

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

H04Q 7/24

H04L 12/28 H04B 3/36



[12] 发明专利申请公开说明书

[21] 申请号 01822619.1

[43] 公开日 2004年4月14日

[11] 公开号 CN 1489870A

[22] 申请日 2001.12.12 [21] 申请号 01822619.1

[30] 优先权

[32] 2000.12.29 [33] US [31] 09/751,262

[86] 国际申请 PCT/US01/48215 2001.12.12

[87] 国际公布 WO02/054646 英 2002.7.11

[85] 进入国家阶段日期 2003.8.13

[71] 申请人 特罗普斯网络公司

地址 美国加利福尼亚州

[72] 发明人 德瓦布哈克图尼·斯利克里什纳

阿马拉沃亚尔·查里

[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

代理人 李 玲

权利要求书5页 说明书30页 附图15页

[54] 发明名称 用于为无线设备提供路由协议的方法和系统

[57] 摘要

本发明描述了一种用于产生连接的方法和设备。该方法包括：服务器广播一个包含服务器地址的信标。该方法还包括：接收该信标的每个客户机都重新广播这个信标。结果，接收到信标的每个客户机都知道了一条通向服务器的路线。

I S S N 1 0 0 8 - 4 2 7 4

1. 一种无线连接的方法，包括：
由一个服务器来广播一个信标；以及
在客户机接收并保存这个信标，该客户机还保留
指示一条通向服务器的路径的信息；以及
由客户机重新广播该信标。
2. 权利要求 1 的方法，其特征在于，包括：
保存接收到的每个信标；
将某个信标指示的一条路径制定为最优路径；
设置在最优路径中指示的缺省网关；以及
仅仅重新广播代表最优路径的信标。
3. 权利要求 1 的方法，其特征在于，服务器广播的信标包括一个设定为零的跳数，该方法还包括：
接收信标的每个客户机使用逐一递增的跳数来重新广播信标；
以使接收信标的每个客户机都了解一条到达服务器的路线以及
这条路径上的跳数。
4. 权利要求 1 的方法，其特征在于，服务器广播的信标包括服务器地址，以使接收信标的每个客户机都了解到服务器地址。
5. 权利要求 1 的方法，其特征在于，还包括：
接收信标的每个客户机都重新广播这个添加了客户机地址的信标；
以使接收信标的每个客户机都具有一条通向服务器的完整路径。
6. 权利要求 1 的方法，其特征在于，该信标包括一个代表当前路由循环的序列号。
7. 权利要求 6 的方法，其特征在于，还包括：一旦客户机接收到一个信标，则
确定是否接收到一个关于这个路由循环的信标；以及

如果没有接收到关于这个路由循环的信标，则根据该信标来保存一条通向服务器的路由选择路径。

8. 权利要求 7 的方法，其特征在于，还包括：如果已经接收到了关于该路由循环的信标，

确定该信标是否具有比这个路由循环的在先信标的序列号更高的序列号；并且如果这样的话，

将当前信标保存在存储器中。

9. 权利要求 6 的方法，其特征在于，还包括：一旦客户机接收到一个信标，

确定当前接收的信标是否代表关于这个路由循环的最优路径；以及

如果当前路径代表最优路径，则对当前信标中的缺省网关进行识别，并且保存缺省网关。

10. 权利要求 2 的方法，其特征在于，还包括：

确定是否存在一个经过识别的在先缺省网关；以及

从存储器中删除在先的缺省网关。

11. 权利要求 1 的方法，其特征在于，还包括：对每个客户机来说，

收集所有信标；以及

选择一个单独信标来进行广播。

12. 权利要求 11 的方法，其特征在于，选择一个信标，包括：

为每个信标识别服务器与客户机之间的许多跳；以及

选择跳数最少的信标。

13. 权利要求 11 的方法，其特征在于，选择一个信标，包括：

为每个信标识别一个流量监视码（TMC）；以及

选择具有最低 TMC 的信标。

14. 权利要求 11 的方法，其特征在于，选择一个信标，包括：

识别一个质量最高的信标；以及

选择具有最高质量的信标。

15. 权利要求 14 的方法，其特征在于，最高质量即为最佳信噪比。

16. 权利要求 14 的方法，其特征在于，最高质量是基于服务器上最多的后端带宽容量。

17. 权利要求 14 的方法，其特征在于，最高质量是基于服务器所处理的最低流量等级。

18. 权利要求 1 的方法，其特征在于，还包括：

向服务器发送一个反向信标；以及

在服务器中构建一个客户机树图，其中服务器具有一条通向所有客户机的路径。

19. 一种用于无线通信的服务器，包括：

一个信标逻辑部分，用于产生一个信标并且广播该信标；以及

一个无线收发信机，用于接收多个反向信标，其中反向信标指示的是一条通向各个客户机的路径；以及

一个客户机树图，用于保存通向各个客户机的路径，以使服务器能够直接或是通过网络上的其他客户机来向任何一个客户机发送数据。

20. 权利要求 19 的服务器，其特征在于，还包括一个监视逻辑部分，用于监视一个网络，该监视逻辑部分使用客户机树图来产生客户机网络的映射图。

21. 一种为一个系统产生路由选择路径的方法，其中该系统包含一个服务器和多个客户机，该方法包括：每个客户机

从服务器接收一个信标；

重新广播接收自上游节点的一个信标；以及

广播一个反向信标上行数据流，这个反向信标指向已知的上游节点，服务器和客户机使用反向信标来建立一个路由选择表。

22. 权利要求 21 的方法，其特征在于，某个客户机中的路由选择表包括一个缺省网关和一条从特定客户机通向各个下游客户机的路径。

23. 权利要求 21 的方法，其特征在于，还包括：服务器广播一个伪反向信标，以便启动反向信标循环。

24. 权利要求 21 的方法，其特征在于，还包括：每个客户机集中接收自下游客户机的反向信标，并且发送一个包含所集中信息的单独的反向信标。

25. 权利要求 21 的方法，其特征在于，客户机的缺省网关接收一个反向信标广播，由此触发客户机，以便启动一个定时器来发送反向信标。

26. 权利要求 21 的方法，其特征在于，还包括：如果客户机接收到多个信标，

对接收到的每个信标的链路质量进行评估；以及

基于具有最佳链路质量的信标来选择缺省网关，并且重新广播该信标。

27. 权利要求 26 的方法，其特征在于，链路质量包括信标可靠性。

28. 权利要求 26 的方法，其特征在于，链路质量包括与服务器后端带宽容量有关的信息。

29. 权利要求 26 的方法，其特征在于，链路质量包括与服务器所处理流量有关的信息。

30. 权利要求 21 的方法，其特征在于，服务器与客户机之间的连接是一个无线连接。

31. 权利要求 21 的方法，其特征在于，服务器与客户机之间的连接是从以下类型的连接中选出的：无线连接、有线连接以及交换连接。

32. 权利要求 21 的方法，其特征在于，还包括：客户机

从多个服务器接收多个信标；以及

选择多个信标中的一个，并将与选定信标相关的服务器设定成它的优选服务器；

由此自己选择属于一个与优选服务器相关的群集。

33. 权利要求 32 的方法，其特征在于，还包括：客户机移动到群集之外；

一旦从新的群集接收到信标，则客户机把与新信标以及新群集相关联的服务器设定成它的优选服务器。

34. 权利要求 33 的方法，其特征在于，还包括：

终止一个路由选择表，其中包含了在先的优选服务器以及在先的缺省网关。

35. 一种已选择路由的无线网络，其特征在于，包括：

多个有线接入点；

多个客户机，每个客户机属于一个受到单独服务器管理的客户机群集，以使客户机通过选择一个将要重新广播的信标来选择群集中的成员。

36. 权利要求 35 的网络，其特征在于，由于客户机从网络中的一个区域移动到另一个区域，因此群集中的成员可以改变。

37. 权利要求 35 的网络，其特征在于，由于向网络中添加了新服务器，因此群集中的成员可以改变。

38. 权利要求 35 的系统，其特征在于，新服务器可以是一个通过添加有线连接而被转换成服务器的客户机。

39. 权利要求 35 的网络，其特征在于，由于客户机选择的服务器无法提供适当的服务，因此群集中的成员可以改变。

用于为无线设备提供路由协议的方法和系统

技术领域

本发明涉及无线系统领域，尤其涉及在无线环境中对服务器与客户机之间的数据进行路由选择。

背景技术

使用无线设备访问互联网正在变得越来越普遍。然而，存取数据通常需要使用智能客户机，或是随着互联网规模扩张而需要高带宽。

目前使用的一种现有技术协议是由 DARPA 赞助开发的 SURAN 等级的协议。这组算法不但包括分层算法，还包括非分层算法。非分层算法缺乏可升级性——随着网络规模增长（ $N =$ 网络中的节点数目），进行路由选择所需要的带宽将会以 N 或 N^2 的形式增长。对所指定的必要数据通信带宽来说，这意味着可能实现的网络规模将会受到很大限制。分层算法非常复杂并且难以实施，这些算法具有与建立和维护层次相关联的开销，并且需要承受与网络监视、路由保持、漫游节点追踪及寻址等等相关联的管理成本。另一种现有技术的协议组是 IETF 内部开发的 MANET 协议序列。但是 MANET 协议组只能低效使用带宽。

图 1 描述了一种现有技术的机制，这种机制需要每个客户机 A - E 110 - 150 保持一个完整树图 125，以便访问那些可以由客户机 120 访问的各个客户机和各个服务器。而这种机制是非常不利的，因为它需要很大的存储器，而这个存储器将会随着网络扩张而扩大。

因此，一种用于互连无线网络的更好协议将会是非常有益的。

发明内容

本发明的一个目的是提供一种改进的无线协议。在这里描述了一种用于产生连接的方法和设备。该方法包括一个广播包括服务器地址的信标的服务器。该方法还包括：接收信标的各个客户机重新广播该信标。结果，接收信标的各个客户机都知道了一条通向服务器的路线。

本发明的其他特征和优点将从附图以及后续详细描述中变得清楚。

附图说明

本发明是借助实例来描述的，但是本发明并不受附图中的图形限制，在这些附图中，相似的参考符号代表相似的部件，并且其中：

图 1 是一个无线连接的现有技术网络的方框图。

图 2 是当前连接结构的一个实施例的网络图。

图 3A 是当前连接结构的另一个实施例的网络图。

图 3B 是当前连接结构的另一个实施例的网络图。

图 4 是可以实现本发明的计算机系统的一个实施例的方框图。

图 5 是可用在本发明之中的客户机和服务器的一个实施例的方框图。

图 6A 是注册信标的一个实施例的概述流程图。

图 6B 是选择一个用于重新广播的信标的一个实施例的流程图。

图 7A 是反向信标的一个实施例的流程图。

图 7B 是对反向信标广播进行定时的一个实施例的流程图。

图 8A 是使用 DRS 来把客户机连接性数据发送到服务器的一个实施例的流程图。

图 8B 是使用源路由来将客户机连接性数据发送到服务器的一个实施例的流程图。

图 9 是包含多个分支的服务器系统的一个实施例的网络图。

图 10 是包含多个服务器的系统的一个实施例的网络图。

图 11 是在群集之间进行漫游的一个实施例的流程图。

具体实施方式

概述

本发明提供了一种可升级的路由选择解决方案，该方案能够有效使用带宽，快速适应于网络拓扑结构与连接性的变化，并且能够自行管理，而且易于部署，这种解决方案能够自动划分网络，以便最优使用那些可用的有线连接，此外这种解决方案易于实现。这种网络结构包括一个或多个有线接入点（服务器），这些接入点同时也是无线网络以及（有线）互联网成员，此外，这种网络结构还包括了作为无线网络成员并且只能经由服务器来访问有线互联网的大量客户机。在一个实施例中，服务器数目受到限制（也许是每一个服务器用于每100个客户机），而客户机则希望能够使用一个为其提供互联网接入、电子邮件等业务的服务器。在一个实施例中，该系统还允许客户机与客户机进行通信。

服务器周期性地将一个信标发送到客户机，这个客户机重新广播该信标。这就使得每个客户机都能确定通向服务器的路径。在一个实施例中，反向信标由客户机发送到服务器。因此，服务器具有通向每个客户机的完整路径。每个客户机都具有一条通向最近邻接节点的路径，并且知道其中哪一条路径是通向服务器的。由此客户机与服务器可以进行通信。在一个实施例中，如果客户机想要经由服务器而与互联网相连，那么它把一个请求发送到上游的邻接客户机，这个请求则要求向前传递到服务器。而服务器同样能将一个消息发送到任何一个客户机。

在另一个实施例中，当客户机想要连接到服务器时，它会经由已知路径而把一个连接请求发送到服务器。这个连接请求包含了通向服务器的已知路径。当服务器接收到该请求时，它会了解到通向发起请求的客户机的路径以及所有的插入节点。服务器使用此信息来对该请求做出响应，并将数据添加到它的路由选择表/客户树图中。

在这个系统中，每个客户机都做出选择，以便成为由单个服务

器提供服务的一个独立客户机组的一部分。其中这些客户机组称为群集。因此，网络自动将其自身划分为多个群集，每一个群集都对应于一个服务器。这样做是非常有益的，因为每个服务器只需对一个客户机组进行寻址。由此优化了群集之间的服务器容量，同时降低了客户机组遭受的响应延迟。

图 2 是当前连接结构的一个实施例的网络图。在一个实施例中，有线网络 210 是互联网。并且在一个实施例中，服务器 220A、220B 是借助有线连接 240 而与有线网络 210 相连的。作为选择，服务器 220A、220B 可以经由另一种高带宽连接而与网络 210 相连。客户机 230A - 230E 可以直接或间接经由连接 250、260 而与服务器 220A - B 相连。在一个实施例中，连接 250、260 是无线连接。而在另一个实施例中，这些连接可以是有线连接或其它类型的连接。在一个实施例中，存在一定数量的第一级客户机 230，这些客户机直接 250 连接到服务器 220。其他客户机 230 则通过一个或多个中间客户机连接到服务器 220。

当服务器 220 广播一个信标时，该信标由所有第一级客户机接收。这个信标用于建立一条从各个客户机通向服务器的路径。第一级客户机是基于直接从服务器接收数据而被定义的。第一级客户机重新广播这个信标数据，并将自身数据附着于该信标数据。由此向第二级客户机指示出：通向服务器的路径包括第一级客户机。以下将会对此进行更详细地描述。

