



(12) 发明专利

(10) 授权公告号 CN 101641909 B

(45) 授权公告日 2012. 07. 18

(21) 申请号 200880009103. 7

(51) Int. Cl.

(22) 申请日 2008. 03. 19

H04L 12/28 (2006. 01)

H04B 7/15 (2006. 01)

(30) 优先权数据

11/687, 951 2007. 03. 19 US

(56) 对比文件

WO 0239620 A1, 2002. 05. 16, 全文.

(85) PCT申请进入国家阶段日

2009. 09. 21

审查员 林甦

(86) PCT申请的申请数据

PCT/US2008/057514 2008. 03. 19

(87) PCT申请的公布数据

W02008/115996 EN 2008. 09. 25

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 H·吴 K·谭 J·赵 Y·张

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 顾嘉运 钱静芳

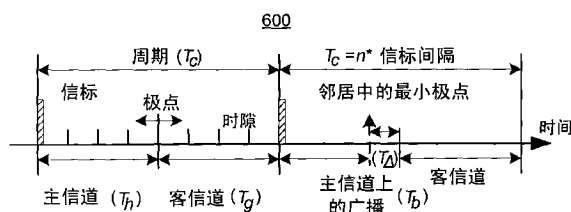
权利要求书 4 页 说明书 14 页 附图 9 页

(54) 发明名称

用于无线自组织网络的分布式重叠多信道媒体访问控制(MAC)

(57) 摘要

描述了用于无线自组织网络的分布式重叠多信道 MAC 的系统和方法。在一方面, 这些系统和方法将由无线网络定义的信道频率分成单个主信道和与该主信道正交的多个客信道。自组织网络中的每一个网络节点都在主信道上操作并持续各自可变且重叠的时间量以维护与其他各个网络节点的网络连接。另外, 每一个网络节点都确定是否及何时从主信道切换到这些客信道中的特定客信道并持续可变时间量以增加自组织网络中的一个或多个相应的通信链路上的与其他网络节点的数据吞吐量。



1. 一种计算机实现的方法,包括:

确定对应于自组织网络中的网络节点的主信道频率;

标识与所述主信道频率正交的一组客信道频率;

由所述节点中的每一个节点来在所述主信道频率上操作并持续各自可变且重叠的时间量以维护与其他节点的网络连接;以及

由所述节点中的每一个节点确定是否以及何时从所述主信道频率切换到所述客信道频率中的一特定客信道频率并持续可变时间量以便本地地优化所述自组织网路中的数据吞吐量。

2. 如权利要求 1 所述的方法,其特征在于,所述主信道频率和所述客信道频率由无线网络协议来定义。

3. 一种互连网络节点的自组织网络中的计算机实现的方法,所述方法包括:

将周期性时间周期分成两个阶段,所述周期性时间周期与无线网络协议的信标时间相关联,所述两个阶段包括连接阶段和数据吞吐量提高阶段;

在所述连接阶段期间将所述节点中的每一个节点配置成维护多个可用信道频率中的主信道频率上的连接,所述可用信道频率由所述无线网络协议来建立;

在所述连接阶段期间将所述节点的至少一个子集配置成在多个客信道频率中的一特定客信道频率上进行通信,每一个客信道频率都与所述主信道频率正交;

在所述连接阶段期间由每一个节点使用所述主信道频率来维护所述自组织网络中的连接;以及

在所述数据吞吐量提高阶段期间由所述节点的至少一个子集使用所述客信道频率中的一个或多个客信道频率来优化所述自组织网络中的数据吞吐量。

4. 如权利要求 3 所述的方法,其特征在于,所述无线网络协议基于 IEEE 802.11。

5. 如权利要求 3 所述的方法,其特征在于,所述主信道频率是所述可用信道频率中具有与所述可用信道频率中的其他信道上的噪声相比最少的噪声量的信道频率。

6. 如权利要求 3 所述的方法,其特征在于,将每一个节点配置成维护所述主信道频率上的连接还包括:

确定所述周期性时间周期期间供每一个节点在所述主信道频率上操作的相应时间量;

计算所述相应时间量的最小值,所述最小值表示所有节点都在所述主信道频率上操作的时间量,所述最小时间量是广播时间段 T_b ;

由所述自组织网络中的每一个节点来在 T_b 期间广播各自的广播消息以维护连接。

7. 如权利要求 3 所述的方法,其特征在于,与将至少所述子集配置成在所述特定客信道频率上进行通信相关联的操作是分布式的,以使得每一个节点都被配置成基于所述节点中的一跳邻居节点的数据吞吐量和连接来在所述客信道频率中的相应信道频率上进行通信。

8. 如权利要求 3 所述的方法,其特征在于,与将至少所述子集配置成在所述特定客信道频率上进行通信相关联的操作还包括:

确定用于表示所述自组织网络中的节点的数据通信模式的节点通信图;

将所述节点通信图转换成链路图,所述链路图表示由所述通信图表示的活动节点之间

的通信链路,每一个活动节点都与阈值数据通信量相关联;

从所述链路图中标识节点的最大独立集 MIS, MIS 节点中的每一个都是与另一节点没有主要冲突的节点,一节点在该节点和另一节点都需要同一节点来进行数据通信时与该另一节点有主要冲突;

将所述链路图转换成信道图,所述信道图表示所述客信道频率;以及

如果对于所述节点的 MIS 中的每一个节点存在所述节点的 MIS 中所表示的至少两个节点,则将所述节点分配给所述信道图所表示的客信道频率中的相应客信道频率。

9. 如权利要求 8 所述的方法,其特征在于,如果所述节点的 MIS 仅表示所述节点中的单个节点,则所述方法还包括移除与所述单个节点相关联的任何信道切换时间表。

10. 如权利要求 8 所述的方法,其特征在于,所述节点的 MIS 使用贪婪算法来确定。

11. 如权利要求 3 所述的方法,其特征在于,与维护所述连接和优化所述数据吞吐量相关联的操作还包括:

在所述周期性时间周期期间对于每一个节点:

如果可提高所述节点和另一节点之间的数据吞吐量,则:

(a) 计算供所述节点在所述周期性时间周期中的极点处从所述主信道频率切换到特定客信道频率的信道切换时间表;

(b) 将所述信道切换时间表传送到所述另一节点以使得所述节点和所述另一节点能够在同步时刻切换到所述客信道频率中的特定客信道频率;以及

(c) 在所述同步时刻将所述节点从所述主信道频率切换到所述特定客信道频率以便将所述节点从所述连接阶段转移到所述数据吞吐量提高阶段,所述节点被配置成在所述周期性时间周期结束时切换回所述主信道频率;并且

如果无法提高所述节点和所述另一节点之间的数据吞吐量,则移除与所述节点相关联的任何信道切换时间表以使得所述节点保持在所述主信道频率上。

12. 如权利要求 11 所述的方法,其特征在于,所述同步的时刻由所述节点根据所述节点的第一本地机器时间来调度,所述另一节点根据与所述另一节点相关联的第二本地机器时间来表示所述同步的时刻,第一和第二本地机器时间彼此独立。

13. 如权利要求 12 所述的方法,其特征在于,计算所述信道切换时间表还包括:

由所述节点来在所述主信道频率上接收信标消息,所述信标消息指示与所述无线网络协议相关联的媒体访问控制 MAC 时间;

由所述节点来确定所述第一本地机器时间与所述 MAC 时间之间的差异;

由所述节点使用所述差异来将表示所述同步的时刻的第一极点从所述节点的本地机器时间转换成基于 MAC 的极点时间;

将所述信道切换时间表中的基于 MAC 的极点时间传送到所述另一节点,所述另一节点对所述信道切换时间表的接收导致所述另一节点:

基于所述另一节点的本地机器时间来将所述基于 MAC 的极点时间转换成表示所述同步的时刻的第二极点时间;

在所述第二极点处从所述主信道频率切换到所述特定客信道频率。

14. 如权利要求 3 所述的方法,其特征在于,对于所述节点中的一节点,优化所述数据吞吐量还包括将该节点配置成保持在客信道频率上持续一时间量,所述时间量:

- (a) 根据对应的通信链路上的通信量来调整；
- (b) 在所述通信链路上的通信增加的情况下增加，并且
- (c) 与所述周期性周期时间成比例以确保不影响该节点上的任何其他通信。

15. 如权利要求 14 所述的方法，其特征在于，对于所述节点 (N_i)，总信道占用时间 (T_{Ni}) 表示与所述节点相关联的所有信道上的传输时间和接收时间，其中对于 N_i 和另一节点 (N_j) 之间的通信链路 ($l_{(i,j)}$)，总占用时间 ($T_{(i,j)}$) 表示所有信道上的信道占用时间，并且其中所述方法还包括：

根据 $T_g^{(i,j)} = w^{(i,j)} T_c$ 来确定供所述节点从所述主信道频率切换到所述客信道频率的相应的极点；并且

其中：

$$T_{Ni} = \sum T_{(i,j)}, \forall T_{(i,j)} > 0,$$

节点 N_i 上的链路 $l_{(i,j)}$ 的时间权重被表示为：

$$w_{N_i}^{(i,j)} = T_{(i,j)} / T_{Ni}, \text{以使得}$$

链路 $l_{(i,j)}$ 的时间权重被表示为：

$$w^{(i,j)} = \min(w_{N_i}^{(i,j)}, w_{N_j}^{(i,j)}); \text{并且}$$

其中 T_g 表示与所述周期性时间周期 (T_c) 成比例的、所述节点在所述客信道频率上操作的时间量。

16. 一种切换信道的方法，所述方法包括：

如下调度对于多个网络节点的信道频率切换操作：

确定自组织网络中的通信链路的最大独立集 MIS；

由跨所述 MIS 链路中的一链路的、所述节点中的一节点来请求所述节点中的另一节点在所提议的极点处调度到多个客信道频率中的特定客信道频率的信道切换，所述提议的极点被定义为周期性时间周期期间的同步的时刻，所述周期性时间周期基于无线网络协议的网络信标，每一个客信道频率都与由全部所述节点用来维护网络连接的主信道频率正交；

从所述另一节点接收确认响应；以及

响应于接收到所述确认响应，在所提议的极点处切换到所述特定客信道频率以本地地优化所述自组织网络中的数据吞吐量。

17. 如权利要求 16 所述的方法，其特征在于，所述节点的所提议的极点基于与所述节点相关联的本地机器时间，并且其中所述另一节点的所提议的极点基于与所述另一节点相关联的不同的本地机器时间。

18. 如权利要求 16 所述的方法，其特征在于，所述 MIS 是本地 MIS 并且每一个节点都确定各自的本地 MIS，并且其中所述方法还包括：

由所述节点中的每一个节点基于与这些节点中在所述节点的一跳范围内的一个或多个邻居节点相关联的通信信息来确定各自的本地 MIS，所述确定独立于与离所述节点远于一跳范围的任何其他节点相关联的通信信息。

19. 如权利要求 16 所述的方法，其特征在于，所述 MIS 是集中式 MIS，并且其中用于确定所述集中式 MIS 的操作还包括基于所述节点中的所有节点之间的通信链路来生成链路图。

20. 如权利要求 16 所述的方法,其特征在于,所述方法还包括:

由所述节点中的一节点来确定所述节点中的所有其他节点都可经由广播消息来直接联系;以及

响应于所述确定:

将所述主信道频率迁移到不同的信道频率;以及

将所述自组织网络中与所述主信道频率正交的其余信道频率分类为客信道频率。

用于无线自组织网络的分布式重叠多信道媒体访问控制 (MAC)

[0001] 背景

[0002] IEEE 802.11 是用于 WLAN(无线 LAN)的技术标准,该技术标准已经广泛用于基础结构和自组织网络两者。自组织网络中的节点通常被配置成在单个(相同的)信道频率上与该自组织网络中的其他节点进行通信。随着传输网络节点的增多,信道容量最终变成数据吞吐量瓶颈。

