PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
H04Q 11/04, H04L 12/56

A2

(11) International Publication Number: **WO 99/02009**

(43) International Publication Date: 14 January 1999 (14.01.99)

(21) International Application Number: PCT/SE98/01314

(22) International Filing Date: 3 July 1998 (03.07.98)

(30) Priority Data:
9702604-1 4 July 1997 (04.07.97) SE

(71) Applicant *(for all designated States except US)*: TELEFONAKTIEBOLAGET LM ERICSSON (publ) [SE/SE]; S-126 25 Stockholm (SE).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: ÖSTER, Gert [SE/SE]; Tamburingränd 8, S-175 48 Järfälla (SE). AXELL, Jörgen [SE/SE]; Vasseurs väg 35, S-182 35 Danderyd (SE). HÅGÅRD, Göran [SE/SE]; Lotterivägen 12, S-129 32 Hägersten (SE). ANDERSSON, Loa [SE/SE]; Skovelvägen 17, S-125 33 Älvsjö (SE).

(74) Agents: FRANKS, Barry et al.; Albihns Patentbyrå Stockholm AB, P.O. Box 3137, S-103 62 Stockholm (SE).

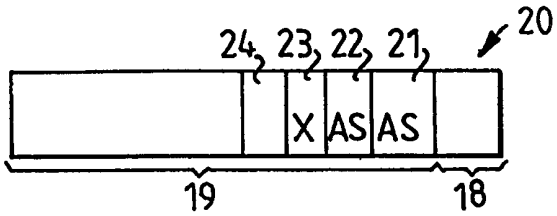
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: LOOP DETECTION

(57) Abstract

The invention relates to a loop-detection means produced by a source switch AS when establishing a route to a destination switch ES wherein the loop-detection means is a loop back cell (20) having the source switch AS in both source (21) and turning point (22) fields.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

Loop Detection

Technical Field of the Invention

The present invention relates to a method and a device for traffic-management in networks.
In particular the present invention relates to a method for detecting loops in networks, in particular in a connection-oriented networks such as, for example, an Asynchronous Transfer Mode (ATM) networks, and to a device for use in this method.

Description of Related Art

There are two major types of networks used for sending information between networks - connection-oriented networks and connection-less networks.

In connection-oriented networks, such as, for example, an Asynchronous Transfer Mode network, a message or information stream is transferred between two end systems (such as a computer or a telephone or the like),e.g. A and B, connected to the network by a connection which is established for, and maintained during the transfer. Thus before any message can be sent a logical/virtual connection set-up phase must take place. During the connection set-up process, which can be initiated by the end systems through signalling to the network or by management systems or by other means, a path through the network is selected, logical channels are allocated on the links between the nodes in the paths, resources are reserved in the links and the nodes, and the logical channels are interconnected in the nodes, which in this type of network comprise switches.

During the connection set-up process the switches register interconnected logical channels in tables so that the information in the subsequent data phase of the

connection can be transferred easily, typically by hardware logic, between interconnected logical channels.

In the case of ATM networks the information in the connections is sent in the form of cells, each containing 53 bytes of which 5 form a header and 48 are the payload. The header contains the identity of the logical channel for the connection on the current link. This identifier is changed as necessary by the switch as the cell passes through the node and is transmitted on the next link. The payload contains information from the stream or message being transferred by the connection.

For connections set-up as a consequence of initiating signals from the end systems, loops typically can not occur as this is prevented by functions in the network which handle the signalling.

In order to ensure that the route is functioning correctly special maintenance packets called loop-back cells can be sent out on the route. A loop back cell contains information in which the source of the loop-back cell and the destination of the cell can be identified. There is also an indicator, called a loop-back indicator, which shows whether the cell is travelling towards or from the loop-back point/destination. Thus if node A wishes to check that the route to node B is functioning correctly it transmits a loop-back cell with A as the source and B as the destination. The cell contains information which states that the cell is a non-user cell, and in particular that it is a maintenance cell. The maintenance cell has a field indicating that it is a loop-back cell. This field informs the destination switch or node, in this case node B, that it must turn around the cell and send it back to the source. Thus a node, upon receiving a loop-back cell, checks the loop-back point/destination field to see if it shall loop the cell back. If it does loop the cell back then it changes the loop-back indicator to show that the cell has been looped. It does not change the source identity or the loop-back identity. There may also be another field in the loop-back cell, called the correlation tag, which remains unchanged. If the route between A and B is functioning correctly then node A will

receive back the loop-back cell from B with the loop-back indicator indicating that the cell has been turned around. This event occurs some time after the insertion of the loop-back cell. This time depends on the length of the route and any queuing or other delays present on it.

