

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织  
国际局



(43) 国际公布日  
2011年10月20日 (20.10.2011)

PCT

(10) 国际公布号  
WO 2011/127642 A1

- (51) 国际专利分类号:  
H04L 12/56 (2006.01)
- (21) 国际申请号: PCT/CN2010/071698
- (22) 国际申请日: 2010年4月12日 (12.04.2010)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (71) 申请人 (对除美国外的所有指定国): **华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。
- (72) 发明人; 及
- (75) 发明人/申请人 (仅对美国): **胡睿 (HU, Rui)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。**陈洪飞 (CHEN, Hongfei)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。**龚钧 (GONG, Jun)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。**翟素平 (ZHAI, Suping)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地

总部办公楼, Guangdong 518129 (CN)。**王小忠 (WANG, Xiaozhong)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。**赵鹤轩 (ZHAO, Hexuan)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为基地总部办公楼, Guangdong 518129 (CN)。

- (74) 代理人: **北京同立钧成知识产权代理有限公司 (LEADER PATENT & TRADEMARK FIRM)**; 中国北京市海淀区西直门北大街32号枫蓝国际A座8F-6, Beijing 100082 (CN)。
- (81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

[见续页]

(54) Title: ROUTING TABLE CONSTRUCTION METHOD AND DEVICE AND ROUTING TABLE LOOKUP METHOD AND DEVICE

(54) 发明名称: 路由表建立方法和装置及路由表查找方法和装置

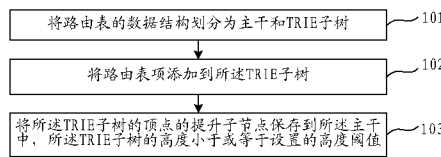


图 4 / Fig. 4

101 PARTITIONING THE DATA STRUCTURE OF A ROUTING TABLE INTO A TRUNK AND TRIE SUB-TREES  
 102 ADDING ENTRIES OF THE ROUTING TABLE TO THE TRIE SUB-TREES  
 103 STORING THE UPGRADE SUB-NODE OF THE VERTEX OF THE TRIE SUB-TREE IN THE TRUNK, THE HEIGHT OF THE TRIE SUB-TREE IS SMALLER THAN OR EQUAL TO A SET HEIGHT THRESHOLD VALUE

(57) Abstract: A routing table construction method and device and a routing table lookup method and device are disclosed. Wherein, the routing table construction method includes: the data structure of a routing table is partitioned into a trunk and TRIE sub-trees (101); entries of the routing table are added to the TRIE sub-trees (102); the upgrade sub-node of the vertex of the TRIE sub-tree is stored in the trunk, the height of the TRIE sub-tree is smaller than or equal to a set height threshold value (103), and the upgrade sub-node of the vertex of the TRIE sub-tree is the sub-vertex among prefix sub-nodes included in the vertex of the TRIE sub-tree. A high compression ratio of a routing table may be guaranteed by partitioning the data structure of the routing table into a trunk and TRIE sub-trees which are constructed with the TRIE algorithm; times for accessing a storage may be reduced by controlling the height of the TRIE sub-trees based on the height threshold value, so as to increase lookup speed and reduce lookup delay.

[见续页]



WO 2011/127642 A1



(84) **指定国** (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), 欧洲 (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT,

RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)。

**本国际公布:**

— 包括国际检索报告(条约第 21 条(3))。

---

(57) **摘要:**

公开了一种路由表建立方法和装置以及路由表查找方法和装置, 其中, 该路由表建立方法包括: 将路由表的数据结构划分为主干和 TRIE 子树 (101); 将路由表项添加到所述 TRIE 子树 (102); 将所述 TRIE 子树的顶点的提升子节点保存到所述主干中, 所述 TRIE 子树的高度小于或等于设置的高度阈值 (103), 所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。路由表的数据结构分为主干和 TRIE 子树, TRIE 子树采用 TRIE 算法建立, 可以保证路由表的高压缩率; 根据高度阈值控制 TRIE 子树的高度, 可以减少访问存储器的次数, 提高查找速率, 降低查找时延。

## 路由表建立方法和装置及路由表查找方法和装置

### 技术领域

本发明实施例涉及通信技术领域，特别涉及一种路由表建立方法和装置及路由表查找方法和装置。

### 背景技术

为了阻止地址空间耗尽，互联网工程任务组（The Internet Engineering Task Force；简称：IETF）于 1993 年提出了无类型域间路由（Classless Inter-Domain Routing；简称：CIDR）的地址结构的临时解决办法。CIDR 是现在实际路由查表的基础，脱离了传统的 A-E 分类地址结构。在 CIDR 中，一个 IP 网络由一个前缀（prefix）表示，该前缀由 IP 地址和标识该 IP 地址有效位的掩码复合表示。CIDR 允许路由器进行路由聚合，即大小相同、在地址空间中相互邻近且路由相同的 2 的整数次幂个子网的路由表条目可以合并成一条路径，并在路由表中只占用一个路由表项。最后的结果是：给定的地址块可按任意比特（bit）界限划分，网络上的计算机将按连续范围分配地址，但这个范围不需要相应的预先定义的类，所以称为无类型编址。CIDR 可以避免网络地址的过快枯竭和骨干路由器的路由表项的快速膨胀，同时使网络地址前缀的长度可以为任意值，进行路由查表时必须进行最长前缀匹配（the longest prefix match）搜索。衡量路由表的查表算法优劣的主要因素包括：路由表查询速度（由访问存储器的次数决定）、路由表所需要的存储器容量、插入删除路由表项的难易程度等。

无类型路由表的数据结构可以存储在一个层状数据结构。路由表的搜索是沿层次向下进行的，因此可以采用二叉树（Binary Tree）或者是根树（Radix Tree）及其变形的数据结构表示，表 1 为一种示例路由表。

表 1

前缀标号	前缀比特值	下一跳地址
P1	111*	H1
P2	10*	H2
P3	1010*	H3
P4	10101*	H4

可以将表1中的前缀存储到一个基本二叉树中，图1 为现有的基本二叉树存储的路由表的示意图，如图1所示，“0”代表向左分支，“1”代表向右分支。其中前缀比特值决定了从根向下的路径，前缀P1、P2、P3、P4分别采用阴影表示。该基本二叉树的节点中没有存储前缀P1-P4的值，P1-P4 的值为抵达对应节点的路径，因此该基本二叉树的路径由所存储的数据（例如：前缀比特值）确定，即该基本二叉树的路径表示了它所要存储的数据。在该基本二叉树中，处于第L层的节点代表了一类地址前L比特均相同的地址空间，且该前L个比特串是由从根节点到该节点的路径上的L比特组成。基本二叉树结构查找和存储效率都很低，比如：对于一组传统的B类地址上划分的子网地址，其前16比特相同，但遍历二叉树时，一次只提取“1”比特，速度很慢，若相应的地址前缀最长为M，则树的深度为M。如果一次检查K个比特，则树的深度可减少到M/K，这样树的内部节点包含的匹配项增加为2的K次幂。这样的树被称为2的K次幂分支树，树的最大层数为M/K。查表算法在每个节点处检查的比特数为K，也就被称为树的步长。

Patricia树是基本二叉树（Binary Tree）的变型，Patricia树允许每个节点指定一个值，测试连续跳过的比特数以加快搜索速度。前缀存放在Patricia树的叶节点上，假设有N个前缀则有N个叶节点，另有N-1个内部节点。冗余的支路链被压缩到了一个节点，因此查找算法不需要做连续的比特匹配而是可以一次匹配多个比特来快速到达叶节点。图2为现有Patricia 树存储

的路由表的示意图，对Patricia树能够较快的搜索到叶节点，Patricia树对较长的前缀的搜索效率比基本二叉树要高，如图2所示，如对于前缀P4，Patricia树经3次匹配可以完成查找，而基本二叉树需要5次。

IP地址前缀可以看成是一个范围，例如10.110.0.0/16实际上表示从10.110.0.0到10.110.255.255之间所有的IP地址，所以，该IP地址前缀可以用范围[10.110.0.0, 10.110.255.255]表示。将所有的IP地址前缀都转换成范围，则最长前缀匹配问题可以转化为找待查IP地址在哪个范围的问题，此问题类似于排序问题，可以用二叉树的方法建立路由表。假设IP地址为8比特，共有7个IP地址，IP地址的掩码和范围如下表2:

表2

序号	掩码表示	范围表示
A	00000000/1	[00000000, 10000000)
B	00000100/6	[00000100, 00001000)
C	00100000/4	[00100000, 00110000)
D	00100000/5	[00100000, 00101000)
E	00101100/6	[00101100, 00110000)
F	00110000/4	[00110000, 01000000)
G	01100000/3	[01100000, 10000000)

如图3所示，为现有采用二叉树存储IP地址掩码范围的示意图，左边是IP地址的区间表示，右边是将区间的端点放入二叉树中。每个节点存储整个路由表项，且一个IP地址前缀要分为2个端点，因此空间利用率低。