[0003] 图 1 示出了自组织网络 100,其包括某一数量的计算设备网络节点(“节点”),例如,节点 1 到 6。在该示例中,节点 1-6 中的每一个都被配置成使用 802.11 标准通信来在相同的单个信道频率(例如,信道 1)上与其他节点中的特定节点进行通信。然而,因为只有一个节点能够在任何特定时刻传输,所以每一节点对都最多利用信道的数据吞吐能力的 1/3。如果太多节点试图在同一信道上,在同一时刻进行通信,则数据吞吐性能快速降低。

[0004] 为了缓解同一频谱带上的相互数据传输干扰,IEEE 802.11 提供多个正交信道,从而在本质上提高了网络数据吞吐能力。例如,802.11b 定义了十一(11)个信道,其中三(3)个是正交的(非重叠的),而 802.11a 定义了十三(13)个正交信道。图 2 示出了自组织网络 200,其中节点对在不同的正交信道上进行通信以减少相互传输干扰。在该示例中,节点对(1,2)在信道 6 上操作,节点对(3,4)在信道 1 上操作,节点对(5,6)在信道 11 上操作。示出通信模式的箭头 202 到 206 中的每一个都包括用于表示跨所示节点之间的链路中的相应链路的不同通信量的不同填充(例如,交叉影线)。在这种情况下,尽管每一节点对都可同时传送数据,但在特定信道上进行通信的节点将无法与被配置成在不同信道上进行通信的任何节点进行通信。

[0005] 已开发了软件控制的信道分配和切换技术以解决所述限制中的某一些。这些常规技术允许节点在各自的信道频率上进行通信,并且同时通过该允许来维护与被配置成在不同的信道上传输的节点的连接以适应不同的通信模式。一示例性的这样的情况在图 3 中示出,其中信道 1 上的节点 3 维护到信道 6 上的节点 2 和信道 11 上的节点 6 两者的链路连接。然而,现有信道切换技术通常使用固定模式来仿真自组织网络中的所有可能的任意通信和连接模式。这一仿真实质上是有问题的,因为它假设所有节点都将遵循类似的相同行为,但并非始终如此。

[0006] 一种常规信道切换技术例如允许传输节点在每一次分组传输后切换信道。这一逐分组切换操作经常由于相应的费时信道切换等待时间而导致非常大量的处理开销。例如,802.11 中的分组传输时间大概是 300 微妙(us)。即使 NIC 能够如 100 微妙那样快地切换并且两个节点能够精确到毫秒地进行同步,这一逐分组级信道切换方案中的分组切换开销仍然过量 33%。在另一示例中,基于组件的常规信道切换技术将特定通信会话中的所有节点配置成在同一信道上传输。在该技术中,信道不在该通信会话的持续时间内改变。尽管该技术与逐分组切换方案相比在本质上减少了信道切换等待时间,但该技术在使用信道分集时是最不灵活的。

[0007] 例如,现有的基于组件的信道分配技术对于诸如图 4 所示的、表示图 2 和 3 的组合

模式的自组织网络配置将无法实现吞吐量增益。参考图 4, 示出了跨信道 1、6 和 11 中的各信道的通信模式 402 到 410。如图所示, 某些节点始终在同一信道上进行通信。例如, 该示例中的节点 3 和 4 仅在信道 1 上进行通信。其他节点跨多个信道工作。例如, 该示例中的节点 1 有时在例如信道 1 上工作以交换节点 3。该示例中的节点 1 还被配置成在信道 6 上操作以便与节点 2 进行通信, 等等。尽管现有 MAC 级信道切换方案对于图 4 的说明性自组织网络可实现吞吐量增益并维护节点连接, 但现有 MAC 级信道切换模式在这些情况下是稍微受限的。

[0008] 例如, 常规 MAC 级信道切换方案只在预定和固定时隙的边缘切换信道, 其中特定时间隙中的所有分组都在同一信道上传输。这一常规超帧级方案是受限的。例如, 现有 MAC 级信道切换方案一般要求节点始终在每一个超帧或时隙的基础上改变信道, 从而使得这些方案太不灵活以至于无法适应较长的信道切换等待时间。另外, 现有 MAC 级信道切换方案将超帧分成控制和数据时间, 这基本上可能是有问题的。例如, 以此方式划分超帧可导致: (a) 对数据时间的利用不足; (b) 由于每一个超帧一个服务机会而导致的每跳减少响应时间的大量数据吞吐量延迟以及 (c) 控制信道时间瓶颈, 因为通常要求每一个节点在每一个超帧的基础上发送信令消息以竞争访问。此外, 这些 MAC 级方案与传统的基于 802.11 的系统不兼容, 从而或者需要修改 802.11 MAC (例如, 扩展省电模式) 或者使用专用 MAC。

[0009] 概述

[0010] 描述了用于无线自组织网络的分布式重叠多信道 MAC 的系统和方法。在一方面, 这些系统和方法将由无线网络协议定义的信道频率分成单个主信道和与该主信道正交的多个客信道。自组织网络中的每一个网络节点都在主信道上操作并持续各自的可变且重叠的时间量, 以维护与其他各个网络节点的网络连接。另外, 每一个网络节点都确定是否及何时从主信道切换到这些客信道中的特定客信道并持续可变时间量以增加自组织网络中的一个或多个相应的通信链路上的与其他网络节点的数据吞吐量。

[0011] 提供本概述是为了用简化的形式介绍将在以下详细描述中进一步描述的一些概念。该概述不旨在标识所要求保护的主题的关键特征或必要特征, 也不旨在用于帮助确定所要求保护的主题的范围。

[0012] 附图简述

[0013] 在附图中, 组件参考标号最左边的数字标识了该组件首次出现的特定附图。

[0014] 图 1 示出包括计算设备网络节点的自组织网络。

[0015] 图 2 示出节点对在不同的正交信道上进行通信以减少相互传输干扰的自组织网络。

[0016] 图 3 示出其中节点在各自的信道频率上进行通信同时维护到被配置成在不同信道上传输的节点之间的连接的自组织网络。

[0017] 图 4 示出其中某些节点始终在同一信道上进行通信, 其他节点跨多个信道进行通信, 并且维护在不同信道上操作的节点之间的连接 (本质上是图 2 和 3 所示的特征的组合) 的自组织网络。

[0018] 图 5 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的示例性系统。

[0019] 图 6 示出根据一个实施例的由信道切换调度器逻辑利用的周期性时间周期 T_c 的

示例性说明。

[0020] 图 7 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的一组示例性节点、链路和信道图。

[0021] 图 8 示出根据一个实施例的转换成示出特定网络节点处的数据吞吐量瓶颈的链路图的示例性节点图。

[0022] 图 9 示出根据一个实施例的示例性信道切换调度协商协议实现。

[0023] 图 10 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的示例性分组缓冲和调度节点体系结构。

[0024] 图 11 示出了根据一个实施例的实现用于自组织网络的分布式 OMMAC 的网络节点的示例性实现。

[0025] 图 12 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的示例性过程。

[0026] 图 13 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的另一示例性过程。

[0027] 详细描述

[0028] 概览

[0029] 描述了用于自组织网络的分布式重叠多信道 MAC (OMMAC) 的系统和方法。分布式 OMMAC 提供用于维护节点到节点连接并优化自组织网络中的数据吞吐量 (即, 性能) 的框架。为此, 这些系统和方法标识各网络节点 (“节点”) 可周期性地在其上同步的信道频率以维护节点到节点的连接。该同步信道是 “主信道”, 其对于自组织网络中的所有节点都是相同的。在该实现中, 主信道表示节点当在最初配置 (即, 建立) 的自组织网络中时在其上首次同步的信道频率。这些系统和方法然后标识与主信道正交的多个信道频率。这些正交信道是 “客信道”。这些节点基于主信道和客信道来共同实现分布式且时间同步的信道调度和切换算法。具体而言, 所有节点都在主信道上操作并持续可变且重叠的时间量以便周期性地向这些节点中的其他节点广播通信和信道信息并由此维护网络连接。另外, 自组织网络中的每一个节点都独立确定是否及何时从主信道切换到特定客信道并持续可变时间量以便本地地优化该自组织网络中的数据吞吐量。

[0030] 用于自组织网络的分布式 OMMAC 的系统和方法的这些和其它方面现将更详细地描述。

[0031] 示例性系统

[0032] 图 5 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC (OMMAC) 的示例性系统 500。在该实现中, 系统 500 包括多个联网的计算设备 502-1 到 502-N (“节点”), 这些计算设备通过自组织网络 504 彼此无线耦合。每一个节点 502 都是相应的计算设备, 诸如通用计算设备 (例如, 个人计算机)、服务器、诸如膝上型计算机等便携式计算设备、诸如个人数字助理 (PDA) 等小型计算设备等。节点 502 直接通过网络 504 来彼此通信而不使用接入点。OMMAC 是分布式的, 因为每一个节点 502-1 到 502-N (即, “节点 502”) 实现 OMMAC 的本地版本以维护节点到节点网络连接并根据协议的底层无线调制技术所指定的精确同步要求来强制实施信道切换策略。在该实现中, 协议是 IEEE 802.11, 但参考这些系统和方法描述的概念适用于各种无线网络协议。

[0033] 每一个节点 502 都包括耦合到系统存储器的一个或多个处理器,该系统存储器包括可由这些处理器中的相应处理器来执行以实现用于自组织网络的分布式 OMMAC 的计算机程序模块。这一系统存储器还包括由计算机程序中的相应计算机程序在程序模块执行期间生成和 / 或使用的程序数据。例如,节点 502-1 包括耦合到系统存储器 508 的一个或多个处理器 506,该系统存储器 508 表示易失性随机存取存储器 (RAM) 和非易失性只读存储器 (ROM)。系统存储器 508 包括包含由处理器 506 来执行的计算机程序指令的程序模块 510。系统存储器 508 还包括由计算机程序指令中的相应计算机程序指令在程序模块执行期间生成和 / 或使用的程序数据 512。在该实现中,例如,程序模块 508 包括重叠多信道 MAC (OMMAC) 模块 516、网络接口控制 (NIC) 驱动程序 518 以及其他程序模块 520,诸如用于提供运行时环境的操作系统、用于分别向 / 从 OMMAC 模块 516 路由 / 接收信息的分组路由模块等。

[0034] 示例性框架

[0035] OMMAC 516 维护网络 504 中的节点到节点 502 连接并根据协议的底层无线调制技术所指定的精确同步要求来强制实施分布式信道切换策略。出于示出一个示例性实施例的目的,OMMAC 516 的操作参考 IEEE 802.11 的示例性协议来描述。为了维护节点到节点 502 连接并根据底层同步要求来强制实施分布式信道切换策略,OMMAC 516 将可用通信信道频率分成单个主信道和多个客信道。该单个主信道和多个客信道在图 5 中被共同表示为“信道集”521。每一个节点 502 都使用主信道来维护与其他节点 502 的网络连接。客信道由节点 502 中的相应节点 (根据以下所描述的信道切换和调度算法) 选择性地使用以增加网络 504 数据吞吐量 (性能)。

[0036] 1. 主信道

[0037] 在该实现中,主信道如下确定。发起者节点 502 扫描所有可用信道频率并将具有最少噪声的信道频率选为主信道。所有节点 502 都可通过节点 502 在主信道上经由广播消息来到达。在该实现中,每一个节点 502 都基于对主信道上的噪声的本地观察经由广播消息 (ACK/NACK 消息) 来对或针对该主信道进行投票。例如,在一个实现中,如果在超时模式 (对于某一可配置数量的时间周期) 中所接收到广播 ACK 消息的数量大于的某一百分比 (例如,50% 等) 的站,则节点 502 确定并投票表决新的主信道。在该实现中,主信道并不限于固定信道频率并且可由节点 502 根据通信链路状态来动态调整 (迁移)。在一不同的实现中,使用除了信道噪声之外的准则来选择主信道。