在第一实施例中，所接收信标的链路质量确定了系统是否重新广播这个信标。如果信标质量良好，那么它会被重新广播。否则不会重新广播。在一个实施例中，链路质量是由持续性来确定的，其中持续性即为最后几个路由循环中接收到某个信标的次数。在一个实施例中，对信标所示的通向服务器的路径来说，其链路质量——也就是可靠性，即为信标所示的通向服务器的路径在一个合理时间可用——是在各个循环中接收到信标的时候得到增强的。只要没有在一个循环中接收到信标，那么与路径相关的链路质量将会降低。而信标只在其链

路质量足够高的时候才会发送。

在另一个实施例中为系统确定了重新广播的深度。因此，举例来说，只有客户机与服务器之间存在 5 个或是更少的跳 (hop) 的时候，客户机才可以重新广播。而在另一个实施例中，其他链路质量因素可用于确定是否重新广播信标，这些因素可以是流量拥塞、上游客户机电池状态、管道宽度、后端（也就是服务器）容量、等待时间或其他因素。在每个客户机接收到信标之后，这些客户机都具有了通向服务器的上游客户机地址。在一个实施例中，每个客户机还具有一条通向服务器的路径。然后，经由客户机将反向信标发送给服务器。而反向信标则允许服务器建立一个完整的客户机树图，由此允许服务器访问所有客户机。此外，反向信标向每个客户机告知哪些下游节点可以经由该客户机而对服务器进行访问。每个客户机具有至少一个上游节点，并且可以具有多个下游节点。上游节点是客户机与服务器之间的节点。对第一级客户机来说，只存在一个上游节点，那就是服务器。而对第四级客户机来说，存在四个上游节点，这四个节点定义了通向服务器的客户机路径。下游节点是从某个客户机接收信标的节点，这些节点定义了经由该客户机而通往服务器的路径。因此，举例来说，对客户机 D 来说，在服务器 - A - B - C - D - E - F - G 这个单独分支中，上游节点是 C、B、A，而下游节点则是 E、F、G。

在另一个实施例中，并不需要发送反向信标。不发送反向信标意味着服务器无法开始向客户机发送一个消息。更确切的说，服务器必须等待一个来自客户机的请求。该请求包括一条通向客户机的路径。并且在这个系统中，客户机与客户机进行通信的唯一方法就是经由服务器来发送消息。在某些无线系统中，由于访问服务器——这种访问提供了对于通用互联网的接入——是主要应用，因此这样做也就足够了。

尽管图 2 只显示了数量有限的服务器 220 和客户机 230，但是本领域技术人员应该了解，在本专利的范围内，可以在从服务器 220 开始并且数量几乎无限的跳上提供数量几乎无限的客户机 220。在一个

实施例中，服务器容量确定了可以连接到服务器的客户机数量。因此，举例来说，如果服务器可以处理 10 个通向不同客户机的同时连接，那么可以有多达 100 个的客户机连接到服务器。这就表明：在任何时间，10 个客户机中只有一个客户机能够连接到服务器。由此确保客户机不必等待服务器。根据由功能（例如语音对数据的等待时间）改变的可接受等待时间，服务器必须支持一定数量并具有各种功能的客户机。

图 3A 和 3B 是当前连接结构的两个实施例的网络图。图 3 显示了一个单独的服务器以及作为信标发送并且对应于一个请求的消息组。

服务器在客户机寻找路由的过程中起到了主要作用。在周期性的时间间隔中，服务器产生一个“信标”，该信标广播到服务器侦听范围内的所有客户机。连续信标广播之间的时间间隔定义了一个路由循环。而这个信标则是一个路由分组——也就是一个包含服务器地址的短数据分组。在一个实施例中，该信标包括以下信息：（1）标识由该信标生成的路由循环的序列号，（2）服务器地址（MAC 或 IP），（3）流量监视码（TMC）。以下将会更详细地描述 TMC 的应用。在一个实施例中可以省略 TMC。并且在一个实施例中，服务器地址可以仅仅包含在信标消息的以太网报头或 IP 报头中。

在一个实施例中，服务器可以把一个跳数计数器设定为 0。这个跳数计数器由每个重新广播信标的客户机递增。这就使得接收客户机能够确定从服务器开始存在多少个跳。在一个实施例中，该信标可以只包含消息序列号。而其他所有相关信息则可在消息的以太网等级报头和/或 IP 报头中被捕获。

信标由服务器直接侦听范围内的所有客户机接收。就一个实施例而言，在图 3A 中，这些客户机显示为客户机 A 310 以及 H 330。并且在一个实施例中，存在一种用于确保接收到广播传输的装置。以下将会对此进行更详细的论述。客户机 310、330 是从服务器开始的一个跳，相对于服务器 300 来说，这种客户机都被称作位于第一级。

一旦接收到信标，每个第一级客户机 310、330 都具有一条与服务器 300 相连的路径。在一个实施例中，每个第一级客户机 310、330 具有以下数据：（1）与服务器相连的连接性，（2）获取服务器接入的装置（因为它知道服务器地址并可将传输引导到该地址），（3）服务器 TMC。在一个很短延迟后，每个第一级客户机 310、330 在把自身地址和 TMC 附加于信标之后将会重新广播该信标。在一个实施例中，该延迟是一个随机延迟，这样一来，并非所有第一级客户机都会在相同时间进行广播。在一个实施例中，可以省略 TMC 数据。并且在一个实施例中，在重新广播信标之前，客户机可以仅仅递增所接收信标的跳数计数器。而在另一个实施例中，客户机可以重新广播那些未曾改变的信标。在一个实施例中，重新广播的信标包括：（1）序列号，（2）服务器地址及其 TMC。作为选择，信标可以只包含跳数和/或序列号。

现在，这个信标由所有客户机接收，这些客户机都是从服务器开始的两个跳（第二级客户机）330、360。在一个实施例中，一旦接收到信标，那么每个第二级客户机 315、335 都了解到：（1）它具有与服务器相连的连接性，（2）一个通向邻接上游客户机的明确路由（发出所接收广播的第一级客户机），（3）经由上游第一级客户机而与服务器相连的完整路径，以及（4）服务器和发出所接收广播的第一级客户机的 TMC。在一个实施例中，每个第二级客户机都了解到：（1）它具有与服务器相连的连接性，（2）一条通向邻接上游客户机的明确路由。在一个实施例中，每个第二级客户机了解到经由邻接上游客户机到达服务器 300 的跳数。第二级客户机 315、335 有可能接收到由第一级客户机重新广播的两个或两个以上信标。在这种情况下，它会选择所提供的这两个路由中的一个并且拒绝另一个。在一个实施例中，选择了具有最佳链路质量的路由。如上所述，在一个实施例中，链路质量包含信标持久性。在一个实施例中，它还可以包含其他链路质量因素。而在另一个实施例中，所选路由是一个与收听到的第一个重新广播相对应的路由，因此这个方案可以命名为“首次

收听路径的路由选择”。在另一个实施例中，如下文将要更详细的描述那样，TMC 可用于评估预期的等待时间，并且可以选择等待时间最少的路径。而第一级客户机（假定 A）310 有可能接收到其他第一级客户机中的一个客户机（假定为 H）330 的广播。由于客户机 A 310 是第一级客户机，因此该客户机已经了解到一个通向服务器的路由。一旦检查了它从 H 330 接收的传输的序列号，由于该客户机已经具有一个带有该序列号的当前路由，因此它了解到应该忽略这个路由更新。现在，处于第二级的各个客户机重新广播信标。在一个实施例中，这些客户机是在将其地址以及 TMC 附着于信标之后才开始重新广播的。在一个实施例中，这些客户机是在递增了返回到服务器的路径的跳数之后才开始重新广播的。而在另一个实施例中，这些客户机重新广播了未曾变化的信标。如上所述，可以根据链路质量、接收信标的优先级来选择这个最佳路径或最佳信标，也可基于其他评估来对此加以选择。通过在各个级别上重复这个处理，具有与服务器相连的连接性（也就是说，它可以通过功能性链路而与服务器相连，在这条功能性链路上有可能存在其他服务器）的每个客户机都了解到自身与服务器之间的连接性。在另一个实施例中，每个客户机都了解到通向服务器的一条完整路径。而在另一个实施例中，每个客户机只知道处于通向服务器的路线上的邻接上游客户机。在一个实施例中，客户机只会重新广播那些达到指定等级的信标。因此，举例来说，相对服务器而言，具有超过 10 个跳的客户机不进行重新广播。在这种情况下，如果客户机位于服务器的可接受等待时间范围之外，那么它无法接收到通向服务器的路径。这种情况可以显示给用户，以使用户能够使用一个替换装置或是移动客户机。由于这些系统都是用于无线广播的，因此这种情况等价于超出范围。移动设备可以返回范围以内。由于信标是周期性广播的，因此在无线设备下一次处于信标范围以内的时候，它会再次接收到一条通向服务器的路径。

在一个实施例中，每个客户机都把它所发现的通向服务器的路径存入一个临时存储器。在一个实施例中，每个客户机只将其缺省网

关地址和邻接的上游客户机地址存入存储器。当接收到反向信标时，客户机还了解到所有下游客户机，其中从这些客户机通向服务器的路由都会经过这个特定客户机。在一个实施例中，客户机还将保存这个信息。并且在一个实施例中，上行和下行路径保存在一个临时存储器中。在一个实施例中，临时存储器是一个路由选择表。而在另一个实施例中，临时存储器可以是一个缓存器。需要注意的是，临时存储器的大小约为连接到某个下游客户机的客户机数目以及将客户机连接到服务器的上游客户机的数据。在另一个实施例中，存储器中的数据是通向服务器的实际路径，存储器的大小大约是通向服务器的路径长度（到达服务器的跳数）。与传统的距离矢量协议、链路状态协议或是那些路由选择表长度约为网络节点（客户机）数目的等价协议相比，这个存储器大小是非常小的。举例来说，假设节点密度相同，那么需要存入客户机存储器的路径大小约为 N 的平方根，其中 N 是节点数目。上述方法描述了网络中的节点（客户机）是如何接收与其通向服务器的连接性有关的最新信息，并且还描述了一种到达服务器的装置。

在图 3A 的系统中使用了反向信标，以使服务器能够接收数据，从而建立一条完整（双向）的路由选择路径。在一个实施例中，反向信标是在服务器发送一个伪反向信标的时候发送的，从而发起该反向信标。而在另一个实施例中，反向信标是在客户机想要开始与服务器进行通信时被发送的。

在一个实施例中，客户机对伪反向信标做出响应，当它想要开始通信或是接收到信标时，客户机会启动一个下行路由建立过程（DRS）。DRS 请求服务器在其自身的路由选择表中建立路由。在一个实施例中，客户机节点向其缺省网关发出一个下行路由建立分组，并且要求网关把这个分组转发到服务器。缺省网关是从客户机开始的邻接上游节点。对第一级客户机来说，缺省网关即为服务器。缺省网关是客户机与服务器通信所使用的邻接上游客户机。它可以在每次接收到信标的时候被复位。一旦接收到 DRS 分组，那么缺省网关

将其 IP 地址附加于 DRS 分组，并将其转发到自己的网关，以及建立一个通向下游客户机的路由，其中客户机地址包含在其路由选择表的 DRS 之中。这个过程将会继续，直到分组到达服务器。服务器使用这个路径，以便建立沿着该路径到达客户机的下行路由。在另一个实施例中，反向信标包含一个链路列表，也就是分支中不同客户机之间的联系，而不仅仅发送 IP 地址。以下将会对此进行更为详细的描述。

在另一个实施例中，每个节点周期性启动一个反向信标广播。这个周期即为 KEEPALIVE 周期。在一个实施例中，对周期的开始时间进行抖动，这样一来，并非所有节点都在相同时间发出反向信标。反向信标包括一个源 (From) 地址、一个启动节点地址和一个目的地 (To) 地址，其中目的地 (To) 地址即为节点的缺省网关地址。一旦接收到反向信标，那么节点缺省网关将会把通向启动节点的路由添加到自己的路由选择表中。然后如上所述，在添加了地址之后，节点缺省网关会继续传递反向信标。每个客户机在各个周期都会发送一个单独的反向信标，并且在此期间集中其他反向信标。因此，如果客户机接收到三个反向信标，那么，在客户机发送自身反向信标的时候，它会把一个单独信标发送到缺省网关，其中包含了来自它所接收的三个反向信标的所有数据。

图 3B 描述了一种开始与服务器进行通信的替换方法。当客户机想要开始与服务器进行通信时，为了建立一个 http 连接或是类似连接，它会访问临时存储器来寻求通向服务器的当前路由。举例来说，当前路由可以显示为 $F \rightarrow G \rightarrow H \rightarrow S$ ，其中 F 标记的是尝试与服务器 S 开始通信的客户机。

客户机 F 将一个启动请求 (IR) 发送到客户机 G。在一个实施例中，IR 是一个数据分组，其中除了一个指向服务器 S 以便启动连接的请求之外，还包含有路径 ($F \rightarrow G \rightarrow H \rightarrow S$)。客户机 G 使用 IR 中包含的路径信息来计算应该将这个分组转发到哪个客户机。在这个实例中，在未曾进行任何改变的情况下，客户机 G 把这个分组转发

到客户机 H。然后，客户机 H 将这个分组转发到服务器 S。一旦接收到 IR，由于接收到了路径 $F \rightarrow G \rightarrow H \rightarrow S$ ，因此服务器知道如何重新返回客户机 F。服务器经由路径 $(S \rightarrow H \rightarrow G \rightarrow F)$ 而向客户机 F 做出关于 IR 接收的应答。这时可以建立一个双向连接。

需要强调的是，在路由循环末端，当前作为网络一部分的每个客户机都知道其缺省网关，其中该网关通向服务器。在一个实施例中，客户机还知道处于其下游并且使用该客户机来到达服务器的所有客户机。

在另一个实施例中，客户机知道它的完整路由。举例来说，客户机 X 的路由可以显示为 $(X \rightarrow B \rightarrow L \rightarrow D \rightarrow S)$ 。此外，在一个实施例中，客户机只知道自己的分支，也就是通往服务器的网关、通往服务器的路径以及处于其下游并且使用该客户机来访问服务器的节点。这种情况可以看作是所建议路由协议的实力——在一个网络架构中，客户机尝试与控制有线互联网接入的服务器进行通信，其中对等连接通常并不是必要的。每个客户机所要保持的链路或路径数目越大，协议的执行也就越是复杂和困难，而带宽的浪费也就越发严重。因此，减少带宽以及存储器需要这些方面的优点超出了不使每个客户机都具有包含所有其他客户机的路由选择表的缺点。在一个实施例中，如以下将要详细描述的那样，服务器具有一条通向各个客户机的路径。因此可以经由服务器来建立客户机与客户机的连接。