为了提高路由表的空间利用率低，可以采用TRIE算法建立并查找路由表，TRIE算法具有压缩率高、更新速度快的优点。但是由于TRIE算法访问节点的次数与、访问存储器的次数和表项长度成正比，随着TRIE算法的关键(key)值增长，路由表高度增加，IPv6转发信息库(Forwarding Information Base ;

简称: FIB) 查找访问存储器的次数较多, 造成吞吐率降低, 性能下降, 时延增长, 容量降低。

## 发明内容

本发明实施例提供一种路由表建立方法和装置及路由表查找方法和装置, 用以解决现有 TRIE 算法中树的高度过高, 引起的查找路由表的速率慢, 时延增长的问题, 提高对路由表的查找速率, 降低查找时延。

本发明实施例提供了一种路由表建立方法, 包括:

将路由表的数据结构划分为主干和 TRIE 子树;

将路由表项添加到所述 TRIE 子树;

将所述 TRIE 子树的顶点的提升子节点保存到所述主干中, 所述 TRIE 子树的高度小于或等于设置的高度阈值, 所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。

本发明实施例还提供了一种路由表查找方法, 其中, 路由表的数据结构包括主干和 TRIE 子树, 所述子树的顶点的提升子节点保存在所述路由表的主干中, 所述方法包括:

在所述路由表的主干中, 查找与关键字最长匹配的提升子节点;

获取所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置;

根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置, 在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

本发明实施例还提供了一种路由表建立装置, 包括:

划分模块, 用于将路由表的数据结构划分为主干和 TRIE 子树;

表项添加模块, 用于将路由表项添加到所述 TRIE 子树;

保存模块, 用于将所述 TRIE 子树的顶点的提升子节点保存到所述主干

中，所述 TRIE 子树的高度小于或等于设置的高度阈值，所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。

本发明实施例还提供了一种路由表查找装置，包括：

主干查找模块，用于在路由表的主干中，查找与关键字最长匹配的提升子节点，所述路由表的数据结构包括主干和 TRIE 子树，所述子树的顶点的提升子节点保存在所述路由表的主干中；

子树位置获取模块，用于获取所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置；

子树查找模块，用于根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

本发明实施例提供了一种路由表建立方法和装置及路由表查找方法和装置，由于路由表的数据结构分为主干和 TRIE 子树，子树为采用 TRIE 算法建立的 TRIE 子树，可以保证路由表的高压缩率；在添加路由表项时，控制 TRIE 子树的高度，可以减少的访问存储器的次数，提高了对查找路由表的速率，降低了查找时延。

## 附图说明

为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作一简单介绍，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

图 1 为现有的基本二叉树存储的路由表的示意图；

图 2 为现有 Patricia 树存储的路由表的示意图；

图 3 为现有采用二叉树存储 IP 地址掩码范围的示意图；

- 图 4 为本发明路由表建立方法实施例的流程图；
- 图 5a 为本发明路由表建立方法实施例中主干和 TRIE 子树的结构示意图；
- 图 5b 为本发明路由表建立方法实施例中多位 TRIE 算法的示意图；
- 图 5c 为本发明路由表建立方法实施例中树位图算法的内位图的示意图；
- 图 5d 为本发明路由表建立方法实施例中树位图算法的外位图的示意图；
- 图 6a 为本发明路由表建立方法实施例添加路由表项分裂前的示意图；
- 图 6b 为本发明路由表建立方法实施例添加路由表项分裂后的示意图；
- 图 6c 为本发明路由表建立方法实施例添加路由表项的流程示意图；
- 图 6d 为本发明路由表建立方法实施例中添加 IP 地址前缀之前的示意图；
- 图 6e 为本发明路由表建立方法实施例中添加 IP 地址前缀之后的示意图；
- 图 7 为本发明路由表查找方法实施例的流程图；
- 图 8 为本发明路由表建立装置第一实施例的结构示意图；
- 图 9 为本发明路由表建立装置第二实施例的结构示意图；
- 图 10 为本发明路由表查找装置第一实施例的结构示意图；
- 图 11 为本发明路由表查找装置第二实施例的结构示意图。

## 具体实施方式

下面通过附图和实施例，对本发明的技术方案做进一步的详细描述。

图 4 为本发明路由表建立方法实施例的流程图，如图 4 所示，该路由表建立方法包括：

101、将路由表的数据结构划分为主干和 TRIE 子树；

102、将路由表项添加到所述 TRIE 子树；

103、将所述 TRIE 子树的顶点的提升子节点保存到所述主干中，所述 TRIE 子树的高度小于或等于设置的高度阈值，所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。



路由表可以存储在层状的数据结构中，可以预先将路由表的数据结构划分为主干和 TRIE 子树，将所述 TRIE 子树的顶点的提升子节点保存到所述主干中，其中，TRIE 子树可以采用 TRIE 算法建立，其中 TRIE 算法建立的树，又称单词查找树，是一种树形结构，是一种哈希树的变种。主干的建立方法可以采用但不限于范围匹配、B 树或三态内容寻址存储器等其他算法。图 5a 为本发明路由表建立方法实施例中主干和 TRIE 子树的结构示意图，如图 5a 所示，采用 TRIE 算法建立的 TRIE 子树 41 的每个节点 42 中可以包括多个前缀子节点，其中 TRIE 子树 41 最上层的节点 42 为顶点，在 TRIE 子树 41 顶点中最上层的前缀子节点为提升子节点 44，TRIE 子树 41 的提升子节点 44 保存在主干 45 中，即提升子节点 44 同时属于 TRIE 子树 41 和主干 45，主干相当于是一个保存提升子节点的数据库。

下面介绍两种 TRIE 算法：

图 5b 为本发明路由表建立方法实施例中多位 (multi-bit) TRIE 算法的示意图，多位 TRIE 算法用 IP 地址的多个位 (bit) 的索引形成数组，将由 IP 地址的不同位形成的数组按其在 IP 地址中的先后顺序关联，查找时一次查找多个位，分级查找。如图 5b 中，前 24 位为第一级，后 8 位为第二级，如果前 24 位在第一存储区域 11 中匹配成功，则可以直接将得到匹配结果发送到转发信息库 13；如果前 24 位在第一存储区域 11 中部分匹配，则将前 24 位和后 8 位结合在第二存储区域 12 中匹配，匹配成功时，可以将得到匹配结果发送到转发信息库 13。

图 5c 为本发明路由表建立方法实施例中树位图 (Tree bitmap) 算法的内位图的示意图，图 5d 为本发明路由表建立方法实施例中树位图算法的外位图的示意图。树位图算法也属于 multi-bit TRIE 算法，树位图算法用位图 (bitmap) 表示节点中有哪些前缀，以及节点有哪些孩子。内位图 (Internal bitmap) 表示节点中有哪些前缀，如图 5c 中的一个 TRIE 子树的节点中可存

放 7 个前缀，但实际上仅在前缀子节点 P1、P2、P3 的位置存放了三个前缀，将存放前缀的内位图的位 (bit) 置为“1”，将未存放前缀的位置为“0”，可以得到“1011000”。外位图 (External bitmap) 可以表示 TRIE 子树的一个节点有哪些孩子，如图 5d 所示，该节点最多可以有 8 个孩子节点，但实际上只有三个孩子节点，则在有孩子节点的位置上置“1”，没有孩子节点的位置上置“0”。

建立该路由表的数据结构时，可以先采用 TRIE 算法建立 TRIE 子树，然后在添加路由表项或删除路由表项的具体过程中，根据设置的高度阈值对保存在所述路由表的主干中的提升子节点进行相应的更新，具体分为以下场景：

#### 场景一、添加路由表项

其中，若所述 TRIE 子树添加路由表项后的高度大于所述高度阈值，则将所述 TRIE 子树中高度大于所述高度阈值的分支从所述 TRIE 子树分裂，得到新 TRIE 子树；

然后，将所述新 TRIE 子树的顶点的提升子节点保存到所述主干中，所述新 TRIE 子树的顶点为所述 TRIE 子树中高度大于所述高度阈值的分支、与所述 TRIE 子树的顶点连接的节点。

图 6a 为本发明路由表建立方法实施例添加路由表项分裂前的示意图，图 6b 为本发明路由表建立方法实施例添加路由表项分裂后的示意图。假设预设的高度阈值为“3”，在向路由表中添加路由表项例如：IP 地址前缀时，如果路由表的所有 TRIE 子树的高度没有超出“3”，则直接将路由表项添加到路由表中对应的位置。如果添加路由表项后，某个 TRIE 子树的高度大于“3”，如图 6a 中，添加路由表项后，高度为“4”，而 TRIE 子树 T 的顶点 N1 的两个分支中，N1→N2→N4→N8 的分支的高度大于高度阈值。此时，如图 6b 所示，为了保证 TRIE 子树 T 的高度可以将该 TRIE 子树 T 从 N2 处分裂，将分支高度大于高度阈值的分支作为新 TRIE 子树 T2，N2 为新 TRIE 子树 T2 的顶点，

N2 最上层的前缀子节点可以作为新 TRIE 子树 T2 的提升子节点保存到主干中，分裂后 T 剩余的部分为 TRIE 子树 T1。其中，将 TRIE 子树分裂得到新 TRIE 子树后，新 TRIE 子树的顶点与 TRIE 子树顶点之间的对应关系不变，新 TRIE 子树与 TRIE 子树的提升子节点之间的关系在主干中有体现。

