



(12) 发明专利

(10) 授权公告号 CN 111165019 B

(45) 授权公告日 2021. 11. 30

(21) 申请号 201880064403.9

(22) 申请日 2018.09.26

(65) 同一申请的已公布的文献号
申请公布号 CN 111165019 A

(43) 申请公布日 2020.05.15

(30) 优先权数据
15/724,080 2017.10.03 US

(85) PCT国际申请进入国家阶段日
2020.04.01

(86) PCT国际申请的申请数据
PCT/CN2018/107566 2018.09.26

(87) PCT国际申请的公布数据
W02019/068246 EN 2019.04.11

(73) 专利权人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72) 发明人 张静静 魏威 钱留荣

(74) 专利代理机构 北京中博世达专利商标代理有限公司 11274

代理人 申健

(51) Int.Cl.
H04W 40/24 (2006.01)
H04L 12/24 (2006.01)

(56) 对比文件
US 2015103844 A1,2015.04.16
CN 106105117 A,2016.11.09
CN 106789660 A,2017.05.31
US 2016212696 A1,2016.07.21
US 2014317256 A1,2014.10.23
US 2016135111 A1,2016.05.12
US 2015236948 A1,2015.08.20
Ermin Sakic.“Towards adaptive state consistency in distributed SDN control plane”.《IEEE》.2017,全文.
Mohamed Aslan.“Adaptive consistency for distributed SDN controllers”.《IEEE》.2016,全文.

审查员 牛威

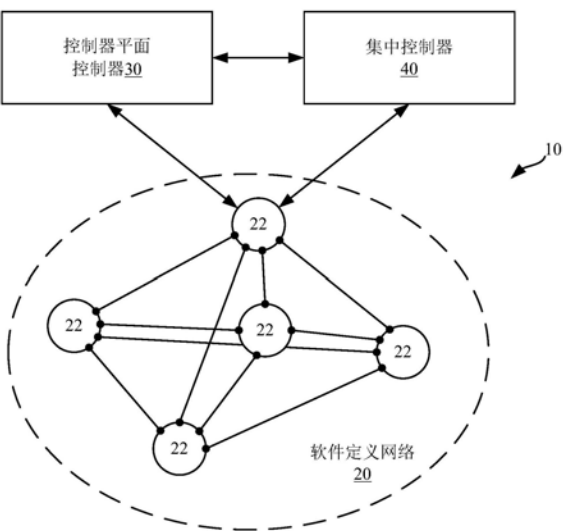
权利要求书1页 说明书18页 附图14页

(54) 发明名称

接入网中的控制器

(57) 摘要

本发明提供了一种基于软件定义网络 (software-defined network,SDN) 中的节点的物理拓扑的控制面管理,标识了接入网等SDN中的树形结构,使用所述树形结构来提供控制面管理。控制器配置所述SDN中的变更,并在配置其控制下的设备之后向其它控制器提供通知。在做配置决定时,控制器基于数据流的类型或方向来对下游和/或上游设备的状态进行优先级排序。控制器通过基于设备的分层位置对通知进行优先级排序来向其它控制器提供通知。



CN 111165019 B

1. 一种设备,其特征在于,包括:

非瞬时性存储器,包括指令;以及

与所述存储器通信的一个或多个处理器,其中所述一个或多个处理器执行所述指令来:

在控制器处接收来自联网设备的信号,所述控制器耦合到软件定义接入网的根节点处的服务器,所述软件定义接入网具有以分层树形结构排列的两个或更多联网设备;

基于所接收的信号,确定要传输给一个或多个受控设备的一个或多个指令,所述一个或多个受控设备位于相对于所述根节点的下游节点中;

将所述一个或多个指令传输给所述一个或多个受控设备;以及

在传输所述一个或多个指令后,将所述一个或多个受控设备的一个或多个变更通知给所述接入网中的一个或多个其它控制器。

2. 根据权利要求1所述的设备,其特征在于:

所述控制器是第一控制器,所述根节点是第一根节点,其中,所述第一控制器耦合到所述第一根节点;以及

所述一个或多个其它控制器包括耦合到所述软件定义接入网的第二根节点的第二控制器。

3. 根据权利要求1所述的设备,其特征在于:

确定所述一个或多个指令包括:响应于所述信号,识别所述一个或多个受控设备,并确定所述一个或多个受控设备的配置信息;以及

传输所述一个或多个指令包括:将所述配置信息传输给所述一个或多个受控设备。

4. 根据权利要求3所述的设备,其特征在于,所述一个或多个处理器执行所述指令来:

基于所述配置信息,配置所述一个或多个受控设备。

5. 根据权利要求1至4中任一项所述的设备,其特征在于,所述一个或多个处理器执行所述指令来:

在未与所述一个或多个其它控制器达成一致的情况下配置所述一个或多个受控设备。

6. 根据权利要求1至4中任一项所述的设备,其特征在于,所述两个或更多联网设备中的每个设备包括控制器、交换机和用户设备中的至少一个。

7. 根据权利要求1至4中任一项所述的设备,其特征在于,所述信号包括状态更新。

8. 根据权利要求1至4中任一项所述的设备,其特征在于,所述信号包括数据路由请求。

9. 根据权利要求1至4中任一项所述的设备,其特征在于,所述一个或多个指令包括用于通过所述接入网路由数据的指令。

10. 根据权利要求1至4中任一项所述的设备,其特征在于,所述一个或多个受控设备的所述一个或多个变更包括对在所述一个或多个受控设备中的每个受控设备处发生的所有事件的日志进行的一个或多个更新。

接入网中的控制器

[0001] 相关申请案交叉申请

[0002] 本发明要求2017年10月3日递交的发明名称为“接入网中的控制器通信”的第15/724,080号美国非临时专利申请案的在先申请优先权,该在先申请的全部内容以引入的方式并入本文本中。

技术领域

[0003] 本技术大体上涉及计算机网络领域,尤其涉及软件定义计算机网络中的控制器与接入网设备之间的通信。

背景技术

[0004] 分布式系统是通过计算机网络连接起来的两个或更多自主计算机的集合,在该计算机网络中,位于这些网络化计算机上的存储器、处理单元等组件进行通信并协调它们的行动以实现共同的目标。这些组件可以通过传递消息相互通信和协调。分布式系统的示例非常广泛,包括基于服务化架构(service-oriented architecture, SOA)的系统、大型多人在线游戏、点对点应用、共享存储系统和共享数据存储,例如文件系统、数据库、乐观复制系统和网页缓存。

[0005] 对分布式系统进行操作可能涉及位于网络中不同位置上的许多交互组件。因此,管理所有这些组件之间的一致性很重要,包括所存储的信息与这些组件执行的操作之间的一致性,以确保分布式系统根据系统的设计方式按照预期运行。为此,可以建立一致性模型。一致性模型是用于管理分布式系统中的一致性的一组规则。这些规则约束分布式系统的操作方式,以确保系统可预测地运行。

发明内容

[0006] 根据本发明的一个方面,提供了一种设备,包括:非瞬时性存储器,包括指令;以及与所述存储器通信的一个或多个处理器,其中所述一个或多个处理器执行所述指令来:在控制器处接收来自联网设备的信号,所述控制器耦合到软件定义接入网的根节点处的服务器,所述软件定义接入网具有以分层树形结构排列的两个或更多联网设备;基于所接收的信号,确定要传输给一个或多个受控设备的一个或多个指令,所述一个或多个受控设备位于相对于所述根节点的下游节点中;将所述一个或多个指令传输给所述一个或多个受控设备;以及在传输所述一个或多个指令后,将所述一个或多个受控设备的一个或多个变更通知给所述接入网中的一个或多个其它控制器。

[0007] 可选地,在任一前述方面中,所述控制器是第一控制器,所述根节点是第一根节点;所述一个或多个其它控制器包括耦合到所述软件定义接入网的第二根节点的第二控制器。

[0008] 可选地,在任一前述方面中,确定所述一个或多个指令包括:响应于所述信号,识别所述一个或多个受控设备,并确定所述一个或多个受控设备的配置信息;传输所述一个

或多个指令包括:将所述配置信息传输给所述一个或多个受控设备。

[0009] 可选地,在任一前述方面中,所述一个或多个处理器执行所述指令来:基于所述配置信息,配置所述一个或多个受控设备。

[0010] 可选地,在任一前述方面中,所述一个或多个处理器执行所述指令来:在未与所述一个或多个其它控制器达成一致的情况下配置所述一个或多个受控设备。

[0011] 可选地,在任一前述方面中,所述两个或更多联网设备中的每个设备包括控制器、交换机和用户设备中的至少一个。

[0012] 可选地,在任一前述方面中,所述信号包括状态更新。

[0013] 可选地,在任一前述方面中,所述信号包括数据路由请求。

[0014] 可选地,在任一前述方面中,所述一个或多个指令包括用于通过所述接入网路由数据的指令。

[0015] 可选地,在任一前述方面中,所述一个或多个受控设备的所述一个或多个变更包括对在所述一个或多个受控设备中的每个受控设备处发生的所有事件的日志进行的一个或多个更新。

[0016] 根据本发明的一个方面,提供了一种用于管理网络一致性的计算机实施方法,包括:在具有分层结构的软件定义网络的控制器处接收一个或多个信号,所述一个或多个信号与第一节点处的第一网络设备的第一状态和第二节点处的第二网络设备的第二状态相关联;确定所述第一网络设备的第一分层位置高于所述第二网络设备的第二分层位置;基于优先考虑所述第一网络设备的所述第一状态,然后再考虑所述第二网络设备的所述第二状态,来配置所述软件定义网络的一个或多个网络操作;以及执行所述一个或多个网络操作。

[0017] 可选地,在任一前述方面中,所述第一和第二网络设备包括网络交换机。

[0018] 可选地,在任一前述方面中,所述第一和第二网络设备包括用户设备。

[0019] 可选地,在任一前述方面中,在所述接入网的所述树形结构中,所述第二网络设备相对于所述第一网络设备处于分层较低的级别。

[0020] 可选地,在任一前述方面中,所述一个或多个信号包括状态更新。

[0021] 可选地,在任一前述方面中,所述一个或多个信号包括数据路由请求。

[0022] 可选地,在任一前述方面中,所述一个或多个网络操作包括向一个或多个网络设备发送指令以通过所述接入网路由数据。

[0023] 可选地,在任一前述方面中,所述一个或多个网络操作包括配置由所述控制器控制的一个或多个网络设备。

[0024] 根据本发明的一个方面,提供了一种非瞬时性计算机可读介质,存储用于管理网络一致性的计算机指令,所述指令在由一个或多个处理器执行时,使得所述一个或多个处理器执行以下步骤:在软件定义接入网的控制器处接收一个或多个信号,所述一个或多个信号标识第一联网设备的第一状态和第二联网设备的第二状态;确定所述第二联网设备的位置在所述第一联网设备的下游;基于所述第二联网设备的所述位置,在向所述接入网中的一个或多个其它控制器传输标识所述第二联网设备的所述第二状态的第二通知之前,向所述一个或多个其它控制器传输标识所述第一联网设备的所述第一状态的第一状态通知。

[0025] 可选地,在任一前述方面中,接收一个或多个信号包括:接收与所述第一状态相关

联的第一信号和与所述第二状态相关联的第二信号;所述第二信号在所述第一信号之前接收。

[0026] 可选地,在任一前述方面中,所述第一和第二联网设备中的每个设备包括控制器、交换机和用户设备中的至少一个。

[0027] 可选地,在任一前述方面中,所述一个或多个信号包括状态更新。

[0028] 可选地,在任一前述方面中,所述一个或多个信号包括数据路由请求。

[0029] 根据本发明的一个方面,提供了一种包括指令的非瞬时性存储器;以及与所述存储器通信的一个或多个处理器,其中所述一个或多个处理器执行所述指令来:在控制器处接收将数据从源网络设备传输到目标网络设备的请求;基于所接收的请求,确定所述数据传输涉及上游通信还是下游通信;在确定所述数据传输涉及上游通信之后,基于优先考虑上游网络设备的状态,为一个或多个网络设备分配网络带宽;以及在确定所述数据传输涉及下游通信之后,基于优先考虑下游网络设备的状态,为一个或多个网络设备分配网络带宽。

[0030] 本发明内容简单地介绍了下文在具体实施方式中进一步描述的一系列概念。本发明内容并非旨在识别所要求保护的主题的关键特征或必要特征,也非旨在帮助确定所要求保护的主题的范围。所要求保护的主题不限于解决背景技术中所述的任意或所有缺点的实施方式。

附图说明

[0031] 图1示出了用于传送数据的网络环境。

[0032] 图2为示例性网络环境的方框图。

[0033] 图3为软件定义网络 (software-defined network, SDN) 网络的示例性实施例的示意图。

[0034] 图4为包括多个控制器的网络环境的一个示例的方框图。

[0035] 图5为包括树形拓扑中的节点的网络环境的一个示例的方框图。

[0036] 图6为根据本技术一实施例的示例性接入网的高级方框图。

[0037] 图7为展示位于节点处的网络设备的示例性接入网的方框图。

[0038] 图8为示例性事件时间轴。

[0039] 图9为描述包括在一个实施例中控制器可以执行的网络操作的过程的流程图。

[0040] 图10为描述一个实施例中的网络节点管理过程的流程图。

[0041] 图11为根据一个实施例的描述基于设备位置的控制器通知过程的流程图。

[0042] 图12为根据一个实施例的描述网络节点管理过程的流程图。

[0043] 图13为描述基于数据流是在上游方向还是下游方向来进行数据路径决定的过程的流程图。

[0044] 图14为计算系统的高级方框图。

具体实施方式

[0045] 所公开的技术针对软件定义网络中的控制器与接入网设备之间的通信。例如,软件定义接入网可以提供多个用户设备(例如客户端设备)与用户网络或互联网等一个或多

