(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
29 June 2006 (29.06.2006)

PCT

(10) International Publication Number
WO 2006/068976 A1

(51) International Patent Classification:
H04L 12/56 (2006.01)

(21) International Application Number:
PCT/US2005/045756

(22) International Filing Date:
16 December 2005 (16.12.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/014,700 20 December 2004 (20.12.2004) US

(71) Applicant (*for all designated States except US*): CONNECTIVITIES, LLC [US/US]; 2100 EAST MAPLE ROAD, Suite 200, Birmingham, MI 48009 (US).

(72) Inventors; and

(75) Inventors/Applicants (*for US only*): LEVENDEL, Isaac [US/US]; 512 North Mcclurg Court #3907, Chicago, IL 60611 (US). METZ, Reinhard [US/US]; 1926 Berkshire Place, Wheaton, IL 60187 (US). HARA, Jacques [US/US]; 726 South Grove Avenue, Barrington, IL 60010 (US).

(74) Agent: SCHNEIDER, Jerold, I.; Akerman Senterfitt, P.O. Box 3188, West Palm Beach, FL 33402-3188 (US).

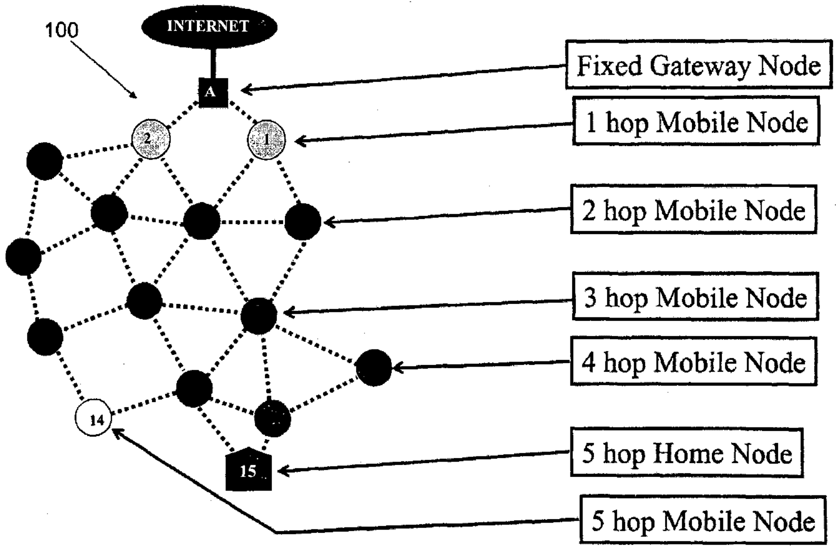
(81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: INTERNET-ORIENTED AD-HOC NETWORK



(57) Abstract: A hierarchical directional internet-oriented ad-hoc network, defined by a software infrastructure, is composed of fixed gateway nodes and a plurality of wireless nodes, which may be fixed or mobile, and which may act as subscribers, routers, or both. The infrastructure hierarchy is defined by the hop count of each node (distance of that node to a fixed gateway node). The software infrastructure includes two tables associated with each node in the network: the upstream routing table which provides shortest routes to fixed gateway nodes through upstream neighbors, and the downstream routing table which provides shortest routes to subscribers through downstream neighbors. These two tables are used by routing algorithms. A peer table can also be used for alternate routes. The maintenance of the aforementioned tables is performed by autonomous algorithms operating locally on each node by receiving and processing signals from their neighbors.

WO 2006/068976 A1

INTERNET-ORIENTED AD-HOC NETWORK

BACKGROUND

Ad-hoc networks are becoming more widely used, especially for mobile wireless devices. An attractive feature of ad-hoc networks is that they do not require a network infrastructure of base stations/fixed gateway nodes to enable communications between wireless nodes. Instead, the wireless nodes are capable of acting as base stations/access points that relay communications for other wireless nodes in the network. Thus, each node can, at various times, act as a source of information, a drain for information, and a router for information.

Traditionally, the focus of ad-hoc networks has been communications between wireless nodes on the network. More sophisticated ad-hoc networks that provide for access to fixed, wired networks have also been proposed. This allows wireless devices to communicate with other types of wired networks, such as the PSTN and the Internet.

One shortcoming associated with known ad-hoc networks, including the more sophisticated ad-hoc networks discussed above, is that they are typically oriented toward enabling communication between nodes, with the direction of such communication being somewhat random. These networks are not as efficient as possible for other types of communication, such as Internet-oriented communication, in which the flow of data is strongly directional (i.e., from fixed gateway nodes downward to wireless nodes and vice versa).

What is needed is a network that can efficiently handle communications such as the Internet that are directionally oriented.

SUMMARY

The aforementioned issues are addressed to a great extent by an ad-hoc network with an internet-oriented, software-defined dynamic infrastructure. The ad-hoc network includes at least one fixed gateway node and a plurality of wireless nodes. As used herein, a fixed gateway node means a node that is in a fixed location and that acts as a gateway, or access point, between the ad-hoc network and another network such as the Internet. In some embodiments, all of the wireless nodes are mobile. In other embodiments, some of the wireless nodes are mobile and some are at fixed locations, which shall be referred to herein as "home nodes." (As used herein, the term "home node" should be understood to refer to a wireless node that is in a fixed location and should not be understood to be limited to a fixed wireless node installed in a residence). At least some of the wireless nodes, and, in some embodiments, all of the wireless nodes, may perform a routing function for other wireless nodes. In embodiments with multiple fixed gateway nodes, the fixed gateway nodes may be connected to the other network via a central node or may be connected directly to the other network. In the latter case, the fixed gateway node serves as a central node.

This ad-hoc network is hierarchical based on distances, measured in hop counts, to fixed gateway nodes. Each of the wireless nodes in the network (which may be fixed wireless nodes or mobile wireless nodes) in the ad-hoc network has a hop count with respect to each fixed gateway node. Any given wireless node may

have one or more neighborhood nodes with which the wireless node can communicate directly. The neighborhood nodes will be either upstream (i.e., closer, as measured by hop count, to the fixed gateway node), downstream (further away, as measured by hop count, from the gateway node), or at the same distance (referred to herein as a peer node).

Each wireless node in the network also has at least one of each of four tables that describe the node's neighborhood and that are used for routing and other functions: 1) a downstream neighbor table, 2) a downstream routing table, 3) an upstream routing table, and 4) a peer table. The upstream routing table lists each upstream node in the wireless node's neighborhood together with a hop count to the fixed gateway node. In embodiments with multiple fixed gateway nodes, there is a plurality of upstream routing tables and each upstream routing table pertains to a different fixed gateway node. The peer routing table lists each peer node in the node's neighborhood along with an associated hop count to the fixed gateway node and, in embodiments with multiple fixed gateway nodes, each node has a separate peer table for each fixed gateway node. The downstream neighborhood table lists each downstream neighbor with respect to a particular fixed gateway node (again, there is a separate downstream neighborhood table for each fixed gateway node in embodiments with multiple fixed gateway nodes). The downstream routing table lists each downstream node (including downstream neighborhood nodes) reachable from the node together with an associated hop count, and in embodiments with multiple fixed gateway nodes, there is a multiplicity of downstream routing tables and each downstream routing table pertains to a different fixed gateway node. The aforementioned tables define the connectivity for the network. A number of triggers are generated during routing and at other times to cause the update of these tables. The tables are also audited periodically, either on an individual node basis or for the tables as a whole.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned advantages and features will be more readily understood with reference to the following detailed description and the accompanying drawings in which:

Figure 1 is a schematic diagram of a network with one fixed gateway node according to a first embodiment.

Figure 2 is a schematic diagram of a network with two fixed gateway nodes according to a second embodiment.

Figures 3a and 3b are conceptual schematic diagrams illustrating two superimposed networks that together comprise the network of Figure 2.

Figure 4 is a schematic diagram of a network with fixed gateway nodes routed through a central node according to a third embodiment.

Figure 5 is a logic diagram illustrating a packet routing process.

Figure 6 is a flowchart illustrating in further detail the processing associated with one of the steps of Figure 5.

Figure 7 is a flowchart illustrating in further detail the processing associated with another of the steps of Figure 5.

Figure 8 is a flowchart illustrating the processing associated with a downstream trigger D1.

Figure 9 is a flowchart illustrating the processing associated with a downstream trigger D2.

Figure 10 is a flowchart illustrating the processing associated with an upstream trigger U1.

Figure 11 is a flowchart illustrating the processing associated with a trigger T4.

Figure 12 is a logic diagram illustrating various processing of a trigger T5 depending upon the difference in hop counts between the sending and receiving nodes.

Figure 13 is a flowchart illustrating in greater detail the processing associated with one of the steps of Figure 12.

Figure 14 is a flowchart illustrating in greater detail the processing associated with one of the steps of Figure 12.

Figure 15 is a flowchart illustrating in greater detail the processing associated with one of the steps of Figure 12.

Figure 16 is a flowchart illustrating in greater detail the processing associated with one of the steps of Figure 12.

Figure 17 is a flowchart illustrating in greater detail the processing associated with one of the steps of Figure 12.

DETAILED DESCRIPTION

In the following detailed description, a plurality of specific details, such as numbers of nodes and hops, are set forth in order to provide a thorough understanding of the embodiments described herein. The details discussed in connection with the preferred embodiments should not be understood to limit the present invention. Furthermore, for ease of understanding, certain method steps are delineated as separate steps; however, these steps should not be construed as necessarily distinct nor order dependent in their performance.

An exemplary network 100 is illustrated in Figure 1. The network 100 includes a fixed gateway node A, a plurality of mobile wireless nodes 1-14, and a home wireless node 15. The fixed gateway node A is connected to an Internet backbone and has wireless transmission and reception capability that allows it to act as an access point for a plurality of wireless nodes. Mobile wireless nodes 1-14 and home wireless node 15 also have wireless transmission and reception capability that allow them to communicate with other wireless nodes in the network and with the fixed gateway node (provided that the fixed gateway node is within range of the wireless transmission and reception system). Each of the mobile nodes 1-14 have the ability to act as routers for other wireless nodes in the network. (In alternative embodiments, only a portion of the mobile nodes have this ability.) The home node 15 does not have the ability to act as a router for other subscriber nodes in the embodiment of Figure 1. Although only one home node 15 is illustrated in Figure 1, it should be understood that there may be a plurality of such home nodes in other embodiments and that some or all of such home nodes may have the ability to act as routers. It should also be understood that, in various embodiments, a particular wireless node, whether it be mobile or fixed, may be configured such that it only acts as a router, only act as a subscriber (i.e., a source or drain of information) or acts as both a router and a subscriber.

As discussed above, the network 100 is an Internet-oriented network. Accordingly, each of the wireless nodes 1-15 can be classified based on the number of hops, or hop count, measured with respect to the fixed gateway node A. Nodes 1 and 2 have a hop count of 1, nodes 3-6 have a hop count of 2, nodes 7-9 have a hop count of 3, nodes 10-13 have a hop count of 4, and nodes 14 and 15 have a hop count of 5.

Each wireless node may have one or more other wireless nodes with which it is directly connected. As used herein, a second node is "directly connected" to a first node when the first node can communicate with the second node using its wireless communication system without requiring any other node to relay messages between the first and second nodes. The set of nodes that are directly connected to a node form the neighborhood for that node. The neighborhood for any wireless node can include nodes with lower hop counts (upstream nodes), nodes with the same hop count (peer nodes), and nodes with lower hop counts (downstream nodes).

Each of the nodes of the network 100 have at least one neighborhood node. For example, the neighborhood for node 5 includes upstream nodes 1 and 2, peer nodes 4 and 6, and downstream nodes 8 and 9. Every node in the network 100 has at least one upstream node (which may be the fixed gateway node A or another wireless node), and some have a plurality of upstream nodes. At any given time in any particular network, a wireless node may have zero (in which case it is isolated), one or many upstream nodes and may have zero, one or many peer nodes and zero, one or many downstream nodes. Each node will have downstream neighborhood tables (DNTs) and peer tables (PTs) that list each downstream and peer neighbor, respectively, along with the corresponding hop count relative to the fixed gateway node.

Each wireless node will also have an upstream routing table (URT) which will include the fixed gateway node with which the URT is associated and all upstream nodes (nodes with lower hop counts) in that node's neighborhood. The URT will also include a hop count for each of the neighboring nodes listed in the URT. Exemplary URTs for nodes 1, 5, and 8 are provided in Tables 1, 2 and 3 below.

Table 1 - URT for Node 1

Node	*Hop Count*
A	1

Table 2 - URT for Node 5

Node	*Hop Count*
1	1
2	1

Table 3 - URT for Node 8

Node	*Hop Count*
4	2
5	2

The PT for a node will have a format similar to that of the URT, but will list peer neighbors rather than upstream neighbors. A detailed discussion of how the URTs and PTs are utilized for routing packets is set forth below.

Each node also has a downstream routing table, or DRT, which the node will utilize in order to determine how to rout packets downstream. The DRT for a node includes each node that is reachable from a node by traveling in a purely downstream direction regardless of the number of hops. In other words, an other node is included in the DRT for a particular node if and only if a path exists from the particular node to the other node, and that path is purely downstream (i.e., each successive node on the path has a higher hop count than the previous node). One result of the foregoing is that routing will always be done through the shortest path as measured by hop count. Another consequence is that the DRT of a node with only upstream and/or peer neighbors will be empty.

Three different types of downstream routing tables may be utilized: DRTs indexed by destination node, DRTs indexed by downstream neighbors, and DRTs double-indexed by both destination node and by downstream neighbors. Examples of the first type of DRT for nodes 1, 2 and 5 and fixed gateway node A are presented below in tables 4-7:

Table 4 - DRT Indexed by Destination Node for Node 1

Node	*Hop Count*	*Through Downstream Neighbors*
4	1	--
5	1	--
8	2	4, 5
9	2	5
10	3	5
11	3	4,5
12	3	4,5
13	3	4,5

14	4	4,5

Table 5 - DRT Indexed by Destination Node for Node 2

Node	*Hop Count*	*Through Downstream Neighbors*
3	1	--
5	1	--
6	1	--
7	2	3,6
8	2	5
9	2	5,6
10	3	3,5,6
11	3	5
12	3	5
13	3	5,6
14	4	3,5,6

Table 6 - DRT Indexed by Destination Node for Node 5

Node	*Hop Count*	*Through Downstream Neighbors*
8	1	--
9	1	--
10	2	9
11	2	8
12	2	8
13	2	8,9
14	3	8,9

Table 7 - DRT Indexed by Destination Node for Fixed Gateway Node A

Node	*Hop Count*	*Through Downstream Neighbors*
1	1	--
2	1	--
3	2	2
4	2	1
5	2	1,2
6	2	2
7	3	2
8	3	1,2
9	3	1,2
10	3	1,2
11	4	1,2
12	4	1,2
13	4	1,2
14	5	1,2
15	5	1,2

Certain aspects of the DRTs listed above are worth noting. First, for all nodes in the DRT that are not directly accessible, the third column of the DRT indicates the directly accessible neighboring nodes through which such non-directly accessible nodes can be reached.

