

[19] 中华人民共和国国家知识产权局

[ 51 ] Int. Cl<sup>7</sup>

H04L 12/18

H04L 12/56



# [12] 发明专利说明书

专利号 ZL 01819246.7

[45] 授权公告日 2005 年 9 月 21 日

[11] 授权公告号 CN 1220353C

[22] 申请日 2001.11.12 [21] 申请号 01819246.7

[30] 优先权

[32] 2000.11.21 [33] EP [31] 00811104.9

[86] 国际申请 PCT/IB2001/002078 2001.11.12

[87] 国际公布 WO2002/043324 英 2002.5.30

[85] 进入国家阶段日期 2003.5.20

[71] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 丹尼尔·鲍尔 约翰·戴戈

伊利亚斯·伊利亚迪斯

保罗·斯科顿

审查员 向琳

[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

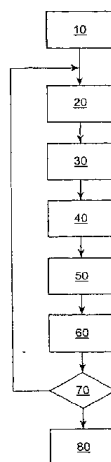
代理人 吴丽丽

权利要求书 3 页 说明书 20 页 附图 7 页

[54] 发明名称 计算数据网络中节点的有效路径成本的方法和设备

[57] 摘要

一种为数据通信网络中的一组节点得出一组有效路径成本的方法，所述一组节点包括一个源节点和若干目的地节点，其中在网络中确定所述一组节点中节点间路径的路径成本，并且每个路径成本包括限制成本和附加成本，所述方法包括首先迭代地确定带宽较高的路径，当存在两个或多个带宽相同的路径时，选择具有与之相关的较低转接延迟的路径。



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

1、一种为数据通信网络中的一组节点得出一组有效路径成本的方法，所述一组节点包括一个源节点和多个目的地节点，其中在网络中确定所述一组节点中节点间路径的路径成本，并且每个路径成本包括限制成本和附加成本，所述方法包括：

a) 把从源节点到任意目的地节点的直接路径的路径成本记录在第一数据集合中，在第一数据集合中，每个记录的成本与通过对应路径到达的节点相关；

b) 从第一数据集合中选择最佳路径成本，其中该最佳路径成本被确定为包含最佳限制成本的路径成本，或者，如果多个路径成本包括相同的限制成本，则最佳路径成本为具有最佳限制成本和最佳附加成本的路径成本；

c) 把在步骤 b) 中选择的最佳路径成本记录在第二数据集合中，在第二数据集合中，记录的成本与通过对应路径到达的节点相关；

d) 从第一数据集合中除去在步骤 c) 中记录在第二数据集合中的成本；

e) 在第一数据集合中，记录从通过对应于步骤 c) 中记录的成本的路径到达的目的地节点到该组节点中除源节点之外的任意其它节点的直接路径的累积路径成本；在第一数据集合中，每个记录的累积路径成本与通过对应路径到达的节点相关；

f) 对于在步骤 e) 中，将其相关成本记录在第一数据集合中的每个节点，比较第一和第二数据集合中与该节点相关的成本，并从第一数据集合除去和到达该节点的任意其它成本相比具有较差限制成本和较差附加成本的任意到达该节点的成本，或者除去和到达该节点的任意其它成本相比具有相同限制成本和较差附加成本的任意到达该节点的成本；

g) 重复步骤 b) -f)，直到在步骤 f) 之后，在第一数据集合中不留下任何成本为止；

从而所得到的第二数据集合包括该组有效路径成本。

2、按照权利要求 1 所述的方法，其中每个限制成本包括表示对应路径的带宽的数值；最佳限制成本是表示最大带宽的成本。

3、按照权利要求 1 或 2 所述的方法，其中每个附加成本包括表示与对应路径相关的转接延迟的数值，最佳附加成本是表示最小转接延迟的成本。

4、按照权利要求 1 或 2 所述的方法，其中至少一个路径包括多个链路和至少一个中间节点。

5、按照权利要求 3 所述的方法，其中至少一个路径包括多个链路和至少一个中间节点。

6、一种得出数据通信网络中的一组节点的一组有效路径成本的设备，所述一组节点包括一个源节点和多个目的地节点，其中在网络中确定该组节点中节点间路径的路径成本，并且每个路径成本包括限制成本和附加成本，该设备包括用于保存路径成本的第一数据集合和第二数据集合的存储器和控制装置，所述控制装置包括：

a) 第一记录装置，用于把从源节点到任意目的地节点的直接路径的路径成本记录在第一数据集合中，在第一数据集合中，每个记录的成本与通过对应路径到达的节点相关；

b) 选择装置，用于从第一数据集合中选择最佳路径成本，其中该最佳路径成本被确定为包含最佳限制成本的路径成本，或者如果多个路径成本包括相同的限制成本，则为具有最佳限制成本和最佳附加成本的路径成本；

c) 第二记录装置，用于把选择装置选择的最佳路径成本记录在第二数据集合中，在第二数据集合中，记录的成本与通过对应的路径到达的节点相关；

d) 用于从第一数据集合中除去第二记录装置中记录在第二数据集合中的成本的装置；

e) 第三记录装置，用于在第一数据集合中，记录从通过对应于第二记录装置中记录的成本的路径到达的目的地节点到该组节点中除源

节点之外的任意其它节点的直接路径的累积路径成本，在第一数据集合中，每个记录的累积路径成本与通过对应路径到达的节点相关；

f) 除去装置，用于对在第一数据集合中记录了相关的累积路径成本的每个节点，比较第一数据集合和第二数据集合中与该节点相关的成本，从第一数据集合中除去和到达该节点的任意其它成本相比具有较差限制成本和较差附加成本的任意到达该节点的成本，或者除去和到达该节点的任意其它成本相比具有相同限制成本和较差附加成本的任意到达该节点的成本；

g) 重复装置，用于重复 b) 至 f) 中所述各装置的操作，直到在第一数据集合中不留下任何成本为止；所得到的第二数据集合包括该组有效路径成本。

7、按照权利要求 6 所述的设备，其中每个限制成本包括表示对应路径的带宽的数值；最佳限制成本是表示最大带宽的成本。

8、按照权利要求 6 或 7 所述的设备，其中每个附加成本包括表示与对应路径相关的转接延迟的数值，最佳附加成本是表示最小转接延迟的成本。

9、按照权利要求 6 或 7 所述的设备，其中至少一个路径包括多个链路和至少一个中间节点。

10、一种数据连网设备，包括用于分别连接到数据通信网络的路径的多个端口，以及与路径耦接、如权利要求 6-9 任一所述的得出一组有效路径成本的设备。

11、一种数据网络，包含通过多个路径互连的多个节点，至少一个节点包含如权利要求 10 中所述的数据连网设备。

## 计算数据网络中节点的有效路径成本的方法和设备

### 技术领域

本发明涉及数据通信网络中的成本 (cost)，更具体地说，涉及为数据通信网络的一组节点得出一组有效路径成本 (path cost) 的方法、设备和计算机程序组件。

### 背景技术

对于不同种类的应用来说，诸如因特网之类大规模网络中的服务质量 (QoS) 的支持正变得越来越重要。为了确保 QoS，连接最终用户的路径必须满足指定的要求。于是在 QoS 保证的规定中，路由判定是一项重要的考虑因素。于是，路由选择协议应知道网络中各种链路支持的通信特性。

为诸如“开放最短路径优先” (OSPF) 协议之类众所周知的路由选择协议提出了 QoS 扩展。在“Implementation and Performance Measurements of QoS Routing Extensions to OSPF”, Apostolopoulos et al. Proceeding of IEEE INFOCOM'99, pp.75-83, New York, March, 1999 中描述了这种扩展的例子。OSPF 协议是链路状态路由选择协议的一个例子。众所周知，在大型网络中，这种协议不能有效工作。从而，为了可缩放性起见，OSPF 网络被分成几个路由区。在“OSPF Version 2”, Moy, Internet Engineering Task Force(IETF) Request for Comments(RFC)2328, April 1998 中进一步描述了 OSPF 协议。

本领域中众所周知的另一种类似的路由选择协议是在“Private Network Interface Specification, Version 1.0”, the ATM Forum, March 1996 中描述的专用网络间接口 (PNNI) 协议。根据 PNNI 协议，网络被分成几群，从而形成路由层次。

