(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103907321 A
(43)申请公布日 2014.07.02

(21)申请号 201280052890.X
(22)申请日 2012.11.14
(30)优先权数据
61/560,226 2011.11.15 US
13/648,961 2012.10.10 US
(85)PCT国际申请进入国家阶段日
2014.04.28
(86)PCT国际申请的申请数据
PCT/US2012/065115 2012.11.14
(87)PCT国际申请的公布数据
WO2013/074697 EN 2013.05.23
(71)申请人 甲骨文国际公司
地址 美国加利福尼亚
(72)发明人 倪威麟 B-D·约翰逊
(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 李晓芳
(51)Int.Cl.
H04L 12/729(2006.01)
H04L 12/707(2006.01)
H04L 12/753(2006.01)

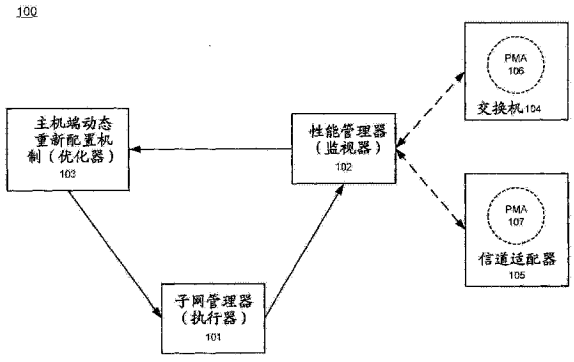
权利要求书2页 说明书10页 附图4页

(54)发明名称

在胖树拓扑中利用虚拟通道的动态分配减轻拥挤的系统和方法

(57)摘要

一种系统和方法可以在胖树拓扑中的具有多个交换机的中间件机器环境中的防止业务拥挤。子网管理器可以扫描中间件机器环境中的子网以发现变化并且维护子网全部连接。性能管理器可以检索来自于与子网中一个或多个组件相关联的一个或多个性能管理代理中的性能和错误相关的信息。然后，主机可以动态地重新配置一个或多个虚拟通道以便改善网络性能。



CN 103907321 A

1. 一种用于在一个或多个微处理器上操作的中间件机器环境中减轻业务拥挤的方法，包括：

经由子网管理器，扫描中间件机器环境中的子网以发现变化并且维持子网完全连接；

经由性能管理器从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息；以及

允许主机动态地重新配置寻址状态信息以便改善网络性能。

2. 如权利要求 1 所述的方法，还包括：

在中间件机器环境中支持胖树拓扑。

3. 如权利要求 2 所述的方法，还包括：

允许中间件机器环境中的胖树拓扑是简单的胖树拓扑。

4. 如权利要求 2 所述的方法，还包括：

允许中间件机器环境中的胖树拓扑是超额预订的胖树拓扑。

5. 如前面任一权利要求所述的方法，还包括：

使用路由算法以对于从源节点转发到目的地节点的分组保证死锁自由。

6. 如权利要求 5 所述的方法，还包括：

允许路由算法经过源节点和目的地节点的最少共同祖先节点转发该分组。

7. 如权利要求 6 所述的方法，还包括：

允许路由算法包括向上阶段和向下阶段，在向上阶段中分组被从源节点转发到最少共同祖先节点，在向下阶段中分组被从最少共同祖先节点转发到目的地。

8. 如前面任一权利要求所述的方法，还包括：

在中间件机器环境中配置一个或多个物理链路以与多个虚拟通道相关联。

9. 如前面任一权利要求所述的方法，还包括：

使用一个或多个性能计数器作为在网络操作期间识别热点和对热点的贡献者的度量。

10. 如前面任一权利要求所述的方法，还包括：

在性能扫描之后识别热点流。

11. 如权利要求 10 所述的方法，还包括：

在发现热点之后将重新指定路径消息转发到所有贡献者。

12. 如权利要求 10 或 11 所述的方法，还包括：

将热点流重新分配到分类为慢通道的虚拟通道。

13. 如权利要求 10 或 11 所述的方法，还包括：

如果牺牲者流与热点流共享向上阶段，则允许简单胖树拓扑中的热点流被分配到分类为慢通道的虚拟通道。

14. 如权利要求 10 或 11 所述的方法，还包括：

如果牺牲者流与热点流共享向上阶段和向下阶段的至少一个，则允许超额预订的胖树拓扑中的热点流被分配到分类为慢通道的虚拟通道。

15. 如权利要求 10 或 11 所述的方法，还包括：

在确定节点不再是热点之后，将所有流从慢通道到快通道引导到节点。

16. 如前面任一权利要求所述的方法，还包括：

使用拥挤控制机制允许牺牲者流避免线头阻塞。

17. 一种计算机程序，包括在由一个或多个计算机系统执行时使得所述一个或多个系统执行如任何一个前述权利要求所述的方法的程序指令。

18. 一种计算机程序产品，包括在其上存储有如权利要求 17 所述的计算机程序的机器可读存储介质。

19. 一种用于在一个或多个微处理器上操作的中间件机器环境中防止业务拥挤的系统，包括：

子网管理器，其在中间件机器环境中扫描子网以发现变化并且维持子网完全连接；

性能管理器，其从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息；以及

主机端栈，其能够动态地重新配置寻址状态信息以便改善网络性能。

20. 如权利要求 19 所述的系统，其中：

所述中间件机器环境中的子网处于胖树拓扑中。

21. 一种非瞬时机器可读存储介质，在其上存储有指令，所述指令在被执行时使得系统执行如下步骤：

经由子网管理器，扫描中间件机器环境中的子网以发现变化并且维持子网完全连接；

经由性能管理器从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息；以及

允许主机动态地重新配置一个或多个虚拟通道以便改善网络性能。

22. 一种程序，在被执行时使得系统执行以下步骤：

经由子网管理器扫描中间件机器环境中的子网以发现变化并且维持子网完全连接；

经由性能管理器从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息；以及

允许主机动态地重新配置一个或多个虚拟通道以便改善网络性能。

在胖树拓扑中利用虚拟通道的动态分配减轻拥挤的系统和方法

[0001] 版权声明：

[0002] 本专利文件的公开的一部分包含受版权保护的材料。版权所有者不反对任何人对专利文件或专利公开的传真复制，因为它出现在专利商标局专利文档或记录中，但在别的方面保留所有版权。