A second aspect of the DRT tables is that not all nodes with higher hop counts that could possibly be reached from a given node are included in the DRT. For example, the DRT for node 2 does not include an entry for node 4 even though node 4 has a higher hop count (2, as compared to a hop count of 1 for note 2) and even though there is a path from node 2 to node 4 through node 5 that does not require any upstream travel. The reason why node 4 is not included in the DRT for node 2 is that the portion of the aforementioned path from node 5 to node 4 is not purely downstream because both node 4 and node 5 have a hop count of 2 (i.e., nodes 4 and 5 are peers). Similarly, node 8 is listed in the DRT for node 2, but no path through node 6 is shown. Again, this ensures that packets will be routed upstream toward the fixed gateway node through the shortest path as measured by hop counts.

A third aspect of the DRT tables is that multiple paths are shown in some instances. For example, the DRT for node 1 shows that node 11 is reachable in three hops via either node 4 or node 5. The choice between possible paths can be made by the node based on a random selection, relative loading of the multiple nodes, or any other technique.

A second type of DRT is indexed by downstream neighbors rather than by destination node. For each downstream neighboring node, the DRT includes a list of all nodes reachable through purely downstream paths along with an associated hop count. This type of DRT is advantageous because its construction is simple - the DRTs of downstream neighboring nodes are simply concatenated. However, this type of DRT requires a search of the DRT table in order to select a shortest path for a particular destination. Examples of this second type of DRT for nodes 2, 3 and fixed gateway node A are set forth below in Tables 8-10 below:

Table 8: DRT Indexed By Downstream Neighbor for Node 2

Nodes Reachable Through Node 3 / HC	*Nodes Reachable Through Node 5 / HC*	*Nodes Reachable Through Node 6 / HC*
3/1	5/1	6/1
7/2	8/2	7/2
10/3	9/2	9/2
14/4	10/3	10/3
	11/3	13/3
	12/3	14/4
	13/3	15/5
	14/4	
	15/5	

Table 9: DRT Indexed By Downstream Neighbor for Node 3

Nodes Reachable Through Node 7 /HC
7/1
10/2
14/3

Table 10: DRT Indexed By Downstream Neighbor for Fixed Gateway node A

Nodes Reachable Through Node 1 /HC	*Nodes Reachable Through Node 2 /HC*
1/1	2/1
4/2	3/2
5/2	5/2
8/3	6/2
9/3	7/3
10/4	8/3
11/4	9/3
12/4	10/4
13/4	11/4
14/5	12/4
15/5	13/4
	14/5
	15/5

As alluded to above, an advantage of using DRTs indexed by downstream neighboring nodes is that they are easily constructed and updated using information from downstream nodes. Each column of the DRTs above represents the downstream cluster of the corresponding downstream neighbor. The downstream cluster for any particular node can be formed by simply forming the union of each of the columns of the DRT for that node, adding 1 to each of the hop counts in the union, and then adding the particular node along with a hop count of 0. Thus, for example, downstream cluster for node 2 (DC_2) is shown below in table 11:

Table 11: DC_i for Node 2

2/0	*Node 2 itself with HC=0*
3/1	------------/
5/1	/ *union of columns of*
6/1	/ *DRT of node 2 with*
7/2	/ *associated hop counts*
8/2	/
9/2	/
10/3	/
11/3	/
12/3	/
13/3	/
14/4	/
15/4	---------/

As will be discussed in further detail below, the DC for a node is sent by that node to its upstream neighbors in a trigger message.

The third type of DRT is double indexed by both destination and downstream neighbor. An example of this type of double-indexed DRT for node 2 is provided in Table 12 below (where "x" signifies that a route exists between the given node and the destination node corresponding to a row through the downstream neighbor corresponding to a column):

Table 12 - Double-Indexed DRT for Node 2

Destination Node	*Nodes Reachable Thru Node 3 / HC*	*Nodes Reachable Thru Node 5 / HC*	*Nodes Reachable Thru Node 6 / HC*
3	x/1		
5		x/1	
6			x/1
7	x/2		x/2
8		x/2	
9		x/2	x/2
10	x/3	x/3	x/3
11		x/3	
12		x/3	
13		x/3	x/3
14	x/4	x/4	x/4
15		x/4	x/4

Double-indexed DRT tables have the advantages of efficiency for both construction and routing. In preferred embodiments, the DRTs are represented as sparse matrices when used with large numbers of nodes.

In the network 100 of Figure 1, there is only a single fixed gateway node A. However, it will be readily apparent that networks sometimes include multiple fixed gateway nodes. An example of a network 200 with the same wireless nodes 1-15 and two fixed gateway nodes A and B is illustrated in Figure 2. As illustrated in Figures 3(a) and 3(b), the network 200 can be thought of as the superimposition of the two networks 300, 400, one with fixed gateway node A and one with fixed gateway node B. Thus, the methods set forth above with respect to the network

100 of Figure 1 can be extended to the two fixed gateway node network 200 of Figure 2 by creating URTs, PTs, and DRTs for each node for each of the individual networks illustrated in Figures 3(a) and 3(b).

Some nodes (e.g., node 1) will have only a single URT because only one fixed gateway node is upstream of that node. Other nodes (e.g., node 3) will have multiple URTs for multiple fixed gateway nodes, but one URT will have a shorter route than the other (node 3 is one hop from fixed gateway node B but is two hops from fixed gateway node A). In this case, the URT corresponding to the shortest distance (smallest number of hops) is designated as the primary URT and the other URT is designated as the secondary URT. The secondary URTs can be used in cases where the path to the primary fixed gateway node associated with the primary URT is blocked. Finally, still other nodes will have multiple URTs with the same minimum distance/hop count. In such cases, both URTs will be designated as primary and both will used for routing purposes. The choice of which of the multiple URTs to use can be based on load balancing, random selection, or some other process.

Maintaining multiple node associations (through primary and secondary URTs or multiple primary URTs as well as in a single URT) is useful and important for three reasons: 1) as a vehicle moves, it may drop its principal association with one fixed gateway node and move to a new one; 2) a failure in part of the network may be recovered by using alternate routing through alternate nodes; and 3) alternate paths may be used for load balancing purposes.

In the network 200 illustrated in Figure 2 , node 3 is only associated with fixed gateway node B and node 1 is only associated with fixed gateway node A.

Also, node 3 is not in either the DRT or the URT for node 1, and vice-versa. One way in which to effect communications between these nodes is via the Internet. However, in other embodiments of the invention, the fixed gateway nodes are linked to a central node which is then connected to the Internet. An example of such a network 400 with fixed gateway nodes A and B linked to central node 410 is illustrated in Figure 4. In such an embodiment, the central node 410 has a downstream routing table for each of the fixed gateway nodes and each of the wireless nodes in the network. Exemplary DRTs are set forth in Tables 13 and 14 below (although not shown below, double-indexed DRTs are also possible):

Table 13 - Central Node DRT with Indexing by Downstream Neighbors

Target Node	*Hop Count*	*Through Downstream Neighbors*
A	1	
B	1	
1	2	A
2	2	A,B
3	2	B
4	3	A
5	3	A,B
6	3	A,B
7	3	B
8	4	A,B
9	4	A,B
10	4	A,B
11	5	A,B
12	5	A,B
13	5	A,B
14	5	A
15	6	A,B

Table 14 - Central Node DRT with Indexing by Destination Nodes

Nodes Reachable Through A /HC	*Nodes Reachable Through B /HC*
A/1	B/1
1/2	2/2
2/2	3/2
4/3	5/3
5/3	6/3
6/3	7/3
8/4	8/4
9/4	9/4
11/5	10/4
12/5	11/5
13/5	12/5
15/6	13/5
	14/5
	15/6

A node associated with multiple fixed gateway nodes A, B, C, etc. will have one set of the URT, PT, DNT and DRT for each of the corresponding fixed gateway nodes A, B, C, etc., respectively.

The routing algorithm from the internet to a subscriber (downstream routing) uses the DRTs to select one of several possible shortest routes to the subscriber. The routing algorithm from a subscriber to the Internet uses the URTs to select one of several possible shortest routes to the Internet. Subscriber to subscriber routing will use both DRTs and URTs. Alternate routing through upstream and downstream neighbors may be chosen in the case of routing failure, for “handover” from one fixed gateway node to another, or for load balancing.

The creation of the routing tables, and hence the network, will now be discussed. The process begins by constructing upstream routing tables. Initially, all wireless nodes have an infinite hop count, no neighbors, and empty URTs, and fixed gateway nodes have a zero hop count, no downstream neighbors and empty DRTs. As wireless nodes detect other nodes (which may be accomplished through periodic broadcast polling messages), the other wireless nodes are registered into that node's PT with an equal infinite hop count. As the fixed gateway nodes detect directly connected wireless nodes, those wireless nodes are assigned a hop count of 1. The wireless nodes detected by the fixed gateway node then propagate the information concerning the fixed gateway node to other nodes they have previously detected as peers and to new wireless nodes detected thereafter (the techniques by which this information is propagated will be discussed in further detail below). In this manner, the upstream hierarchy is established.

The DRT construction process can be triggered in either of two ways: 1) when the process of URT construction reaches nodes without downstream neighbors; or 2) when a node modifies its URT. In addition, events encountered during packet routing operations will also trigger modifications to the routing tables as discussed in further detail below.

Use of the routing tables to perform routing operations will now be discussed with reference to the logic diagram 500 of Figure 5. The process begins when the next packet arrives at step 510. If the packet is intended for the node at which it is received at step 520, the process is complete and step 510 is repeated. Otherwise, the direction of routing required - upstream, downstream, or subscriber-to-subscriber - is determined. There are several ways in which the routing direction of a packet can be determined. In some embodiments, each node can have separate buffers for upstream, downstream and subscriber-to-subscriber packets. In other embodiments, the routing process determines the direction based on the destination. In still other embodiments, the packets include a flag that indicates the direction. Other techniques will be readily apparent to those of skill in the art and are within the purview of the invention.

If downstream routing is required, subroutine 530 is performed. If upstream routing is required, subroutine 540 is performed. Finally, if subscriber-to-subscriber routing is required, subroutine 550 is performed.

The downstream routing subroutine 530 of Figure 5 is illustrated in further detail in the flowchart 600 of Figure 6. A downstream neighbor is selected from the DRT at step 531. If the destination node is a downstream neighbor, the packet is transmitted directly to that node. If a destination node is not a downstream neighbor (i.e., is not directly connected) but there is only a single path to that node available, the downstream neighbor node associated with that path is chosen. Otherwise, if multiple paths to the destination node are available, a choice between the available paths is made. The choice can be made any number of ways, including random selection from among the available paths, selection of the first available path found in the routing tables, selection of the least loaded downstream neighbor, etc. As will be discussed further below, peer routing is also possible.

If the selection of a downstream neighbor at step 531 was successful (i.e., a downstream neighbor was found in the routing tables) at step 532, an attempt to transmit the packet to the selected downstream neighbor is made at step 533. If the packet was successfully transmitted to the selected downstream neighbor at step

534, the downstream routing subroutine ends and control returns to step 510 of Figure 5 for processing of the next packet.

If the attempt at step 533 to transmit the packet to the selected downstream neighbor was unsuccessful at step 534, then a trigger D1 is generated at step 536 and a routing failure procedure is initiated at step 537. Triggers, including the trigger D1, are messages that trigger a routing table update process upon the occurrence of some event. Triggers and the updating of routing tables will be discussed in further detail below. The routing failure procedure of step 637 may be handled in a number of ways. One possibility is that the packet is simply dropped, which will result in the sender failing to receive an acknowledgment from the destination node. Another possibility is to send a failure message to the sending node. This will allow the sending node to send another packet as soon as possible (i.e., without waiting for a timeout for an acknowledgment message). This may be desirable for time-sensitive applications, but there is a performance penalty associated with sending such failure messages. Other possibilities will be apparent to those of skill in the art.

In addition to the trigger D1 of step 536, a second trigger D2 will be generated at step 538 if no downstream neighbor could be located in the DRT at step 531. The D2 trigger occurs because the upstream neighbor's DRT indicated that a path to the destination node was available through a node but that node's DRT does not include the destination node. The processing of the D2 and other triggers will be discussed in further detail below.

The upstream routing subroutine 540 of Figure 5 is illustrated in further detail in the flowchart 700 of Figure 7. An upstream neighbor is selected from the

URT at step 541. If the destination node is the upstream neighbor, the packet is transmitted directly to that node. If a destination node is not a an upstream neighbor (i.e., is not directly connected) but there is only a single path to that node available, the upstream neighbor node associated with that path is chosen. (Note that this will be the case where the hop count of the receiving node is 1, because the only upstream neighbor that will be fixed gateway node.) Otherwise, if multiple paths to the destination node are available, a choice between the nodes in the URT (excluding the fixed gateway node, which cannot be directly connected if multiple paths exist) is made. As discussed above in connection with the downstream routing process of Figure 6, the choice can be made any number of ways, including random selection from among the available paths, selection of the first available path found in the routing tables, selection of the least loaded upstream neighbor, etc. Again, peer routing is also possible.

If the selection of an upstream neighbor at step 541 was successful (i.e., an upstream neighbor was found in the routing tables) at step 542, an attempt to transmit the packet to the selected upstream neighbor is made at step 543. If the packet was successfully transmitted to the selected upstream neighbor at step 544, the upstream routing subroutine ends and control returns to step 510 of Figure 5 for processing of the next packet.

If the attempt at step 543 to transmit the packet to the selected downstream neighbor was unsuccessful at step 544, then a trigger U1 is generated at step 546. Again, the processing of triggers will be discussed in further detail below. After the U1 trigger is generated at step 546, or if an upstream neighbor could not be located at step 542, a routing failure procedure is initiated at step 546. Like the downstream routing failure procedure, the upstream routing failure procedure of step 546 may be handled in a number of ways. One possibility is that the packet is simply dropped, which will result in the sender failing to receive an acknowledgment from the destination node. A second possibility is to send a failure message to the sending node.

The subscriber-to-subscriber routing subroutine 550 of Figure 5 functions by utilizing a combination of the upstream and downstream routing procedures. When a subscriber node wishes to send a packet to another subscriber node that is not in that node's DRT, the packet is sent using the upstream routing subroutine 540 described above in connection with the flowchart 700 of Figure 7. When the packet reaches the central node, the central node will send the packet downstream using the downstream routing subroutine 530 described above in connection with the flowchart 600 of Figure 6.

The routing algorithms discussed above do not use the nodes in the PTs to route packets to peers. Thus, the PTs are only used in the event of changes to the routing tables (e.g., through trigger messages as will be discussed in further detail below). However, as alluded to above, the routing algorithms may be modified to use the PTs. In some embodiments, the PTs are used as alternate upstream routes. In other embodiments, the PTs may be used for downstream routing. In such cases, because peer neighbors do not necessarily include the same subscribers in their DRTs, the construction of the DRTs is modified to include the DRTs of peers as well. This allows for the use of alternate downstream routes through peers whenever available and useful without modification of the downstream routing process.

