

[51] Int. Cl.
H04L 12/24 (2006.01)



[21] 申请号 200480038068.3

[11] 公开号 CN 1898902A

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 杨 凯 刘 杰

代理人 杨 凯 刘 杰

[86] 国际申请 PCT/SE2004/001637 2004.11.9

[87] 国际公布 WO2005/060162 英 2005.6.30

[85] 进入国家阶段日期 2006.6.19

[71] 申请人 艾利森电话股份有限公司

地址 瑞典斯德哥尔摩

[72] 发明人 G·莫尔纳 G·托思

B · P · 格勒 A · R · 诺尔

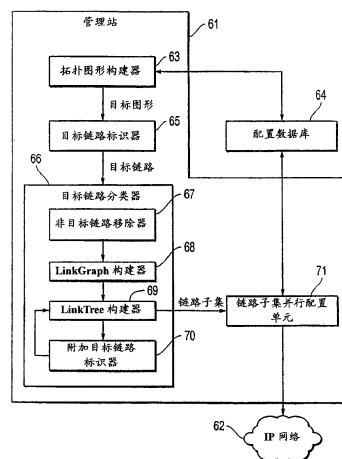
权利要求书 6 页 说明书 16 页 附图 9 页

[54] 发明名称

在基于因特网协议 (IP) 的网络中集中式配置
链路范围类型的被管理对象

[57] 摘要

一种从集中式管理节点配置基于 IP 的网络中的链路范围类被管理对象的方法。基于 IP 的网络 (62) 包括至少一个管理站 (61)、一组网络节点, 以及网络节点之间和管理站与网络节点之间的多个通信链路。优选地, 准备网络的开放最短路径优先 (OSPF) 拓扑图形, 并标识要配置的一组目标链路 (21)。然后将目标链路分类成 N 个不相交的子集 $T_1 - T_N$ (22)。然后以并行方式配置每个子集的链路, 其中起始于子集 T_1 , 并依次逐个处理每个子集 (23)。可以通过如下步骤对目标链路分类: 从 OSPF 拓扑图形中移除不需要配置的非目标链路 (31), 构建 LinkGraph 以确定该 OSPF 拓扑图形中余下的链路之间的相关性 (32), 根据 LinkGraph 构建 LinkTree 以基于链路之间的相关性将目标链路分类成子集 (33)。



1. 一种对基于因特网协议 (IP) 的网络进行链路范围配置的方法, 所述基于 IP 的网络具有至少一个管理站、一组网络节点以及所述网络节点之间和所述管理站与所述网络节点之间的多个通信链路, 所述方法包括如下步骤:

准备所述网络的拓扑图形;

标识要配置的一组目标链路;

将所述目标链路分类成 N 个不相交的子集 T_1 - T_N ; 以及

以并行方式配置每个子集中的链路, 起始于子集 T_1 , 并依次逐个处理每个子集。

2. 如权利要求 1 所述的方法, 其特征在于准备拓扑图形的所述步骤包括基于开放最短路径优先 (OSPF) 路由选择协议准备拓扑图形, 并且对所述目标链路分类的所述步骤包括如下步骤:

从 OSPF 拓扑图形中移除不需要配置的非目标链路;

确定所述 OSPF 拓扑图形中余下的链路之间的相关性; 以及

基于所述链路之间的相关性将所述目标链路分类成子集。

3. 如权利要求 2 所述的方法, 其特征在于确定所述链路之间的相关性的所述步骤包括构建 LinkGraph, 构建 LinkGraph 的所述步骤包括如下步骤:

对于所述 OSPF 拓扑图形中的每个目标链路, 将新节点置于所述 LinkGraph 中;

对于在 LinkGraph 中设置的每个节点, 根据所述 OSPF 拓扑图形创建邻节点的全网;

将表示所述 OSPF 拓扑图形中的所述管理站的节点添加到所述 LinkGraph; 以及

将表示所述管理站的节点连接到从所述 OSPF 拓扑图形中的所述管理站始发的链路。

4. 如权利要求 3 所述的方法, 其特征在于基于所述链路之间的相关性将所述目标链路分类成子集的所述步骤包括, 根据所述 LinkGraph 构建 LinkTree, 构建 LinkTree 的所述步骤包括如下步骤:

(a) 将表示所述管理站的节点指定为第一起点;

(b) 将所述第一起点连接到与所述第一起点相邻节点的所有链路添加到所述 LinkTree;

(c) 选择与所述第一起点相邻的节点, 选择相邻节点的所述步骤包括如下步骤:

(c)(1) 如果有一个相邻节点具有比任何其他相邻节点多的邻节点, 则将具有最大数量的尚未添加到所述 LinkTree 的邻节点的相邻节点选择作为第二起点; 以及

(c)(2) 如果一个以上相邻节点具有最大数量的尚未添加到所述 LinkTree 的邻节点, 则从具有最大数量的邻节点的相邻节点中随意选择一个相邻节点作为第二起点;

(d) 将除已经在所述 LinkTree 中的链路之外的从所述第二起点始发的所有链路添加到所述 LinkTree;

(e) 选择所述 LinkTree 中的一个节点作为第三起点, 选择所述 LinkTree 中的一个节点的所述步骤包括如下步骤:

(e)(1) 如果所述 LinkTree 中有一个节点具有比任何其他节点多的邻节点, 则选择所述 LinkTree 中具有最大数量的尚未添加到所述 LinkTree 的邻节点的节点;

(e)(2) 如果一个以上节点具有最大数量的尚未添加到所述 LinkTree 的邻节点, 则选择距离所述第一起点最远的具有最大数量的邻节点的节点作为第三起点; 以及

(e)(3) 如果一个以上节点具有最大数量尚未添加到所述 LinkTree 的邻节点, 且具有最大数量邻节点的所有节点与所述第一起点距离相同, 则从具有最大数量邻节点的这些节点中随意选择一个节点作为第三起点;

(f) 将除已经在所述 LinkTree 中的链路之外的从所述第三起点始发的所有链路添加到所述 LinkTree;

(g) 判断是否已将所述 LinkGraph 中的所有节点添加到所述 LinkTree 中;

(h) 如果已经将所述 LinkGraph 中的所有节点添加到所述 LinkTree, 则将所述 LinkTree 中的所有链路分类成不相交的子集 T_i 。

5. 如权利要求 4 所述的方法, 还包括如下步骤:

判断是否有任何没有被添加到所述 LinkTree 的目标链路;

如果有尚未被添加到所述 LinkTree 的目标链路, 则创建包含尚未被添加到所述 LinkTree 的所述目标链路的链路子图形; 以及

重复步骤 (a) 至 (h) 以创建不相交的子集 T_{i+1} 。

6. 如权利要求 5 所述的方法, 其特征在于以并行方式配置每个子集中的链路的步骤包括如下步骤:

在所述 OSPF 拓扑图形中构造包含未被配置的非目标链路的框架; 以及

以并行方式配置相同层次上的所有目标链路的节点, 前提是上次配置的节点在所述框架中。

7. 一种对基于因特网协议 (IP) 的网络进行链路范围配置的管理站, 所述基于 IP 的网络具有一组网络节点以及所述网络节点之间和所述管理站与所述网络节点之间的多个通信链路, 所述管理站包括:

准备所述网络的拓扑图形的部件;

标识要配置的一组目标链路的部件;

将所述目标链路分类成 N 个不相交的子集 T_1 - T_N 的部件; 以及

以并行方式配置每个子集中的链路的部件, 起始于子集 T_1 , 并依次逐个处理每个子集。

8. 如权利要求 7 所述的管理站, 其特征在于准备拓扑图形的所述部件包括基于开放最短路径优先 (OSPF) 路由选择协议准备拓扑

图形的部件。

9. 如权利要求 8 所述的管理站, 其特征在于用于分类所述目标链路的所述部件包括:

从 OSPF 拓扑图形移除不需要配置的非目标链路的部件;

确定所述 OSPF 拓扑图形中余下的链路之间的相关性的部件;

以及

基于所述链路之间的相关性将所述目标链路分类成子集的部件。

10. 一种存储在媒体上且在基于因特网协议 (IP) 的网络中的管理站的处理器上运行的计算机软件程序, 所述基于 IP 的网络具有一组网络节点以及所述网络节点之间和所述管理站与所述网络节点之间的多个通信链路, 所述程序可用来通过使所述管理站执行如下步骤来对所述 IP 网络进行链路范围配置:

准备所述网络的拓扑图形;

标识要配置的一组目标链路;

将所述目标链路分类成 N 个不相交的子集 T_1 - T_N ; 以及

以并行方式配置每个子集中的链路, 起始于子集 T_1 , 并依次逐个处理每个子集。

11. 如权利要求 10 所述的计算机软件程序, 其特征在于所述程序使所述管理站基于开放最短路径优先 (OSPF) 路由选择协议准备拓扑图形, 并且所述程序使所述管理站执行如下步骤来对所述目标链路分类:

从所述 OSPF 拓扑图形移除不需要配置的非目标链路;

