

30 June 2004

Description

Method for forming routing information in an ATM communication network

5

If a source switching node S in an ATM communication network which operates according to the principles of the PNNI protocol of the ATM Forum (ATM Forum Technical Committee Private Network-Node-Interface (PNNI) Specification, Version 1.0) receives a connection set-up message from a terminal which is connected to it, an item of routing information must be determined in the said terminal for the entire route through the network as far as the destination switching node D to which the desired destination terminal is connected or at which a junction to another network is to be made and the said item of routing information is to be added, in a form in accordance with the PNNI protocol, to the connection set-up message before being passed on to following switching node.

The said ATM communication networks can be structured into numerous subnetworks ("peer groups") comprising physical switching nodes and physical connecting lines ("physical links"). In accordance with PNNI protocol, the nodes of a (hierarchically lowest) peer group from its centre determine a so-called representative node ("peer group leader") which represents the entire peer group in the form of a single, logical, exemplary node (referred to as "logical group node" or even "parent node") in a hierarchically higher peer group. A hierarchically higher peer group is formed from a plurality of such parent nodes and the connecting lines which network them, one connecting line ("logical link") representing a logically formed subset of all those physical connecting lines which connect every two boundary nodes of the two adjacent (hierarchically lower "child") peer groups to one another and have been provided with an

30 June 2004

identical identifier (referred to as "aggregation token") by administration.

The hierarchy continues recursively in further hierarchy levels: a peer group leader selection can also be made once more in the hierarchically higher peer group. The peer group leader selected here represents again, in the hierarchically next higher peer group, the entire hierarchy area which has been set up under it, as if this hierarchy area were a single node. In this peer group there are again logical, exemplary connecting lines between every two adjacent nodes, such a connecting line representing in turn a logically formed subset of all those physical connecting lines which are bounded by in each case one physical switching node in the adjacent hierarchy areas.

The hierarchical exemplary network (for illustration: 3-dimension grid) in accordance with the PNNI protocol is made complete by adding further, purely logical connecting lines, the so-called "uplinks", which, in accordance with the PNNI protocol, connect every two nodes (physically - if the node at the lower end of the uplink is a physical node - or logically) from peer groups of different hierarchical levels to one another.

For example, an uplink (also referred to as "initial uplink") leads from the boundary node of a hierarchically lowest peer group which is connected via a physical connecting line to a boundary node in an adjacent peer group, to a representative node, the so-called "upnode", i.e. to that representative node "ancestor node" (i.e. parent node or "grandparent node" or grand...grandparent node) of the adjacent boundary node which is a directly adjacent node to precisely one specific ancestor node of the boundary node on this side in a common hierarchically higher peer group. Such an (initial) uplink results in all the ancestor nodes (of the boundary node on this side) which however each belong to a hierarchically lower peer group as the

30 June 2004

said common hierarchically higher peer group each contribute to the hierarchy structure one uplink (also referred to as "induced uplink") to the aforesaid upnode.

5 The hierarchical structure which is ultimately based on corresponding configuration data of the individual nodes can be handled very flexibly here. In particular, the individual nodes of a grand... grandparent peer group can have different numbers of
10 subhierarchy levels together with respective peer groups.

 The exchange, in accordance with PNNI protocol, of data packets, "Hello packets" and PNNI topology state data packets (PTSPs) via so-called routing
15 control channels ensures that every physical switching node of a hierarchically lowest peer group acquires the same knowledge with regard to the topology of the said group and of all the hierarchically higher peer groups above it in the hierarchy, including all the uplinks,
20 and in addition acquires the same knowledge with regard to the occupancy rate of all the nodes and connecting lines contained therein as well as the same knowledge with regard to their properties (accessibility, capabilities, features, costs).

25 The topology knowledge which is acquired can be stored in the form of a graph G1 in a node. It would be assumed that the respective present switching node (the one which has created this graph G1 for itself) is marked in particular as the source node S.

30 If a terminal connected to the source node expresses the wish to be connected to the terminal of a specific destination address, the data exchanged by PNNI routing protocol make it possible to determine in the graph G1 that destination node D which specified
35 the accessibility of the destination terminal and at the same time belongs to the hierarchically lowest possible peer group.

 The ATM Forum Technical Committee Private Network-Node-Interface (PNNI) in the specification,

30 June 2004

Version 1.0, Annex H, does not, however, provide the possibility here of also including in the routing search advantageous alternative routes via one or more peer groups with return to the peer group which has
5 already been passed through, and as a result it is sometimes not possible to fulfil a switching request appropriately.

The object of the method according to the invention is to determine an optimum connecting path
10 taking into account the existing occupancy rate information relating to the all the nodes and connecting lines of this graph G1 and taking into account their properties as well as the expressed requirements of the present wish for a connection, and
15 of converting the information which is determined into an item of routing information in such a way that the said information satisfies the PNNI protocol specifications and can therefore be provided along with the connection set-up message on the path to the next
20 connection node. The object is achieved by means of the method on the basis of the preamble of Patent Claim 1 by means of its distinguishing features.