Triggers will now be discussed in greater detail. As mentioned above, triggers are messages that are generated upon the occurrence of some event that trigger the updating of routing tables at the receiving node. The processing of triggers is handled locally by the node receiving the trigger, and the processing of a trigger may generate another trigger of the same type or of a different type. As discussed above, three triggers - D1, D2 and U1 - are generated by the routing algorithms. The processing of these triggers will be discussed in detail.

Trigger D1 occurs when a packet cannot be sent successfully to a downstream neighbor in a node's DRT. The processing of trigger D1 is shown in the flowchart 800 of Figure 8. Upon receipt of a D1 trigger, the downstream neighbor N_k to which the packet could not be sent is taken out of the downstream neighborhood table at step 810. If the DRT is of the type indexed by downstream neighbor, the column of the DRT corresponding to the unavailable downstream neighbor is updated at step 820. (In embodiments in which the DRTs are indexed by destination node or are double indexed by destination node and downstream neighbor, appropriate modifications to the network tables are made.) The downstream cluster is then computed at step 830 by calculating the union of the columns of the DRT and adding the node N_i and its hop count 0 (represented symbolically as {Ni, 0} in Figure 8) as discussed above. Next, a T4 trigger message including the downstream cluster is sent to upstream neighbors at step 840 (and to peers in embodiments in which peer routing is implemented) so that these neighbors can update their routing tables. The process is then complete.

Trigger D2 occurs when a packet directed to a destination node is received at a node that does not have the destination node in its DRT. The processing of trigger D2 is shown in the flowchart 900 of Figure 9. Upon receipt of a D2 trigger, the downstream cluster of the receiving node is calculated at step 910 and a new T4 trigger message including the downstream cluster is sent to upstream neighbors at step 920 to trigger the update of the routing tables of the upstream node that sent the packet. The process is then complete.

Trigger U1 occurs when a packet cannot be sent successfully to an upstream neighbor in a node's URT. The processing of trigger U1 is shown in the flowchart 1000 of Figure 10. The process begins by removing the upstream neighbor to which the packet could not be sent from the URT for that node at step 1010. If the URT is not empty at step 1020 (meaning there is another upstream neighbor through whom packets can be sent), the process ends. If the URT is empty at step 1020, node tables are re-computed at step 1030 as follows: the peer table becomes the upstream routing table, and the downstream neighborhood table becomes the peer table. The downstream neighborhood table and downstream routing tables are then empty. If the URT is still empty at step 1040, the hop count for that node is set to infinity at step 1050 and processing ends. If the URT is not empty at step 1040, the downstream cluster for the node is set to $\{N_i + 0\}$ at step 1060 and a T4 trigger message including the downstream cluster is sent to the upstream neighbors in the URT at step 1070 and processing ends.

The T4 trigger is generated during the processing of the routing triggers as discussed above. The purpose of the T4 trigger is to propagate downstream connectivity changes to upstream nodes in order to update their DRTs. The processing of a T4 trigger is illustrated by the flowchart 1100 of Figure 11. The process begins at step 1110 with increasing by 1 the hop counts of the nodes in the received Trigger T4 and updating the downstream routing table which, in embodiments with DRTs indexed by downstream neighbor, involves replacing the corresponding column in the DRT with the new column received in the T4 trigger message. If the hop count for the node is zero at step 1120 (signifying that the highest level node has been reached), the process ends as there are no further upstream nodes. If the hop count is not zero at step 1120 (signifying that there is an upstream neighbor), the downstream cluster is calculated at step 1130 and sent to all upstream nodes in the URT in a new T4 trigger message at step 1140 and the process is complete.

In addition to triggers T1-T4, there is a trigger T5. The T5 trigger is generated by a periodic broadcast. That is, each node periodically broadcasts its node ID and hop count to inform neighboring nodes of its presence. When a broadcast message from another node is received that indicates a change of some kind, the T5 trigger is the mechanism that propagates the change through the network.

T5 trigger processing is illustrated by the flowchart 1200 of Figure 12. Processing begins at step 1201, where the hop count of the node receiving the T5 trigger message (HC_i) is compared to the hop count (HC_k) of the node that sent the T5 trigger message. If the hop count of the receiving node is more than 2 hops downstream of the sending node, the processing at step 1210 is performed. If the hop count of the receiving node is exactly 2 hops downstream of the sending node, the processing of step 1220 is performed. If the hop count of the receiving node is exactly 1 hop downstream of the sending node, the processing of step 1230 is performed. If the hop counts are equal, the processing of step 1250 is performed.

Finally, if the receiving node is upstream of the sending node (the hop count of the sending node is greater than or equal to the hop count of the receiving node plus one), the processing of step 1270 is performed.

The processing of step 1210 is illustrated in the flowchart of Figure 13. The process begins at step 1211, where the DNT, URT and PT of the receiving node are searched to determine whether the node ID (N_k) of the sending node is listed in any of those tables as being in the neighborhood of the receiving node. If so, the node is taken out of the corresponding table at step 1212. Then, or if the sending node was not in any of the neighborhood tables at step 1211, all nodes in the URT and PT for the receiving node are removed from those tables and added to the downstream neighborhood table DNT and the hop counts in the DNT are set to hop count of the sending node plus 2 at step 1213. The sending node is entered in the URT of the receiving node at step 1214, and the hop count for the receiving nose is set to the hop count of the sending node plus 1 at step 1215. Finally, the node ID and hop count of the receiving node are sent to other nodes in the receiving node's neighborhood in a new T5 trigger message at step 1216 and the process is complete. In other embodiments, the sending of the new T5 trigger message at step 1216 is delayed until the next periodic broadcast. It is also possible to not update the hop counts at step 1213 but rather update them upon receipt of a periodic broadcast message from the neighboring nodes, which will contain the updated hop count after the T5 trigger message is sent to the neighboring nodes at step 1216.

The processing of step 1220 is illustrated in the flowchart of Figure 14. The process begins at step 1221, where the DNT, URT and PT of the receiving node is searched to determine whether the node ID (N_k) of the sending node is listed in any of those tables as being in the neighborhood of the receiving node. If so, the node is taken out of the corresponding table at step 1222. Next, the PT, DNT and URT are updated at step 1223. The nodes previously listed in the peer table PT are added to the downstream neighbor table DNT, the nodes previously listed in the URT are moved to the PT, and the hop counts are appropriately modified. The sending node is entered in the URT of the receiving node at step 1224, and the hop count for the receiving nose is set to the hop count of the sending node plus 1 at step 1225. Finally, the node ID and hop count of the receiving node are sent to other nodes in the receiving node's neighborhood in a new T5 trigger message at step 1226 and the process is complete. The alternative embodiments and methods discussed above in connection with Figure 13 are applicable to the processing of Figure 14 as well.

The processing of step 1230 (the hop count of the receiving node is one greater than the hop count of the sending node) is illustrated in the flowchart of Figure 15. If the sending node is listed in the URT of the receiving node at step 1231, then processing is complete as there has been no change in the relative position of the sending and receiving nodes. If the sending node is not in the URT of the receiving node, then the DNT and PT of the receiving node are checked at step 1232 to determine if the sending node is listed in either of those tables. If so, the sending node is taken out of the corresponding table at step 1233. Next, or if the sending node was not in any of the neighborhood tables at step 1232, the sending node is added to the URT of the receiving node at step 1234 as the sending node has a lower hop count than the receiving node. The downstream cluster of the receiving node is computed at step 1235. Next, a check is made at step 1236 to determine if the sending node was previously listed in the downstream neighborhood table of the receiving node. If so, a T4 trigger message including the downstream cluster is sent to the sending node at step 1237 and processing is complete. If the sending node was not previously listed in the receiving node's downstream neighborhood table at step 1236, the T4 trigger message with the downstream cluster calculated at step 1235 is sent to the nodes in the receiving node's upstream routing table at step 1238 and processing is complete.

The processing of step 1250 is illustrated in the flowchart of Figure 16. If the sending node (whose hop count is equal to the receiving node's hop count) is already listed in the peer table of the receiving node at step 1251, then processing is complete as there has been no change in the relative position of the sending and receiving nodes and nothing more need be done. If the sending node is not in the PT of the receiving node at step 1251, then the DNT and URT of the receiving node are checked at step 1252 to determine if the sending node is listed in either of those tables. If so, the sending node is taken out of the corresponding table at step 1253. Next, or if the sending node was not in the peer table at step 1252, the sending node is added to the PT of the receiving node at step 1254. Next, if the URT is not empty at step 1255, processing is complete.

If the URT is empty at step 1255 (meaning that there is no upstream node and hence no way to reach the fixed gateway node), then the three neighborhood tables are re-computed at step 1256. First, the nodes listed in the PT are moved to the URT (i.e., since no upstream node is available, packets destined for the fixed gateway node will be routed through a peer). Then, nodes listed in the downstream neighborhood table are moved to the peer table, and the downstream neighborhood table and downstream routing table are left empty. If the URT is still empty at step 1257 (i.e., there were no peers in the PT), then no path to the fixed gateway node is available and the hop count for the receiving node is set to infinity at step 1258 and processing is complete. If, however, the URT was not empty at step 1257, the downstream cluster is calculated at step 1259 and sent to the upstream neighbors at step 1260 and processing is complete.

The T5 trigger discussed above will generally propagate downstream because it is initiated by a new RF association with a fixed gateway node. This downstream propagation will work even when nodes are isolated (i.e., have an infinite hop count) because the comparison between an infinite hop count with a finite hop count will select the processing of step 1210.

The processing of step 1270 (the hop count of the sending node is at least one greater than the hop count of the receiving node) is illustrated in the flowchart of Figure 17. If the sending node is already listed in the downstream neighbor table of the receiving node at step 1271, then the hop count of the sending node is set equal to the hop count of the receiving node plus one at step 1272 and a new T5 trigger message including the hop count of the receiving node is sent to the sending node at step 1273 and processing is complete. If the sending node is not in the DNT of the receiving node at step 1271, then the PT and URT of the receiving node are checked at step 1274 to determine if the sending node is listed in either of those tables. If so, the sending node is taken out of the corresponding table at step 1275. Next, or if the sending node was not in the peer table or URT at step 1274, the sending node is added to the downstream neighbor table of the receiving node at step 1276. The hop count of the sending node is set equal to the hop count of the receiving node plus one at step 1277. Next, if the URT is not empty at step 1278, the node identification and hop count of the receiving node are sent to the sending node at step 1273 and processing is complete. It should be noted that, in alternative embodiments, it is also possible to wait for the next periodic broadcast from neighboring nodes to update the hop counts rather than updating the hop counts at step 1256.

If the URT is empty at step 1278 (meaning that there is no upstream node and hence no way to reach the fixed gateway node), then the three neighborhood tables are re-computed at step 1279. First, the nodes listed in the PT are moved to the URT (i.e., since no upstream node is available, packets destined for the fixed gateway node will be routed through a peer). Then, nodes listed in the downstream neighborhood table are moved to the peer table, and the downstream neighborhood table and downstream network table are left empty. If the URT is still empty at step 1280 (i.e., there were no peers in the PT), then no path to the fixed gateway node is available and the hop count for the receiving node is set to infinity at step 1281 and step 1273 is performed. If, however, the URT was not empty at step 1280, the downstream cluster is calculated at step 1282 and sent to the upstream neighbors in a T4 trigger message at step 1283. Then, the node identification and hop count of the receiving node are sent to the sending node at step 1273 and processing is complete. It should be noted that the alternatives discussed above in connection with Figure 13 (i.e., not updating the hop counts at step 1279 and waiting until the next periodic broadcast to send the hop counts and node IDs rather than sending them at step 1273) are also applicable to this processing.

In addition to the triggers described above, a mechanism to remove obsolete links from the upstream routing table, peer table and downstream network table is necessary. This mechanism can take the form of periodic audits in which all of the nodes in the aforementioned tables are checked periodically. Alternatively, an activity watchdog for each neighbor node can be set up to trigger action if the neighbor node has been idle for too long.

The first mechanism involves periodically auditing the entries for the URT, PT and DNT. In the URT audit, the time since each node in the URT has been heard from is checked. If a node has been silent for more than a threshold period of time, the node is removed from the URT and the processing associated with a T3 trigger (steps 1010-1070 of Figure 10) is performed. If a peer node in the PT has been inactive for more than a threshold period of time, it is simply removed from the peer table. Finally, in the DRT audit, the time since each node in the DRT has been heard from is checked. If a downstream node in the DRT has been inactive for more than a threshold period of time, the processing associated with a T1 trigger (steps 810-840 of Figure 8) is performed.

The processing of the second mechanism, activity watchdogs, works in the same fashion described above in connection with the periodic audits whenever an activity watchdog indicates that a node has been idle for more than a threshold period of time.

Any given mobile node will be associated with a growing number of fixed gateway nodes over time as that mobile node moves around. As a practical matter, preferred embodiments maintain only a limited number of associations (e.g., 3) with fixed gateway nodes. In such embodiments, the maintained associations are referred to as principal associations. One way that this can be accomplished is by having a new association replace an old association and trigger its elimination in the neighborhood tables. If a principal association with a fixed gateway node becomes "broken" (i.e., communications with the fixed gateway node become impossible), the association becomes inactive but is still maintained (i.e., it is not dropped as in the case where a new association replaces an old one) as a peer node. It is important to continue maintaining inactive associations because they may become active again (e.g., a mobile unit makes a U-turn or goes around a curve in the road).

In the embodiments described above, a correction in a DRT is immediately propagated upward. Since URT corrections propagate downward, this will create successive waves of DRT update propagation upward as the URT corrections propagate downward. In this manner, infrastructure updates are propagated immediately, which results in fewer mishandled packets. However, this requires more CPU resources. Alternatively, the algorithms discussed herein can be modified so that any upstream propagation of DRT updates only occurs when a node that has modified its DRT has no downstream neighbors. This would delay infrastructure updates, but would be computationally more economical.

Obviously, numerous other modifications and variations of the present invention are possible in light of the above teachings. It is therefore to be understood that within the scope of the appended claims, the invention may be practiced otherwise than as specifically described herein.

WHAT IS CLAIMED IS:

1. A network comprising:

at least one fixed gateway node, the fixed gateway node being in communication with the Internet; and

a plurality of wireless nodes, each of the wireless nodes including wireless transceivers, each of the wireless nodes includes an upstream routing table and a downstream routing table, each of the wireless nodes being configured to use its upstream and downstream routing tables to make routing decisions, some of the wireless nodes being mobile wireless nodes, some of the mobile wireless nodes being configured to act as a relay for other wireless nodes that cannot directly access the at least one fixed gateway node;

wherein at least one first wireless node has an upstream routing table listing a second node and a corresponding hop count relative to the at least one fixed gateway node for the second node, the corresponding hop count for the second node being lower than a hop count for the first wireless node, and wherein at least one third wireless node has a downstream routing table listing at least one fourth wireless node with a corresponding hop count relative to the at least one fixed gateway node, the corresponding hop count for the fourth wireless node being higher than the hop count for the third wireless node.