The connection set-up procedure takes some time, in the order of tens to hundreds of milliseconds per node traversed, and thus introduces some latency for the end systems. However once the connection is established then the information can be transferred through the nodes/switches in an efficient way by the switch hardware. When a fault occurs in the network the connection is typically broken and a new connection has to be set up. The new connection will then be routed so as to avoid the troubled part of the network.

In large connection-less networks (by connection-less it is meant that no fixed communication is set up between communicating devices but that information is sent by the best route available at the time) e.g. the Internet, where information is sent as discrete packets of information between nodes, e.g. A, B, the packets comprise a header i.e. a part which contains information about, amongst other, its identification, where it came from (its source), where it must go (its destination), its length and other useful information. Each node uses a routing protocol to exchange routing information on how to best reach other nodes in a network. A description of a routing protocol is described in HUITEMA, CHRISTIAN, "Routing in the Internet", ISBN 0-13-132192-7, Chapter 4.4 to Chapter 4.2.6., the disclosures of which are incorporated within this application by reference. When these packets of information are being routed between nodes in a network there is always the possibility that a loop can occur. This may happen when a link or a node in a route is closed down or otherwise temporarily or permanently is unable to accept a new package of information from a preceding node. The preceding node in this case then forwards packets of information via an alternative path which it has stored in its routing information protocol. It is however possible that this alternative path at some stage contains a link which has a routing information protocol which leads

back to this preceding node. In this case the packets are routed from the preceding node via the alternative path back to the preceding node again. Thus a loop is formed. If a loop is formed then system resources are wasted on unnecessarily sending the packets on hops around the loop. Information packages caught in a loop can be considered as being worthless as even if the loop is broken it is unlikely that they can be delivered to their destinations at the correct time.

Loops are, however, usually transient problems. This is because the routing information protocols are updated with information on the network status and hence the primary and alternative paths for each node are constantly also being updated. This updating can be performed by, for example, exchanging distance vectors or information on link states between nodes.

Distance vectors contain information for each destination in the network, usually one element per destination, on the time, distance or 'cost' of sending a packet from the particular node to that destination. Each node exchanges, by a routing protocol, distance vectors with all its neighbouring nodes. These exchanges take place periodically and ,possibly, also when events occur which lead to changes in the costs. The distance vectors received from neighbours and knowledge about the cost of the links to the neighbours are used by nodes to recalculate its distance vectors. When a packet is received for forwarding to a certain destination the corresponding distance vector states which link should be used to forward the packet at the lowest cost. The path with the lowest cost is then used to route the packet. If a node or a link to a node fails in a particular path then its neighbouring nodes note that the cost of using this failed node or link is infinity and therefore an alternative path must be chosen. The information about the failed node is sent to other neighbouring nodes in an updated distance vector. These nodes then update their own distance vectors and send them to other neighbouring nodes and so on. Thus the information on the failed link slowly propagates through the network. Each updated distance vector influences all the neighbouring nodes and it takes a number of exchanges of distance vectors between neighbouring nodes before the distance vectors converge

and a new steady-state is re-established. During this convergence period it is possible that packages which were sent on the original path are influenced by the changing distance vectors and end up in a loop. This loop naturally ends when the distance vectors have converged. However during the convergence period, which can last up to the order of a minute, scarce processor and bandwidth resources are wasted.