个其它网络之间的连接。接入网通常包括多个网络节点,这些网络节点中的一部分网络节点之间存在连接链路。

[0046] 在一个实施例中,控制器可以做出决定并配置网络设备,然后将变更通知给其它控制器。控制器可以直接控制和配置网络设备,并将随后进行的变更通知给其它控制器。在一个示例中,这利用了接入分发树的独特特征,接入分发树在执行网络操作之前不需要管理一致性。本技术可以使控制器能够更快地响应网络事件,而无需实现与其它控制器的一致性。

[0047] 通过在控制和配置网络设备之后向其它控制器发送通知,可以保持接入网中的多个控制器之间的一致性。这允许备份控制器在主控制器的接入分发树中保持关于网络设备当前状态的更新信息,以便在主控制器发生故障时,备份控制器可以快速接管。

[0048] 在一个实施例中,该方法适用于接入分发树的主控制器。主控制器控制接入网的一个独立部分。可能不需要与其它控制器协调来执行常规网络操作,例如路由数据传输和配置网络设备。因此,主控制器在控制和配置网络设备之前可能不需要实现与其它控制器的一致性,而某些传统方法可能需要。

[0049] 根据一个实施例,控制器基于优先考虑上游设备的状态(例如在线或离线),然后再考虑下游设备的状态,来做出决定。控制器决定可以包括关于如何将数据从源设备路由到另一网络设备处的目标目的地的决定,或者关于如何配置网络设备的决定。在一个实施例中,根据分层树形结构来排列接入网。可以在彼此直接处于上游和下游的每对设备之间提供单个物理通信连接。

[0050] 在一个实施例中,来自下游设备的信息,包括控制面信号,可以经过上游设备。上游设备在树形结构的接入网中可能处于更中心的位置。对于来自任何给定网络设备的数据,包括来自下游设备的数据,能否成功通过接入网传输到达根节点,上游设备的状态可能更具有决定性。例如,给定上游设备A和下游设备B,如果A离线,那么无论B是在线还是离线,来自B的数据都无法传输到根节点。当A在线时,B的状态就与确定是否能够成功将数据从B路由到根节点有关。因此,在接入网中的数据传输的上下文中确定数据路径决定时,可以优先考虑上游设备的状态。

[0051] 根据一个实施例,控制器在报告下游设备的状态之前报告上游设备的状态。考虑到接入网的分层树形结构,上游设备位于接入网中更中心的节点处,并且相对于下游设备更多地通过拓扑连接到网络中的其它节点。由于上游设备的连接性更强,因此它们更有可能在促进数据传输方面发挥作用。此外,上游设备的状态最终决定了来自下游设备的控制面信号和数据面信号等数据能否到达更多上游设备。来自下游设备的数据经过上游设备,以便到达更上游的网络设备和控制器。因此,在一个实施例中,在通知下游设备的状态之前将上游设备的状态通知给控制器。

[0052] 当主控制器发生故障时,备份控制器接管。在这样的事件期间,备份控制器可以基于其对发生故障的主控制器的接入分发树状态的了解做出决定。如果备份控制器具有关于上游设备状态的最新信息,那么其可以做出更好的数据路径决定,因为这些数据路径更有可能经过上游设备而不是下游设备。通过在向其它控制器发送通知时优先考虑上游设备,当主控制器发生故障时,备份控制器更有可能顺利成功地接管。

[0053] 在一个实施例中,接入网中的接入分发树可以包括不止一个控制器,包含一个主

控制器以及一个或多个从控制器。这些控制器在执行网络操作时可以相互协调。在一个实施例中,在做控制器决定时以及在向其它控制器发送关于网络设备状态的通知时,优先考虑上游设备。如果在控制和配置网络设备与向其它控制器发送通知之间发生延迟,那么这可以将这种延迟的影响降到最小。

[0054] 根据一个实施例,提供了一种技术,避免了在做出和执行控制器决定时在多个控制器或轮询设备上寻求一致。通过直接作用于网络设备,控制器可以更快地响应网络事件。通过将网络变更通知给其它控制器以及通过优先考虑上游设备,可以将多个控制器之间的一致性的损失减到最小。可以提供更低的网络时延、更高的网络可用性和整体提升的用户体验。在替代性实施例中,上述通信方案可以组合应用,也可以单独应用。

[0055] 分布式系统建立在计算机网络之上,计算机网络也称为数据网络。数据网络是相互连接的网络设备的集合,这些网络设备包括路由器、交换机、网桥、无线收发器、台式计算机、智能手机和其它通过数据网络传输数据的设备,等等。每个网络设备可以是数据网络中的一个节点,该节点可以可通信地连接到数据网络中的一个或多个其它节点。每个节点与至少一个设备相关联。与通用数据网络相比,分布式系统是一种具有如下附加功能的数据网络:协调多个网络设备以实现共同目标,例如,将数据从源网络设备传输到数据网络中的目标目的地。

[0056] 图1示出了用于传送数据的示例性网络环境50。网络50包括基站58,基站58包括拥有覆盖范围54的一个或多个天线。基站(base station,BS) 58可以包括通信控制器。网络环境50包括多个用户设备(user equipment,UE) 56和一个网络52,例如回传网络。描述了两个UE,但是可以存在更多UE。基站58中的通信控制器可以是能够通过尤其是与UE 56建立上行连接(虚线)和/或下行连接(点线)来提供无线接入的任何组件。基站58可以包括增强型基站(enhanced base station,eNB)、微微蜂窝基站、毫微微蜂窝基站和其它无线启用设备。UE 56可以是能够与移动电话、智能电话、平板电脑、传感器等BS 58建立无线连接的任何组件。网络52可以是允许在BS 58与远程终端(未示出)之间交换数据的任何组件或组件的集合。在一些实施例中,网络52可以包括各种其它无线设备,例如中继器、毫微微蜂窝基站等。

[0057] 在一个实施例中,可以将分布式系统实现为一个或多个软件定义网络(software-defined network,SDN)。SDN是一种针对计算机网络的方法,该方法创建计算机网络的底层功能的抽象并提供用于控制和管理计算机网络的行为的接口。具体地,SDN通过将网络的数据转发功能(即,数据面)与其管理功能(即,控制面)分离来简化现代网络,管理功能包括先前由网络节点处的设备执行的数据路由和资源分配功能。换言之,控制面和数据面分别是控制和数据转发功能的抽象,由SDN以不同的方式处理,即使在本技术的实施例中它们可以在同一物理网络基础设施上实现。在替代性实施例中,可能存在单独的物理网络基础设施用于实现控制面功能。

[0058] 信号以控制面数据包和数据面数据包的形式沿着控制面和数据面发送。虽然控制面数据包和数据面数据包可以沿着同一物理网络基础设施发送,但是SDN可以将它们区分并以不同的方式处理。例如,SDN可以优先考虑发送控制面数据包,然后发送数据面数据包,特别是在网络流量拥塞或网络带宽较低的情况下。

[0059] 支持软件定义网络的网络节点处的网络设备可以用于执行数据面功能,而控制面功能可以由SDN控制器提供。控制器是一种控制和配置网络设备以便于将数据从源网络设

备传输到网络中的预期目的地的设备。控制器通过以下方式来实现这一点：从网络设备接收信息，并基于其通过SDN的控制层接收的信息向网络设备发出指令。控制器可以实现为软件应用或硬件组件。控制器可以实现为SDN中节点处的网络设备的一部分，也可以实现为SDN的中心服务器的一部分。可替代地，控制器也可以实现为连接到SDN的独立设备。SDN可以对控制器进行抽象并将其视为与数据面上的网络节点分离的独立实体。

[0060] 图2为可以实践所公开技术的各实施例的示例性网络环境10的框图。网络环境包括软件定义网络 (software-defined network, SDN) 20、至少一个控制面控制器30和至少一个集中控制器40。SDN 20、控制面控制器30和集中控制器40通过光学方式、电气方式或无线方式相互通信。虽然示出了控制器30与SDN 20分开，但是，例如，控制器30可以在SDN 20中的其中一个节点22处实现。控制器30可以在其中一个节点22处的服务器或其它计算设备中实现，或者在远离SDN 20的计算设备中实现。

[0061] 在一个实施例中，SDN 20可以实现为提供用户设备56与网络52之间的连接的接入网。例如，SDN 20可以包括多个网络节点，这些节点提供多个可配置数据路径，从而提供用户设备56与网络52之间的连接。

[0062] 在一个实施例中，SDN 20是一个分组交换网络，在该网络中，数据流量使用多个数据包或多个帧沿着网络路径或网络路线进行传输。基于控制器30计算的和/或节点22开发的路径，这些数据包可以沿着通过多协议交换 (multiprotocol label switching, MPLS) 或通用MPLS (Generalized MPLS, GMPLS) 等信令协议建立的流量工程标记交换路径进行路由或交换。SDN 20包括使用光学链路、电气链路或无线链路相互耦合的多个节点22。SDN 20还可以包括多个域，例如自治系统 (autonomous system, AS) 域或IGP区域，每个域都可以包括对应于同一地址管理和/或路径计算责任的一组网元。这些域通过物理机制 (例如位置、连接等) 和/或逻辑方法机制 (例如网络拓扑、协议、通信层等) 来组织。不同域相互耦合，每个域都包括一些节点22。

[0063] 节点22为支持通过SDN 20传输多个数据包的任何设备或组件。例如，节点22可以包括网桥、交换机、路由器或这些设备的各种组合。节点22可以包括用于从其它节点22接收数据包的多个入端口、确定将帧发往哪些节点22的逻辑电路、用于向其它节点22传输帧的多个出端口。在一些实施例中，至少一些节点22是标记交换路由器 (label switched router, LSR)，用于修改或更新在标记交换网络20中传输的多个数据包的标记。在一些实施例中，一些节点22是标记边缘路由器 (label edge router, LER)。例如，处于SDN 20边缘的节点22用于插入或移除在标记交换网络110与外部网络之间传输的多个数据包的标记。沿着一条路径的第一节点22和最后一个节点22有时分别称为源节点或头端节点和目的节点或尾端节点。虽然示出SDN 20中存在四个节点22，但是SDN 20可以包括任何数量的节点22。另外，节点22可以位于SDN 20中的不同域内，并且可以用于跨多个域进行通信。例如，对应于不同域的节点22可以沿着跨多个域建立起来的路径来交换多个数据包。

[0064] 控制面控制器30为用于协调SDN 20内各种活动的任何设备。控制器30可以包括网络管理系统 (Network Management System, NMS) 或操作支持系统 (Operations Support System, OSS)。具体地，控制面控制器30从SDN 30接收路由请求并返回对应的路径信息。控制面控制器30可以执行路径计算并向至少一个节点22转发计算出的路径。控制面控制器30可以位于SDN 30外部的组件中，例如位于外部服务器中，还可以位于SDN 20内部的组件中，

例如位于节点22处的服务器中。

[0065] 集中控制器40为用于协调控制面控制器30进行的各种活动的任何设备,例如使用多个控制面控制器来管理SDN 20内的控制面功能。集中控制器40可以位于SDN 20外部的服务器等组件中,还可以位于SDN 20内部的节点22处服务器等组件中。在一个实施例中,集中控制器执行本文描述的各个过程以生成网络配置。网络配置可以提供给一个或多个控制面控制器30。

[0066] 控制器可以基于OpenFlow协议来操作。SDN是一种能够在不需要物理接入到网络中设备的情况下对网络流量进行可编程集中控制的网络拓扑。SDN可以采用利用传输控制协议/互联网协议(Transmission Control Protocol/Internet Protocol,TCP/IP)的互联网协议(Internet Protocol,IP)网络。SDN可以将数据面等数据转发能力与路由、资源和先前在网络节点上执行的控制面等其它管理功能分离开来。将网络的控制面与数据面分离使得网络控制器能够通过全局优化的流量工程和路由来有效地控制网络流量,而全局优化的流量工程和路由与本地优化的最短路径优先不同。SDN还可以简化网络操作,或者甚至有能力通过扩展的数据路由向量使网络扁平化。

[0067] 可以设计一种控制器来控制和管理整个网络,例如接入网(例如包括千兆比特无源光网络(gigabit passive optical network,GPON)、以太网无源光网络(Ethernet passive optical network,EPON)、G.Fas.)。SDN控制器的示例包括开放网络操作系统(open network operating system,ONOS)和Open Daylight(ODL)。控制器可以利用各种协议,包括netconf、OpenFlow或私有供应商协议。在一些示例中,在供应商设备与SDN控制器之间提供硬件抽象层。Voltha是能够提供的硬件抽象层的一个示例。在一些示例中,OpenFlow和/或Netconf协议可以用于在VOLTHA硬件抽象层与ONOS或ODL控制器之间进行通信。

[0068] 控制面控制器可以包括未示出的路径计算引擎(path computation engine,PCE)或者与PCE进行通信。PCE是用于基于路径计算请求等为SDN 20执行所有或部分路径计算的任何设备或组件。PCE可以从控制面控制器30、节点22或两者接收用于计算路径的信息。PCE处理该信息以获取路径。例如,PCE计算路径并确定包括沿着该路径的LSR的节点22。然后,PCE可以向控制面控制器30或直接向至少一个节点22发送所有或部分计算出的路径信息。此外,PCE通常耦合到或包括流量工程数据库(traffic-engineering database,TED)、P2MP路径数据库(P2MP Path database,PDB)、P2P路径数据库、光通道性能监测(optical performance monitor,OPM)、物理层约束(physical layer constraint,PLC)信息数据库或它们的组合,他们都可以用于计算路径。PCE可以位于SDN 20外部的组件中,例如位于外部服务器中,还可以位于SDN 20内部的组件中,例如位于节点22处的组件中。PCE可以配置为控制面控制器30的一部分。可替换地,PCE可以配置为集中控制器40的一部分。

