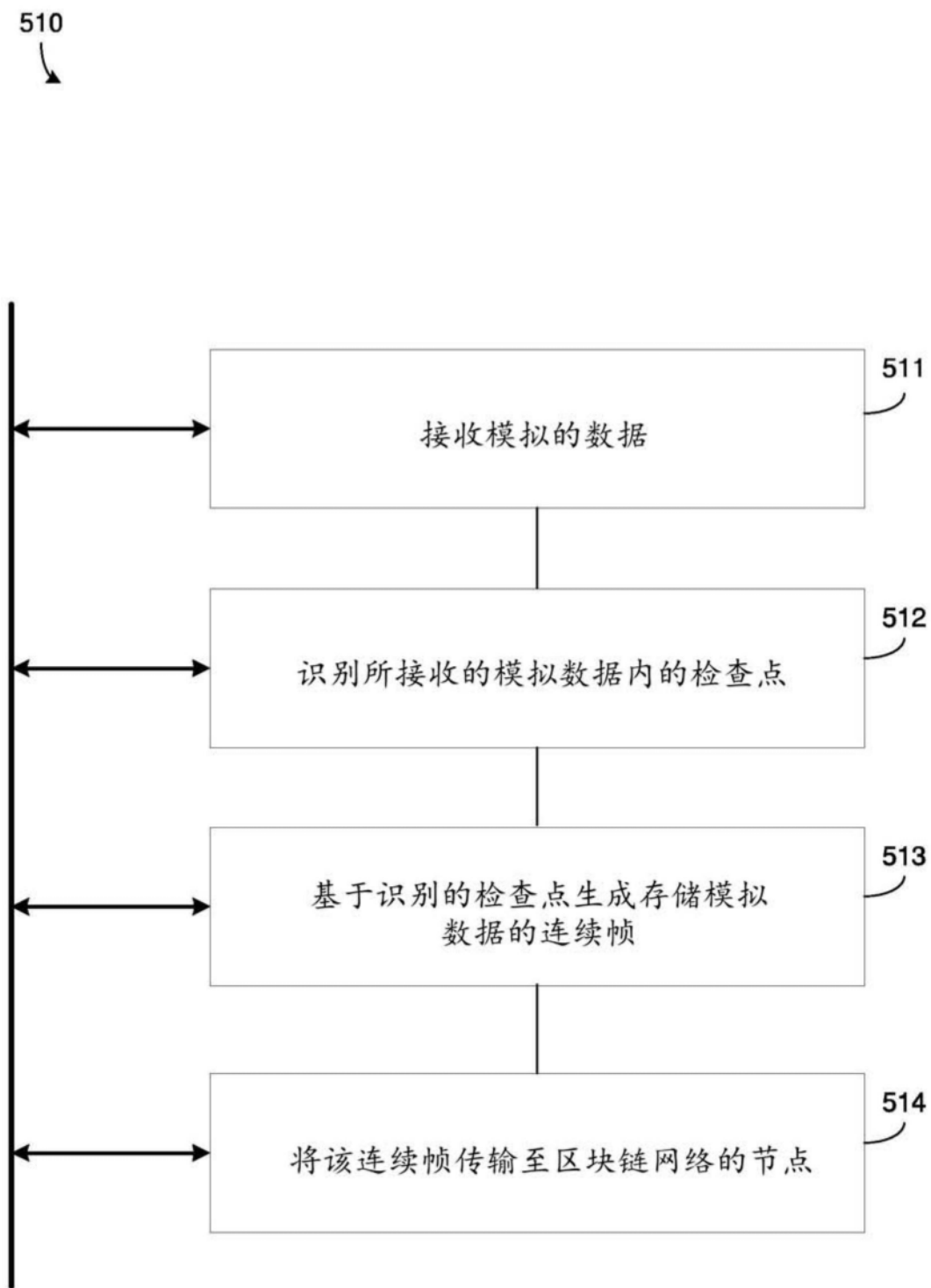


图5A

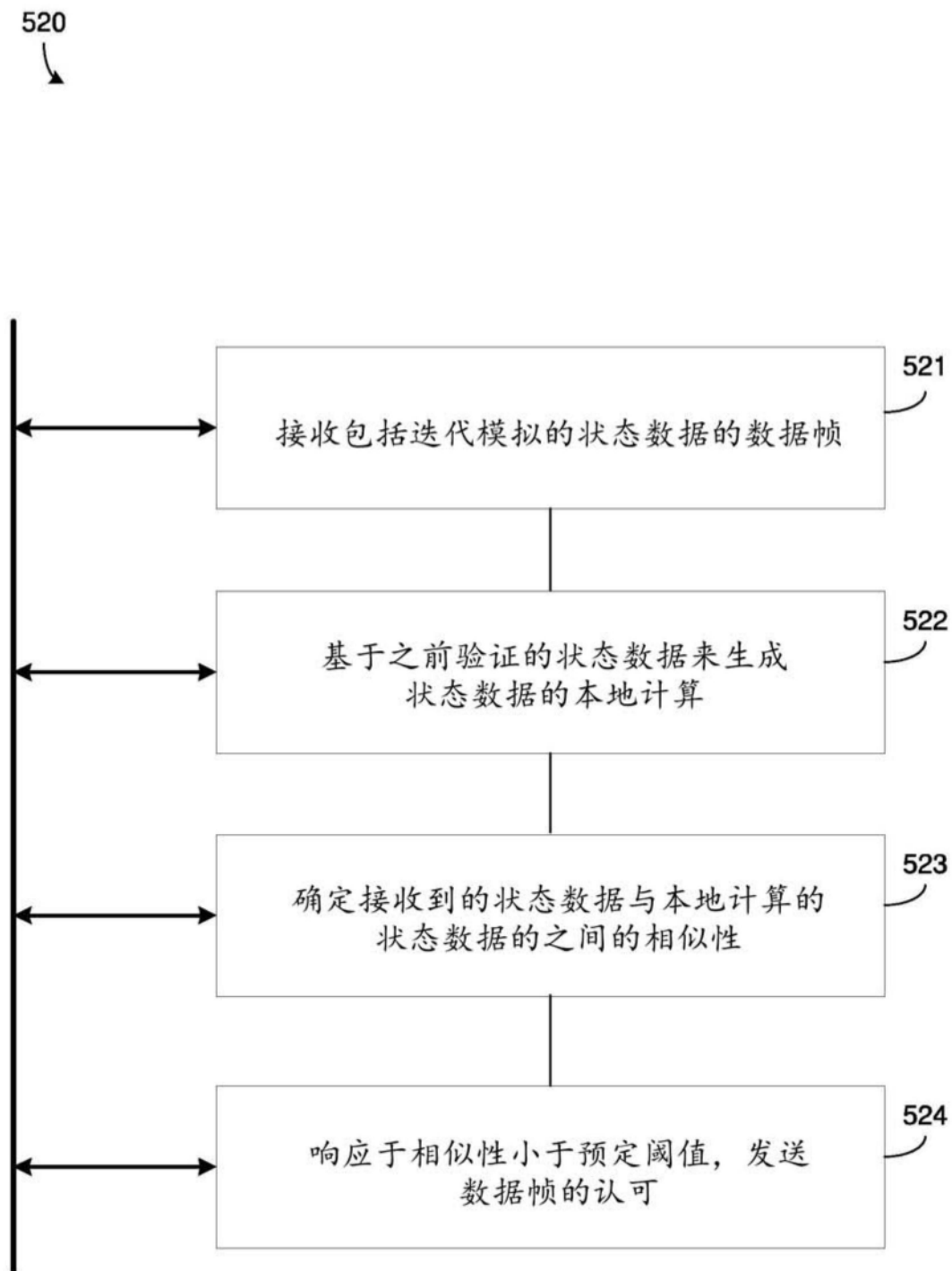