Another reason for the transient nature of loops is that in most connection-less networks there are protocols in which each packet of information has a counter, usually in its header, called "Time to Live" which ensures that the package is dropped, i.e. destroyed and removed from the network, after either a certain time has lapsed or, more usually, after the package has passed through a predetermined number of nodes. Depending on the value of the "Time to Live" counter value when the loop occurs and on the delays in the loop, a loop can exist for period of up to approximately one minute. During this period scarce router processor resources are wasted on processing the packets in the loop which, as mentioned above, are considered worthless and are doomed to be discarded. A description of a "Time to Live" counter can be found in HUITEMA, CHRISTIAN, "Routing in the Internet", ISBN 0-13-132192-7, Chapter 3.3.1 "The Internet Header" which is incorporated within this application by reference.

Connection-less networks are flexible and messages can be sent quickly as there is no need to establish a reserved connection between source and destination before sending a message. These networks however require considerable processor capacity to determine the destination of each package being transmitted and loops can occur in the system which waste resources.

In order to improve the performance of networks, it has been proposed to form so-called "label switching networks" (also known as "multi-protocol label switching networks") in which a connection-less network is superimposed on a connection-oriented network. In other words a connection-less device such as a router is

associated with a connection-oriented device such as a switch in, for example, an ATM network. Each router uses its routing protocol to build up its routing table with destination information which states which is the best route towards each specific destination, i.e. which node, called the downstream node, it must forward traffic to in order to reach the desired destination. It will then request a logical channel number, known generically as a 'label', from that downstream neighbour for the traffic/packet to each destination. When the node gets a corresponding request from its upstream neighbours it can order its switch component to interconnect these channels. As this process is being carried out in the nodes in the network, link layer connections from each source to each destination will be established. These connections will follow the same paths as the best paths which have been decided by the routing protocols and distance vectors.

A problem with such label switching networks is that the router network can cause a loop which is then superimposed onto the connection-oriented network. As there is no special mechanism in a connection-oriented network for detecting loops it is possible for such loops to exist for relatively long periods of time before the router network notices in the normal way that there is a loop.

Summary

A problem with connection-less, e.g. router, networks superimposed on connection-oriented, e.g. switch and/or ATM, networks is that the existence of a loop causes a waste of system resources. Present art routing protocols designed for router networks do not provide guarantees against loops. They merely provide conditions in which the protocol rules ensure that the maximum life time of a loop is limited. This means that if a loop occurs then system resources are wasted until the loop reaches its maximum life time or the system undergoes some change which coincidentally breaks the loop.

The present invention seeks to reduce the wastage of system resources when a loop occurs by providing active means for detecting loops and for breaking loops in a connection-oriented network, in particular an ATM network, and more especially a network comprising a connection-less network superimposed on a connection-oriented network. This is achieved by a method in which when a source switch establishes a connection to a destination switch or node in a connection-oriented network it inserts a loop-detecting means, for example a loop-detecting cell, into the route. This loop-detecting cell contains identifying means which can be recognised by the source node if the loop-detecting cell returns to the node. It there is no loop then this cell will continue to the destination point of the route where it will be discarded. If there is a loop then the loop-detecting cell will return to the source node. It will be recognised by the source node and this will alert the source node that there is a loop present and action can then be taken to avoid the loop, e.g. the source node can inform an associated network controlling device, e.g. a router, that there is a loop present. This has the advantage that a loop can be detected within a few milliseconds instead of approximately one minute.

In a preferred embodiment of the invention the loop-detecting means is a loop-back cell in which, in the information fields of the cell, the source node or switch is both the source and destination (i.e. turning point) of the loop-back cell. If there is no loop present then the loop-back cell cannot return to the source node or switch without being (intentionally) looped as it is transmitted on a link which only leads to the destination point of the route. If there is a loop present then a loop-back cell which has the source identity in both the source and destination/loop-back fields will return to the source node or switch. The loop-back indicator shows that the cell is travelling towards the turning/loop-back point. The node or switch detects the loop by realising that it is itself the source and destination of the loop-back cell. It checks if it is the destination for the loop-back cell and if it is not the destination then the cell is transmitted along its logical channel. In order to identify that it is the originator of the loop-back cell it can use the source identity field or the correlation

tag, or both. It shall also check that the loop-back indicator shows that the cell is flowing in the direction towards the loop-back point.