技术领域

[0003] 本发明一般涉及计算机系统，并且具体地涉及在中间件机器环境中防止线头阻塞和业务拥挤。

背景技术

[0004] 互连网络在下一代超极计算机、群、和数据中心中起着有益的作用。诸如无限带(IB)技术之类的高性能网络技术在高性能计算领域正代替专有的或低性能的解决方案，其中高带宽和低延迟是关键要求。例如，IB安装用于诸如Los Alamos National Laboratory的Roadrunner、Texas Advanced Computing Center的Ranger、和Forschungszcntrum Juelich的JuRoPa之类的巨型计算机中。

[0005] IB首次在2000年10月被标准化为称为未来I/O和下一代I/O的两个旧技术的合并。由于它的低延迟、高带宽、和主机端处理资源的有效利用，它在高性能计算(HPC)社区内已作为建立大型和可升级计算机群的解决方案而获得公认。事实上用于IB的系统软件是OpenFabrics Enterprise Distribution(OFED)，其由专家开发并且由OpenFabrics联盟维护。OFED是公开资源并且可用于GNU/Linux和Microsoft Windows。

发明内容

[0006] 这里描述的是可以在胖树拓扑中具有多个交换机的中间件机器环境中防止线头阻塞和业务拥挤的系统和方法。子网管理器可以扫描中间件机器环境中的子网以发现变化并且维持子网完全连接。性能管理器可以从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息。然后，主机可以动态地重新配置一个或多个虚拟通道以便改善网络性能。

附图说明

[0007] 图1示出了根据本发明的实施例的在中间件环境中的优化反馈循环的例示。

[0008] 图2示出了根据本发明的实施例的用于在中间件机器环境中减轻网络拥挤的示范性流程图。

[0009] 图3示出根据本发明的实施例的虚拟通道的动态分配以减轻胖树拓扑状中的拥挤的例示。

[0010] 图4示出根据本发明的实施例的虚拟通道的动态分配以减轻超额预订的胖树拓

扑状中的拥挤的例示。

具体实施方式

[0011] 网络业务模式的算法可预测性通过虚拟化和多核系统的引入而降低。当多个虚拟化的客户端驻留在相同的物理硬件上时，网络业务变为可能导致网络中的热点的多个业务模式的重叠。如果多个流指向单个端点，则热点出现。用于热点的公共源包括复杂的业务模式，归因于虚拟化、虚拟机图像的迁移、用于容错的检查点和恢复机制、以及存储和 I/O 业务。

[0012] 当热点存在于网络中时，为该热点指定的流可能降低不指定到该热点的其它流（称为牺牲者流）的性能。这是由于由拥挤的热点产生的线头 (HOL) 阻塞现象。

[0013] 避免此问题的一种方法是使用诸如在硬件中评估的拥挤控制 (CC) 机制之类的 CC 机制。但是，硬件中评估的阻塞控制机制可能不总是可用的，例如由于同时存在于大规模群中的旧和新装置的混合。此外，适当的 CC 参数的选择高度地依赖于拓扑并且不正确的参数可能导致性能降低。另外，由于拥挤控制机制在动态地调整发送者的注入率，所以一些振荡可能出现在流当中。

[0014] 根据本发明的实施例，系统和方法可以防止在诸如使用胖树拓扑的具有多个交换机的中间件机器环境之类的互联网络中的线头阻塞和业务拥挤。子网管理器可以扫描中间件机器环境中的子网以发现变化并且保持子网完全连接。性能管理器可以从与子网中一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息。然后，主机可以动态地重新配置一个或多个虚拟通道以便改善网络性能。

[0015] 无限带 (IB) 结构

[0016] 根据本发明的实施例，可以在无限带 (IB) 结构中防止业务拥挤，其是串行点对点技术。每个 IB 网络、或子网可以包括利用交换机和点到点链路互连的一组主机。单个子网可升级到多于一万个节点并且两个或更多个子网可以利用 IB 路由器互连。子网内的主机和交换机利用局部标识符 (LIDs) 编址，例如单个子网被限制在 48151 个单播地址。

[0017] IB 子网可以采用负责初始化和起动子网的至少一个子网管理器 (SM)，所述子网包括位于子网中的交换机、路由器和主机信道适配器 (HCA) 上的所有 IB 端口的配置。SM 的职责也包括路由表计算和部署。网络的路由目的在于获得所有源与目的地对之间的完全连接、死锁自由、和负载平衡。路由表可以在网络初始化时间被计算，并且每当拓扑变化时可以重复此处理以便更新路由表并保证最佳性能。

[0018] 在初始化时，在发现阶段 SM 开始，其中 SM 进行网络扫描以便发现所有交换机和主机。在发现阶段期间，SM 也可以发现任何其它 SM 存在并且协商谁应当是主 SM。当完成发现阶段时，SM 可以进入主阶段。在主阶段中，SM 继续进行 LID 分配、交换机配置、路由表计算和部署、以及端口配置。在这一点上，子网准备好使用。

[0019] 在子网被配置之后，SM 可以监视网络变化（例如链路停止，添加设备，或移除链路）。如果在监视进程期间检测到变化，则消息（例如，陷阱 (trap)）可以被转发到 SM 并且 SM 可以重新配置网络。重新配置处理、或繁重的扫描处理的一部分是网络的重新路由，其可以被执行以便保证所有源与目的地对之间的完全连接、死锁自由、和适当的负载平衡。

[0020] IB 网络中的 HCA 可以利用队列对 (QP) 彼此通信。在通信安装期间创建 QP，并且

提供诸如 QP 数目、HCA 端口、目的地 LID、队列大小、和传输服务之类的一组初始属性。另一方面，在通信中与 HCA 相关联的 QP 在通信结束时被毁坏。HCA 可以处理许多 QP，每个 QP 由一对队列，发送队列 (SQ) 和接收队列 (RQ) 构成。在参与通信的每个终端节点处存在一个此类对。发送队列保存工作请求以被传输到远程节点，而接收队列保存关于对从远程节点接收的数据做什么的信息。除了 QP 之外，每个 HCA 可以具有与一组发送和接收队列相关联的一个或多个完成排队 (CQ)。CQ 保存对发送和接收队列发布的对于工作请求的完成通知。