2. The network of Claim 1, wherein the at least one fixed gateway node is directly connected to the Internet.

3. The network of Claim 1, wherein the at least one fixed gateway node includes a plurality of fixed gateway nodes.

4. The network of Claim 1, wherein the at least one fixed gateway node is connected to the Internet via a central node.

5. The network of Claim 4, wherein the central node is connected to at least one other fixed gateway node.

6. The network of Claim 1, wherein each of the wireless nodes is configured to act as a relay for other wireless nodes that cannot directly access the at least one fixed gateway node.

7. The network of Claim 1, wherein each wireless node further comprises a peer table that lists all directly accessible nodes having a same hop count to the at least one fixed gateway node as the wireless node.

8. The network of Claim 1, wherein all of the wireless nodes are mobile nodes.

9. The network of Claim 1, wherein some of the wireless nodes are mobile nodes and other wireless nodes are fixed nodes.

10. The network of Claim 1, wherein the downstream routing tables are indexed by destination node.

11. The network of Claim 1, wherein the downstream routing tables are indexed by downstream neighbor.

12. The network of Claim 1, wherein the downstream routing tables are indexed by both destination node and downstream neighbor.

13. A method for routing a packet received performed by a first wireless node in an ad-hoc network that includes at least one fixed gateway node comprising the steps of:

selecting a node to which to transmit the packet from an upstream routing table if the direction of the packet is upstream, the upstream routing table including a list of all directly accessible nodes having a hop count less than the first wireless node relative to the fixed gateway node;

selecting a downstream node to which to transmit the packet from a downstream routing table if the direction of the packet is downstream, the downstream routing table including a list of all directly connected nodes and all downstream nodes having at least one shortest path that passes through the wireless node; and

transmitting the packet to the selected downstream node.

14. The method of Claim 13, further comprising the step of receiving the packet from another wireless node prior to performing either of the selecting steps.

15. The method of Claim 14, wherein the packet is received in either an upstream buffer or a downstream buffer, and either upstream routing or downstream routing are performed depending on whether the packet is received in the upstream buffer or the downstream buffer.

16. The method of Claim 13, in which the packet includes a flag indicating whether the packet is to be routed upstream or downstream.

17. The method of Claim 13, further including the step of determining a direction of the packet based on a destination included in the packet.

18. The method of Claim 14, further comprising the steps of transmitting a trigger message including the downstream cluster if a destination node associated with the packet is not listed in the downstream routing table.

19. The method of Claim 13, further comprising the steps of updating the downstream routing table by removing the selected node if the packet could not be transmitted to the selected node, calculating the downstream cluster based on the updated routing table, and transmitting a trigger message including the downstream cluster to nodes listed in the upstream routing table.

20. The method of Claim 13, further comprising the steps of removing the selected node from the upstream routing table if the direction of the packet is upstream and the packet could not be transmitted to the selected node.

21. The method of Claim 20, further comprising the steps of, if the packet could not be transmitted to the selected node, moving the nodes in a peer table of the wireless node to the upstream routing table, calculating the downstream cluster, and transmitting the downstream cluster to nodes listed in the upstream routing table.

22. A method for updating a downstream routing table at a wireless node in an ad-hoc network including a least one fixed gateway node, the method comprising the steps of:

receiving a message from a first node, the first node being downstream of the wireless node, the message including a downstream routing table for the first node;

updating the downstream routing table associated with the wireless node based on information in the downstream routing table of the first node; and transmitting the downstream routing table associated with the wireless node to a node listed in an upstream routing table associated with the wireless node.

23. The method of Claim 22, wherein the downstream routing tables are indexed by destination node.

24. The method of Claim 22, wherein the downstream routing tables are indexed by downstream neighbor.

25. The method of Claim 22, wherein the downstream routing tables are indexed by both destination node and downstream neighbor.

26. A method comprising the steps of:

receiving a periodically broadcast message sent by an other node in an ad-hoc network at a wireless node, the network including at least one fixed gateway node, the message including a node identification associated with the other node and a hop count associated with the other node, the hop count representing a number of hops in a shortest path to the fixed gateway node; and

updating upstream routing tables, downstream routing tables and peer tables as necessary depending on the node identification and hop count received in the periodically broadcast message.

27. The method of Claim 26, wherein, if the hop count in the message indicates that the other node is at least 3 levels closer to the fixed gateway node than the wireless node, then all nodes in the peer routing table and the upstream routing table are moved to the downstream routing table and the hop counts of such nodes are modified accordingly, the other node is entered into the upstream routing table, the hop count of the wireless node is set to one greater than the hop count of

the other node, and a message including the hop count and node identification of the wireless node is broadcast.

28. The method of Claim 26, wherein, if the hop count in the message indicates that the other node is two hops closer to the fixed gateway node than the wireless node, then the nodes listed in the peer table are moved to the downstream routing table with hop counts modified accordingly, the nodes listed in the upstream routing table are moved to the peer table with hop counts modified accordingly, the other node is entered into the upstream routing table, the hop count of the wireless node is set to one greater than the hop count of the other node, and a message including the hop count and node identification of the wireless node is broadcast.

29. The method of Claim 26, wherein, if the hop count in the message indicates that the other node is one hop closer to the fixed gateway node than the wireless node, then, if the other node is not present in the upstream routing table, the other node is removed from the downstream routing table and peer routing table if it is present in either of those tables, the other node is entered into the upstream routing table, and the downstream routing table is updated and sent to at least one upstream neighbor.

30. The method of Claim 26, wherein, if the hop count in the message is the same as the hop count of the wireless node, then, if the other node is not already in the peer table, it is removed from any other table in which it appears and entered into the peer table, and, if the removal of the other node from the upstream routing table results in an empty upstream routing table, the other node is entered

into the upstream routing table and the downstream routing tables are calculated and sent to any upstream nodes.

31. The method of Claim 26, wherein, if the hop count in the message indicates that the other node is downstream of the wireless node, the hop count and node identification of the wireless node are sent to the other node, the other node is entered into the downstream neighborhood table if not already present in that table, the hop count of the other node in the downstream neighborhood table is set equal to one greater than the hop count of the wireless node, the other node is removed from the upstream routing table or the peer table if it appears in either of those tables, and, if the upstream routing table has become empty as a result, the nodes listed in the peer table are moved to the upstream routing table, and the downstream routing table update is calculated and sent to any upstream neighbors.

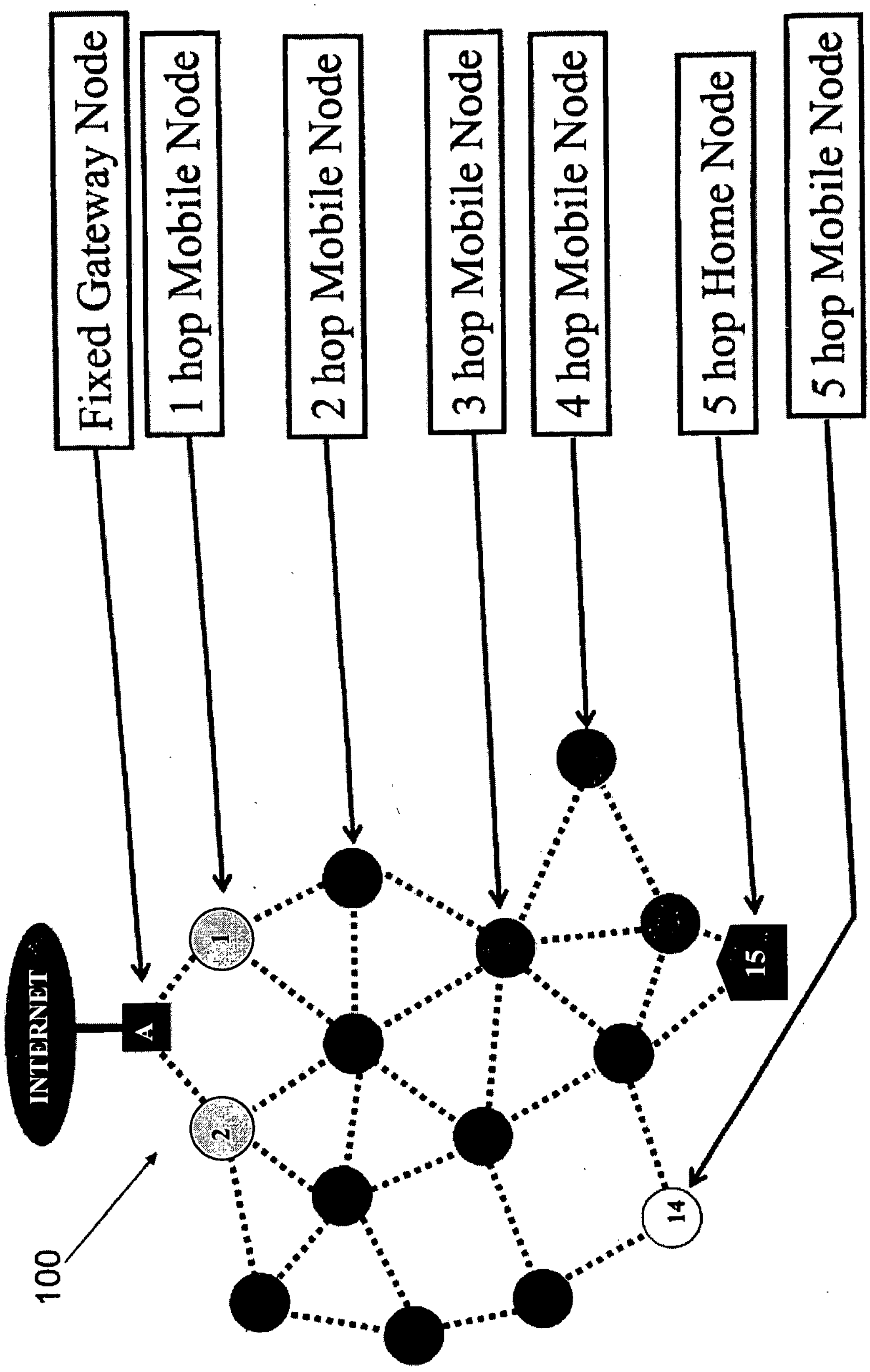

Fixed Gateway Node
1 hop Mobile Node
2 hop Mobile Node
3 hop Mobile Node
4 hop Mobile Node
5 hop Home Node
5 hop Mobile Node
INTERNET
A
1
2
14
15
100