In one embodiment of the invention, in order to ensure that packages of information are not sent into a potential looping path, after sending out a loop detection cell the source node can queue incoming traffic until a certain time has elapsed. This time is chosen such that there is a significant probability that if there is indeed a loop then the loop detection cell will return to the source node before the time elapsed. However the elapsed time must not be too long as this leads to unacceptable delays in the processing of the packets and requires large buffers. The time is therefore selected dependent on the type of information being sent and the expected length of the route.

In an other embodiment of the invention, in order to avoid delaying packets unnecessarily the node starts to send packets without delay along the route after the loop detection cell has been sent. This means that an assumption is made that the new route will not cause a loop. This assumption is acceptable in the case that the occurrence of loops is rare and/or it is unacceptable to delay packets.

Brief Description of the Drawings

Figure 1 shows a schematic diagram representing one embodiment of a simplified network according to the invention.

Figure 2 shows schematically one embodiment of a loop-detecting cell according to the invention.

Detailed Description of Embodiments

Figure 1, which will be used to illustrate the basic idea of the invention, shows an example of a hypothetical simple connection-less network superimposed on a

connection-oriented network consisting of nodes A, B, C, D and E. In this embodiment, each node has a connection-less device, shown for the sake of example as routers, AR, BR, CR, DR and ER respectively, and a connection-oriented device, shown for the sake of example as ATM switches AS, BS, CS, DS, and ES respectively. It is naturally possible for the present invention to be applied to hybrid networks in which connection-less devices and/or connection-oriented devices and/or combinations of connection-less and connection-oriented devices are present.

The routers and switches are interconnected by links L1, L2, L3, L4 and L5. Each node A-E has a route determining means, for example in the form of a routing table, resp. RT_A, RT_B, RT_C, RT_D, RT_E which contains route information means, for example in the form of route information entries E_{A-B}, E_{A-C}, E_{A-D}, E_{A-E}, etc. to E_{X-Y} (where $_X$ is the source and $_Y$ is the destination node) showing how packets are to be routed from the resp. node to the other nodes in the network. These entries in the route tables are built up from route cost identifying means, in a manner which is known from the prior art, for example in the form of distance vectors, which each node transmits to its neighbouring nodes in the network, and which will not be described in detail here. As an example, the route table for node A, RT_A, contains information on how to send information packets from node A to node B, from node A to node C, from node A to node D and from node A to node E. Note that there are several ways that node A can send packets to node E, i.e. via link L1, L3, L4, L5 or via links L2, L4, L5. Normally node A would send packets to node E via links L2, L4, L5 as this gives the shortest and therefore "cheapest" route, (assuming that the cost of using a link is the same for all links in the network) thus the distance E_{A-E} from source node A to destination node E would have a cost of 3. If link L2 is broken then node A would use links L1, L3, L4, L5 instead and the distance from node A to node E would be increased to 4 and the route information entry would be changed to reflect this.

Similarly node B has a distance to node E, E_{B-E}, which is 3, if node 3 is intact and which rises to 4 if node 3 fails.

For node D the distance to node E, E_{D-E}, is 1 unless link L5 is broken in which case the distance becomes infinity - i.e. it becomes impossible to reach node E from node D.

The routers AR-ER use their routing tables to establish connections through their switches AS-ES as described previously. Thus in a normal situation router AR when it wishes to send traffic to E requests a label from node CR, i.e. in the case of ATM a logical channel number, CLEa-c for the traffic to E. Packets destined for E received by A will then be forwarded to C through that channel.

Router CR will ask D for a label, CLEc-d, for traffic for E and can then order its switch CS to interconnect CLEa-c with CLEc-d thereby allowing the traffic from A to E to be handled by switch CS and thus bypassing, and saving resources in, router CR.

In the same way nodes X which are upstream of A for traffic towards E will ask A for a label CLEx-a which then allows AR to order its switch AS to interconnect CLEx-a with CLEa-c.