确定所述 OSPF 拓扑图形中余下的链路之间的相关性; 以及

基于所述链路之间的相关性将所述目标链路分类成子集。

12. 如权利要求 11 所述的计算机软件程序, 其特征在于所述程序使所述管理站通过执行如下步骤构建 LinkGraph 来确定所述链路之间的相关性:

对于所述 OSPF 拓扑图形中的每个目标链路, 将新节点置于所述 LinkGraph 中;

对于在 LinkGraph 中设置的每个节点, 根据所述 OSPF 拓扑图形创建邻节点的全网;

将表示所述 OSPF 拓扑图形中的所述管理站的节点添加到所述 LinkGraph; 以及

将表示所述管理站的节点连接到从所述 OSPF 拓扑图形中的所述管理站始发的链路。

13. 如权利要求 12 所述的计算机程序, 其特征在于所述程序使所述管理站通过执行如下步骤根据 LinkGraph 构建 LinkTree 来将所述目标链路分类成子集:

(a) 将表示所述管理站的节点指定为第一起点;

(b) 将所述第一起点连接到与所述第一起点相邻节点的所有链路添加到所述 LinkTree;

(c) 选择与所述第一起点相邻的节点, 选择相邻节点的所述步骤包括如下步骤:

(c)(1) 如果有一个相邻节点具有比任何其他相邻节点多的邻节点, 则将具有最大数量的尚未添加到所述 LinkTree 的邻节点的相邻节点选择作为第二起点; 以及

(c)(2) 如果一个以上相邻节点具有最大数量的尚未添加到所述 LinkTree 的邻节点, 则从具有最大数量的邻节点的相邻节点中随意选择一个相邻节点作为第二起点;

(d) 将除已经在所述 LinkTree 中的链路之外的从所述第二起点始发的所有链路添加到所述 LinkTree;

(e) 选择所述 LinkTree 中的一个节点作为第三起点, 选择所述 LinkTree 中的一个节点的所述步骤包括如下步骤:

(e)(1) 如果所述 LinkTree 中有一个节点具有比任何其他节点多的邻节点, 则选择所述 LinkTree 中具有最大数量的尚未添

加到所述 LinkTree 的邻节点的节点;

(e)(2) 如果一个以上节点具有最大数量的尚未添加到所述 LinkTree 的邻节点, 则选择距离所述第一起点最远的具有最大数量的邻节点的节点作为第三起点; 以及

(e)(3) 如果一个以上节点具有最大数量尚未添加到所述 LinkTree 的邻节点, 且具有最大数量邻节点的所有节点与所述第一起点距离相同, 则从具有最大数量邻节点的这些节点中随意选择一个节点作为第三起点;

(f) 将除已经在所述 LinkTree 中的链路之外的从所述第三起点始发的所有链路添加到所述 LinkTree;

(g) 判断是否已将所述 LinkGraph 中的所有节点添加到所述 LinkTree 中;

(h) 如果已经将所述 LinkGraph 中的所有节点添加到所述 LinkTree, 则将所述 LinkTree 中的所有链路分类成不相交的子集 T_i 。

14. 如权利要求 13 所述的计算机程序, 其特征在于所述程序使所述管理站执行如下步骤:

判断是否有任何没有被添加到所述 LinkTree 中的目标链路;

如果有尚未被添加到所述 LinkTree 的目标链路, 则创建包含尚未被添加到所述 LinkTree 的所述目标链路的链路子图形; 以及

重复步骤 (a) 至 (h) 以创建不相交的子集 T_{i+1} 。

15. 如权利要求 14 所述的计算机程序, 其特征在于所述程序使所述管理站通过执行如下步骤以并行方式配置每个子集中的链路:

在所述 OSPF 拓扑图形中构造包含未被配置的非目标链路的框架; 以及

以并行方式配置相同层次上的所有目标链路的节点, 前提是上次配置的节点在所述框架中。

在基于因特网协议（IP）的网络中 集中式配置链路范围类型的被管理对象

技术领域

一般来说，本发明涉及因特网协议（IP）通信网络。更具体且以非限定的方式来说，本发明涉及在基于 IP 的网络中从集中式管理节点配置链路范围类型的被管理对象的方法。

背景技术

因特网协议（IP）是一种独立于主机的物理连接来连接主机的通信协议。一般来说，IP 主机是包括 IP 协议栈和应用程序的计算机。如果一组主机直接连接，即它们位于同一电缆上，则它们可以直接相互通信。该配置称为 IP 网络或子网络（或仅仅是 IP 子网）。当这些 IP 主机未直接连接时，即存在多个物理上分离的链路时，需要路由器来提供物理上分离的 IP 子网上的主机之间的 IP 连接性。路由器连接不同 IP 子网。最大的基于 IP 的计算机网络是因特网，它由通过路由器连接的大量 IP 子网构成。当路由器连接至少两个子网时，这些子网上的主机可以经由路由器彼此交谈。当然，每个子网上直接连接的主机可以直接彼此交谈，但是当这些子网的其中之一上的主机要与另一个子网上的主机交谈时，通信经由路由器。路由器本身是一种具有转发主机发送的接收 IP 分组的专门硬件和软件的计算机。

路由器具有许多允许它们支持多种协议和服务并执行其他功能的实施功能。路由器功能由变量参数控制。这些参数值的给定集合称为配置。单个路由器的配置管理称为单元配置管理，而网络中多个路由器和主机的当前配置称为网络配置。网络配置管理包括规划

和设置网络的操作功能。这些功能可以包括路由选择协议、转发策略、虚拟专用网络、某些服务质量（QoS）特征等。此外，还有如 IOverPPP 连接的链路相关的配置。每个路由器执行网络配置中其单独部分，如第 1 层物理连接的特定属性、第 2 层数据链路层接口、软件配置、元素安全性等。

对于给定路由器，这些参数值的集合是该路由器的配置。同样地，网络中这些路由器配置的集合的并集是网络配置。但是，这些集合不是析取的。路由器配置具有与网络中其他路由器相关的一些变量以及与其他路由器无关的一些变量。这些变量参数可以按如下分类：

- 路由器范围参数：这些参数仅与路由器本身相关。更改路由器范围参数不会对其他路由器的操作产生直接影响（例如，更改路由器的主机名、访问密码、路由选择进程 ID 等）。

- 链路范围参数：这些参数与通过一条链路（可以是如 PPPoverSerial 连接的物理链路，或可以是如开放最短路径优先（OSPF）邻接连接的逻辑链路）连接的多个路由器相关。链路范围参数必须在由配置的链路连接的路由器中具有一致的值（例如对于构建正确邻接的相邻路由器，OSPF Hellointerval 必须是相同的）。

- 区域范围参数：这些参数（例如自治系统（AS）的相同 AS 编号或 OSPF 区域的相同 OSPF AreaID）对于属于一个逻辑域或区域的一组路由器是相关的。

如上所述，路由器的功能基本是确定要将接收到的 IP 分组转发到哪里，并将该分组转发到它的目的地。可以使用静态或动态路由选择将转发信息提供给路由器。在静态路由选择的情况中，网络管理员手动将路由器设置在每个路由器的路由选择表中。在动态路由选择的情况中，路由器使用路由选择协议来确定网络中现存的路由。路由器使用路由选择协议确定的信息维护它们自己的路由选择表。

当网络中存在多于某个数量的路由器（例如 4 或 5 个）时，可

以优选使用动态方法。这意味着在每个路由器中启动路由选择协议，并将该路由器配置为正确地操作。IP 网络中的通用路由选择协议的其中一种是 OSPF 协议。该协议的配置可采用与变量参数相同的方式被分类，即路由器范围、链路范围和区域范围。路由器范围指在路由器中配置 OSPF 进程的过程。链路范围指配置 OSPF 链路（邻接）的过程。区域范围指配置 OSPF 区域的过程。

在目前 IP 网络管理中，网络管理员使用如下元素管理方法的其中之一手动执行大多数配置操作：

- 命令行界面 (CLI)：这是用于路由器配置管理的最广泛使用的方法。利用 CLI，通过远程登陆或控制台连接访问路由器，并且使用某个命令集，网络管理员输入命令从路由器检索信息并设置参数。CLI 命令集可能非常大。CLI 的最佳示例是 Cisco CLI，它是事实上的标准。

- 配置文件编辑：该方法是 CLI 方法的特殊用法。在该情况中，网络管理员编辑包含一系列 CLI 命令的配置文件。然后使用文件传输协议 (FTP) 或普通文件传输协议 (TFTP) 将该配置文件下载到相关路由器。该过程可能需要一些 CLI 交互，例如如果路由器仅具有 FTP 或 TFTP 客户机，则从路由器启动下载。

- 基于菜单的元素管理器：有少数路由器没有 CLI 接口，但是具有可通过远程登陆或控制台连接访问的菜单驱动的系统。网络管理员可以使用该接口查看或设置路由器配置。该方法使用得非常少。