为了提供全局准确的路径选择，应考虑所涉及的每个路由域（不论它是 OSPF 区，还是 PNNI 群）的通过特性。“ On Scalable QoS Routing, Performance Evaluation of Topology Aggregation”, Hao et al, Proceedings of IEEE INFOCOM'2000, Tel Aviv, Israel, March 2000和“ QoS Routing: a Precomputation Perspective”, Orda et al, Proceeding of IEEE INFOCOM'99 pp.128-136, Tel Aviv, Israel, March 2000都建议借助网络的层次组织，可显著增加路径计算效率。

“ Transition Matrix Generation for Complex Node Representation”, Iliadis et al, Proceedings of the IEEE ATM Workshop'99, pp489-500, Kochi, Japan, May 1999中说明的是一种利用转换矩阵表示路由域的通过特性的方法，所述转换矩阵给出路由域的每对入口-出口节点的路径成本。转换矩阵中的每一项是对应一对入口-出口节点的通过特性的最小表示（minimal representation）。可利用限制成本度量，附加成本度量，或者这两者描述通过特性（traversing characteristics）。只利用一种度量描述的转换特性可被看作一维。利用 N 种度量描述的转换特性可被看作 N 维。路径的附加度量是路径中所有链路的所有附加度量的和。路径的限制度量是路径中所有链路的限制度量中的最小者。附加度量的例子包括延迟和管理成本。限制度量的例子包括带宽。本领域中，通过特性的最小表示被称为有效边界（frontier）。下面简要说明计算有效边界的常规技术。可按照两种标准对这些技术分类：

计算结果：一些技术计算网络中从一个源入口-出口节点到所有其它入口-出口节点的有效边界。其它技术计算所有各对入口-出口节点之间的有效边界。要认识到属于前一类别的技术必须应用于所有的入口-出口节点。

延迟功能：一些技术只支持固定延迟功能。其它技术既可应用于固定延迟，又可应用于取决于带宽的延迟。

下述三种众所周知的算法可被用于计算最小表示：

在“ Efficient Frontier Formulation for Additive and Restrictive

Metrics in Hierarchical Routing”, Bauer et al, Proceedings of the IEEE ICC'00, pp.1353-59, New Orleans, June 2000 中描述的 Multi-Dijkstra 算法;

在 “ Introduction to Algorithms”, Corment et al. the MIT Electrical Engineering and Computers Science Series, MIT Press, ISBN 0-262-031412-8, 1989 中, 在 “ On a Routing Problem”, Bellman, Quarterly of Applied Mathematics, Vol.16, No.1, pp.87-90, 1958 中, 以及在 “ Flows in Networks”, Ford Jr. et al. Princeton University Press, 1962 中描述的 Bellman-Ford 算法; 和

在 “ Algorithm 97(Shortest Path)”, Floyd, Communications of the ACM, Vol.5, No.6, 1965 中描述的 Floyd-Warshall 算法。

所有这三种算法都适用于恒定延迟。但是, 只有 Bellman Ford 和 Floyd Warshall 算法能够处理取决于带宽的延迟。另外, 只有 Multi-Dijkstra 和 Bellman-Ford 算法能够为单源节点对所有节点(one source to all nodes) 解决有效边界问题。对于计算有效边界来说, Multi-Dijkstra 和 Floyd-Warshall 一般比 Bellman-Ford 算法简单。于是应根据应用选择这些算法。例如, 如果只存在少数入口-出口顶点, 并且只考虑恒定延迟, 则 Multi-Dijkstra 算法优于 Floyd -Warshall 算法。

一般来说, 计算从一个节点到路由域的所有其它节点的有效边界的常规技术计算复杂, 应用有限。希望提供一种更简单的计算这种有效边界的方法。另外, 希望提供一种解决“单源节点对所有节点”问题的方法。

## 发明内容

根据本发明, 提供一种为数据通信网络中的一组节点得出一组有效路径成本的方法, 所述一组节点包括一个源节点和若干目的地节点, 其中在网络中确定所述一组节点中节点间路径的路径成本, 并且每个路径成本包括限制成本和附加成本, 所述方法包括:

a) 把从源节点到任意目的地节点的直接路径的路径成本记录在第一数据集合中，在第一数据集合中，每个记录的成本与通过对应路径到达的节点相关；

b) 从第一数据集合中选择最佳路径成本，其中该最佳路径成本被确定为包含最佳限制成本的路径成本，或者，如果若干路径成本包括相同的限制成本，则最佳路径成本为具有最佳限制成本和最佳附加成本的路径成本；

c) 把在步骤 b) 中选择的最佳路径成本记录在第二数据集合中，在第二数据集合中，记录的成本与通过对应路径到达的节点相关；

d) 从第一数据集合中除去在步骤 c) 中记录在第二数据集合中的成本；

e) 在第一数据集合中，记录从通过对应于步骤 c) 中记录的成本的路径到达的目的地节点到该组节点中除源节点之外的任意其它节点的直接路径的累积路径成本；在第一数据集合中，每个记录的累积路径成本与通过对应路径到达的节点相关；

f) 对于在步骤 e) 中，将其相关成本记录在第一数据集合中的每个节点，比较第一和第二数据集合中与该节点相关的成本，并从第一数据集合除去和到达该节点的任意其它成本相比具有较差限制成本和较差附加成本的任意到达该节点的成本，或者除去和到达该节点的任意其它成本相比具有相同限制成本和较差附加成本的任意到达该节点的成本；

g) 重复步骤 b) -f)，直到在步骤 f) 之后，在第一数据集合中不留下任何成本为止；

从而所得到的第二数据集合包括该组有效路径成本。

这提供了一种更简便并且更快速的计算有效边界的方法。另外，该方法简化了“单源节点对所有节点”问题的求解。此外，该方法既适合于恒定延迟，又适合于取决于带宽的延迟。

每个限制成本最好包括表示对应路径的带宽的数值；最佳限制成本是表示最大带宽的成本。每个附加成本最好包括表示与对应路径相

关的转接延迟的数值，最佳附加成本是表示最小转接延迟的成本。至少一个路径可包括若干链路和至少一个中间节点。

本发明还延伸到一种包括计算机程序代码装置的计算机程序组件，所述计算机程序代码当被加载到数据处理系统的处理器中时，配置处理器，以便执行如前所述的得出一组有效路径成本的方法。

从另一方面看本发明，提供一种得出数据通信网络中的一组节点的一组有效路径成本的设备，所述一组节点包括一个源节点和若干目的地节点，其中在网络中确定该组节点中节点间路径的路径成本，每个路径成本包括限制成本和附加成本，该设备包括用于保存路径成本的第一数据集合和第二数据集合的存储器和控制装置，所述控制装置包括：

a) 第一记录装置，用于把从源节点到任意目的地节点的直接路径的路径成本记录在第一数据集合中，在第一数据集合中，每个记录的成本与通过对应路径到达的节点相关；

b) 选择装置，用于从第一数据集合中选择最佳路径成本，其中该最佳路径成本被确定为包含最佳限制成本的路径成本，或者如果多个路径成本包括相同的限制成本，则为具有最佳限制成本和最佳附加成本的路径成本；

c) 第二记录装置，用于把选择装置选择的最佳路径成本记录在第二数据集合中，在第二数据集合中，记录的成本与通过对应的路径到达的节点相关；

d) 用于从第一数据集合中除去第二记录装置中记录在第二数据集合中的成本的装置；

e) 第三记录装置，用于在第一数据集合中，记录从通过对应于第二记录装置中记录的成本的路径到达的目的地节点到该组节点中除源节点之外的任意其它节点的直接路径的累积路径成本，在第一数据集合中，每个记录的累积路径成本与通过对应路径到达的节点相关；

f) 除去装置，用于对在第一数据集合中记录了相关的累积路径成本的每个节点，比较第一数据集合和第二数据集合中与该节点相关的

成本，从第一数据集合中除去和到达该节点的任意其它成本相比具有较差限制成本和较差附加成本的任意到达该节点的成本，或者除去和到达该节点的任意其它成本相比具有相同限制成本和较差附加成本的任意到达该节点的成本；

g) 重复装置，用于重复 b) 至 f) 中所述各装置的操作，直到在第一数据集合中不留下任何成本为止；所得到的第二数据集合包括该组有效路径成本。

本发明延伸到包括分别连接数据通信网络的路径的若干端口，以及如前所述得出一组有效路径成本的设备的数据连网设备。本发明还延伸到数据网络，包含通过若干路径互连的若干节点，至少一个节点包含这样的数据连网设备。