[0069] 请求设备向PCE发送路径计算请求。该请求设备是请求由PCE来执行路径计算的任何客户端应用。该请求设备还可以是提出这种请求的任何网络组件,例如控制面控制器30,或LSR等任何节点22。例如,该请求设备可以在SDN 20的单个域内或跨多个域向PCE请求P2MP路径或P2P路径。另外,该请求设备可以通过路径计算请求和/或通过链路状态公告(link state advertisement,LSA)的广播信令等向PCE发送至少一些路径所需信息。

[0070] 在一个实施例中,SDN的中心服务器可以访问描述SDN的网络信息。通过从SDN中节

点处的网络设备接收网络信息、对SDN中的网络设备轮询,或读取先前存储在中心服务器处的网络信息,可以访问网络信息。网络信息可以包括与网络中的节点有关的流量信息、物理拓扑信息和/或与SDN中的节点有关的其它信息。

[0071] 图3为SDN网络100的示例性实施例的示意图。网络100包括SDN控制器130、多个网络节点122和多个末端节点124。网络节点122包括交换机、路由器、网桥和/或用于在网络中接收和/或转发数据的任何其它设备。控制路径由虚线表示,数据路径由实线表示。系统配置、管理信息和路由/转发表信息通过控制路径在网络控制器130与网络节点122之间交换。

[0072] 数据包从末端节点124接收并通过数据路径在网络节点122之间转发。例如,来自充当发布者的末端节点124-1的数据由充当入边界路由器(Ingress Border Router,IBR)的网络节点122-2接收、通过充当发射路由器(Transit Router,TR)的网络节点122-1进行路由,并且通过充当出边界路由器(Egress Border Router,EBR)的网络节点122-3传递到充当目的节点的末端节点124-2。边界路由器是连接到SDN域外部的至少一个节点的SDN域的边缘上的路由器,IBR是从SDN域外部接收流量的SDN边界路由器,EBR是在SDN域外部发送流量的SDN边界路由器。TR是在SDN域内传输流量的SDN路由器并且没有接口连接SDN域外部。根据通过LSP传输的业务流,单个边界路由器可以用作IBR、EBR或两者。末端节点124为用于传输、接收、发起和/或终止数据的任何网元,或者在替换实施例中,为IP网络、MPLS网络等其它网络。在一些实施例中,网络控制器130为用于控制网络节点122和末端节点124的通用网络控制器。网络控制器130为用于执行控制路径和/或控制面功能的任何设备,例如创建网络地图和定义路由表中的信息,其中路由表定义如何路由传入数据包。网络控制器130还用于执行控制面的管理和控制功能,包括路由和资源管理。网络节点122和末端节点124包括根据某个标准通过网络100接收和传输数据的设备。网络节点122和末端节点124中的至少一些节点和网络控制器130可以符合如在开放网络基金会(Open Networking Foundation,ONF)文档OpenFlow交换机规范第1.3.4版、ONF文档OpenFlow控制器交换机NDM同步第1.0版和ONF文档软件定义网络:网络新规范,ONF白皮书(统称为OpenFlow)中定义的某个标准。

[0073] 尽管图3中未示出,但是控制器130从网络节点接收数据并向网络节点传输消息,其中数据和消息可以用于管理和控制网络节点(例如,控制器能够基于从网络节点接收到的数据来确定网络拓扑)。一些传入消息或部分传入消息被转换成标准独立格式以便由网络控制器130中的一些模块进行处理。标准独立格式以抽象网络控制数据模型为基础,该模型提供各种标准格式的属性或特性的抽象。网络控制器130通过OpenFlow等各种应用程序接口(application programming interface,API)协议与网络节点进行交互。控制器确定网络的全局网络拓扑。根据全局网络拓扑、状态信息、动态业务流/量信息和其它网络状态信息,网络控制器确定如何分配资源和路由应用、信息和/或流经网络的数据包。

[0074] 图4为包括用于管理和控制多个网络节点322的多个SDN控制器330的另一示例性网络环境300的框图。可以在需要管理大量网络节点的实施方式中使用多个控制器,还可以使用多个控制器来提供高可用性。在图5中,控制器330-1包括与节点322-1、322-2和322-3的控制链路。控制器330-2包括与节点322-7、322-8和322-9的控制链路。控制器330-3包括与节点322-4、322-5和322-6的控制链路。

[0075] 在这种方式下,将节点划分为三个节点分区,还称为节点集群,其中每个分区包括

三个网络节点。同样地,可以称三个控制器构成所有网络节点对应的控制器集群。通过使用多个控制器,在发生故障时一个控制器能够接管另一个控制器,因此使用多个控制器可以提高可用性。节点322-1、322-2和322-3构成由控制器330-1控制的节点集群。节点322-7、322-8和322-9构成由控制器330-2控制的节点集群。节点322-4、322-5和322-6构成由控制器330-3控制的节点集群。

[0076] 在一个实施例中,节点或节点处的设备只与单个主控制器进行通信。该节点可以连接到其它控制器,但是只在给定时间点与主控制器进行通信。然而,如果该节点对应的主控制器发生故障,则连接到该节点的另一控制器可以接管由发生故障的控制器控制的节点的控制面功能。例如,为了响应指示控制器330-1发生故障的事件,对节点322-1、322-2和322-3的控制可以转移到控制器330-2。

[0077] 在一个实施例中,可以将包括多个节点的SDN配置为由多个控制器进行控制。开放式应用编程接口(application programming interface,API)服务,例如OpenFlow协议,可以管理数据面与控制面之间的交互,而且能够实现网络内的网络节点和SDN控制器的非供应商特定组合。因此,软件定义网络结合开放式API服务可以为现代网络带来许多好处,包括:增加网络虚拟化、灵活控制和利用网络、根据具体需求定制网络。

[0078] 图5为网络环境350的框图,继续如图4所示的网络节点322的示例。图5示出了包括接入网内的中间节点之间的连接链路的物理拓扑。对节点和链路结构的研究揭示了接入网的公共属性,即,树形拓扑拥有至少一个根节点,一个或多个下游节点的数据穿过该至少一个根节点。

[0079] 在图5中,节点322-1是每个其它节点的根节点。树形拓扑由根节点322-1和下游节点构成。图5具有源自根节点322-1的五个独立的连接路径。树的第一分支包括根节点322-1和下游节点322-2和322-3。树的第二分支包括根节点322-1和下游节点322-2和322-6。树的第三分支包括根节点322-1和下游节点322-5和322-9。树的第四分支包括根节点322-1和下游节点322-4和322-7。树的第五分支包括根节点322-1和下游节点322-4和322-8。

[0080] 在如图所示的树形拓扑中,所有节点并不像如图1所示的网状架构等许多架构中那样互连。分支没有连接边缘,因此不直接进行通信。相反,分支通过一个或多个中间连接节点进行通信。例如,每个分支上的叶节点没有直接的连接链路。因此,节点之间的通信必须经过至少一个上游节点才能到达另一叶节点。例如,来自节点322-3的通信必须经过节点322-3、322-1和322-5才能到达节点322-6。

[0081] 传统上,在为SDN中节点进行控制器分配时,网络配置不考虑网络的树形拓扑。此外,尚未考虑不同控制器分配的流量影响。因此,SDN的正常运行时间尚未得到优化。

[0082] 例如,考虑到图6中的控制器分配。SDN中的不同节点发生故障会对整个SDN产生不同的影响。例如,节点322-1是SDN中所有其它节点的根节点。如果根节点322-1发生故障,则与SDN中所有其它节点的通信都中断。同样地,如果控制根节点322-1的控制器330-1发生故障,则SDN中的所有通信都中断。然而,如果叶节点322-7、322-8或322-9发生故障,则只有与这些节点的通信会中断。同样地,如果控制器330-2发生故障,则只有与叶节点322-7、322-8或322-9的通信会中断。

[0083] 在一个实施例中,本技术包括网络控制器与设备之间的一种简单、高效的通信方案。该通信方案利用具有分层树形结构的网络的独特拓扑,改进了传统的一致性管理方法。

这种网络的一个示例是接入网,接入网可以实现为SDN。

[0084] 接入网将用户及其客户端设备,例如台式电脑、平板电脑和智能手机,连接到网络服务提供商。更具体地,接入网由多个联网设备组成,这些设备可以包括控制器、交换机、客户端设备和在接入网中充当节点的其它网络设备。为了将成本降到最低,使用最少数量的网络连接,例如网络电缆线,来构建接入网。通过根据分层树形结构排列网络设备及其连接来实现这一点。在这种树形结构中,根节点位于树的根基处。然后,连接以树形方式从根节点向外扩展以连接其它网络设备。树底部的设备位于树的叶节点处。在一个实施例中,直接连接到接入网的一个或多个中心服务器的设备位于树的根节点处。因此,例如,根节点可以是具有到另一网络或服务器的上游连接的节点。虽然本文结合接入网描述了本技术,但是应理解,本技术也可以在具有类似分层树形拓扑的其它类型的网络上实施。

[0085] 对于接入网的分层结构,可以将网络设备描述为相对于彼此处于上游或下游。沿着从叶节点设备到根节点或中心服务器的最短路径,位于接入网的树形结构中的较高分层位置上的设备被理解为相对于位于较低分层位置上的设备处于上游。相反,沿着所述相同路径,位于较低分层位置上的设备被理解为相对于位于较高分层位置上的设备处于下游。此外,接入网可以细分为一个或多个接入分发树。接入分发树是包括一个或多个网络设备的接入网中的独立部分,这些网络设备中包括接入网的根节点处的网络设备。根节点处的网络设备可以通过从根节点向外扩展的连接进一步连接到其它网络设备。来自不同接入分发树的网络设备之间不直接连接或进行通信。

[0086] 接入分发树的网络设备可以通过单个控制器来控制,该控制器与接入分发树的根节点处的网络设备之间可以存在至少一个通信连接。在一个实施例中,该通信连接可以是连接控制器和网络设备的物理网络电缆,其中控制器和网络设备可以是两个独立的设备。在替代性实施例中,控制器可以是网络设备中的组件,通信连接可以实现为网络设备中的电路的一部分。连接到接入分发树的根节点的控制器可以是接入分发树的主控制器。主控制器是如下控制器:该控制器连接到接入分发树的根节点,并且可以通过其它控制器直接或间接控制接入分配中的所有网络设备。在接入分发树由单个主控制器控制的场景中,主控制器不必与同一接入分发树中的其它控制器协调。此外,由于来自接入网中的不同接入分发树的网络设备互不交互,因此主控制器不必与其它接入分发树的其它控制器协调。

[0087] 可替代地,可以通过附加从控制器来控制接入分发,从控制器可以可通信地连接到接入分发树的根节点下游的一个或多个节点处的一个或多个网络设备。在一个实施例中,主控制器以及一个或多个从控制器连接到接入分发树中的每个网络设备并且各自连接到不同且相互不包含的一组网络设备。从控制器与主控制器协调,并协助主控制器管理接入分发树。当接入分发树较大并且包括许多网络节点和设备时,除了主控制器之外,通常还使用从控制器。

[0088] 当主控制器发生故障时,只有与主控制器相关联的接入分发树离线。由其它控制器控制的其它接入分发树不会因为主控制器发生故障而离线。类似地,当从控制器发生故障时,只有由从控制器控制的网络设备离线。由同一接入分发树中的其它控制器控制的其它网络设备不会因为从控制器发生故障而离线。

[0089] 如果主控制器发生故障,则与其它接入分发树相关联的控制器可以是备份控制器,备份控制器可以接管对主控制器的接入分发树的控制。类似地,当从控制器发生故障

时,同一接入分发树的主控制器可以接管对先前由从控制器控制的网络设备的控制。接管过程可以快速发生,以将接入网部件的停机时间减到最短,保持接入网的高可用性。可以提供多个网络控制器之间的一致性以实现快速接管过程。控制器可以使网络设备的当前状态保持最新,因此可以在没有进一步延迟的情况下接管控制。

[0090] 管理网络中的一致性的传统方法包括在网络设备集群中达成一致的方法(例如Raft)。该集群选择一个领导设备,然后,该领导设备复制集群中每个设备上的网络操作的日志。该方法可能会导致时延较高且网络响应时间较慢。控制器在对网络中的设备进行任何变更之前达成一致。控制器之间达成一致需要至少一个往返时间(round-trip time, RTT)。此外,该方法可能导致可用性较低。一个设备集群没有足够的设备来满足在控制器之间达成一致的仲裁要求。当不满足仲裁要求时,控制器无法达成一致,因此无法执行网络操作。

[0091] SCL也可能存在时延问题。当控制器无法就网络设备的当前状态达成一致时,使用SCL方法的控制器直接轮询网络设备以获得它们的状态信息。然而,在时间和网络资源方面,轮询都是一个代价高昂的过程。

[0092] 图6示出了根据本技术实施例的示例性接入网700,突出接入网700的分层级别。接入网700是分布式系统的一个示例,还可以实现为软件定义网络(software defined network, SDN)。接入网700包括多个网络节点处的多个网络设备720,标记为1a至3g。这些网络设备720可以包括控制器、交换机、客户端设备和可以在接入网700中充当节点的其它网络设备720。虽然图6中描述了十四个网络设备720或节点,但是应理解,在替代性实施例中,接入网700可以包括任何数量的网络节点处的任何数量的网络设备720。虽然图7描述了每个节点有一个网络设备720,但是应理解,在替代性实施例中,单个网络节点处可以有多个网络设备720。

[0093] 接入网700还可以包括未在图6中明确示出的一个或多个控制器,这些控制器控制和配置接入网700中的节点处的网络设备。控制器可以是中心服务器或网络设备720的软件或硬件组件,还可以是接入网700中的节点处的独立网络设备720。控制器控制和配置网络设备720,以便将数据从源设备直接传输到接入网700中的节点处的目标目的地。控制器通过从网络设备720接收信息并基于其接收的信息发出指令来实现这一点。控制器可以执行一个或多个网络操作,例如做出数据路径决定、配置网络设备、分配网络带宽以及发送通知。根据一个实施例,网络设备720可以使用TCP连接相互通信。应理解,在替代性实施例中,可以使用其它类型的通信连接。