[0021] 子网管理员 (SA) 是与主 SM 相关联以存储关于子网的不同信息的子网数据库。与 SA 的通信可以帮助终端节点通过经由指定的 QP(例如 QP1) 发送一般服务管理数据报 (MAD) 来建立 QP。发送者和接收者需要诸如源 / 目的地 LID、服务级别 (SL)、MTU 等等之类的信息以建立 QP。此信息可以从 SA 提供的被称为路径记录的数据结构中检索。为了获得路径记录，终端节点可以对 SA 执行路径记录查询，例如利用 SubnAdmGet/SubnAdmGetable 操作。然后，SA 可以将请求的路径记录返回到终端节点。

[0022] SM 也负责利用呈现在每个交换机和 / 或每个 HCA 中的子网经营代理商 (SMA) 监视网络变化。SMA 利用陷阱和通告将诸如新的连接、断开连接、和端口状态变化之类的变化通信到 SM。

[0023] 陷阱是发送以提醒终端节点关于特定事件的消息。陷阱可以包含具有描述事件的细节的通告属性。不同的陷阱可以被定义用于不同的事件。为了降低陷阱的不必要的分布，IB 应用事件转发机制，其中终端节点需要明确地预订它们想要被通知的陷阱。

[0024] 用于性能管理的优化反馈循环

[0025] 图 1 示出了根据本发明的实施例的在中间件环境中的优化反馈循环的例示。如图 1 所示，在中间件环境中的优化反馈循环包括执行器 (例如子网管理器 101)、监视器 (例如性能管理器 102)、和优化器 (例如交换机 104)。

[0026] 子网管理器 101 可以周期性地扫描子网以发现变化并且维持完全连接的子网。此外，性能管理器 102 可以从子网中的每个组件中周期性地收集信息以便分析网络性能，并且主机端栈 103 可以动态地重新配置用于网络配置的寻址状态信息。

[0027] 另外，诸如交换机 104 或信道适配器 105 之类的子网中的每个设备可以执行性能管理代理 (PMA)106 或 107。每个 PMA 可以与一组性能监视和误差监视寄存器相关联。性能管理器 102 可以例如利用性能管理数据报 (MAD) 从这些寄存器中检索性能和错误相关信息。

[0028] 性能管理是由 IB 提供的一般管理服务中的一个以从 IB 组件中检索性能统计和误差信息。每个 IB 设备可以执行 PMA 和最少集合的性能监视和误差监视寄存器。此外，IB 规范也定义允许附加性能和误差计数器的监视的一组可选属性。

[0029] 通过向给定设备的 PMA 发出性能 MAD，性能管理器 (PM) 可以从这些寄存器中检索性能和错误相关的信息。PM 然后运行检索并且向 PMA 返回结果。PM 可以使用此信息检测初期故障并且基于此信息，PM 可以向 SM 建议推荐或需要的路径变化和性能优化。

[0030] 性能管理涉及性能调节，包括找到并删除瓶颈。如图 1 所示的优化反馈循环可以应用于利用虚拟通道的动态分配支持以借助于 SM、PM、和具有主机端动态再配置能力的主机栈减轻网络拥挤。在子网中，SM 周期性地扫描子网以发现变化并且维护全部连接的子网。PM 可以周期性地从子网中的每个组件收集信息以便分析网络性能。在分析之后，PM 将有关

信息转发到重新配置虚拟通道以便改善网络性能的主机栈。

[0031] 根据本发明的实施例，路由算法可以在热点存在期间使用多个虚拟通道 (VL) 以改善性能。VL 可以在路由表生成期间被静态地分配并且可以避免阻塞的负面影响，假设拓扑是平衡的、完全填充的并无故障的胖树。另外，利用虚拟通道的动态分配以减轻网络拥挤的机制可以被设计为识别热点流并且动态地分配虚拟通道。

[0032] 与 IB 拥挤控制相比较，利用虚拟通道的动态分配以减轻网络拥挤，消除了对贡献者的源扼杀的需要。此外，因为 IB CC 可以动态地调整发送者的注入率，所以 IB CC 参数可以导致所有流之间的振荡。结果，IB CC 解决方案可能不适合于更持久性质的拥挤问题，因为振荡可以降低总体网络吞吐量。当多个作业在相同的系统上运行并且竞争网络资源时、或当系统不被平衡在它上运行的应用时，此类持久的拥挤问题在业务已经离开失败的链路时出现。持久的拥挤问题可以通过以下方法被处理 ：首先检测它们，并且此后动态地重新分布 VL 资源以便获得可以是在系统启动时不可能静态地实现的平衡。

[0033] 根据本发明的实施例，SM 可以随着 PM 启动而被使用。由于 PM 周期性地查询子设备内的每个组件中的性能计数器引起的额外开销可以对数据业务具有最小影响，只要 SM 运行在专用节点上。

[0034] 根据本发明的实施例，中间件机器环境中的子网处于胖树拓扑中。优选地，中间件机器环境中的胖树拓扑是简单的胖树拓扑。优选地，中间件机器环境中的胖树拓扑是超额预订的胖树拓扑。

[0035] 根据本发明的实施例，中间件机器环境中的一个或多个交换机使用路由算法以保证用于从源节点转发到目的地节点的分组的死锁自由。

[0036] 优选地，中间件机器环境中的一个或多个交换机允许路由算法经由源节点和目的地节点的最小共同祖先节点转发分组。优选地，主机允许路由算法包括向上阶段和向下阶段，在向上阶段中，分组被从源节点转发到最小共同祖先节点，在向下阶段中，分组被从最小共同祖先节点转发到目的地。

[0037] 优选地，中间件机器环境中的一个或多个交换机配置中间件机器环境中的一个或多个物理链路以与多个虚拟通道相关联。

[0038] 优选地，性能管理器使用一个或多个性能计数器作为在网络操作期间识别热点和到热点的贡献者的度量。优选地，性能管理器可以在性能扫描之后识别热点流。在发现热点之后性能管理器可以将重新指定路径 (repath) 消息转发到所有贡献者。