[0038] 2. 客信道

[0039] 与主信道正交的信道频率是客信道。节点 502 在客信道上与不同的节点 502 进行通信以增加网络 504 的数据吞吐量 (即,性能)。在该实现中,OMMAC 516 根据在该信道上所观察到的干扰来对客信道进行分类。对特定节点 502 的这一干扰是由该特定节点 502 的广播范围内的所有其他节点 502 产生的平均观察。(许多 OMMAC 516 操作都基于来自其他节点的 OMMAC 516 信息来以节点级执行,并由此产生术语“分布式 OMMAC”)。在该实现中,每一个节点 502 都在节点恭敬地 (respectfully) 在其上操作主信道和客信道上检测到各自的干扰电平。在这种情况下,节点 502 用周期性的广播消息来将这一信息传送给其广播范围内的其他节点 502。在该实现中,节点的本地 OMMAC 516 实现信道切换和调度操作以使得首先将具有最少干扰的客信道用于节点到节点通信。

[0040] 3. 节点极点

[0041] 节点 502 停留在主信道上的时间量是：(a) 可变的；(b) 与覆盖特定网络信标的相邻节点 502 也在该主信道上的时间部分重叠的；以及 (c) 取决于通过节点 502 和该节点 502 的广播范围内的其他节点 502 之间的通信链路的双向通信的。当节点 502 不在主信道上时，该节点 502 在客信道上。鉴于上述内容，每一个节点 502 都由其本地 OMMAC 516 来分别配置成确定该节点是否以及何时应从主信道切换到所调度的客信道以提高数据吞吐量。每一个节点的本地 OMMAC 516 为该特定节点 502 定义信道切换点（即，“极点”）。节点的极点相对于基于所利用的无线协议（例如，IEEE 802.11 等）的特定信标间隔的周期性时间周期（“ T_c ”）来定义。出于示例性说明的目的，节点的极点被示为“极点”523。

[0042] 图 6 示出了根据一个实施例的由每一个节点的信道切换调度器逻辑（该逻辑由 OMMAC 516 来实现）利用的周期性时间周期 T_c 的示例性说明。出于示例性说明和描述的目的，图 6 的各方面参考图 5 的组件来描述。在该描述中，附图标记最左边的数字指示其中首次引入所引用的组件的特定附图。参考图 6，其示出 T_c 表示每一个节点 502 被配置成在主信道和客信道的组合上操作的集体时间量。具体而言，每一个节点 502 都被本地地（即，由 OMMAC 516）配置成在单个主信道上操作并持续特定且可变的时间量（“ T_h ”）。每一个节点 502 都还被本地地配置成在多个客信道中的特定客信道上操作并持续特定且可变的时间量（“ T_g ”）。由此，

$$[0043] \quad T_c = T_h + T_g \quad (1)$$

[0044] 在该实现中，周期长度通过将整数 n 乘以与底层无线协议相关联的信标间隔来确定。例如，被配置成实现 IEEE 802.11 标准的节点的默认信标间隔是 100 个时间单位 (TU)，即 1024 毫秒。

[0045] 如图 6 所示，时间周期 T_c 被分成多个数量 m 个时隙。时隙表示节点被调度在信道上的时间粒度。例如，如果周期性周期是 102.4 毫秒 ($n = 1$)，并且 $m = 8$ ，即每一个周期都被分成 8 个时隙，则 OMMAC 516 按 12.8 毫秒的粒度来调整主和客信道。极点指示节点 500 从一个信道跳转 / 切换到另一个信道的特定时间。不同的节点 502 可定义其自己的极点，但特定无线链路的发送者和接收者对同步各自的极点并同时跳转到同一信道（以下将在题为“示例性时间同步”的部分中更详细地描述）。在该实现中，极点被限制在时隙的边界处。因此，整数值表示极点。例如，如果 $m = 8$ ，则可使用 1 字节（8 位）来表示极点。在该实现中，例如，一个周期中可能存在多个极点。另外，两个节点之间的链路的极点根据该链路上的通信量来调整，如将在以下更详细地讨论的。

[0046] 在周期的开始处，所有节点 502 都返回到主信道并在主信道上停留至少 T_b ，该 T_b 被称为广播时间。OMMAC 516 调度在每一个周期中的该时间段内传输的广播分组。在该实现中，例如，节点 502 可在主信道上停留超过 T_b ，即， $T_h > T_b$ 。令 $T_\Delta = T_h - T_b$ 。在该时间期间，该节点仍然在主信道上但用于接收单播消息。时隙中的极点，即 T_h 在范围 $[b, m]$ 内有效。如果 T_h 等于 m ，则对应的节点 502 仍然在主信道上并持续整个周期并且不为该特定节点调度客信道。在该实现中，选择 b 的值以使得保留信标和广播分组的传输机会（例如， $b = 1$ ）。

[0047] 示例性信道切换调度

[0048] 现在描述供每一节点 502 用于调度客信道来切换的分布式算法。在通信链路处调度的调度客信道的各方面首先根据客信道的最大独立集 (MIS) 和着色问题的组合来一般

化。然后该算法的示例性分布式实现基于对其使用贪婪算法的 MIS 所产生的信令开销和益处之间的折衷,根据上述示例性框架来呈现。

[0049] 4. 用于最大化吞吐量的客信道调度

[0050] 如上所示,节点 502 保持在主信道上的时间量是可变的,该时间量与覆盖信标时间段内相邻节点的时间量部分重叠。节点/站在每一个周期中将周期 T_c 中的其余时间花费在客信道上。如果链路的源和目的地节点 502 同时切换到同一客信道,则该链路上的通信与主信道和其他客信道上的通信正交。因为 802.11 提供用于处理链路层处的竞争的 CSMA/CA (载波侦听多路存取/冲突避免),所以将链路上的双向通信视为原子并将其调度到周期 T_c 中的相同时隙。因此,信道切换是根据链路上的通信的;链路是双向的。

[0051] OMMAC 516 客信道调度操作确定选择哪一个通信链路,并且还确定在相应的客信道上通信多久以便基本上最大化总数据吞吐量(不恶化)。如果对于信道数不存在约束并且每一个链路上的通信都是贪婪的,并且因为每一个 802.11 站一次都仅涉及一个链路,则 OMMAC 516 选择最大数量的“非主要冲突”活动链路。如果两个链路需要相同的节点 502 来进行通信,则这两个链路是“主要冲突”。如果一通信链路上在每一个周期 T_c 都存在特定通信量,则该通信链路是“活动的”。

[0052] 令 V 和 E 分别表示网络 504 中的整个节点 502 和链路集(链路被示为各个通信链路 532-1 到 532-N),而 G 表示该网络的原始通信图。假设 G 中的节点总数为 N ,这些节点由 $I = 1, 2, \dots, N$ 来索引,从而使得使用 V_i 来表示 G 中的节点 i 。令 $l_{(i,j)}$ 表示节点 i 和节点 j 之间的双向链路 532,从而使得 $l_{(i,j)} = l_{(j,i)}$ 。OMMAC 516 将网络 504 的原始通信图 G (节点图)转换为链路图 G' ,其中该链路图 G' 中所表示的每一个节点 502 都是该原始节点图中的活动链路。如果 G 中的两个节点的对应链路是主要冲突,则在 G' 中的这两个节点之间存在链路。因此,图 G' 中的最大独立集对应于 G 中的非主要冲突链路集以使得选择最多数量的链路(最大化吞吐量)。

[0053] 该最大独立集问题众所周知是 NP 完整的,并因此已经提议了大量试探性算法以解其中不再可向其添加节点的 MIS (最大独立集)。在该实现中, G' 中的 MIS 对应于 G 中的最大且不可再调度链路的链路集。一种对 MIS 选择的公知试探是贪婪算法,该贪婪算法将图 G' 中的所有节点按其等级来进行排序并首先移除具有最低等级的节点,从而移除所选节点及其所有邻居节点。该最低等级节点选择继续直到移除所有节点。所有所选节点 502 都表示 MIS,并且 MIS 的数量在多个节点 502 具有相同等级的情况下可能不止一个。

[0054] 5. 信道数的约束

[0055] 在生成了 MIS 后,OMMAC 516 通过着色来解决信道数的约束。具体而言,OMMAC 516 将链路图 G' 中的 MIS 转换成图 G'' ,其中 G'' 中的每一个节点都是 G' 的 MIS 中的节点,并且如果 G 中的两个节点的对应链路是“次要冲突”,则在 G' 中的这两个节点之间存在链路。此处,在原始信道图 G 中,两个链路在这两个链路无法同时在同一信道上并行工作时是次要冲突。因此,如果两个链路是主要冲突,则它们必须也是次要冲突,但反之不然。图 G'' 被称为信道图并且利用着色算法来为 G'' 中所表示的节点 502 调度信道。OMMAC 516 选择 G'' 的 MIS 并将首先表示的客信道分配给 MIS 中的这些节点。OMMAC 516 然后从 G'' 中移除这些节点以及客信道集中的所使用的客信道。

[0056] OMMAC 516 继续 MIS 选择和信道分配,直到分配了所有节点 502 或直到客信道集为

空。在该实现中,将最后一个 MIS 分配给主信道,因为主信道也与所有客信道正交。如上所述,将链路分配给不同客信道以提升性能。因为信道切换引入某些切换等待时间,并且为了减少这些等待时间,链路集可被配置成停留在主信道上,因为该主信道也与所有客信道正交。鉴于此,并且在集中式算法中,被选择在主信道上工作的 MIS 可以是任何 MIS,例如,第一个 MIS。然而,为了打破分布式情形中的死锁,选择最后一个 MIS。这一死锁可能例如在多个节点/链路同时将其自身认为是在主信道上工作的节点/链路且不使用客信道的情况下发生。

[0057] 图 7 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的一组示例性节点、链路和信道图。在该示例中,图 7(a) 示出了网络拓扑结构和通信模式,其出于示例性说明的目的而与图 4 中的示例相同。该节点图中的五个活动链路被转换成图 7(b) 的链路图。OMMAC 516 基于上述贪婪算法:首先选择具有最低等级的节点,来选择 MIS。在图 7(b) 的示例中,MIS 是唯一的并且 OMMAC 516 选择 MIS(链路 (1,2)、(3,4) 和 (5,6)) 中的节点。OMMAC 516 将该链路图转换成图 7(c) 的示例性信道图。因为节点图中的所有六个节点都处于彼此的通信范围内,所以所有链路都冲突。因此,只有一个链路能够在任何时刻工作。因此,在图 7(c) 中的信道图中,所有三个节点都是连接的,因此连同主信道 1 一起利用两个客信道 (6 和 11)。可分别对这三个链路使用信道 1/6/11(例如,请参考图 4)。

[0058] 6. 信道切换的开销

[0059] 对于每一个节点 502, OMMAC 516 本地地解决由信道切换引入的开销并确定节点 502 是否是网络 504 中的数据吞吐量瓶颈。具体而言,令 T_{sc} 和 T_{Ni} 表示切换信道的的时间开销和节点 N_i 在每一个周期的信道占用时间,如果 $T_{sc} + T_{Ni} > T_c$,则信道切换影响节点 N_i 中的吞吐量。图 8 示出了根据一个实施例的转换成示出特定网络节点 502 处的数据吞吐量瓶颈的链路图 (b) 的示例性节点图 (a)。具体而言,图 8 示出了其中无法增加数据吞吐量,而相反数据吞吐量由于信道切换的开销而由于某种原因降低的情形。(图 3 说明了类似的情形)。例如,参考图 3,请注意,所有节点都在同一信道上,因此不存在信道切换的成本。参考图 8(b),链路图示出在对应的 MIS 中只存在一个节点,即链路 (3,2) 或者链路 (3,6)。如果选择 MIS = 链路 (3,2) 来在客信道上工作,则在节点 3 处存在特定开销。然而,链路 (3,2) 和 (3,6) 两者都需要涉及节点 3。因此,如果节点 3 正在实现信道切换操作,则降低总吞吐量。请注意,在调度一个链路的情况下,无法增加数据吞吐量。在这一情形中,OMMAC 516 移除对于节点 502 的信道切换调度。