进一步地，在添加路由表项的过程中，若所述 TRIE 子树分裂后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，可以将所述 TRIE 子树的顶点合并到所述上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

根据上图 6b，假设添加路由表项后，需要再从 N3 对 TRIE 子树 T1 进行分裂，分裂后的 T1 只剩下节点 N1，此时分裂后的 T1 的高度为“1”，小于高度阈值，如果此时分裂后的 T1 具有上一级的 TRIE 子树，则可以将分裂后的 T1 合并到上一级 TRIE 子树中。图 6c 为本发明路由表建立方法实施例添加路由表项的流程示意图，如图 6c 所示，结合上述的图 6a 和图 6b，该添加路由表项的流程具体可以包括：

201、TRIE 子树 T 中的某个节点 N1（顶点）发现该 TRIE 子树的高度大于高度阈值 h。

202、从节点 N1 向 TRIE 子树 T 的祖先方向查找，找到高度为 h 的节点 N2。

203、以节点 N2 为顶点，从 TRIE 子树 T 分裂出新 TRIE 子树 T2，T 剩余的部分为 T1。

204、判断 T1 的高度是否小于预设的高度阈值 h，如果 T1 有上一级 TRIE 子树（也可以称为父子树、父亲等）且 T1 的高度小于 h，执行 205，否则结束。

205、将 TRIE 子树 T1 合并入其上一级 TRIE 子树，将合并后的 TRIE 子树作为 TRIE 子树 T，并返回 201。

为了详细说明路由表项添加的具体过程，以向路由表中添加的路由表项为 IP 地址前缀为例，假设 TRIE 子树的高度阈值为“2”，每个 TRIE 子树的节点中的前缀子节点为 3 级，每个节点共 7 个前缀子节点，用 14 位标识 IP 地址前缀，主干存储在三态内容寻址存储器（Ternary Content Addressable Memory；简称：TCAM）中。如图 6d 所示，为本发明路由表建立方法实施例中添加 IP 地址前缀之前的示意图，其中，在 TRIE 子树 A 的一些前缀子节点中保存的 IP 地址前缀，其中 P1 保存的 IP 地址前缀为“1011101\*\*\*\*\*”，P2 保存的 IP 地址前缀为“10111001\*\*\*\*\*”，P3 保存的 IP 地址前缀为“10111010\*\*\*\*\*”，P4 保存的前缀为“1011100000\*\*\*\*\*”，P5 保存的 IP 地址前缀为“1011100001\*\*\*\*”，P6 保存的 IP 地址前缀为“10111000001\*\*\*”，P7 保存的 IP 地址前缀为“10111000010\*\*\*”，P8 保存的 IP 地址前缀为“10111011100\*\*\*”，P9 保存的 IP 地址前缀为“10111011101\*\*\*”，P10 保存的 IP 地址前缀为“10111011110\*\*\*”。TRIE 子树 A 的顶点 N1 中的提升子节点虽然为保存 IP 地址前缀，但根据路由表的路径，N1 的提升子节点 T1 对应的 IP 地址前缀应该为“101110\*\*\*\*\*”，因此，主干 TCAM 中保存 TRIE 子树 A 提升子节点 T1 的 IP 地址前缀“101110\*\*\*\*\*”。同理，主干 TCAM 中还可能保存其他的 TRIE 子树提升子节点的 IP 地址前缀，如：“001\*\*\*\*\*”。假设需要添加的 IP 地址前缀为“1011100001010\*”，需要在 TRIE 子树 A 中添加一个前缀子节点 P11，但添加 P11 后，由于 TRIE 子树 A 增加了节点 N4 后的高度为“3”，大于设置的高度阈值“2”，因此需要进行分裂。

如图 6e 所示，为本发明路由表建立方法实施例中添加 IP 地址前缀之后的示意图，由于 TRIE 子树 A 中将 N2 和 N4 所在的分支高度大于高度阈值，因此将该分支从 TRIE 子树 A 中分裂为新 TRIE 子树 A2，剩余的部分为 A1（包括节点 N1 和 N3），此时 A2 的顶点 N2 的提升子节点 T2 对应的 IP 地址前缀应

该为“101110000\*\*\*\*\*”；A1 提升子节点 T1 的 IP 地址前缀与 A 相同，为“101110\*\*\*\*\*”。将 A2 的提升子节点 T2 的 IP 地址前缀“101110000\*\*\*\*\*”保存到主干 TCAM 中，T2 与 T1 的对应关系在主干中有体现，完成分裂过程。

#### 场景二、删除路由表项；

其中，若所述 TRIE 子树删除路由表项后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，可以将所述 TRIE 子树的顶点合并到上一级 TRIE 子树中，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

其中，将所述 TRIE 子树的顶点合并到上一级 TRIE 子树中的过程与向上一级 TRIE 子树添加路由表项的过程相似，也可以判断上一级 TRIE 子树的高度是否大于高度阈值，如果大于高度阈值，则可以对上一级 TRIE 子树进行分裂处理，还有可能对再上一级 TRIE 子树进行合并或分裂处理，经过层层合并和分裂，保证每一个没有上一级 TRIE 子树的 TRIE 子树的高度为设定的高度阈值。

本实施例路由表的数据结构分为主干和 TRIE 子树，子树为采用 TRIE 算法建立的 TRIE 子树，因此可以保证路由表的高压缩率；根据设置的高度阈值，可以控制 TRIE 子树的高度，因此减少了的访问存储器的次数，提高了查找速率，降低了查找时延。

图 7 为本发明路由表找方法实施例的流程图，如图 7 所示，该路由表找方法包括：

301、在路由表的主干中，查找与关键字最长匹配的提升子节点；

其中，路由表可以存储在层状的数据结构中，路由表的数据结构包括主干和 TRIE 子树，TRIE 子树的顶点的提升子节点保存到在所述主干中，其中，TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。TRIE 子树可以采用 TRIE 算法建立，主干的建立方法可以采用但不限于范围匹配、B 树或三态内容寻址存储器等其他算法。路由表具体结构

可以参照上述实施例中的相关描述和图 5a。

其中，在路由表的主干中查找与关键字最长匹配的提升子节点的具体过程可以包括：从保存在所述路由表的主干中的各个 TRIE 子树的提升子节点中，查找与所述关键字匹配的各个第一前缀；从所述各个第一前缀中，选取掩码长度最长的第一前缀作为所述与关键字最长匹配的提升子节点。其中，查找与所述关键字对应的各个第一前缀时，可以采用关键字中的部分数据与主干中的各个提升子节点进行匹配，匹配方法可以根据主干的建立方法选择。

302、获取所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置；

与关键字最长匹配的提升子节点的地址信息可以指向该提升子节点所在的 TRIE 子树的顶点，根据该地址信息可以获取该提升子节点对应的 TRIE 子树的顶点的位置。

303、根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

303 具体可以包括：

根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述与关键字最长匹配的提升子节点对应的所述 TRIE 子树各个节点的前缀子节点中，查找与所述关键字匹配的各个第二前缀；

从所述各个第二前缀中，选取掩码长度最长的第二前缀作为所述查找结果。

其中，查找与所述关键字对应的各个第二前缀时，可以采用关键字中的部分数据与 TRIE 子树各个节点的前缀子节点进行匹配，匹配方法可以采用建立 TRIE 子树时所使用的 TRIE 算法。

以在上述实施例中图 6e 的路由表中查找为例，具体说明查找的过程。假设查找的关键字为“10111000011000”，现在主干 TCAM 中查找与关键字匹配

的提升子节点，该与关键字匹配的提升子节点应该为一个或多个保存在 TCAM 中的提升子节点对应的 IP 地址前缀。以关键字“10111000011000”在图 6e 所示的路由表中查找后，命中的提升子节点为两个 IP 地址前缀：

“101110\*\*\*\*\*”和“101110000\*\*\*\*”，分别对应指向 TRIE 子树 A1 和 A2。由于 TRIE 子树 A2 对应的 IP 地址前缀“101110000\*\*\*\*”的掩码长度比 TRIE 子树 A1 对应的 IP 地址前缀“101110\*\*\*\*\*”长，因此，可以选则 TRIE 子树 A2 采用 TRIE 算法进行进一步的查找。在 TRIE 子树 A2 中，以关键字中未匹配的数据“11000”继续进行匹配，最后查找到 TRIE 子树 A2 的节点 N2 中的 P5 的 IP 地址前缀“1011100001\*\*\*\*”。因此可以得出该关键字命中 P5 的 IP 地址前缀。如果命中的 IP 地址前缀不只是一个，可以选取掩码长度最长的 IP 地址前缀作为最终的查找结果。

本实施例路由表的数据结构分为主干和采用 TRIE 算法建立的 TRIE 子树，可以保证路由表的高压缩率；根据设置的高度阈值，可以控制 TRIE 子树的高度，因此在查找路由表时，可以减少访问存储器的次数，提高查找速率，降低查找时延。