- 简单网络管理协议 (SNMP)：SNMP 是因特网工程任务组 (IETF) 标准协议，它提供一种元素监视和配置的标准方法。管理信息库 (MIB) 定义被管理对象及其属性。SNMP 是一种获取并设置这些属性值的协议。多个应用使用 SNMP。有称为 MIB 的浏览器，它仅浏览给定 MIB，并获取或设置有关一个目标路由器的属性。有一些应用可以处理多于一个路由器。但是在实践中，SNMP 通常用于监视、收集统计数字以及故障管理，而非配置管理。此情况的一

个原因是 MIB 大多包含提供统计数字的只读属性。另一个原因是标准 MIB 不涵盖任何情况，所以由专有的 MIB 而非该标准 MIB 会更好地管理许多类型的路由器。

- 基于 HTTP 的元素管理：一些类型的路由器具有 Web 服务器服务。超文本传输协议（HTTP）浏览器可以访问该配置和其他信息。用户可以在提供的超文本标记语言（HTML）页面上检索或设置参数。

在这些元素管理方法中，SNMP 最适用于应用。CLI 是针对手动配置管理设计的，尽管可以修改它以供应用程序使用。可以通过应用程序来协助配置文件编辑。基于菜单和基于 Web 服务器的方法不是为应用程序使用而设计的。这些方法仅适用于手动配置管理。

此外，还存在提供某个层次的网络管理的应用程序。这些程序可以分为两个基本集合，路由器供应商提供的应用程序（例如，Cisco 公司的 CiscoWorks）以及其他软件开发公司提供的应用程序（例如 HP OpenView）。

对于远程管理，远程登陆协议主要用于访问路由器。可以直接对目标路由器执行远程登陆或通过远程登陆到目标的相邻路由器，并从该相邻路由器远程登陆到目标路由器来间接执行远程登陆。但是，这种类型的管理总体上不知道配置属性的范围。有一个对范围意识配置的唯一已知描述，即在 LISA 2000 第 14 次系统管理会议的会刊 P. Krishnan 等人的文档标题为“IP 网络的配置操作的排序”（此后，Krishnan）。但是，如下文所示出的，Krishnan 方法并不是一个满意的解决方案。

现有技术的 IP 配置管理方法的最重要特征是每个目标路由器是逐个且彼此无关地配置。网络管理员“构思”设计操作并通过逐个配置相关路由器来实现它。第一步是管理员定义要更改的参数和要设置的值。然后将这些更改付诸于相关路由器。第一部分是逻辑性的，而下一部分则是实质性的。因此，第一步是在管理员的构思中或纸张上完成。然后他对相关路由器执行必要的元素管理操作。以此方

式管理链路范围 OSPF 参数可能产生配置成本问题、排序问题、冗长的操作时间以及取消和错误处理的问题。下文将论述这些问题的每一种。

配置成本问题：OSPF 链路具有链路范围属性。这些属性存储在路由器中并且必须对于适合的 OSPF 邻接具有一致的值。OSPF 链路的逻辑配置只需定义这些链路范围属性的值。但是物理配置需要在配置的 OSPF 链路所连接的每个路由器中设置这些值。在点到点链路的情况下，这意味着两个路由器。但是，在如广播或非广播多路访问（NBMA）的其他情况中，可能有多于两个路由器。再者，如果要实现多条 OSPF 链路，目标路由器的数量会倍增。考虑到两个重要的链路范围 OSPF 参数 Hello-和 Deadinterval，管理员必须为每个目标链路定义新的值。在逻辑上有两个参数要更改，但是管理员需要访问多个路由器并在每个路由器上设置这两个值。理论上所需的配置工作（在本示例中为设置两个参数）与实际的配置工作（以相同的值在多个路由器中设置两个参数）之间的差异可能是非常烦琐的。此外，网络管理员需要在多个路由器上多次做相同的事。这增加了网络配置中人为失误的概率。若存在一种既能降低网络管理员的工作量又能降低网络配置中人为失误的概率的配置方法将是有利的。

排序问题：另一个更重要的问题是排序问题。大型 IP 网络的管理将最可能采用集中式，而非分布式。一般来说，因为仅一些网络操作中心负责网络，配置更改（元素配置管理）是从这些中心执行的。因此，在操作期间保持与每个目标路由器的 IP 连接性是非常重要的。在小网络中，目标路由器的数量不多，这可能不是个大问题。但是，当路由器的数量在几百台时，元素配置的次序则是重要的。为了理解该问题，需要回顾 OSPF 协议是如何处理链路（即邻接）的。

相邻 OSPF 路由器构建邻接。此通道用于通信、广告已知路由以及使 OSPF 数据库同步。在无适合通信的情况下，OSPF 无法使用一些连接，并且某些路由不可用于路由选择计算。因此，这些路

由不可用于通信业务。由此，可能无法从网络的某些点访问一些路由器、主机或子网。OSPF 邻接对于正确的 OSPF 路由选择是至关重要的。建立 OSPF 邻接基于链路范围属性。一般性的规则是这些参数必须具有一致的值。在点到点链路上，两个相邻路由器必须具有 OSPF 链路的一致 OSPF 链路范围属性。在广播、NBMA 或点到多点连接中，邻接在广告一致的链路范围值的相邻点之间建立。如果参与路由器广告与其他路由器不同的值，则其他路由器不与之建立邻接，该参与路由器无法建立与其他路由器的邻接。

还应该理解如何在 OSPF 链路上更改 OSPF 链路范围参数。当网络管理员要更改工作中的 OSPF 链路上的链路范围属性并访问它的一个端点时，该管理员必须考虑当链路范围属性更改时，该 OSPF 链路可能丢失，直到其他端点具有一致的值为止。一个重要因素是链路配置时间，即该链路上第一个路由器访问到最后一个路由器中最后一个参数设置之间的时间。链路丢失的概率取决于初始 Hello-、Deadinterval 和该配置时间。相邻路由器之间的传输速率可以忽略不计。在某些环境中，链路可能在操作期间保持完整，而在另一些环境中，链路可能在以新链路范围值建立新链路之前暂时丢失。当要配置的链路位于网络的树部分时此行为是重要的。

图 1 是说明现有技术遇到的排序问题的简化网络示意图。在图示的情况中，第一个访问是从管理站 11 对最近的路由器 R-1 12，其中执行期望的链路范围更改。然而，如果链路在访问其他端之前丢失，则管理站无法达到最远的路由器 R-3 14 或可能无法达到中间路由器 R-2 13。因此，管理站无法设置最远的路由器 R-3 中的新链路范围值，并无法建立新链路。因此，随机配置顺序可能容易导致永久路由器丢失。若存在一种既能解决排序问题并能避免配置网络时丢失路由器的配置方法会是有利的。

长操作时间：考虑大型网络的场合，其中以许多链路为目标来进行链路范围值更改，并因此要配置许多路由器，操作时间是重要

的。在此配置操作期间，不建议启动其他配置操作，因为此操作会影响路由选择。在配置操作期间，当一些目标链路暂时丢失时发生瞬时路由选择更改。以常规方式（逐个且依次地）配置目标路由器可能导致长操作时间。若存在一种缩减与网络配置相关联的操作时间的配置方法会是有利的。

取消和错误处理：当网络操作员改变构思或意识到他启动一个不正确的配置操作时，他可能希望取消它。最安全的解决方案当然是不取消，并让操作完成。但是，在此情况中，他可能需要执行大量的附加配置，而仅仅是为了返回先前的状态。因此，能够取消配置操作是有用的附加功能，但是其实现并不容易。关于取消的问题在于有段时间无法取消配置操作。如果在配置链路的一个端点之后而在配置另一个（一些）端点之前取消操作，则将丢失链路。这更可能发生在以并行方式配置许多链路时。因此，当启动取消时，已经启动配置的任何链路必须完成操作，但是不应启动其他链路的配置。以常规方法正确取消不是主要问题，但是在基于软件的解决方案中，尤其在并行执行的情况中，应该仔细考虑此情况。

另一个重要的考虑是一些元素管理操作可能因多种原因而失败。如果发生此类型的错误，则应该采用与取消相同的方式停止操作。但是，仅停止是不够的。同样重要的是从中央管理站处理引起错误的状况。但是，采用现有技术的解决方案，这并不总是可实现的。采用随机配置，可能导致的情况是中央管理站无法做任何事来解决错误，而技术人员必须在故障的路由器处手动校正错误。

Krishnan 文献（上文引用）提出的解决方案基于当前路由选择来计算链路和路由器次序。在管理站和目标路由器之间具有正向和反向路由，它构建了定义次序的树。该树从叶到根。Krishnan 解决方案的重要特征和局限是（1）仅考虑对称路由选择；（2）从路由器本身获取路由选择信息；（3）未考虑任何取消；以及（4）未考虑错误处理。