[0060] 7. 极点决定

[0061] 先前各段已经描述了当在特定信道上调度通信链路时 OMMAC 516 如何建立。现在描述 OMMAC 516 如何确定用于节点的信道切换操作的极点。对于节点 N_i , T_{Ni} , 总信道占用时间由所有信道上的传输时间和接收时间两者组成。对于链路 $l_{(i,j)}$, 令 $T_{(i,j)}$ 表示所有信道上的信道占用时间。使得

$$[0062] \quad T_{Ni} = \sum T_{(i,j)}, \quad \forall T_{(i,j)} > 0 \quad (2)$$

[0063] 节点 N_i 上的链路 $l_{(i,j)}$ 的时间权重是:

$$[0064] \quad w_{N_i}^{(i,j)} = T_{(i,j)} / T_{Ni} \quad (3)$$

[0065] 因此,将链路 $l_{(i,j)}$ 的时间权重定义为

$$[0066] \quad w^{(i,j)} = \min(w_{N_i}^{(i,j)}, w_{N_j}^{(i,j)}) \quad (4)$$

[0067] 对于链路 $l_{(i,j)}$, 调度极点以使得

$$[0068] \quad T_g^{(i,j)} = w^{(i,j)} T_c \quad (5)$$

[0069] 根据等式 (5) 的选择客信道的时间保证 : (a) 客信道上的时间根据该链路上的通信量来调整, 从而使得在通信增加的情况下分配更多的时间 ; 以及 (b) 与周期时间的比例确保不影响对应节点 502 上的其他通信的需求的目标。

[0070] 在一个实现中, 例如, OMMAC 516 实现用于为每一个节点 502 调度信道切换的集中式算法。该集中式算法始终为自组织网络通信和节点提供完整且已更新的信息, 而分布式算法提供本地化的这一信息。集中式算法 MIS 选择可具有多个解, 而将只选择一个 MIS 解。相反, 使用分布式算法, 结果由在邻居节点处采取的多个步骤来确定。这意味着使用分布式算法, 每一个节点的决策都影响整个结果, 但不控制该整个结果。例如, 与集中式算法相比, 这一分布式算法中所采取的步骤是异步的, 并且可以是并行的、冲突的, 或者甚至导致死锁情形。(该后一种情形, 如上所述, 通过使用对应于最后一个 MIS 的主信道和信道协商协议来解决)。

[0071] 在该实现中, 例如, 用于为每一个节点 502 调度信道切换的集中式算法如下 :

[0072] (a) 评估对应于节点 502 的信道占用时间并且在 $T_{cs} + T_{Ni} > T_c$ 的情况下移除节点 N_i ; 所得节点图是 G 。

[0073] (b) 如上所述地将节点图 G 转换为链路图 G' 并且使用贪婪算法来从 G' 中选择 MIS。

[0074] (c) 将链路图 G' 转换成信道图 G'' 并从 G'' 中选择 MIS 并且分配客信道, 直到 G'' 或客信道为空。

[0075] (d) 根据链路的时间权重来调度信道切换极点。

[0076] 用于信道切换的示例性分布式算法

[0077] OMMAC 516 如下设计 : 1) 每一个节点 502 仅具有网络 504 的部分更新信息, 因此 OMMAC 516 本地地根据从一跳邻居节点 502 获取的信息来优化信道切换决定 (该决定独立于来自远于一跳范围的节点的信息) ; 2) 每一个链路都异步地在客信道上调度, 因此在集中式模式中 OMMAC 516 链路调度根据其他已调度的链路来调整 ; 以及 3) OMMAC 516 实现用于分布式信道切换的客信道协商的协议并适应于通信。这些方面将在以下更详细地描述。

[0078] 8. 本地 MIS

[0079] 无线网络 504 中的每一个节点 502 通常都只获取其通信范围内的通信和信道的信息。由此, 由每一个节点生成 / 观察的网络通信图 G 是部分的。因此, 对于每一个节点 502 的 MIS 选择可以是不同的。在该小节中, 讨论用来自邻居节点 502 的信息来更新节点的信息。具体而言, OMMAC 框架 516 提供主信道上的链路层广播。由此, 每一个节点 502 都被配置成传送周期性广播消息以便与其他节点 502 交换通信和信道信息。在该实现中, 例如, 每一个节点 502 都每秒传送一次这一广播消息, 但可使用其他周期性时间帧。为了减少可能导致过期信息的延迟消息的机会, 每一个节点 502 都广播其通信和信道信息而不是为其他节点中继信息。在另一实现中, 这一信息交换独立于广播消息来实现。例如, 在一个实现中, 节点的通信和信道信息被承载在正在进行的通信首部中。

[0080] 9. 异步信道切换调用

[0081] 在集中式模式中,OMMAC 516 为网络 504 的整个链路图计算 MIS。OMMAC 516 然后使用贪婪算法(最低等级优先)来分配所有链路。贪婪(最低等级优先)算法是已知着色算法的示例。在实际分布式模式中,每一个节点 502 都异步地协商信道切换调度,这意味着节点 502 基于从邻居节点 502 接收到的所观察到的信道切换调度来作出调度决定。这一异步调度通过协商消息来平滑网络 504 上的任何消息激增,并且在主信道时间上平衡数据吞吐量负载。在一个实现中,并且为了进一步减少协商消息的开销,OMMAC 516 生成覆盖对应于多个时间周期(例如,请参见图 6)的信道切换活动的信道切换调度。

[0082] 在一个实现中,OMMAC 516 实现供节点 N_i 如下调度信道切换模式的分布式算法。

[0083] • 节点 502 周期性地广播节点 502 上的任何活动链路的相关的占用时间 T_{Ni} 、时间权重和信道。

[0084] • 节点 502 收集从邻居节点 502 接收到的广播消息以构造表示所有二跳邻居节点和活动链路的图 G 。例如,二跳节点信息通过评估来自一跳节点的广播消息来获取。为了示出这点,请假设存在五个节点的链拓扑结构,这五个节点为节点 A 到 E,如下: A-----B-----C-----D-----E。在该示例中,节点 A 只能听见 B,B 能够听见 A 和 C,...E 只能听见 D。节点 C 从节点 D 接收广播,该广播包含与 $D > C$ 和 $D > E$ 相关联的通信信息。因此节点 C 知道存在节点 E,并且 E 不是 C 的邻居,因为 C 无法直接听见 E。类似地,C 知道 A。以此方式,每一个节点都可从二跳内的所有节点收集信息。

[0085] • 节点 502 评估对应于每一个邻居节点 502 的信道占用时间;如果 $T_{cs} + T_{Nj} > T_c$,则从 G 中移除邻居节点 N_i 。

[0086] • 节点 502 将节点图 G 转换成链路图 G' 。然后节点 502 评估该链路图以检查所有活动链路的调度,在节点的对应链路已被调度的情况下将该节点移至 MIS。节点 502 继续使用贪婪算法来从 G' 中标识 MIS 中的节点。

[0087] • 节点 502 将链路图 G' 转换成信道图 G'' ,从 G'' 中选择 MIS,如上所述,并且基于调度的链路信道设置来分配客信道,直到 G'' 或客信道集为空。

[0088] • 如果链路 $l_{(i,j)}$ 在 G'' 中并且未被调度,则节点 502 根据该链路的时间权重来调度信道切换极点(请参见等式 (3))。

[0089] 10. 用于调度协商的示例性协议

[0090] 图 9 示出根据一个实施例的示例性信道切换调度协商协议实现。在该示例中,每一个节点 N_i 和 N_j 都表示图 5 的各个节点 502。参考图 9,并且基于上述分布式算法,如果节点 N_i 902 决定用邻居 N_j 904 来调度信道切换,则 N_i 向节点 N_j 发送切换信道(SC)请求消息以指示其希望用 N_j 来调度信道切换。(出于示例性说明的目的,这一 SC 请求消息被示为图 5 中的“其他程序数据”538 的相应部分)。在该实现中,SC 请求消息包括所提议的极点和客信道。

[0091] 参考框 906、908 和 910,并且在该实现中,各个消息发送和接收节点响应于发送/接收消息来启动各自的定时器以避免对于节点 502 的邻域内的节点的同步的信道切换调度协商。(示例性定时器逻辑 528 将在以下段落 [0062] 处的题为“定时器逻辑”的部分中更详细地描述)。响应于接收到该请求消息,节点 N_j 检查 $T_{sc} + T_{Nj} < T_c$ 是否成立。请回想, T_{sc} 和 T_{Ni} 分别表示切换信道的的时间开销和节点 Ni 在每一个周期的信道占用时间;如果 $T_{sc} + T_{Ni}$

$> T_c$, 则信道切换影响节点 N_i 中的吞吐量。如果通过 (即, $T_{sc} + T_{N_j} < T_c$), 则 N_j 验证所提议的客信道是否已由任何一跳邻居节点 502 保留。如果所提议的客信道尚未保留, 则 N_j 用确认 (ACK) 消息 (单播) 来向节点 N_i 确认该请求。在这种情况下, 所提议的极点处的已调度的信道切换动作由 N_i 和 N_j 在每一个周期中执行。在替换方案中, N_j 向作出请求的节点发送非确认 (NACK) 消息以拒绝和取消信道切换请求。在该实现中, 节点 N_j 将拒绝这一请求的原因通知给 N_i 。如果拒绝信道切换请求的原因是由于已占用的信道, 则 N_j 将不同的所建议的候选客信道承载到节点 N_i , 如果可能的话。

[0092] 在一个实现中, 节点 502 可以例如用 ACK/NACK 消息来向其他节点 502 提供反馈以确认 / 拒绝切换请求。如果结果是 NACK 或者在可配置的时间量过去后 (超时) 未在节点 N_i 处接收到反馈, 则节点 N_i 将信道切换请求视作失败的请求。否则, 在该实现中, 节点 N_i 和节点 N_j 两者都广播 SC 通知消息两次: 第一个这样的消息在分别在节点 N_j 或 N_i 处发送或接收 SC 响应消息后广播。在该实现中, 这一双重通知基本上防止了丢失通知广播的任何影响。如果节点在当前时间周期中的广播信道时间实现信道切换协商, 则第二个 SC 通知消息在下一时间周期中的广播时间传送。后一种情形在图 9 中示出。请注意, 在该实现中, 两个节点 502 可能将例如由于极点变化或先前调度的客信道上的调度延长而协商新的切换调度。在这种情况下, 并且在该特定实现中, 本地 OMMAC516 分别基于下一个和第三个周期中的广播时间来分别调度这两个广播。

[0093] 示例性网络节点设计

[0094] 参考图 1, 如上所述, 每一个节点 502 都实现系统 500 的信道切换策略的相应部分。在一个实现中, 该信道切换策略是集中式的。在另一个实现中, 该信道切换策略是分布式的。具体而言, OMMAC 516 包括用于控制 NIC 驱动程序 518, 管理操作系统非实时软件中断并遵从任何传统的 NIC 驱动程序调度 / 缓冲要求的逻辑。例如, 在一个实现中, OMMAC 516 包括通信测量逻辑 522、信息交换逻辑 524、信道切换调度器逻辑 526 (“按切换逻辑调度”526)、定时器逻辑 528 以及分组缓冲逻辑 530。该示例性次序不暗示或限制体系结构次序或该逻辑的各部分之间的数据流。相反, 所呈现的逻辑次序 522 到 530 是出于描述的目的。现在描述这些模块的各示例性方面。