Figure 1

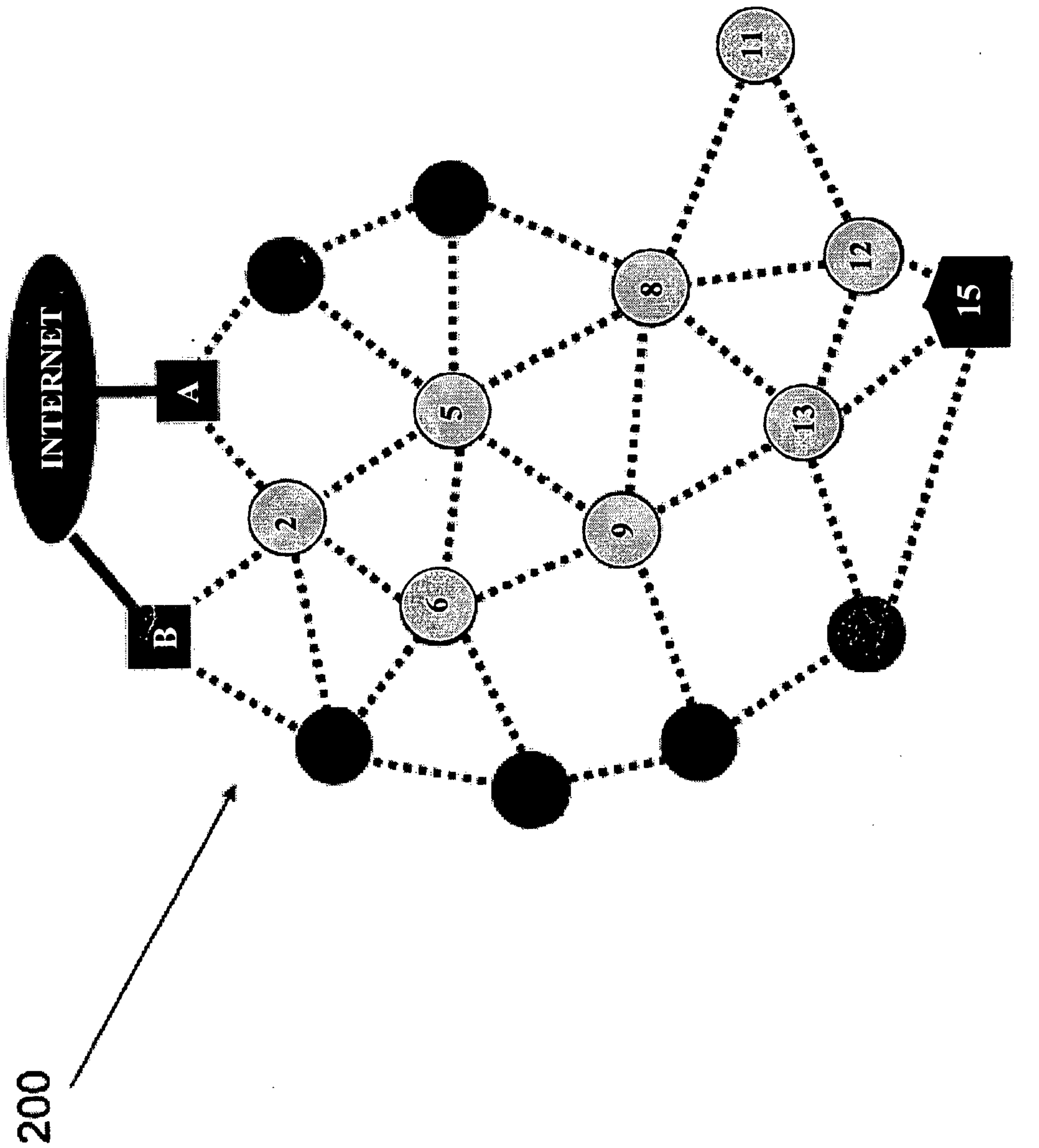


Figure 2

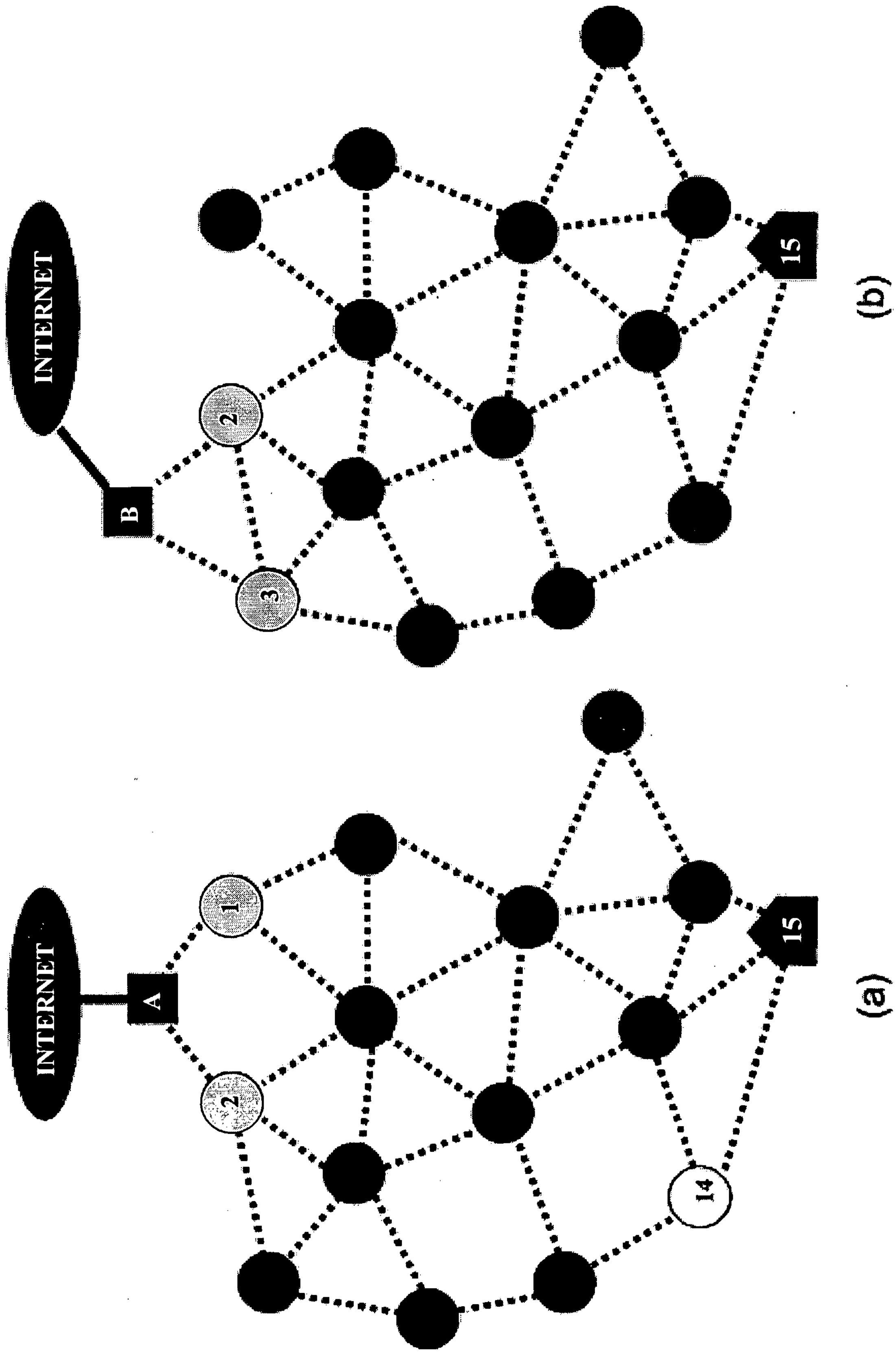


Figure 3

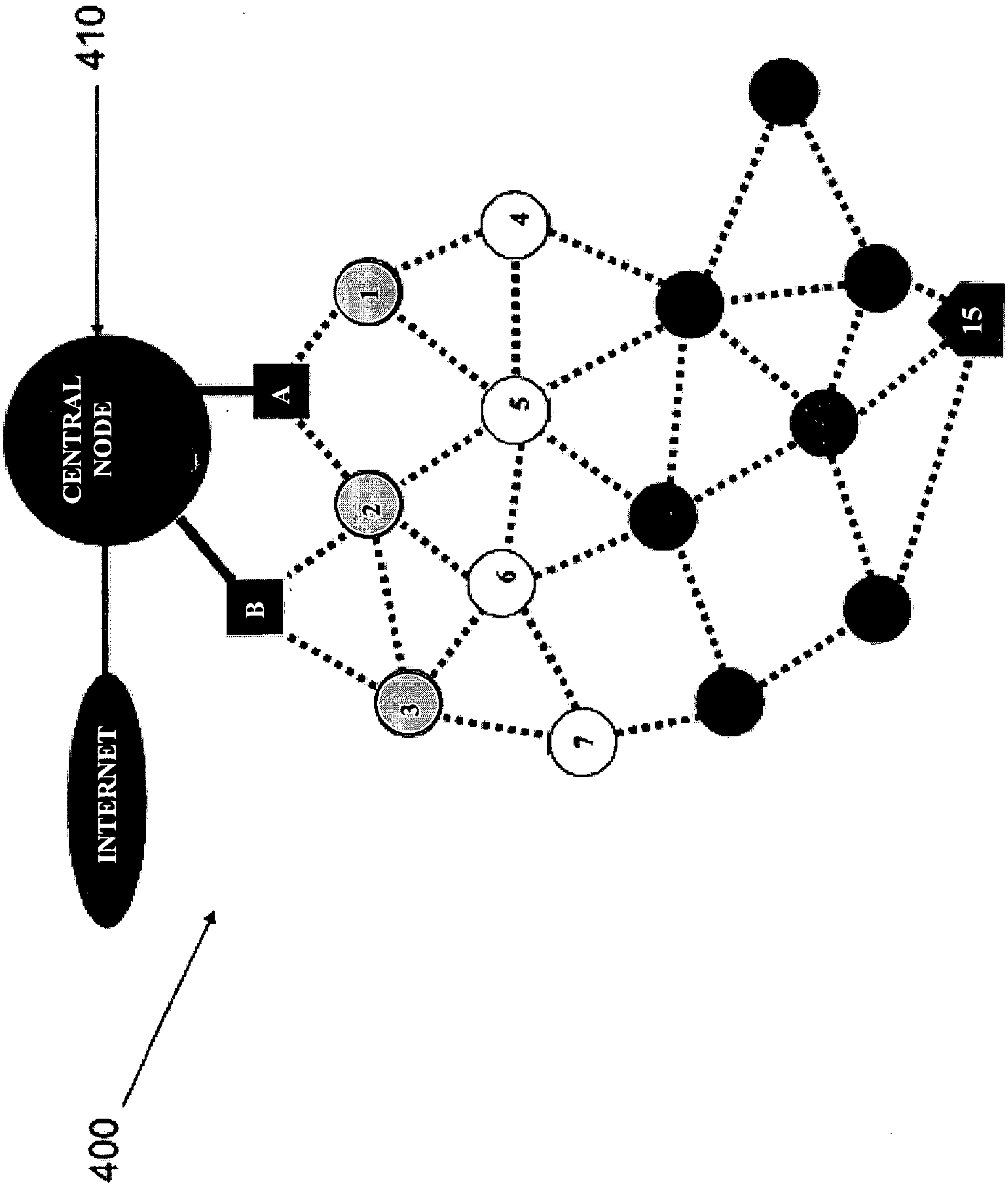


Figure 4

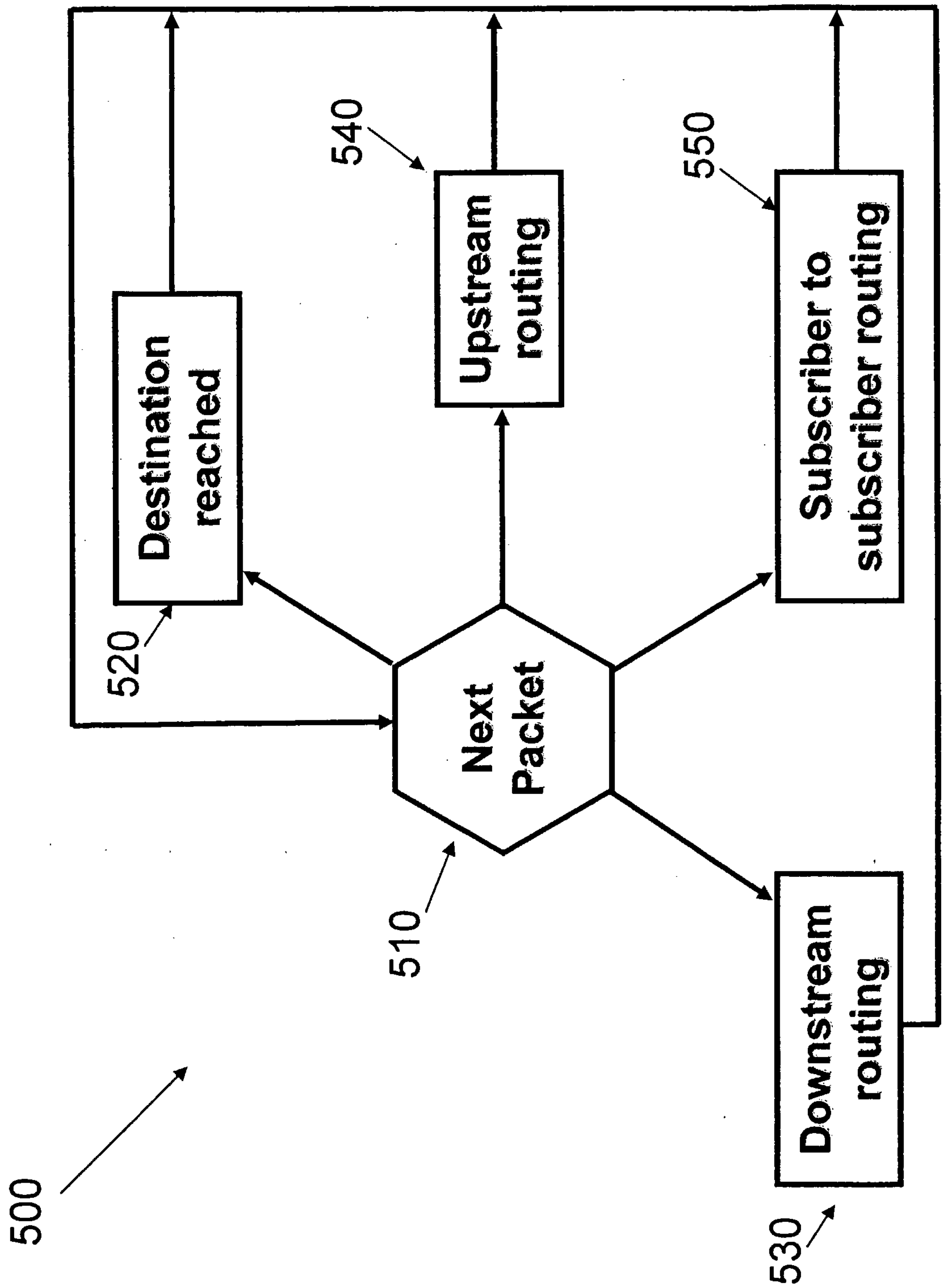

500
Next Packet
510
Destination reached
520
Upstream routing
540
Subscriber to subscriber routing
550
Downstream routing
530

Figure 5

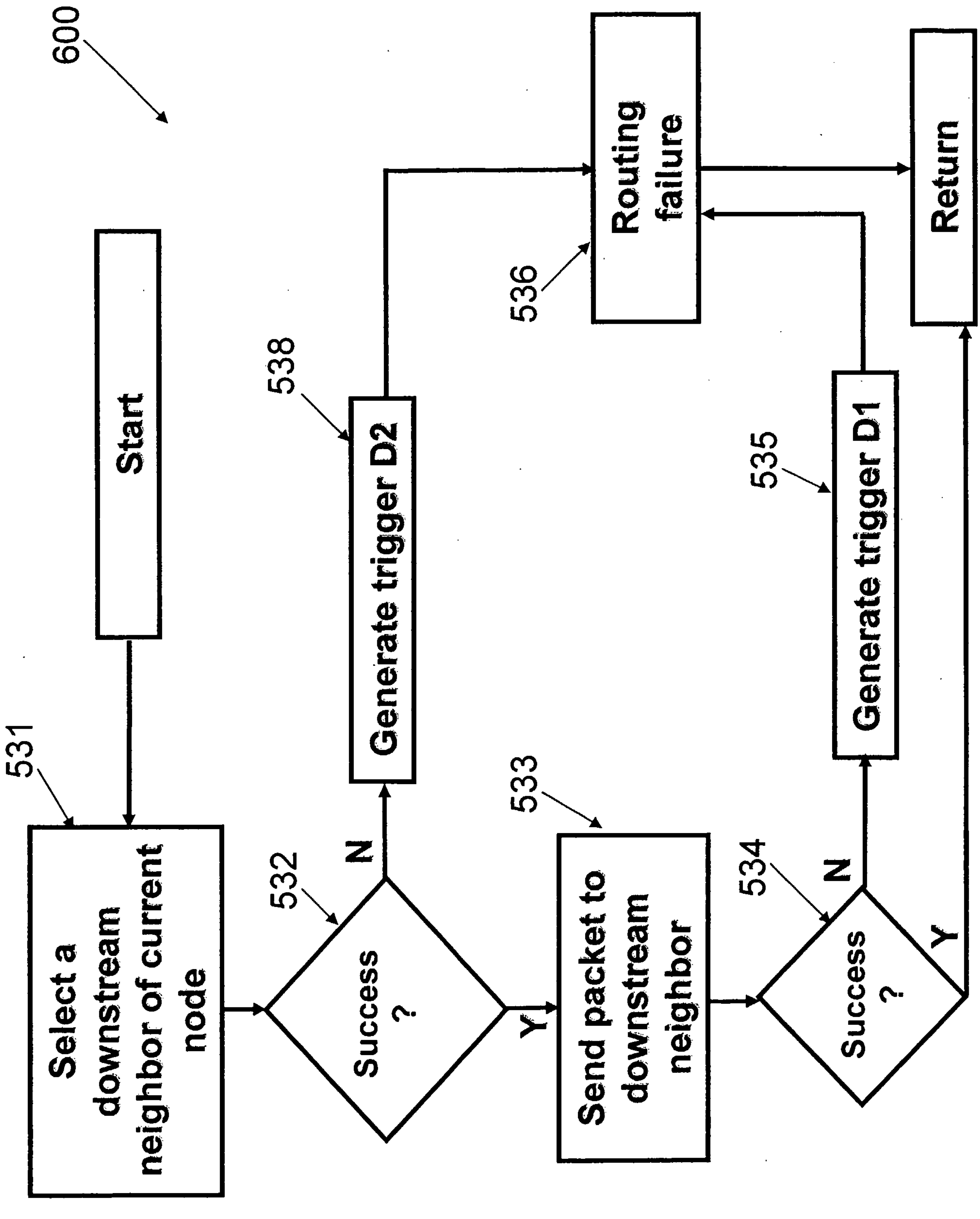

600
Start
531
Select a downstream neighbor of current node
532
Success ?
N
Y
538
Generate trigger D2
533
Send packet to downstream neighbor
534
Success ?
N
Y
535
Generate trigger D1
536
Routing failure
Return