A graph G2 is determined which is obtained from the graph G1 in that all those nodes and connecting
25 lines which do not comply with the aforesaid conditions are removed from the graph G1.

Then, an optimum connecting path is determined in that alternative routes via hierarchically higher peer groups with subsequent return to hierarchically
30 lower peer groups which have already been passed through are also taken into account. In this context, the exit nodes and re-entry nodes for one and the same peer group are different, otherwise such an alternative route would constitute a highly superfluous loop and
35 would not constitute an optimum connecting path.

Even if everything which follows is concentrated on alternative-type routes, it is imperative not to overlook the fact that, even for a normal case in which an optimum route without

30 June 2004

alternatives is available, the method according to the invention also finds this route and equally correctly forms the corresponding routing information for it.

The resulting optimum route can in principle contain any desired number of junctions from a hierarchically higher to a hierarchically lower peer group, and conversely from a hierarchically lower to a hierarchically higher peer group, in which case in principle any desired number, i.e. zero, one, two, ... or $n \leq 102$ hierarchy levels may be skipped at each individual junction.

In accordance with the PNNI protocol, the routing information is also provided, as a sequence of information elements, so-called "designated-transit-list-information elements (DTLs)", along with the connection set-up message a preceding information element (repeat indicator) indicating the stack-type handling of these DTLs (push and pop operations). In this context, each DTL information element contains the description of precisely one route through precisely one hierarchical peer group in the form of one or more node link pair indications and a pointer which refers to one of these node link pairs. The route described by the uppermost DTL information element of the stack begins here at the source node S and contains only indications relating to nodes and connecting lines in the hierarchically lowest peer group and ends, if appropriate, with the indication of an uplink which leads to an upnode at which the route continues, specifically in the way described in the next lower stacked DTL. Each next lower DTL located in the stack contains indications for in each case one route through the hierarchically next higher peer group, in which case they begin with the indication of the respective ancestor node of the source node, possibly followed by further node and connecting line indications from the same peer group and a possible uplink indication as a termination. The lowest DTL in the stack contains indications relating to a route through the

hierarchically highest required peer group, beginning with the indication of the respective ancestor node of the source node and ending at a node in whose hierarchy area the destination node with the connected
5 destination terminal is located.

The described design, in accordance with the PNNI protocol, of the DTL stack initially gives the appearance that routes which consist of any desired sequences of hierarchically higher and hierarchically
10 lower nodes may not be taken into account, and that it is indicated that the algorithm for searching for an optimum connecting path should be configured in such a way that routes with such sequences (that is to say with alternative routes via hierarchically higher peer
15 groups) are eliminated from the outset, as is the case in the specifications of the PNNI protocol, Version 1.0, Annex H.

However, the method according to the invention solves the problem of representing such a route which
20 is associated with alternative routes and also forms for the route routing information which satisfies the specifications in accordance with the PNNI protocol.

It is characteristic of the solution according to the invention that an equivalent sequence of nodes
25 and connecting lines is derived from a prescribed sequence of hierarchically higher and hierarchically lower nodes in which uplinks from it would also have to be passed through in the downwards direction, which equivalent sequence never runs in a descending manner
30 with respect to the hierarchy levels of these nodes, that is to say in which uplinks never have to be passed through in the downwards direction. By virtue of what is achieved by this, one and the same hierarchically higher (logical) node can occur repeatedly in the
35 sequence (loops) but, owing to the connecting lines indicated, it is unambiguously ensured that the exit and re-entry boundary nodes in the respective child peer groups are always different, which ultimately

30 June 2004

means that one and the same physical node is never passed through more than once.

The determination of the best route in a switching node which determines the route and the routing information is explained below:

A graph G3 is derived from the abovementioned graph G2, in that all the ancestor nodes of the source node S are removed, as are all the (horizontal) connecting lines which lead away from the said nodes and which would lead from precisely these ancestor nodes to their adjacent nodes in the corresponding hierarchically higher peer groups, as well as all the induced uplinks which lead away from these ancestor nodes in the upwards direction.

In accordance with the PNNI protocol, an optimum route is determined in a known manner from the source node S to the destination node D based on the graph G3, for example using the Dijkstra routing algorithm, in which case the uplinks remaining in the graph G3 do not have to be treated differently from all the other (horizontal) connecting lines.

The best route obtained is a sequence F1 in a general notation:

node-n(=D), link-n-1,...,node-i+1, link-i,...,link-1,
node-1(=S).

It is in the nature of the Dijkstra routing algorithm that the respective best route is determined not only to a single, specific destination node D, but also to all the nodes of the network from among which the route which is of interest, for example to the destination node D, is being sought afterwards. In this context, this route is initially determined in the form of the sequence F1 by means of the Dijkstra algorithm

Then, the sequence is turned round and the sequence F2:
node-1(=S),link-1,...,link-i, node-i+1,..., link-n-1,
node-n(=D)
is formed.