若存在一种解决上文论述的问题的 IP 配置方法会是有利的。本发明提供这样一种方法。

发明概述

本发明是一种在 IP 网络中从集中式管理节点对开放最短路径优先 (OSPF) 链路进行链路范围配置的方法。该方法提出 (1) 一种排序问题的解决方案; (2) 通过并行执行加快操作时间; (3) 适合的取消; (4) 良好的错误处理; 以及 (5) 比现有技术解决方案更简单的次序计算。

由此在一个方面, 本发明针对一种配置基于 IP 的网络的方法, 该基于 IP 的网络具有至少一个管理站、一组网络节点和网络节点之间以及管理站与网络节点之间的通信链路。该方法包括如下步骤: 准备网络的 OSPF 拓扑图形; 标识要配置的一组目标链路; 将目标链路分类成 N 个不相交的子集 T_1-T_N ; 以及以并行方式配置每个子集中的链路, 起始于子集 T_1 , 并依次逐个处理每个子集。可以通过如下步骤分类目标链路: 从 OSPF 拓扑图形中移除不需要配置的非目标链路, 确定该 OSPF 拓扑图形中余下的链路之间的相关性, 并基于链路之间的相关性将目标链路分类成子集。

链路之间的相关性通过构建 LinkGraph 来确定。为 OSPF 拓扑图形中的每个目标链路在 LinkGraph 中设置新节点来构建 LinkGraph。对于在 LinkGraph 中设置的每个新节点, 根据 OSPF 拓扑图形创建邻节点的全网 (full mesh)。此后将表示 OSPF 拓扑图形中的管理站的节点添加到 LinkGraph; 以及将表示管理站的节点连接到从 OSPF 拓扑图形中的管理站始发的链路。

可以根据 LinkGraph 构建 LinkTree 基于链路之间的相关性将目标链路分类成子集。可以通过将表示管理站的节点指定为第一起点来构建 LinkTree。然后, 将第一起点连接到与该第一起点相邻的节点的所有链路添加到 LinkTree。然后将与第一起点相邻的节点选为第二

起点。此选择可以通过选择具有最大数量的尚未添加到 LinkTree 的邻节点的相邻节点来实施（如果有相邻节点比任何其他相邻节点具有更多邻节点）。如果一个以上相邻节点具有最大数量的尚未添加到 LinkTree 的邻节点，则从具有最大数量的邻节点的相邻节点中随意选择第二起点。

将除已经在 LinkTree 中的链路之外的从第二起点始发的所有链路添加到 LinkTree，并选择 LinkTree 中的另一个节点作为第三起点来延伸 LinkTree。可以通过选择具有最大数量的尚未添加到 LinkTree 的邻节点的节点来选择第三起点（如果 LinkTree 中有节点比任何其他节点具有更多邻节点）。如果一个以上节点具有最大数量的尚未添加到 LinkTree 的邻节点，则选择距离第一起点最远的具有最大数量的邻节点的节点作为第三起点。如果多于一个节点具有最大数量尚未添加到 LinkTree 的邻节点，且具有最大数量邻节点的所有节点与第一起点距离相同，则从具有最大数量邻节点的节点中随意选择第三起点。

将除已经在 LinkTree 中的链路之外的从第三起点始发的所有链路添加到所述 LinkTree 来延伸 LinkTree。如果已经将 LinkGraph 的所有节点添加到 LinkTree，则 LinkTree 中的所有链路分类成不相交的子集 T_i 。然后判断是否有任何没有被添加到 LinkTree 中的目标链路。如果是的话，则创建包括尚未被添加到 LinkTree 的目标链路的链路子图形，以及重复上述步骤来创建不相交的子集 T_{i+1} 。当创建了所有子集时，可以通过如下步骤以并行方式配置每个子集中的链路：在 OSPF 拓扑图形中构造包括未被配置的非目标链路的框架；以及以并行方式配置相同层次上的所有目标链路的节点，前提是上次配置的节点在该框架中。

附图简介

图 1（现有技术）是说明现有技术遇到的排序问题的简化网络示

意图;

图 2 是说明本发明方法的优选实施例中整体算法的步骤的流程图;

图 3 是说明本发明方法的优选实施例中分类算法的步骤的流程图;

图 4A-4B 是说明由 LinkGraph 构建 LinkTree 的算法步骤的流程图的一部分;

图 5 是适于配合本发明使用的示范 OSPF 拓扑图形;

图 6 是由 OSPF 拓扑图形推导出来且用来查找链路之间的相关性的示范 LinkGraph;

图 7 说明由图 6 的示范 LinkGraph 构造 LinkTree 的过程;

图 8 说明示出在第一子集 T_1 中分类的链路的图 7 的 LinkTree;

图 9 说明当移除子集 T_1 中的链路的邻点时所产生的图形;

图 10 是图 9 的图形的 LinkGraph;

图 11 是由图 10 的 LinkGraph 构建的 LinkTree; 以及

图 12 是根据本发明原理用于对 IP 网络执行链路范围配置的管理站的简化框图。

实施例的详细说明

本发明提出一种从集中式管理节点配置基于 IP 的网络中的链路范围类型的被管理对象的改进方法。依据开放最短路径优先 (OSPF) 协议描述一个示范实施例, 因为 OSPF 具有非常明显的链路范围、路由器范围和区域范围被管理对象, 可体现配置链路范围参数的问题。为配置而访问路由器可以通过直接连接或远程方式来实现。对于直接连接, 网络管理员的终端、控制台或工作站具有至路由器的直接连接。例如使用串行控制台连接独立于被管理 IP 设施来进行连接。对于远程配置, 网络管理员经由被管理 IP 网络从连接到路由器的机器来访问路由器, 例如使用远程登陆登录到路由器。本发明的优选

实施例提出一种执行远程配置的方法。

在今天复杂且大型的 IP 网络中，网络配置常常涉及配置可以视为逻辑实体的网络功能，如服务、路径、诸如 OSPF 的协议或它们的子集。由此，当配置许多路由器时执行大量配置操作。这样当网络操作员期望在网络配置中实施更改时，他必须执行多个元素配置操作。与一个以上路由器相关的配置操作称为多目标操作。一个示例是当网络操作员希望更改网络中多个 OSPF 链路上的 OSPF Hellointerval 设置。本发明的优选实施例提出一种执行多目标操作的方法。

本发明可采用软件形式来实施。该管理软件为网络管理员提供 OSPF 链路范围操作。由此，管理员只需定义目标链路和链路范围参数的新值，然后软件完成余下的操作。因此，在该优选实施例中，本发明提出一种用于远程多目标链路范围 OSPF 配置的可实施软件解决方案。

本发明适用于任何拓扑和任何路由选择（对称的和非对称的）。本发明通过在大型复杂网络中查找可采用并行方式配置的最大数量目标链路（即使它们之间没有拓扑相关性）来加速配置操作。

图 2 是说明本发明方法的优选实施例中的整体算法的步骤的流程图。本发明实施一种算法，该算法通过以并行方式配置所选的目标链路来以尽可能少的步骤配置那些链路。该算法是一种贪心图形算法，其中在步骤 21，取 G （网络的 OSPF 拓扑图形）和 T （目标链路的集合）作为输入。在步骤 22，该算法将目标链路分类成 N 个不相交的子集 $T(T_1-T_N)$ 。在分类之后，算法在步骤 23 逐个提取子集，起始于 T_1 并以并行方式配置它们的元素。

图 3 是说明本发明方法的优选实施例中的分类算法 22 的步骤的流程图。该分类算法是一种递归算法，是整体算法的一个重要部分。在步骤 31，该分类算法通过删除图形中与非目标链路连接的路由器节点来从拓扑图形 G 中移除非目标链路。在步骤 32，构建所说的

LinkGraph 以发现链路之间的相关性。对于每个目标链路将新节点置于图形中，并为每个路由器创建初始图形中路由器节点的邻节点的全网（参见下文图 6-10 中的示例）。最后，在步骤 33，按图 4A-4B 所示和所述，算法使用 LinkGraph 构建 LinkTree。

图 4A-4B 是说明用于从 LinkGraph 中构建 LinkTree 的算法 33 的步骤的流程图的一部分。在步骤 41，计数器被设置为 1。在步骤 42，将表示管理站的节点 M 标识为起点，并将其置于 LinkTree 中。在步骤 43，将始发于 M 的所有边置于 LinkTree 中。在步骤 44，判断是否一个以上尚未在 LinkTree 中的叶（节点）具有最大数量邻节点。如果不是的话，因此单个叶具有最大数量邻节点，则在步骤 45 选择具有最大数量邻节点的叶作为变量 S。如果一个以上叶具有相等最大数量的邻节点，则方法从步骤 44 移至步骤 46，其中判断具有最大数量邻节点的任何叶是否在 LinkGraph 中具有大于 2 的度。如果是的话，则方法移至步骤 47，其中判断是有一个度大于 2 且距离 M 最远的叶，还是有多度大于 2 且距离 M 最远的叶。如果有一个度大于 2 且距离 M 最远的叶，则方法移至步骤 48，其中将度大于 2 且距离 M 最远的这个叶选为变量 S。但是，如果有多度大于 2 且距离 M 最远的叶，则方法移至步骤 49，其中从这多个满足这些标准的叶中随意选择一个叶作为变量 S。