框图

图 4 是一个计算机系统的实施例，其中该计算机系统可以与本发明一起使用。然而对本领域技术人员来说，其他替换系统或是不同的系统架构也可使用，这一点是显而易见的。

图 4 所述的数据处理系统包括一条总线或其他用以传递信息的内部通信装置 445，以及一个连接到总线 445 用于处理信息的处理器 440。该系统还包括一个随机存取存储器 (RAM) 或是其他易失存储设备 450 (称为存储器)，该存储器与总线 445 相连，以便保存信息和处理器 440 执行的指令。在处理器 440 执行指令的过程中，主存储

器 450 还可用于保存临时变量或是其他中间信息。该系统还包括一个只读存储器 (ROM) 和/或静态存储器设备 420, 这些设备与总线 440 相连, 以便保存静态信息和用于处理器 450 的指令, 并且系统中还包含一个数据存储设备 425, 例如磁盘或光盘以及相应的磁盘驱动器。数据存储设备 425 与总线 445 相连, 以便保存信息和指令。

该系统还可以连接到诸如阴极射线管 (CRT) 或液晶显示器 (LCD) 这种显示设备 470, 这个显示设备经由总线 465 而与总线 445 相连, 以便将信息显示给计算机用户。在一个字符数字输入设备 475 中包含了字符数字和其他按键, 这个设备可以经由总线 465 而与总线 445 相连, 以便将信息和命令选择传递到处理器 440。附加的用户输入设备是光标控制设备 480, 例如鼠标、轨迹球、输入笔或是光标方向键, 该设备经由总线 465 而与总线 445 相连, 以便将方位信息和命令选择传递到处理器 440, 并且在显示设备 470 上控制光标移动。

另一个可选连接到计算机系统 430 的设备是通信设备 490, 用于经由网络访问分布式系统的其他节点。通信设备 490 可以包括多种可以在市场上买到的网络外围设备中的一种, 例如那些用于连接到以太网、令牌环网、互联网或广域网的设备。需要注意的是, 图 4 所述系统中的任何或所有组件以及相关硬件都可用于本发明的不同实施例。

本领域技术人员可以了解, 任何系统结构都可根据某种实施方式而用于不同目的。实施本发明的控制逻辑部分或软件可以保存在主存储器 450、大容量存储设备 425 或者其他可以由处理器 440 本地或远程访问的存储介质中。其他存储介质可以包括软盘、存储卡、闪存或是 CD-ROM 驱动器。

对本领域技术人员来说, 显而易见的是, 这里描述的方法和处理可以作为在主存储器 450 或只读存储器 420 中保存并由处理器 440 执行的软件来实施。这个控制逻辑部分或软件还可以存在于包含计算机可读介质的制造商品中, 其中该介质具有这里所描述的计算机可读程序代码, 大容量存储设备 425 可以读取这些代码, 以使处理器 440

根据这里的方法和教导来进行操作。

本发明的软件还可嵌入在一个手持或便携设备中，其中该设备包括上述计算机硬件组件的一个子集。举例来说，手持设备可以配置为只包含总线 445、处理器 440 以及存储器 450 和/或 425。手持设备也可配置成包括一组按钮或输入信令的组件，由此用户可以从一组可用选项中进行选择。手持设备也可配置成包括一个输出设备，例如将信息显示给手持设备用户的液晶显示器或是显示单元矩阵。常规方法可用于实现这种手持设备。在给出了这里所提供的发明内容的情况下，对本领域技术人员来说，为这种设备实施本发明是显而易见的。

图 5 是可用于本发明的服务器和客户机的一个实施例的方框图。服务器 510 包括一个与客户机 550 进行通信的无线收发信机 515，还包括一个有线网络连接 535，用于与互联网这类有线网络进行通信。

服务器 510 中的信标逻辑部分 520 发出一个信标，其中包含了服务器的服务器 ID528。服务器 ID528 保存在存储器 526 中。该信标由定时器 524 定时。在一个实施例中，周期性发送信标。在一个替换实施例中，可以动态确定信标广播周期。举例来说，该周期可以基于网络稳定性或是网络时效来确定。例如对一个节点经常移动的网络来说，与基于蜂窝电话的网络一样，这个周期可以短于那些节点通常比较稳定的网络的周期，其中后一种网络中的节点类似于通常固定的便携式电脑。信标之间的周期可以小到一秒钟的若干分之几，也可长达好几分钟。在一个实施例中，信标是在每一分钟被发送的。该信标由无线收发信机 515 广播到广播范围内的所有客户机。

在一个实施例中，服务器 510 还从客户机接收一个指示通向各个客户机的路径的反向信标。这个反向信标被用于产生一个路由选择表。而这个路由选择表则为服务器 510 提供了通向所有客户机的已知路径。

在一个实施例中，反向信标不但包含各个客户机地址，还包含了客户机之间的链路。因此，服务器 510 接收一个关于各个分支的路

径结构，而不仅仅接收地址。服务器 510 使用这个路径结构来构造一个客户机树图 540。客户机树图 540 描述了整个网络布局。网络监视逻辑部分 537 可以使用客户机树图 540 来构建一个受服务器 510 控制的群集以及节点之间联系的可视图像。这使得网络监视逻辑部分 537 还能评估连接性和质量。

在一个实施例中，反向信标是由服务器 510 使用伪反向信标逻辑部分 522 发起的。伪反向信标是一个关于第一级客户机的指示符，其中该客户机在预定周期内应该对一个实际反向信标做出响应。通常，任何一级客户机的反向信标都是在从缺省网关接收到开销的时候启动的（如果是第一级客户机的话，那么缺省网关即为服务器）。在一个实施例中，伪反向信标被用于同步定时，这样一来，所有客户机不是在同一时间发送反向信标信号。服务器 510 被设计成了从不同客户机接收请求。在一个实施例中，这些请求包括通向客户机的路径。因此，临时存储器 543 保存了向服务器发出请求的客户机的 IP 地址。

通过经由有线网络连接 535 来访问有线网络，该请求由请求实施逻辑部分 530 使用服务器 510 内部数据来加以实现，该请求也可以通过转发到恰当的客户机而得到实现。当服务器 510 接收到一个关于该请求的响应时，服务器 510 使用来自存储器 543 的路径 546 来将该响应转发到请求数据的客户机 550。通过这种方式，服务器将数据提供给客户机 550，而且服务器还向客户机 550 提供了一个周期性信标，以使客户机 550 能够确定通向服务器 510 的路径。客户机 550 经由网络，直接或通过其他客户机而与服务器 510 相连。其中该网络可以是一个无线网络，也可以是一个交换有线网或是其它类型的网络。客户机 550 包括一个收发信机 555，用于在网络上发送和接收数据。在一个实施例中，无线网络中的收发信机 555 是一个无线收发信机。而在另一个实施例中，客户机 550 经由有线连接而与服务器相连，在这种情况下，描述为无线收发信机 550 的逻辑部分可以是一个有线连接，例如调制解调器、以太网卡、DSL 连接等等。

客户机 550 直接或通过其他客户机而从服务器 510 接收信标，并且把这个数据传递到服务器路径逻辑部分 570。服务器路径逻辑部分 570 则确定是否应该保存以及重新广播该路径。客户机 550 可以接收多个信标，但是通常只会重新广播一个信标。以下将会更详细地讨论多个服务器和多个信标的问题。服务器路径逻辑部分 570 确定将哪一个信标选定为通向服务器的路径。在一个实施例中，这个决定是根据链路质量做出的。信标质量评估逻辑部分 580 对接收到的各个信标的链路质量进行评估。在一个实施例中，信标质量评估 580 追踪客户机 550 接收到的所有信标，并且确定可靠性（一个周期内收听到信标的次数）、链路质量（信噪比）、路径的拥塞电平，还有可能确定其他链路特征。服务器路径逻辑部分 570 对其中每一个特征加以考虑，以便确定所要保存的路由 567。

在一个实施例中，服务器路径逻辑部分 570 对来自先前路由循环中的数据进行评估。信标可靠性，也就是最近发送了 X 次信标之后接收到这个信标的次数，被用于确定这个路由循环中保存的信标是否即为这个信标。在另一个实施例中，服务器路径逻辑部分 570 可以使用一种替换标准来确定是否选择这个特定路径。

在服务器路径逻辑部分 570 识别了信标的时候，与信标相关联的路由数据保存在存储器 565 的路由选择表 567 中。在一个实施例中，预定时段内只有一个通向服务器的单独路径会得到保存以及重新广播。而在另一个实施例中，每个信标都将得到保存，并且信息被添加到了路由选择表 567 中，但只有一个单独信标被重新广播。在另一个实施例中，对接收信标的每个服务器来说，一条路径将被保存，并且信标将会重新广播。由此使得客户机能够具有通向多个服务器的路径，这在无法使用服务器的情况下将是非常有利的。

然后，存储器逻辑部分 575 将客户机自身地址以及 TMC 添加到信标中，并将这个数据回送到重新广播逻辑部分 560。在一个延迟之后，重新广播逻辑部分 560 将这个经过修改的信标传递到无线收发信机 555。在一个实施例中，延迟是由随机延迟逻辑部分 562 产生的。

并且在一个实施例中，对每个客户机 550 来说，延迟预定为某个数值。而在另一个实施例中，对于各个重新广播来说，延迟是由随机延迟逻辑 562 随机产生的。在另一个实施例中，随机延迟逻辑部分 562 可以不存在。

在一个实施例中，客户机 550 还包括反向信标逻辑部分 585。反向信标逻辑部分 585 对上行到达客户机的路径进行广播。在一个实施例中，当从客户机缺省网关接收到反向信标时，反向信标逻辑部分 585 将被触发。反向信标提供了一个所有客户机的列表，其中这些客户机通向服务器的路由将会经过这个把反向信标广播给服务器的客户机。因此，服务器接收到一个反向信标，其中包含了通向所有客户机的路由，并且这个反向信标仅仅是通过第一级客户机发送到服务器的。这就为服务器提供了通向所有客户机的全部路径，并使服务器能够具有一个完整的路由选择表，其中包括了处于其分区中的所有客户机。在一个实施例中，客户只在其路由选择表中包含了朝向服务器的缺省网关，以及相同分支中的下游客户机。因此，与非分割网络相比，存储器和路由开销需要受到了限制。

在另一个实施例中，反向信标可以由一个下行路由建立（DRS）或一个启动请求所取代。以下将对这两个处理进行更为详细的描述。

客户机 550 还包括执行标准客户机功能 590 的逻辑部分。这些客户机功能取决于独立的客户机。举个例子，对一个电话来说，标准的客户机功能 590 应该包括拨号功能、通讯簿功能等等。

信标选择

图 6A 是注册信标的一个实施例的流程图。该处理始于方框 610。在方框 615，服务器确定是否到了再次发出信标的时候。该信标将被周期性发送。如果定时器到期，那么处理将会继续进行到方框 620。否则，该处理将返回方框 615，以便等待定时器到期。

在方框 620 中计算服务器的流量监视码（TMC）。TMC 指示的是服务器提供请求的次数。在一个实施例中，TMC 仅仅是接收请求

与发出响应之间的时间，也就是服务器延迟。在另一个实施例中，TMC 可以表示流量负载，例如在一个时段上接收的请求个数。并且在一个实施例中，TMC 是一个单独的数字，它表示相对负载和延迟。而在另一个实施例中，可以不对 TMC 进行计算。

在方框 625，服务器将信标广播到所有客户机。这个信标包含了服务器自身地址，如果恰当的话，该信标还包含服务器的 TMC。

在方框 630，客户机判定是否接收到一个信标。在一个实施例中，客户机通常监视所有广播，其中可以对指向其自身的信标进行监视，也可对指向所有客户机的信标进行监视。在这种情况下，如果客户机检测到信标，那么它会继续进行到方框 640。否则它会继续等待一个广播。

在方框 640，客户机判定是否在这个循环中接收到其他信标。在一个实施例中，该循环由序列号表示。并且在一个实施例中，每个服务器都具有一个所要浏览的序列。而在一个实施例中，如果客户机从多个服务器接收到多个信标，那么它会保存信标，并且基于通向服务器的跳数以及路径 TMC 来为该请求选择服务器。

在另一个实施例中，该循环由时间标识，如果从多个服务器接收到多个信标，那么会对这些信标加以比较，并且只会保存一个路径。在任何一种情况下，只有一个单独信标会被重新广播。由此将网络自动划分为不同服务器管理的不同群集。如果在这个循环中并没有其他信标，那么该处理将会继续进行到方框 645。在一个实施例中，为了收集到所有可用信标，这个决定是在经过了足够的时间量之后才做出的。在方框 645，客户机保存通向服务器的路径以及 TMC。在一个实施例中，客户机将其缺省网关识别为邻接的上游节点，其中该节点通向服务器。

在方框 650，客户机自身数据被附加于信标之上。在一个实施例中，该数据包括客户机地址和客户机自身的 TMC。而在另一个实施例中，其中只包含了客户机地址。

在方框 655，客户机经过一个延迟之后重新广播这个经过修改的

信标。现在，客户机具有了一条通向服务器的有效路径，并且在其广播范围内将其数据传递到所有其他客户机。此时，客户机可以等待下一个信标。如果客户机已经向服务器发出请求，那么已经存储的路径可用于发送这个请求。以下将对这个处理进行更为详细的描述。

如果在方框 640 发现在一个循环中接收到了一个以上的信标，那么处理将会继续进行到方框 660。

在方框 660 对这些路径进行比较。在一个实施例中，第一级的比较即为客户机与服务器之间的跳数。因此，与第二级以及其他级别相比，优先级将会提供给第一级信标。在一个实施例中，还对路径的 TMC 进行比较。因此，举例来说，使用一个极度繁忙的客户机而具有一个第二级信标还不如使用两个利用不足的客户机来具有一个第三级信标更为有益。因此，在一个实施例中，除了信标等级之外，还对 TMC 加以考虑。在一个实施例中还考虑到了信标可靠性。可靠性表示多长时间收听一次信标。如果未曾有规律的收听到信标，例如信号渐弱，那么该信标有可能是不可靠的。这种可靠性级别也可用于评估所要选择的信标。在一个实施例中，链路质量还可用于评估信标。并且在另一个实施例中，链路质量即为链路信噪比。在选择路径时，可以对所有这些因素以及其他有用因素加以考虑。