[0039] 中间件机器可以将热点流再分配到分类为慢通道的虚拟通道。如果牺牲者流与热点流共享向上阶段，则中间件机器可以允许简单胖树拓扑中的热点流被分配到分类为慢通道的虚拟通道。如果牺牲者流与热点流共享向上阶段和向下阶段的至少一个，则中间件机器可以允许超额预订的胖树拓扑中的热点流被分配到分类为慢通道的虚拟通道。中间件机器可以将热点流再分配到分类为慢通道的虚拟通道。中间件机器可以在确定节点不再是热点之后将所有流引导到从慢通道到快通道的节点。中间件机器可以利用拥挤控制机制允许牺牲者流避免线头阻塞。

[0040] 图 2 示出了根据本发明的实施例的用于在中间件机器环境中减轻网络拥挤的示范性流程图。此流程图所示的处理可以由处理器执行。如图 2 所示，在步骤 201 处，子网管理器扫描中间件机器环境中的子网以发现变化并且完成子网完全连接 ；然后，在步骤 202

处，性能管理器可以从与在子网中可以是硬件的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关信息。最后，在步骤 203 处，系统允许主机动态地重新配置网络连接，诸如寻址状态信息，以便改善网络性能。

[0041] 减轻胖树拓扑中的阻塞

[0042] 根据本发明的实施例，优化反馈循环机制可以应用于任何拓扑和路由算法。在一个示例中，由于相对于免于死锁的简单性，可以使用胖树。系统可以动态地更新拥挤的连接并且将拥挤的业务流移动到结构中不同的虚拟通道，并且从而保证拥挤效果对不遭受拥挤的连接没有影响。

[0043] 图 3 示出根据本发明的实施例的虚拟通道的动态分配以减轻胖树拓扑状中的拥挤的例示。如图 3 所示，中间件机器环境 300 包括多个叶交换机，例如交换机 301 到 303，和多个节点，诸如在胖树拓扑中连接到叶交换机的服务器节点 1-6。另外，叶交换机 301-303 可以利用一个或多个物理链路 I-VI 连接到中间交换机或根交换机 310。

[0044] 根据本发明的实施例，每个物理链路可以支持一个或多个虚拟通道 (VL)。VL 是在相同物理链路上的具有单独的缓冲、流控制、和拥挤管理资源的逻辑信道。如图 3 所示，每个物理链路 I-VI 可以具有多个虚拟通道，诸如慢通道（虚拟通道 1）和快通道（虚拟通道 0），并且所有分组流可以被配置为最初在快通道上运行。

[0045] 路由算法可以保证在胖树拓扑中的死锁自由。路由算法可以包括两个阶段 :其中分组被从源转发的向上阶段，和当分组被向目的地转发时的向下阶段。这两个阶段之间的转变出现在最小共同祖先处，其是可以经由它的向下端口到达源和目的地的中间交换机或根交换机 310。

[0046] 当多个虚拟化的客户端驻留在相同的物理硬件上时，网络业务变为可能导致在网络中的热点的多个业务模式的重叠。在如图 3 所示的示例中，当来自于诸如节点 1、节点 3 和节点 6 之类的贡献者的多个流（在点线中）被运往终端节点 5 时，终端节点 5 可以变为热点。

[0047] 为热点指定的流可以降低其它流的性能。在上述示例中，可以存在从节点 2 到节点 3 的另一个流。从节点 1 到节点 5 的流和从节点 2 到节点 3 的流的向上阶段共享物理链路 I，因为物理链路 I 被指定处理从叶交换机 301 到叶交换机 302 和 303 的业务。由于线头 (HOL) 阻塞现象，从节点 2 到节点 3 的阻塞可以变为牺牲者流（虚线）。

[0048] 系统可以通过在相同物理链路上的不同虚拟通道分布两个流的两个向上阶段，例如通过将网络流分割成为慢通道和快通道业务。在发现节点 5 是热点之后，系统可以触发将例如重新指定路径陷阱的消息转发到诸如节点 1、3、和 6 之类的所有潜在贡献者。然后，系统可以引导从节点 1 到节点 5 的流以经过物理链路 I 上的虚拟通道 0，其被指定为慢通道。另外，如果新的流被引导到现有热点，即节点 5，则新的流可以被移动到慢通道。相反，如果节点 5 不再是热点，则指向节点 5 的所有流可以被移动回到分类为物理链路 I 上的快通道的虚拟通道 1。

[0049] 图 4 示出根据本发明的实施例的虚拟通道的动态分配以减轻超额预订的胖树拓扑状中的拥挤的例示。如图 4 所示，中间件机器环境 400 包括多个叶交换机，例如交换机 401 到 404，和多个节点，诸如在胖树拓扑中连接到叶交换机的服务器节点 1-12。另外，叶交换机 401-403 可以利用一个或多个物理链路 I-VI 连接到中间交换机或根交换机 410。

[0050] 在此超额预订的胖树中，用于转发分组的向下路径由若干目的地共享，而不是如图 3 所示专用于单个目的地。图 4 中超额预订的胖树是 2:1 超额预订的胖树，因为每个向下路径由两个目的地共享。

[0051] 如图 4 所示，从节点 1、5、和 10 到节点 9 的业务流可以在超额预订的胖树中产生 HOL 阻塞的负面影响。因此，热点处于节点 9 处，并且节点 1、5 和 10 是贡献者。

[0052] 可以存在两个当链路被超额预订时，牺牲者流可以遭受 HOL 阻塞的情况，一个在向上阶段并且一个在向下阶段。

[0053] 如图 4 所示，从节点 2 到节点 7 的牺牲者流与从贡献者节点 1 到节点 9 的热点流经由物理链路 I 共享从叶交换机 401 到中间 / 根交换机 410 的向上阶段。这与如图 3 所示的示例相似，其中性能下降是由于上游链路被与拥挤贡献者节点 1 共享。

[0054] 也如图 4 所示，从节点 2 到节点 11 的牺牲者流与从贡献者节点 1 到节点 9 的热点流经由物理链路 I 共享从叶交换机 401 到中间 / 根交换机 410 的向上阶段。另外，从节点 2 到节点 11 的牺牲者流与所有热点贡献者共享从中间 / 根交换机 410 到叶交换机 403 的向下阶段。在这种情况下，性能下降发生在与拥挤贡献者节点 1 共享的下流链路处，即使牺牲者流的目的地节点即节点 11 是与热点不同的节点。