图 8 为本发明路由表建立装置第一实施例的结构示意图，如图 8 所示，该路由表建立装置包括：划分模块 51、表项添加模块 52 和保存模块 53。

其中，划分模块 51，用于将路由表的数据结构划分为主干和 TRIE 子树；表项添加模块 52，用于将路由表项添加到所述 TRIE 子树；

保存模块 53，用于将所述 TRIE 子树的顶点的提升子节点保存到所述主干中，所述 TRIE 子树的高度小于或等于设置的高度阈值，所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。

具体地，在路由表存储的数据结构中，划分模块 51 将路由表的数据结构划分为主干和 TRIE 子树，采用 TRIE 算法建立的 TRIE 子树，采用其他算法例如：范围匹配、B 树或三态内容寻址存储器等建立主干。其中表项添加模块

52 将路由表项添加到所述 TRIE 子树后，保存模块 53 将 TRIE 子树的顶点的提升子节点保存在主干中，且预先设置了高度阈值，TRIE 子树的高度小于或等于设置的高度阈值。如果采用 TRIE 算法建立的 TRIE 子树，在添加路由表项例如：IP 地址前缀后的高度大于高度阈值，则可以将添加路由表项后的 TRIE 子树分裂，将路由表的数据结构中的 TRIE 子树的高度控制为高度阈值。路由表的数据结构建立之后，可能需要删除其中的路由表项，删除一定的路由表项之后，如果删除路由表项后的 TRIE 子树的高度小于或等于高度阈值，将删除路由表项后的 TRIE 子树与上一级 TRIE 子树合并。本实施例中建立路由表的具体过程，可以参见本发明路由表建立方法实施例中的相关描述。

本实施例由于路由表的数据结构分为主干和 TRIE 子树，TRIE 子树采用 TRIE 算法建立，因此可以保证路由表的高压缩率；根据设定的高度阈值，更新模块可以控制 TRIE 子树的高度，因此减少了访问存储器的次数，提高了查找速率，降低了查找时延。

图 9 为本发明路由表建立装置第二实施例的结构示意图，如图 9 所示，在本发明路由表建立装置第二实施例的基础上，路由表建立装置还可以包括：分裂模块 54，用于若所述 TRIE 子树添加路由表项后的高度大于所述高度阈值，则将所述 TRIE 子树中高度大于所述高度阈值的分支从所述 TRIE 子树分裂，得到新 TRIE 子树；

此外，保存模块 53，还用于将所述新 TRIE 子树的顶点的提升子节点保存到所述主干中，所述新 TRIE 子树的顶点为所述 TRIE 子树中高度大于所述高度阈值的分支、与所述 TRIE 子树的顶点连接的节点。

或者，进一步地，路由表建立装置还可以包括：合并删除模块 56，用于若所述 TRIE 子树分裂后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到所述上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。



再进一步地，合并删除模块 56 还可以用于，若所述 TRIE 子树删除路由表项后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

具体地，在路由表存储的数据结构中，划分模块 51 可以预先将路由表的数据结构划分为主干和 TRIE 子树，TRIE 子树的顶点的提升子节点保存在主干中；采用 TRIE 算法建立 TRIE 子树，具体的建立 TRIE 子树的方法可以参照本发明路由表建立方法实施例的相关示例的描述，采用范围匹配、B 树或 TCAM 等方法建立 TRIE 子树所在的主干。表项添加模块 52 将路由表项添加到所述 TRIE 子树后，保存模块 53 将 TRIE 子树的顶点的提升子节点保存在主干中，且预先设置了高度阈值，TRIE 子树的高度小于或等于设置的高度阈值。在添加路由表项时，如果添加路由表项后的 TRIE 子树的高度大于设置的高度阈值，分裂模块 54 将所述 TRIE 子树中高度大于所述高度阈值的分支从所述 TRIE 子树分裂，得到新 TRIE 子树。如果分裂后的 TRIE 子树的高度小于高度阈值，且 TRIE 子树具有上一级 TRIE 子树，合并删除模块 56 将所述 TRIE 子树的顶点合并到所述上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

路由表的数据结构建立之后，可能需要删除其中的路由表项例如：IP 地址前缀，删除一定的路由表项之后，如果删除路由表项后的 TRIE 子树的高度小于或等于高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，合并删除模块 56 还可以用于：将所述 TRIE 子树的顶点合并到上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

本实施例路由表的数据结构分为主干和 TRIE 子树，TRIE 子树采用 TRIE 算法建立，因此可以保证路由表的高压缩率；在添加路由表项时，分裂单元和保存单元可以将大于高度阈值的 TRIE 子树分裂；在删除路由表项时，删除单

元和添加单元可以将高度小于高度阈值的 TRIE 子树合并，从而控制 TRIE 子树的高度，减少了的访问存储器的次数，提高了查找速率，降低了查找时延。

图 10 为本发明路由表查找装置第一实施例的结构示意图，如图 10 所示，该路由表查找装置包括：

主干查找模块 71，用于在路由表的的主干中，查找与关键字最长匹配的的提升子节点，所述路由表的数据结构包括主干和 TRIE 子树，所述 TRIE 子树的顶点的提升子节点保存在所述路由表的主干中；

子树位置获取模块 72，用于获取所述与关键字最长匹配的的提升子节点对应的 TRIE 子树的顶点的位置；

子树查找模块 73，用于根据所述与关键字最长匹配的的提升子节点对应的 TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

具体地，路由表可以存储在层状的数据结构中，路由表的数据结构包括主干和 TRIE 子树，TRIE 子树的顶点的提升子节点保存到在所述主干中，其中，TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。TRIE 子树可以采用 TRIE 算法建立，主干的建立方法可以采用但不限于范围匹配、B 树或三态内容寻址存储器等其他算法。主干查找模块 71 在所述路由表的数据结构的主干中，可以查找与关键字最长匹配的的提升子节点；子树位置获取模块 72 则可以根据与关键字最长匹配的的提升子节点中的地址信息，获取对应的 TRIE 子树的顶点位置；子树查找模块 73 根据与关键字最长匹配的的提升子节点对应的 TRIE 子树的顶点的位置，可以在对应的 TRIE 子树中采用 TRIE 算法查找到该关键字对应的查找结果。

本实施例路由表的数据结构分为主干和采用 TRIE 算法建立的 TRIE 子树，可以保证路由表的高压缩率；根据设置的高度阈值，可以控制 TRIE 子树的高度，因此在查找路由表时，通过主干查找模块查找到的与关键字最长匹配的

提升子节点，子树位置获取模块可以获取关键字对应的 TRIE 子树位置，然后子树查找模块在对应的 TRIE 子树中查找关键字对应的查找结果，可以减少访问存储器的次数，提高查找速率，降低查找时延。

图 11 为本发明路由表查找装置第二实施例的结构示意图，在本发明路由表查找装置第一实施例的基础上，如图 11 所示，该路由表查找装置的主干查找模块 71 包括：

第一前缀子模块 711，用于从保存在所述路由表的主干中的各个 TRIE 子树的提升子节点中，查找与所述关键字匹配的各个第一前缀；

最长匹配子模块 712，用于从所述各个第一前缀中，选取掩码长度最长的第一前缀作为所述与关键字最长匹配的提升子节点。

进一步地，子树查找模块 73 包括：

第二前缀子模块 731，用于根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述与关键字最长匹配的提升子节点对应的所述 TRIE 子树各个节点的前缀子节点中，查找与所述关键字匹配的各个第二前缀；

查找结果子模块 732，用于从所述各个第二前缀中，选取掩码长度最长的第二前缀作为所述查找结果。

具体地，第一前缀子模块 711 从保存在所述路由表的主干中的各个 TRIE 子树的提升子节点中，查找与所述关键字对应的各个第一前缀后，最长匹配子模块 712，从各个第一前缀中选取掩码长度最长的第一前缀作为与关键字最长匹配的提升子节点。子树位置获取模块 72 根据与关键字最长匹配的提升子节点获取对应的 TRIE 子树的顶点的位置；然后，第二前缀子模块 731 根据与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在与关键字最长匹配的提升子节点对应的所述 TRIE 子树各个节点的前缀子节点中，查找与所述关键字对应的各个第二前缀；查找结果子模块 732 从所述各个第二前

缀中选取掩码长度最长的第二前缀作为所述查找结果。

本实施例路由表的数据结构分为主干和采用 TRIE 算法建立的 TRIE 子树，可以保证路由表的高压缩率；根据设置的高度阈值，可以控制 TRIE 子树的高度，因此在查找路由表时，在主干查找模块的各个子模块查找到的与关键字最长匹配的提升子节点后，子树位置获取模块可以获取关键字对应的 TRIE 子树位置，然后子树查找模块的各个子模块在对应的 TRIE 子树中查找关键字对应的查找结果，可以减少访问存储器的次数，提高查找速率，降低查找时延。

本领域普通技术人员可以理解：实现上述方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成，前述的程序可以存储于计算机可读取存储介质中，该程序在执行时，执行包括上述方法实施例的步骤；而前述的存储介质包括：ROM、RAM、磁碟或者光盘等各种可以存储程序代码的介质。

最后应说明的是：以上实施例仅用以说明本发明的技术方案，而非对其限制；尽管参照前述实施例对本发明进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本发明各实施例技术方案的范围。