图5B

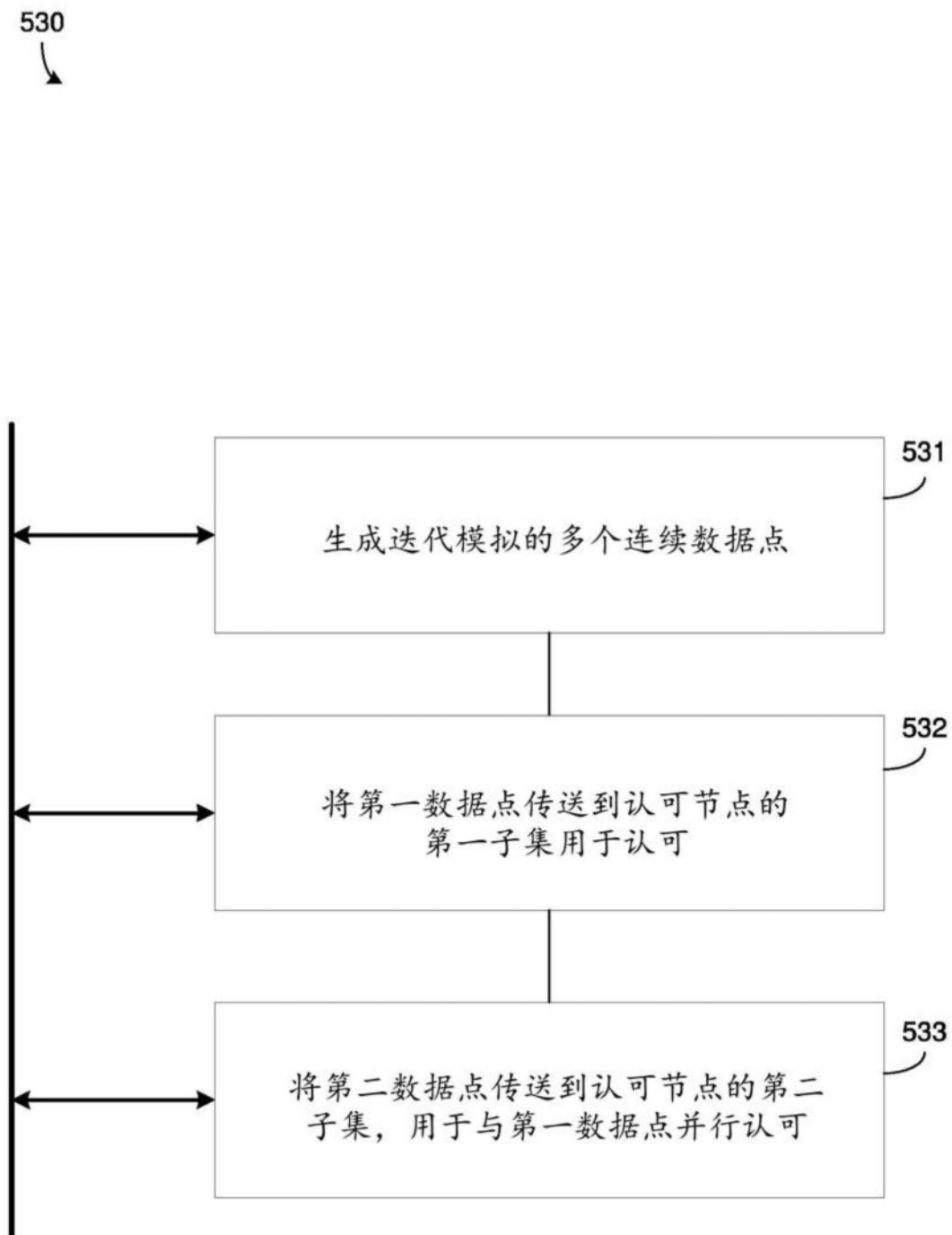


图5C