[0094] 如图6所示,接入网100采用分层树形结构来排列。网络设备720可以通过以树状方式从根节点1a和1b延伸的连接向外扩展。这样,网络设备720可以处于分层树形结构中不同级别的位置,其中,网络设备1a和1b处于树的第1级,设备2a至2e位于树的第2级,设备3a至3g位于树的第3级。直接连接到服务提供商,例如直接连接到中心服务器的网络设备720包括根节点1a和1b。仅直接连接到较高级别的设备720而未连接到较低级别的设备720的网络设备720处于接入网700中的叶节点。在一个示例中,根节点1a和1b可以连接到服务提供商,例如服务提供商的一个或多个中心服务器。

[0095] 在一个示例中,网络设备720之间的连接仅在接入网700的树形结构中不同分层位置处的两个网络设备720之间进行。处于同一分层级别的两个网络设备720不能相互连接。

例如,节点2a和2b不能相互连接。此外,网络设备720可以连接到处于较低分层级别位置的一个或多个网络设备720,但可能只连接到处于较高级别位置的一个网络设备720或连接到中心服务器。例如,节点2a可以连接到节点3a和3b,但可能不会同时连接到节点1a和1b。

[0096] 沿着从叶节点,例如节点3a,到根节点1a或中心服务器的任何最短路径,处于较高级别位置的网络设备720相对于处于较低级别位置的网络设备720处于上游。同样地,沿着同一类型的路径,处于较低级别位置的网络设备720相对于处于较高级别位置的网络设备720处于下游。例如,节点2a和3a都处于节点1a的下游,但节点3d并不处于节点1a的下游,因为节点1a并不沿着从节点3d到节点1a的最短路径下降。考虑到接入网700的分层树形结构,应理解,上游节点处于更中心的位置,并且处于包含从网络设备720流向服务提供商的很大一部分数据的路径中。

[0097] 接入网700可以进一步划分为一个或多个接入分发树。接入分发树是接入网700中的独立部分,接入网700包括根节点处的一个或多个网络设备以及连接到根节点控制器且处于其下游的一个或多个网络设备720。接入分发树可以由一个主控制器控制,还可以由包括一个主控制器以及一个或多个从控制器的多个控制器控制,这些控制器相互协调以控制和配置同一接入分发树中的网络设备720。

[0098] 图7示出了拥有多个网络设备的另一示例性接入网800。这些网络设备包括路由器822、交换机824和客户端设备826,例如智能手机和台式电脑。具体地,接入网800包括两个路由器822a、822b、四个网络交换机824a、824b、824c、824d,以及八个客户端设备826a至826i。接入网800还包括两个简单的接入分发树。第一个树包括路由器822a和路由器822a下游的所有网络设备。第二个树包括路由器822b和路由器822b下游的所有网络设备。

[0099] 在一个实施例中,可以为每个接入分发树分配一个或多个控制器,这些控制器控制和配置该接入分发树中的网络设备。通过以这种方式布置接入网800,第一个和第二个接入分发树可以独立运行。这是因为接入网800中的网络设备连接到中心服务器以便与服务提供商相连,而不是直接与其它接入分发树中的网络设备进行横向通信。所以,第一个接入分发树的控制器也不需要与第二个接入分发树的控制器协调来执行网络操作,反之亦然。因此,这两个树的控制器可以在接入网800中执行典型的网络操作,而不必就所有网络设备的状态达成一致。但是,可能需要保持某种程度的一致性,以便在主控制器发生故障时,备份控制器可以立即接管控制。可以将多个控制器分配给单个接入分发树,并且这些控制器可能需要相互协调以在同一接入分发树中执行网络操作。例如,如果交换机824c离线,则分配给交换机824c的从控制器可能需要将交换机824c的状态通知给主控制器,以便主控制器可以将流量从交换机824c转移到设备826h。

[0100] 图8为示出从网络设备到控制器的一系列事件通知的时间轴。参考图8,交换机824c和824b可以向控制器发送事件通知e0和e1。控制器可以接收来自两个交换机824a和824b的事件通知,并且基于对控制器所接收的信息的分析来发送更新u0 (update u0) 或更新u1 (update u1) 作为响应。根据一个实施例,e0可以指示交换机824b离线,而e1可以指示交换机824b在线。响应于e0,控制器可以通过更新u1 (update u1) 进行响应,以将网络流量从交换机824b转移出去。例如,控制器可以向交换机824c发送指令以限制其网络带宽使用率或将其设置为零。响应于e1,控制器可以通过更新u0 (update u0) 进行响应,以将网络流量向交换机824b转移。例如,控制器可以向交换机824b发送指令以将其网络带宽使用率设

置为某个预定量。

[0101] 根据所公开技术的一个实施例,网络设备使用TCP协议相互通信并与控制器进行通信。TCP协议在时序上是不可知的,因为它们不保证事件通知的顺序和时间会反映事件在接入网中发生时的实际顺序和时间。因此,一起工作的多个控制器可能需要就相同的事件顺序达成一致,以便在执行网络操作时防止冲突。

[0102] 如图8所示,虽然第一事件通知e0可能在第二事件通知e1之前已经发送,但是控制器可能在e1之后的时间收到e0。因此,控制器可能认为e0反映了最新状态更新且交换机824b离线,即使交换机824b稍后如事件e1所指示的那样上线。基于对交换机824b的状态的这种理解,即使交换机824b在线,控制器最终也会发出指令u1来将流量从交换机824b转移出去。根据一个实施例,提供了一种用于处理事件通知的技术,处理方式是使控制器获得关于网络设备状态的更多最新信息。

[0103] 在一个实施例中,接入网中的不同接入分发树的网络设备不直接相互通信。因此,可能没有必要跨不同的接入分发树来路由数据。不同接入分发树的控制器可能不需要在做出路径决定和在它们自己的接入分发树中配置网络设备之前达成一致。根据一个实施例,在发送关于接入网中设备状态的任何变更的通知以更新其它接入分发树的控制器之前,控制器直接控制和配置该控制器被分配到的接入分发树中的设备。

[0104] 图9为根据一个实施例的描述网络节点管理过程1000的流程图。在一个实施例中,过程1000由控制器作为一系列网络操作的一部分执行,以便在响应于事件而配置设备之后提供通知更新。

[0105] 在步骤1002处,接入网中的控制器接收来自网络设备的信号,该网络设备是例如另一控制器、交换机、客户端设备或任何其它网络设备。根据一个实施例,信号可以包含关于一个或多个网络设备的状态的信息,例如,设备是在线还是离线、它们的当前处理速度,或它们的当前网络流量和带宽。信号可以可替代地或另外包括关于网络事件的信息,例如,设备何时离线或上线、数据传输的开始和完成、用户请求,或在接入网中的设备处发生的任何其它事件。在一个实施例中,信号可以是发送给控制器的请求,用于设置将数据从源设备传输到目标目的地的数据路由路径。应理解,在各种实施例中,信号可以包含关于接入网的状态和操作的任何其它信息。

[0106] 在步骤1004处,控制器分析在步骤1002处接收的信号,并确定要执行的一个或多个网络操作。如上文所述,网络操作可以包括做出数据路径决定、执行数据传输、控制和配置网络设备、向其它控制器发送通知或可以执行的任何其它网络操作。参考图7,例如,控制器可以响应于关于交换机824a离线的信号而确定执行第一和第二网络操作。第一网络操作可以包括重新分配网络带宽,第二网络操作可以包括将交换机824a的状态和控制器对网络带宽进行的变更通知给其它控制器。

[0107] 在步骤1006处,基于在步骤1002处接收的信号和在步骤1004处选择的网络操作,控制器确定要联系的一个或多个网络设备。在一个实施例中,控制器可以确定其需要配置交换机,以便可以向该交换机分配更多网络带宽。此外,控制器可以确定其需要向另一接入分发树的另一控制器发送通知。

[0108] 在步骤1008处,控制器可以基于在步骤1004处选择的网络操作来确定要发送给步骤1006处确定的一个或多个网络设备的一个或多个指令。参考图7,例如,控制器可以基于

将交换机824c离线这一情况通知给控制器的信号来确定要发送两个指令。第一个指令发送给交换机824c,以减少分配给交换机824c的网络带宽。第二个指令发送给交换机824d,以增加其带宽分配。在步骤1010处,由控制器以及在步骤1006处确定的设备来执行这些指令。

[0109] 在步骤1012处,控制器向其它控制器发送关于所执行的任何操作和网络设备状态的任何变更的通知。在一个实施例中,主控制器将以下情况通知给备份控制器:交换机824c现在离线,并且网络带宽已从交换机824c重新分配到交换机824d。在一个实施例中,将时间戳作为通知的一部分发送,以指示何时发生网络事件和通知中描述的变更。通过使用时间戳,控制器可以按时间顺序维护网络事件的日志。

[0110] 通过在通知其它控制器之前发送控制和配置网络设备的指令,控制器能够更快地执行网络操作和响应网络事件。与传统方法相比,这避免了由于必须与其它控制器寻求一致或保持网络一致性而导致的执行网络操作时的任何延迟。然而,通过使用通知,仍然可以维护不同接入分发树中的控制器之间的一致性,从而允许备份控制器在主控制器发生控制器故障时立即接管主控制器的工作。在主控制器向接入网中的其它控制器发送的通知中使用时间戳允许控制器维护网络事件的日志及其时间顺序。

[0111] 在一个实施例中,来自下游设备的数据经过上游设备才能到达根节点。因此,在做数据路径决定时,可以优先考虑更上游设备的状态。因为数据将经过更上游的节点,所以其状态可以优先于下游节点的状态。

[0112] 图10为描述一个实施例中的网络节点管理过程1100的流程图。在一个实施例中,过程1100由控制器执行,作为基于上游和/或下游设备的状态(例如在线或离线)来配置网络节点的一部分。

[0113] 在步骤1102处,控制器接收标识第一和第二网络设备的状态的一个或多个信号。例如,参考图7,控制器可以从交换机824c接收以状态更新为形式的信号,该信号指示交换机824c离线,并且从交换机824b接收第二信号,第二信号指示交换机824b的状态是在线。

[0114] 在步骤1104处,控制器确定第一和第二网络设备的分层位置。例如,控制器可以确定交换机824b位于第3级,而交换机824c位于第2级。

[0115] 在步骤1106处,控制器基于第一和第二设备的分层位置对它们进行排序。例如,控制器可以将排名1分配给交换机824c,将排名2分配给交换机824b。

[0116] 在步骤1108处,控制器基于优先考虑排名较高的网络设备的状态来确定要对一个或多个网络设备执行的一个或多个网络操作。例如,由于交换机824c离线,即使交换机824b在线,控制器也可决定为交换机824b分配较少的带宽,因为交换机824c的排名比交换机824b的高。

[0117] 在步骤1110处,控制器执行所述一个或多个网络操作。如图10所示,系统可以优先考虑位于接入网中较高级别的设备的状态,以确保控制器做出的决定更可能使数据从任何给定网络设备成功传输到中心服务器。

[0118] 由于上游节点位于接入网中更中心的位置,所以在促进通过接入网进行数据传输时,它们发挥作用的频率更高。因此,优先考虑关于上游设备状态的更新可能更有益。例如,备份控制器可以更新上游设备的状态,以便在发生控制器故障时,备份控制器更能成功接管主控制器。在一个实施例中,控制器在发送关于更多下游设备的状态的通知之前将上游设备的状态通知给其它控制器。

[0119] 图11为根据一个实施例的描述基于设备位置的控制器通知过程1200的流程图。在一个示例中,过程1200由控制器执行以确定状态通知的优先级并向网络中的其它控制器发出状态通知。

[0120] 在步骤1202处,接入网中的控制器接收关于第一网络设备处的第一事件的信息。在一个实施例中,该事件可以是第一网络设备的状态变更,例如,交换机824c离线。在步骤1204处,控制器接收处于第一网络设备下游的第二网络设备的信息。在一个实施例中,第二网络设备是交换机824b,事件是交换机824b上线。

[0121] 在步骤1206处,控制器向一个或多个其它控制器发送关于第一网络设备的状态的第一通知。在一个实施例中,主控制器向备份控制器发送第一通知。由于第一网络设备,即交换机824c,已离线,所以主控制器通知备份控制器交换机824c离线。因为在同一数据路径中第一网络设备相对于第二网络设备处于上游,所以控制器选择通知第一网络设备的状态。

[0122] 在步骤1208处,控制器向一个或多个控制器发送关于第二网络设备的状态的第二通知。在一个实施例中,主控制器向备份控制器发送第二通知。由于第二网络设备,即交换机824b,已上线,所以主控制器通知备份控制器交换机824b在线。

[0123] 图12为根据一个实施例的描述接入网中的网络节点管理过程1300的流程图。在一个示例中,过程1300由控制器执行以管理接入网中的网络节点并维持一致性。在步骤1302处,接入网的控制器接收来自第一网络设备的第一信号。在一个实施例中,参考图8,第一网络设备可以是交换机824c,信号可以是来自交换机824c的指示交换机824c离线的状态通知。

[0124] 在步骤1304处,控制器接收来自第一网络设备下游的第二网络设备的第二信号。在一个实施例中,第二网络设备是交换机824b,信号是指示交换机824b在线(例如,指示交换机824b与交换机824c之间存在链路)的状态通知。

[0125] 在步骤1306处,控制器确定第一和第二网络设备的分层位置,包括第一和第二网络设备的相对位置。在一个实施例中,控制器确定交换机824c位于接入网的分层树形结构中的第2级,交换机824b位于第3级。此外,控制器确定交换机824c相对于交换机824b处于上游,比交换机824b高一级。

[0126] 在步骤1308处,控制器可以尝试识别状态信息之间的差异。然后,因为控制器已经识别出第一设备相对于第二设备处于上游,所以控制器可以通过优先考虑第一设备的状态来解决任何已识别的差异。在一个实施例中,控制器识别出状态信息与交换机824c的状态不一致。然后,控制器通过以下方式来解决这种差异:优先考虑交换机824c的状态,以确定交换机824b也离线或者由于交换机824c离线而无法到达。