The physical source node node-1=S is naturally of the hierarchically lowest level. According to the invention, all the other nodes may be as often as desired hierarchically higher or hierarchically lower physical or logical nodes in sequence. In particular, the destination node node-n=D does not necessarily have to be the hierarchically highest of the nodes occurring in the sequence.

A link link-i proves to be horizontal if node-i and node-i+1 are assigned to the same hierarchy level, that is to say belong to the same hierarchical peer group. A link link-i proves to be an uplink in the upwards direction (or respectively downwards direction) if the hierarchy level of node node-i is lower (or respectively higher) than the hierarchy level of node node-i+1 .

According to the invention, a sequence F3 in which the nodes never descend in the prescribed sequence with respect to their hierarchy level is derived from the sequence F2. Switching nodes and links of F2 may be replaced by others or cancelled here. To this end, it will be assumed that an auxiliary variable, referred to here as CurrentNodeLevel , which is initialized with the hierarchy level of the node node-1=S , and a second Boolean auxiliary variable, referred to here as $\text{BelowHighestReachedLevel}$, which is initialized with FALSE, are used. In an iteration loop all the components of the sequence F2 (the links and the nodes) are run through, starting at source node node-1=S and, from time to time, replacements or cancellations are performed - see the following algorithm:

```

-----
BelowHighestReachedLevel := FALSE;
35 current_node := node-1; //d.h. = Source Node S
CurrentNodeLevel := Hierarchy level of the
current_node;

for i:=1 step 1 to n-1 do

```

```

    if hierarchy level of node-i+1 lower than
      CurrentNodeLevel then
        if BelowHighestReachedLevel = FALSE then
          Determine that ancestor node of node-i+1 whose
5          hierarchy level is equal to the CurrentNodeLevel.
          Replace link-i which is an uplink, passed through
          in the downwards direction, through the associated
          horizontal link (with the same aggregation token).
          For how this can be done: see subtask-1 after
10          this algorithm.
          Replace node i+1 with the ancestor node which has
          been determined.
          BelowHighestReachedLevel := TRUE;
        else
15          Cancel link-i and node-i+1 from the sequence.
        end
      else
        if BelowHighestReachedLevel = TRUE then
          Replace link-i with that assigned (induced) uplink
20          or else horizontal link which starts from the
          level given by CurrentNodeLevel and leads to the
          node-i+1. For how this can be done: see
          subtask-2 after this algorithm.
          Node-i+1 is retained in the sequence without being
25          changed.
          BelowHighestReachedLevel := FALSE;
        else
          Retain link-i and node-i+1 in the sequence without
          being changed.
30        end
        CurrentNodeLevel := Level of the node-i+1;
      end

    Next i;
35 -----

```

Subtask-1:

The associated horizontal link in the hierarchically higher peer group is to be determined for a prescribed uplink:

5 The graph G_1 has m links (horizontal links and uplinks taken together). The number k from the set $1, 2, \dots, m$ constitutes a pointer to the information which is of interest with regard to precisely one link (for example its identity indications). In particular,
10 it will be assumed that there is a table RelationTbl with m elements. The elements represent the assignment chain from the initial uplink to the possibly induced uplink, to the uplink which may be induced from it anew, etc., to the horizontal link, induced from it, in
15 a hierarchically higher peer group:

$\text{RelationTbl}[j_1] := j_2$; // if there is no value k from 1 to m with $\text{RelationTbl}(k) := j_1$, then j_1 is an initial uplink

20 $\text{RelationTbl}[j_{q-1}] := j_q$;
.....
 $\text{RelationTbl}[j_{r-1}] := j_r$;
.....
 $\text{RelationTbl}[j_{s-1}] := j_s$;
25 $\text{RelationTbl}[j_s] := 0$;

which signifies:

link- j_1 is an uplink and induces link- j_2
link- j_{r-1} is an uplink and induces link- j_r
30 link- j_{s-1} is an uplink and induces link- j_s
link- j_s is a horizontal link.
// If there is no value j_{s-1} from 1 to m with $\text{RelationTbl}(j_{s-1}) := j_s$ but there is an entry $\text{RelationTbl}[j_s] := 0$; then j_s is a horizontal link in a
35 hierarchically lowest peer group.

Assuming that the link- i to be replaced corresponds to j_{q-1} , then the table RelationTbl is run through until $\text{RelationTbl}[j_s] := 0$. j_s identifies the horizontal link to be used.

