

(19) Korean Intellectual Property Office (KR)**(12) Official gazette of unexamined patent (A)****(51) Int. Cl.⁷
H04L 12/56****(11) Publication number 10-2005-0077938
(43) Publication date August 4, 2005**

(21) Application No.	10-2004-0005630
(22) Application date	January 29, 2004

(71) Applicant	KT Corporation 206 Jeongja-dong, Bundang-gu, Seongnam-si, Gyeonggi-do
(72) Inventors	LEE, Yeongseok 126-1007 Hanbit Apts., Eoeun-dong, Yuseong-gu, Daejeon LIM, Taesung 123-706 Hanbit Apts., Eoeun-dong, Yuseong-gu, Daejeon KIM, Dongjoo Expo Apts. (101-309), Jeonmin-dong, Yuseong-gu, Daejeon
(74) Agents	HWANG, Uiin LEE, Junghoon

Examination
requested: No**(54) METHOD FOR IMPLEMENTING PACKET-SPECIFIC LOAD BALANCING IN A MULTI-PROTOCOL LABEL SWITCHING NETWORK USING MULTI-LABEL ASSIGNMENT****Abstract**

The present invention discloses a method for implementing packet-specific load balancing in a multi-protocol label switching (MPLS) network, using multi-label assignment.

The method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment of the present invention comprises a first step for, when receiving routing information from adjacent routes in a MPLS (Multi Protocol Label Switching) network, verifying interface information for the same routing information received for a specific destination; a second step for generating FECs (Forwarding Equivalence Classes) corresponding to each interface receiving the same routing information by using the same routing information and the interface information; and, a third step for generating and assigning separate labels for each FEC. In addition, in the present invention, even when the routing information is the same, packets are transmitted by assigning different labels to each path, which enables load balancing across multiple links, thereby enhancing the efficiency of network resources. In addition, the method of the present invention combines the internal label or EXP bit value in the forwarding label, even when linked with traffic engineering or virtual private network (VPN) services, which are additional functions in MPLS, thereby allowing for packet-specific load balancing across multiple links, rather than a single link, when generating the label path to a specific destination.

Representative figure

FIG. 2

Specification**Brief Description of Figures**

FIG. 1 is a block diagram depicting the configuration of a simplified MPLS network for describing the load balancing method according to the present invention.

FIG. 2 is a flowchart for describing the load balancing method in FIG. 1 according to the present invention

FIG. 3 depicts the LSP generation information according to the load balancing implementation of the present invention in an MPLS network.

FIG. 4 is a flowchart for describing the load balancing method according to the present invention, in scenarios where traffic engineering or virtual private network (VPN) services are linked within the MPLS domain.

FIG. 5a depicts the configuration of the conventional MPLS header.

FIG. 5b depicts the configuration of an MPLS header, with a traffic engineering header and a VPN header inserted.

Detailed Description of Invention

Purpose of Invention

Technical Field to which the Invention Pertains and Background Art

The present invention relates to a method for implementing load balancing on an MPLS (Multi Protocol Label Switching) network, and more specifically, the present invention relates to a method that facilitates packet-specific load balancing for equal-cost paths by generating different FECs (Forwarding Equivalence Classes) for each interface, and assigning a different label value to each FEC when routing information for the same destination is received with identical metric costs via plurality of different interfaces.

Conventionally, packet forwarding in IP networks delivers packets to a destination on a hop-by-hop basis, by referencing a routing table calculated by a routing protocol. At this time, when routing information for the same destination is received with identical metric costs from multiple interfaces, equal-cost load balancing is applied across the corresponding paths.

To implement load balancing across a plurality of interfaces, some or all of - source MAC address, destination MAC address, source-destination MAC address, IP of the source, IP of the destination, source-destination IP, source port, destination port, source-destination port information - are combined in a router of an IP network, assigned a hash code, and then hashing is performed to facilitate load balancing between the equal-cost paths. At this time, to implement more granular load balancing, information from an upper layer is utilized.

However, in MPLS networks, label switching routers (LSR: Label Switch Router) differ from those in IP networks because they purely switch pre-assigned label values and deliver packets on a hop-by-hop basis, and during forwarding, the routing table is not separately referenced. In MPLS networks, IP packets are initially classified based on their characteristics (in other words, whether the corresponding packets utilize the same path within the network and have the same processing routine). Packets with identical characteristics are classified into the same FEC (Forwarding Equivalence Class), and only one label value is assigned to each FEC and the next-hop determination is made. The packets are ultimately reclassified into a FEC with identical characteristics, and one end-to-end label switching path (LSP: Label Switching Path) is determined through label assignment and next-hop recalculation for the FEC.