[0095] 示例性通信测量和信息交换

[0096] 通信测量逻辑 522 例如生成已发送和接收到的分组的日志。通信测量逻辑 522 还为传入和传出分组计算相关联的通信量。在该实现中, 在逻辑 522 处测得的通信使用已知的时间滑动窗平均算法来实现这些计算以便提供对应于先前的可配置时间段的平均通信量。信息交换逻辑 524 传播对应于相应节点 502 的信道切换时间表。出于示例性说明的目的, 这些日志和信道切换时间表被示为“其他程序数据”538 的相应部分。信息交换逻辑 524 还组织从相邻节点 502 接收到的通信信息和信道切换时间表。调度和切换逻辑 526 利用任何这样的所接收到的信道切换时间表来实现示例性信道切换调度操作。

[0097] 示例性调度和切换逻辑

[0098] 调度和切换逻辑 526 基于由通信测量逻辑 522 收集的节点专用信息 (自信息) 以及由信息交换逻辑 524 收集的邻居节点 502 信息来执行例如上述贪婪算法。调度和切换逻辑 526 还访问 / 利用 NIC 驱动程序 518 所展示的应用程序编程接口 (API) 534。API 534 提供到 NIC 驱动程序 518 的接口以切换信道并同步信道切换时间表以使得节点 502 中的相应

节点同时切换到同一信道。现在描述节点信道切换同步的示例性实现。

[0099] 跨多个信道来同步节点之间的同步信道切换操作是有挑战性的,因为在不同信道上工作的节点更有可能丢失同步。使用常规的信道切换技术,网络节点之间的信道切换时间的差异经常导致分组重传,并且甚至可能导致分组丢失。多个因素影响信道切换同步。这些因素包括例如,所利用的特定同步协议、时钟漂移、温度等。调度和切换逻辑 526 用允许节点 502 同时切换到特定信道频率的实用节点信道切换同步方案来解决这些挑战。

[0100] 在该实现中,调度和切换逻辑 526 在基于 802.11 超帧的周期性时间周期上操作。(也请参考与描述这一周期性时间周期的图 6 相关联的上述描述)。802.11 协议信标在广播消息时间段的开始处广播。接收到的信标指示 802.11 MAC 时间。响应于接收到信标,每一个节点 502 更新其各自的 802.11 时间戳(即,在每一个周期的广播时间)。每一个节点 502 然后在每一个周期确定本地机器时间与 MAC 时间的任何差异。在调度和切换逻辑 526 计算对应节点 502 的信道切换时间表时,该调度和切换逻辑 526 将与该时间表相关联的时间转换成 802.11 MAC 时间。这些转换在信息交换逻辑 524 将该时间表向邻居节点 502 的任何传送之前执行。响应于接收到来自其他节点 502 的这些信道切换时间表,调度和切换逻辑 526 将所指示的 802.11 MAC 时间转换成相关联的本地机器时间以调度信道切换事件。

[0101] 在该实现中,NIC 驱动程序 518 展示 API 534,其允许切换逻辑 526 请求 NIC 驱动程序 518 向作出请求的节点 502 发送 MAC 时间(例如,802.11 MAC 时间)。API 534 的该相应部分允许作出请求的节点 502 计算机器时间与 MAC 时间的差异。在该实现中,MAC 时间粒度是 1 微秒,并且 MAC 时间的同步误差是大致也是 1 微秒的传播延迟误差。机器时间粒度例如通常是 100 纳秒。因此,并且在该特定实现中,切换逻辑 526 提供微秒级节点信道切换同步。

[0102] 此外,因为每一个节点 502 都在同一主信道上操作并且 802.11 MAC 只在该主信道上广播信标,所以在向网络 504 添加新节点时,该新节点使用 802.11 扫描相位来实现网络发现操作。

[0103] 定时器逻辑

[0104] 出于示例性描述的目的,描述 WINDOWS 操作系统的定时器方面。这些所描述的方面类似于其他操作系统实现。WINDOWS 上的默认定时器粒度是 10 毫秒并且应用程序通常能够获取 100 纳秒级的准确时间测量。对于内核中的严格的定时器要求,WINDOWS 提供用于将定时器粒度调整为大致 1 毫秒的 API。然而,在调整定时器粒度后,并且尽管平均定时器误差小于 1 毫秒,但定时器误差在操作系统(OS)负载非常高时仍然可能大于 10 毫秒。其一个原因是因为软件和硬件事件通常被置于供处理的基于优先级的队列中。硬件事件通常具有从供处理的事件队列中移除的最高优先级,而软件事件具有许多可能的优先级。因此,存在对基于操作系统定时器中断使用软件定时器来实现信道切换时间表的实际限制。

[0105] 与常规信道切换定时器相反,OMMAC 516 基于与操作系统相关联的多处理时间片来实现软件定时器逻辑 528。系统 500 的定时器逻辑 528 解决常规软件定时器的限制以便为分组和信道切换调度提供定时器。具体而言,定时器逻辑 528 将定时器线程的优先级设为最高软件优先级以使得这些定时器线程将从事件队列中迅速移除并进行处理。这一优先级调整基本上减少了在常规信道切换实现中很明显的信道切换等待时间。对于分组和信道切换调度,定时器逻辑 528 响应于分组时间表的定时器线程的到期/激发来通知分组缓冲

逻辑 530 向特定邻居节点 502 发送分组。(这一缓冲逻辑在以下题为“示例性分组缓冲逻辑”的部分中描述)。在另一示例中,当信道切换定时器激发时,定时器逻辑 528 通知切换逻辑 526 向 NIC 驱动程序 518 传送信道切换命令/请求(例如,经由 API 534)。

[0106] 示例性数据流

[0107] 图 10 示出了根据一个实施例的用于自组织网络的分布式 OMMAC 的另一示例性节点体系结构。具体而言,图 10 示出了分组路由逻辑、OMMAC 516 的数据和控制平面部分和 NIC 驱动程序 518 之间的示例性数据流。数据平面表示分组缓冲逻辑。这一分组缓冲逻辑在图 5 中被示为分组缓冲逻辑 530。控制平面包括通信测量逻辑 522、信息交换逻辑 524、分布式通用调度和切换逻辑 526 和定时器逻辑 528,如以上参考图 5 所描述的。因此,并且出于示例性描述的目的,图 10 的各方面参考图 5 的组件来描述。在该描述中,当组件附图标记的最左边的数字以“5”开头时,该组件在图 5 中首次引入。

[0108] 参考图 10,OMMAC 516 包括数据平面和控制平面。该数据平面从分组路由逻辑 1002 接收分组,如数据通信箭头 1004 所示。该数据平面还从 OMMAC 516 的控制平面部分接收消息(例如,协商消息、信令消息等)。(来自控制平面的消息用从控制平面到数据平面的、方向从左到右的黑色(填充的)箭头来表示)。如图所示,数据平面分类器逻辑 1006 基于分组目的地并基于该分组将会在主信道上还是在特定客信道上上传送来将所接收到的分组/消息(统称为分组)放入节点目的地队列 1008。这些目的地队列在图 10 中被示为例如“目的地 1”到“目的地 n”。节点目的地和信道标识由相应的分组首部来提供。箭头 1010 表示来自 NIC 驱动程序 518(到 NIC 硬件 1012 的接口)的传入(上行链路)消息。如图所示,OMMAC 516 将表示数据的传入消息传送到路由逻辑 1002。OMMAC 516 将不是数据的传入消息传送到控制平面。

[0109] 如先前部分中所指示的,网络节点的本地 OMMAC 516 包括定时器逻辑 528,其用于触发对应节点的极点在周期性时间周期期间从主信道切换到相应客信道(在该周期性时间周期结束时该节点切换回主信道)。节点的极点基于由通信测量逻辑 522 的通信测量提供的通信以及由其他节点提供的通信测量。为此,该节点与其他节点交换其特定通信测量以使得这些其他节点可作出其自己的极点决定。在图 10 的示例中,从控制平面到数据平面的从左到右的箭头表示数据通信,该数据通信的至少一部分是供排队并传送到自组织网络中的其他各个节点的该特定节点的通信测量。这一通信由箭头 1014 示出。另外,并且如先前部分中所描述的,分布式信道调度逻辑 526 调度对应于该节点的信道切换时间(例如,基于贪婪算法)并将这些时间表传送到其他节点以供进行信道切换同步操作。

[0110] 示例性分组缓冲逻辑

[0111] 现在更详细地描述图 10 的数据平面(也由图 5 的分组缓冲逻辑 530 来表示)的操作。该分组缓冲逻辑缓冲来自分类器逻辑 1006 和来自控制平面的分组以便将这些分组传送到自组织网络中的其他网络节点。在该实现中,缓冲逻辑 530 将分组缓冲在按优先级排序的目的地队列 1004 中。(这一优先级排序不是先进先出优先级排序,而是基于由 OMMAC 516 的一个或多个相应部分(例如,定时器逻辑 528)和/或其他程序模块提供的特定优先级)。例如,在优先级队列中,OMMAC 516 向诸如上述信道切换请求等信令消息提供比其他应用通信相关联的消息高的相应优先级。

[0112] 在该实现中,例如,每一个节点 502 都实现用于控制分组向在不同时刻调度的不

同的目的地节点的传输的每节点邻居 (per-node-neighbor) 目的地队列 1006。如果多个邻居节点队列 1006 在同一时间 (例如, 在主信道上) 是活动的 (例如, 活动队列展示阈值通信量) 且是有效的, 则分组根据循环次序来服务。如图所示, 分组缓冲和调度逻辑, 而不是 NIC 驱动程序 518 或对应的硬件 1012 维护优先级队列。这允许切换和调度逻辑 526 在调度对应的信道时控制所有分组传输时间。

[0113] 根据循环次序来服务分组不同于根据先进先出 (FIFO) 调度来服务分组。这是因为节点 502 根据不同队列中的各个优先级来缓冲分组。始终首先服务具有较高优先级的队列, 然后服务具有较低优先级的队列。具有相同优先级的队列通过循环调度来服务。对于 FIFO, 分组根据到达时间来服务。相反, 并且在该实现中, 服务来自队列的较高优先级分组, 即使该较高优先级分组晚于以较低优先级排队的分组到达。循环操作也不同于 FIFO 队列处理操作。例如, 请假设队列 1 和队列 2 具有某一优先级, 并且队列 1 的分组到达率两倍于队列 2 的分组到达率。在 FIFO 的情况下, 给予队列 1 两倍于队列 2 的机会, 而在循环的情况下, 给予队列 1 与队列 2 相同的机会。

[0114] 在常规信道切换模式切换信道时, 由 MAC 或在硬件中缓冲的分组通常由于在切换信道处发生的硬件重置而被丢弃。另外, 分配给数据吞吐性能的 MAC 或硬件缓冲空间的大小通常是受限的, 从而经常导致缓冲区溢出情况。为了解决常规系统的这些限制, 分组缓冲和调度逻辑只在先前传送的分组已由 NIC 驱动程序 518 服务时才向 NIC 驱动程序 518 发送分组。为了提供该信息, NIC 驱动程序 518 向信息交换模块 524 传送成功状态 (请参见上行链路箭头 1010)。信息交换模型 524 然后将该状态传送到分组缓冲和调度逻辑。在这种情况下, 并且在先前的分组在 NIC 驱动程序 518 处待决时, 分组缓冲和调度逻辑停止任何下一分组直到接收到这一成功状态。另外, 在该实现中, 切换和调度逻辑 526 使用定时器逻辑 528 来配置任何已调度的信道切换事件之前的分组消耗时间。该分组消耗时间向无线 NIC 硬件 1008 提供在信道切换操作之前传送任何缓冲分组的时间。在该实现中, 分组消耗时间是 1 毫秒。