These connections established as described above will follow the paths determined by the routing tables RTx in the routers. If some event causes these routers to generate a loop then this will lead to a connection loop.

Consider, for example, what happens if there is a break in link L4. This will be detected by router CR which will try to inform its neighbours, routers AR and BR, that its distance to D and E is infinite. If, for any reason, this message to B is lost then AR will believe that it can reach D via B at a cost of 3 and AR will inform C of this. This leads to a routing loop.

Previous to this event, A has a channel CLEa-c to C for the traffic towards E, and B has a corresponding channel CLEb-c for its traffic towards E.

As a result of this event node A will release its channel to C and will instead ask B for a channel to B, CLEa-b for its traffic towards E. BS will interconnect that with the existing channel CLEb-c. C will ask A for a channel CLEc-a for its traffic towards E. Router AR will order switch AS to interconnect CLEc-a with CLEa-b and router CR will order switch CS to interconnect CLEb-c with CLEc-a, thus forming a loop connection. As long as the loop persists, all packets towards E will circulate in the loop and cause a waste of links and switch resources and eventually lead to problems such as overflows in switch buffers and the like.

The present invention reduces this problem of wasted resources by providing a loop-detection means. In the preferred embodiment of the invention shown in figure 2 the loop-detection means comprises a specially modified maintenance loop-back cell 20 which is produced and transmitted by a switch whenever it establishes a communication channel to another switch. Cell 20 has a header 18 and a payload 19. This loop detection cell 20 has the originating switch identified both in the source field 21 as the source as well as in the turning point field 22. This cell 20 can also have a loop-back indicator field 23 which shows if the cell has been looped back. Thus if switch AS is to establish communication with switch ES it produces a loop detection cell 20 in which the cell will contain AS as the source and AS as the turning point. This cell 20 is then sent on the route which should lead to switch ES and normally it should never come back to switch AS. If switch AS subsequently receives back any cell which contains AS in both source and turning point fields (21, resp. 22) and with the loop-back indicator 23, if present, showing that the cell has not been looped back then it means that a loop has been formed. The source can also be identified by some other source field, for example, a correlation tag 24 as specified in the ITU-T 610 specification. Switch AS can then take appropriate action to prevent the loop being used and/or to break the loop and/or send an alarm

out of the network. In this way a loop can be detected in the time it takes the loop-detection cell to hop round the loop in the newly established route. Generally it can be considered that loops contain only a few nodes and links. Thus the time that a loop detection cell takes to hop around the loop generally will be short and a loop will be detected quickly in a method according to the invention.

If there is no loop then the loop-detecting cell will continue to the destination node and be discarded, or possibly looped back with the loop-back indicator changed to 'looped'.

If there is a loop but the loop-detecting cell is lost or expires before returning to the switch which produced it then the loop will be removed in due time in the normal way.

In this preferred embodiment of the method according to the invention the switch which produces the loop-detecting cell assumes that there will not be any loops present on the route and forwards packets of information on the route after sending the loop-detecting cell. This avoids delaying the packets of information but can lead to some packets being lost if there is indeed a loop. The number of packets lost will, however, owing to the more rapid loop detection enabled by use of a loop-detection cell, be less than the number which would have been lost anyway if no loop-detection cell had been sent. Thus this embodiment leads to an improved transmission performance and no signal delay.

In a second embodiment of the method according to the invention the switch which produces the loop-detecting cell assumes that there is a loop present on the route. In this case it has a timer which is set to expire a predetermined time interval after the loop-detection cell has been sent on the alternate route. The length of the timer is chosen to correspond to an appropriate maximum loop length. Such a maximum length could, for example, be chosen to correspond to the time which a packet would take to hop around a loop of an arbitrary number of links, or could be a set

number of transmission periods. Packets with a destination on the route are stored until the end of the timer period. If the loop-detection cell has not yet returned to the switch then it is assumed that there is no loop present and the stored packets of information are transmitted on the route. This embodiment leads to more improved transmission quality as no packets of information will be lost in the loop at the cost of introducing a delay into the transmission of one or more packets of information.