In this manner, the method for setting the path in an MPLS network uses only one specific path, even if multiple paths exist for the same destination. Consequently, the load balancing function cannot be implemented between links, leading to a problem of inefficient use of network resources.

To address this inefficiency, a method of applying MPLS traffic engineering is applied; however, this approach still encounters the issue of only one label being assigned to an FEC with the same destination.

In addition, for services such as traffic engineering or VPNs (Virtual Private Networks), which are MPLS applications, a separate labeling header is added to the inside (inner) for traffic engineering or VPN purposes, in addition to the default forwarding label. However, when delivering packets, only the default external label value is referenced, thus perpetuating the same problem.

Technical Problem to be Solved by the Invention

Therefore, to resolve the above-described problems, the objective of the present invention is to improve the method for generating FECs and assigning label values to the FECs in a MPLS network in order to implement packet-specific load balancing functionality across multiple links, thereby enhancing the efficiency of network resources.

Configuration and Action of the Invention

To achieve the above-described objectives, the method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment of the present invention comprises a first step for, when receiving routing information from adjacent routes in a MPLS (Multi Protocol Label Switching) network, verifying interface information for the same routing information received for a specific destination; a second step for generating FECs (Forwarding Equivalence Classes) corresponding to each interface receiving the same routing information by using the same routing information and the interface information; and, a third step for generating and assigning separate labels for each FEC.

Herein, preferred embodiments of the present invention will be described in greater detail with reference to the accompanying figures.

FIG. 1 is a block diagram depicting the configuration of a simplified MPLS network for describing the load balancing method according to the present invention; FIG. 2 is a flowchart for describing the load balancing method in FIG. 1 according to the present invention.

The label switching routers (hereinafter referred to as “LSR”)(R11 to R14) use routing protocols to propagate routing information to adjacent LSRs and exchange the information. Through this process, each LSR generates a routing table that includes the routing information for the entire MPLS network (step 201).

These LSRs use labels to exchange routing information with the adjacent LSRs and continuously update their routing tables.

At this time, if the LSRs have not completed updates to the same routing information through one interface, they verify whether the updates have been completed through a plurality of different interfaces (step 202).

In other words, in calculating the paths to a specific destination using the routing table, the LSR refers not only to the routing information to that specific destination but also the interface information propagated by the corresponding routing information.

For instance, in LSR (R12) in FIG. 2, the routing information indicating that the delivery of information to destination address 3.3.3.3 may be routed through LSR (R14), and the delivery of information to LSR (14) may be routed through LSR (R13, R15) is received from the LSRs (R13, R15) via interfaces fe 0/0 and fe 0/1, respectively. In other words, LSR (R12) receives the routing information indicating that the delivery of information to destination 3.3.3.3 is possible through LSR (R13) or LSR (R14), and updates the routing table accordingly.

In this case, the conventional LSR (R12) processes the routing information for the destination 3.3.3.3 in an identical manner, selecting a path for routing through either LSR (R13) or LSR (R14). However, in the present invention, whereas the destination 3.3.3.3 remains the same, the path selection differs, and as a result, the routing process is handled differently.

In this approach, when identical routing information is delivered through interfaces fe 0/0 and fe 0/1 of a different plurality (N), the LSR (R12) considers not only the routing information but also the interface information, and generates N forwarding tables (FIB: Forwarding Information Base) corresponding to the number of interfaces (N)(step 203).

Next, the LSR (R12) generates N FECs corresponding to each FIB through N generated FIBs (step 204).

To achieve this, the LSR (R12) establishes a one-to-one correlation between each interface's information and the destination address in the FIBs, and generates hash codes for each correlation. In other words, the LSR establishes a one-to-one correlation between each interface's information and the destination address to generate a hash code and FEC for each pair, as shown in {destination address (3.3.3.3), interface 1 (fe 0/0)}, {destination address (3.3.3.3), interface 2 (fe 0/1)}.

In prior art, the hash code is generated solely based on the destination address. Therefore, even if the path differs, only one FEC is generated. In contrast, in the present invention, interface information is added to generate the hash code. As a result, even if the destination remains the same, different paths (interfaces) generate a plurality of different FECs

corresponding to each path.