## 权利要求

1、一种路由表建立方法，其特征在于，包括：

将路由表的数据结构划分为主干和 TRIE 子树；

将路由表项添加到所述 TRIE 子树；

将所述 TRIE 子树的顶点的提升子节点保存到所述主干中，所述 TRIE 子树的高度小于或等于设置的高度阈值，所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。

2、根据权利要求 1 所述的路由表建立方法，其特征在于，还包括：

若所述 TRIE 子树添加路由表项后的高度大于所述高度阈值，则将所述 TRIE 子树中高度大于所述高度阈值的分支从所述 TRIE 子树分裂，得到新 TRIE 子树；

将所述新 TRIE 子树的顶点的提升子节点保存到所述主干中，所述新 TRIE 子树的顶点为所述 TRIE 子树中高度大于所述高度阈值的分支、与所述 TRIE 子树的顶点连接的节点。

3、根据权利要求 2 所述的路由表建立方法，其特征在于，还包括：

若所述 TRIE 子树分裂后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到所述上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

4、根据权利要求 1 所述的路由表建立方法，其特征在于，还包括：

若所述 TRIE 子树删除路由表项后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

5、根据权利要求 1-4 任一所述的路由表建立方法，其特征在于，还包括：

采用 TRIE 算法建立所述 TRIE 子树，采用范围匹配、B 树或三态内容寻址存储器建立所述主干。

6、一种路由表查找方法，其特征在于，路由表的数据结构包括主干和 TRIE 子树，所述 TRIE 子树的顶点的提升子节点保存在所述路由表的主干中，所述方法包括：

在所述路由表的主干中，查找与关键字最长匹配的提升子节点；

获取所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置；

根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

7、根据权利要求 6 所述的路由表查找方法，其特征在于，所述在所述路由表的主干中，查找与关键字最长匹配的提升子节点，包括：

从保存在所述路由表的主干中的各个 TRIE 子树的提升子节点中，查找与所述关键字匹配的各个第一前缀；

从所述各个第一前缀中，选取掩码长度最长的第一前缀作为所述与关键字最长匹配的提升子节点。

8、根据权利要求 6 或 7 所述的路由表查找方法，其特征在于，所述根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果，包括：

根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述与关键字最长匹配的提升子节点对应的所述 TRIE 子树各个节点的前缀子节点中，查找与所述关键字匹配的各个第二前缀；

从所述各个第二前缀中，选取掩码长度最长的第二前缀作为所述查找结果。

9、一种路由表建立装置，其特征在于，包括：

划分模块，用于将路由表的数据结构划分为主干和 TRIE 子树；

表项添加模块，用于将路由表项添加到所述 TRIE 子树；

保存模块，用于将所述 TRIE 子树的顶点的提升子节点保存到所述主干中，所述 TRIE 子树的高度小于或等于设置的高度阈值，所述 TRIE 子树的顶点的提升子节点为所述 TRIE 子树的顶点包括的前缀子节点中的子顶点。

10、根据权利要求 9 所述的路由表建立装置，其特征在于，还包括：

分裂模块，用于若所述 TRIE 子树添加路由表项后的高度大于所述高度阈值，则将所述 TRIE 子树中高度大于所述高度阈值的分支从所述 TRIE 子树分裂，得到新 TRIE 子树；

所述保存模块，还用于将所述新 TRIE 子树的顶点的提升子节点保存到所述主干中，所述新 TRIE 子树的顶点为所述 TRIE 子树中高度大于所述高度阈值的分支、与所述 TRIE 子树的顶点连接的节点；

或者还包括：

合并删除模块，用于若所述 TRIE 子树分裂后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到所述上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

11、根据权利要求 10 所述的路由表建立装置，其特征在于，所述合并删除模块还用于，若所述 TRIE 子树删除路由表项后的高度小于所述高度阈值，且所述 TRIE 子树具有上一级 TRIE 子树，将所述 TRIE 子树的顶点合并到上一级 TRIE 子树，从所述主干中删除所述 TRIE 子树的顶点的提升子节点。

12、一种路由表查找装置，其特征在于，包括：

主干查找模块，用于在路由表的主干中，查找与关键字最长匹配的提升子节点，所述路由表的数据结构包括主干和 TRIE 子树，所述 TRIE 子树的顶点的提升子节点保存在所述路由表的主干中；

子树位置获取模块，用于获取所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置；

子树查找模块，用于根据所述与关键字最长匹配的提升子节点对应的

TRIE 子树的顶点的位置，在所述 TRIE 子树中采用 TRIE 算法查找所述关键字对应的查找结果。

13、根据权利要求 12 所述的路由表查找装置，其特征在于，所述主干查找模块包括：

第一前缀子模块，用于从保存在所述路由表的主干中的各个 TRIE 子树的提升子节点中，查找与所述关键字匹配的各个第一前缀；

最长匹配子模块，用于从所述各个第一前缀中，选取掩码长度最长的第一前缀作为所述与关键字最长匹配的提升子节点。

14、根据权利要求 12 或 13 所述的路由表查找装置，其特征在于，所述子树查找模块包括：

第二前缀子模块，用于根据所述与关键字最长匹配的提升子节点对应的 TRIE 子树的顶点的位置，在所述与关键字最长匹配的提升子节点对应的所述 TRIE 子树各个节点的前缀子节点中，查找与所述关键字匹配的各个第二前缀；