返回到步骤 46，如果判断没有具有最大数量邻节点的叶在 LinkGraph 中具有大于 2 的度（即有多度等于 2 的叶），则方法移至步骤 51，其中判断是有度等于 2 且最靠近 M 的单个叶，还是有多度等于 2 且最靠近 M 的叶。如果有一个度等于 2 且最靠近 M 的叶，则方法移至步骤 52，其中将度等于 2 且最靠近 M 的单个叶选为变量 S。但是，如果有多度等于 2 且最靠近 M 的叶，则方法移至步骤 53，其中从这多个满足这些标准的叶中随意选择一个叶作为变量 S。由此在步骤 45、48、49、52 或 53 选择了一个叶，方法进行到步骤 54，其中将 S 设置为所选的叶。然后方法移至图 4B。

在步骤 55, 方法在 LinkTree 中置入所有始发于 S 且不返回该 LinkTree 的边。在步骤 56, 判断是否已将 LinkGraph 中的所有节点置于 LinkTree 中。如果是的话, 方法移至步骤 57, 其中将 LinkTree 的所有叶置于子集 T_i 中。然而, 如果 LinkGraph 中的所有节点尚未被置于 LinkTree 中, 则方法移至步骤 58, 并通过从初始 LinkGraph 中减去已经被置于 LinkTree 中的节点来构建新的 LinkGraph。在步骤 59, 步进计数器 (i) 按一 (1) 递增, 然后方法返回到步骤 42 并重复该过程。

当初始 OSPF 拓扑图形仅由为 M 的一个节点组成时, 定义子集 T_i , 并准备将其传递到配置算法。

图 5 是适于配合本发明使用的示范 OSPF 拓扑图形。在图示的示例中, 这些链路编号为 1-10。网络中的每个链路都是目标链路, 所以无法通过减去非目标链路的端点来缩减图形。根据图 4A-4B 所示的过程, 第一步骤是构建 LinkGraph 以发现链路之间的相关性。图 6 示出结果 LinkGraph。然后利用 LinkGraph 按照上述规则构造 LinkTree。

图 7 说明根据图 4A-4B 的过程从图 6 的 LinkGraph 构造 LinkTree 的过程的示例。首先, 选择节点 M, 并将边 (M, 1)、(M, 2) 和 (M, 4) 添加到 LinkTree。图 7 中将这链路标记为 “a”, 以指定首先将它们添加到 LinkTree。下一步是调查实际的 LinkTree 的叶有多少非 LinkTree 节点邻点。在此情况中, 节点 1 具有两个这种邻节点 (节点 3 和节点 5); 节点 2 具有两个这种邻节点 (节点 6 和节点 8); 并且节点 4 也具有两个这种邻节点 (节点 3 和节点 7)。由此, 必须就应该将这三个节点 (节点 1、2 和 4) 中的哪一个节点设为 S 作出决策。所有这三个节点在 LinkTree 中为一个步阶深 (即度等于 2), 没有单个节点最靠近 M, 所以根据图 4A 的步骤 53 随意选择一个节点。

在图示的示例中, 选择节点 4 并将其设为 S。由此, 添加到 LinkTree 的下一个边是 (4, 3) 和 (4, 7)。图 7 中将这链路标记为 “b”, 以

指定将它们第二批添加到 LinkTree。在下一步中，再次调查实际 LinkTree 的叶以确定哪个节点具有最大数量非 LinkTree 节点邻点。在此点上，确定节点 1 和节点 3 都具有一个非 LinkTree 节点邻点，而节点 7 和节点 2 都具有两个非 LinkTree 节点邻点。但是在此情况中，节点 7 在 LinkTree 中是两个步阶深度（即度大于 2），而节点 2 仅为一个步阶深度（即度等于 2）。因此，根据图 4A 的步骤 48，将节点 7 选为 S，并将边（7, 8）和（7, 10）添加到 LinkTree。图 7 中将这此链路标记为“c”，以指定第三批将它们添加到 LinkTree。

遵循相同的过程，再次调查实际 LinkTree 的叶以确定哪个节点具有最大数量非 LinkTree 节点邻点。在此点上，确定节点 8 和节点 10 都具有一个非 LinkTree 节点邻点。这两个节点在 LinkTree 中是三个步阶深（即度大于 2），所以根据图 4A 的步骤 49 随意选择一个节点。在图示的示例中，选择节点 8 并将其设为 S，并将边（8, 6）添加到 LinkTree。图 7 中将边（8, 6）标记为“d”，以指定第四批将它添加到 LinkTree。

在下一步中，再次调查实际 LinkTree 的叶以确定哪个节点具有最大数量非 LinkTree 节点邻点。在此点上，确定节点 6 具有最大非 LinkTree 节点邻点，并根据图 4A 的步骤 45 将其选为 S。然后将边（6, 9）和（6, 5）添加到 LinkTree，并在图 7 中将它们其标记为“e”，以指定第五批将它们添加到 LinkTree。这样就完成了图 7 所示的 LinkTree 的构造。

在构建 LinkTree 之后，通过查找 LinkTree 中的叶来确定可并行配置的链路的集合。由此 T_1 （即第一步中配置的链路集合）是{1, 2, 3, 5, 9, 10}。图 8 中示出这些链路。然后删除这些链路的邻点。图 9 示出结果图形，图 10 示出它的 LinkGraph。在下一步中，更新 LinkGraph，图 11 示出了结果 LinkTree。由此，集合 T_2 是{4, 6, 7, 8}。

元素管理操作按 LinkGraph 管理的次序来执行。以并行方式配置相同层次的链路。按如下方式从图形 G 得出实际目标链路连接的

路由器之间的次序。首先在初始 OSPF 拓扑图形中构造框架、即非目标链路和未被配置的链路构成的子图形。其结果是连接的图形。在链路配置的过程中，路由器修改次序中的唯一约束是上次配置的路由器必须在框架中。其他路由器的次序是随意的；可采用并行方式执行它们的配置。在成功配置与该链路连接的所有其他路由器之后，才能修改最后一个路由器。测试已经示出当每个目标链路属于相同区域时，算法总是精确的。

通过应用如下实施规则来提供取消。在算法的实际阶段中，以并行方式配置一些目标链路。按如上所述访问路由器，算法不步进到下一阶段 (T_{i+1})，直到未成功配置实际链路中的每个链路为止。当启动取消时，必须完整配置已经配置路由器的实际链路。

图 12 是根据本发明原理用于对 IP 网络 62 进行链路范围配置的管理站 61 的简化框图。IP 网络包括一组网络节点以及网络节点之间和管理站与网络节点之间的多个通信链路。拓扑图形构建器 63 例如从管理站内部或外部的配置数据库 64 获取 IP 网络的配置信息。将拓扑图形发送到目标链路标识器 65，目标链路标识器 65 从要配置的拓扑图形中标识出一组目标链路。将所标识的一组目标链路发送到目标链路分类器 66，目标链路分类器 66 将这些目标链路分类成 N 个不相交的子集 T_1-T_N 。在目标链路分类器内，非目标链路移除器 67 从拓扑图形中移除未标识为目标链路的链路。然后将缩减的拓扑图形传递到 LinkGraph 构建器 68，它构建 LinkGraph 用于确定链路之间的相关性。然后将这些相关性发送到 LinkTree 构建器 69，它构建 LinkTree 用于基于链路之间的相关性将目标链路分类成 N 个不相交的子集 (T_1-T_N)。附加目标链路标识器 70 判断是否还有其他尚未置于该链路树中的目标链路，以及如果是的话，LinkTree 构建器构建另一个 LinkTree 以便将余下的目标链路分类成子集。

当已经将所有目标链路分类成子集时，链路树构建器 69 将这些子集 T_1-T_N 标识给链路子集并行配置单元 71，该单元以并行方式配置

每个子集中的 IP 网络链路，起始于子集 T1，依次逐个处理每个子集直到子集 TN。当完成该配置时，配置单元更新配置数据库 64。

本领域技术人员将认识到的，可以在广泛应用范围之上修改和变更本申请中所描述的创新概念。因此，申请专利的主题范围不应限于上文论述的任何特定示范原理，而是由所附权利要求定义。

图 1
(现有技术)