Figure 6

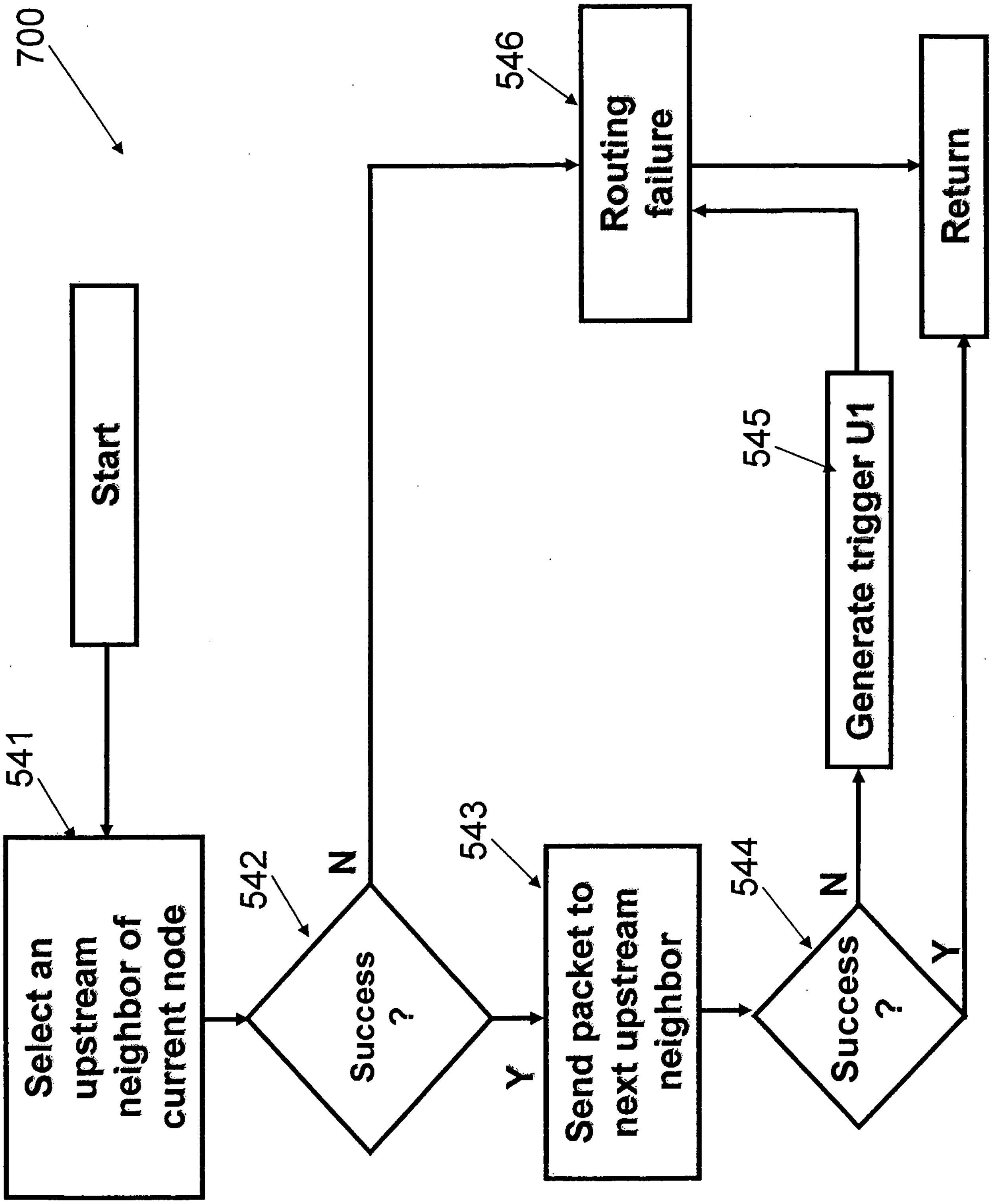

700
Start
541
Select an upstream neighbor of current node
542
Success ?
N
Y
543
Send packet to next upstream neighbor
544
Success ?
N
Y
545
Generate trigger U1
546
Routing failure
Return

Figure 7

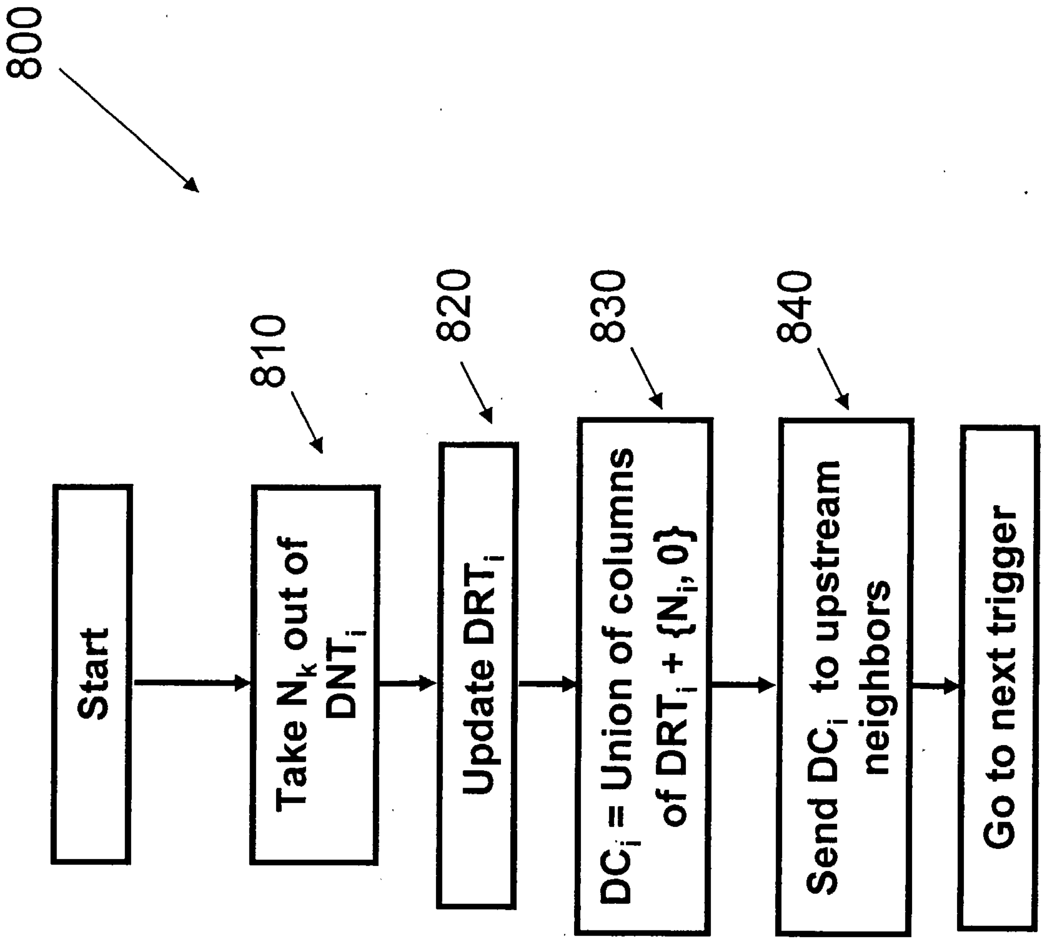

800
Start
810
Take Nk out of DNTi
820
Update DRTi
830
DCi = Union of columns of DRTi + {Nj, 0}
840
Send DCi to upstream neighbors
Go to next trigger

Figure 8

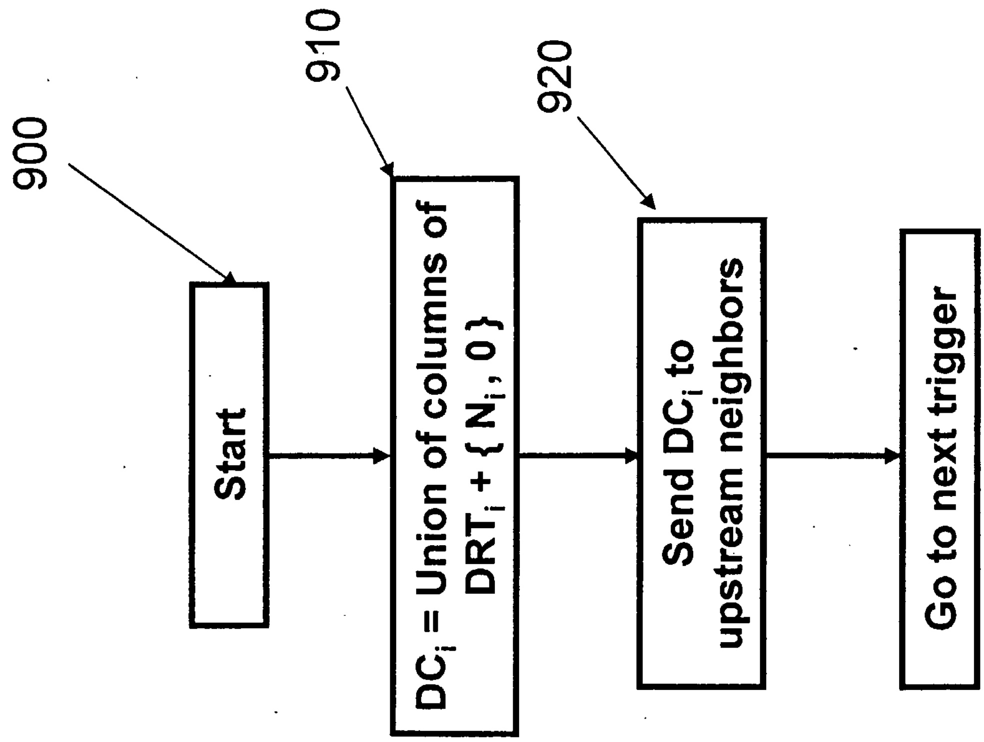

900
910
920
Start
DC_i = Union of columns of DRT_i + { N_i , 0 }
Send DC_i to upstream neighbors
Go to next trigger

Figure 9

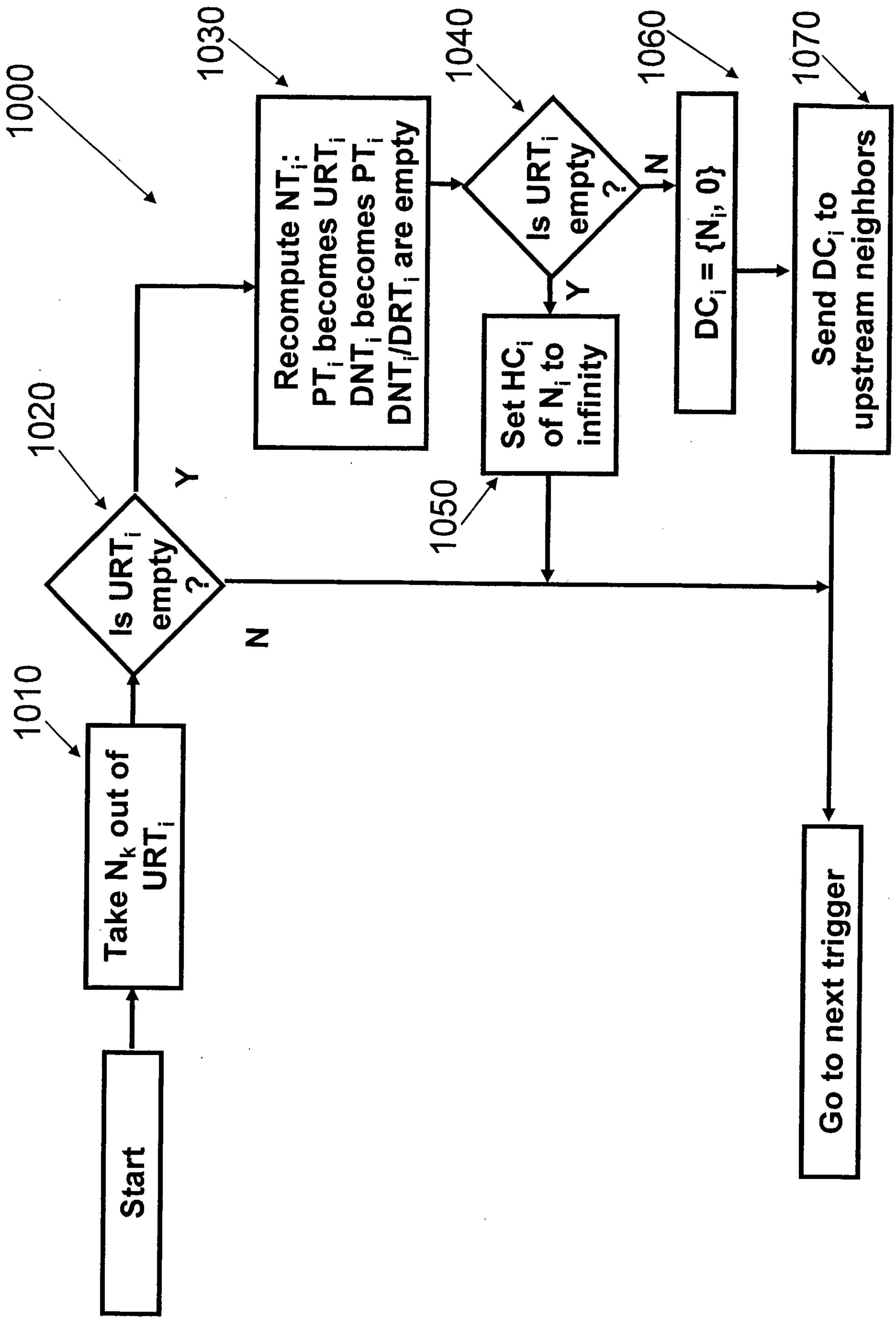

1000
Start
1010
Take N_k out of URT_i
1020
Is URT_i empty ?
Y
N
1030
Recompute NT_i:
PT_i becomes URT_i
DNT_i becomes PT_i
DNT_i/DRT_i are empty
1040
Is URT_i empty ?
Y
N
1050
Set HC_i of N_i to infinity
1060
DC_i = {N_i, 0}
1070
Send DC_i to upstream neighbors
Go to next trigger