The choice of which embodiment of the invention to use naturally depends on the type of information in the data packets - a delay may be acceptable in a data transmission which however cannot accept the loss of a few packets of information without suffering a discernible loss of quality, while a telephony conversation may be able to accept a loss of quality but not a transmission delay. It is possible for a node or switch to determine which embodiment to use for each packet it receives: some being transmitted immediately while others are delayed.

In a third embodiment of the present invention, not shown, the loop detection cell is produced at pre-programmed or predetermined intervals.

In a fourth embodiment of the present invention, not shown, the loop detection cell is not produced each time a route is established but after a route has been established a pre-programmed or predetermined number of times.

The invention is naturally not limited to networks having only 5 nodes and 5 links but is adaptable to networks of any size.

While the inventive concept has been illustrated by the example of a connection-less network superimposed on a connection-oriented network, it is of course conceivable to use the present invention in purely connection-oriented networks as well as in networks comprising a connection-oriented network or switch.

The invention is not limited to loop-detecting means based on a loop-back cell but can use any other cell which can be suitably designed or modified to inform an originating device that a loop exists.

Claims

1. Method for detecting loops in a network comprising nodes (A-E) and links (L1-L5) interconnecting said nodes wherein at least one node comprises a connection-oriented network device, for example, a switch (AS-ES),
characterised by the steps of:
a) a source connection-oriented network device, for example switch (AS), produces a loop-detecting means (20) whenever it sets up a communication route with a destination connection-oriented network device, for example switch (ES);
b) said source connection-oriented network device (AS) sends said loop-detecting means toward said destination connection-oriented network device (ES) via said communication route; and,
c) said source connection-oriented network device (AS) determines a loop is present if said loop-detecting means returns to said source connection-oriented network device (AS).

2. Method according to claim 1 characterised by the steps of:
d) said source connection-oriented network device (AS) starts a loop-detection timer at, or after, step b) above;
e) said source connection-oriented network device (AS) stores packets for said destination connection-oriented network device (ES) until said timer expires; and,
f) said source connection-oriented network device (AS) sends said packets via said communication route if said loop-detecting means does not return to said source connection-oriented network device (AS) before the timer has expired.

3. Method according to claim 1 or 2 characterised in that said loop-detecting means is a loop-back cell produced by said source connection-oriented network device (AS), said loop-back cell having fields which identify said source connection-oriented network device (AS) as both source and turning-point.

4. Method according to any of the previous claims characterised in that said network comprises at least one node (A-E) comprising a connection-oriented network device (AS, BS, CS, DS, ES) and a connection-less network device (AR, BR, CR, DR, ER).

5. Loop-detecting cell (20) having a source identifying means, for example, source field (21) and/or correlation tag (24), and a turning point identifying means, for example, turning point field (22), characterised in that the device (AS-ES) identified in said source identifying means (21, 24) is the same as the device, (AS-ES) identified in said turning point identifying (22).

6. Loop-detecting cell (20) according to claim 5 characterised in that it comprises a loop-back indicator means such as a loop-back field (23) for indicating if said cell has been looped back.