In this manner, following the generation of the FECs, when the label for the corresponding network information is assigned by the router at the entry point of the MPLS domain, known as the ingress label edge router (Ingress LER), this router references the number of combinations of routing information and interface information generated by the FIB and generates N LFIBs (Label Forwarding Information Bases) (step 206).

The plurality of label information assigned to each of the plurality of FECs for the specific destination in step 205 is added to the LFIB.

Therefore, when the ingress LER (R12) calculates the path for transmitting IP packets to a specific destination, it calculates the next-hop for each table through each LFIB. Thus, the IP packets are transmitted via a plurality of different output interfaces (step 207).

In this manner, the IP packets are transmitted across a plurality of interfaced paths rather than being transmitted through one path, thereby enabling load balancing.

In the case that the LSR (R12) receives routing information for the same destination through one interface in step 202, as in prior art, only one FIB is generated (step 208), and only one FEC is generated through this one FIB (step 209).

One label is generated and assigned for one generated FEC (step 210), and only one LFIB is generated, and the label is added to the LFIB (step 211).

Therefore, IP packets are transmitted to the destination through the one generated LFIB (step 212).

FIG. 3 depicts the LSP generation information according to the load balancing implementation of the present invention in an MPLS network.

If IP packets are transmitted from LSR (R0) to LSR (R7) via LSRs (R1 to R6) and the load balancing according to the present invention is not implemented, the IP packets are transmitted through only one of the four paths: R0→R1→R3→R7, R0→R1→R4→R7, R0→R2→R5→R7, or R0→R2→R6→R7.

However, if load balancing according to the present invention is implemented, IP packets are transmitted through all four paths: R0→R1→R3→R7, R0→R1→R4→R7, R0→R2→R5→R7, and R0→R2→R6→R7, as depicted in FIG. 3, thereby enabling the efficient usage of network resources.

FIG. 4 is a flowchart for describing the load balancing method according to the present invention, in scenarios where services such as traffic engineering or virtual private network (VPN) (which are additional functions of MPLS) are linked within the MPLS domain.

The MPLS header is configured as depicted in FIG. 5a, and in the case of being linked with additional services such as traffic engineering or VPN, a traffic engineering header (52) and a VPN header (53) are sequentially added and inserted to the labeling header (51) for packet forwarding, as depicted in FIG. 5b. In this case, a separate label for each additional service is added using an internal label format.

Therefore, LSR (R12) verifies the presence of an internal table in the MPLS header. This verification allows the LSR to determine whether additional services are linked to the MPLS. If such additional services are indeed linked, the LSR must consider them and implement load balancing.

To achieve this, the LSR (R12) initially verifies the presence of internal label values within the MPLS header (step 401).

Upon verification, if an internal label is present within the MPLS header, the LSR (R12) uses the internal label value and the external label value to generate the hash code (step 402).

In addition, the MPLS header includes a 3-bit EXP (Experimental) field for packet differentiation, and in step 402 the external label values, excluding these EXP bits, may be combined to generate the hash code.

Once the hash code is generated, the LSR (R12) combines the generated hash codes to generate N FIBs corresponding to each interface (step 403). Subsequently, the LSR uses each FIB to generate N FECs corresponding to each FIB (step 404).

The LSR (R12) generates and assigns labels corresponding to each of the generated N FECs (step 405) to generate N LFIBs (step 406). The process of generating these LFIBs is identical to the process for generating LFIBs described in FIG. 2.

Once the LFIBs are generated, the LSR (R12) calculates the next-hop for the label through each LFIB, and transmits the IP packets across the plurality of different paths (step 407).

In step 401, if an internal label is not present, in other words, if functions of traffic engineering, VPN, or the like, are not applied (which are additional functions in the MPLS), the same process depicted in FIG. 2 is followed to execute load balancing. Therefore, if an internal label is not present in FIG. 4, the same reference numeral from FIG. 2 is assigned, and its description thereof is omitted.

Effect of the Invention

As described above, the method for implementing load balancing of the present invention transmits packets by assigning different labels to each path, where these paths are unique for the same routing information; the present invention enables load balancing across multiple links, thereby enhancing the efficiency of network resources.

In addition, the method for implementing load balancing of the present invention can also allow for packet-specific load balancing by utilizing multiple links, rather than the same link, when generating label paths to a specific destination, and this is achieved by combining internal label values or EXP bit values with forwarding labels, even when a traffic engineering or VPN service is linked with an MPLS network.

(57) Scope of Claims

Claim 1.