[0115] 示例性实现

[0116] 图 11 示出了根据一个实施例的实现用于自组织网络的分布式 OMMAC 的网络节点的示例性实现 1100。该特定示例示出了节点 502 (也请参见图 5), 该节点 502 具有在 WINDOWS 平台上实现的 OMMAC 516。该表示只是出于示例性目的, 作为用于实现本地 OMMAC 516 的许多可能的平台中的一个样本平台。例如, 节点的本地 OMMAC 也可以在 Linux 平台、UNIX 平台等上实现。参考图 11, 该实现的栈模块在图 11 中用单阴影填充来示出。例如, OMMAC 516 被实现为 WINDOWS XP 平台上的 IM (媒体间) NDIS (网络驱动程序接口规范) 驱动程序。NIC 小型端口驱动程序 518 (也请参见图 5) 提供: (a) 提供 802.11 MAC 时间 (由对应于 802.11 自组织网络中的所有节点 502 的信标来同步) 的接口 534 (也请参见图 5); (b) 用于接收和响应 NIC 驱动程序处的对信道切换的请求的接口 534 以及 (c) 等等。命令行工具 1102 提供用于控制 OMMAC 516 的行为的用户界面。这一行为包括例如, 改变诸如周期时间、时隙大小、切换信道的保护时间等参数。该示例还示出与“Win 套接字”进行通信的、在用户空间中运行的“应用程序”, 与无线 NIC 进行通信的内核空间中的“TCP/IP”和“NDIS”。同样, 这只是用于实现用于自组织网络的分布式重叠多信道 MAC 的系统和方法的网络节点的一个是示例性实现。

[0117] 示例性过程

[0118] 图 12 示出根据一个实现的用于自组织网络的分布式重叠多信道 MAC 的示例性过程。在一个实现中,过程 1200 的操作由图 5 的网络节点 502 的各个程序模块 510 来实现。出于示例性说明和描述的目的,过程 1200 的操作参考图 5 到 11 的组件和各方面来描述。在该描述中,范围从 100 到 999 的组件 / 操作 (步骤) 附图标记的最左边的数字表示其中该组件 / 操作被首次引入的附图。类似地,范围从 1000 到更大数字的组件 / 操作附图标记的最左边的两个数字表示其中该组件 / 操作被首次引入的附图。

[0119] 参考图 12,框 1202 的操作确定对应于自组织无线网络中的各网络节点的主信道频率 (“主信道”)。框 1204 的操作标识与主信道正交的一组客信道频率 (“客信道”)。从由无线网络协议提供的多个信道频率中选择 (即,确定和标识) 主信道和客信道。这一无线网络协议包括例如,IEEE 802.11。框 1206 的操作由自组织无线网络中的每一个节点来在主信道上操作并持续各自可变且重叠的时间量以维护与其他网络节点的网络连接。框 1208 的操作由每一个节点确定是否以及何时从主信道切换到各自被分配的客信道并持续可变时间量以便本地地优化自组织网络中的数据吞吐量。

[0120] 图 13 示出根据一个实施例的用于自组织网络的分布式重叠多信道 MAC 的另一示例性过程。框 1302 的操作将无线网络协议的周期性时间周期分成两个阶段。这两个阶段包括节点连接阶段和数据吞吐量提高阶段。该周期性时间周期与由这些网络节点中的相应节点根据无线网络协议来传送的周期性信标消息相关联。在一个实现中,无线网络协议基于 IEEE 802.11 无线网络协议。

[0121] 框 1304 的操作在连接阶段期间将自组织网络中的每一个网络节点配置成维护多个可用信道频率中的单个主信道频率上的连接。该多个可用信道频率由无线网络协议提供并根据该无线网络协议。框 1306 的操作在连接阶段期间将至少两个节点配置成在与主信道正交的多个客信道中的一个客信道上进行通信。框 1308 的操作在连接阶段期间由每一个网络节点使用主信道来维护自组织网络中的各个节点之间的连接。框 1310 的操作在数据吞吐量提高阶段期间由网络节点的至少一个子集使用一个或多个各自被分配的客信道来优化自组织网络中的数据吞吐量。以此方式,自组织网络中的网络节点维护节点到节点网络连接并提升该自组织网络的性能。

[0122] 结论

[0123] 虽然以上各部分已经用对结构特征和 / 或方法操作或动作专用的语言描述了用于无线自组织网络的分布式重叠多信道 MAC,但在所附权利要求中定义的实现不必限于所描述的具体特征或动作。例如,OMMAC 516 (请参见图 5) 信道切换等待时间可通过用硬件定时器来替换由非实时 OS 提供的软件定时器来减少。或者,在一替换实现中,定时器准确度通过向 OS 添加实时支持来提高。由此,上述用于无线自组织网络的分布式重叠多信道 MAC 的具体特征和操作是实现所要求保护的主题的示例性形式。

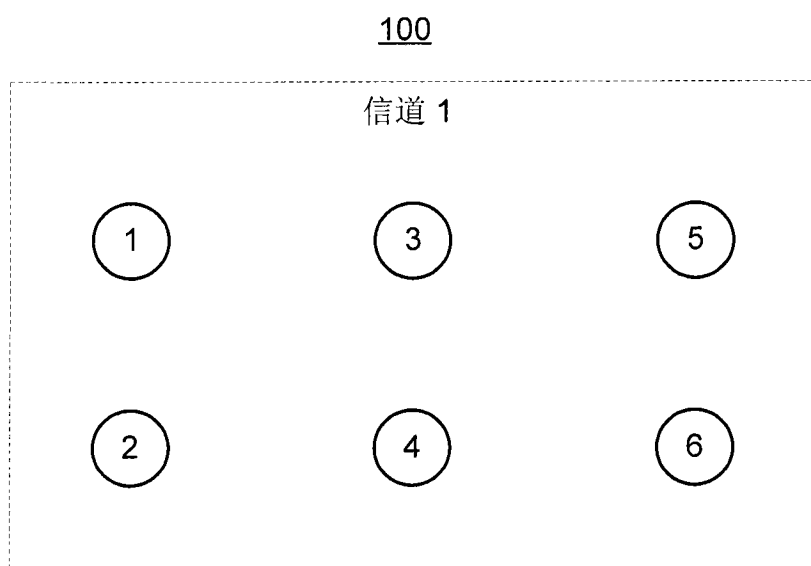


图 1 （现有技术）

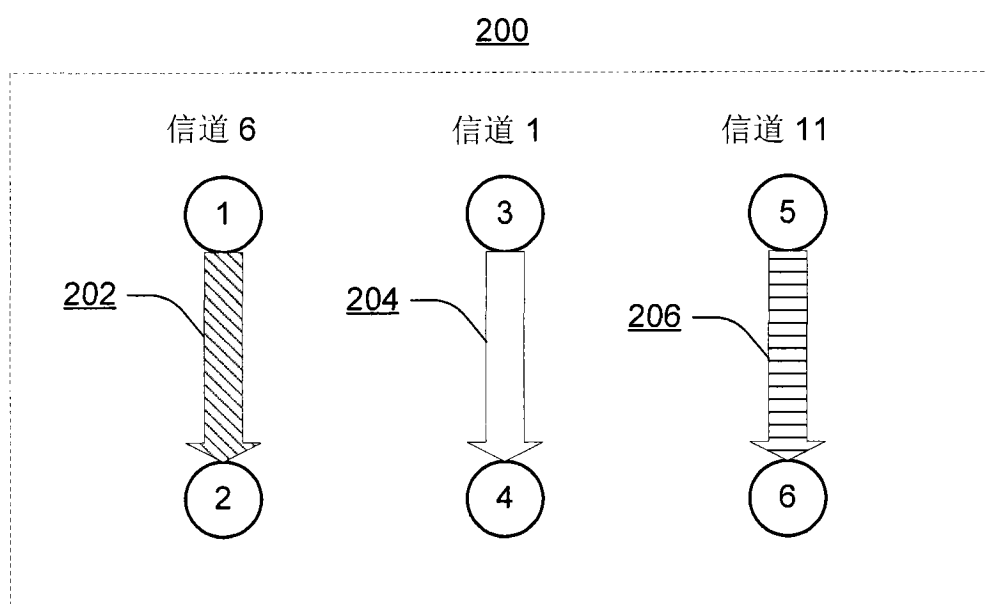


图 2 （现有技术）

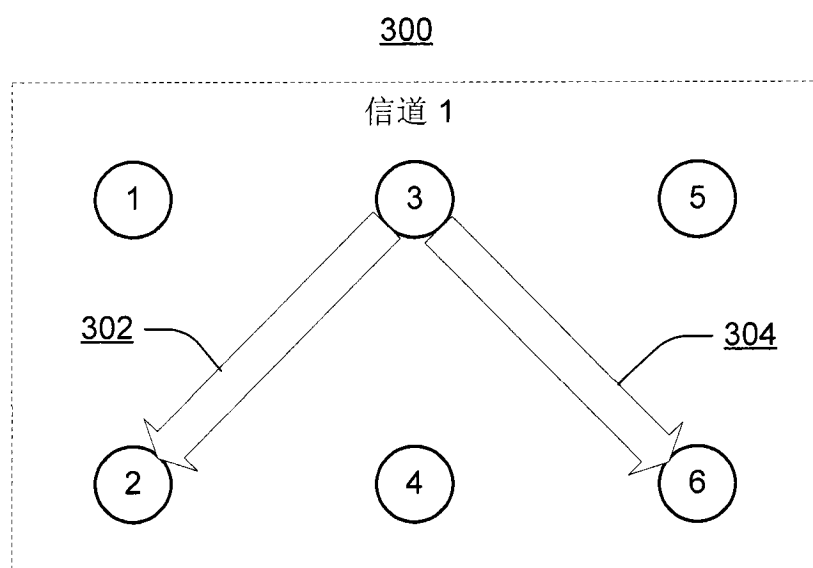


图 3 (现有技术)

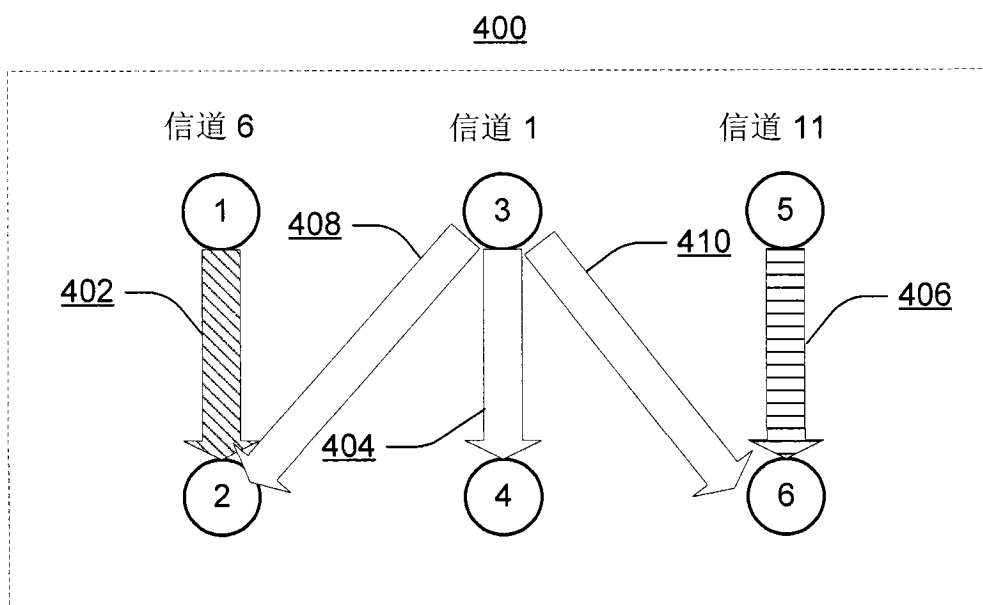


图 4 (现有技术)