#### 附图说明

下面将参考附图，举例说明本发明的优选实施例，其中：

图 1 图解说明恒定延迟路径的有效边界的一个例子；

图 2a 是具有四个入口-出口节点的例证网络的简化图；

图 2b 图解说明了对应于群聚网络的转换矩阵；

图 3-13 是对应于该网络的定向图；

图 14 图解说明了从网络的节点 0 到节点 1 的有效边界；

图 15 图解说明了从网络的节点 0 到节点 2 的有效边界；

图 16 图解说明了从网络的节点 0 到节点 3 的有效边界；

图 17 是对应于根据本发明的得出一组有效路径成本的方法的例子的流程图；

图 18 是具体体现本发明的数据连网设备的方框图。

#### 具体实施方式

在继续说明本发明之前，定义和解释相关术语及措辞。

这里，措辞“限制成本”用于描述随链路的大小或特征，例如带宽而变化的成本。限制成本  $C$  可被定义成  $C = \text{Max-带宽}$ ，或者定义成  $C = 1/\text{带宽}$ 。根据限制成本的定义，路径的最弱链路确定限制成本。与

限制成本相反的是附加成本。附加成本取决于，例如链路的延迟。

这里，单词“节点”或者“顶点”用作路由器、交换机、桥接器、桥式路由器、集线器，以及通信网络中传送或接收信息的任意其它连网设备的类属术语。

可借助定向图模拟网络。使用下述约定：

网络的节点被称为定向图的顶点。

两个网络节点之间的链路被称为定向图的两个顶点之间的直接路径或者棱边 (edge)。

假定  $G(V, E)$  为指定时刻时的网络的定向图。 $V$  是一组顶点， $E$  是一组定向棱边。于是对于所有  $v_i, v_j \in V$  (如果  $v_i$  和  $v_j$  是连接的)，存在棱边  $v_i \xrightarrow{\varepsilon_{v_i, v_j}} v_j \in E$ 。可利用多个平行棱边连接顶点。连接两个顶点  $v_i$  和  $v_j$  的一组棱边被表示成  $E_{v_i, v_j} \triangleq \{\varepsilon_{v_i, v_j}^0, \varepsilon_{v_i, v_j}^1, \dots, \varepsilon_{v_i, v_j}^n\}$ 。符号  $\varepsilon_{v_i, v_j}$  代表连接顶点  $v_i$  和  $v_j$  的任意棱边。

假定  $s$  和  $t$  为定向图  $G(V, E)$  的两个顶点。假定  $P_{s,t}$  为连接  $s$  和  $t$  的一组路径  $s \xrightarrow{P_{s,t}} t$ 。如果不存在从  $s$  到  $t$  的路径，则  $P_{s,t} = \phi$ 。长度  $\|\rho\| = n$  的路径  $\rho \in P_{s,t}$  是一系列  $\{\varepsilon_{v_0, v_1}, \varepsilon_{v_1, v_2}, \dots, \varepsilon_{v_{n-1}, v_n}\}$  的  $n$  条棱边，从而

$$v_0 = s$$

$$v_n = t$$

$$\varepsilon_{v_i, v_{i+1}} \in E_{v_i, v_{i+1}} \quad \forall i \in [0, \dots, n-1].$$

如果在  $v = v_i$  或  $v = v_{i+1}$  的情况下，存在棱边  $\varepsilon_{v_i, v_{i+1}} \in \rho$ ，则认为顶点  $v$  是路径  $\rho$  的一部分。

关于路径和棱边的度量提供和网络中的可用资源有关的信息。现在定义可用带宽及转接延迟的度量。但是，要认识到这里给出的意见适用于符合这里提供的定义的任意限制度量和附加度量。

对于棱边  $\varepsilon_{v_i, v_j} \in E$ ，可用带宽将用  $B(\varepsilon_{v_i, v_j}) \in \mathbf{R}^+$  表示。可用带宽被认为是限制性的，因为路径的可用带宽是该路径通过的棱边的可用带宽中的最小者。对于长度为  $n$  的从  $s$  到  $t$  的可行路径  $\rho$  来说，可用带宽为

$$B(\rho) = \min_{\varepsilon \in \rho} \{B(\varepsilon)\} \quad (1)$$

假定，棱边的转接延迟是被请求带宽  $b$  的函数。从而，对于棱边

$\varepsilon \in E$  和被请求带宽  $b$  来说, 转接延迟由  $D(\varepsilon, b) \in \mathbf{R}^+$  给出。如果  $b > B(\varepsilon)$ , 则  $D(\varepsilon, b) = \infty$ 。转接延迟被认为是累积的, 因为路径的转接延迟是通过的棱边的转接延迟之和。对于长度为  $n$  的从  $s$  到  $t$  的可行路径来说, 转接延迟为

$$D(\rho, b) = \sum_{\varepsilon \in \rho} D(\varepsilon, b) \quad (2)$$

对于棱边  $\varepsilon \in \rho$ , 如果被请求带宽  $b > B(\varepsilon)$ , 则  $D(\rho, b) = \infty$ 。否则,  $D(\rho, b)$  有限。

下面, 定义一对入口-出口顶点的二维转换特征。转换特征的维是带宽和延迟。随后检查相对于一组入口-出口顶点的扩展。

研究前面提及的代表一个路由域的定向图  $G(V, E)$ 。假定  $s, t \in V$  为两个入口-出口顶点。做出路由判定所关心的从  $s$  到  $t$  的通过 (traversing) 度量是最大可用带宽  $B^{\max}$ , 以及被请求带宽  $b$  的最大转接延迟函数  $D^{eff}(b)$ 。这两者都取决于连接  $s$  和  $t$  的一组路径  $P_{s,t}$ 。具体地说,

$$B_{s,t}^{\max} \stackrel{\Delta}{=} \max_{\rho \in P_{s,t}} (B(\rho)), \text{ 和} \quad (3)$$

$$D_{s,t}^{eff}(b) \stackrel{\Delta}{=} \min_{\rho \in P_{s,t}} \{D(\rho, b)\} \quad (4)$$

根据可用带宽和转接延迟的定义, 得出结论, 对于  $b \leq B_{s,t}^{\max}$ , 则  $D_{s,t}^{eff}(b) < \infty$ , 并且对于  $b > B_{s,t}^{\max}$ , 则  $D_{s,t}^{eff}(b) = \infty$ 。于是, 由于  $D_{s,t}^{eff}$  代表对于每个带宽, 具有最小延迟的路径, 因此  $D_{s,t}^{eff}$  足以描述入口-出口对  $s, t$  的通过度量。  $D_{s,t}^{eff}$  还确定适用带宽及从  $s$  到  $t$  的转接延迟的有效边界。根据定义,  $D_{s,t}^{eff}$  取决于路径组  $P_{s,t}$ 。但是, 只有路径组  $P_{s,t}$  的子集对有效边界产生影响。下面将该子集称为  $s$  和  $t$  的有效路径或者  $P_{s,t}^E$

$$P_{s,t}^E \stackrel{\Delta}{=} \left\{ \rho \in P_{s,t} \mid \exists b : D(\rho, b) < \infty \wedge D(\rho, b) = D_{s,t}^{eff}(b) \right\} \quad (5)$$

有效边界可被看成一系列的分段, 每一段对应于路径的传输延迟函数

$$D_{s,t}^{eff}(b) = \left\{ \begin{array}{l} D(\rho_1, b) \forall b \in (b_0, b_1) \\ D(\rho_2, b) \forall b \in (b_1, b_2) \\ \vdots \\ D(\rho_n, b) \forall b \in (b_{n-1}, b_n) \\ \infty \forall b > b_n \end{array} \right\} \quad (6)$$

其中  $b_n = B_{s,t}^{\max}$ ,  $b_0 = 0$ , 在  $i \in [1, n]$  的情况下, 如下给出  $\rho_i$ :

$$\rho_i = P_{s,t}^E, \text{ 从而 } D(\rho_i, b) = D_{s,t}^{\text{eff}}(b) \forall b \in [b_{i-1}, b_i]. \quad (7)$$