[0055] 系统可以例如通过将网络流分割成为慢通道和快通道业务，通过在相同物理链路上的不同虚拟通道分布流。

[0056] dFtree 算法

[0057] 根据本发明的实施例，例如 dFtree 算法的路由算法可以被使用以在利用优化反馈循环的网络操作期间动态地执行 VL 的分配。性能管理器利用硬件端口计数器监视网络以检测拥挤并且通过将流分类为或者慢通道（对拥挤的贡献者）或者快通道（拥挤的牺牲者）来优化当前 VL 分配。然后，可以使用主机端动态再配置方法应用优化。此方法的效果是对拥挤做出贡献的所有流被迁移到单独的 VL（慢通道）以便避免对拥挤没有做出贡献的流（牺牲者流）的线头阻塞的负面影响。

[0058] 路由算法可以使用各种度量以动态地识别热点流，诸如 IB 性能计数器 :XmitWait 和 XmitData。IB 计数器 XmitWait 是当选择的端口有数据要发送，但是例如由于不足的信用或由于缺少仲裁而在整个计时单元 (tick) 期间实际上没有发送数据时的计时单元的数目。这里，计时单元是对时钟周期采样的 IB 架构硬件。此外，IB 计数器 XmitData 是在所有 VL 上发送的以双字的数据的总数。另外，路由算法可以使用诸如间隔之类的其它度量，其是每个性能扫描之间的秒的数目，以识别热点流。

[0059] 基于公式 Δ xmitWait/ 间隔，可以对于终端节点的远程交换机端口计算拥挤指示值。拥挤指示值将标准化的端口拥挤定义为每秒 XmitWaits 的数目。如果拥挤指示值超过阈值，则它指示终端节点是热点。

[0060] 具有高拥挤指示值的超额预订的终端节点或者是对拥挤的贡献者或者是牺牲者流。例如，在图 4 中在终端节点 1、5、10 处的贡献者和在终端节点 2 处的牺牲者，所有可以具有高拥挤指示值。另一方面，对于它的远程交换机端口具有高拥挤指示值的终端节点指示它是终端点热点。例如，在图 4 中连接到终端节点 9 的远程交换机端口可以具有高拥挤指示值。

[0061] 可以基于公式，例如 Δ xmitWait*4/ 间隔，为每个端口测量发送者端口带宽。此公

式源自于表示性能扫描之间传输的字节数的 XmitData 性能计数器。在公式中，XmitData 性能计数器乘以 4，因为 XmitData 计数器是以 32 位字为单位测量的。

[0062] 端口利用可以被定义为实际带宽与最大支持的链路带宽之间的比率。

[0063] 根据本发明的实施例，dFtree 实施方式包括两个算法：用于识别热点流的第一算法（例如如下面所示的算法 1）和用于再分配热点流到分类为慢通道的虚拟通道的第二算法（例如如下面所示的算法 2）。

[0064]

算法 1 检测终端点热点和它的贡献者

保证： 子网运行并且 PM 不断地扫描

1: **for** sw_{src} = 0 to sw_{max} **do**
2: **for** $port_{sw}$ = 0 to $port_{max}$ **do**
3: **if** *remote_port*($port_{sw}$) = = HCA **then**
4: **if** $congestion_{port}$ > Threshold then
5: **if** $port_{sw}$ ≠ hot-spot then
6: Mark $port_{sw}$ as $hotspot_{port}$
7: **end if**
8: Encapsulate $hotspot_{LID}$ in a repath trap
9: Encapsulate *slow lane* as $SL_{repath\ trap}$
10: **for** hca_{src} = 0 to hca_{max} **do**
11: **if** $congestion_{port}$ > Threshold **then**
12: **if** *hca* ≠ $hotspot_{LID}$ contributor **then**
13: **if** $Utilisation_{port}$ < 0.5 **then**
14: Mark *hca* as $hotspot_{LID}$ contributor

```
15:                Forward repath trap to HCA
16:              end if
17:            end if
18:          end if
19:        end for
20:      else if congestion_port < Threshold then
21.        if port_sw = = hot-spot then
22.          Clear port_sw as hotspot_port
23.          Encapsulate hotspot_LID in a unpath trap
24.          Encapsulate fast lane as SL_repath trap
25.          for hca_src = 0 to hca_max do
26.            if hca is hotspot_LID contributor then
27.              Clear hca as hotspot_LID
28.              Forward unpath trap to HCA
29.            end if
30.          end for
31.        end if
32.      end if
33.    end if
34.  end for
35: end for
```

算法 2 将 QP 重新配置到慢/快通道

保证： 主机接收重新指定路径陷阱

```
1: for QP_i = 0 to QP_max do
2:  if DLID_QP = = DLID_repath trap then
3:    Reconfigure SL_QP according to SL_repath trap
4:  end if
5: end for
```

[0065]

[0066] 算法 1 可以在性能扫描的每个迭代之后运行。算法检查终端节点的远程交换机端口是否具有超过阈值的拥挤指示值。例如，用于确定拥挤的拥挤阈值可以被设置为每秒

100000XmtWait 计时单元。基于每个端口计算 XmtWait 计数，因此即使网络大小增加，确定拥挤的阈值也是可用的。

[0067] 如果终端节点的远程交换机端口具有超过阈值的拥挤指示值，则结论是终端节点是热点并且远程交换机端口被标记为热点端口。在发现终端点热点之后，第一算法触发重新指定路径陷阱转发到所有潜在贡献者。此重新指定路径陷阱封闭拥挤的节点的 LID。

[0068] 热点流的检测取决于性能扫描的间隔。如果热点出现在刚好迭代 n 之后，则热点检测和‘慢通道’分配可以仅仅在迭代 n+1 处，即 t 秒以后执行。

[0069] 拥挤指示值和端口利用率可以用于识别潜在贡献者。超过阈值的拥挤指示值指示终端节点可以或者是热点贡献者或者是牺牲者流，而端口利用率可以用于把公平份额链路与拥挤的链路区分开。

[0070] 例如，如果节点 A 和节点 B 同时向节点 C 发送。即使节点 A 和 B 都具有超过阈值的拥挤指示，它们也向节点 C 接收链路带宽的公平份额。因此，如果拥挤指示值大于阈值并且端口利用率小于 50％，则算法将终端节点标记为用于热点的潜在贡献者并且转发重新指定路径陷阱。