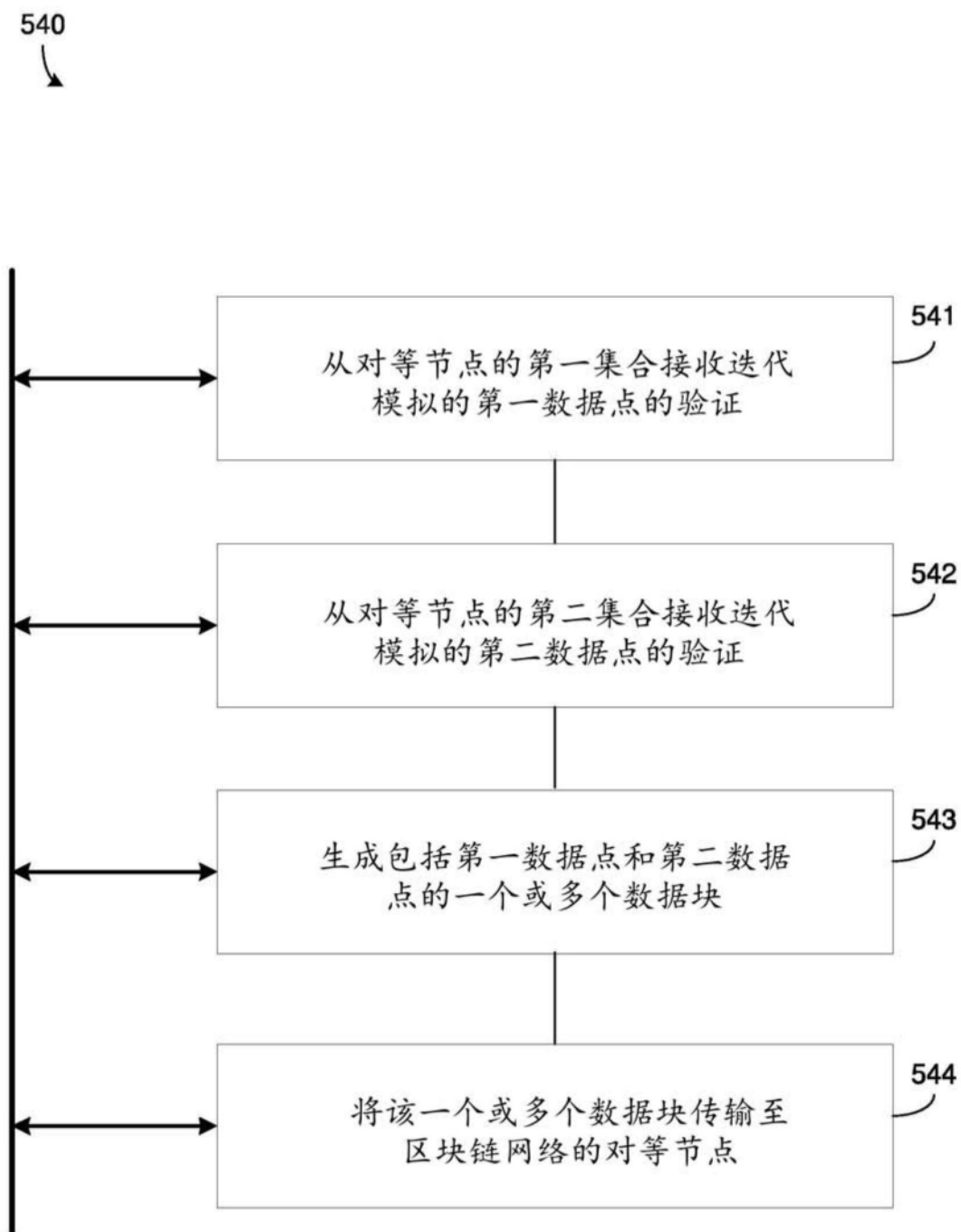


图5D

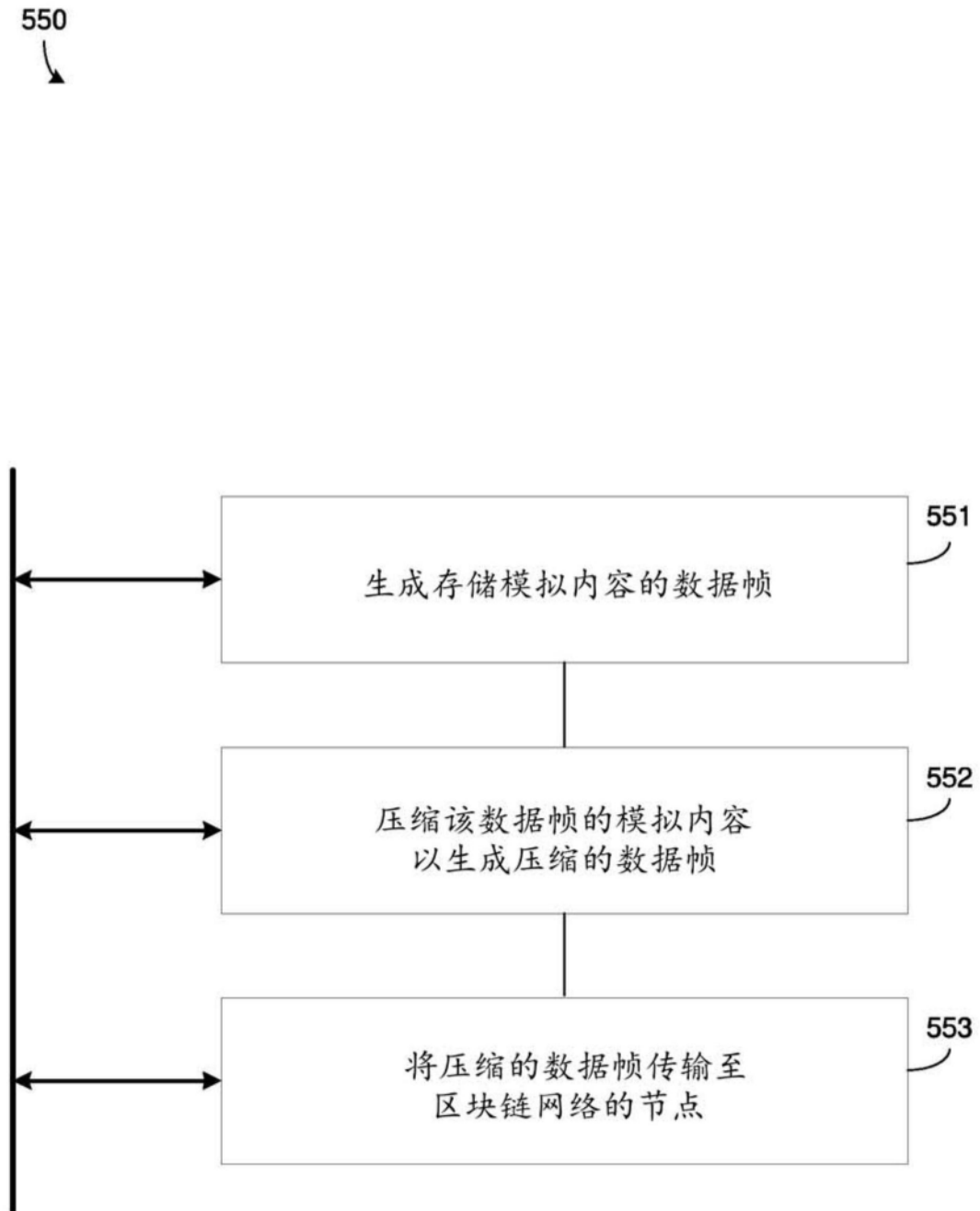


图5E