Figure 10

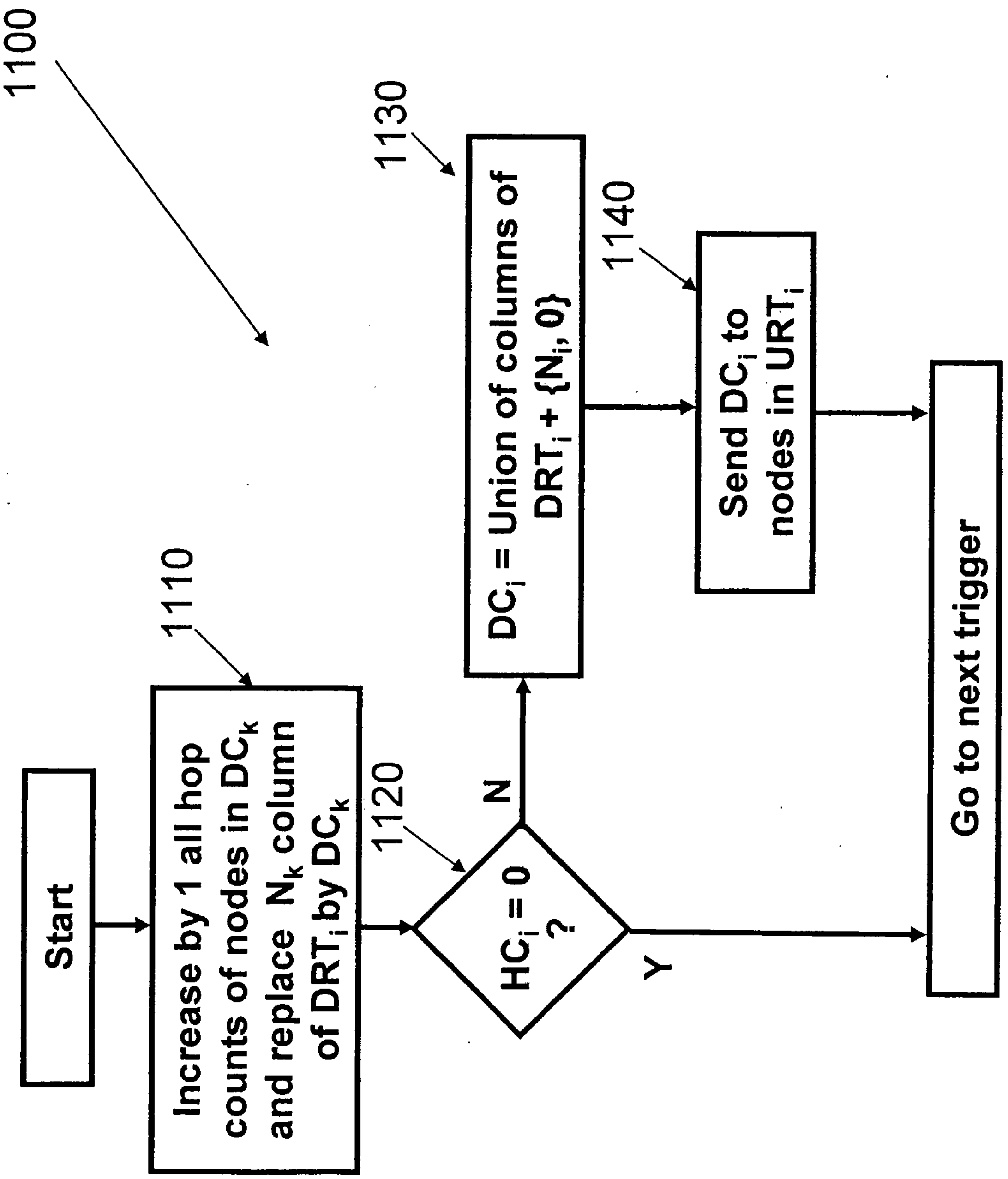

1100
Start
1110
Increase by 1 all hop counts of nodes in DC_k and replace N_k column of DRT_i by DC_k
1120
HC_i = 0 ?
N
Y
1130
DC_i = Union of columns of DRT_i + {N_i, 0}
1140
Send DC_i to nodes in URT_i
Go to next trigger

Figure 11

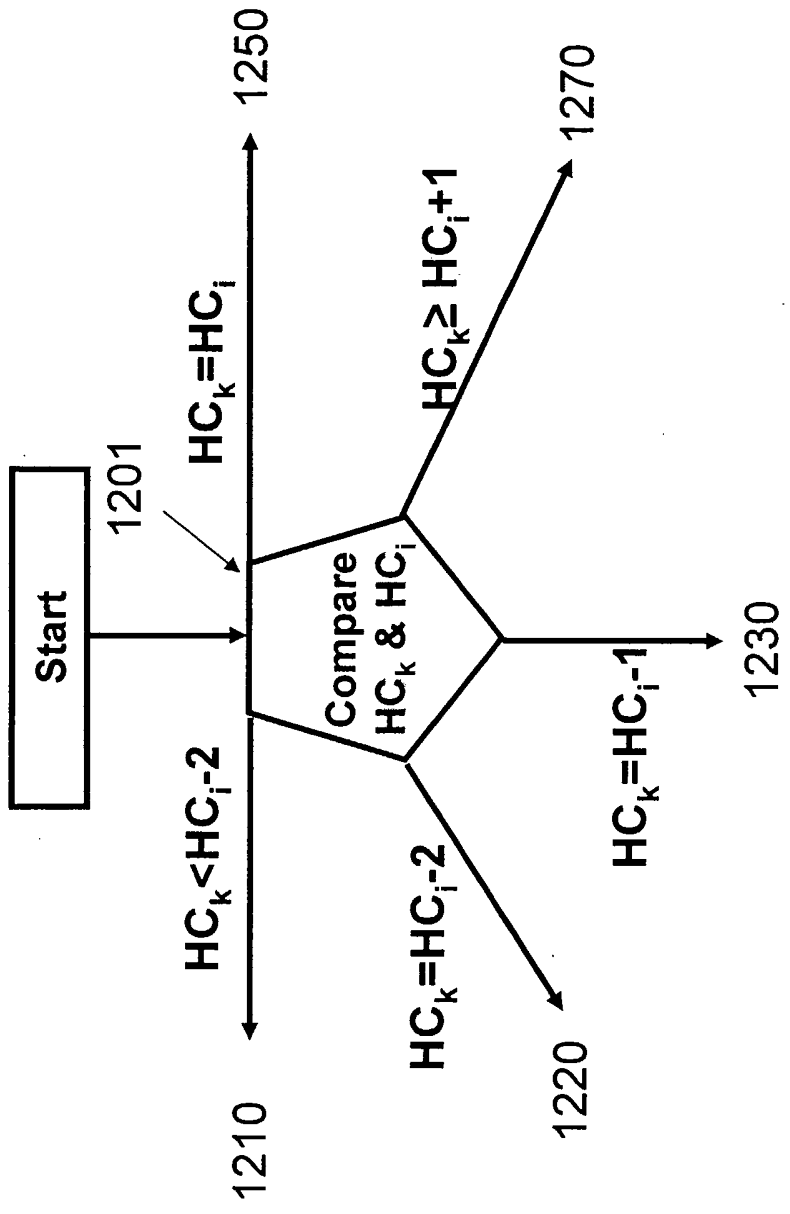

Start
1201
Compare
HC_k & HC_i
$HC_k=HC_i$
1250
$HC_k \geq HC_i+1$
1270
$HC_k=HC_i-1$
1230
$HC_k=HC_i-2$
1220
$HC_k<HC_i-2$
1210

Figure 12

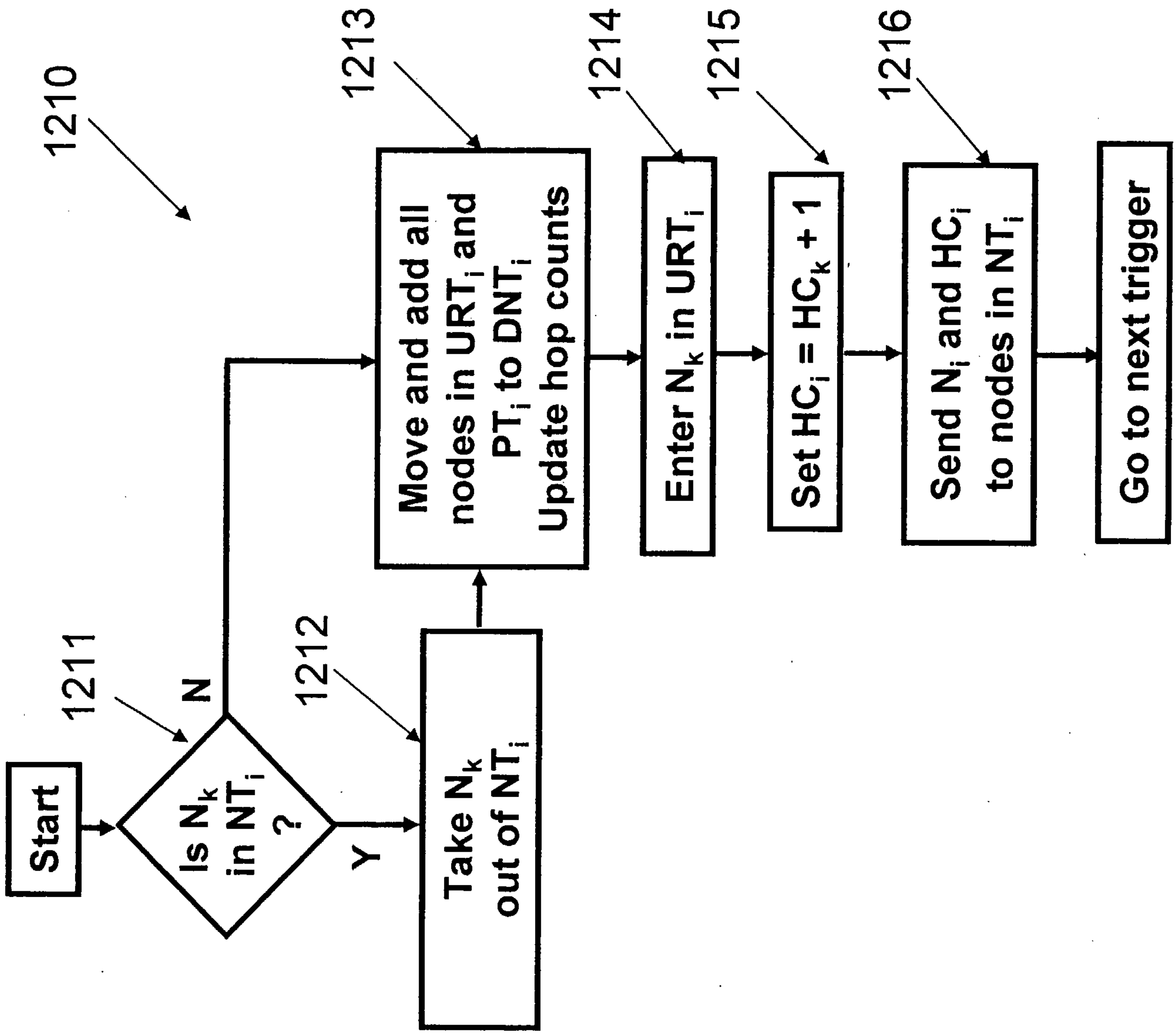

1210
Start
Is N_k in NT_i ?
1211
N
Y
1212
Take N_k out of NT_i
1213
Move and add all nodes in URT_i and PT_i to DNT_i Update hop counts
1214
Enter N_k in URT_i
1215
Set $HC_i = HC_k + 1$
1216
Send N_i and HC_i to nodes in NT_i
Go to next trigger

Figure 13

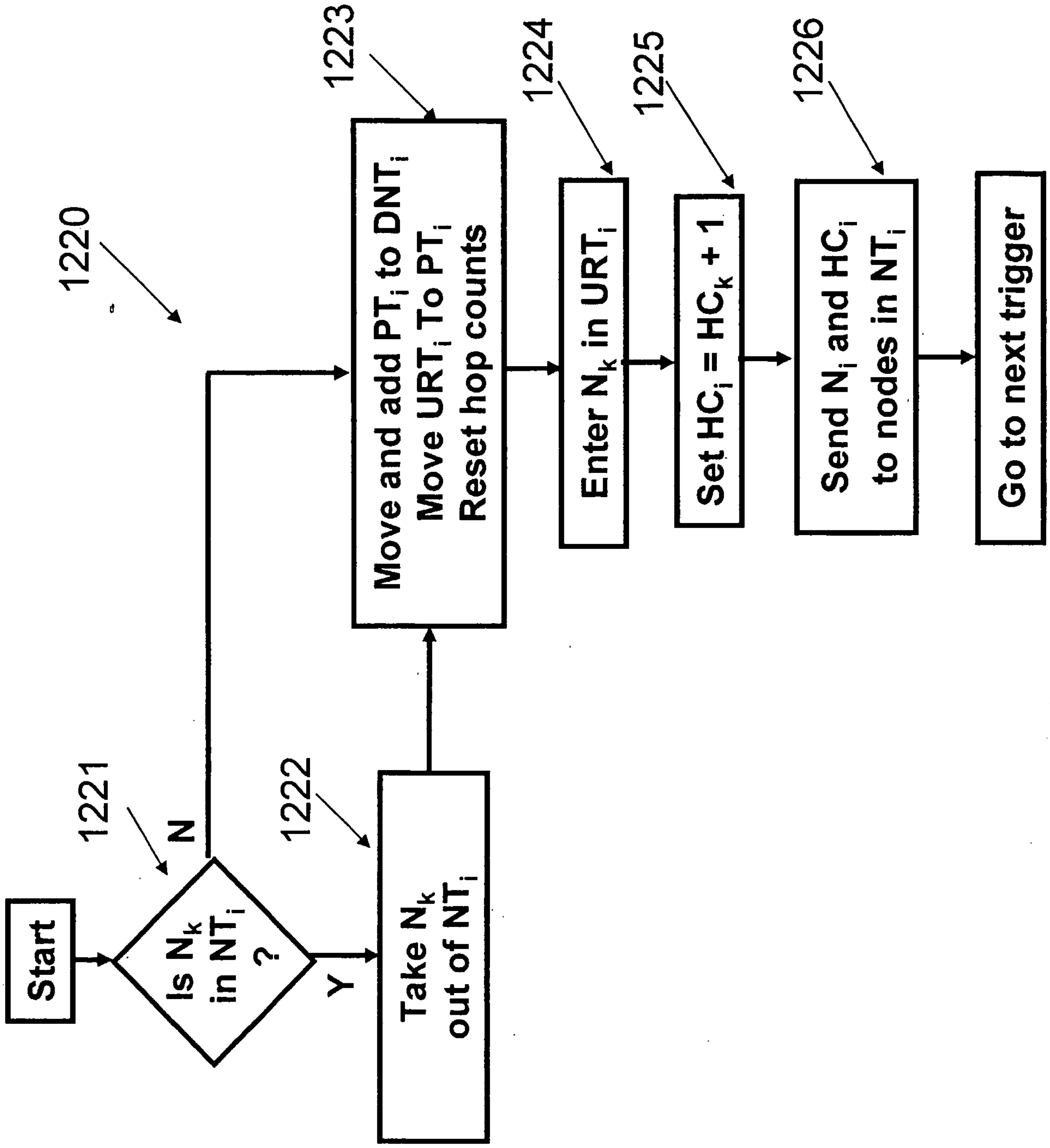

1220
Start
1221
Is N_k in NT_i ?
N
Y
1222
Take N_k out of NT_i
1223
Move and add PT_i to DNT_i
Move URT_i To PT_i
Reset hop counts
1224
Enter N_k in URT_i
1225
Set $HC_i = HC_k + 1$
1226
Send N_i and HC_i to nodes in NT_i
Go to next trigger