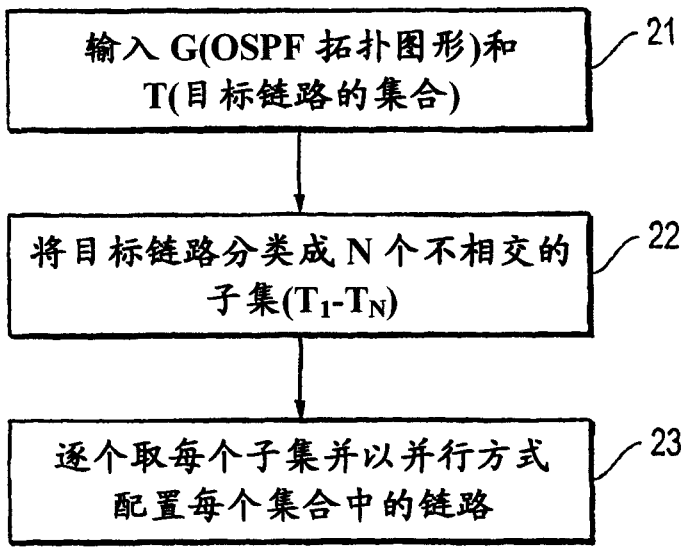
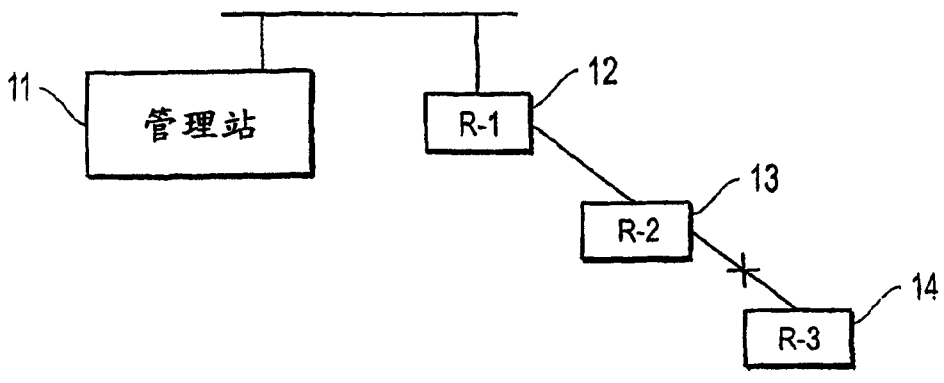