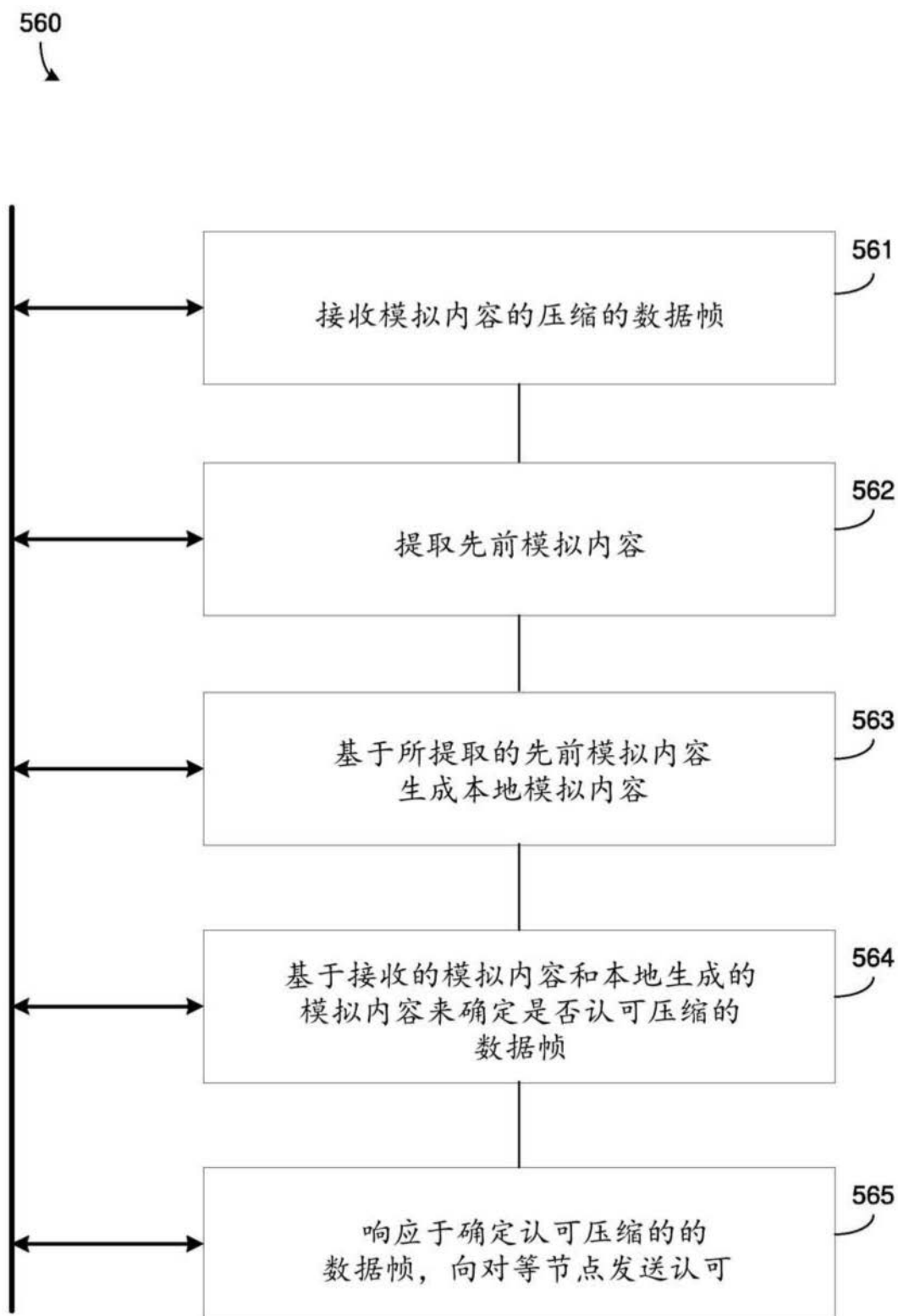


图5F

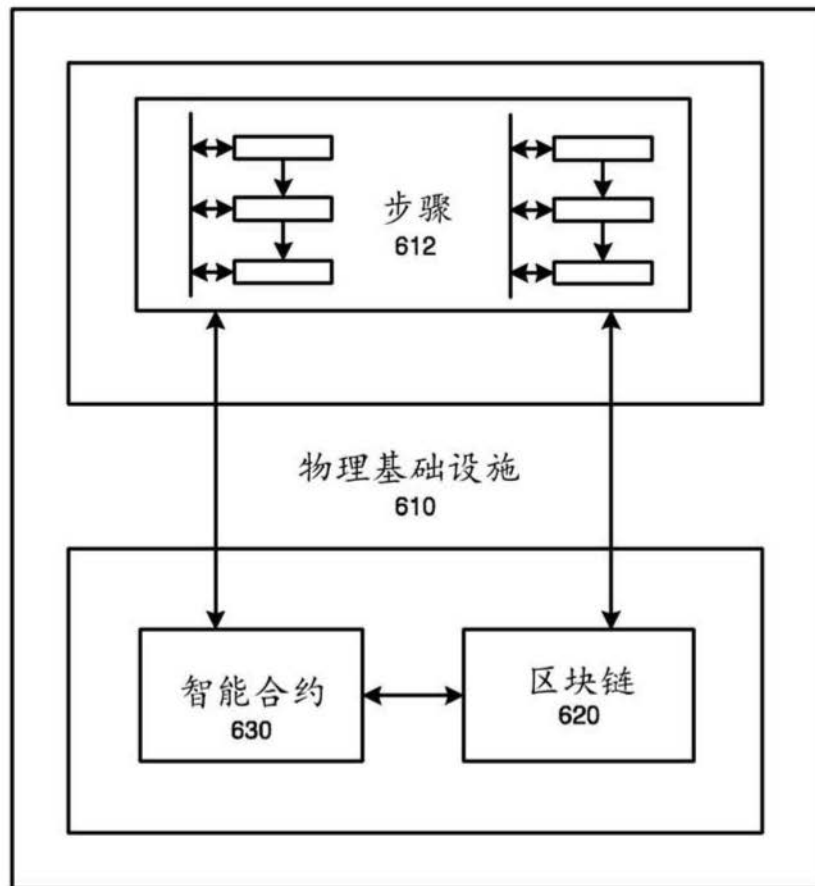