注意  $\rho_i$  可以不是唯一的, 因为可能存在满足等式(7)的多个路径。另外, 对于一些形式的延迟函数来说, 可能发生单一路径影响多个分段的情况。例如, 在等式(6)中, 对于一些  $i \neq j$ , 有可能  $\rho_i \equiv \rho_j$ 。

现在参见图 1, 图中举例说明了恒定延迟路径的有效边界的一个例子。这些路径是  $D(\rho, b)$  与  $b$  无关, 从而可被写成  $D(\rho)$  的路径。本例中,  $P_{s,t}^E$  包括四个路径。这些路径的转接延迟-带宽特征被表示成有效边界上的黑点。这些有效边界的可用带宽和转接延迟完整地确定  $D_{s,t}^{\text{eff}}$ 。阴影区还包括不是  $P_{s,t}^E$  一部分的无效解答 (solution), 这些无效解答被表示成白点。对于恒定延迟路径, 能够如下定义可实现的最小延迟

$$D_{s,t}^{\min} = \min_{\rho \in P_{s,t}^E} \{D(\rho)\} \quad (8)$$

一般情况下, 对于被请求的带宽  $B_{s,t}^{\max}$ , 不可能实现  $D_{s,t}^{\min}$ 。只有当路径  $\rho \in P_{s,t}^E$  的延迟  $D(\rho)$  都相等时, 才可能实现  $D_{s,t}^{\min}$ 。例如, 如果  $P_{s,t}^E$  只包含单一路径, 则情况就是这样。于是, 对于任意网络, 如果  $D_{s,t}^{\text{eff}}(B_{s,t}^{\max}) = D_{s,t}^{\min}$ , 则存在“单一路径解答”。

现在假定定向图包括称为  $R_0, \dots, R_{N-1} \in V$  的  $N$  个入口-出口顶点。对于每对入口-出口顶点  $R_i, R_j$ , 存在一个有效边界  $D_{R_i, R_j}^{\text{eff}}(b)$ 。该组所有有效边界完整地确定网络的通过度量。用称为转换矩阵的矩阵给出任意一对入口-出口顶点之间的所有有效边界的代数表达。可如下规定该转换矩阵:

$$M = \begin{Bmatrix} - & D_{R_0, R_1}^{\text{eff}}(b) & \dots & D_{R_0, R_{N-1}}^{\text{eff}}(b) \\ D_{R_1, R_0}^{\text{eff}}(b) & - & \dots & D_{R_1, R_{N-1}}^{\text{eff}}(b) \\ \vdots & \vdots & \ddots & \vdots \\ D_{R_{N-1}, R_0}^{\text{eff}}(b) & D_{R_{N-1}, R_1}^{\text{eff}}(b) & \dots & - \end{Bmatrix} \quad (9)$$

转换矩阵是基础路由域的布局的代数表示。在它完整保持关于带宽和延迟度量的通过特性的意义上, 它是路由域的准确表示。

如前所述, 本发明解决的问题涉及转换矩阵的计算。具体地说, 该问题涉及转换矩阵每一项的计算。参考图 2a, 研究包含四个入口-出口顶点 0、1、2 和 3 的例证网络。参见图 2b, 每一项是每对入口-

出口顶点之间的有效边界。

在本发明的一个优选实施例中，并行地并且以“降序”方式迭代确定对应于各个顶点的有效边界。“降序”意指首先找出带宽较大的有效边界点，在具有相同带宽的点之中，首先找出延迟较小的点。通过考虑有效边界的图形表示（如前所述，延迟沿 y 轴，带宽沿 x 轴），这种方法能够从右（较高的带宽）向左（较低的带宽）有效地清除有效边界。下面简要地详细说明根据本发明确定有效边界的程序的优选例子。不过，预先给出一些预备定义。

### 预备定义

假定  $G(V, E)$  为代表指定时刻时的网络的图。V 是一组顶点，E 是一组定向棱边。于是对于所有  $v_i, v_j \in V$ （如果  $v_i$  和  $v_j$  是连接的），存在棱边  $v_i \xrightarrow{\varepsilon_{v_i, v_j}} v_j \in E$ 。可利用多个平行棱边连接顶点。连接两个顶点  $v_i$  和  $v_j$  的一组棱边被表示成  $E_{v_i, v_j} \triangleq \{\varepsilon_{v_i, v_j}^0, \varepsilon_{v_i, v_j}^1, \dots, \varepsilon_{v_i, v_j}^n\}$ 。符号  $\varepsilon_{v_i, v_j}$  代表连接顶点  $v_i$  和  $v_j$  的任意棱边。对于棱边  $\varepsilon_{v_i, v_j} \in E$ ，其用带宽和延迟来表示的度量（成本）将由  $C(\varepsilon_{v_i, v_j}) \triangleq (B(\varepsilon_{v_i, v_j}), D(\varepsilon_{v_i, v_j}))$  表示，其中  $B(\varepsilon_{v_i, v_j})$  和  $D(\varepsilon_{v_i, v_j})$  分别表示与该棱边相关的可用带宽和延迟。假定这些度量是非负实数。换句话说， $B(\varepsilon_{v_i, v_j}) \in \mathbf{R}^+$ ，并且  $D(\varepsilon_{v_i, v_j}) \in \mathbf{R}^+$ 。

如下比较两个成本  $C_i = (B_i, D_i)$  和  $C_j = (B_j, D_j)$ ：

当且仅当  $B_i = B_j$ ，并且  $D_i = D_j$  时，成本  $C_i$  等于  $C_j$  ( $C_i = C_j$ )。否则，成本  $C_i$  不等于  $C_j$  ( $C_i \neq C_j$ )。

当且仅当  $B_i \geq B_j$ ， $D_i \leq D_j$ ，并且 ( $C_i \neq C_j$ ) 时，成本  $C_i$  好于  $C_j$ ，或者说，成本  $C_j$  差于  $C_i$ 。

当且仅当  $B_k = \min(B_i, B_j)$  并且  $D_k = D_i + D_j$  时，成本  $C_k$  扩大成本  $C_i$  和  $C_j$  ( $C_k = \text{ext}(C_i, C_j)$ )。

### 过程

假定  $v_0$  是要从其计算有效边界的源顶点。假定  $r_i, i=1 \dots N-1$  为从  $v_0$  到达顶点  $v_i$  的成本列表。假定  $f_i, i=1 \dots N-1$  为与顶点  $v_i$  相关的有效边界。假定  $\mathfrak{R}$  和  $\mathfrak{S}$  分别为包含对应列表的数据集，即  $\mathfrak{R} = \{r_1, \dots, r_{N-1}\}$  和  $\mathfrak{S} = \{f_1, \dots, f_{N-1}\}$ 。这些数据可被排列成表格。

在继续说明之前，给出下述辅助定义：

$i$  为迭代计数值；

$v_{m(i)}$  为第  $i$  次迭代的标记顶点；

$E_{v_j}$  是从顶点  $v_j$  发出的一组棱边；

$r_j$  是从  $v_0$  到达顶点  $v_j$  的成本列表；

$f_j$  是包含和顶点  $v_j$  对应的有效边界的列表；

下面是呈一系列规则 1) — 3) 形式的过程的伪代码表示。

初始化：

$i=1$

$m(1)=0$

$f_0(\infty, 0)$

$\mathcal{R} = \phi$

$\mathcal{S} = \phi$

第  $i$  步迭代：

1) 对于集合  $E_{v_{m(i)}}$  中的每个棱边  $\varepsilon_{v_{m(i)}, v_j}$ ，**do**

a) 根据  $f_{m(i)}$  列表的最后一项和成本  $C(\varepsilon_{v_{m(i)}, v_j})$ ，计算扩展路径  $v_0 \rightarrow v_{m(i)} \xrightarrow{\varepsilon_{v_{m(i)}, v_j}} v_j$  的成本  $C_j = (B_j, D_j)$ ，即， $C_j = \text{ext}(f_{m(i)}, C(\varepsilon_{v_{m(i)}, v_j}))$ 。

b) **if** 等于或差于列表  $f_j$  的任意项，**then** 丢弃  $C_j$ ，

c) **else if**  $C_j$  等于或差于列表  $r_j$  的任意项，**then** 丢弃  $C_j$ ，

**else do**

d) 通过使列表继续按照带宽，以及延迟（对于相同带宽的项）分类，把  $C_j$  输入列表  $r_j$ 。

e) 丢弃列表  $r_j$  的差于  $C_j$  的任意项。

**done**

2) **if**  $\mathcal{R}$  数据集为空，**then** STOP。

3) **else do**

a) 选择  $\mathcal{R}$  数据集的最佳项，假定包含在列表  $r_k$  中。最佳项可被认为是具有最大带宽的一项，并在存在多项的情况下，具有最低延迟的一项。根据另一标准，例如中继段计数，或者随意地打破剩