[0127] 在步骤1310处,控制器基于所接收的信号确定要执行的一个或多个网络操作。具体地,在一个实施例中,控制器基于交换机824c离线这一假设确定重新分配网络带宽。在步骤1312处,控制器基于所确定的一个或多个网络操作来选择一个或多个网络设备。在一个实施例中,在确定交换机824c离线后,控制器将带宽从交换机824c重新分配到交换机824d。

[0128] 在步骤1314处,控制器向一个或多个所选网络设备发送一个或多个指令。在一个实施例中,控制器向交换机824d发送指令,指示交换机824d将收到控制器必须管理的网络带宽的增加。控制器可以向交换机824b发送指令,指示由于交换机824c离线,所以没有为交

交换机824b分配任何带宽。

[0129] 在步骤1316处,控制器确定第一和第二网络设备以及一个或多个所选网络设备的分层位置,以及它们相对于彼此的位置。在一个实施例中,控制器确定交换机824c和交换机824d都处于接入网的分层树形结构中的第二级,交换机824b处于第三级。此外,控制器确定交换机824c相对于交换机824b处于上游。

[0130] 在步骤1318处,控制器向一个或多个其它控制器发送第一通知,第一通知是关于在步骤1316处讨论的网络设备中具有最高分层位置的网络设备的状态。在步骤1320处,控制器向一个或多个其它控制器发送第二通知,第二通知是关于在步骤1316处讨论的网络设备中具有第二高的分层位置的网络设备的状态。在步骤1322处,控制器向一个或多个其它控制器发送第三通知,第三通知是关于在步骤1316处讨论的网络设备中具有第三高的分层位置的网络设备的状态。在一个实施例中,控制器可以向备份控制器发送指示交换机824c离线的第一通知、指示交换机824d的带宽分配增加的第二通知,以及指示没有为交换机824b分配带宽的第三通知。

[0131] 在一个实施例中,对于下游通信,优先考虑下游设备的状态,然后再考虑上游设备的状态。对于上游通信,优先考虑上游设备的状态,然后再考虑下游设备的状态。

[0132] 图13为描述基于数据流是在上游方向还是下游方向来进行数据路径决定的过程的流程图。在一个示例中,过程1300由控制器执行以管理接入网中的网络节点并维持一致性。

[0133] 在步骤1402处,控制器接收将数据从源网络设备传输到目标网络设备的请求。例如,参考图8,源网络设备可以是路由器822b,目标网络设备可以是客户端设备826f。

[0134] 在步骤1404处,控制器确定数据传输是在上游方向进行还是在下游方向进行。在一个实施例中,控制器基于源网络设备和目标网络设备的分层位置来实现这一点。例如,控制器可以确定路由器822b位于第1级,而客户端设备826f位于第4级。考虑到数据传输方向为从第1级到第4级,控制器确定数据传输方向为下游方向。如果控制器确定数据传输方向为上游,则过程前进到步骤1406。否则,如果控制器确定数据传输方向为下游,则过程前进到步骤1408。

[0135] 在步骤1406处,控制器基于优先考虑上游网络设备的状态来为一个或多个网络设备分配网络带宽。在一个实施例中,控制器通过以下方式来实现这一点:确定数据需要经过才能到达目标网络设备的一个或多个网络设备,以及这一个或多个网络设备的状态、目标设备的状态。例如,响应于将数据从客户端设备826f传输到路由器822b的请求,控制器确定数据在到达路由器822b之前将需要经过交换机824b和824c。然后,控制器可以确定交换机824b在线、交换机824c在线、路由器822b离线。在这种情况下,由于路由器822b处于交换机824b和824c的上游,所以控制器在决定网络带宽分配时优先考虑路由器的状态,然后再考虑这两个交换机的状态。因此,即使交换机824b和824c都在线,控制器也可为路由器822b分配较少的网络带宽,因为它是离线的。

[0136] 在步骤1408处,控制器基于优先考虑下游网络设备的状态来为一个或多个网络设备分配网络带宽。在一个实施例中,控制器通过以下方式来实现这一点:确定数据需要经过才能到达目标网络设备的一个或多个网络设备,以及这一个或多个网络设备的状态、目标设备的状态。例如,响应于将数据从路由器822b传输到客户端设备826f的请求,控制器可以

确定数据在到达客户端设备826f之前将需要经过交换机824c和824b。控制器还可以确定交换机824b和824c在线,而客户端设备826f离线。在这种情况下,由于客户端设备826f处于交换机824b和824c的下游,所以控制器在决定网络带宽分配时优先考虑客户端设备的状态,然后再考虑这两个交换机的状态。因此,即使交换机824b和824c都在线,控制器也可为路由器822b分配较少的网络带宽,因为客户端设备826f是离线的。

[0137] 图14为可以用于实施各种实施例的网络设备900的框图。特定网络设备可以利用示出的所有组件或仅组件的一个子集,且设备之间的集成程度可能不同。此外,网络设备900可以包含组件的多个实例,例如多个处理单元、处理器、存储器、发射器、接收器,等等。网络设备900可以包括配备有一个或多个输入/输出设备的处理单元901,这些输入/输出设备包括网络接口、存储接口等。处理单元901可以包括连接到总线970的中央处理器(central processing unit,CPU)910、存储器920、大容量存储设备930,以及I/O接口960。总线970可以为任何类型的若干总线架构中的一个或多个,包括存储器总线或者存储器控制器、外设总线,等等。

[0138] CPU 910可以包括任何类型的电子数据处理器。存储器920可以包括任何类型的系统存储器,例如静态随机存取存储器(static random access memory,SRAM)、动态随机存取存储器(dynamic random access memory,DRAM)、同步DRAM(synchronous DRAM,SDRAM)、只读存储器(read-only memory,ROM)或其组合,等等。在一个实施例中,存储器920可以包括在开机时使用的ROM以及在执行程序时使用的存储程序 and 数据的DRAM。在实施例中,存储器920是非瞬时性的。在一个实施例中,存储器920包括:信号分析模块920A,用于分析所接收的信号以获取关于网络设备状态、网络事件以及用户和机器发起的请求的信息;节点排序模块920B,用于确定信号所来自网络设备的分层位置,并根据网络设备之间的分层关系对这些网络设备进行排序;网络操作模块920C,用于基于从网络设备接收的信息来确定要执行哪些类型的网络操作以及由哪些网络设备执行这些操作;数据路径规划模块920D,用于确定将数据从源设备传输到目标目的地、形式为一系列网络设备的路径;以及控制器通知模块920E,用于生成并向其它控制器发送关于网络设备状态以及在网络设备上执行的任何操作的通知。

[0139] 大容量存储设备930可以包括任何类型的存储设备,其用于存储数据、程序和其它信息,并使这些数据、程序和其它信息可通过总线970访问。大容量存储设备930可以包括如下项中的一种或多种:固态硬盘、硬盘驱动器、磁盘驱动器、光盘驱动器,等等。

[0140] 处理单元901还包括一个或多个网络接口950,网络接口950可以包括以太网电缆等有线链路,和/或到接入节点或者一个或多个网络980的无线链路。网络接口950允许处理单元901通过网络980与远程单元通信。例如,网络接口950可以通过一个或多个发射器/发射天线以及一个或多个接收器/接收天线提供无线通信。在一个实施例中,处理单元901耦合到局域网或广域网以用于处理数据以及与远程设备进行通信,所述远程设备包括其它处理单元、互联网、远程存储设施等。

[0141] 应理解,本发明可以具体体现为许多不同的形式且不应被解释为仅限于本文所阐述的实施例。相反,提供这些实施例使得本主题透彻且完整,并且会向本领域技术人员充分传达本发明。实际上,本主题旨在覆盖这些实施例的替代、修改和等效物,这些替代、修改和等效物包含在所附权利要求书限定的本主题的范围和精神内。另外,在以下本主题细描述

中,阐述了许多特定细节以便提供对本主题的透彻理解。然而,所属领域的普通技术人员将清楚到,可以在没有这样具体细节的情况下实践本请求保护的主体。

[0142] 根据本发明的各种实施例,本文所述的方法可以使用执行软件程序的硬件计算机系统来实施。此外,在非限制性实施例中,实施方式可以包括分布式处理、组件/对象分布式处理和并行处理。可以构造虚拟计算机系统处理以实施本文所述的方法或功能中的一种或多种,本文所述的处理器可以用于支持虚拟处理环境。

[0143] 本文结合根据本发明实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图来描述本发明的各方面。将理解,流程图和/或框图中的每个方框以及流程图和/或框图中的方框的组合可以通过计算机程序指令来实现。这些计算机程序指令可以提供给通用计算机、专用计算机或其它可编程数据处理装置的处理器以生成机器,这样,这些通过计算机的处理器或其它可编程指令执行装置执行的指令创建用于实施流程图和/或框图的方框中指定的功能/动作的机制。

[0144] 非瞬时性计算机可读介质包括磁性存储介质、光存储介质和固态存储介质等各种类型的计算机可读介质,但具体不包括信号。应理解,软件可以安装在设备上并且随该设备一同出售。可替代地,可以获取软件并将其加载到设备中,包括通过磁盘介质或者以网络或分配系统的任何方式获取软件,例如,包括从软件开发者所有的服务器或从非软件开发者所有但为其所用的服务器获取软件。例如,软件可以存储在服务器上,以便通过互联网分发。

[0145] 本文中所用的术语仅仅是出于描述特定方面的目的,并且并不打算限制本发明。除非上下文清楚说明,否则本文所用的量词“一个”和指示代词“所述”包括其复数形式。应进一步了解,本说明书中所用的术语“包括”用于说明存在所述特征、整数、步骤、操作、元件和/或组件,但并不排除存在或添加一个或多个其它特征、整数、步骤、操作、元件、组件和/或它们的组合。

[0146] 本发明的描述仅出于说明和描述目的而提出,并非旨在详尽无遗或以任何所公开的形式限制本发明。在不偏离本发明的范围和精神的前提下,多种修改和改变对本领域技术人员而言是显而易见的。选择和描述的本发明各个方面以便更好地解释本发明的原理和实际应用,并且使本领域技术人员能够理解本发明适合预期特定用途的各种修改。

[0147] 出于本文档的目的,与所公开的技术相关联的每个过程可以连续执行并由一个或多个计算设备执行。过程中的每个步骤可以由与其它步骤中使用的相同或不同的计算设备执行,并且每个步骤不一定由单个计算设备执行。

[0148] 虽然已经以特定于结构特征和/或方法动作的语言描述了主题,但是应该理解的是,权利要求书定义的主题不必局限于上面描述的具体特征或动作。相反,上文描述的具体特征和动作被公开为实现权利要求的示例形式。