600

图6A

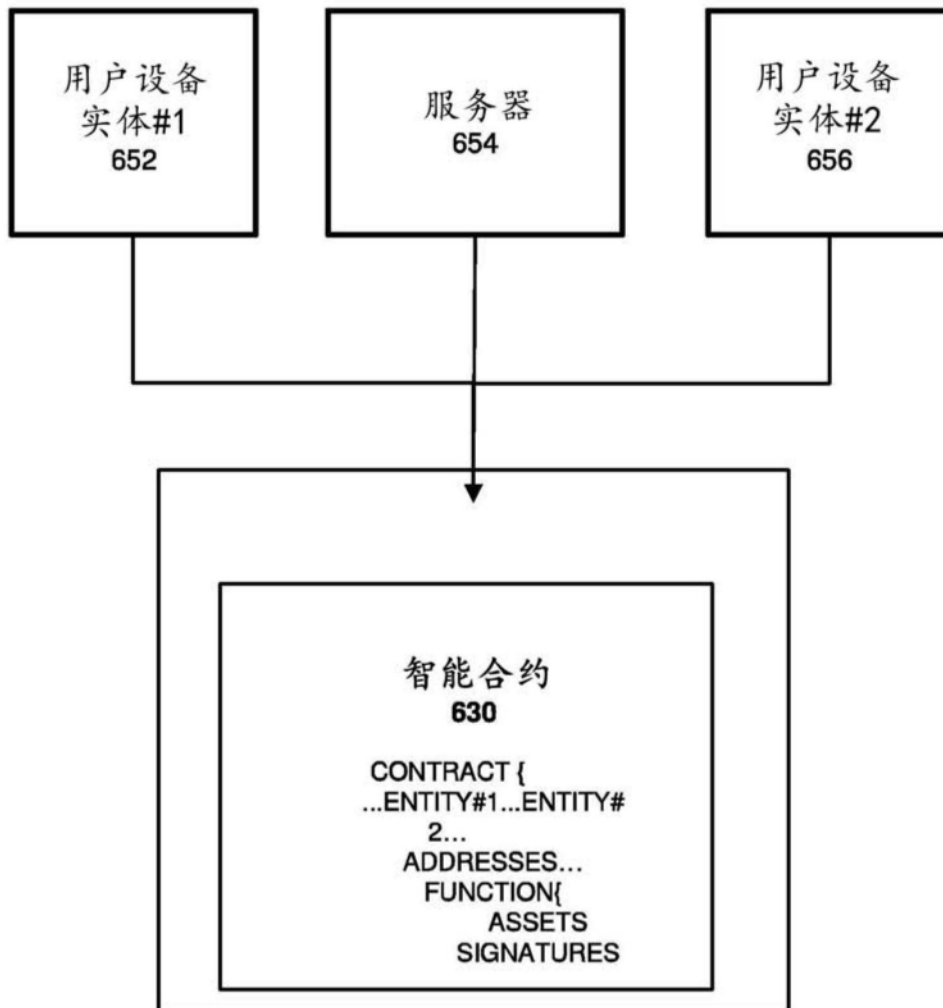
650

图6B