1 / 1

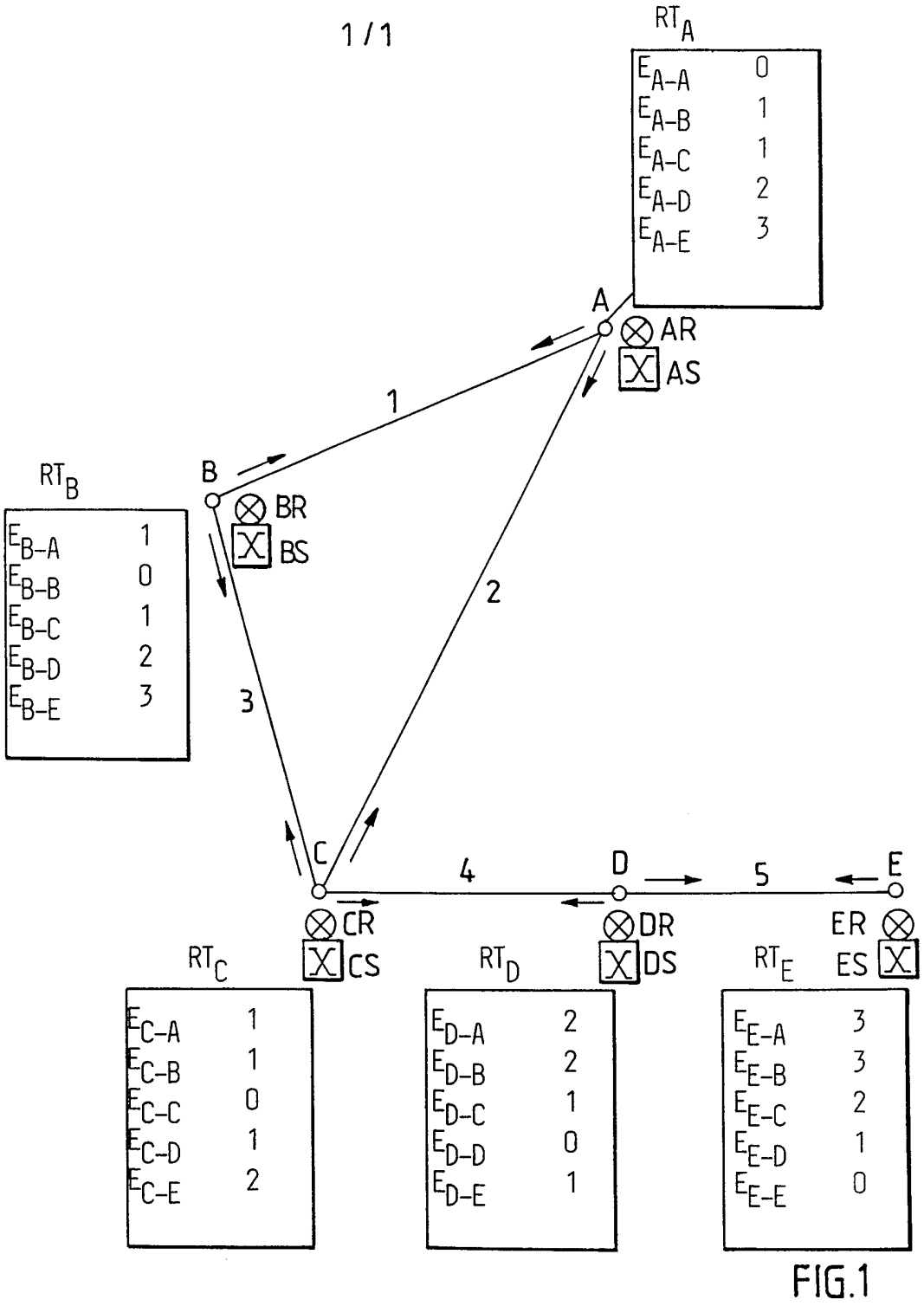

RT_A
E_A-A 0
E_A-B 1
E_A-C 1
E_A-D 2
E_A-E 3
A
AR
AS
1
B
RT_B
E_B-A 1
E_B-B 0
E_B-C 1
E_B-D 2
E_B-E 3
BR
BS
2
3
C
4
D
5
E
CR
CS
DR
DS
ER
ES
RT_C
E_C-A 1
E_C-B 1
E_C-C 0
E_C-D 1
E_C-E 2
RT_D
E_D-A 2
E_D-B 2
E_D-C 1
E_D-D 0
E_D-E 1
RT_E
E_E-A 3
E_E-B 3
E_E-C 2
E_E-D 1
E_E-E 0

FIG.1

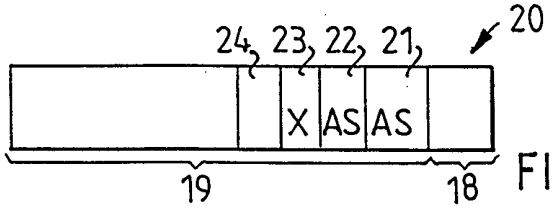

24
23
22
21
20
X
AS
AS
19
18

FIG.2