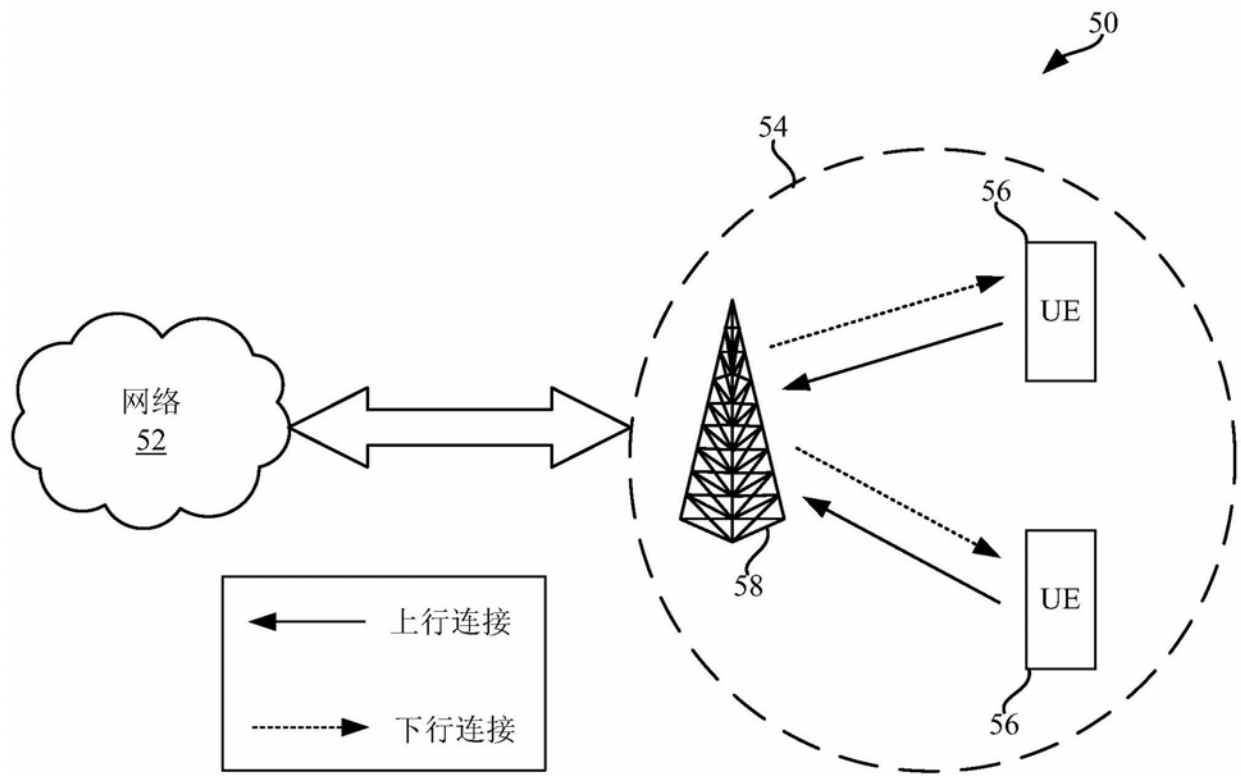


图1

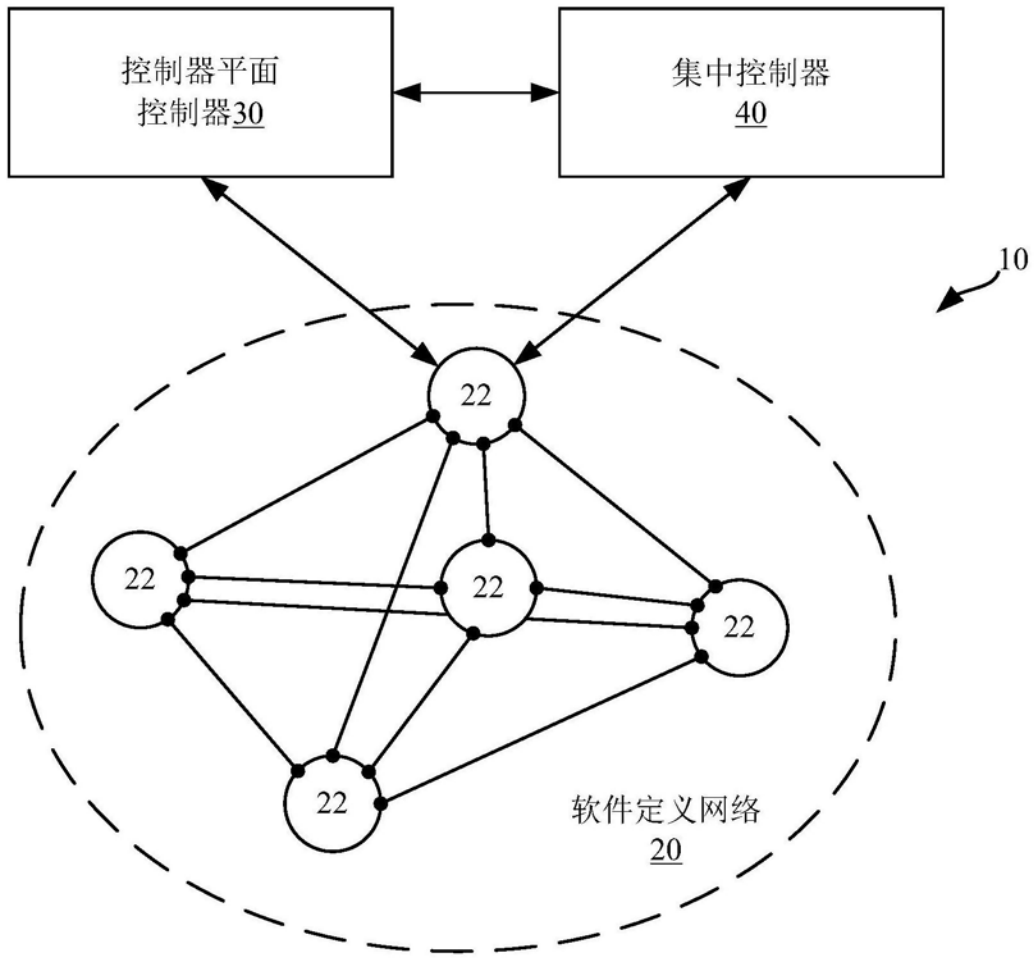


图2

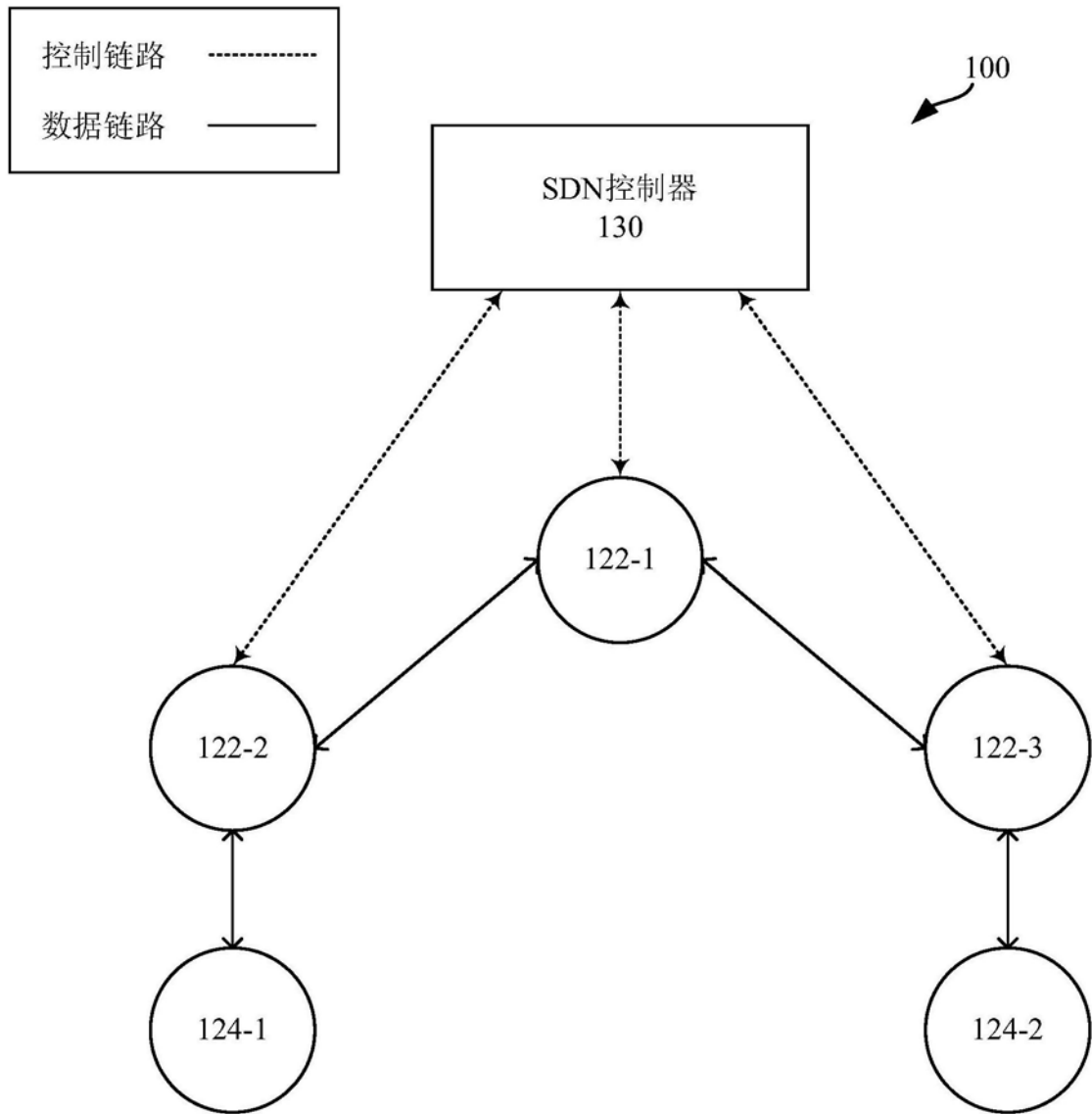


图3

300

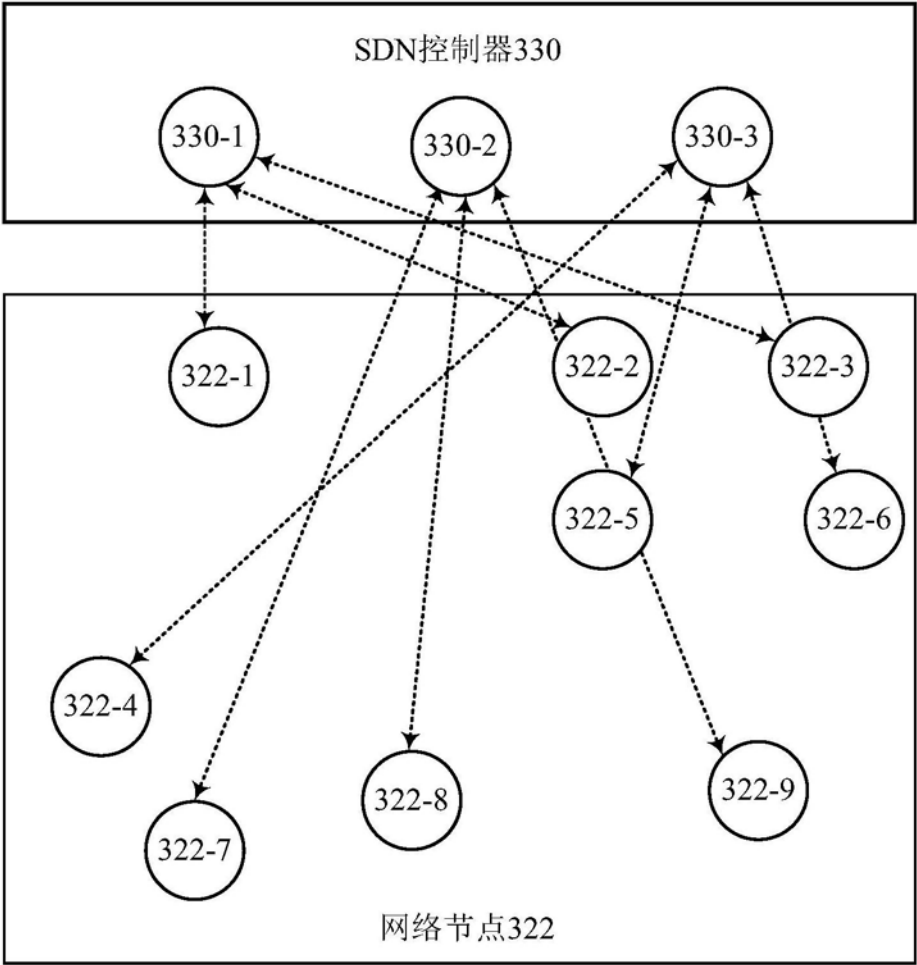


图4

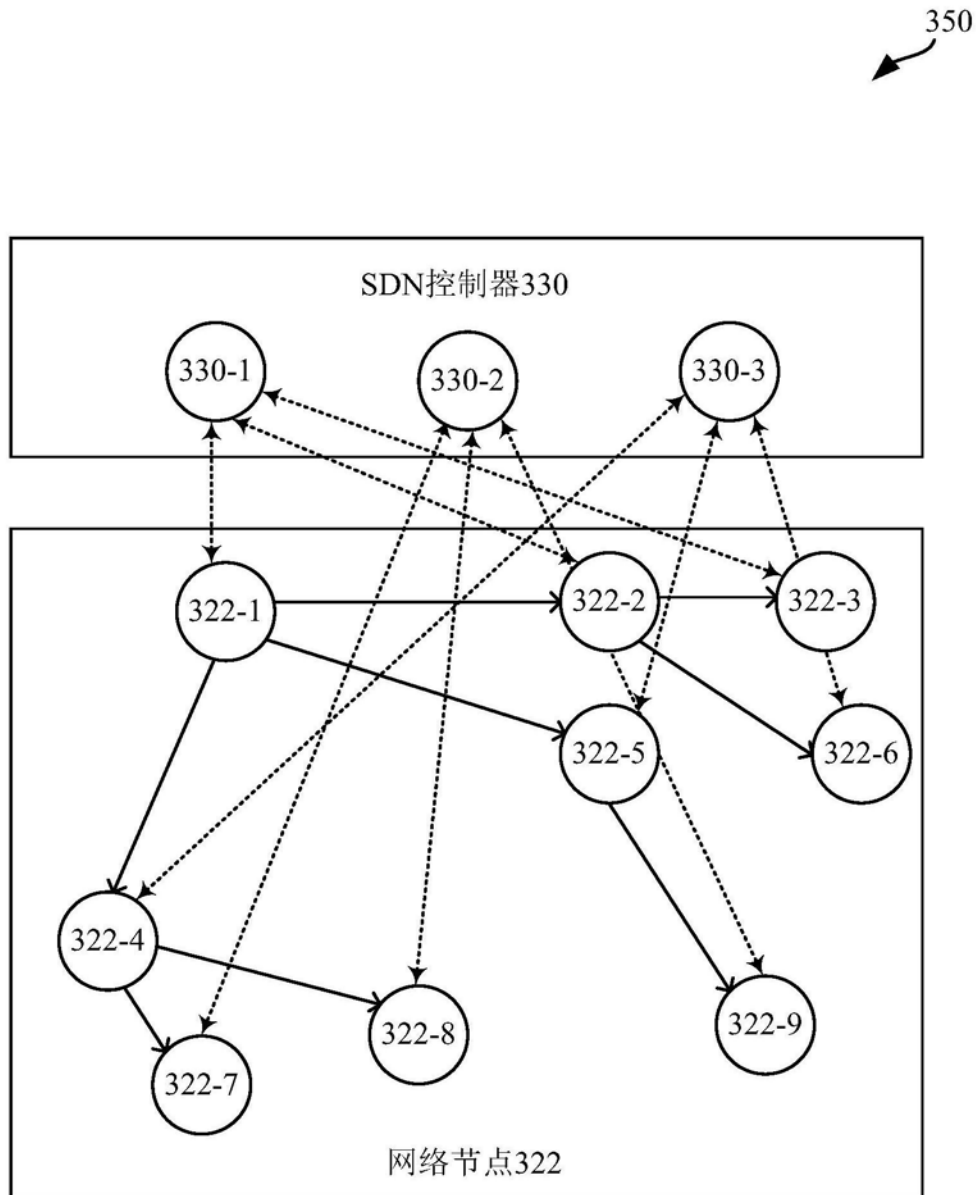