在方框 665，基于预定标准来选择“最佳路径”。目前存在几种用于选择信标的可能标准。在一个实施例中，该标准可以是信标可靠性（也就是最后 N 个周期中在这个路径上的接收到信标的次数）。在另一个实施例中，该标准可以是链路强度（举例来说，链路强度可以由路径中各个链路上的信噪比总和来表征）。而在另一个实施例中，选择信标的标准可以是信标所指示路径上的近期拥塞（例如，可以通过 TMC 总和来为沿着始于服务器的路径上的各个节点测量拥塞）。在另一个实施例中，这个标准可以是通向服务器的路径上的跳数。在一个实施例中，这个标准可以包括存在于发出信标的服务器上的回程量，因此，客户机可以选择归属于一个服务器具有更大后端容量的群集。在一个实施例中，这个标准可以是群集中的客户机数

量，这样一来，客户机可以选择加入一个客户机数量最少的群集。对本领域技术人员来说，系统中可以使用任何一种或是全部上述标准以及它们的组合，这一点是显而易见的。并且非常明显，在具有单个服务器或多个服务器的网络环境中，上述标准适用于由客户机进行的信标选择。

然后，该处理继续进行到方框 645。

图 6B 描述了在存在一个以上服务器的情况下对信标进行评估的一个替换实施例。在一个实施例中，单独节点可以从多个服务器接收信标。在这种情况下，在一个实施例中只会保存一个单独信道和一个通向单独服务器的连接。然而在另一个实施例中，通向多个服务器的路径将会得到保存。并且在一个实施例中，举例来说，一个单独的客户机可以从多个服务器接收到多种性质的信标。然后客户机可以建立通向多个客户机中每一个客户机的路径。然而，客户机仅仅设置一条路径作为缺省网关，并且在一个实施例中，客户机仅仅重传一个单独信标。然后，当客户机从下游客户机接收到一个请求时，它会为该请求确定目的服务器，并且还在适当方向上发送该请求。

这种处理始于方框 670。它是一种监视状态。在方框 672，一个信标被接收。这个信标的路由编号连同始发服务器的标识一起被保存。

在方框 674，该处理判定信标是否来自当前缺省网关。如果信标来自当前缺省网关，那么该处理继续进行到方框 676。在方框 676，当前缺省网关上的时戳将会更新。在方框 678，当前缺省网关的链路质量数据也会更新。需要注意的是，该信标是一个将要重新广播的信标，除非是在重新广播的时间之前，否则将会建立一个新的缺省网关。在这种情况下，新的缺省网关即为重新广播的信标。然后，该处理返回方框 672，以便等待接收另一个信标。

如果在方框 674 发现信标并非来自缺省网关，那么该处理继续进行到方框 680。

在方框 680，该处理确定新信标的链路质量是否优于当前缺省网

关的链路质量。如果不是的话，该处理将会继续进行到方框 682。

在方框 682，该处理确定信标指示的路径是否已处于路径数据库中。在一个实施例中，经由一个信标接收的所有路径都会得到保存。在一个实施例中，该数据库周期性清除那些未曾在预定时段内接收到的信标。

如果没有接收到路径，那么该处理继续进行到方框 684。在方框 684，新路径被存入数据库。在一个实施例中，新路径将被保存，其中包括到达服务器的跳数以及与信标相关联的其他参数。在方框 686，信标链路质量连同时戳一起被保存。如果该路径已经存在于数据库中，那么在方框 684，该处理直接进行到方框 686，在这里关于该路径的链路质量将会更新。在一个实施例中，在方框 686，如果信标来自另一个服务器，那么由该信标标识的路径将被设置为一个替换的缺省网关。如果无法使用缺省网关连接的服务器，那么这将允许节点切换到替换服务器。然后，该处理返回到方框 672，以便等待接收一个新的信标。

在方框 680，如果发现链路质量优于当前保存的缺省网关链路质量，那么该处理继续进行到方框 688。

在方框 688，信标指示的新路由是在路由选择表中设置的。

在方框 690 建立了新的缺省网关。该处理然后返回到方框 672。需要注意的是，新的缺省网关是由客户机在下一次重新广播信标时被重新广播的。如果接收到同一缺省网关的信标，那么缺省网关将在客户机下一次重新广播信标的时候被重新广播。

图 7A 是一个反向信标实施例的流程图。该处理始于方框 705。在一个实施例中，该处理经由一个伪反向信标启动，并由一个服务器发起。并且在一个实施例中，处于某一级的客户机并不重新广播它们的信标。这种情况出现在等待时间长到了无法接受的那些等级，例如从服务器开始的四十个客户机，其中等待时间达到了一分钟以上。作为选择，这种情况有可能是在经过一个预定时段之后才发生，其中这个预定时段将被计算，以使信标能够到达所有的客户机。

在方框 710，最低级的第一客户机将一个包含其地址的反向信标发送到其缺省网关。而反向信标则指向服务器。在一个实施例中，客户机对这个针对缺省网关的数据进行广播。因此，接收消息的所有其他节点都会丢弃这个数据，但是实际上，缺省网关将会评估这个消息。在一个实施例中，反向信标是特殊的消息，它是针对一个特殊端口而被发送。该端口将消息识别为反向信标。在一个实施例中，该消息由上游客户机，也就是缺省网关来进行应答。

在方框 712，缺省网关或当前节点接收反向信标。

在方框 715，接收到反向信标的节点把通向下游客户机的链路添加到其路由选择表中。在一个实施例中首先对该链路进行验证。对每个下游客户机来说，可以通过发送一个消息和接收一个应答来完成这个操作。因此可以从客户机访问客户机所在树图中的向下分支。

在方框 720，该处理确定当前节点是否即为目标服务器。如果当前节点不是目标服务器，那么该处理返回到方框 725。

在方框 725，节点将反向信标发送到缺省网关。反向信标包含客户机自身地址以及通向下游客户机的链路，其中该客户机可以经由这条链路而被访问，这个反向信标也是针对于服务器的。实际上，这就构成了当前客户机下方的网络部分的一个完整网络图。然后，该处理返回方框 712，邻接节点接收反向信标，而该处理则继续进行。在一个实施例中，每个客户机都会收集那些在一个循环中接收自下游客户机的反向信标，并且每个客户机将会提出组合列表。这样一来，连续提高通路的客户机发送越来越大的消息，而在每个反向信标周期中，每个客户机仍只发送一个反向信标。在另一个实施例中，客户机将会在发送消息之前等待一个预定时段。并非所有下游客户机都会发送其反向信标。然而，附加数据还包含在后续的反向信标中。

在方框 720，如果节点是服务器，那么该处理继续进行到方框 730。在方框 730，服务器把来源于反向信标的数据添加到路由选择表中。服务器从所有第一级客户机中接收反向信标。在一个实施例中，服务器构建所有下游客户机的一个客户机树图。客户机树图是这

样一种构造，它显示了由某个客户机提供服务的整个群集中的节点间连接。这个客户机树图使用来自反向信标的链路数据而构建这个树图。需要注意的是，每个分支只知道其自身，然而服务器是从所有分支接收反向信标的。换句话说，客户机知道完整的下游树图，并且只具有一个上游客户机。服务器也知道完整的下游客户机树图。

在方框 735，服务器保存客户机树图。这时，服务器知道了通向所有已接收其信标的客户机的路径。这就允许经由服务器来进行客户机与客户机之间的通信。它还允许服务器在其群集中对任何客户机进行寻址，而不需要等待一个来自客户机的请求。然后，该处理在方框 740 结束。

图 7B 是反向信标定时的一个实施例的流程图。该处理始于方框 750。在方框 755，客户机 A 接收一个反向信标广播（RBB）。

在方框 760，该处理确定 RBB 针对的是否为客户机 A。如果是的话，那么该处理继续进行到方框 792。在方框 792，RBB 中的链路——如上所述，该链路由下游客户机收集——添加到客户机 A 的缓存器中。这个缓存器被用于构建 RBB 发送的上行数据流。

在方框 794，通向发起 RBB 的下游客户机的连接将会刷新。在一个实施例中，客户机地址添加到客户机 A 的路由选择表中。然后，该处理继续进行到方框 784，其中系统等待发送其反向信标广播，其中包含了这个链路。需要注意的是，每次当客户机 A 接收到新的 RBB 时，方框 755、760、792 以及 794 中显示的处理才会出现。

在方框 760，如果发现 RBB 针对的并不是客户机 A，那么该处理继续到方框 765。

在方框 765，该处理确定 RBB 是否由客户机 A 的缺省网关所发送。如果不是的话，该处理继续进行到方框 770，并且将在不进行任何操作的情况下丢弃 RBB。然后，该处理结束于方框 796。

在方框 765，如果发现 RBB 由 A 的缺省网关所发起，那么该处理继续进行到方框 780。

在方框 780，时钟被复位。这种开销 RBB 被用于对客户机 A 的时钟进行同步。执行时钟同步的好处在于：涉及网络改变的信息比其他方式更快的上传到服务器。由此减少了协议的收敛时间，这一点是非常有益的。

在方框 782，客户机 A 将发送反向信标广播的时间设定为 $KEEPALIVE-\Delta$ 。Δ 的值表示发送反向信标之前的变化时间。在来自客户机 A 的缺省网关的开销 RBB 启动定时器之前，客户机 A 不会发起一个反向信标广播。

在方框 784，该处理确定是否到了客户机发送 RBB 的时间。如果不是的话，该处理继续进行到方框 786。在方框 786，下游 RBB 被收集。这个处理是相对于方框 792 和 794 来进行描述的。客户机 A 继续接收 RBB 并且建立连接，以及将链路添加到其缓存器中。

在方框 784，如果定时器到期，也就是到了发送 EBB 的时间，那么该处理将会继续进行到方框 788。在方框 788，客户机 A 把针对其缺省网关的 RBB 连同传递 RBB 的请求一起发送到服务器。然后，该处理结束于方框 796。

在一个实施例中，第一级客户机从服务器接收到一个伪反向信标广播，以便启动其时钟。并且在另一个实施例中，每个客户机的 KEEPALIVE 周期由一个很小的抖动因素所改变，以使所有同级客户机免于在同一时间进行广播。这样一来，反向信标是通过客户机上传的，每个客户机只在单个循环中发送一个单独的反向信标广播。循环长度可以随意设定。在一个实施例中，反向信标循环对应于信标循环长度。并且在另一个实施例中，反向信标循环可以设定为另一个值。举例来说，在一个非常稳定的系统中，反向信标循环可以是一个小时，而在一个不稳定的系统中，反向信标循环有可能远远小于一分钟。

使用替换方法来建立服务器到客户机的路径

图 8A 是使用下行路由建立过程 (DRS) 或是启动请求 (IR) 来将客户机连接性数据发送到服务器的一个实施例的流程图。该处理始

于方框 805。在方框 810，客户机接收到一个信标。以上已就图 6 和图 7 对信标选择处理进行了描述。客户机具有一个选定缺省网关，并且具有一条通向它所选择的服务器的路径。

在方框 815，客户机将一个分组发送到它的缺省网关，这个分组针对的是服务器，其中包含了它的完整路径。如果这是一种 DRS 方法，那么这个操作将是响应于接收到一个信标而发生的。在一个实施例中，如果将客户机设置成使用 DRS 来传递数据，那么在接收到信标之后的一个预定时段中，客户机会发送这个针对服务器的 DRS 分组。如果这是一种 IR 方法，那么这个操作将会响应于客户机具有一个关于服务器的请求而发生。该请求包含了通向客户机的完整路径，如果恰当的话，在这个请求中还包含任何一个请求。

在方框 820，上游节点接收分组。

在方框 825，这个节点对其通向客户机的路由进行刷新，其中节点从该客户机接收分组，如果客户机仍未处于其路由选择表中，那么节点建立一条新链路。客户机还会添加那些包含在从客户机开始的完整路径中的任何中间客户机。

在方框 830，该处理确定当前节点是否为服务器。如果不是的话，该处理继续进行到方框 835。

在方框 835，节点将分组转送到它的缺省网关，在这个分组中包含了其自身信息。这个处理将会重复，直到服务器接收到分组。如果该节点即为服务器，那么处理继续进行到方框 840。

在方框 840，服务器创建一个客户机树图，该树图表示其群集结构。如果分组是一个 IR 分组，并且在这个分组中存在客户机请求，那么服务器还会执行任何恰当的操作。这时，服务器具有一条通向产生该分组的客户机的完整路径。随着这种分组的接收，越来越多的路由选择表可以被填充。该处理结束于方框 845。

图 8B 是使用源路由来将客户机连接性数据发送到服务器的一个实施例的流程图。

该处理始于方框 850。在方框 855，客户机接收其信标。这个信

标包含了服务器到客户机的完整路径，其中包含了所有中间节点。

在方框 860，客户机判定它具有一个关于服务器的请求。在一个实施例中，这个请求是一个标准请求。

在方框 865，客户机将一个标准的数据分组发送到其缺省网关。在这个数据分组中包含了客户机到服务器的完整路由。

在方框 870，客户机的缺省网关接收数据分组，并对其进行检查，以便确定其自身是否即为目的地。

如果当前节点即为数据分组的最终目的地，那么该处理继续进行到方框 875。否则，当前节点检查数据分组并将其发送到路由中指示并作为当前节点缺省网关的下一个节点。然后，该处理返回到方框 870。

在方框 875，最终目的地接收到数据分组，其中，最终目的地通常是服务器。如果恰当的话，最终目的地还对数据分组进行处理。数据分组的处理在本领域是已知的。在一个实施例中，目的节点返回客户机请求的数据。如果恰当的话，也执行客户机请求的其他处理。然后，该处理结束于方框 880。

需要注意的是，在这个系统中，中间客户机并没有建立路由选择表，而是添加了数据分组指示的链路。因此，每个独立数据分组必须包含通向服务器的完整路由。在一个实施例中，服务器未曾建立一个客户机树图和路由选择表。而在另一个实施例中，服务器无需建立上述这些信息，因为应答路径是在各个数据分组中规定的。