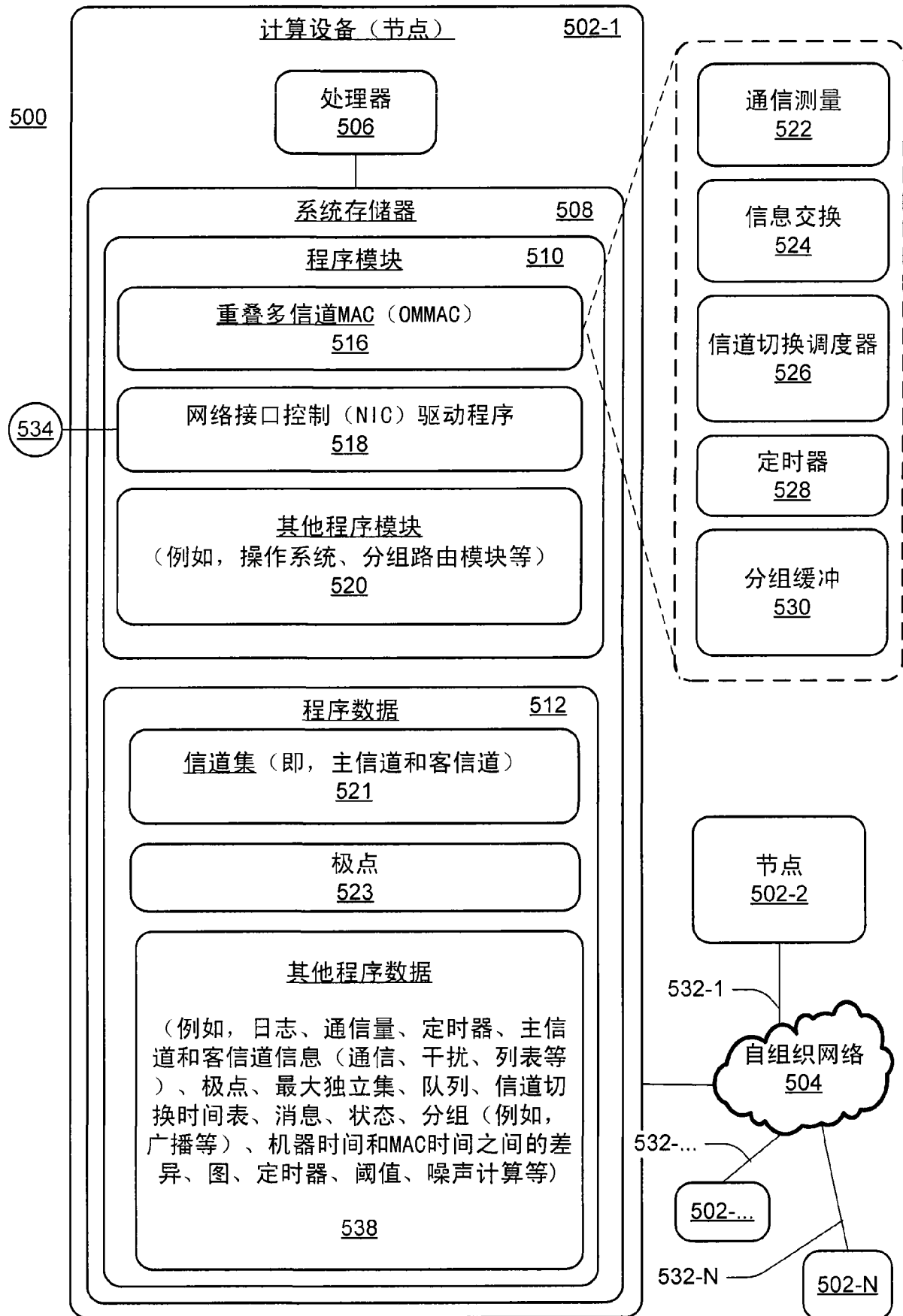


图 5

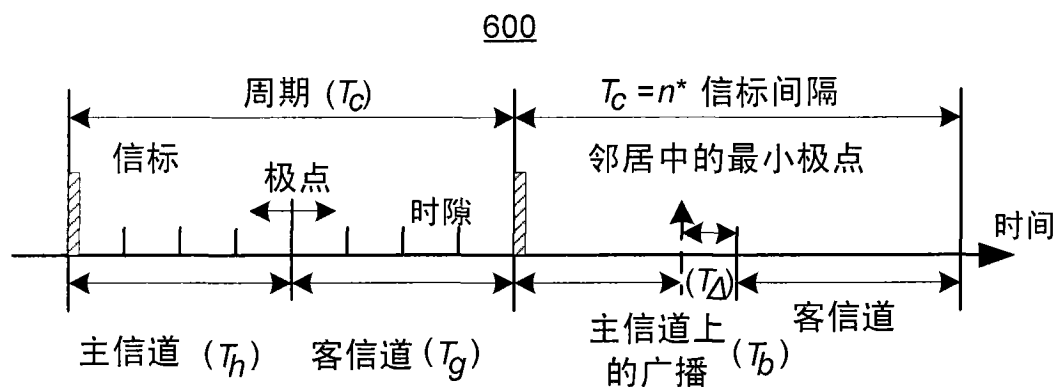


图 6

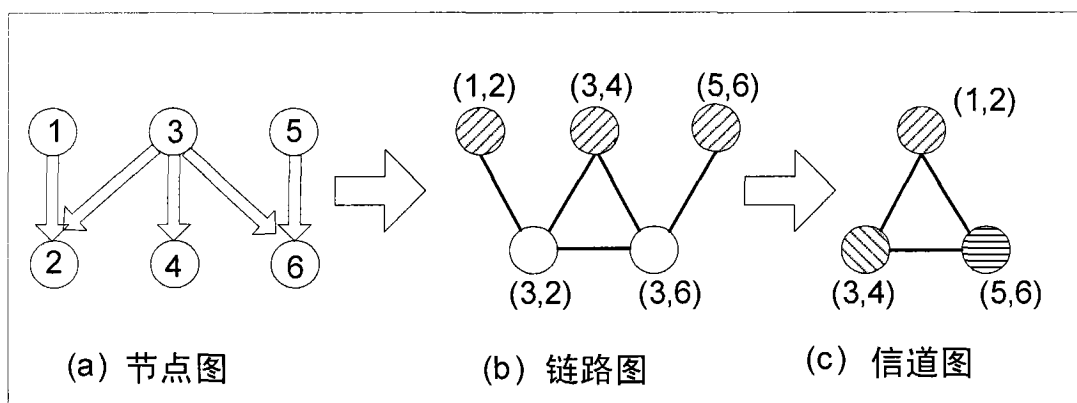


图 7

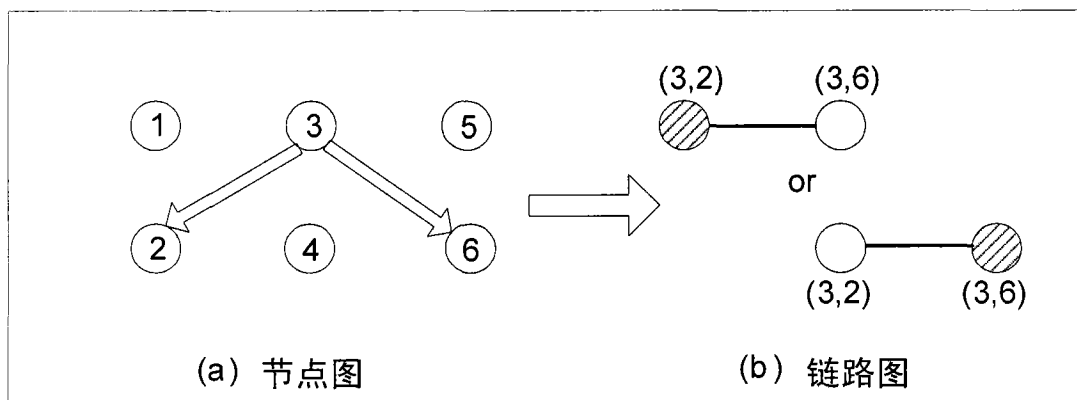


图 8

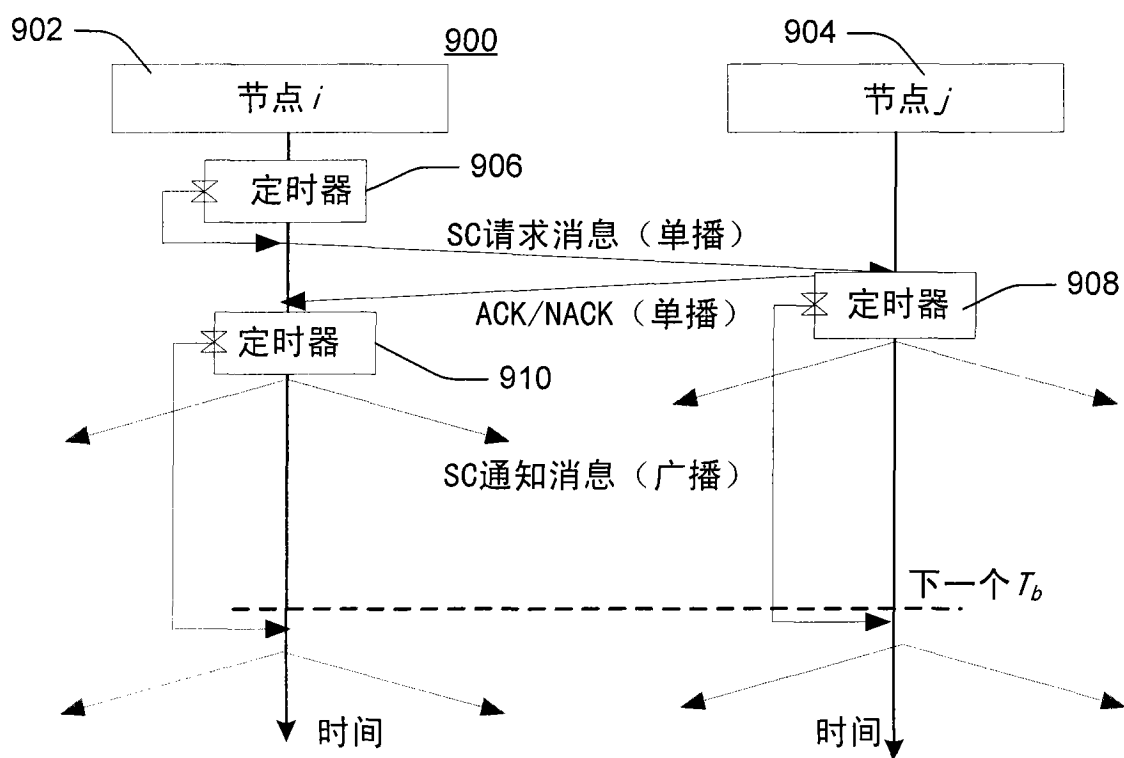