Subtask-2:

For all m links of the graph $G1$ there is a table of the type:

5 $LinkLevelTbl[k]$ = lowest hierarchy level of the two boundary nodes of the link k ; for all $k=1, \dots, m$. link- i is designated by j_{q-1} . The table $RelationTbl$ is run through starting from $RelationTbl[j_{q-1}]$ in order to move on from one link to the next link and, at the same
10 time, $CurrentNodeLevel$ is always compared with the entries in $LinkLevelTbl$. Assuming that the value of $CurrentNodeLevel$ is equal to the value of $LinkLevelTbl[j_{r-1}]$, then j_{r-1} identifies the link which is being sought and which is to replace link- i .

15 It will be assumed that a sequence $F4$ is formed from the sequence $F3$, for example as follows:

```

for i:= 1 step 1 to n-1 do
    if link-i = uplink then
20         Insert that ancestor node behind link-i from
           source node  $S$  which is of the same hierarchy
           level as node- $i+1$ . In turn, insert behind it
           that horizontal link  $H$  which is assigned to the
           link- $i$  by virtue of the identical aggregation
25         token (the  $RelationTbl$  is run through starting
           at  $j_1 = \text{link-}i$  and  $H = j_s$  is found).
           end
    next i;
```

30 It will be assumed that a sequence of DTLs is formed from $F4$ in that the sequence $F3$ is broken up behind each uplink and a DTL information element with syntax which complies with PNNI protocol is formed from each sub-sequence generated in this way - which
35 describes completely the task, according to the invention, for the source node S .

According to the invention, a loop (in terms of the method according to the invention an alternative route via one or more peer groups with return to a peer

30 June 2004

group which has already been passed through, at a re-entry node which has not yet been passed through) can be included even if the information elements of the routing information are filled up again when the connection set-up message arrives in a physical switching node which is to be passed through (transit node). This is explained below with reference to the arrival of a connection set-up message in the first physical switching node of a hierarchically lowest peer group:

If the instantaneous hierarchically lowest peer group is left when a connection set-up message is passed on, then the respective uppermost DTL in the stack must previously be removed. If even one specific hierarchy area is departed from, all those DTL information elements which are uppermost in the stack and which contain route sections through the respective peer groups of the hierarchy area which is to be departed from must previously be removed. If a hierarchically lowest peer group is entered anew when a connection set-up message is being passed on, then new route sections must be determined and new respective DTLs formed. The pointers in the individual DTLs must always be set and moved forward in such a way that when a connection set-up message is received the pointers of all the received DTLs each point to a node link pair which contains either the receiving physical, hierarchically lowest node or else one of its ancestor nodes.

The boundary node (S'), as entry node into a further peer group, determines a new optimum route section as far as a destination node D'. The destination node D' is to be taken from the node link pair which follows that node link pair in the DTL which is uppermost in the stack of the DTLs received and to which the respective pointer points.

If this is not possible because the pointer is already pointing to the last node link pair, the same applies in relation to the next lower DTL in the stack,

30 June 2004

etc. It is within the sense of the invention that the PNNI protocol prescribes that complete and entire correspondence also to the received link indication, namely how the node node D' is arrived at, is imperative. Any attempt, for example, to arrive there in an even better way could yield a better route section but could also lead at the same time to the hierarchy area represented by the destination node D' not being entered at the envisaged boundary node from where the continuation of the connection set-up could end in a blind alley. That is to say, in addition to D', the horizontal link link-to-D' is also determined. Link-to-D' is taken from the node-link-pair which is associated with that DTL in which D' is located and to which the respective pointer points when the DTL is received.

The boundary node S', which takes into account the fact that it is only a transit node for the present connection set-up wish and that the received DTL stack is incomplete, therefore forms a possibly reduced graph G1' on the basis of its own graph G1, as follows:

All the nodes - together with adjacent links - with a hierarchy level which is higher than or equal to the hierarchy level of the node D' are removed from the graph G1, but D' itself is not, neither are those uplinks for which D' is an upnode and which are at the same time assigned to the link-to-D'. That is to say the following test will be carried out for all uplinks which have D' as an upnode:

As in subtask-1, a j_{q-1} can be assigned to an uplink. It will be assumed that table RelationTbl is run through starting from the RelationTbl[j_{q-1}] up to an entry RelationTbl[j_s] := 0. If j_s corresponds to the link-to-D', then the uplink may remain in the graph G1'. Otherwise it is removed.

Taking G1' as a basis, the graphs G2' and G3' are formed entirely analogously to the way in which the source node S formed the graphs G2' and G3', and, as described above, a DTL stack is determined, S' assuming

the function of S, and D' assuming the function of D (see description above).

5 All the DTLs, with the exception of the last of all (which contains D') are transferred from the resulting DTL stack, and the DTL stack which is to be passed on is thus completed.

10 An inclusion of the abovementioned loops in the method according to the invention requires that connecting lines exist which are connected to different boundary switching nodes of the first subnetwork or different representatives of the subnetworks of a higher hierarchy level; this is the only way that alternative routes are produced.