图 9 是一个描述包含单个服务器 S 的系统分支划分的框图。可以看出，23 个客户机直接或间接的与服务器相连。分支 910 只了解同一分支上处于其下游的其他成员和它的上游网关。因此，举例来说，客户机 C10 具有一条通向客户机 C11 - C13 的路径，而不具有通向其他客户机的路径。在重传信标时，这样做将会显著降低开销，因为第一级客户机 C10 只会接收到 3 个反向信标消息，而不是 23 个。这种情况同样适用于其他分支。只有服务器知道还有其他分支存在，并且只有服务器具有一条通向所有分支的路径。

广播范围 920 是相对于客户机 C17 来描述的。可以看出，客户机 C17 的广播由多个客户机接收，其中包括了处于其他分支的某些客户机。然而，对那些其他分支的客户机，也就是 C23 和 C14 来说，它们忽略了来自 C17 的广播，因为它们已经接收到一个通向服务器的更短路径。而将 C17 作为缺省网关的三个客户机则会受到该广播的影响。

路由选择表 930、940 还描述了将系统划分成独立分支所造成的通信简化。可以看出，在路由选择表 930 中，终端客户机 C3（不具有与之相连的下游客户机）只具有一条单独的记录。

除了缺省网关 C1 之外，上游客户机 C2 的路由选择表 940 中还包括所有下游客户机，也就是 C3 - C6。对每个下游客户机而言，C2 在其路由选择表 940 中具有一个第一跳地址。

因此，在所示实例中，客户机上的最大路由选择表是 C1 的表格。C1 把服务器 S 作为缺省网关。然而，C1 还具有通向客户机 C2 - C6 的下行路由。因此包括缺省网关在内，最大路由选择表包含 6 个地址。与一个各个客户机都具有一条对应于其他各个客户机的记录的标准系统相比较，也就是说，在这个实例中，对应于各个客户机的各个路由选择表都具有 23 条记录。通过显著缩小客户机上的路由选择表大小，系统中节省了相当多的带宽。

图 10 是一个多服务器系统的实施例的框图。在这个框图中包含了多个服务器 X、Y 以及 Z。每个服务器 X、Y、Z 都发出一个信标，其中 X 的信标用 X1 标记，Y 的信标用 Y1 标记，Z 的信标用 Z1 标记。每个客户机接收一个或多个信标，它们重新广播一个经过修改的信标，其中包含了由客户机附加的数据。尽管这些广播显示为指向客户机的单个箭头，但是广播通常指向范围内的所有客户机。

需要注意的是，对某些客户机，例如客户机 I 来说，它们从多个服务器接收信标。在一个实施例中，每个客户机都对重新广播的单个信标加以确定。然而在一个实施例中，客户机会保存来自各个服务器的信标。在一个实施例中，如果首次选择的路径并不运作，那么上

述处理将使具有可用替换路径的客户机能够使用替换路径。对每个客户机来说，图 10 还描述了 S 所示的一个主服务器和一个如 G 所示的缺省网关。在这个图中描述了如何将所有节点分为独立群集，其中每个群集都与一个服务器相关联。由此，把数据发送到任何一个服务器的客户机数目将会显著减少。这就降低了路由开销，并使系统能够进行自我优化，而且提高了可用性。

网络能将其自身自动划分为群集，以便在服务器上最优使用可用回程，网络的这种属性使网络扩缩性更强并且更易于管理。网络扩缩性更强是因为每个群集都非常有效地成为一个独立的路由域。单独群集中的客户机无需与其他群集中的客户机交换路由分组。这意味着带宽方面的路由开销将会得到改善，由此更有效地使用网络容量。同时，创建群集意味着每个服务器只对自身群集中的少量客户机负责，因此，与管理和协调客户机相关联的网络管理问题将会分解成每个服务器都可处理的比特大小的字节片（chunk）。如果服务器只具有受限的处理能力，那么这种操作将是非常有益的。在一个实施例中，这个架构中的每个服务器都执行以下一个或多个功能：客户机验证、计费、网络监视、带宽供应以及为群集中的客户机确保一定级别的服务质量（QoS）。

客户机验证是这样一种处理，借助于此，服务器对加入其群集的新客户机的身份和访问权限进行验证。计费包括追踪各个客户机的网络使用情况。举例来说，为了进行计费，服务器可以保持一张关于其客户机的表格以及这些客户机在最后的计费时段中上传/下载的数据量。网络监视包括实时维护一张网络映射表，其中显示了当前处于该群集中的客户机以及它们的相互连接。服务器维护的客户机树图/客户机数据库可用于产生一张映射表，其中实时显示了网络状态。而对监视网络状态与活动以及流入网络的流量而言，这个表是非常有用的，并且对管理用户网络的服务提供商来说，这个表格也是非常有价值的。该表格还可以在故障隔离中起到作用——如果节点将要崩溃或是链路出现故障，由此导致与一个或多个用户有关的服务受到损失，

那么服务提供商可以参考映射表来定位故障源头。

很容易看出，随着服务器管理的网络规模的增长，由服务器执行的这些功能中的每一个都在复杂性方面有所增长。这里给出的网络结构和路由协议的一个优点在于：群集的创建导致每个服务器工作量的降低。举例来说，在一种具有一个服务器和 100 个客户机的情况下，如果将附加服务器引入网络，那么每个服务器的工作量将会减半，因为现在每个服务器只管理 50 个客户机。当然，在这里假设服务器具有相同的容量。由于每个独立客户机都对所要选择的服务器进行评估，如果一个服务器具有大于另一个服务器的容量，那么其中一个群集将会大于另一个群集。

这种网络架构的另一个优点在于：它支持用户在覆盖区域内进行漫游。如果可以移动的客户机系统在单个群集区域周围漫游，那么它与服务器之间的路径（由此它的路由选择表）将会在路由周期所设定的时间标度上发生改变。如果路由周期足够小，假设为 1 秒，那么就如经由下载速度测量或者传输等待时间等参数的测量所显示的那样，路由改变不会造成性能上的显著降低。

图 11 是客户机在群集之间漫游的一个实施例的流程图。该处理始于方框 1110，也就是当移动客户机从一个群集移动到另一个群集的时候，假设移动客户机是从服务器 X 管理的群集 X 移动到由服务器 Y 管理的群集 Y。

在方框 1120，客户机停止从服务器 X 接收信标。这种情况发生在当客户机超出服务器 X 的信标范围时。在一个实施例中，客户机实际上并不需要停止接收信标。更确切的说，来自服务器 X 的信标的质量可能不足以满足使用的需要。

在方框 1130，客户机开始直接或经由群集 Y 中的客户机中继地从服务器 Y 接收信标。客户机有可能已经预先从群集 Y 接收信标，但是仍把服务器 X 选作其通向网络的主路由。因此，由于客户机此时已不再从服务器 X 接收信标，因而服务器 Y 将被选为主服务器。

在方框 1140，通向服务器 X 的路由以及通向群集 X 中所有客户

机的路由终止或是失效。在一个实施例中，这种情况可能发生在在一个预定时段之后，例如，这种情况发生在三个信标循环之后，其中并未在这几个循环中接收到来自服务器 X 的信标。在一个实施例中，当路由失效时，客户机可以从其路由选择表中删除这些路由。并且在在一个实施例中，为了评估链路质量，当路由失效之后，客户机可以在一个时段中仍然保存这些路由。

在方框 1150，客户机根据处于群集 Y 时所接收的路由分组来产生一个新的路由选择表。如通过路由协议所确定的那样，这个路由选择表包含了通向服务器 Y 以及群集 Y 中某些客户机的直接和间接路由。上文详细描述了通过捕获数据来填充路由选择表的处理。这时，客户机再次具有了通向服务器的连接。然后，该处理终止于方框 1160。上述处理无需按照这个顺序进行。相反，在客户机从一个把服务器 X 作为主选定服务器的区域移动到一个并非如此的区域之后，各个步骤可能以任何顺序出现。

这样一来，当客户机从一个群集移动到另一个群集时，路由协议将会导致客户机中产生一个新的路由选择表。由于所述步骤序列可以很快进行，因此用户不会体验到由此产生的中断。路由协议由此可以用一种几乎完全与蜂窝电话网络中的通用方式相同的方式来支持漫游，其中，关键的不同之处在于：在这里，这种漫游是在一个多跳数据网络中得到的。

客户机从一个群集漫游到另一个群集是这样一种情况，其中一个群集的成员可能会发生改变。另一种有可能出现这种漫游的情况出现在把新服务器添加到网络中的时候。在这种情况下，所添加的服务器启动其自身群集，并且某些旧群集的客户机成员现在将会与新服务器产生联系，由此产生两个群集，其中只有一个群集是先前存在的。响应于新服务器或回程点的添加，网络通过创建新群集以及分割旧群集来对其自身重新进行配置，在这个过程中使用的方法就是本路由协议的一个特征，该特征具有如下优点：它使网络能够对自身进行快速调整，以便最优使用可用回程。回程是与客户机通信的服务器的带宽

和数据处理性能。对那些试图提高网络容量的服务供应商来说，这个特征具有非常显著的好处。

在另一种情况下，群集成员有可能发生改变，那就是一个客户机无线电设备连接到一个有线连接，由此同时具备了有线和无线连接的情况。这意味着所述客户机将会变成一台服务器，并且可以开始借助信标来通告其服务。由此将会以一种类似于先前段落中所描述的方法而在新服务器周围形成一个群集。这种情况的用途在于：举例来说，处于密集市区并具有大量用户的服务提供商可以通过为一个或多个用户另外提供 DSL 线路或是 T1 线路这种有线连接而选择把用户升级为服务器。然后网络对自身进行调整，以便使用新近引入的有线回程。对服务供应商来说，这是一种增加网络容量的便利方法。对那些为用户整体提供无限互联网接入的服务提供商来说，通过引入新服务器/接入点或是通过在其归属提供有线连接而将客户机升级为服务器，由此无缝添加附加容量，这种能力显然是非常有益的。与那些只要有需要就在网络中添加新回程点的要求相比，这种能力为服务提供商提供了一种更易于提高网络容量的方法。

在以上描述中，本发明是参考特定的示范性实施例来描述的，然而很明显，各种修改和变化可以应用于本发明，而不脱离附加权利要求中所述的本发明的较宽实质和范围。因此，说明书及附图仅仅看作是说明性的，它们并不具有限制意义。

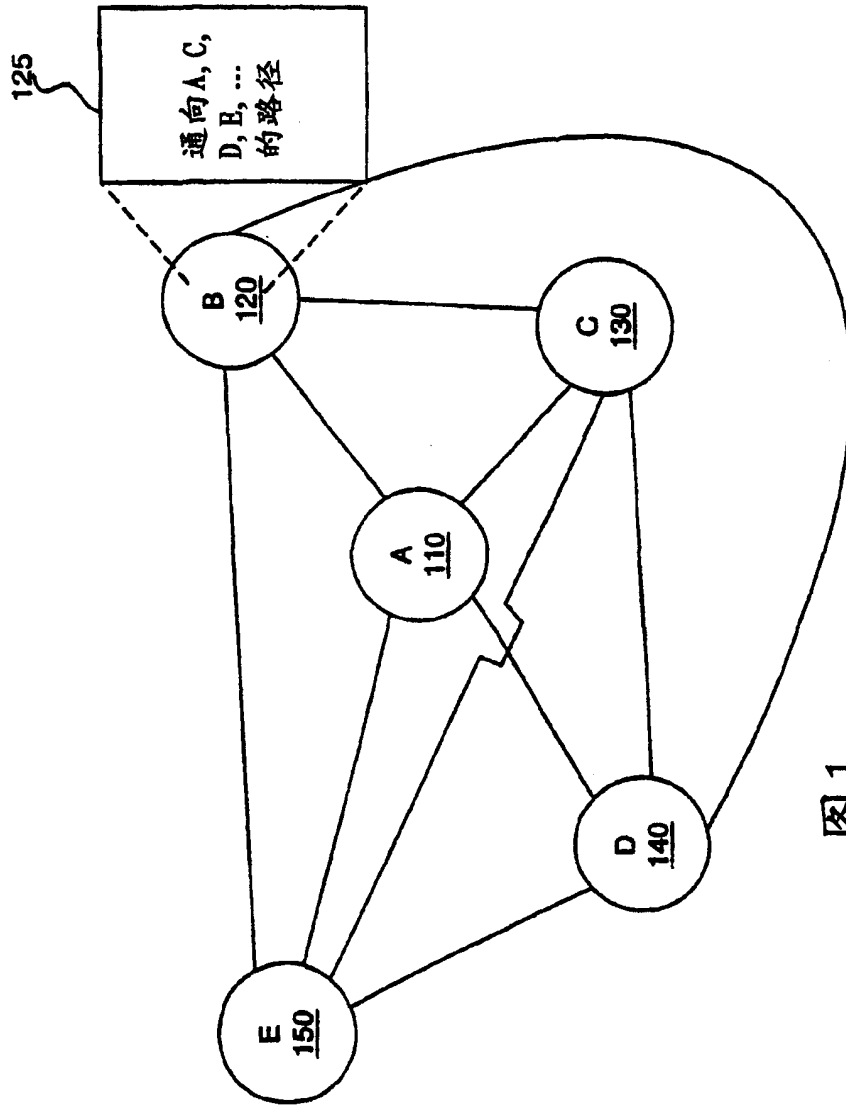


图1
(现有技术)