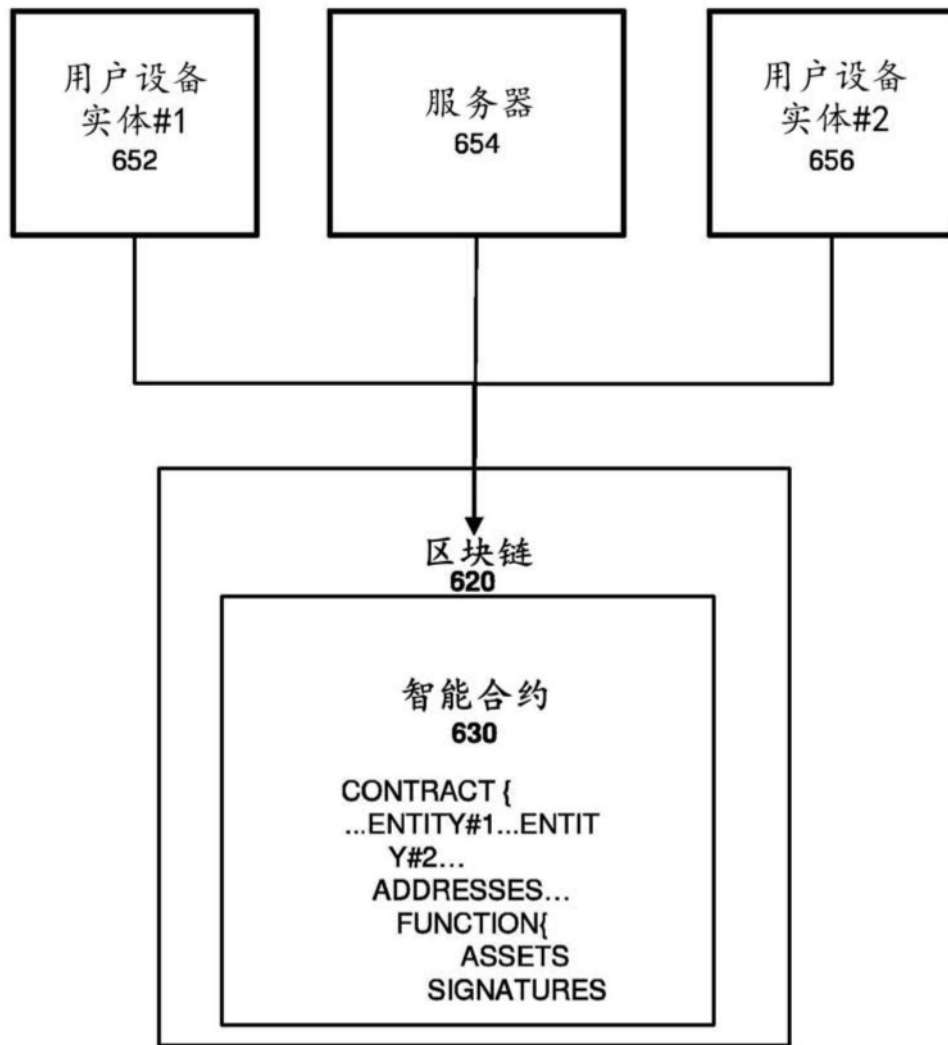
650

图6C

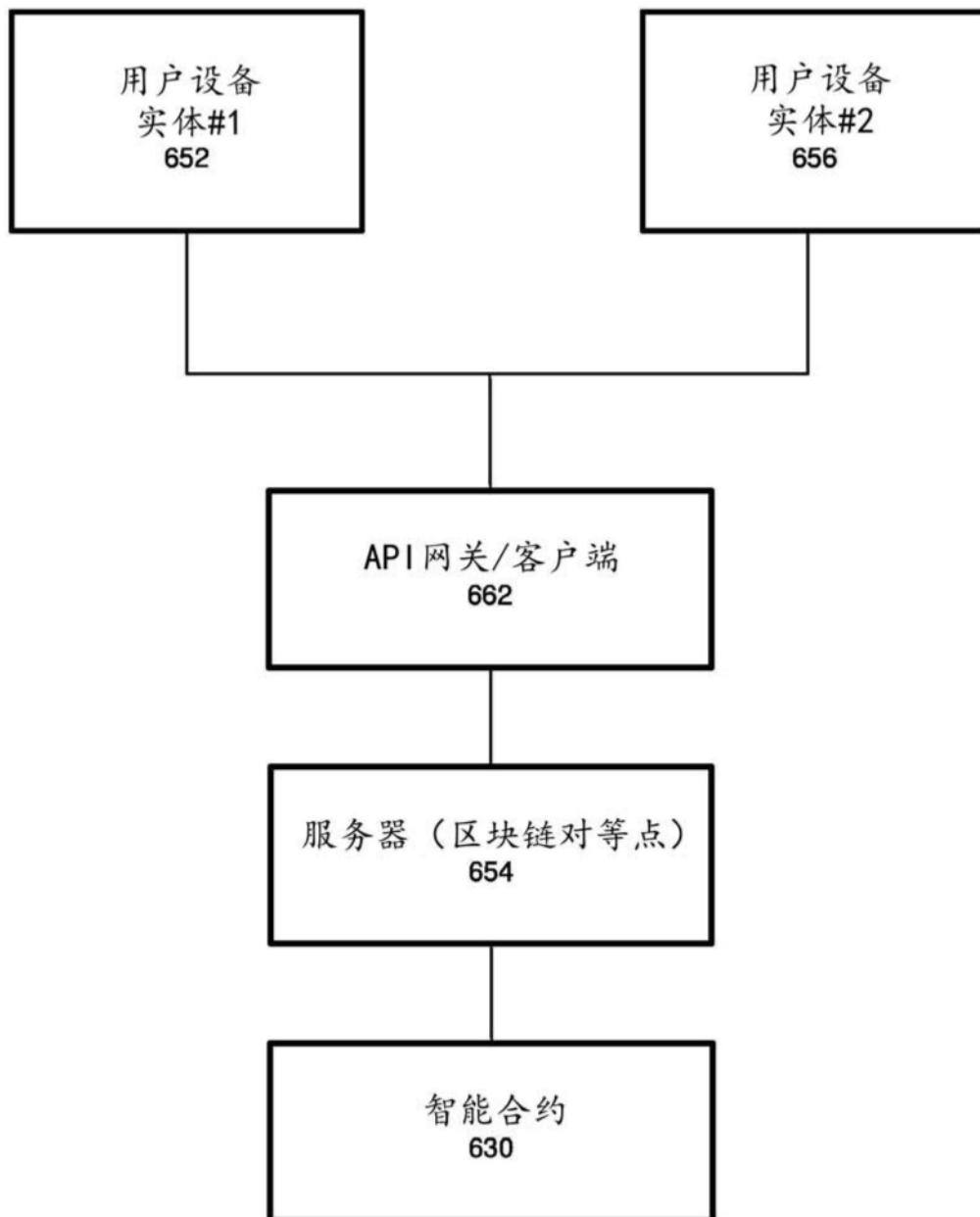
660