15 If it has been possible to determine more economic alternatives to loopless routes for a connection to the desired properties, routing information which contains indications relating to the connecting lines which have been checked is generated. In the method according to the invention, no knowledge
20 of the network topology of the further subnetworks (peer groups) - that is to say outside its own branch of the hierarchy in the switching node which performs the formation of the route - is necessary. However, it is essential that there are at least two connecting
25 lines from different boundary switching nodes of the first subnetwork to at least one further subnetwork.

The method according to the invention can be realized in the source switching node if the source switching node and the destination switching node are
30 associated with the same subnetwork and the routing information is formed with one or more loops to further subnetworks. In this context, the formation of routing information for a connection set-up message can be performed firstly by the source switching node for a
35 connection set-up. However, as an alternative it is possible for the source switching node forming the routing information in terms of the method according to the invention to be itself a transit communication system so that the connection set-up message is

30 June 2004

received and the routing information contained therein is further processed and developed for the following route to the destination switching node.

5 If the source switching node is associated with a first subnetwork and the destination switching node is associated with a further subnetwork, detectable bottlenecks can likewise be avoided by including loops in the route.

10 It is possible not to include connecting lines between subnetworks of a prescribable higher order in the route search by using a common assignment character for a plurality of connecting lines. This means that the inclusion of further subnetworks in the route search is limited to specific areas of the hierarchy of
15 the ATM communication network. If it is assumed that subnetworks are structured on the basis of geographical criteria, route searches over geographically far distant subnetworks are prohibited, which permits an upper limit for the expenditure in terms of resources
20 and thus for the economic expenditure for a connection set-up to be prescribed.

According to a further advantageous refinement of the method according to the invention, further subnetworks can be included in such a way that a
25 subnetwork is left repeatedly or else loops in further subnetworks are permitted, without which loops it would not be possible for the connection to come about with the same conditions. As a result of this expansion of the test to the presence of connecting lines,
30 connections in which the route resources and switching capacities of the first subnetwork are highly restricted can also be realized.

Further advantageous developments of the method according to the invention can be found in the rest of
35 the subordinate claims.

The formation of the routing information in switching nodes of an ATM communication network is explained in more detail by way of example with reference to Figures 1 to 3. Figures 1 to 3 show one

5 Figure 1 shows the ATM communication network from the point of view of the source switching node A.1.

10

15

20

25

30

35

The nodes are connected to one another by means of physical connecting lines (physical links). Links

30 June 2004

pb1,2,3 and pc1,2 between nodes of additional subnetworks are assigned additional information.

Legend:

Initial letter p= physical link,
5 h= horizontal link,
 u= initial uplink,
 U= induced uplink

pb1,hb1, and ub1, or pb2,hb2 respectively, and ub2 and
10 pb3,hb3 respectively, and ub3 and pc1, uc1, Uc1
respectively and hc1 and pc2, uc2, Uc2 and hc2,
respectively, are identified with a respective
identical aggregation token, by way of example.

The nodes in which the routing information is
15 formed according to Figures 1 to 3 are seen in each
case only by the peer groups which are outlined with
thick lines (the knowledge base stored in the
respective node comprises indications relating to these
peer groups). Instead of the physical connecting lines
20 which lead out from the hierarchically lowest peer
group, they see the uplinks which are respectively
assigned. Only the respective boundary nodes
themselves know about this assignment, but they do not
communicate this knowledge to the other nodes of the
25 peer group.

A route from the source node A.1 to the
destination node A.6 is sought. The connecting lines
pa5, pa6, which would permit a direct route from the
source node A.1 to the destination node A.6, are
30 blocked. The physical path which the connection set-up
should take is indicated by a thick line.

Activity of the source node A.1:

The graph G1 will be assumed to be stored in the source node A.1 in the form of a list of links together with their boundary nodes, i.e.

- 5 G1(A.1) - see Figure 1:
 (pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4 , A.5), (pa5 : A.5 , A.1), (pa6 : A.6 , A.5),
 (ub1 : A.2 , B), (ub2 : A.3 , B), (ub3 : A.4 , B),
 (uc1 : A.5 , C), (uc2 : A.6 , C).
 10 (hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
 (Uc1 : A , C), (Uc2 : A , C),
 (hc1 : C , AB), (hc2 : C , AB),

- The block lines are removed, specifically
 (pa5 : A.5 , A.1), (pa6 : A.6 , A.5), and
 15 the graph G2(A.1) is determined:

- (pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4 , A.5),
 (ub1 : A.2 , B), (ub2 : A.3 , B), (ub3 : A.4 , B),
 (uc1 : A.5 , C), (uc2 : A.6 , C).
 20 (hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
 (Uc1 : A , C), (Uc2 : A , C),
 (hc1 : C , AB), (hc2 : C , AB).