余线路。

b) 从列表  $r_k$  中除去该项，并将其放入列表  $f_k$ 。

c) 增大计数值:  $i=i+1$ 。

d) 设置  $m(i) = k$  (从而  $v_{m(i)} = v_k$ )

**done**

返回步骤。

例子

参见图 3，现在说明具体体现本发明，确定从顶点 0 到例证的四节点网络的剩余顶点 1、2 和 3 的有效边界的方法的例子。要认识到本发明同样适用于具有四个以上或者四个以下节点的网络。

步骤 1:

现在参见图 4，该方法从顶点 0 开始，并且考虑从顶点 0 发出的所有棱边。三个棱边分别提供与顶点 1、2 和 3 的连接。目前不存在与顶点 0 相关的成本。于是，按照上述规则 1d) 把相对于顶点 1、2 和 3 的相应棱边的成本插入  $\mathfrak{R}$  数据集合中。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & 8,10 & 9,5 \end{Bmatrix}$$

在  $\mathfrak{R}$  数据集合中，用带宽表示的最佳成本是到顶点 3 的成本：9，5。随后按照规则 3b)，把该成本从  $\mathfrak{R}$  数据集合转移为顶点 3 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & 8,10 & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \end{Bmatrix}$$

该方法现在转移到具有相关成本 9，5 的顶点 3。

步骤 2:

现在参见图 5，顶点 3 具有去往顶点 1 和 2 的两个棱边。图 5 中表示了由与设置在  $\mathfrak{S}$  数据集合中的顶点 3 (9，5) 相关的成本扩展的这些棱边的成本。由于在  $\mathfrak{S}$  数据集合中仍然不存在关于顶点 1 和 2 的项目，因此这些成本被输入  $\mathfrak{R}$  数据集合。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & 8,10 & - \\ 6,9 & 7,6 & - \end{Bmatrix}$$

最宽的带宽值对应于顶点 2: 8, 10。从  $\mathfrak{R}$  数据集合把该值转移为顶点 2 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \\ 6,9 & 7,6 & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \end{Bmatrix}$$

该方法现在转移到具有相关成本 8, 10 的顶点 2。

### 步骤 3:

现在参见图 6, 顶点 2 具有去往顶点 1 和 3 的两个棱边。图 6 中表示了由与设置在  $\mathfrak{S}$  数据集合中的顶点 2 (8, 10) 相关的成本扩展的这些棱边的成本。到达顶点 1 的成本等于 (6, 12), 它差于  $r_1$  列表中的成本 (6, 9), 于是, 按照规则 1c), 该成本被丢弃。到达顶点 3 的成本等于 (1, 11), 它差于  $f_3$  列表中的成本 (9, 5)。于是, 按照规则 1b), 该成本被丢弃。最宽的带宽值对应于顶点 2: 7, 6。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \\ 6,9 & 7,6 & - \end{Bmatrix}$$

从  $\mathfrak{R}$  数据集合把该值转移为顶点 2 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \\ 6,9 & - & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \end{Bmatrix}$$

该方法转到具有相关成本 7, 6 的顶点 2。

### 步骤 4:

参见图 7, 顶点 2 具有去往顶点 1 和 3 的两个棱边。图 7 中表示了由与设置在  $\mathfrak{S}$  数据集合中的顶点 2 (7, 6) 相关的成本扩展的这些棱边的成本。到达顶点 1 的成本等于 (6, 8), 它既不等于又不差于  $r_1$  的任意项。于是, 按照规则 1d), 该成本被输入  $\mathfrak{R}$  数据集合。但是, 项目 (6, 9) 差于 (6, 8)。于是, 按照规则 1e) 丢弃项目 (6, 9)。到达顶点 3 的成本等于 (1, 7), 它差于  $f_3$  列表中的成本 (9, 5)。于是, 按照规则 1b), 该成本被丢弃。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \\ 6,8 & - & - \end{Bmatrix}$$

最宽的带宽值对应于顶点 1: 6, 8。从  $\mathfrak{R}$  数据集合把该值转移为顶点 1 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \\ - & - & - \\ - & - & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \end{Bmatrix}$$

该方法现在转到具有相关成本 6, 8 的顶点 1。

#### 步骤 5:

现在参见图 8, 顶点 1 具有去往顶点 2 和 3 的两个棱边。到达顶点 2 的成本等于 (2, 9), 它差于  $f_2$  列表中的成本 (7, 6)。于是, 按照规则 1b), 该成本被丢弃。到达顶点 3 的成本等于 (3, 11), 它差于  $f_3$  列表中的成本 (9, 5)。于是, 按照规则 1b), 该成本被丢弃。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ 4,1 & - & - \end{Bmatrix}$$

最宽的带宽值对应于顶点 1。从  $\mathfrak{R}$  数据集合把该值转移为顶点 1 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \\ 4,1 & - & - \end{Bmatrix}$$

该方法现在转到具有相关成本 4, 1 的顶点 1。

#### 步骤 6:

参见图 9, 顶点 1 具有去往顶点 2 和 3 的两个棱边。到达顶点 2 的成本等于 (2, 2), 它既不等于又不差于  $f_2$  的任意项。于是, 按照规则 1d), 该成本被输入  $\mathfrak{R}$  数据集合。到达顶点 3 的成本等于 (3, 4), 它既不等于又不差于  $f_3$  的任意项。于是, 按照规则 1d), 该成本被输

入  $\mathfrak{R}$  数据集合。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & 2,2 & 3,4 \end{Bmatrix}$$

最宽的带宽值对应于顶点 3: 3, 4。从  $\mathfrak{R}$  数据集合把该值转移为有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & 2,2 & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \\ 4,1 & - & - \\ - & - & 3,4 \end{Bmatrix}$$

该方法现在转到具有相关成本 3, 4 的顶点 3。

步骤 7:

现在参见图 10, 顶点 3 具有去往顶点 1 和 2 的两个棱边。到达顶点 1 的成本等于 (3, 8), 它差于  $f_1$  列表中的成本 (4, 1)。于是, 按照规则 1b), 该成本被丢弃。但是, 到达顶点 2 的成本等于 (3, 5), 它既不等于又不差于  $f_2$  或  $r_2$  的任意项。于是, 按照规则 1d), 该成本被输入  $\mathfrak{R}$  数据集合。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & 2,2 & - \\ - & 3,5 & - \end{Bmatrix}$$

最宽的带宽值对应于顶点 2: 3, 5。从  $\mathfrak{R}$  数据集合把该值转移为顶点 2 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & 2,2 & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \\ 4,1 & - & - \\ - & - & 3,4 \\ - & 3,5 & - \end{Bmatrix}$$

该方法现在转到具有相关成本 3, 5 的顶点 2。

步骤 8:

参见图 11，顶点 2 具有去往顶点 1 和 3 的两个棱边。到达顶点 1 的成本等于 (3, 7)，它差于  $f_1$  列表中的成本 (4, 1)。于是，按照规则 1b)，该成本被丢弃。到达顶点 3 的成本等于 (1, 6)，它差于  $f_3$  列表中的成本 (3, 4)。于是，按照规则 1b)，该成本被丢弃。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & 2,2 & - \end{Bmatrix}$$

最宽的带宽值对应于顶点 2: 2, 2。从  $\mathfrak{R}$  数据集合把该值转移为顶点 2 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \\ 4,1 & - & - \\ - & - & 3,4 \\ - & 3,5 & - \\ - & 2,2 & - \end{Bmatrix}$$