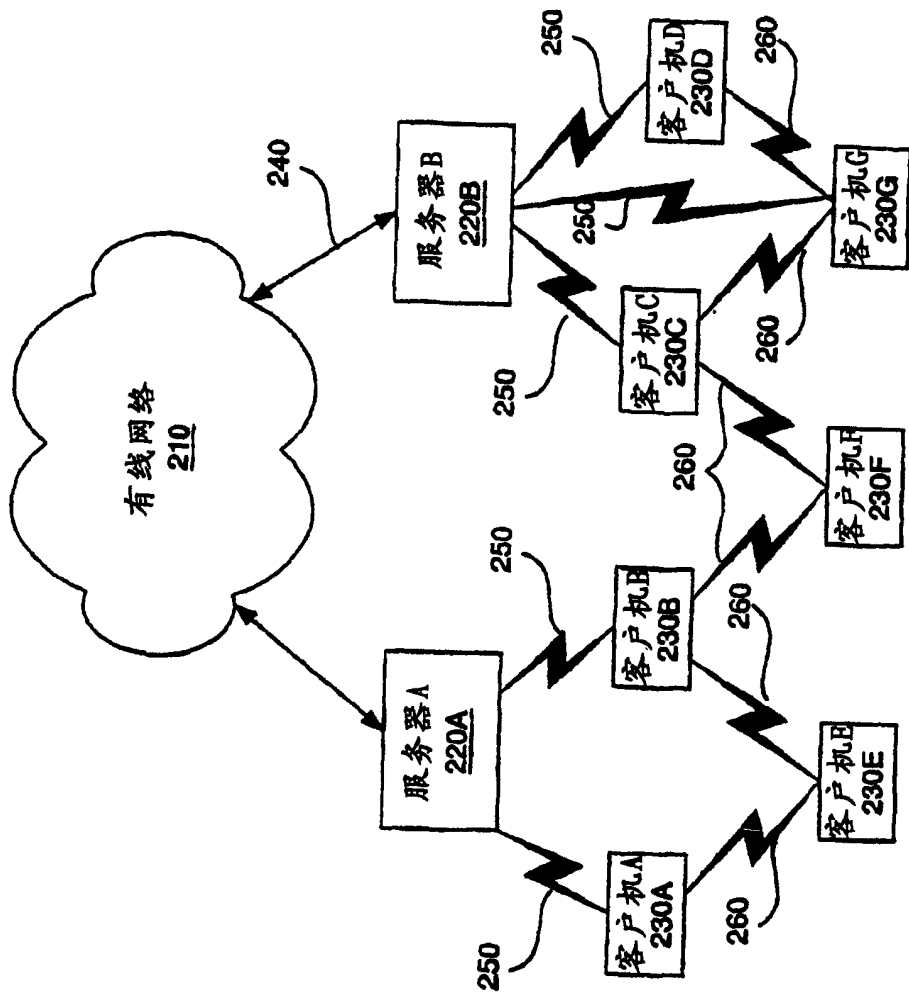


图2

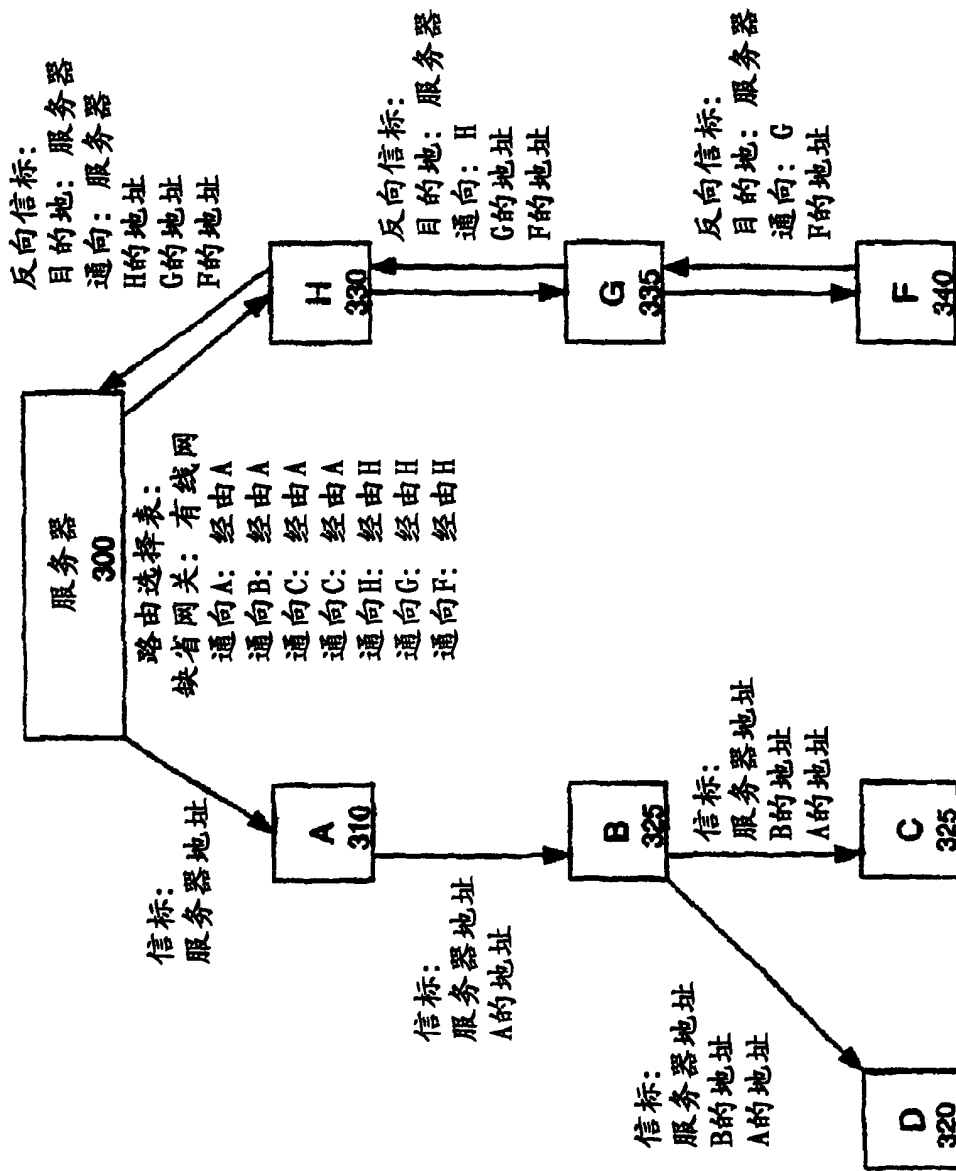


图 3A

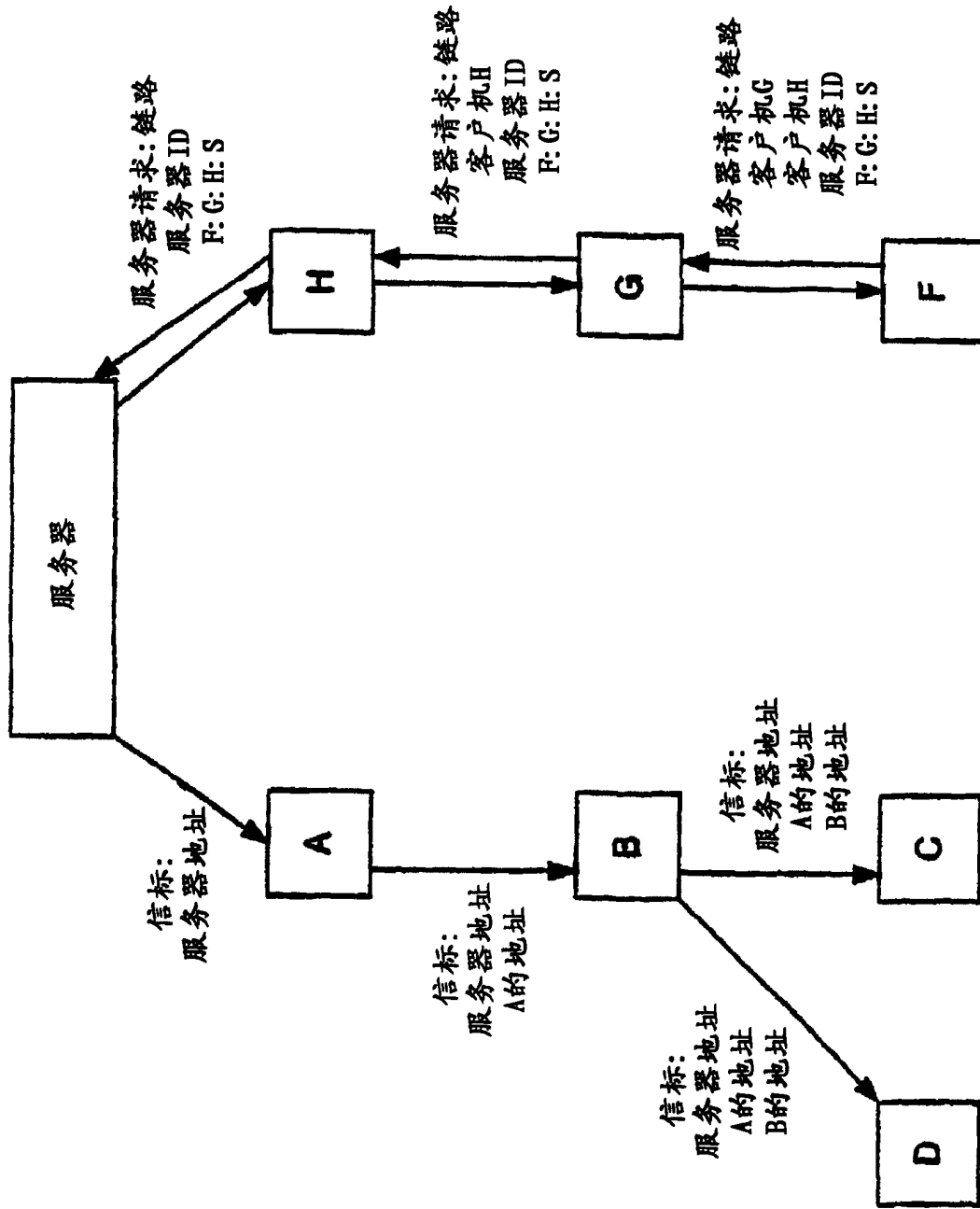


图 3B

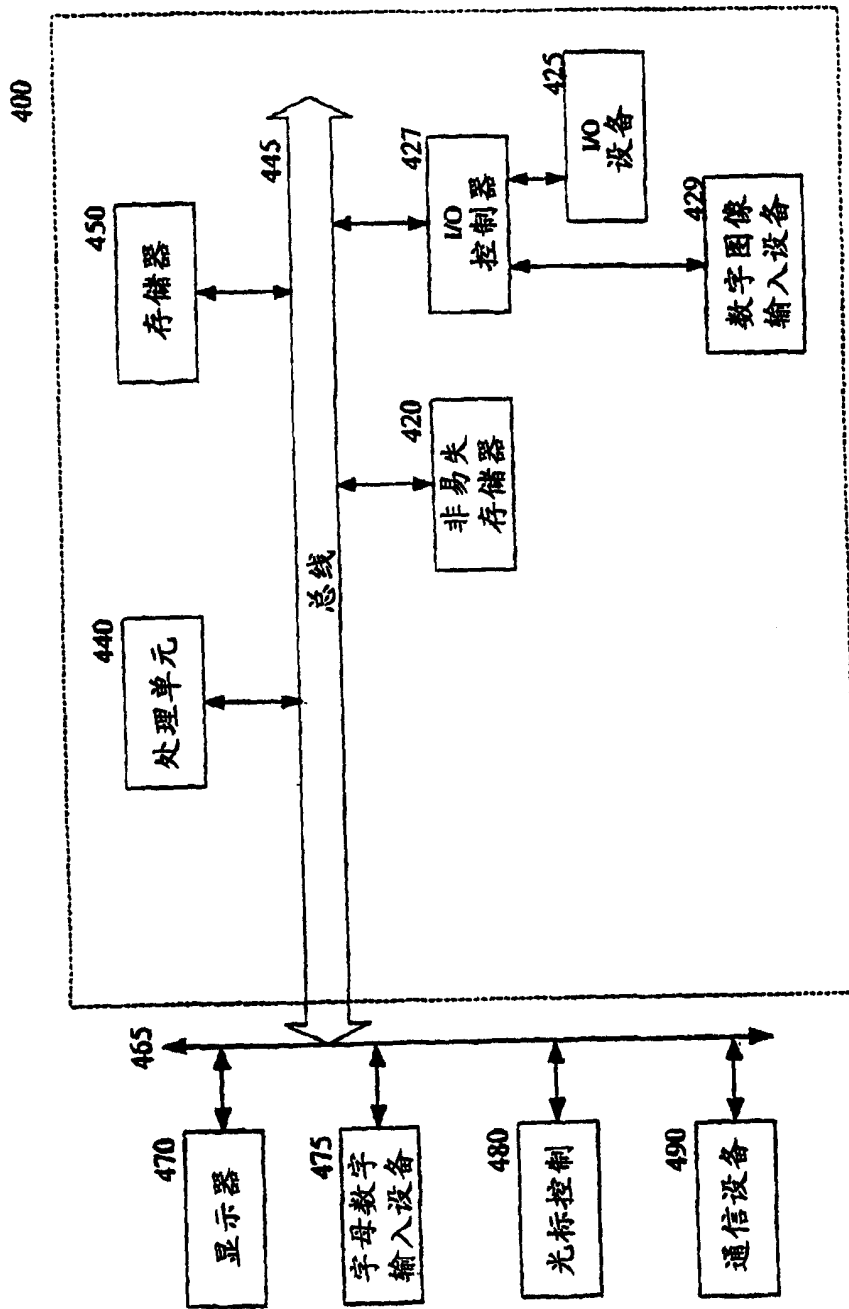


图4

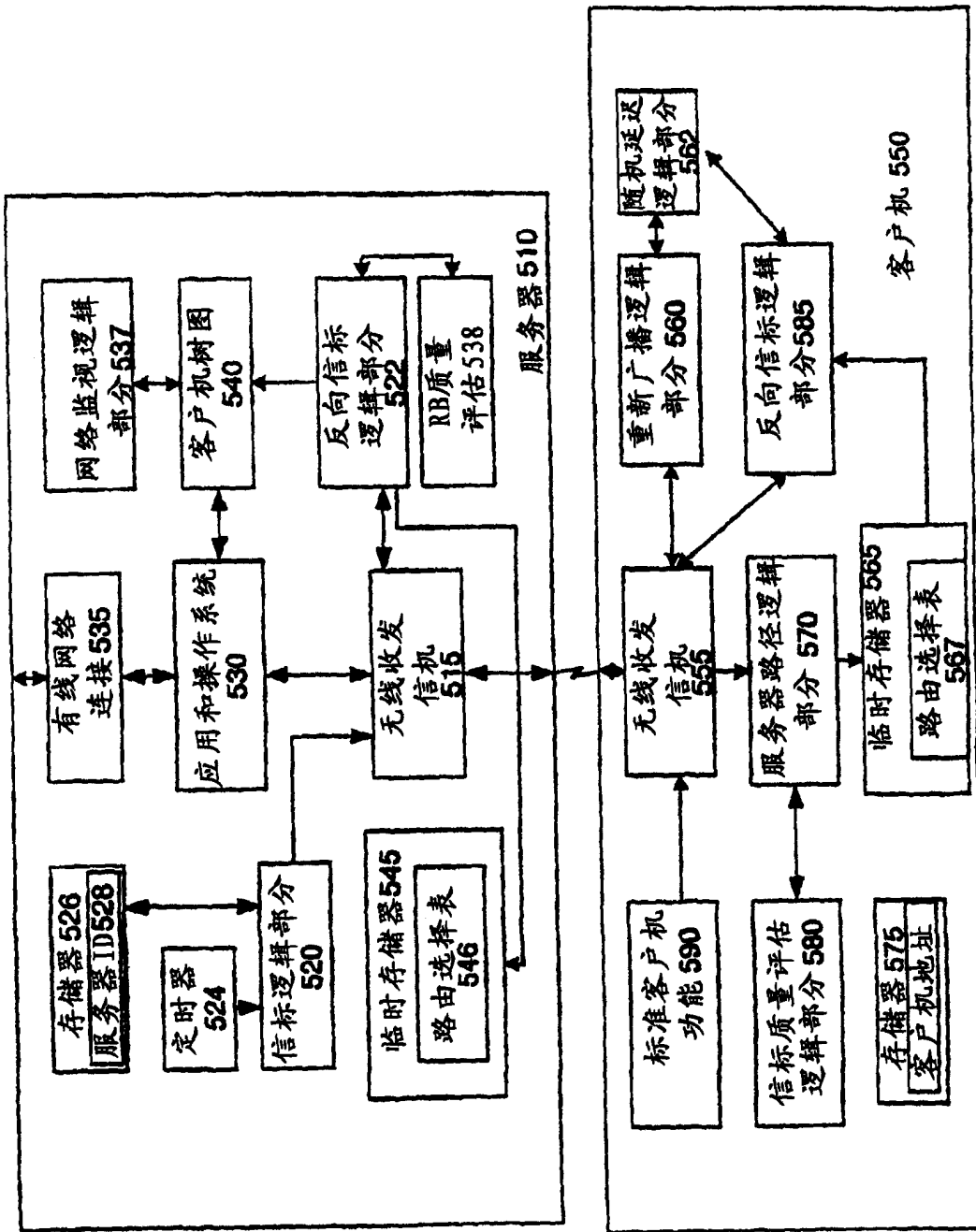