- All the ancestor nodes together with the adjoining lines are removed, specifically

- 25 (hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
 (Uc1 : A , C), (Uc2 : A , C),
 (hc1 : C , AB), (hc2 : C , AB),

and in this way graph G3(A.1) is determined:

- (pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4 , A.5),
 30 (ub1 : A.2 , B), (ub2 : A.3 , B), (ub3 : A.4 , B),
 (uc1 : A.5 , C), (uc2 : A.6 , C).

The application of the Dijkstra routing algorithm yields the sequence F1:

- 35 Destination node D = A.6, uc2, C, uc1, A.5, pa4, A.4, ub3, B, ub1, A.2, pa2, A.1=Source Node S.

The inverted sequence = F2 is:

Source Node S=A.1, pa2, A.2, ub1, B, ub3, A.4, pa4, A.5, uc1, C, uc2, A.6 = Destination Node D.

The sequence F3 is determined:

A.1, pa2, A.2, ub1, B, hb3, A, Uc1, C, hc2, AB.

The sequence F4 is determined:

A.1, pa2, A.2, ub1, A, hb1, B, hb3, A, Uc1, AB, hc1, C,
5 hc2, AB.

The information elements of the DTL stack are derived therefrom. The sequence F4 is split up after each uplink and an information element DTL is formed from each of the sub-sequences produced, as they are
10 transmitted to the next physical node. The pointer points to the x-th bracketed node link pair.

1. DTL: (A.1, pa2), (A.2, ub1), pointer=2
2. DTL: (A, hb1), (B, hb3), (A, Uc1), pointer=1
- 15 3. DTL: (AB, hc1), (C, hc2), (AB, x'00 00 00 00),
pointer=1

Activity of the transit node B.2 (entry node in the further subnetwork TB):

20 The transit node B.2 receives the following routing information ri:

1. DTL: (A, hb1), (B, hb3), (A, Uc1), pointer=2
2. DTL: (AB, hc1), (C, hc2), (AB, x'00 00 00 00),
pointer=1

25 Node B.2 has the network which can be seen by it stored in the form of a graph G1(B.2) - see the thick outlining in Figure 2:

(pb4: B.1, B.2), (pb5: B.1, B.5), (pb6: B.4, B.5), (pb7: B.3, B.4), (pb8: B.2, B.3),
30 (ua1: B.2, A), (ua2: B.3, A), (ua3: B.4, A),
(hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
(Uc1 : A, C), (Uc2 : A , C),
(hc1 : C , AB), (hc2 : C , AB).

Because this is transit traffic, G1'(B.2) is
35 formed, in that

initially D'(B.2) and link-to-D'(B.2) are determined:

D'(B.2) = A

Link-to-D'(B.2) = hb3.

All the nodes from the hierarchy level which is greater than or equal to the logical node A, and the respective adjacent links, with the exception however of D'(B.2) = A itself and of those uplinks for which A is the upnode and which are correlated with link-to-D'(B.2)=hb3, are removed from the graph G1(B.2) in the transit node.

The graph G1'(B.2) is produced:

(pb4: B.1,B.2), (pb5: B.1,B.5), (pb6: B.4,B.5), (pb7: B.3,B.4), (pb8: B.2,B.3),
 (ua3: B.4, A)

Since the transit node B.1 is not aware of any blockages, graph G1'(B.2) = graph G2'(B.2) applies. Since no ancestor nodes of the B.1 can be further away from it, G1'(B.2) = G2'(B.2) = G3'(B.2) applies.

The Dijkstra routing algorithm which is used yields a sequence F1:

D'(B.2) =A, ua3, B.4, pb7, B.3, pb8, B.2=S'(B.2)

In the inverted order, this yields F2:

S'(B.2) =B.2, pb8, B.3, pb7, B.4, ua3, A=D'(B.2)

The operations for forming F3 do not yield any changes,

i.e. sequence F2 = sequence F3.

Sequence F4 is formed from sequence F3:

S'(B.2) =B.2, pb8, B.3, pb7, B.4, ua3, B, hb3, A=D'(B.2)

The DTL information elements of the routing information ri are formed from sequence F4:

1. DTL: (B.2, pb8), (B.3, pb7), (B.4, ua3), pointer=2
 2. DTL: (B, hb3), A=D'(B.2), pointer=1
- the last (=2.) DTL of which is not accepted.

Thus, the following routing information ri is sent in the DTL stack format from the entry node B.2 to the further node B.3 in the further subnetwork TB:

1. DTL, newly formed:
 (B.2, pb8), (B.3, pb7), (B.4, ua3), pointer=2
2. DTL, received and further processed:
 (A, hb1), (B, hb3), (A, Ucl), pointer=2
3. DTL, received and further processed:

30 June 2004

(AB, hc1), (C, hc2), (AB,x'00 00 00 00), pointer=1

Activity of the transit node A.4 in the first subnetwork TA:

5 The node A.4 receives the following routing information ri:

1. DTL: (A, hb1), (B, hb3), (A, Uc1), pointer=3
2. DTL: (AB, hc1), (C, hc2), (AB,x'00 00 00 00),
pointer=1

10 Node A.4 has the network which can be seen by it stored in the form of a graph G1(A.4) which corresponds to the graph G1(A.1) stored by the source node A.1, see above and see Figure 1 (outlined with thick lines).