现在该方法转到具有相关成本 2, 2 的顶点 2。

### 步骤 9

现在参见图 12，顶点 2 具有去往顶点 1 和 3 的两个棱边。到达顶点 1 的成本等于 (2, 4)，它差于  $f_1$  列表中的成本 (4, 1)。于是，按照规则 1b)，该成本被丢弃。但是，到达顶点 3 的成本等于 (1, 3)，它既不等于又不差于  $f_3$  列表中的任意项。于是，按照规则 1d)，该成本被输入  $\mathfrak{R}$  数据集合。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & 1,3 \end{Bmatrix}$$

最宽的带宽值对应于顶点 3: 1, 3。从  $\mathfrak{R}$  数据集合把该值转移为顶点 3 的有效边界。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \\ - & - & - \end{Bmatrix} \quad \mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ - & - & 9,5 \\ - & 8,10 & - \\ - & 7,6 & - \\ 6,8 & - & - \\ 4,1 & - & - \\ - & - & 3,4 \\ - & 3,5 & - \\ - & 2,2 & - \\ - & - & 1,3 \end{Bmatrix}$$

该方法转到具有相关成本 1, 3 的顶点 3。

### 步骤 10

现在参见图 13, 顶点 3 具有去往顶点 1 和 2 的两个棱边。到达顶点 1 的成本等于 (1, 7), 它差于  $f_1$  列表中的成本 (4, 1)。于是, 按照规则 1b), 该成本被丢弃。到达顶点 2 的成本等于 (1, 4), 它差于  $f_2$  列表中的成本 (2, 2)。于是, 按照规则 1b), 该成本被丢弃。

$$\mathfrak{R} = \begin{Bmatrix} r_1 & r_2 & r_3 \\ - & - & - \end{Bmatrix}$$

由于在  $\mathfrak{R}$  数据集中没有留下其它项目, 因此按照规则 2) 结束该方法。

### 最终结果:

从顶点 0 到顶点 1、2、3 的有效边界如下:

$$\mathfrak{S} = \begin{Bmatrix} f_1 & f_2 & f_3 \\ 6,8 & 8,10 & 9,5 \\ 4,1 & 7,6 & 3,4 \\ - & 3,5 & 1,3 \\ - & 2,2 & - \end{Bmatrix}$$

图 14-16 分别图解说明了从顶点 0 到顶点 1、2 和 3 的有效边界。

现在参见图 17, 在本发明的一个优选实施例中, 提供一种包含计算机程序代码的计算机程序组件, 所述计算机程序代码当被加载到数据处理系统的处理器中时, 配置处理器执行得出数据通信网络中的一组节点的一组有效路径成本的方法, 所述一组节点包括一个源节点和若干目的地节点, 其中在网络中确定该组节点中节点间路径的路径成本, 每个路径成本包括诸如表示可用带宽的数值之类的限制成本和诸

如表示转接延迟的数值之类的附加成本。

该方法包括在步骤 10, 把从源节点到任意目的地节点的直接路径的路径成本记录在数据处理系统的存储器中的第一数据集合中。在第一数据集合中, 每个记录的成本与通过对应路径到达的节点相关。

在步骤 20, 从第一数据集合中选择最佳路径成本。该最佳路径成本被确定为包含最佳限制成本的路径成本, 或者如果若干路径成本包括相同的限制成本, 则为具有最佳限制成本和最佳附加成本的路径成本。例如, 如果限制成本是表示带宽的数值, 则最佳限制成本是表示最大带宽的成本。类似地, 如果附加成本是表示转接延迟的数值, 则最佳附加成本是表示最小转接延迟的成本。

在步骤 30, 把在步骤 20 中选择的最佳路径成本记录在数据处理系统的存储器中的第二数据集合中。在第二数据集合中, 记录的成本与通过对应的路径到达的节点相关。

在步骤 40, 从第一数据集合中除去在步骤 30 中记录在第二数据集合中的成本。这种除去可涉及删除、标记或者把该成本排除在考虑范围之外的任意其它操作。

在步骤 50, 从通过对应于步骤 30 中记录的成本的路径到达的目的地节点到该组节点中的任意其它节点的直接路径的累积路径成本被记录在第一数据集合中。在第一数据集合中, 每个记录的累积路径成本与通过对应路径到达的节点相关。

在步骤 60, 对于在步骤 50 中, 把其相关成本记录在第一数据集合中的每个节点, 比较第一和第二数据集合中与该节点相关的成本。从第一数据集合除去和到达该节点的任意其它成本相比, 到达该节点的具有较差限制成本和较差附加成本的任意成本, 或者和到达该节点的任意其它成本相比, 到达该节点的具有相同限制成本和较差附加成本的任意成本。

在步骤 70, 重复步骤 20-60, 直到在步骤 60 之后, 在第一数据集合中不留下任何成本。在步骤 80, 所得到的第二数据集合包括该组有效成本。换句话说, 该组有效成本包括源节点的有效边界。要认识到

数据处理系统可以是路由器、交换机、桥接器、桥式路由器、集线器、或者在通信网络中传送或接收信息的任意其它网络连网设备。

现在参见图 18，在本发明的另一实施例中，提供一种数据连网设备 100，所述数据连网设备 100 包括连接数据通信网络 170 的输入/输出 (I/O) 端口 110，与 I/O 端口 110 耦接的处理器 120，和与处理器耦接的存储器 130。存储器包括分配的存储第一数据集合 140 的空间，分配的存储第二数据集合 150 的空间和控制逻辑程序代码 160。操作上，数据连网设备得出数据通信网络中包括源节点和若干目的地节点的一组节点的一组有效路径成本。数据处理设备可包括数据通信网络的源节点。在网络中确定该组节点中节点之间路径的路径成本，并且每个路径成本包括限制成本和附加成本。操作上，控制逻辑 160 配置处理器，以便：

a) 把从源节点到任意目的地节点的直接路径的路径成本记录在第一数据集合 140 中，在第一数据集合中，每个记录的成本与通过对应路径到达的节点相关；

b) 从第一数据集合 140 中选择最佳路径成本，其中该最佳路径成本被确定为包含最佳限制成本的路径成本，或者如果若干路径成本包括相同的限制成本，则为具有最佳限制成本和最佳附加成本的路径成本；

c) 把在步骤 b) 中选择的最佳路径成本记录在第二数据集合 150 中，在第二数据集合 150 中，记录的成本与通过对应的路径到达的节点相关；

d) 从第一数据集合 140 中除去在步骤 c) 中记录在第二数据集合 150 中的成本；

e) 在第一数据集合 140 中，记录从通过对应于步骤 c) 中记录的成本的路径到达的目的地节点到该组节点中的任意其它节点的直接路径的累积路径成本，每个记录的累积路径成本与通过对应路径到达的节点相关；

f) 对于在步骤 e) 中，把其相关成本记录在第一数据集合中的每

个节点，比较第一数据集合 140 和第二数据集合 150 中与该节点相关的成本，从第一数据集合 140 中除去和到达该节点的任意其它成本相比，到达该节点的、具有较差限制成本和较差附加成本的任意成本，或者除去和到达该节点的任意其它成本相比，到达该节点的、具有相同限制成本和较差附加成本的任意成本；和。

g) 重复步骤 b) -f)，直到在步骤 f) 之后，在第一数据集合中不留下任何成本为止。

所得到的第二数据集合包括该组有效成本。虽然，在前面参考图 18 说明的本发明的实施例中，控制逻辑 160 是计算机程序代码，不过要认识到，在本发明的其它实施例中，至少可利用硬连线逻辑电路部分实现控制逻辑。另外要认识到，至少一个路径可包括若干链路和至少一个中间节点。

总之，在前面说明的本发明的优选实施例中，为数据通信网络中包括源节点和若干目的地节点的一组节点得出一组有效路径成本，其中在网络中确定该组节点中节点间路径的路径成本，每个路径成本包括限制成本和附加成本，包括首先迭代识别带宽较高的路径，并且当存在两个或多个带宽相同的路径时，选择具有与其相关的较低转接延迟的路径。