Figure 14

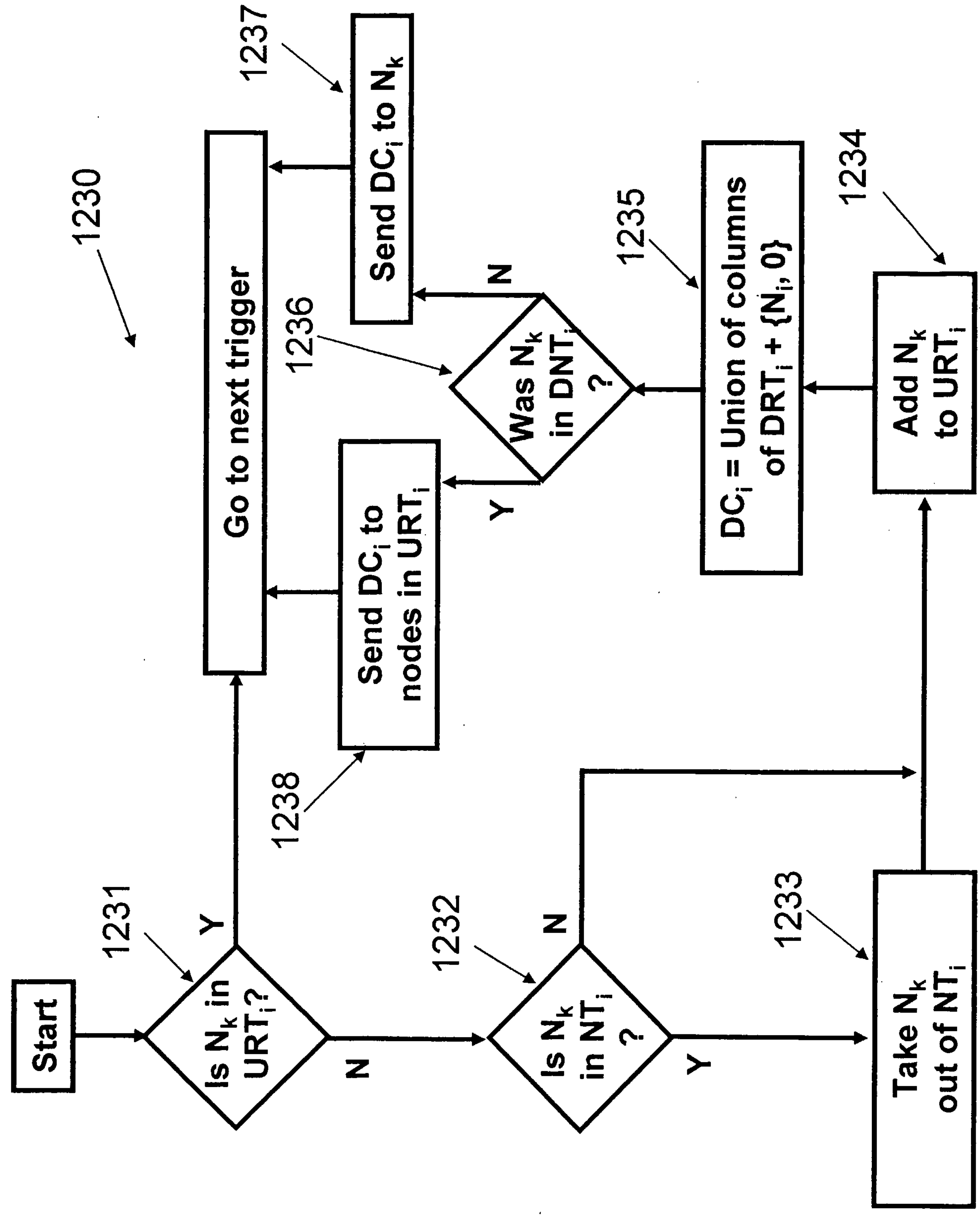

1230
Start
Is N_k in URT_i?
1231
Y
N
Is N_k in NT_i ?
1232
N
Y
Take N_k out of NT_i
1233
Add N_k to URT_i
1234
DC_i = Union of columns of DRT_i + {N_i, 0}
1235
Was N_k in DNT_i ?
1236
N
Y
Send DC_i to N_k
1237
Send DC_i to nodes in URT_i
1238
Go to next trigger

Figure 15

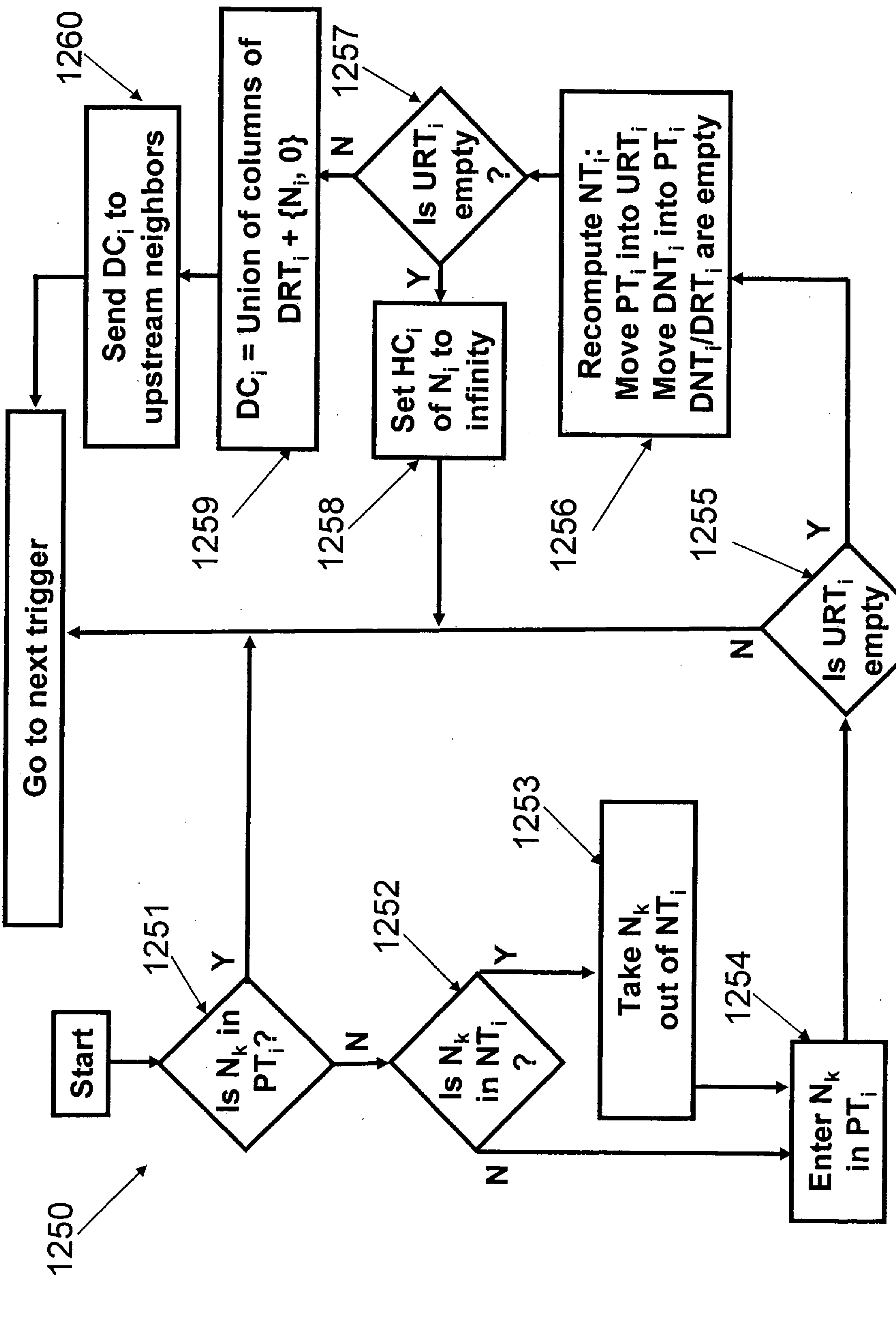

1250
Start
1251
Is N_k in PT_i?
Y
N
1252
Is N_k in NT_i ?
Y
N
1253
Take N_k out of NT_i
1254
Enter N_k in PT_i
1255
Is URT_i empty ?
Y
N
1256
Recompute NT_i:
Move PT_i into URT_i
Move DNT_i into PT_i
DNT_i/DRT_i are empty
1257
Is URT_i empty ?
Y
N
1258
Set HC_i of N_i to infinity
1259
DC_i = Union of columns of DRT_i + {N_i, 0}
1260
Send DC_i to upstream neighbors
Go to next trigger

Figure 16

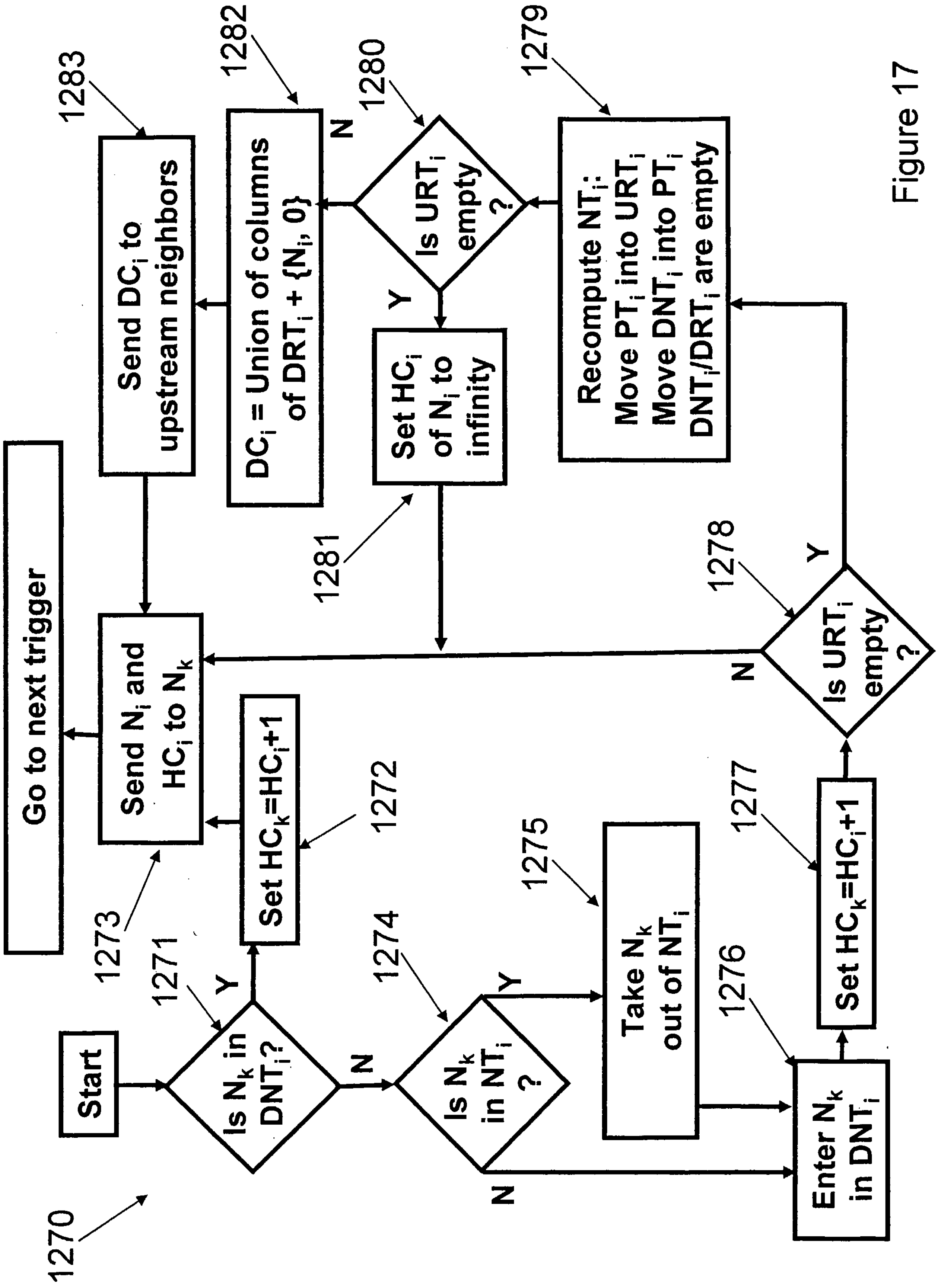

1270
Start
Is N_k in DNT_i?
1271
Y
N
Set HC_k=HC_i+1
1272
Send N_i and HC_i to N_k
1273
Go to next trigger
Is N_k in NT_i ?
1274
Y
N
Take N_k out of NT_i
1275
Enter N_k in DNT_i
1276
Set HC_k=HC_i+1
1277
Is URT_i empty ?
1278
Y
N
Recompute NT_i:
Move PT_i into URT_i
Move DNT_i into PT_i
DNT_i/DRT_i are empty
1279
Is URT_i empty ?
1280
Y
N
Set HC_i of N_i to infinity
1281
DC_i = Union of columns of DRT_i + {N_i, 0}
1282
Send DC_i to upstream neighbors
1283

Figure 17

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2005/045756

A. CLASSIFICATION OF SUBJECT MATTER
H04L12/56

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)
H04L

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 practical, search terms used)

EPO-Internal, WPI Data, PAJ, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2002/012320 A1 (OGIER RICHARD G ET AL) 31 January 2002 (2002-01-31) paragraphs [0045], [0046], [0064], [0065], [0067], [0068], [0070], [0071], [0086] - [0091]	1-14,17, 20,22-26
X	EP 1 480 387 A (EATON CORPORATION) 24 November 2004 (2004-11-24) paragraphs [0020] - [0022]	1-6,8,9, 13,14,17
A	US 2002/173321 A1 (MARSDEN IAN A ET AL) 21 November 2002 (2002-11-21) paragraphs [0005] - [0008], [0018]	1-31

-/--

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"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)

"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

"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

"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

"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.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 March 2006	04/04/2006

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Weinmiller, J

2

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	PERKINS C E ET AL: "HIGHLY DYNAMIC DESTINATION-SEQUENCED DISTANCE-VECTOR ROUTING (DSDV)FOR MOBILE COMPUTERS" COMPUTER COMMUNICATION REVIEW, ACM, NEW YORK, NY, US, vol. 24, no. 4, 1 October 1994 (1994-10-01), pages 234-244, XP000477054 ISSN: 0146-4833 paragraphs [002.], [003.]	1-31

INTERNATIONAL SEARCH REPORT

Information on patent family members

In tional application No
PCT/US2005/045756

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2002012320 A1	31-01-2002	AU 4748801 A	24-09-2001
		WO 0169862 A2	20-09-2001
EP 1480387 A	24-11-2004	CA 2467387 A1	19-11-2004
		MX PA04004719 A	06-09-2004
		US 2004233855 A1	25-11-2004
US 2002173321 A1	21-11-2002	CN 1463522 A	24-12-2003
		EP 1393507 A2	03-03-2004
		WO 02093844 A2	21-11-2002
		JP 2004525587 T	19-08-2004