[0071] 此外，如果新的流指向现有热点，则新的流可以被移动到慢通道。相反，如果终端节点不再是热点，则指向终端节点的所有流可以被返回到它的被分类为快通道的虚拟通道。

[0072] 当重新指定路径陷阱由潜在贡献者接收到时，可以执行算法 2。主机可以检索所有有效的 QP 并且将它们与重新指定路径陷阱中的 DLID 进行比较。如果在 QP 的一个中找到匹配的 DLID，则该 QP 被重新配置为使用慢通道。最初，所有 QP 利用快通道初始化。

[0073] 反过来参考图 3，dFtree 算法可以避免当拥挤的流被呈现时在 PM 检测节点 5 是热点之后的 HOL 阻塞。然后，将节点 5 封闭为热点 LID 的重新指定路径陷阱被转发到贡献者和牺牲者流的源节点。当发送者（热点贡献者或牺牲者流）接收到重新指定路径陷阱时，发送者检索所有有效的 QP 并且将目的地 LID 与重新指定路径陷阱 LID 进行比较。如果 QP 具有匹配的目的地 LID，则该 QP 可以被重新配置到慢通道。可以存在用于有关流的微小差错，因为 QP 被重新配置到慢通道。在重新配置之后，牺牲者流恢复它的吞吐量，因为 dFtree 算法在解决 HOL 阻塞的分隔的 VL（慢通道）中放置拥挤的流。

[0074] 可以使用一个或多个传统的通用或专门数字计算机、计算设备、机器或微处理器，包括一个或多个处理器存储器和 / 或根据本公开的教导编程的计算机可读存储介质，方便地实现本发明。合适的软件编码能够容易地由熟练的程序员基于本公开的教导来准备，这对软件领域的技术人员将是明显的。

[0075] 在一些实施例中，本发明包括计算机程序产品，其是上面或其中存储指令的存储介质或计算机可读介质（介质），指令可以用于编程计算机来执行本发明的处理中的任何一个。存储介质能够包括以下中的一个或多个：任何类型的盘，包括软盘、光盘、DVD、CD-ROM、微驱动器、和磁光盘、ROM、RAM、EPROM、EEPROM、DRAM、VRAM、闪速存储器件、磁或光卡、纳米系统（包括分子存储器 IC）或适合于存储指令和 / 或数据的任何类型的介质或设备，但是不局限于此。

[0076] 本发明的上面的描述是为了说明和描述的目的而提供的。它不意欲是穷举的或将本发明限制于公开的精确的形式。许多修改和变化能够对本领域的实践者是明显的。选择

并且描述实施例以便最佳地说明本发明的原理和它的实际应用，从而使得本领域技术人员能够理解用于各个实施例中的并且具有适合于预期的特定使用的各种修改的本发明。预期本发明的范围由以下权利要求书和它们的等价物定义。

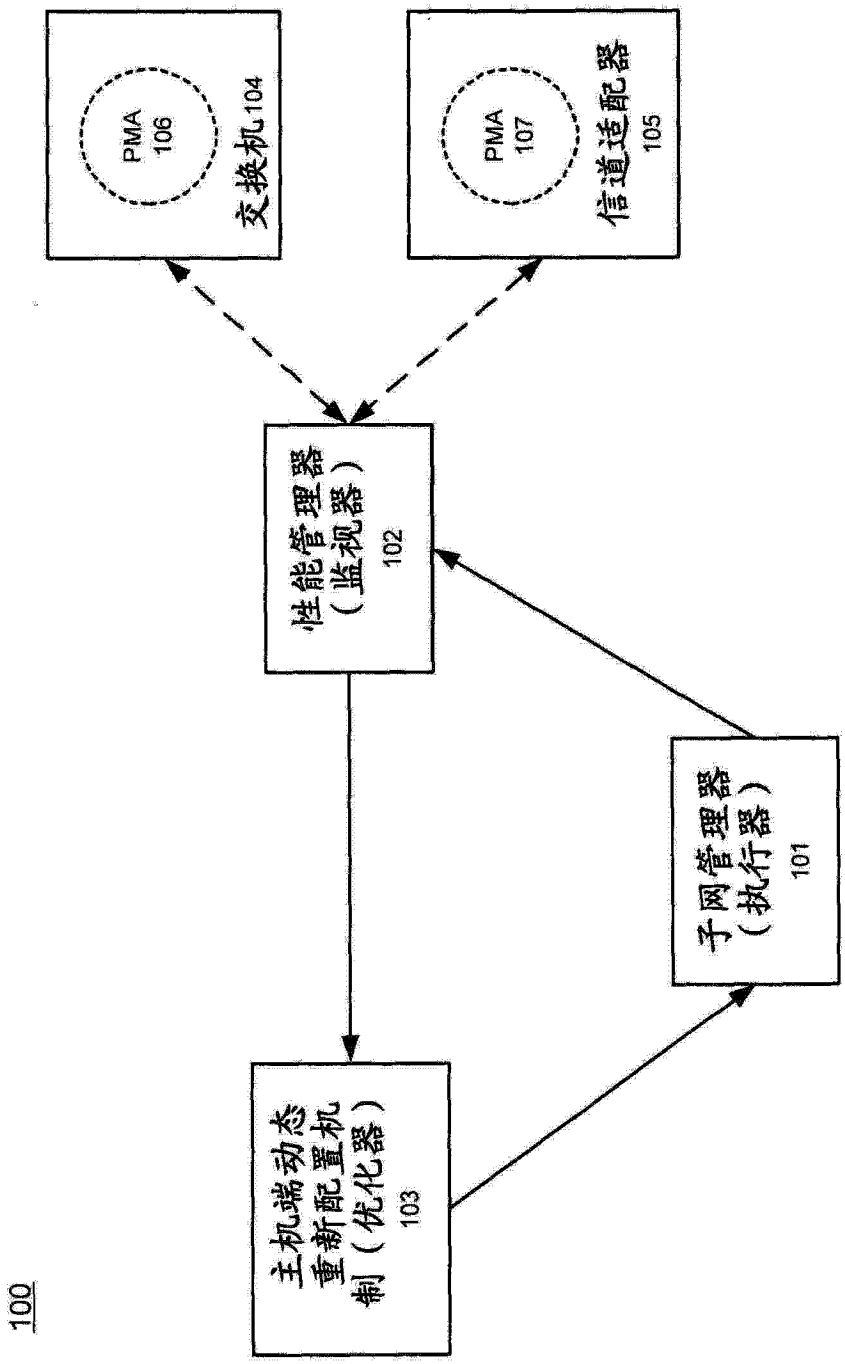

PMA
106
交换机104
PMA
107
信道适配器
105
性能管理器
（监视器）
102
子网管理器
（执行器）
101
主机端动态
重新配置机
制（优化器）
103
100