15 Because this is transit traffic, G1'(A.4) is formed in that initially D'(A.4) and link-to-D'(A.4) are determined:

D'(A.4) = C

Link-to-D'(A.4) = hc1.

20 All the nodes of the hierarchy level which is greater than or equal to the hierarchy level of D'(A.4) = C and the respective adjacent links, but with the exception of D'(A.4) = C itself and of those uplinks for which D'(A.4) = C is an upnode and which are
25 correlated with link-to-D'(A.4) =hc1, are removed from the graph G1(A.4).

Graph G1'(A.4) is produced:

(pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4 ,
A.5), (pa5 : A.5 , A.1), (pa6 : A.6 , A.5),
30 (ub1: A.2 , B), (ub2: A.3 , B), (ub3: A.4 , B),
(uc1 : A.5, C),
(hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
(Uc1 : A, C)

35 The blocked links are removed and this yields graph G2'(A.4):

(pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4 ,
A.5),
(ub1: A.2 , B), (ub2: A.3 , B), (ub3: A.4 , B),
(uc1 : A.5, C),

30 June 2004

(hb1 : B , A), (hb2 : B , A), (hb3 : B , A),
(Uc1 : A, C)

If all the ancestor nodes still contained,
together with the adjoining connecting lines, are
5 removed, this yields graph G3' (A.4):

(pa2 : A.2 , A.1), (pa3 : A.3 , A.1), (pa4 : A.4, A.5),
(ub1: A.2 , B), (ub2: A.3 , B), (ub3: A.4 , B),
(uc1 : A.5, C).

By means of a Dijkstra routing algorithm, the
10 following sequence F1 is determined therein as the best
route from the transit node A.4 to the representative C
of the additional further subnetwork TC:

D' (A.4)=C, uc1, A.5, pa4, A.4= S' (A.4)

By inverting the order, the sequence F2 is obtained:

15 S' (A.4) = A.4, pa4, A.5, uc1, C=D' (A.4)

Since the sequence F2 is never descending with
regard to the hierarchy level of the nodes which occur,
the operations for forming the sequence F3 do not yield
any changes: F3 = F2.

20 The sequence F4, namely:

S' (A.4) = A.4, pa4, A.5, uc1, AB, hc1, C=D' (A.4)

is obtained from the sequence F3.

The following DTL information elements of the
routing information ri are derived from the sequence
25 F4:

1. DTL: (A.4, pa4), (A.5, uc1), pointer=2
 2. DTL: (AB, hc1), (C, x'00 00 00 00), pointer=1
- the last (=2.) DTL of which is not accepted.

Thus, the following DTL stack contents are
30 transferred from the transit node A.4 to the node A.5:

1. DTL, newly formed: (A.4, pa4), (A.5, uc1),
pointer=2
2. DTL, received and further processed: (A, hb1), (B,
hb3), (A, Uc1), pointer=3
- 35 3. DTL, received and further processed: (AB, hc1),
(C, hc2), (AB,x'00 00 00 00), pointer=1

30 June 2004

Activity of the transit node C.1 in the additional further subnetwork TC:

The node C.1 receives the following routing information ri:

- 5 1. DTL: (AB, hc1), (C, hc2), (AB,x'00 00 00 00),
pointer=2.

The node C.1 has the network which can be seen by it stored in the form of a graph G1(C.1), see Figure 3:

- 10 (pc3: C.1, C.2), (pc4: C.2, C.3), (pc5: C.3, C.4),
(pc6: C.1, C.4),
(uab1: C.1, AB), (uab2: C.4, AB)
(hc1: C, AB), (Hc2: C, AB)

Because this is transit traffic, G1'(C.1) is
15 formed in that initially D'(C.1) and link-to-D'(C.1) are determined:

D'(C.1) = AB

link-to-D'(C.1) = hc2.

All the nodes of a hierarchy level which is
20 greater than or equal to the hierarchy level of D'(C.1) = AB, as well as the respective adjoining links, but with the exception of D'(C.1) = C itself and of those uplinks for which D'(C.1) = AB is an upnode and which are correlated with link-to-D'(C.1)=hc2, are removed
25 from the graph G1(C.1).

The graph G1'(C.1) is obtained:

(pc3: C.1, C.2), (pc4: C.2, C.3), (pc5: C.3, C.4),
(pc6: C.1, C.4),
(uab2: C.4, AB)

30 Since the node C.1 does not find any blocked connecting lines (these are located in the first subnetwork TA), G1'(C.1) = G2'(C.1).