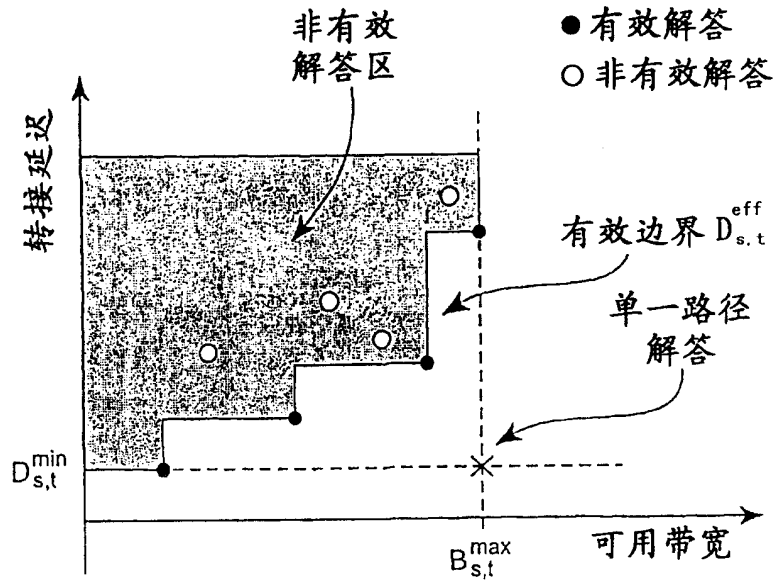


图1

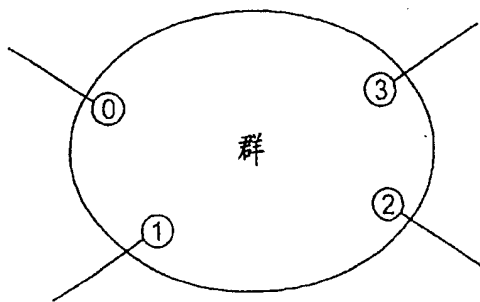


图2A

	0	1	2	3
0				
1				
2				
3				

转换矩阵

图2B

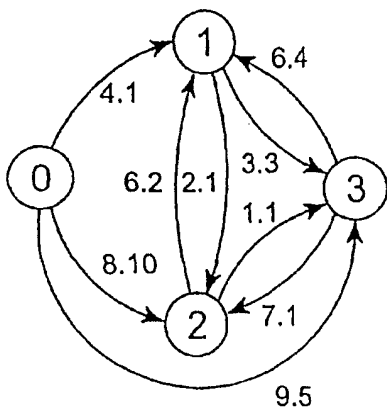


图 3

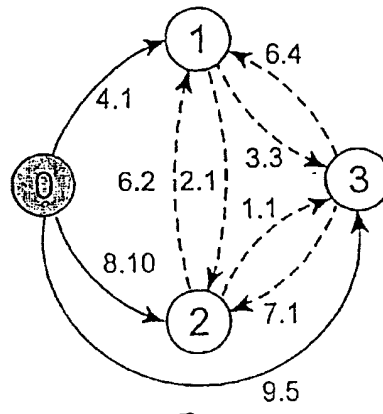


图 4

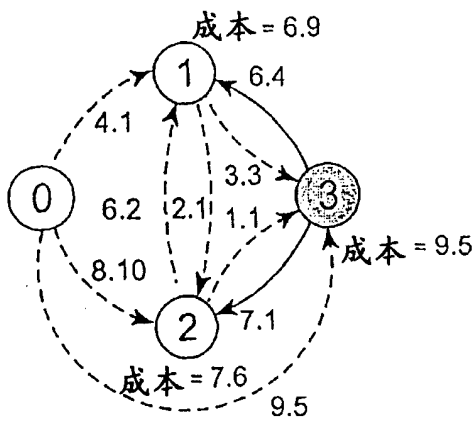


图 5

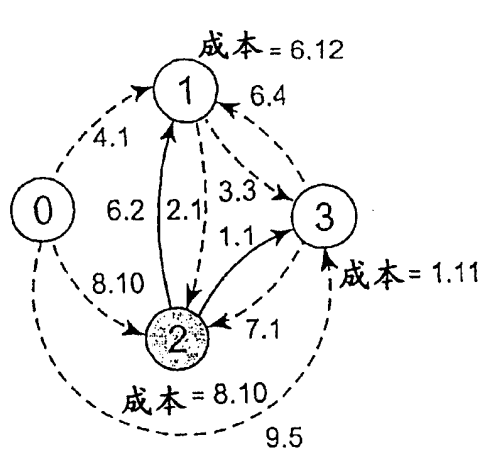


图 6

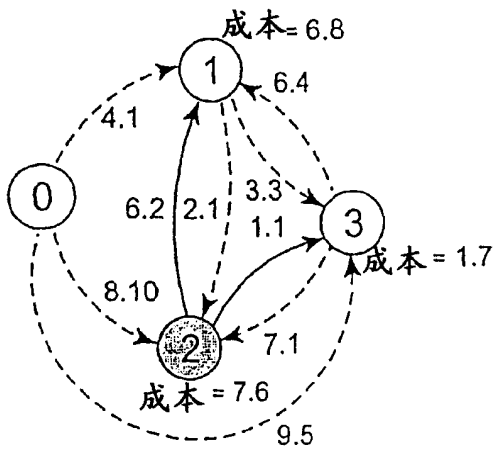


图7

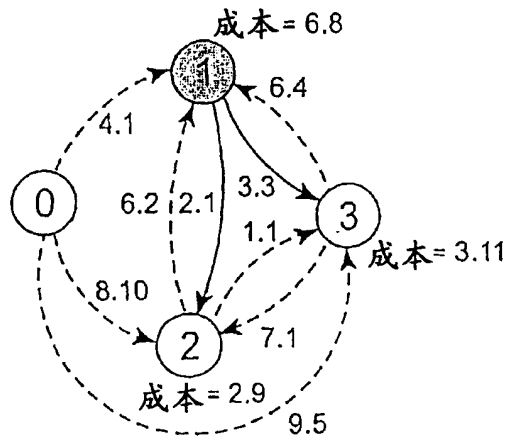


图8

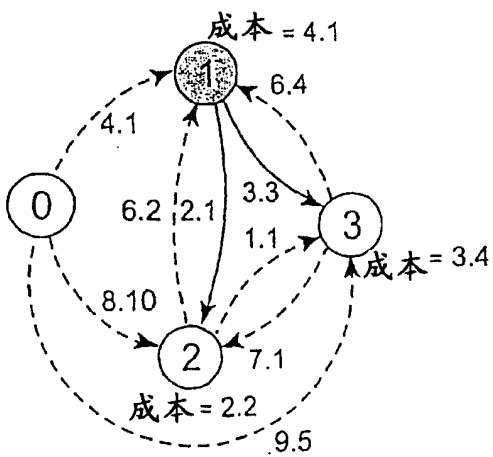