图 1

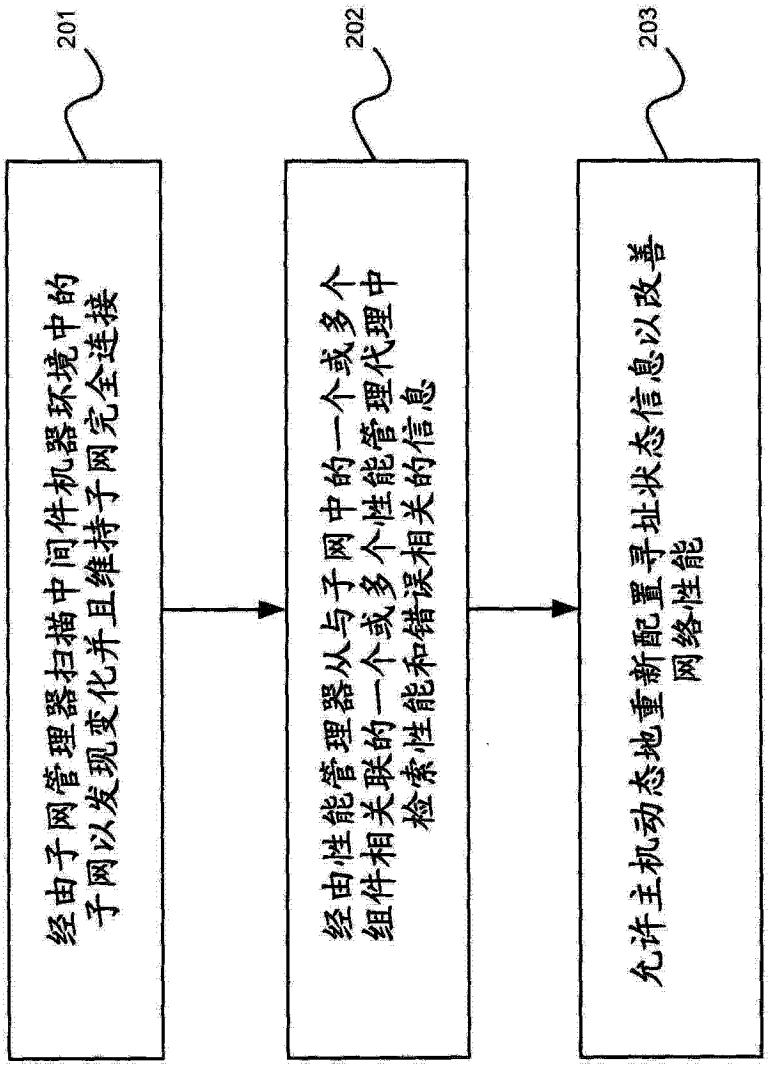

201
经由子网管理器扫描中间件机器环境中的子网以发现变化并且维持子网完全连接
202
经由性能管理器从与子网中的一个或多个组件相关联的一个或多个性能管理代理中检索性能和错误相关的信息
203
允许主机动态地重新配置寻址状态信息以改善网络性能