图 5

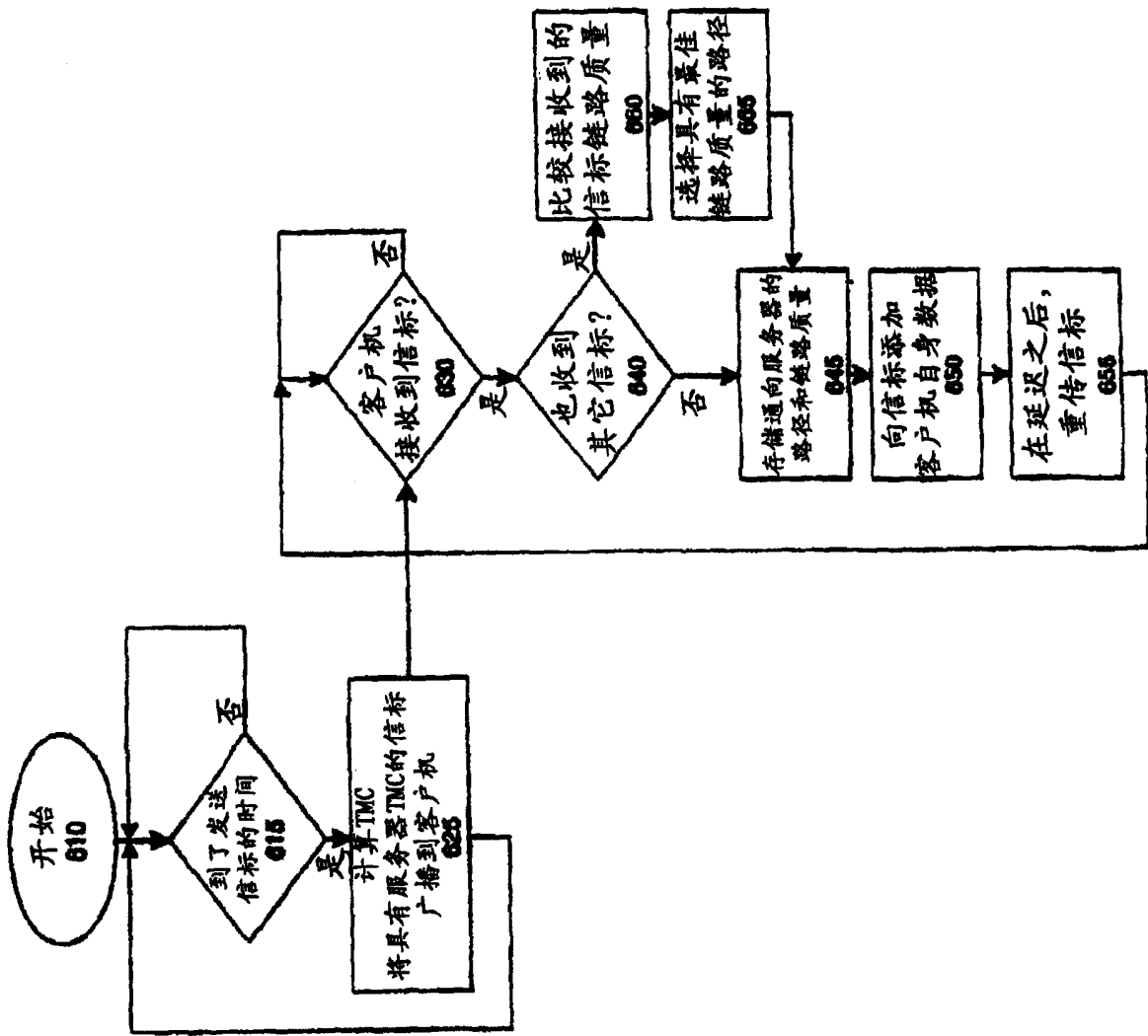


图 6A

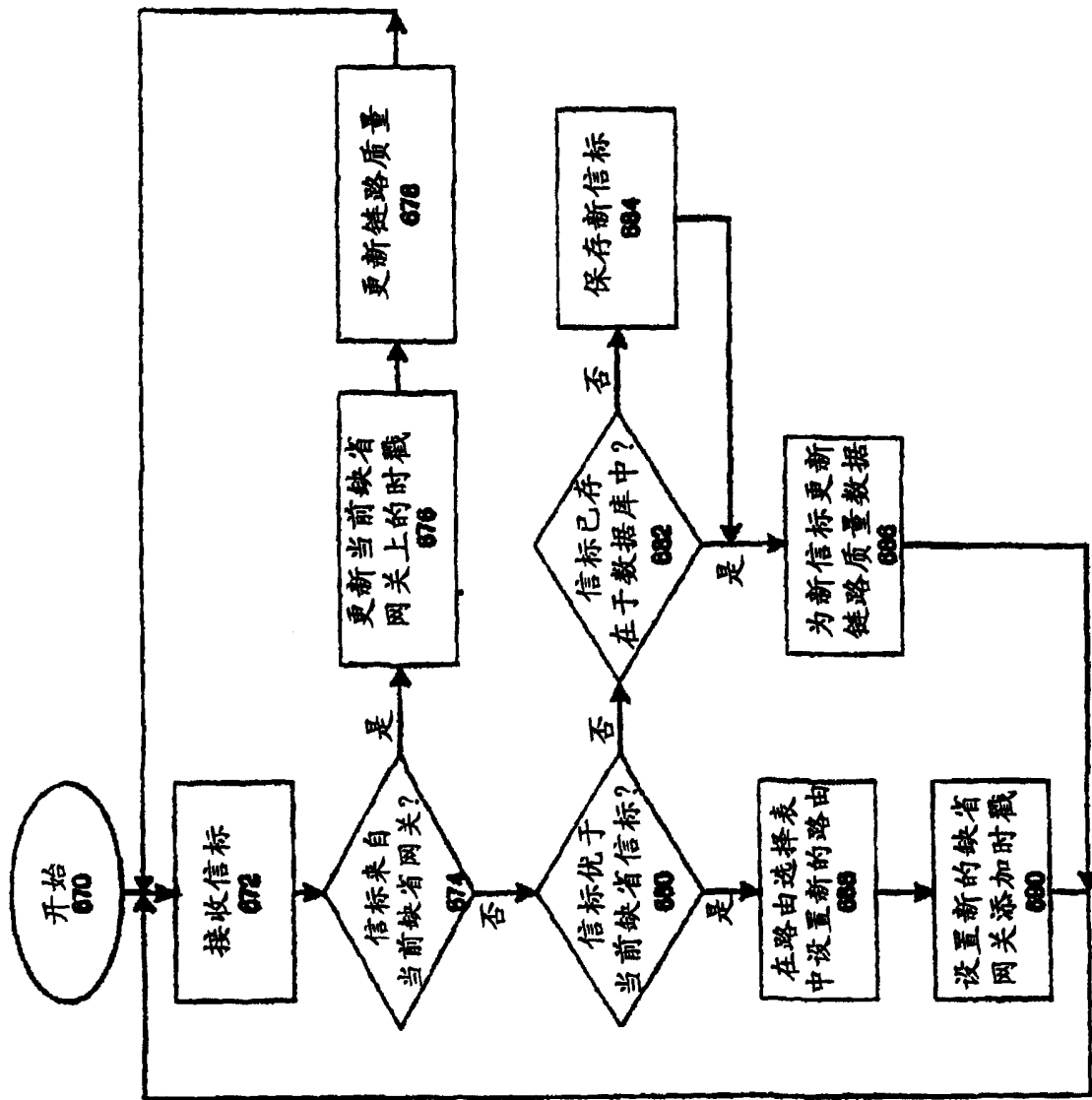


图 6B

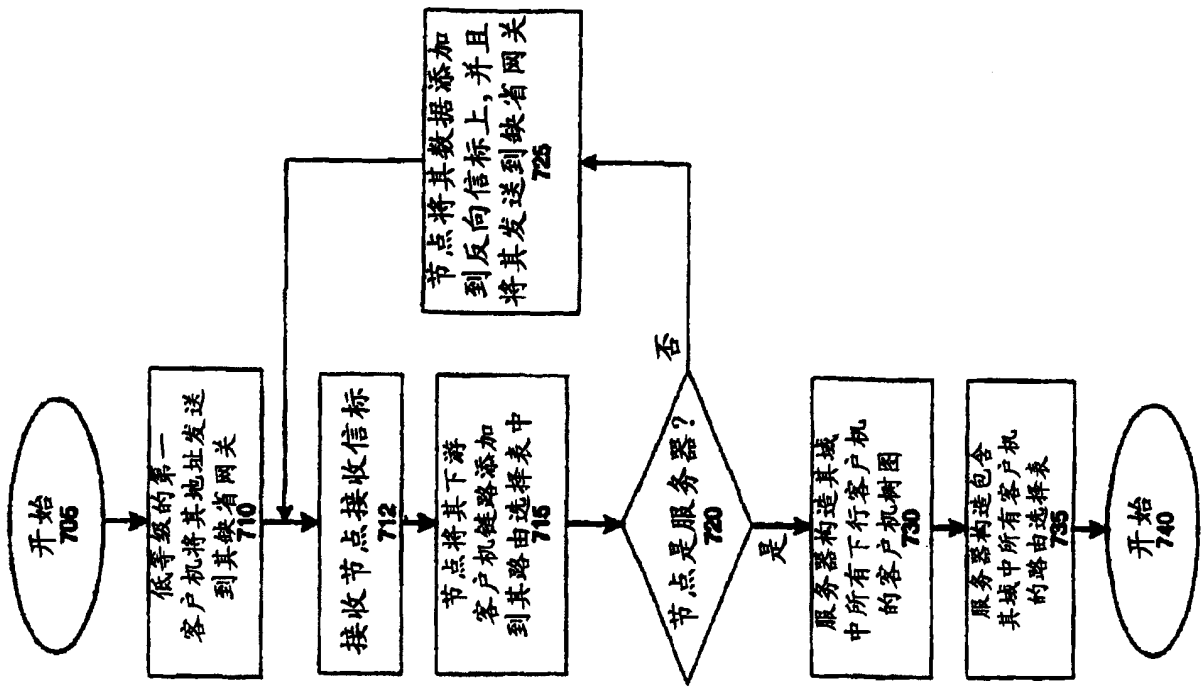


图 7A

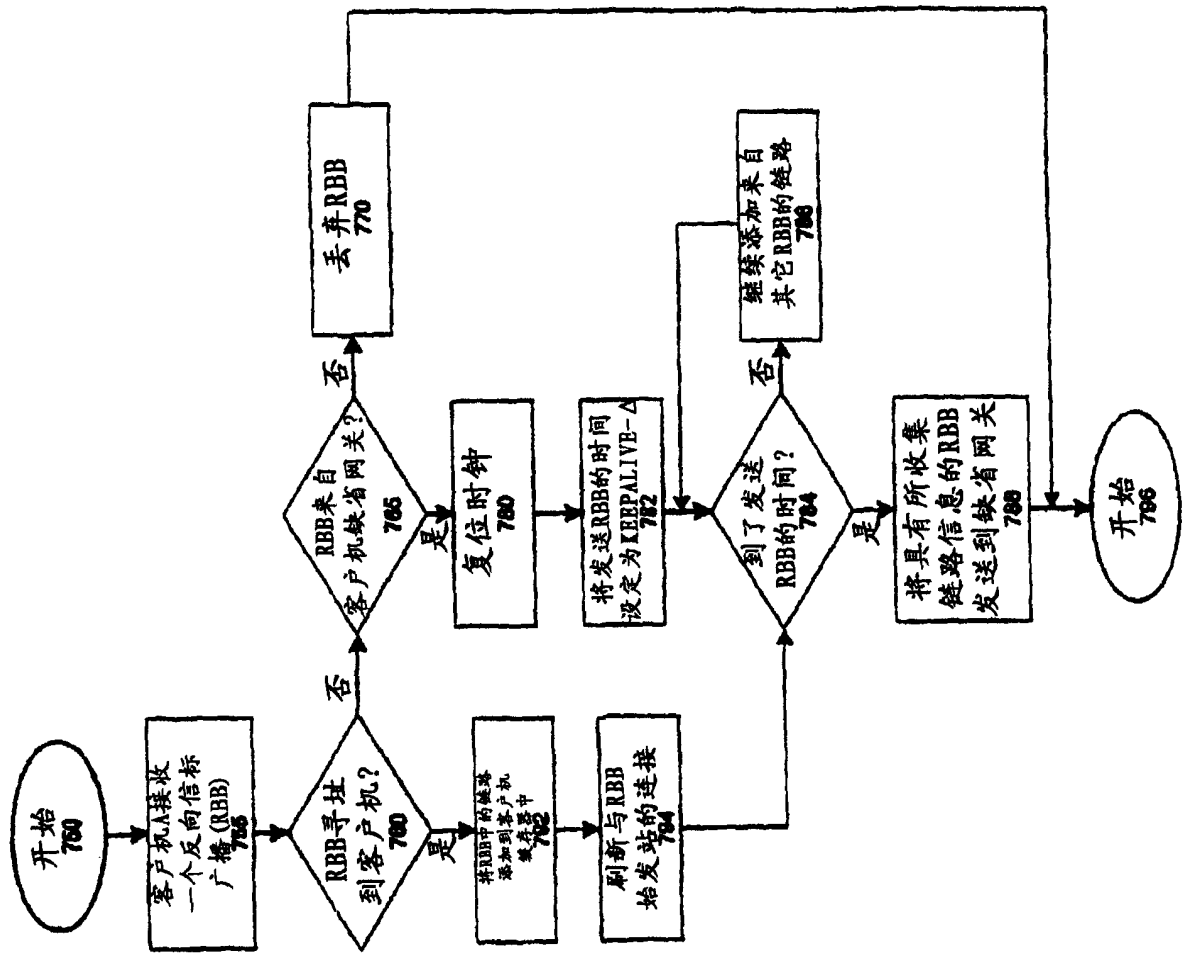


图 7B