图9

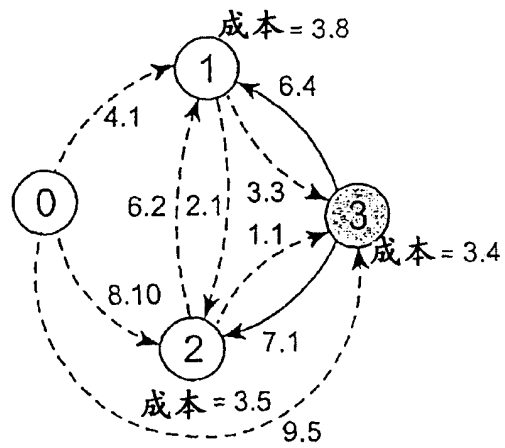


图10

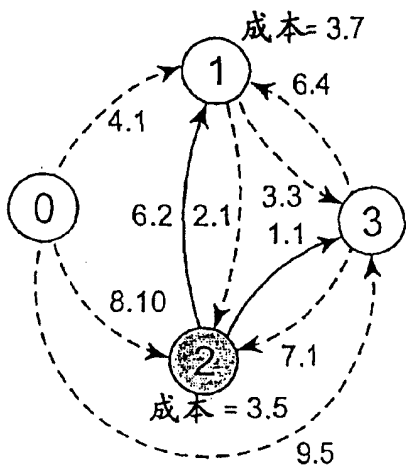


图 11

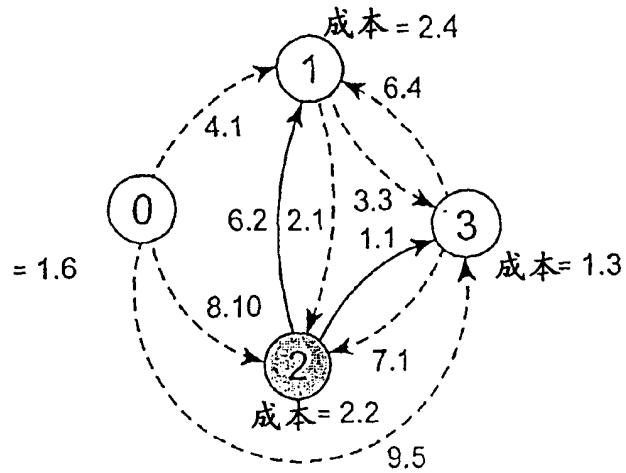


图 12

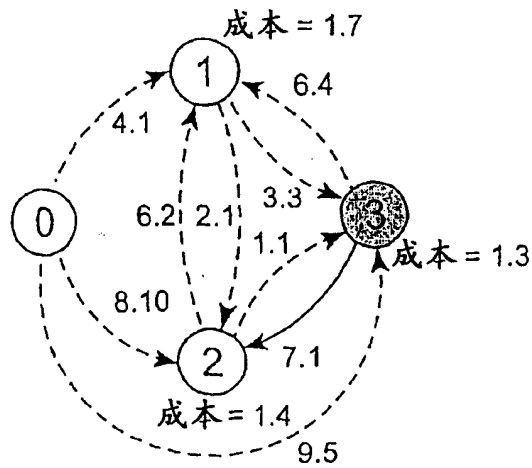


图 13

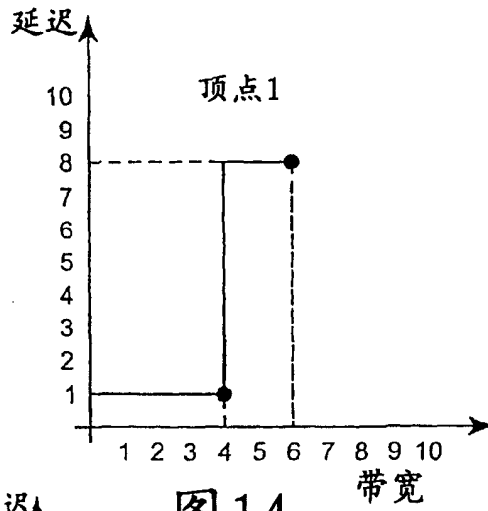


图 14

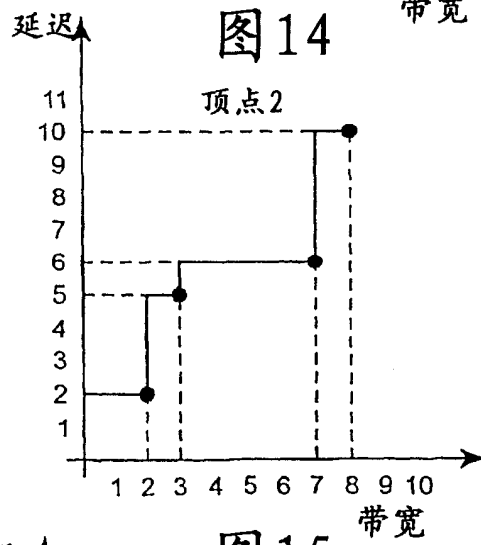


图 15

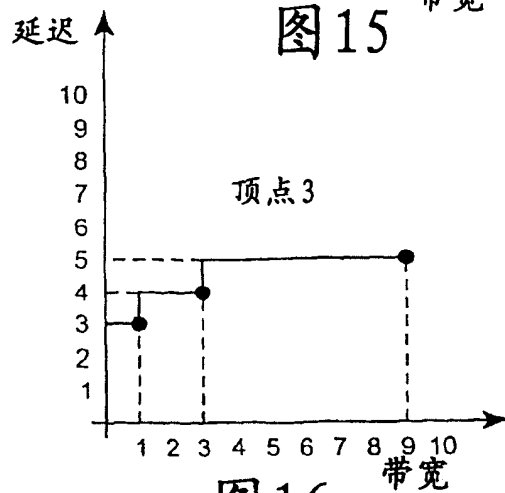


图 16

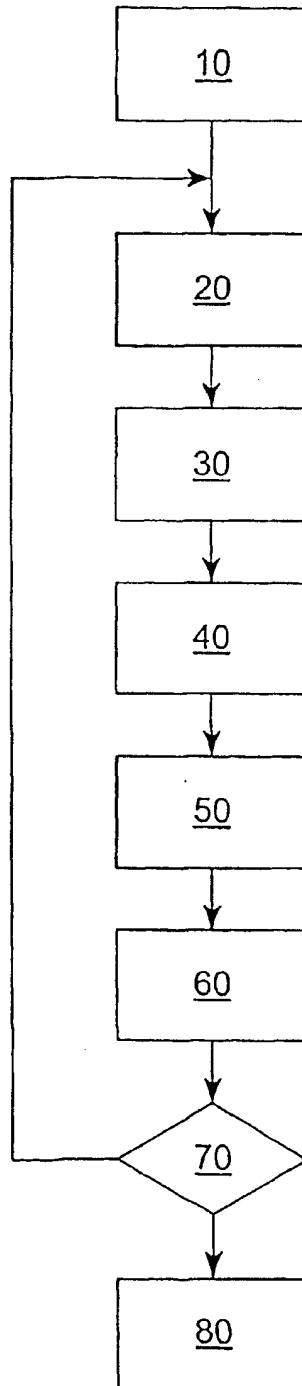


图 17

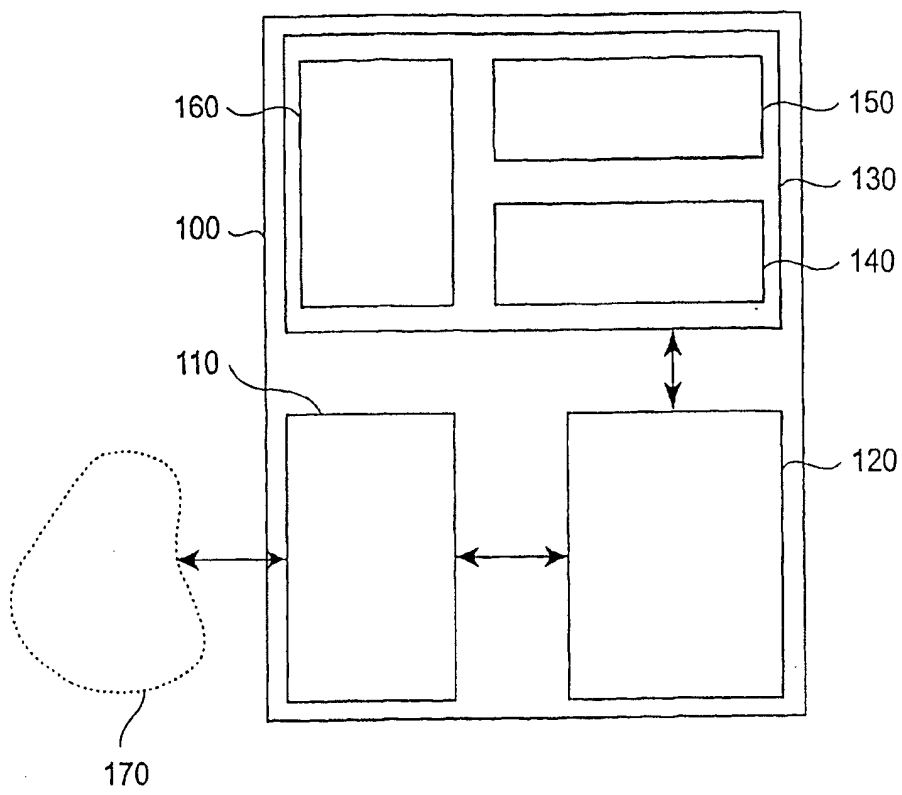


图18