图 2

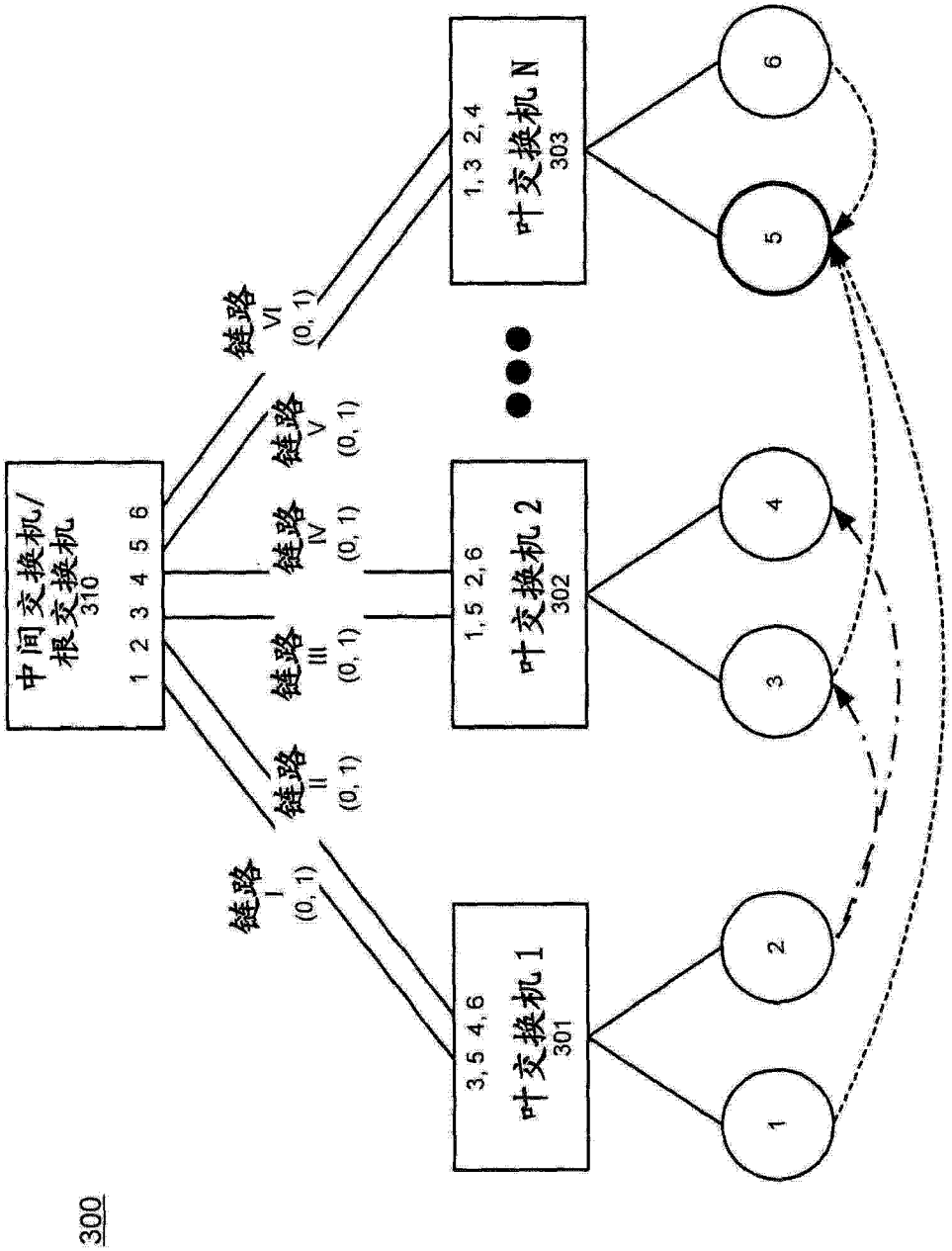

中间交换机/
根交换机
310
1 2 3 4 5 6
链路
I
(0, 1)
链路
II
(0, 1)
链路
III
(0, 1)
链路
IV
(0, 1)
链路
V
(0, 1)
链路
VI
(0, 1)
3, 5 4, 6
叶交换机 1
301
1, 5 2, 6
叶交换机 2
302
1, 3 2, 4
叶交换机 N
303
1
2
3
4
5
6
300

图 3

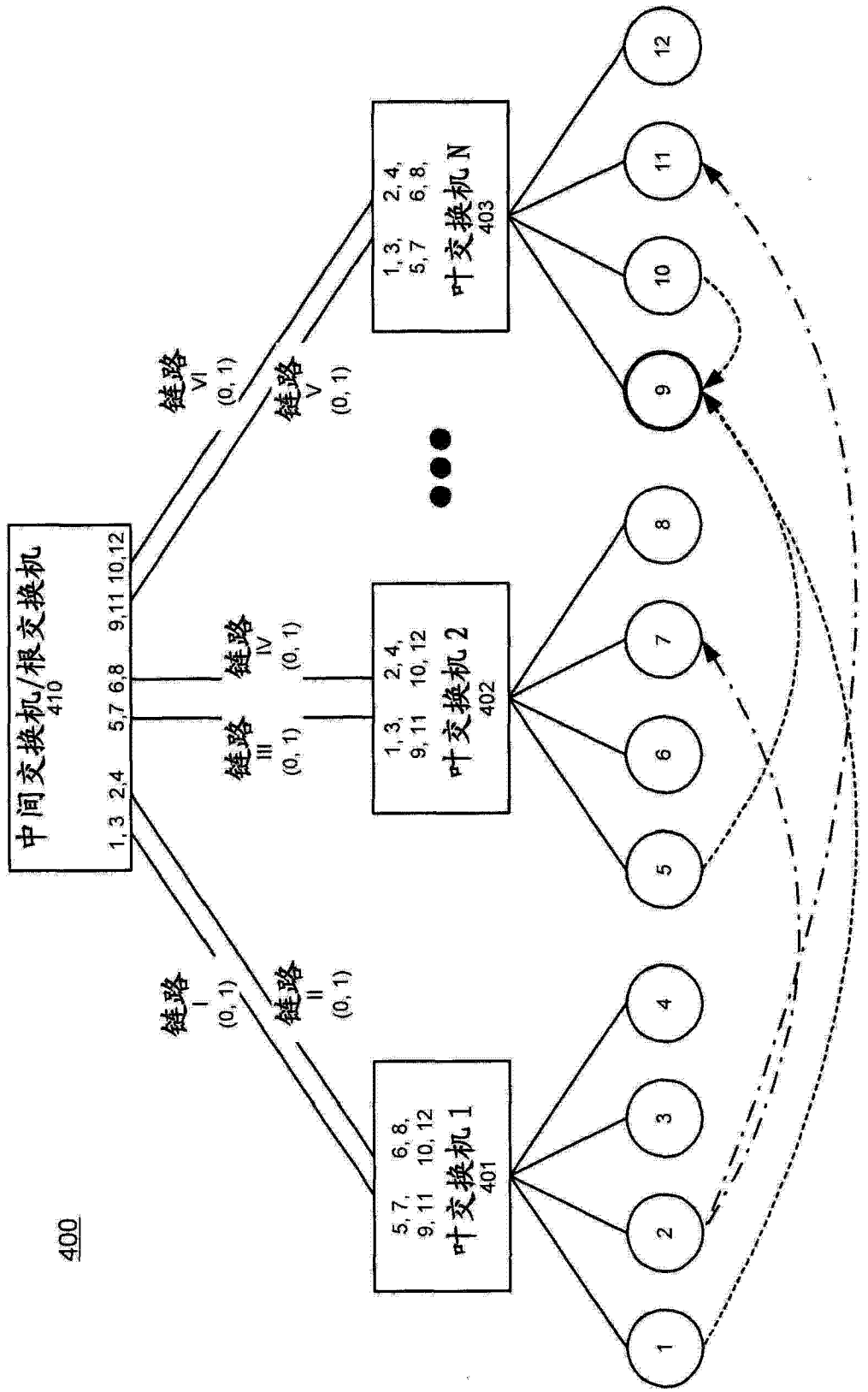

400
中间交换机/根交换机
410
1, 3 2,4 5,7 6,8 9,11 10,12
链路 I (0, 1)
链路 II (0, 1)
链路 III (0, 1)
链路 IV (0, 1)
链路 V (0, 1)
链路 VI (0, 1)
叶交换机 1
401
5, 7, 6, 8,
9, 11 10, 12
叶交换机 2
402
1, 3, 2, 4,
9, 11 10, 12
叶交换机 N
403
1, 3, 2, 4,
5, 7 6, 8,
1
2
3
4
5
6
7
8
9
10
11
12

图 4