图6D

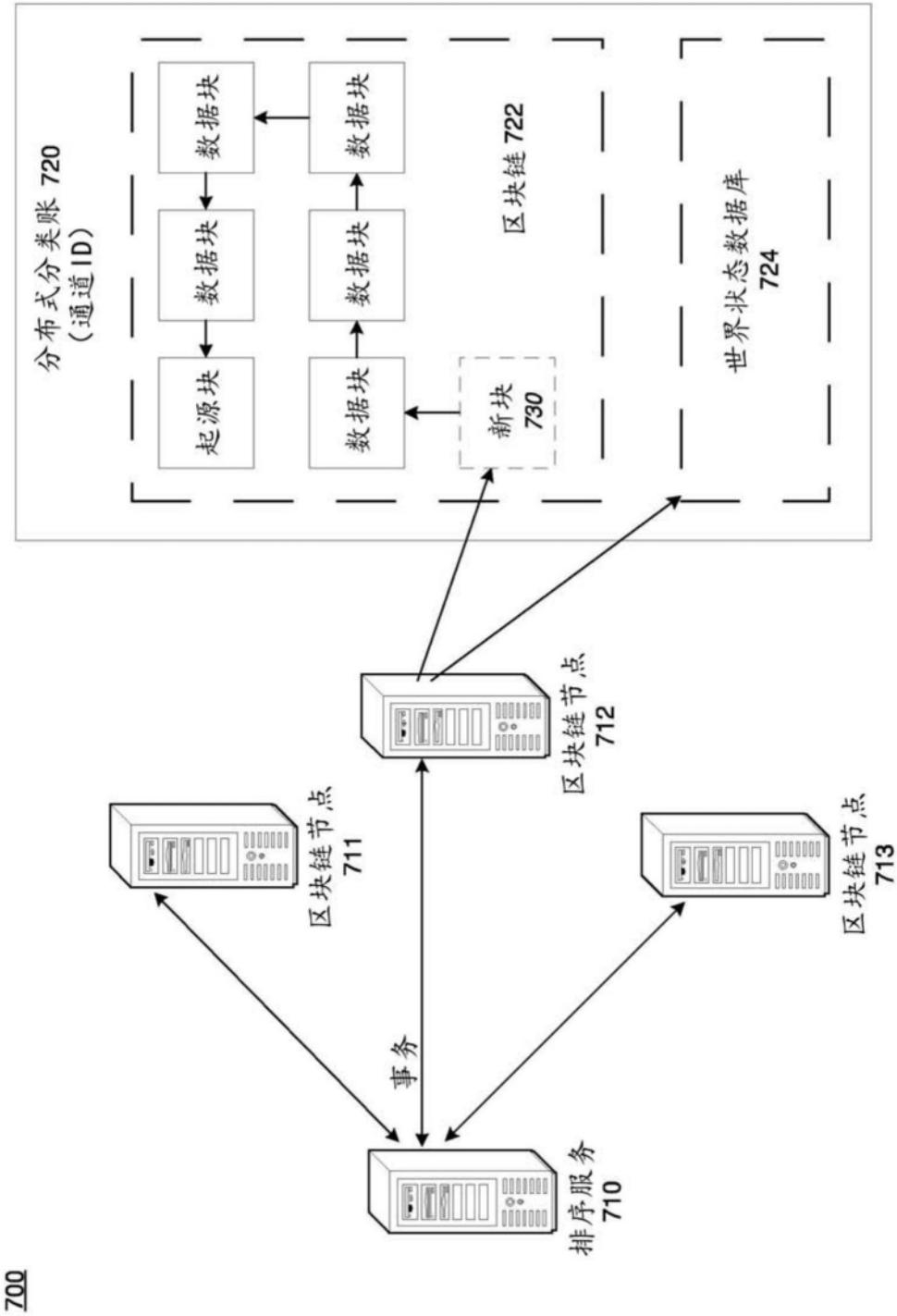


图7A

数据块 730