图 9

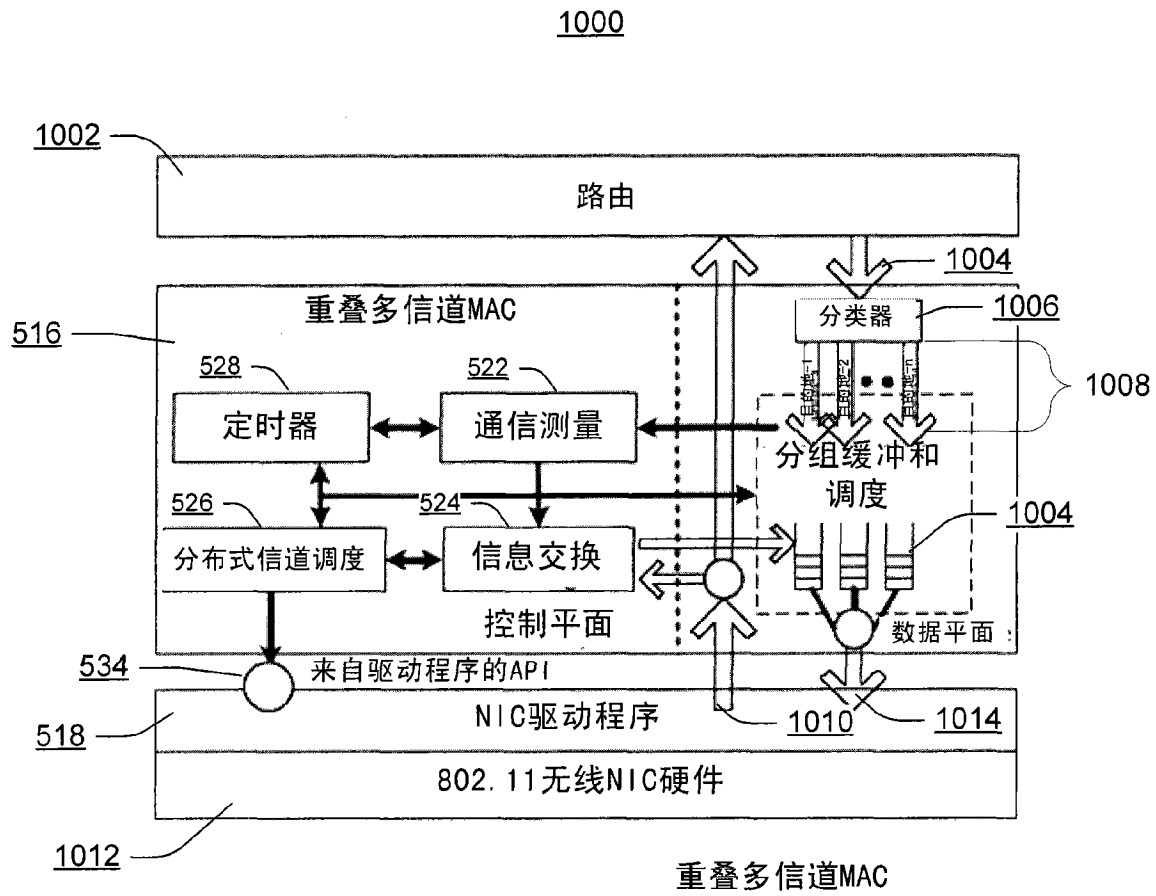


图 10

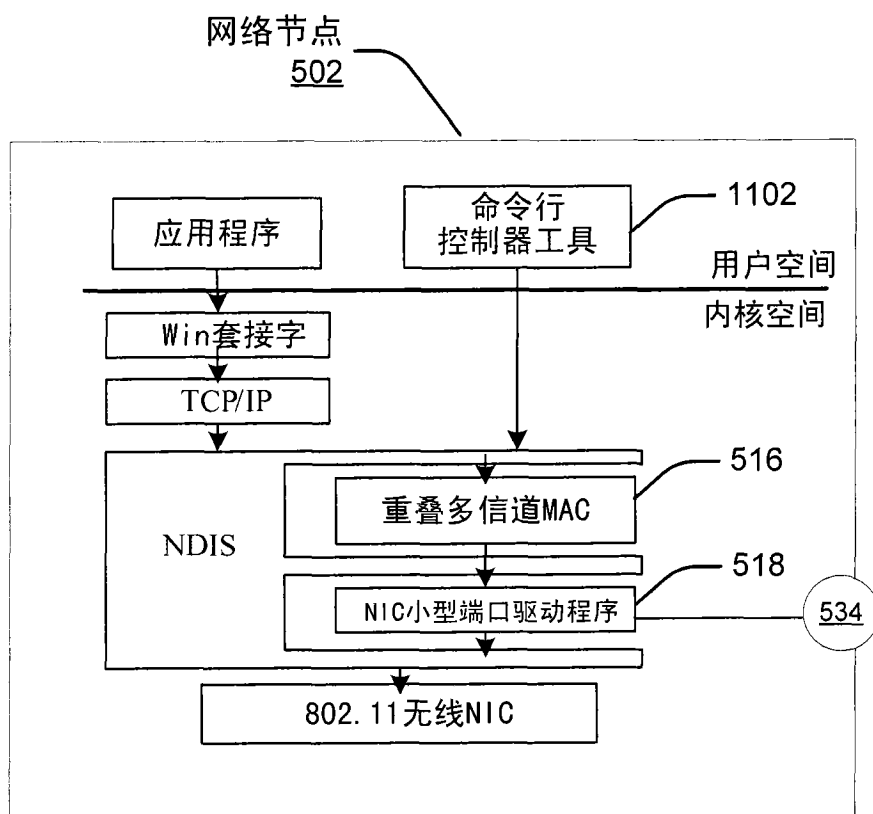


图 11

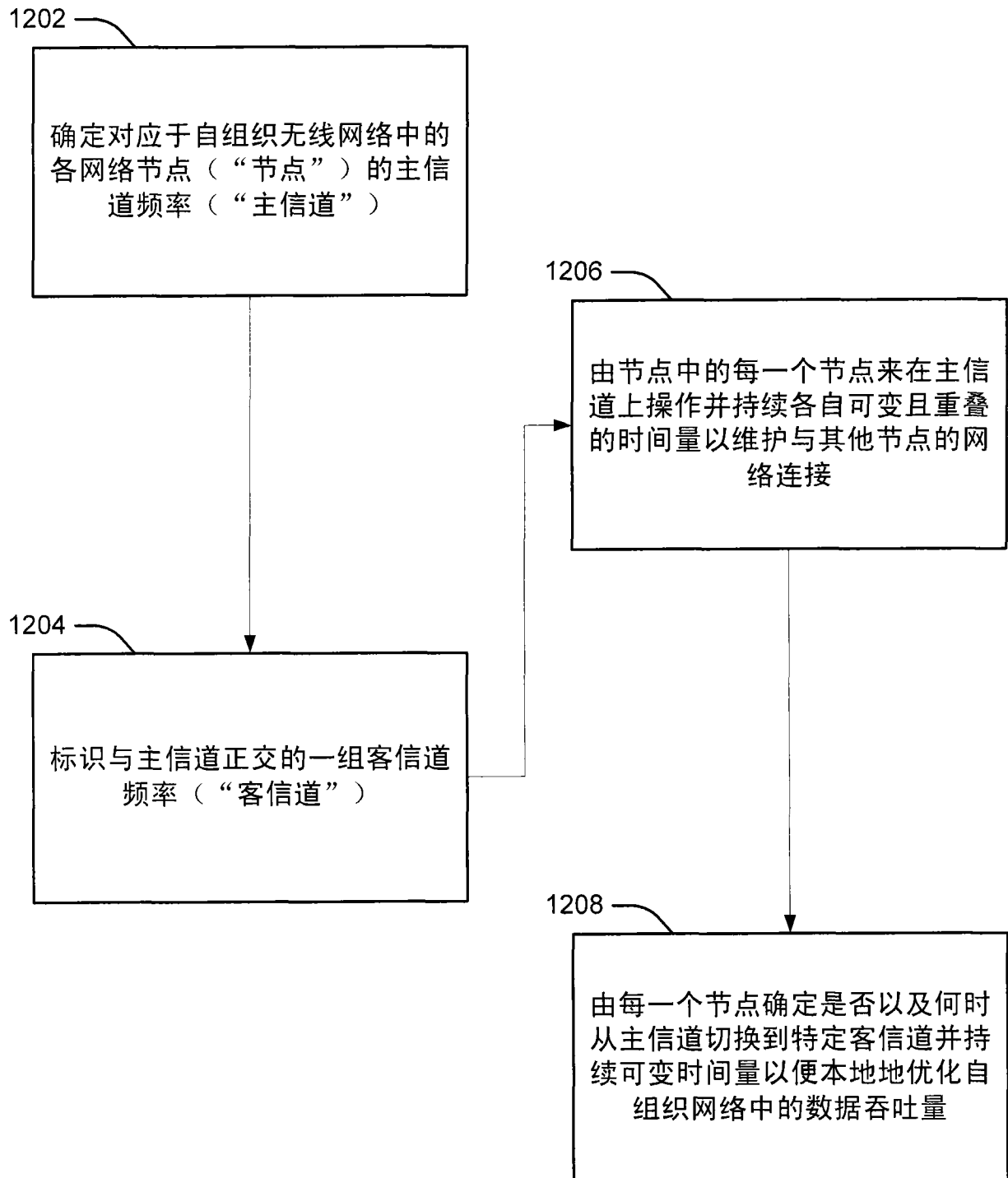
1200

图 12

1300

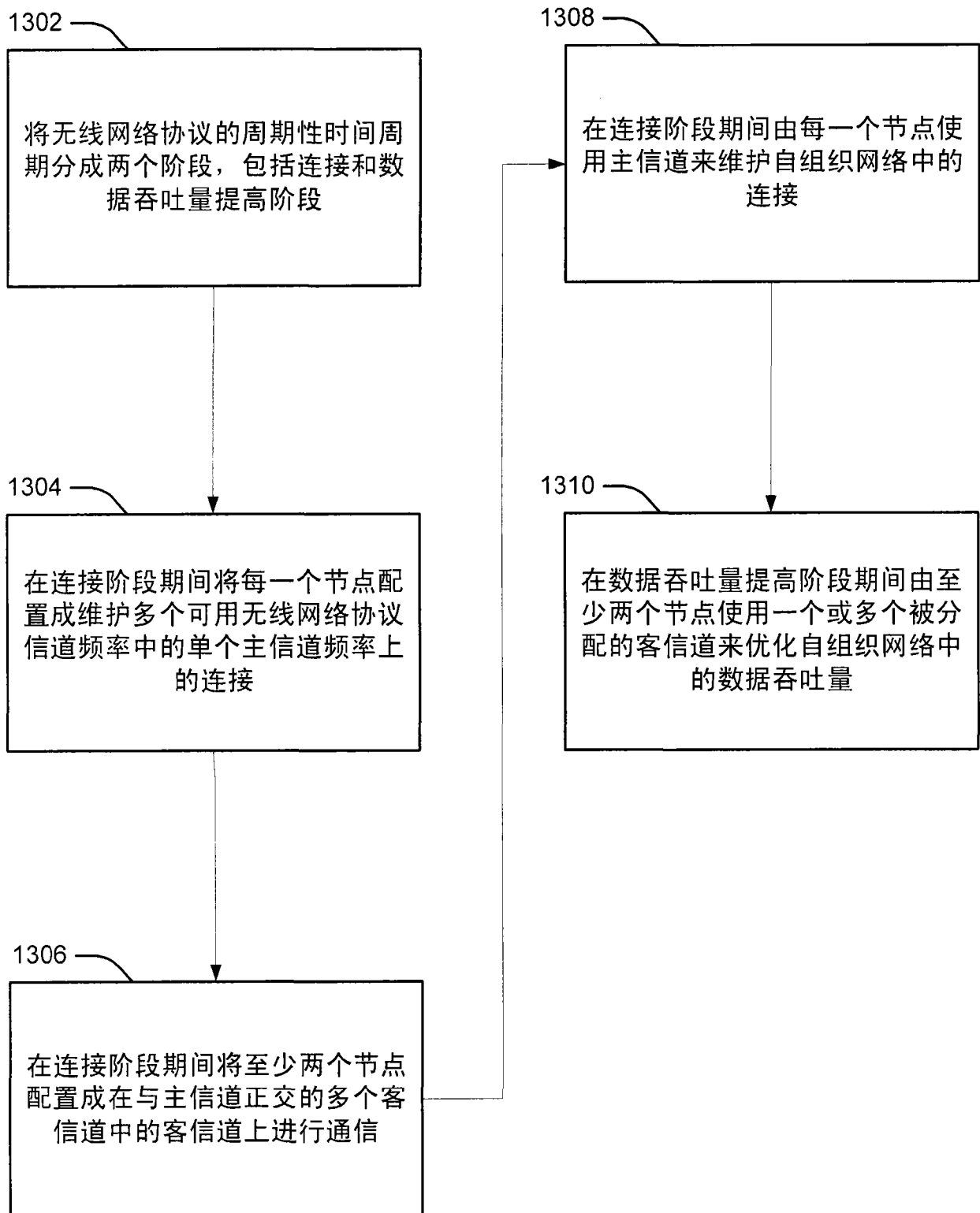


图 13