Because no ancestor nodes relating to node C.1 can be additionally removed therefrom:

35 G1'(C.1) = G2'(C.1) = G3'(C.1)

Using the Dijkstra routing algorithm, the node C.1 will determine the sequence F1 as the best route:

D'(C.1)=AB, uab2, C.4, pc6, C.1=S'(C.1)

30 June 2004

By inverting the order, sequence F2 is obtained:

S'(C.1), pc6, C.4, uab2, AB=D'(C.1)

Since the sequence F2 is never descending with regard to the hierarchy level of the nodes which occur, the operations for forming the sequence F3 do not yield any changes:

F3 = F2.

The sequence F4, specifically:

10 S'(C.1)=C.1, pc6, C.4, uab2,C,hc2, AB=D'(C.1)
is obtained from the sequence F3.

The following DTL information elements of the routing information ri are derived from the sequence F4:

- 15 1. DTL: (C.1, pc6), (C.4, uab2), pointer=2
2. DTL: (C,hc2), (AB,x'00 00 00 00), pointer=1
the last (=2.) DTL of which is not accepted.

Thus, the following DTL stack is transferred from the node C.1 to the further node C.4 in the additional further subnetwork TC:

- 20 1. DTL, newly formed: (C.1, pc6), (C.4, uab2),
pointer=2
2. DTL, received and further processed: (AB, hc1),
(C, hc2), (AB,x'00 00 00 00), pointer=2.

25

Activity of the re-entry node A.6 for the second re-entry into the first subnetwork TA:

The node A.6 receives the following routing information ri:

- 30 1. DTL: (AB, hc1), (C, hc2), (AB,x'00 00 00 00),
pointer=3.

If node A.6 detects that the destination terminal is directly connected to it, it transmits the connection set-up message along the respective UNI interface (no longer PNNI interface) to the said destination terminal, in which case, in accordance with UNI protocol, no DTL information elements will be transmitted as well. The route is terminated.

30 June 2004

Patent Claim

1. Method for forming routing information (ri) in an ATM communication network, comprising switching nodes
5 (A.1..6, B.1..5, C.1..4), for a connection setup message from a source switching node (A.1) to a destination switching node (A.6),
- the switching nodes (A.1..6, B.1..5, C.1..4) being
10 assigned in each case to subnetworks (TA, TB, TC) and being networked to one another via connecting lines (pa1..6, pb1..8, pc1..6) and being unambiguously combined according to an ascending hierarchical order to form subnetworks (TAB) of a higher order, which
15 subnetworks are further combined into subnetworks (TABC) of the next higher order, and
- in which method all the topology information is taken into account via connecting lines within that branch of the communication network hierarchy which starts from the switching node (A.1, B.2, B.4, A.4, A.5, C.1, C.4) which
20 performs the formation of routing information, for the formation of routing information in the ascending direction with respect to the communication network hierarchy in accordance with the private network node protocol (PNNI protocol), and
- 25 in which method even advantageous routing loops are taken into account via one or more subnetworks (TB, TC, TAB, TABC) of any hierarchical level with different exit and re-entry switching nodes (A.2,A.4; A.5,A.6) in such a way that connecting lines (pb3, pc2) in the descending
30 direction with respect to the communication network hierarchy are also included by the switching node (A.1, A.5) which forms routing information (ri).

30 June 2004

2. Method according to Claim 1, characterized in that the switching nodes (A.1..6, B.1..5, C.1..4) operate in accordance with the principles of the private network node protocol (PNNI protocol).

3. Method according to Claim 1 or 2, characterized in that the formation of routing information in the source switching node (A.1) includes the routing loops for a connection set-up message.

4. Method according to Claim 1 or 2, characterized in that the formation of routing information in a transit switching node (A.5) includes the routing loops for a connection set-up message.
- 5 5. Method according to one of Claims 1 to 4, in which the source switching node (A.1) and the destination switching node (A.6) belong to the same subnetwork (TA) and the routing information (ri) is formed including one or more loops to form further
10 subnetworks (TB, TC).
6. Method according to one of Claims 1 to 4, in which the source switching node (A.1) and the destination switching node (A.6) do not belong to the same subnetwork and the routing information (ri) is
15 formed including one or more loops to form subnetworks of any hierarchy level during the course of the progression of the connection set-up.
7. Method according to one of Claims 1 to 6, characterized in that connecting lines (pcl, pc2)
20 between subnetworks (TA,TC) starting from a prescribable hierarchy level are not taken into account for the formation of the routing information (ri) which is subject to loops, in that a common assignment character is determined for a plurality of connecting
25 lines.
8. Method according to one of Claims 2 to 7, characterized in that the routing information (ri) is structured into information elements (DTL1..3) by the starting ATM switching node (VK1), one information
30 element (DTL1..3) being formed for the first subnetwork (TA) and for each order of subnetworks (TAB,TABC) included in the route.