图 2

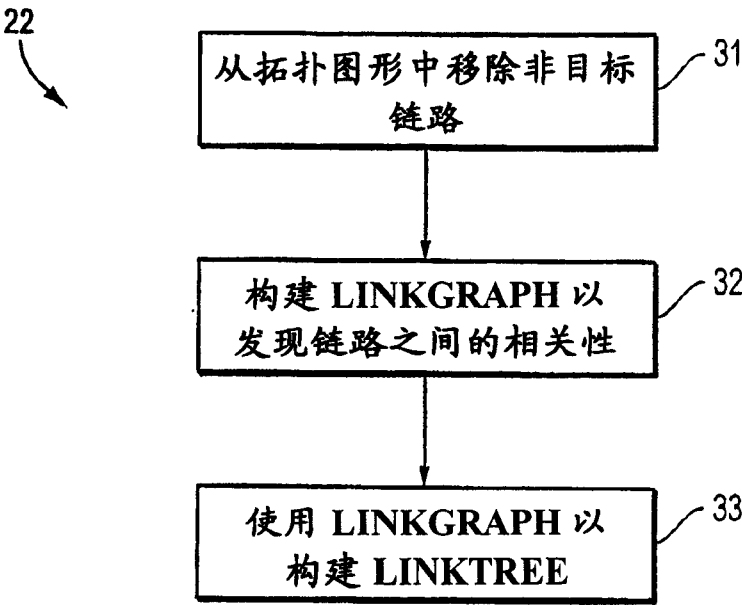


图 3

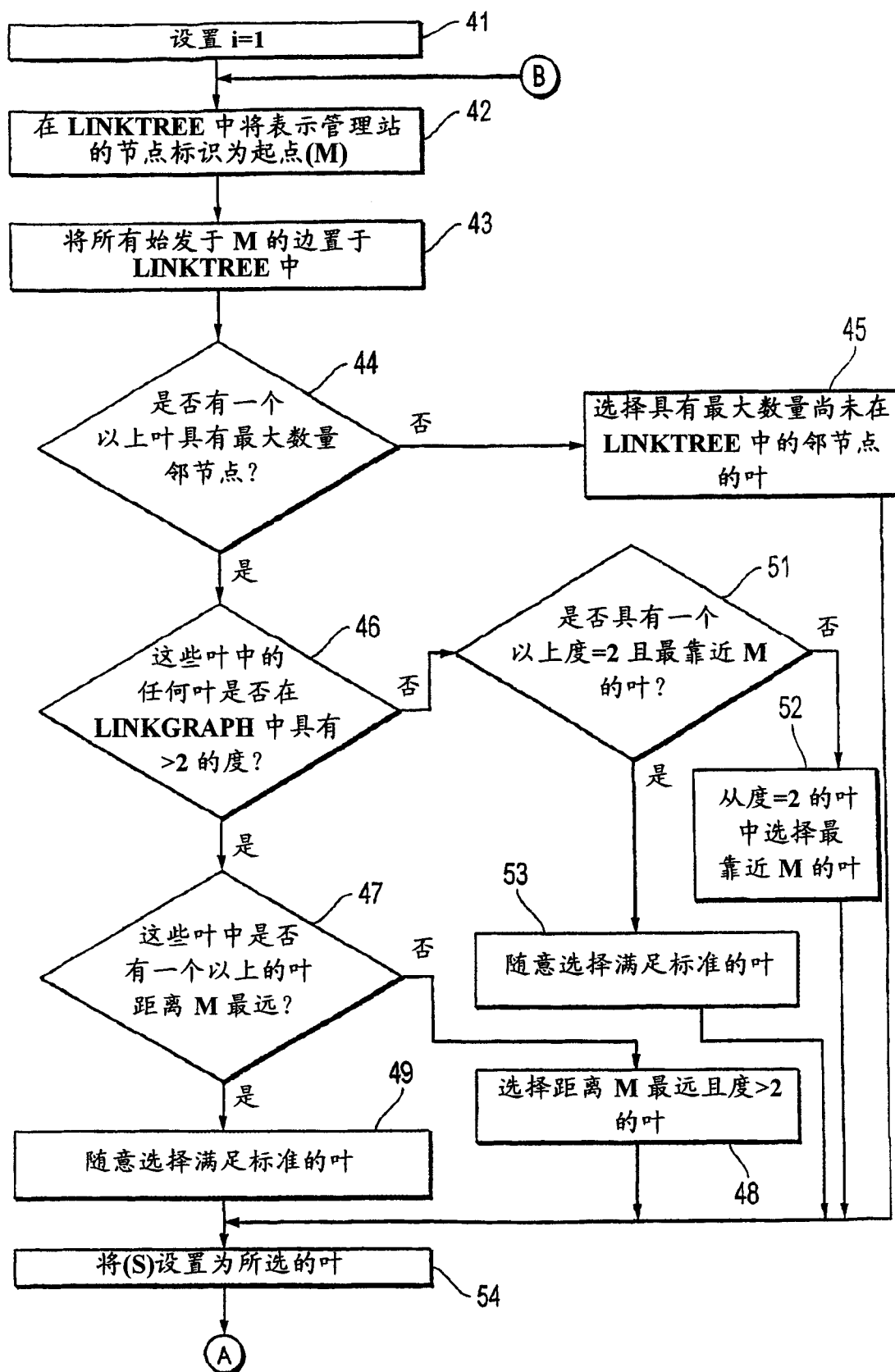


图 4A

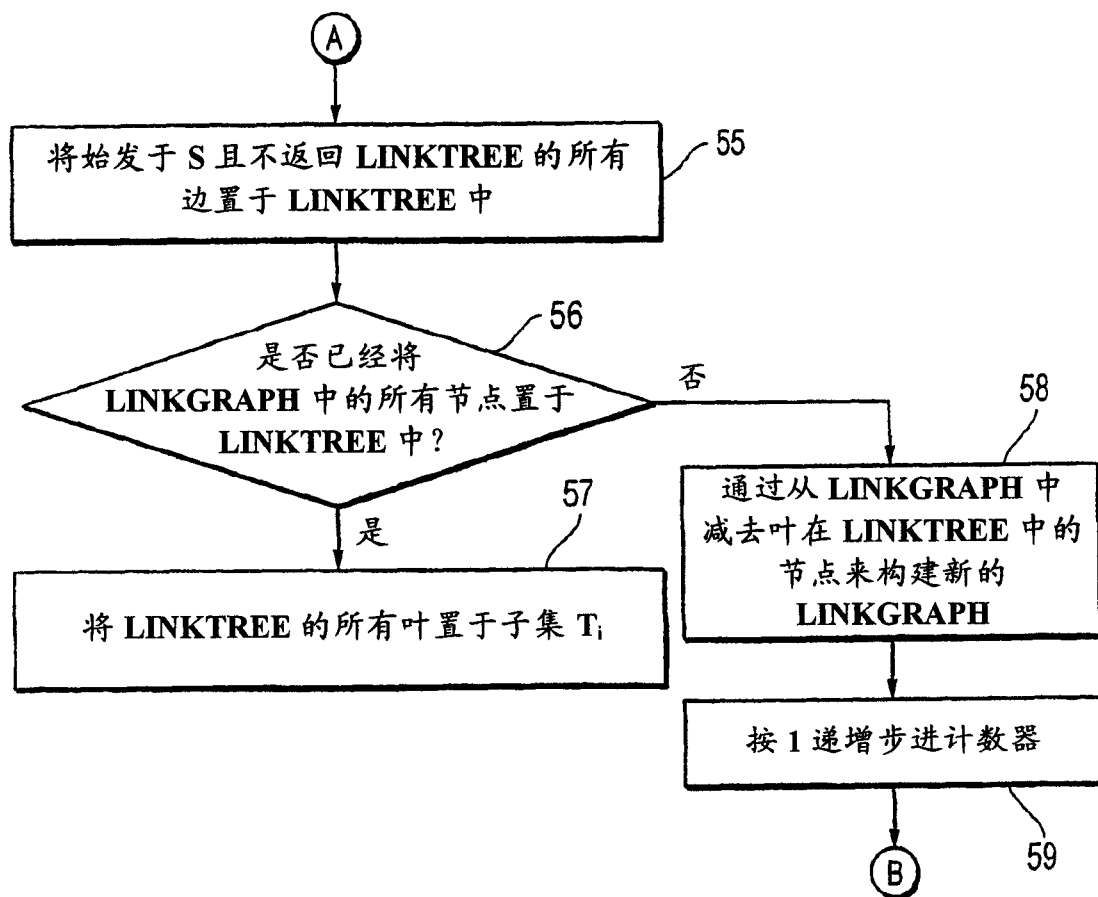


图 4B

图 5

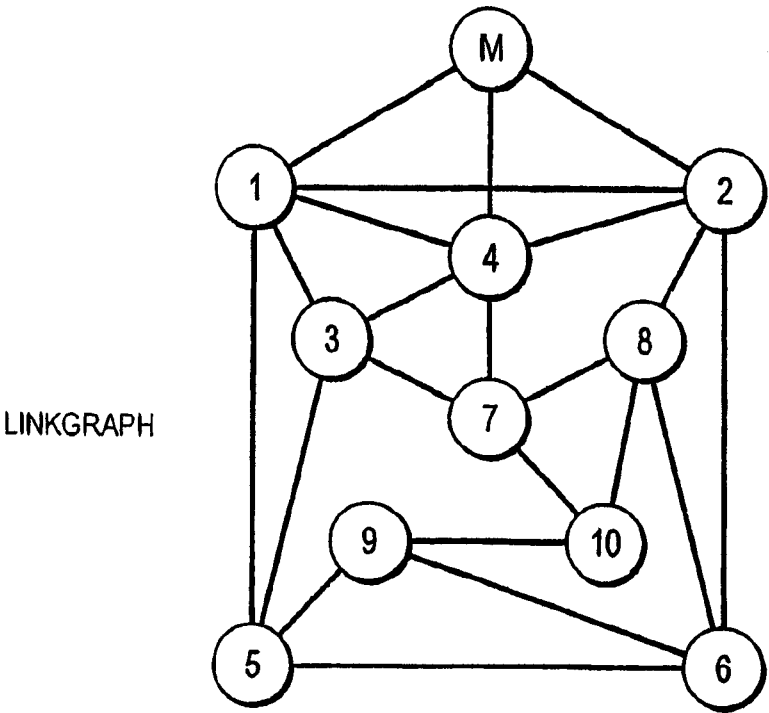
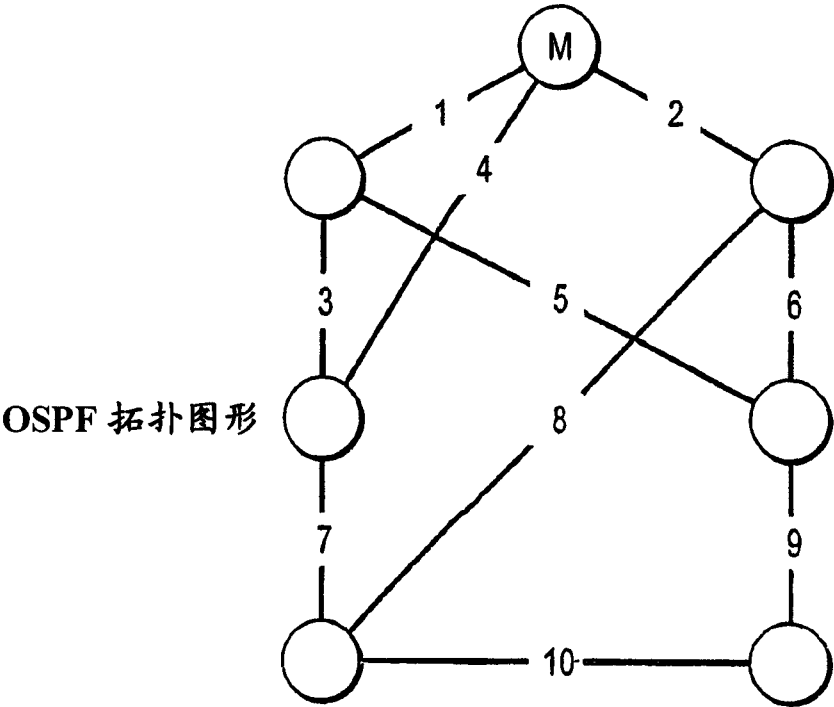


图 6

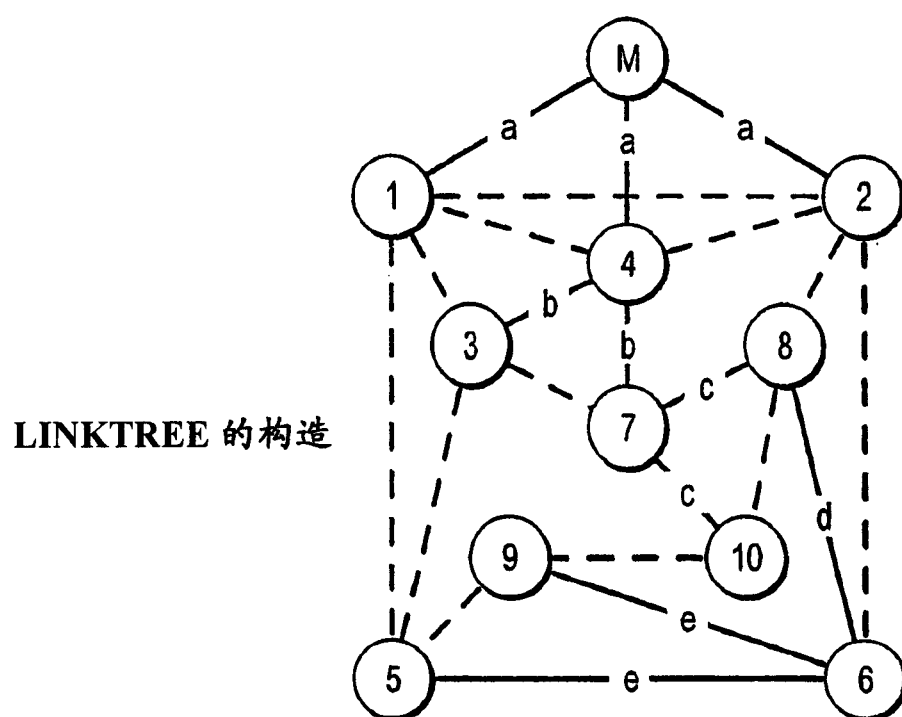
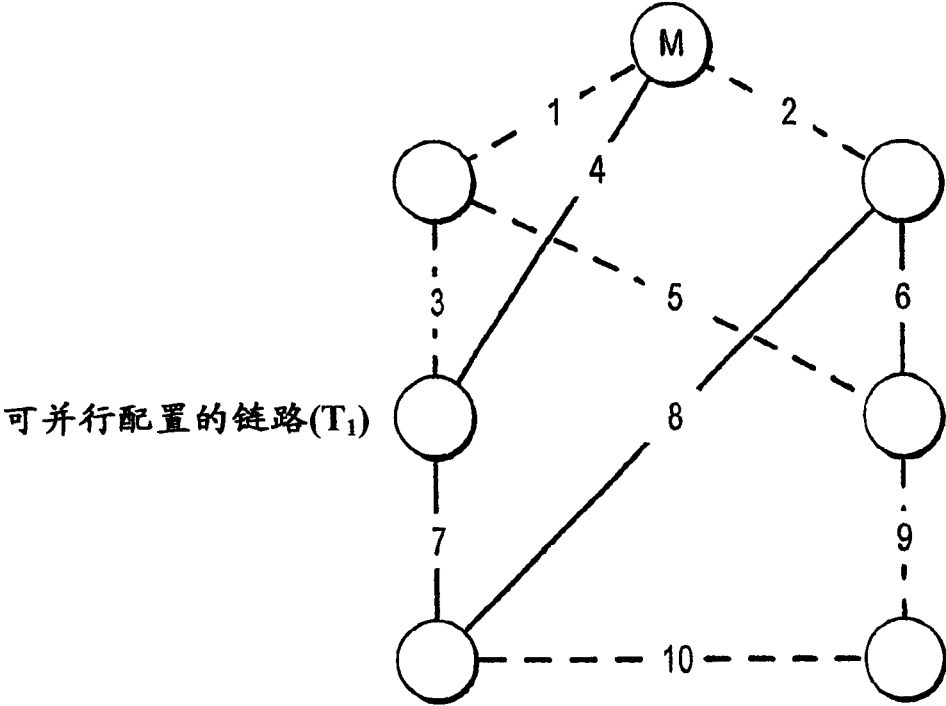