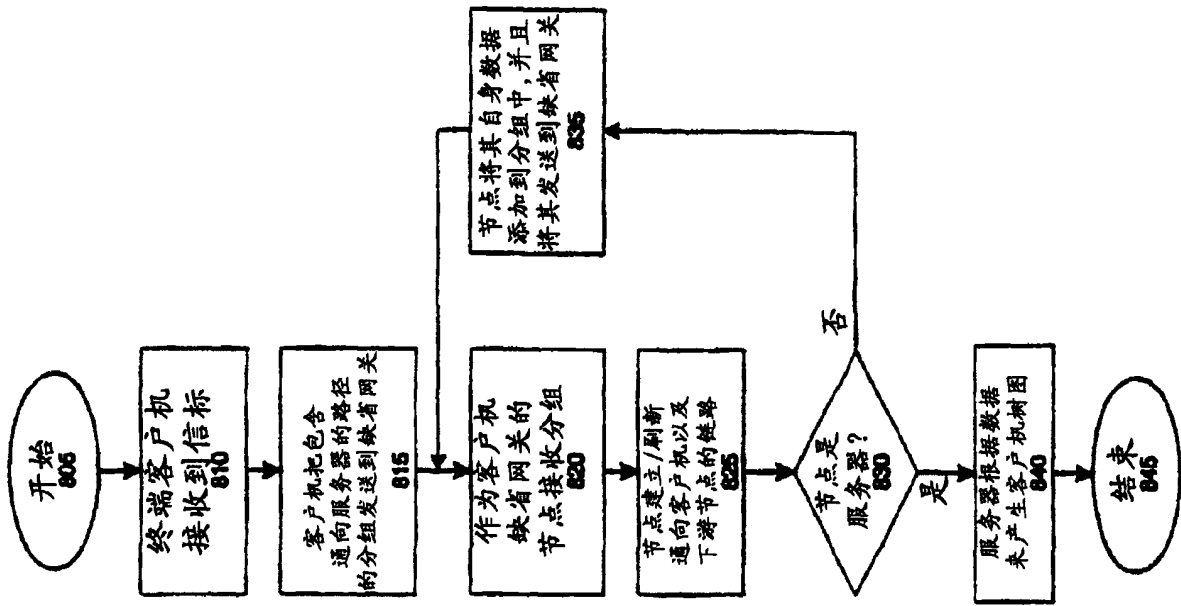


图 8A

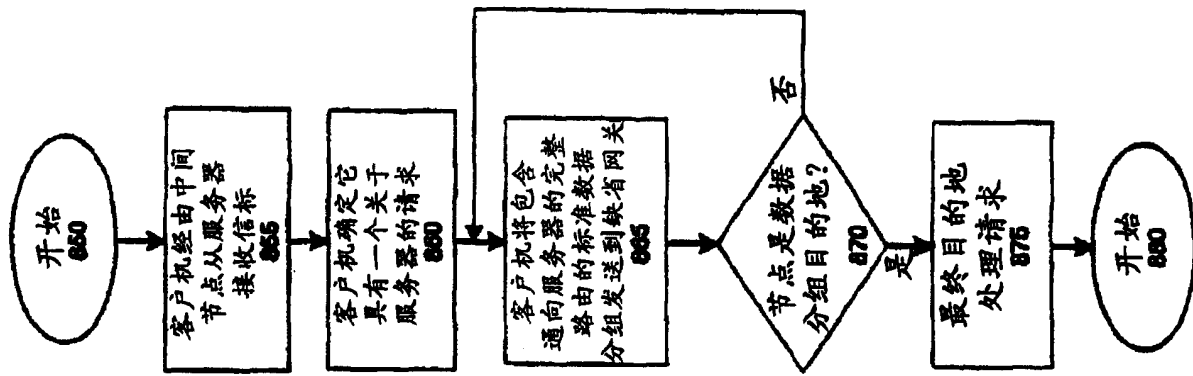


图 8B

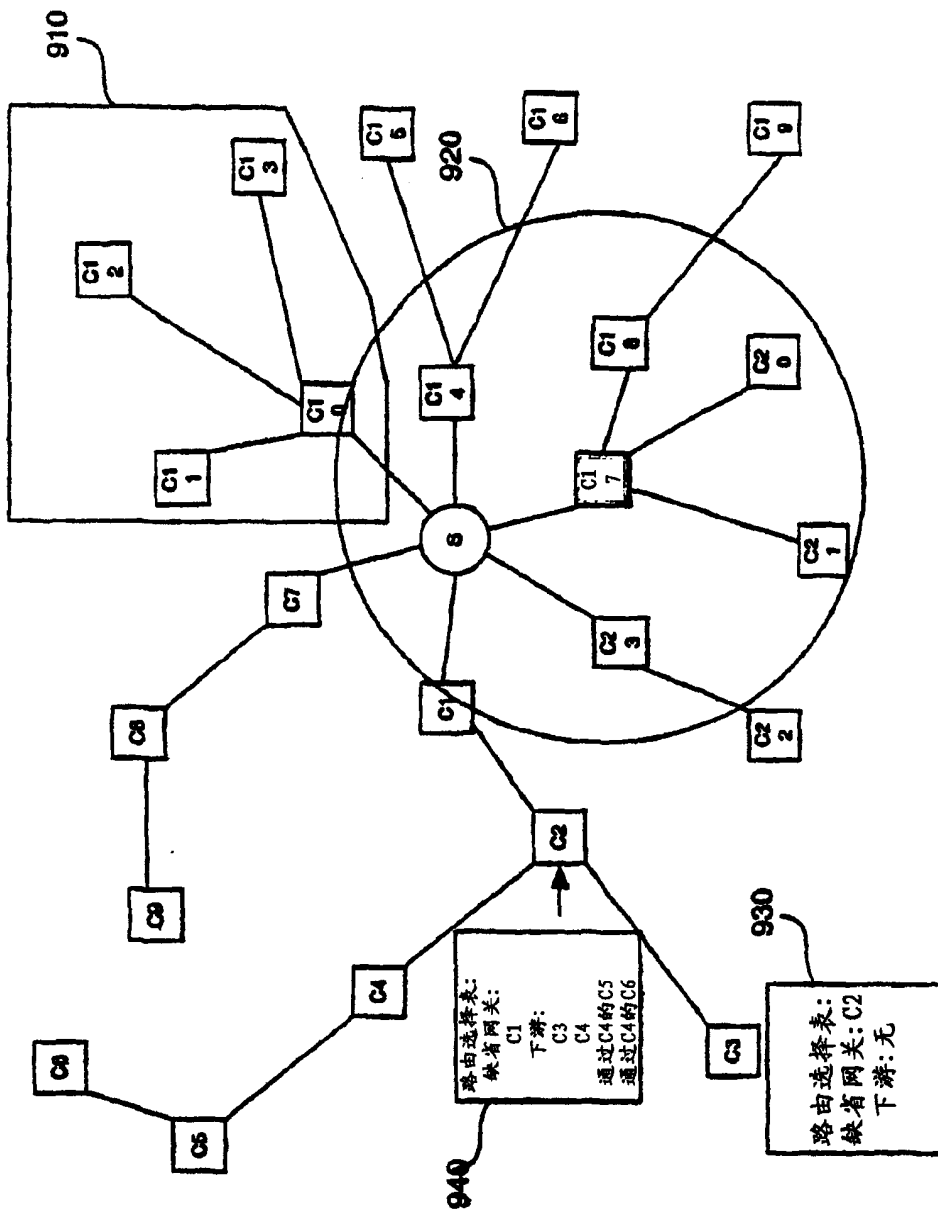


图9

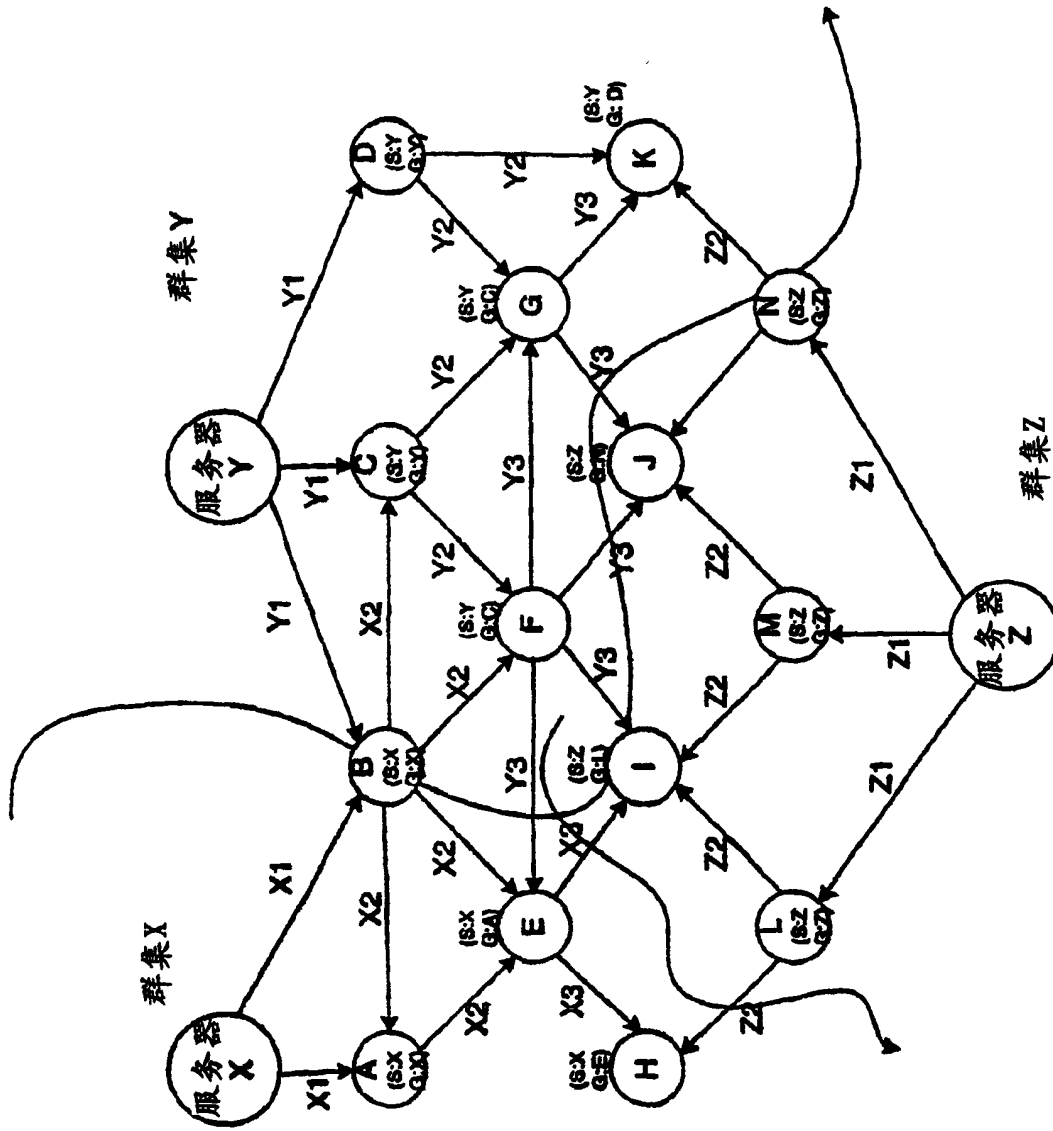


图10

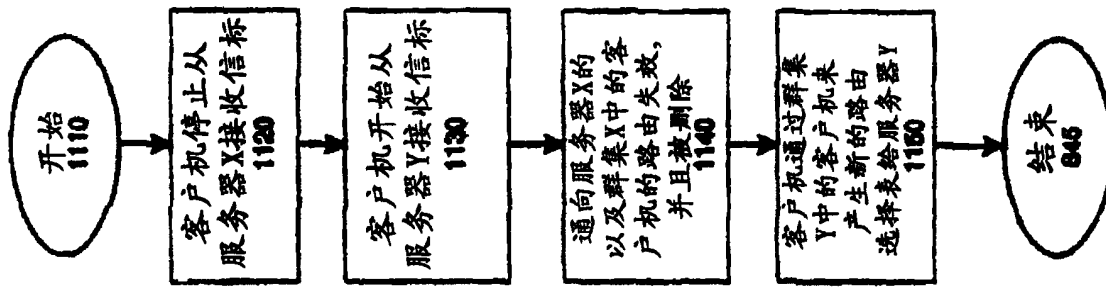


图11