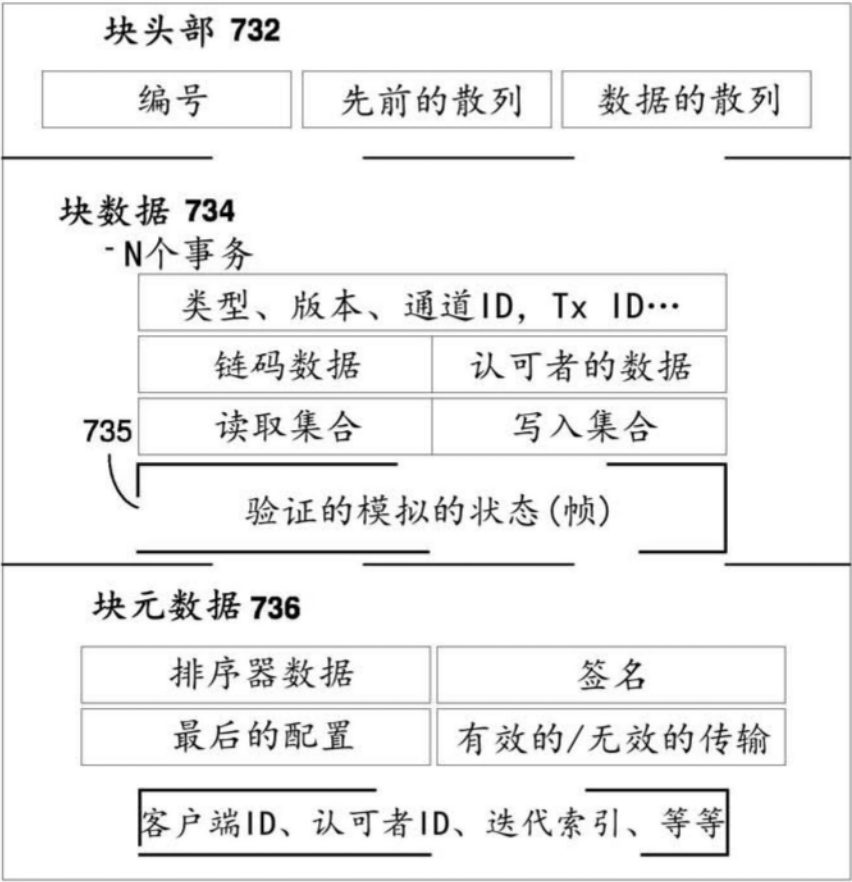


图7B

800

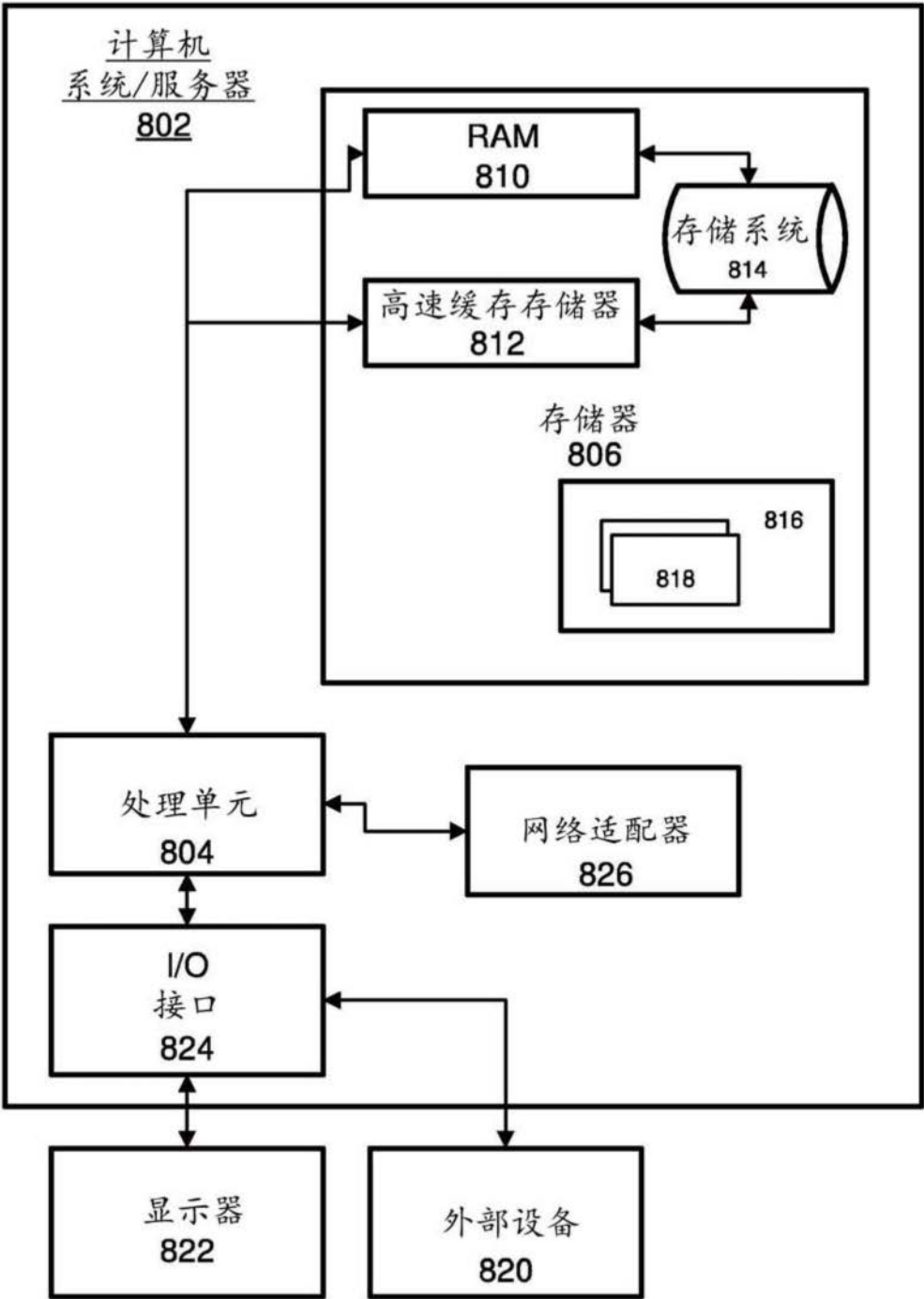


图8