图5

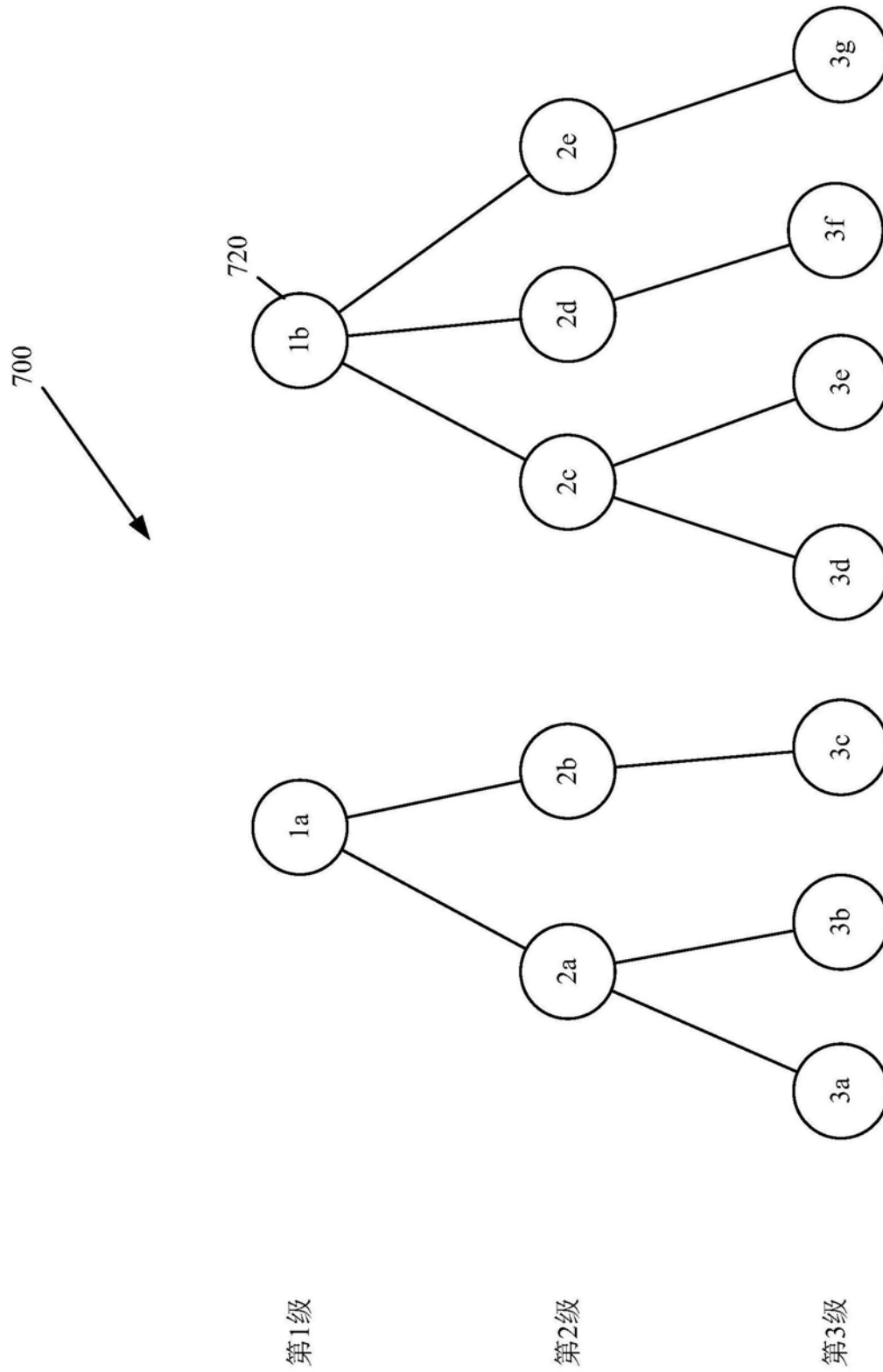


图6

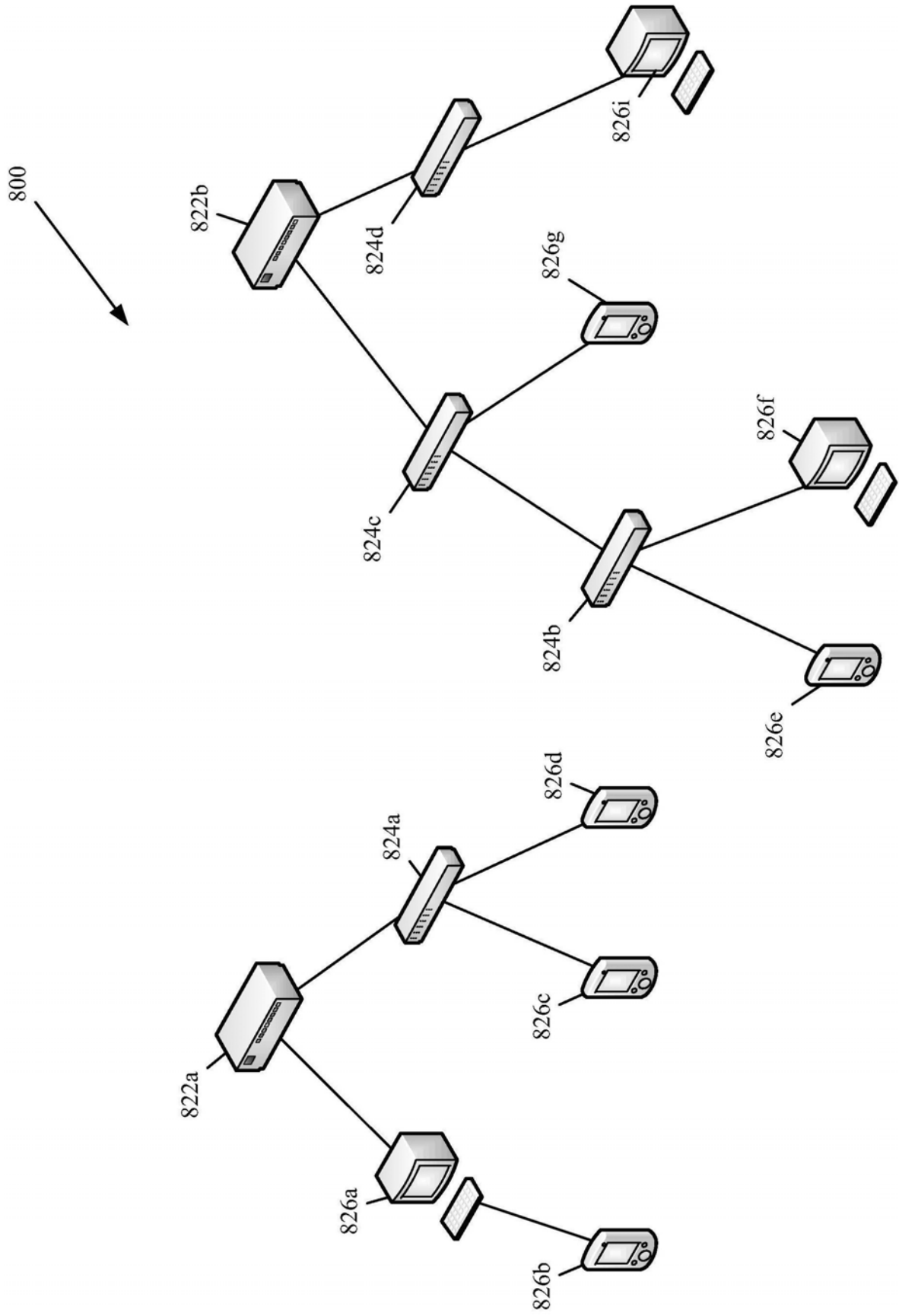


图7

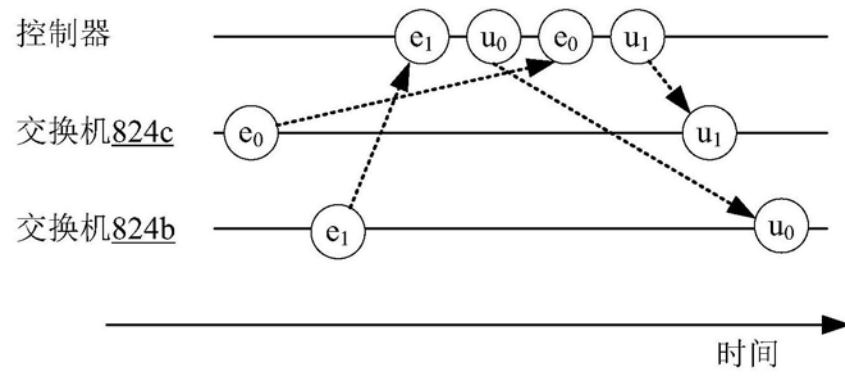


图8

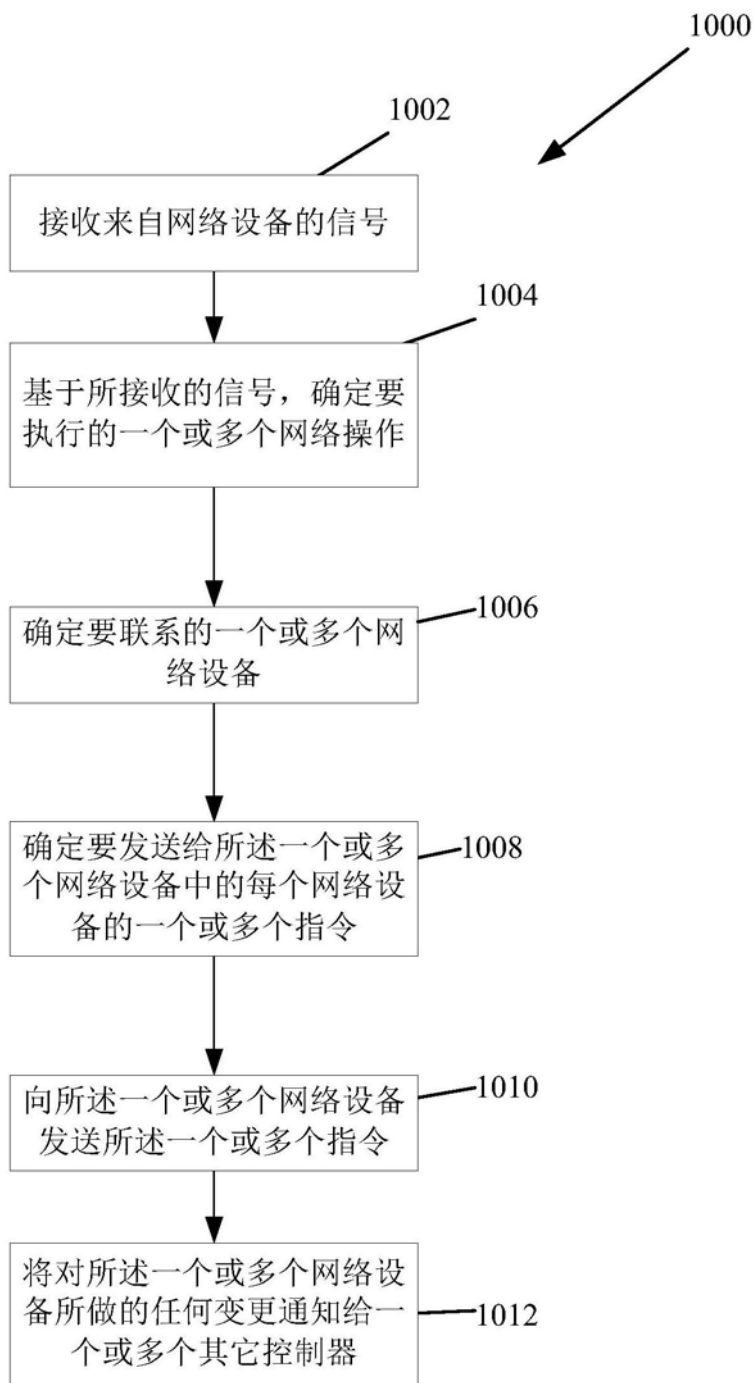


图9