查找结果子模块，用于从所述各个第二前缀中，选取掩码长度最长的第二前缀作为所述查找结果。



1/6

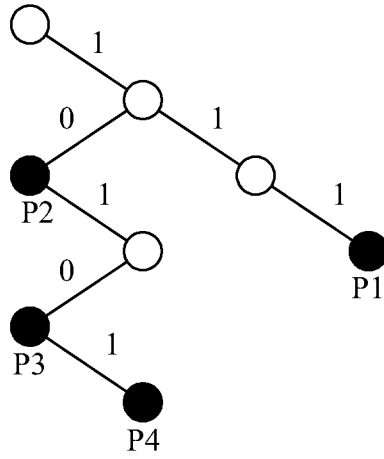


图 1

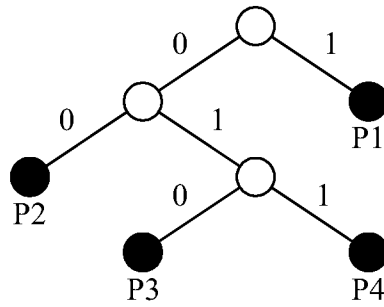


图 2

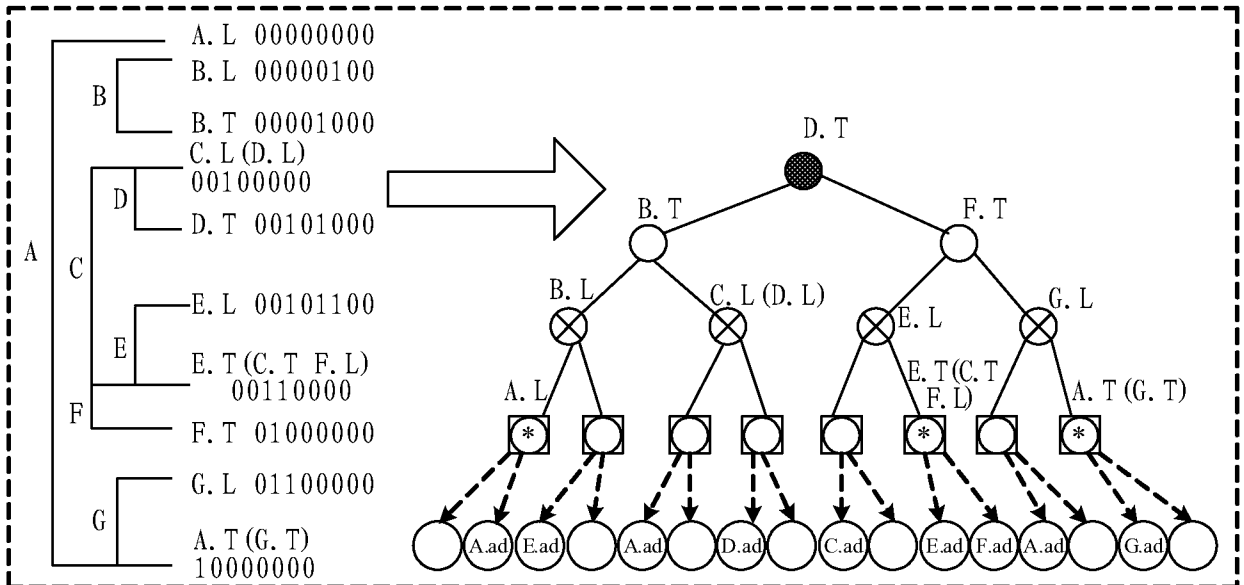


图 3

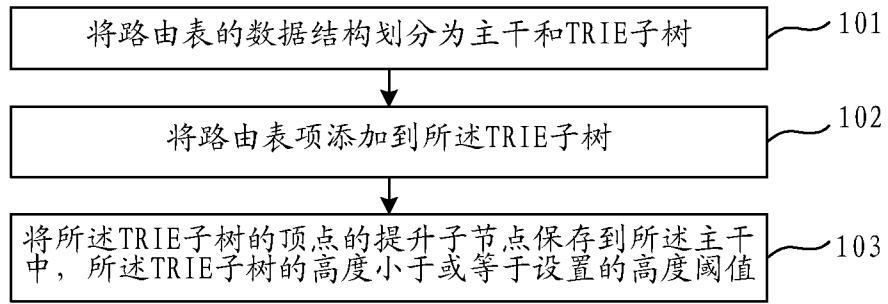


图 4

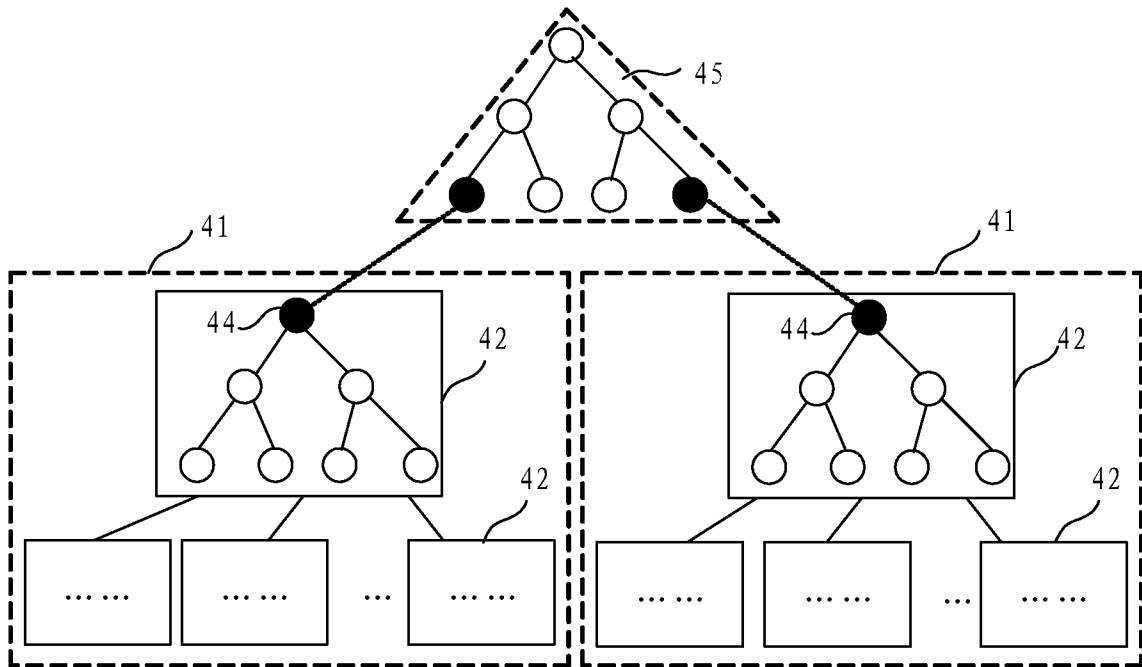


图 5a

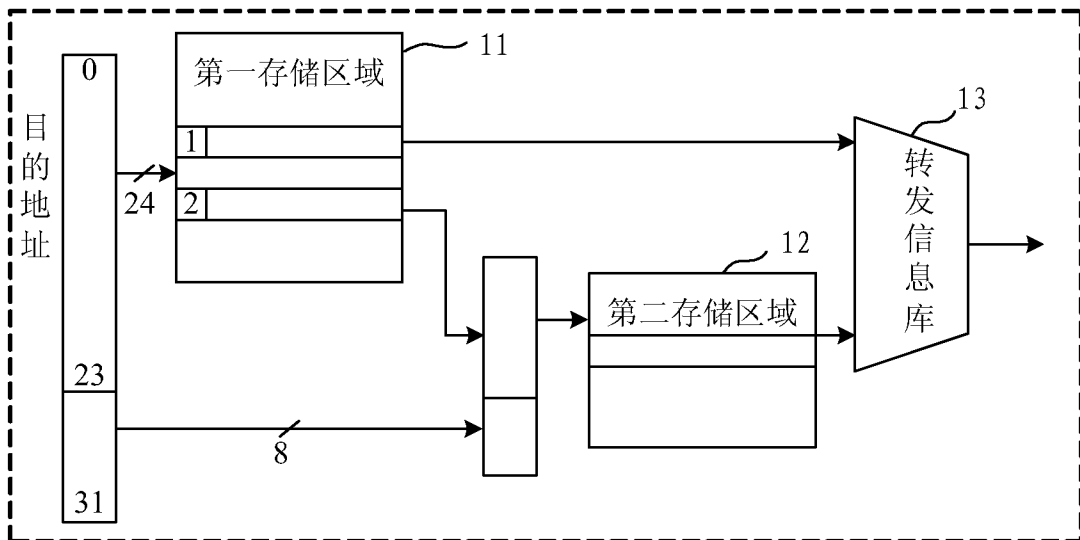


图 5b

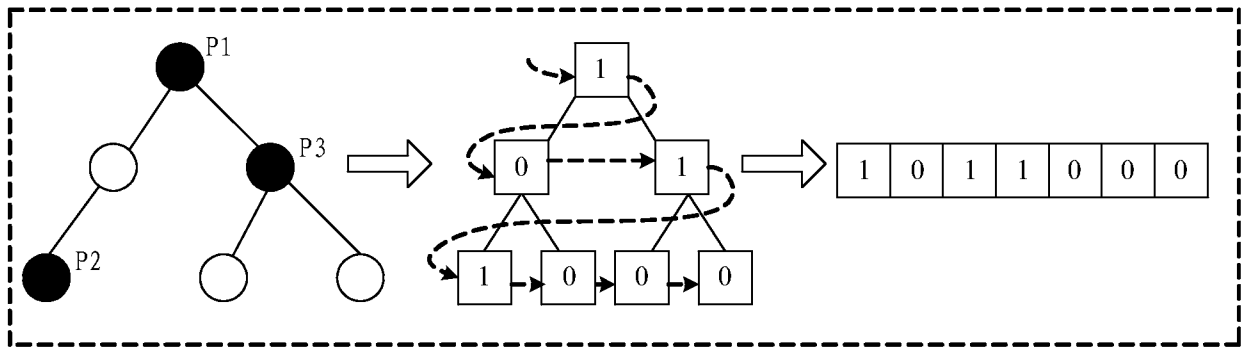


图 5c

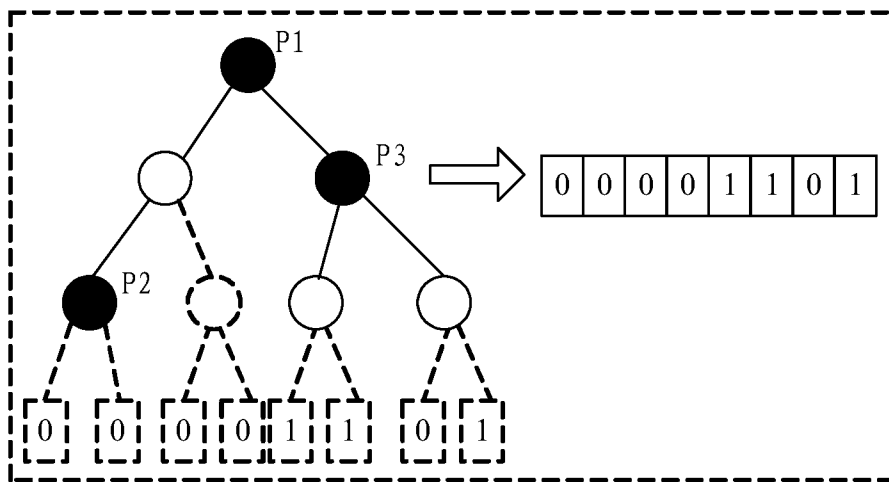


图 5d

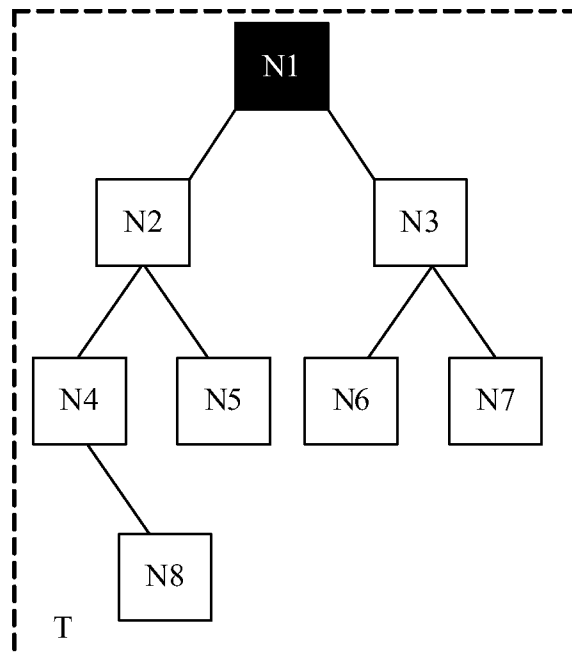


图 6a

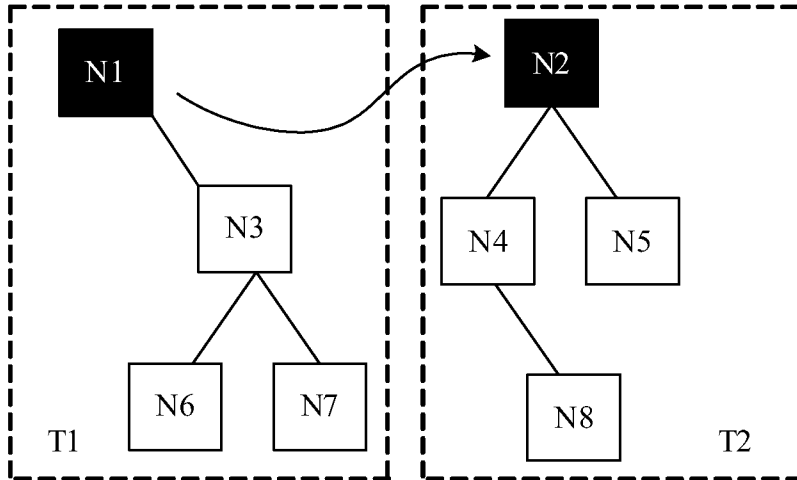


图 6b

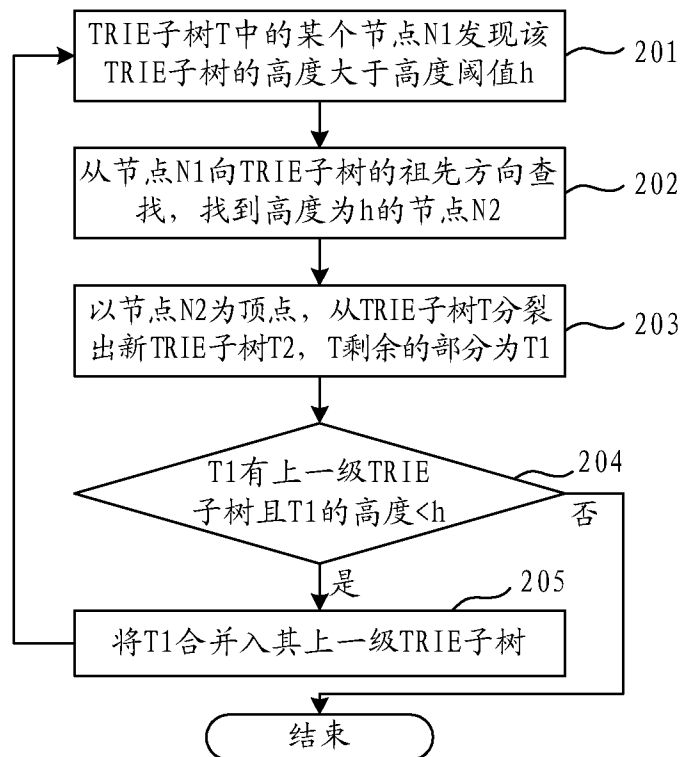


图 6c

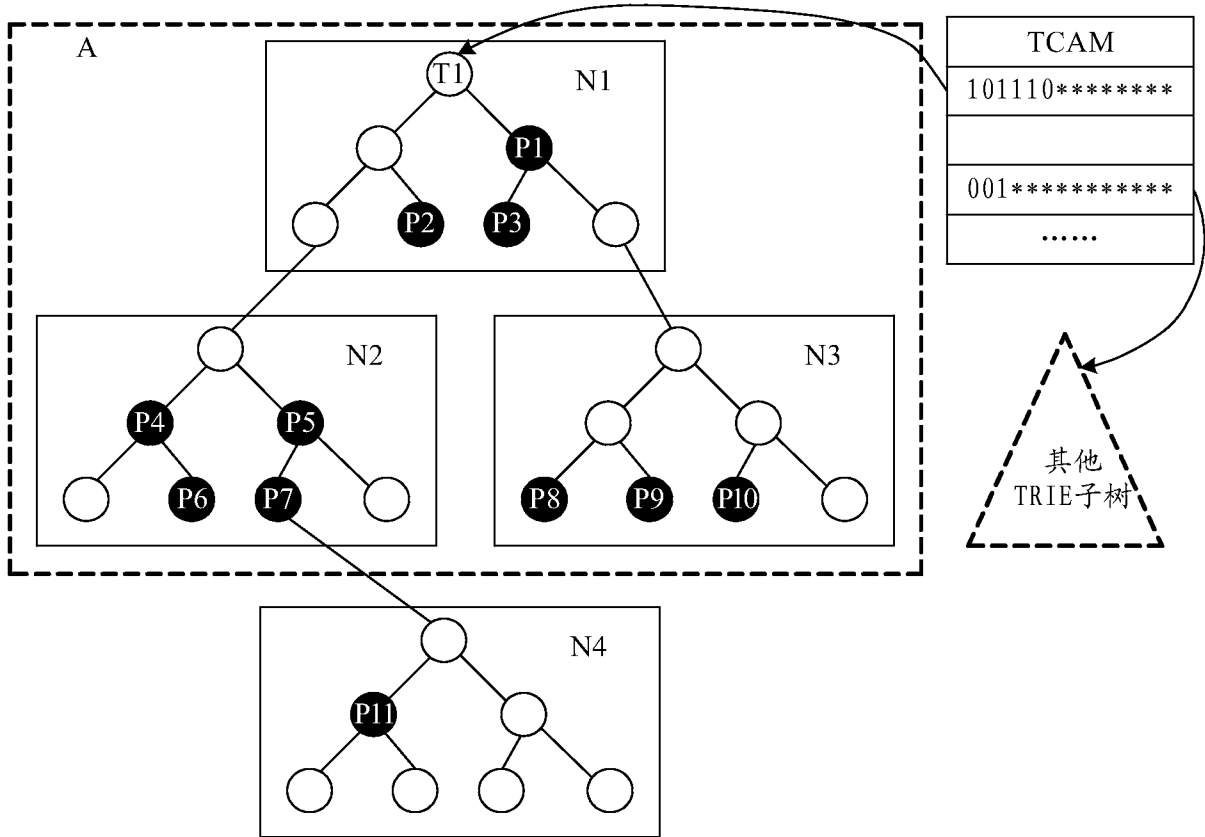


图 6d

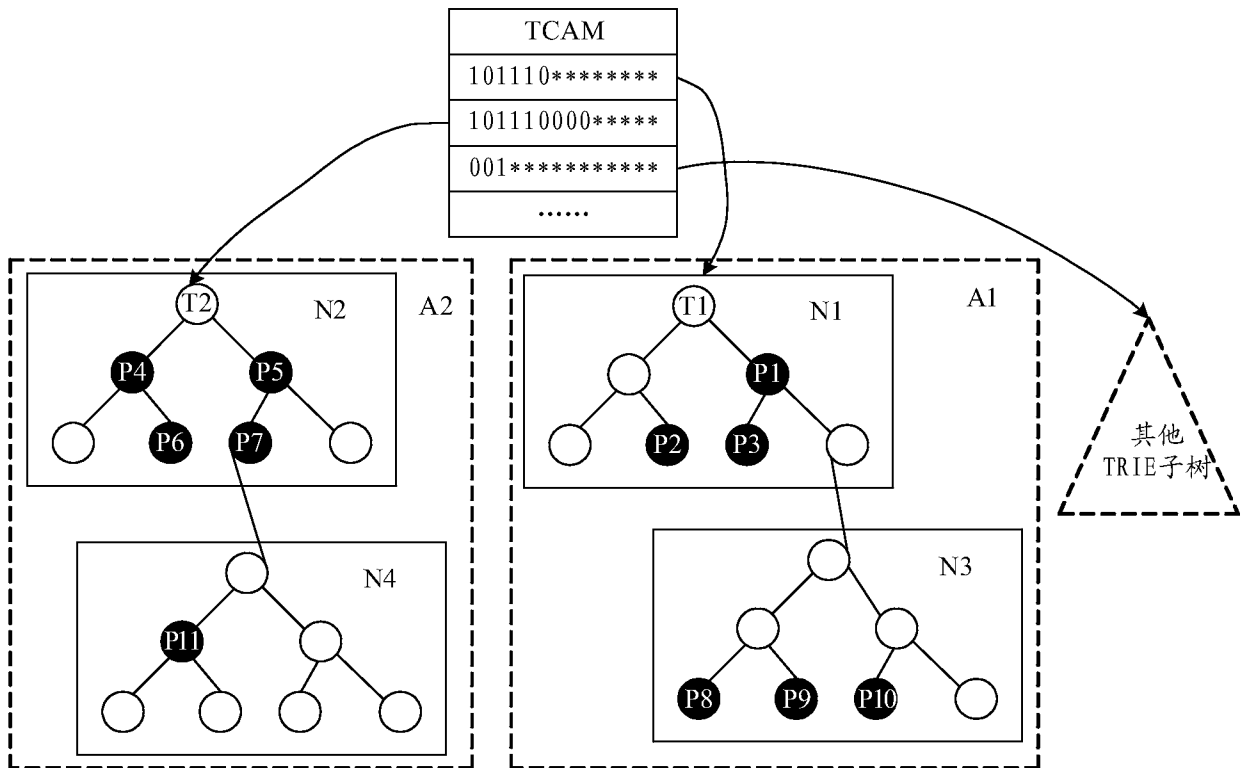


图 6e

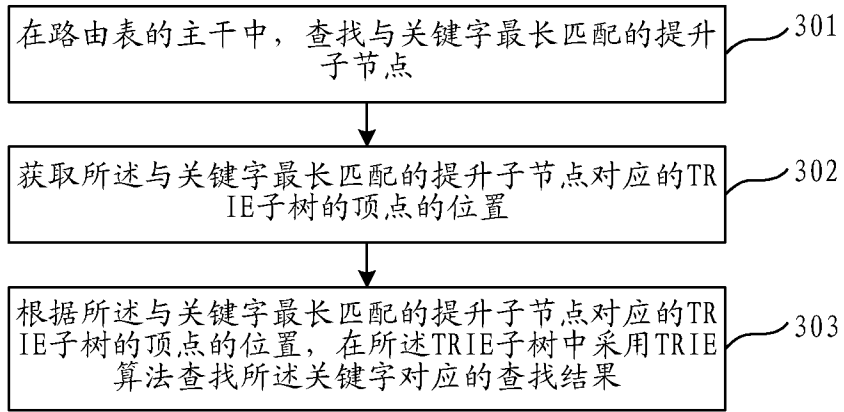


图 7

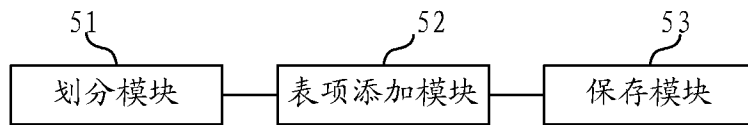


图 8

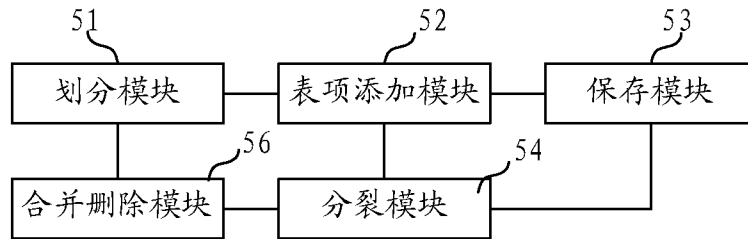


图 9

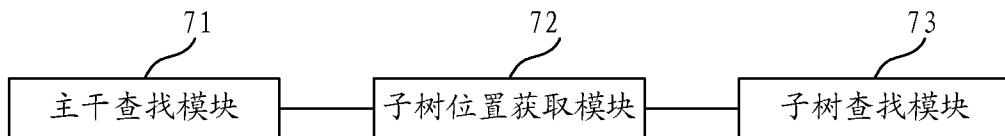


图 10

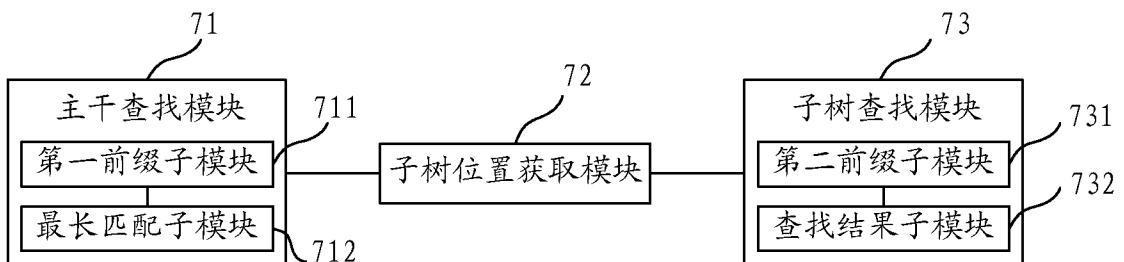


图 11

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2010/071698

## A. CLASSIFICATION OF SUBJECT MATTER

H04L 12/56 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: H04L, G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPIEPODOC CNPAT CNKI TRIE, routing (table), (sub) tree, prefix+, construct+, lookup, entry/entries, node

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 1787477 A (SANXING COMM TECH RES CO LTD B) 14 Jun. 2006(14.06.2006) Abstract, Description Page 1 Lines 23-27, Page 6 Line 29- Page 7 Line 9, Page 9 Lines 22-24, Page 10 Lines 3-6, Fig 3	1, 5-6, 9, 12
A		2-4, 7-8, 10-11, 13-14
Y	CN 101491015 A (UNIV FLORIDA RES FOUND INC) 22 Jul.2009(22.07.2009) Description Page 1 Paragraph 1, Page 7 Lines 12-14, Fig 3	1, 5, 9