图 7

图 8



相邻链路的 OSPF 拓扑图形

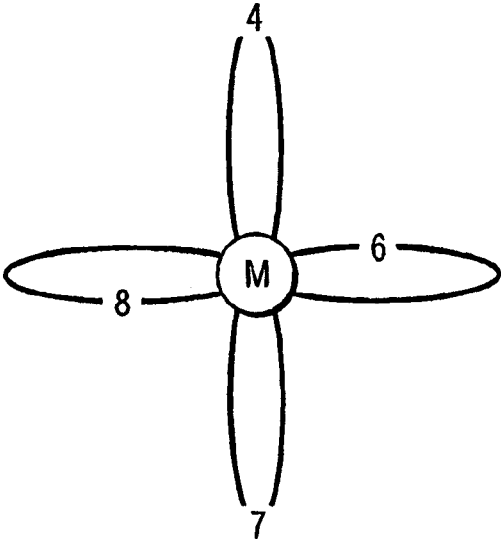


图 9

图 10

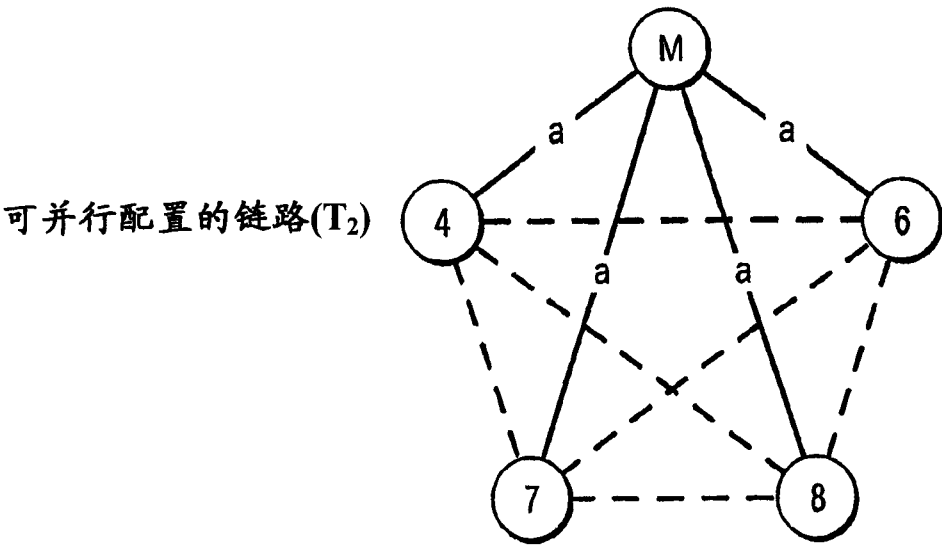
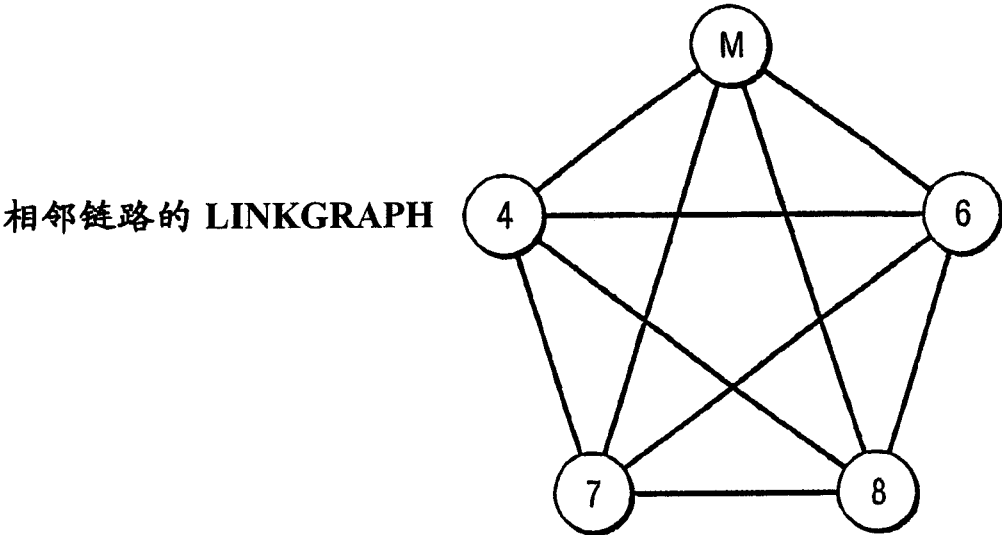


图 11

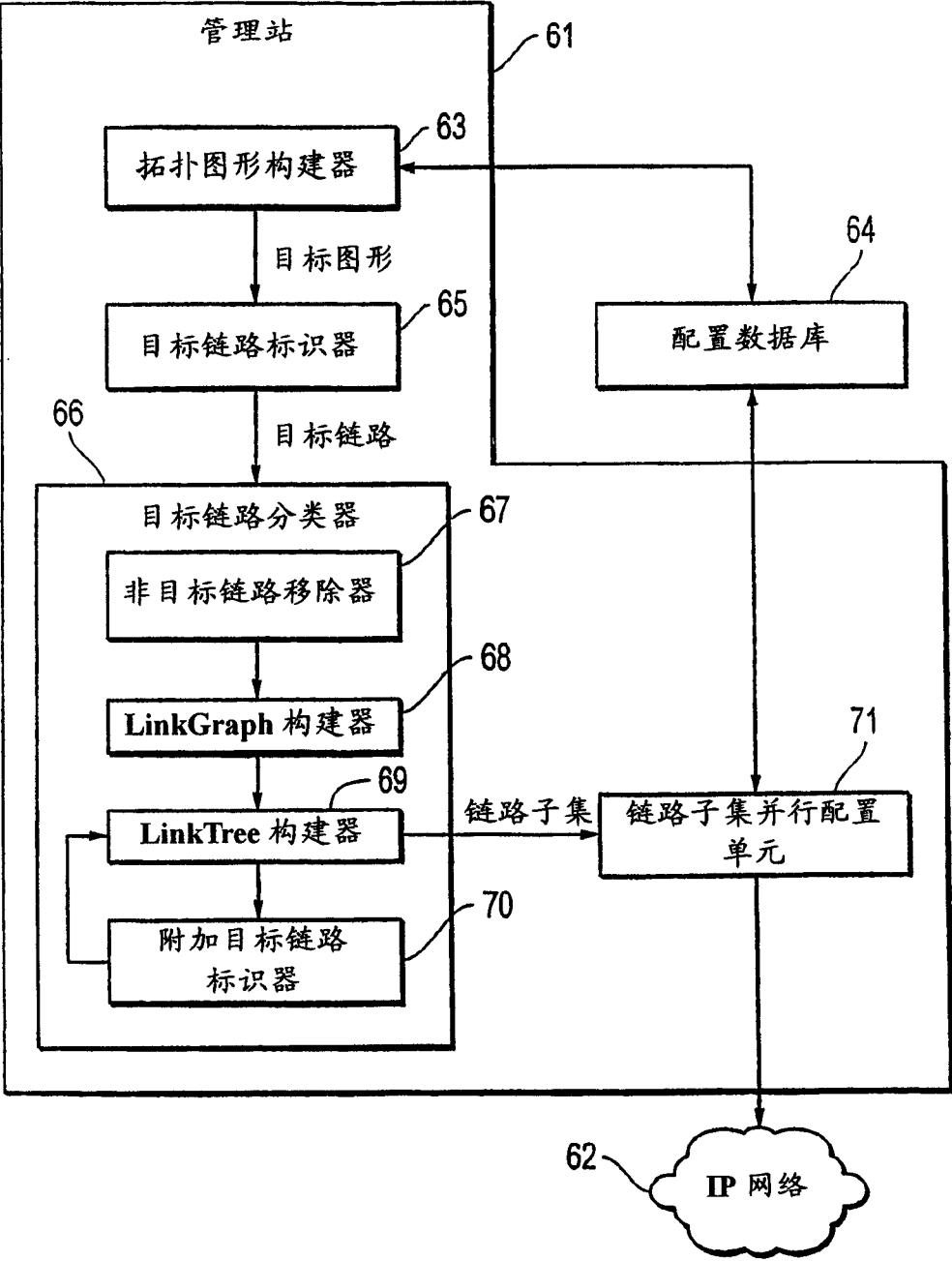


图 12