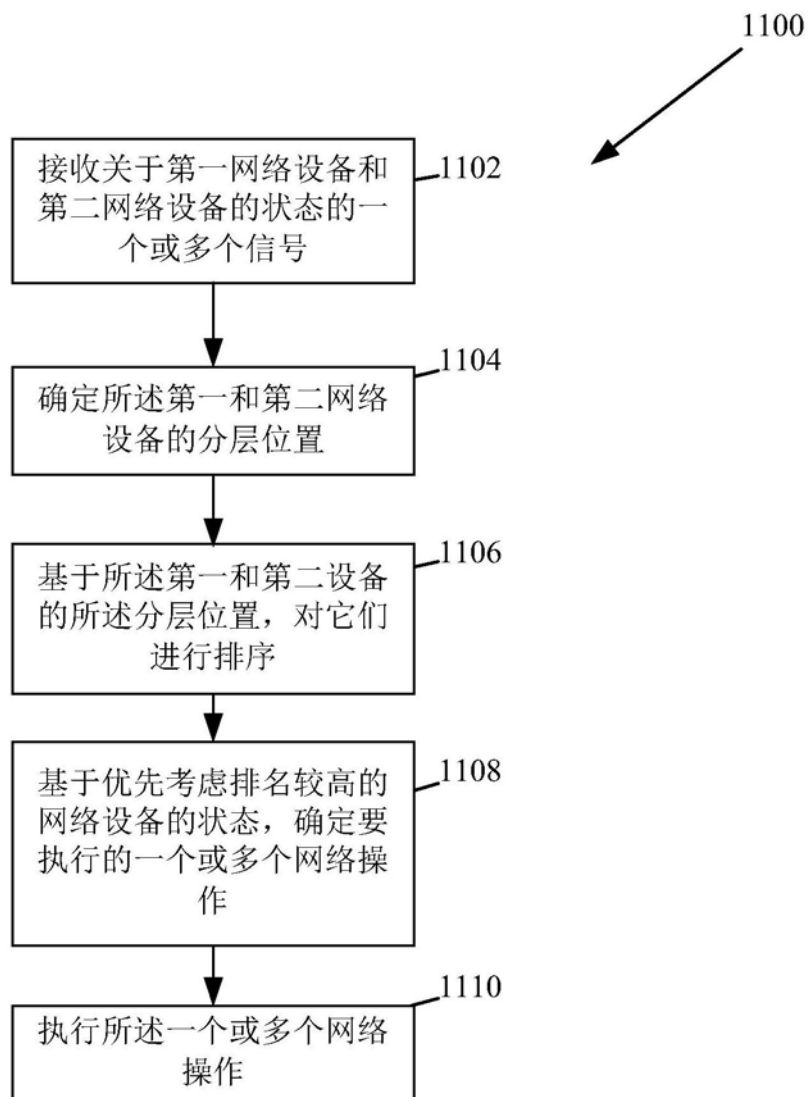


图10

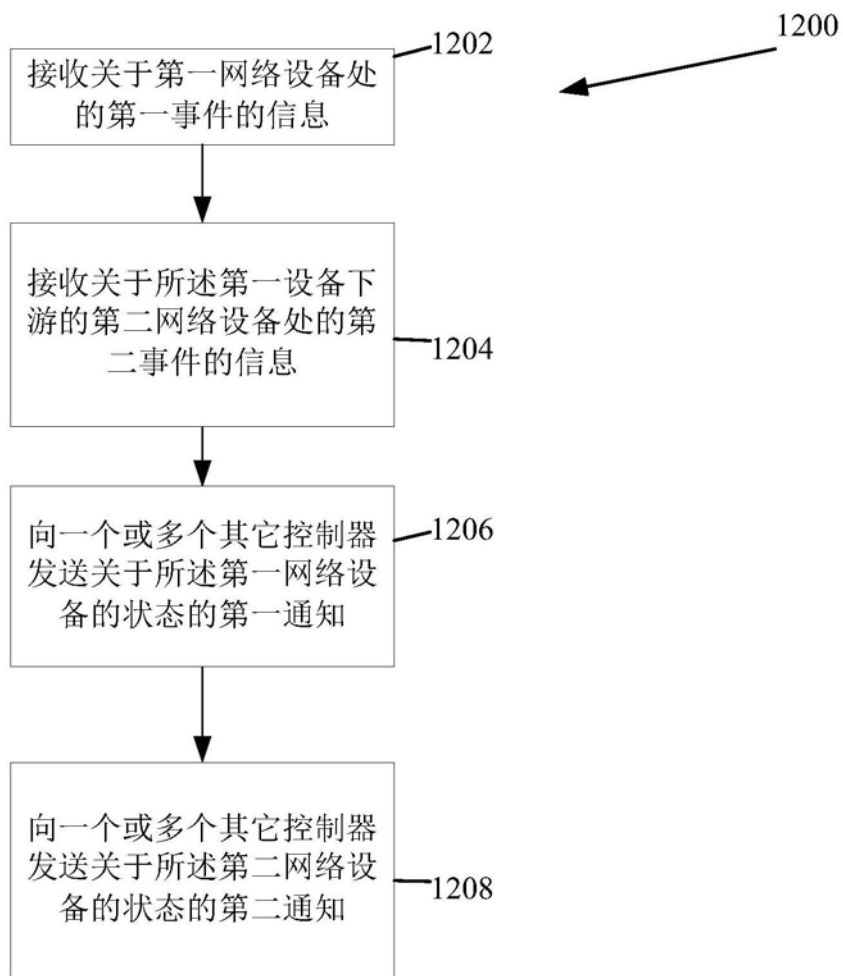


图11

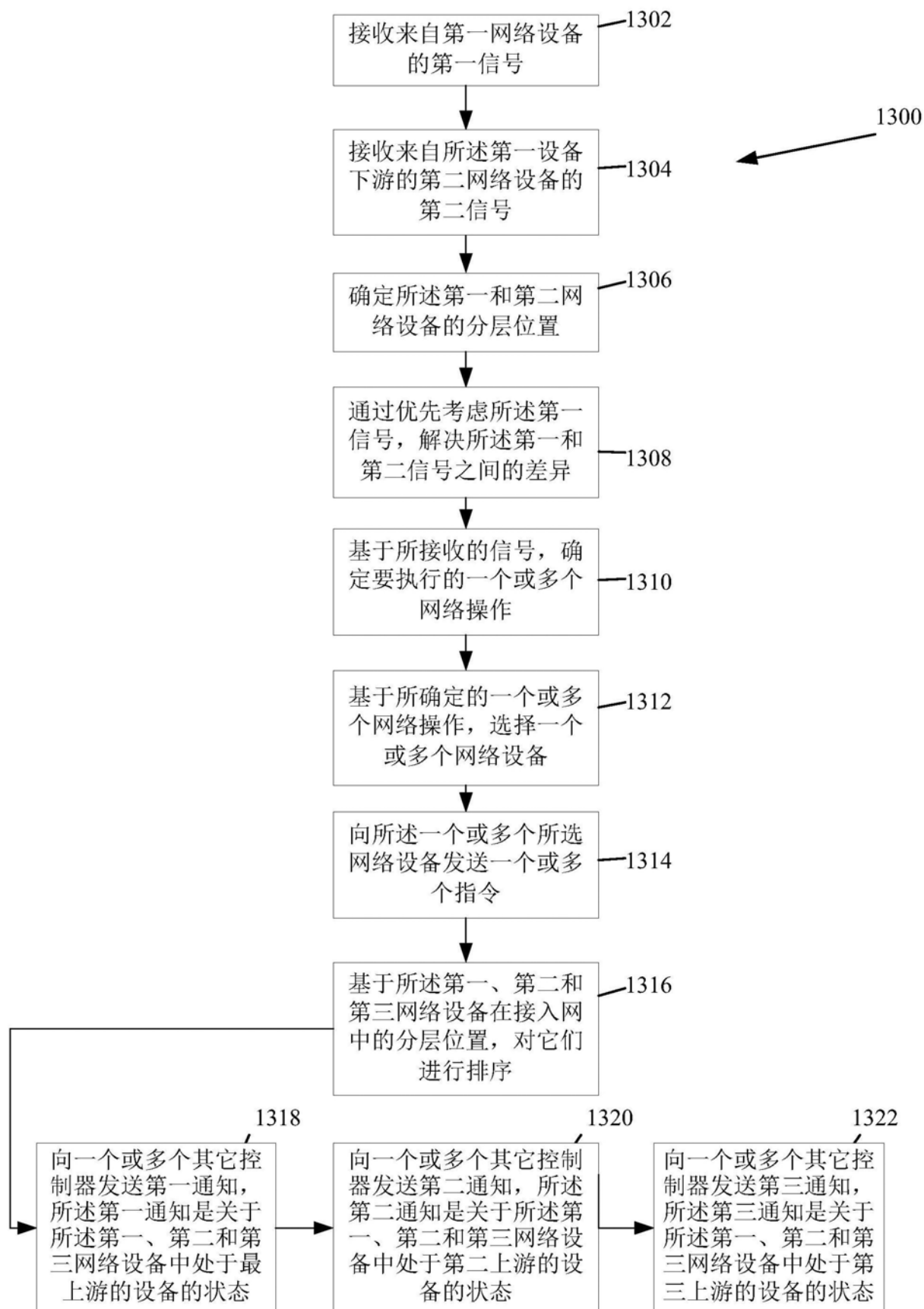


图12

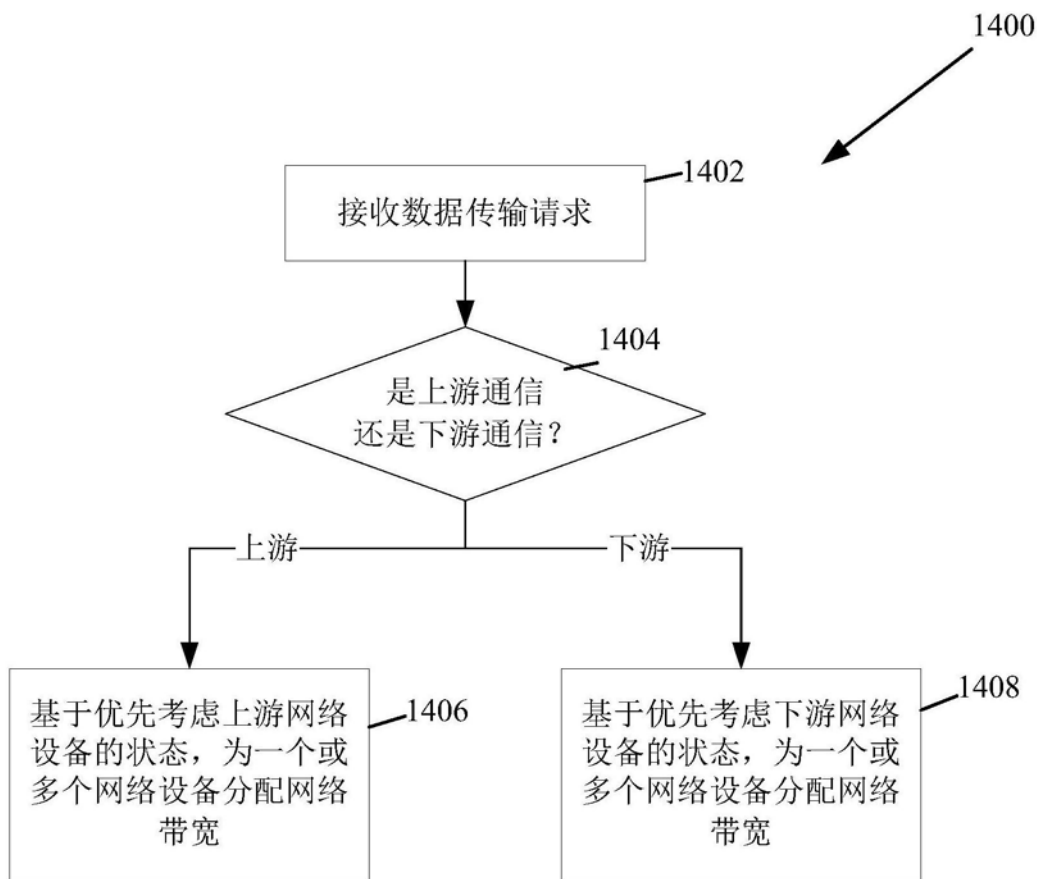


图13

900

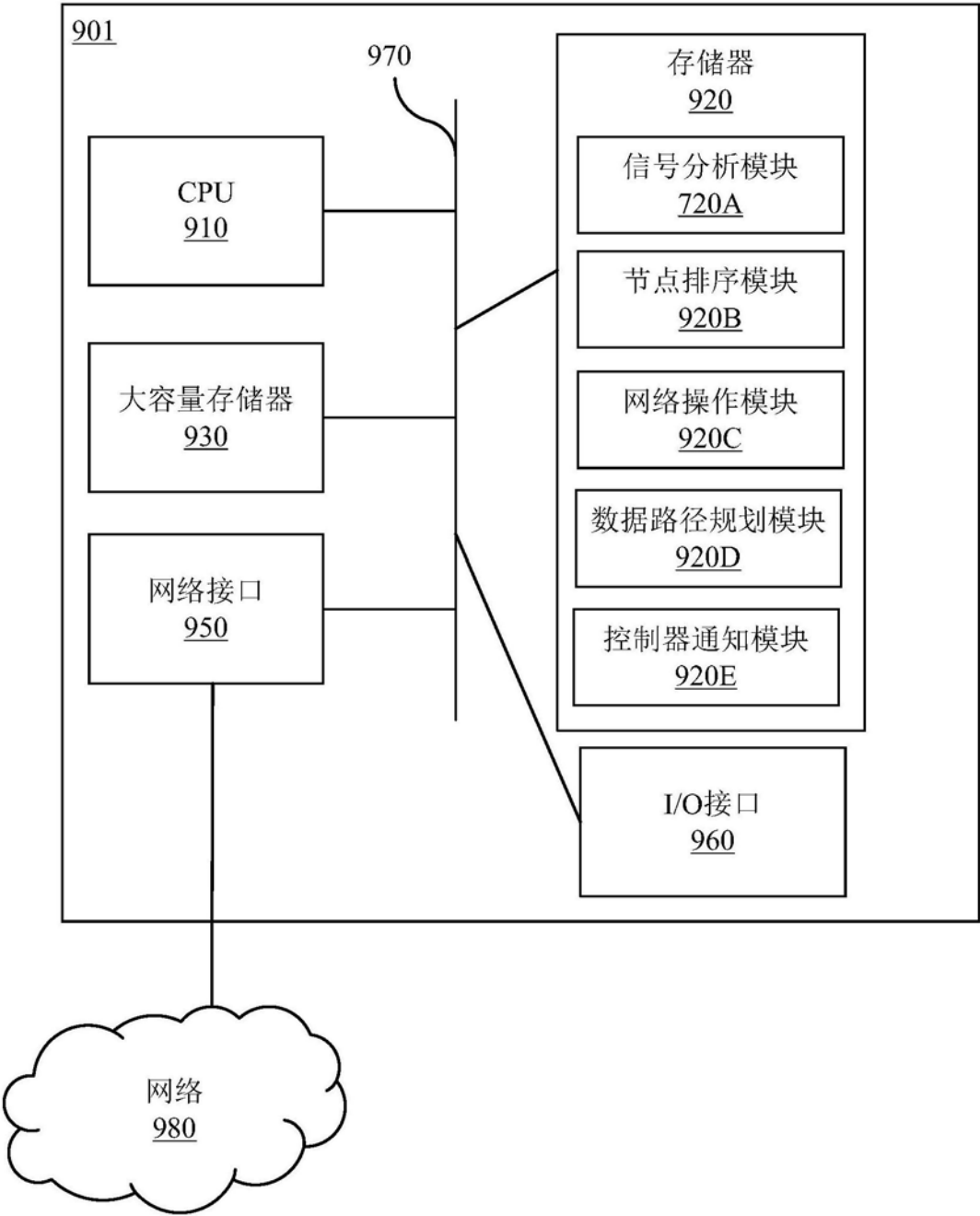


图14