Y	CN 101141389 A (HUAWEI TECH CO LTD) 12 Mar. 2008(12.03.2008) Abstract, Description Page 9 Lines 2-19, Claim 1, Figs 4-6	6, 12

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:	“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
“A” document defining the general state of the art which is not considered to be of particular relevance	“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
“E” earlier application or patent but published on or after the international filing date	“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
“L” document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)	“&” document member of the same patent family
“O” document referring to an oral disclosure, use, exhibition or other means	
“P” document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search 10 Jan. 2011(10.01.2011)	Date of mailing of the international search report <b>20 Jan. 2011 (20.01.2011)</b>
---	--

Name and mailing address of the ISA/CN  
The State Intellectual Property Office, the P.R.China  
6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China  
100088  
Facsimile No. 86-10-62019451

Authorized officer

**HAN, Yan**

Telephone No. (86-10)62411764

**INTERNATIONAL SEARCH REPORT**  
Information on patent family members

International application No.  
PCT/CN2010/071698

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 1787477 A	14.06.2006	None	
CN 101491015 A	22.07.2009	WO 2008014188 A1	31.01.2008
		EP 2055050 A1	06.05.2009
		INMUMNP 200900388 E	22.05.2009
		US 2009327316 A1	31.12.2009
CN 101141389 A	12.03.2008	CN 101141389 B	16.06.2010



国际检索报告

国际申请号  
PCT/CN2010/071698

<b>A. 主题的分类</b>		
H04L 12/56 (2006.01) i		
按照国际专利分类(IPC)或者同时按照国家分类和 IPC 两种分类		
<b>B. 检索领域</b>		
检索的最低限度文献(标明分类系统和分类号)		
IPC: H04L, G06F		
包含在检索领域中的除最低限度文献以外的检索文献		
在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))		
WPI EPODOC CNPAT CNKI 路由(转发)表, 特里, 树, 顶点, 前缀, 构建, 查找, 节点, TRIE, routing (table), (sub) tree, prefix+, construct+, lookup, entry/entries, node		
<b>C. 相关文件</b>		
类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
Y	CN 1787477 A (北京三星通信技术研究有限公司) 14.6 月 2006(14.06.2006) 摘要, 说明书第 1 页第 23-27 行、第 6 页第 29 行-第 7 页第 9 行、第 9 页第 22-24 行、第 10 页第 3-6 行, 图 3	1, 5-6, 9, 12
A		2-4, 7-8, 10-11, 13-14
Y	CN 101491015 A (佛罗里达大学研究基金公司) 22.7 月 2009(22.07.2009) 说明书第 1 页第 1 段、第 7 页第 12-14 行, 图 3	1, 5, 9
Y	CN 101141389 A (华为技术有限公司) 12.3 月 2008(12.03.2008) 摘要, 说明书第 9 页第 2-19 行, 权利要求 1, 图 4-6	6, 12
<input type="checkbox"/> 其余文件在 C 栏的续页中列出。 <input checked="" type="checkbox"/> 见同族专利附件。		
* 引用文件的具体类型: “A” 认为不特别相关的表示了现有技术一般状态的文件 “E” 在国际申请日的当天或之后公布的在先申请或专利 “L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的) “O” 涉及口头公开、使用、展览或其他方式公开的文件 “P” 公布日先于国际申请日但迟于所要求的优先权日的文件		“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件 “X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性 “Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性 “&” 同族专利的文件
国际检索实际完成的日期 10.1 月 2011(10.01.2011)		国际检索报告邮寄日期 20.1 月 2011 (20.01.2011)
ISA/CN 的名称和邮寄地址: 中华人民共和国国家知识产权局 中国北京市海淀区蓟门桥西土城路 6 号 100088 传真号: (86-10)62019451		授权官员  韩燕  电话号码: (86-10) 62411764

国际检索报告  
关于同族专利的信息

国际申请号  
**PCT/CN2010/071698**

检索报告中引用的 专利文件	公布日期	同族专利	公布日期
CN 1787477 A	14.06.2006	无	
CN 101491015 A	22.07.2009	WO 2008014188 A1	31.01.2008
		EP 2055050 A1	06.05.2009
		INMUMNP 200900388 E	22.05.2009
		US 2009327316 A1	31.12.2009
CN 101141389 A	12.03.2008	CN 101141389 B	16.06.2010