A method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment comprising

a first step for, when receiving routing information from adjacent routes in a MPLS (Multi Protocol Label Switching) network, verifying interface information for the same routing information received for a specific destination;

a second step for generating FECs (Forwarding Equivalence Classes) corresponding to each interface receiving the same routing information by using the same routing information and the interface information; and,

a third step for generating and assigning separate labels for each FEC.

Claim 2.

The method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment of claim 1,

wherein the generation of FECs is characterized by generating FECs for each interface, and generating a hash code for each pair of interface information and received routing information, corresponding in a one-to-one manner to the address of the specific destination.

Claim 3.

The method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment of claim 1, is further characterized by comprising

verifying the presence of an internal label within the MPLS header, and if the internal label is present, it is combined with the external label to generate a hash code, and this hash code is then used to generate FECs corresponding to each interface.

Claim 4.

The method for implementing packet-specific load balancing in a multi-protocol label switching network using multi-label assignment of claim 1, is further characterized by comprising

verifying the presence of an internal label within the MPLS header, and if the internal label is present, the EXP (Experimental) bit of the MPLS header is combined with the external label to generate a hash code, and this hash code is then used to generate FECs corresponding to each interface.

Figures

FIG. 1

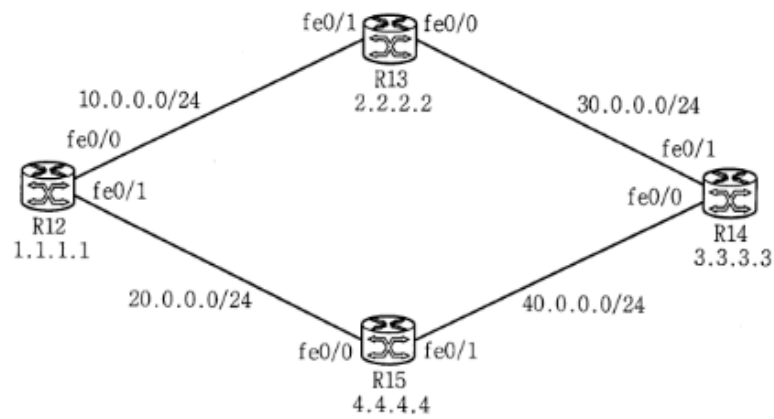


FIG. 2

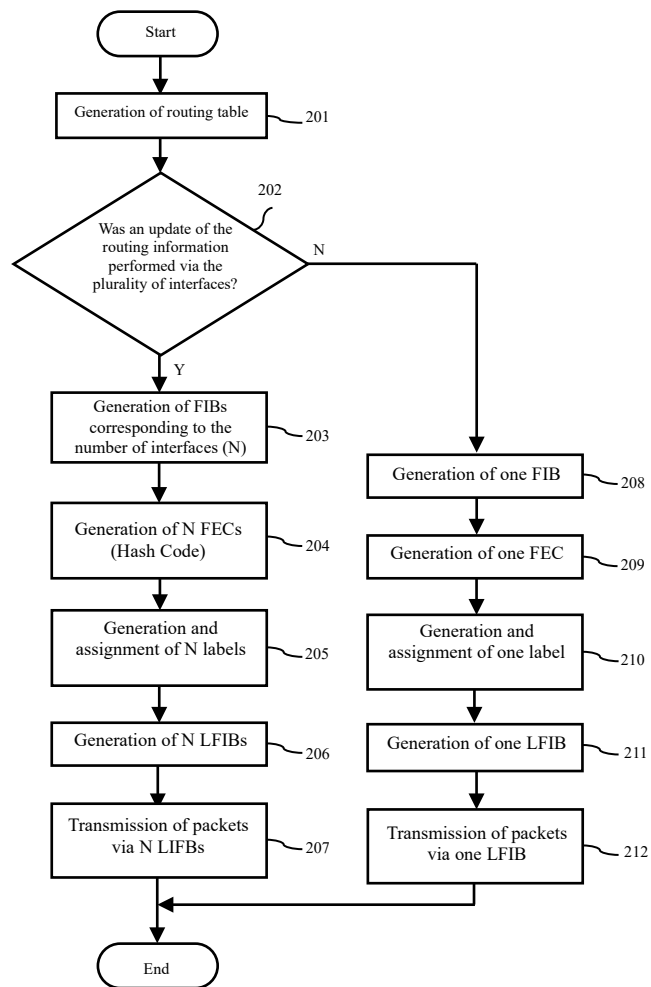


FIG. 3

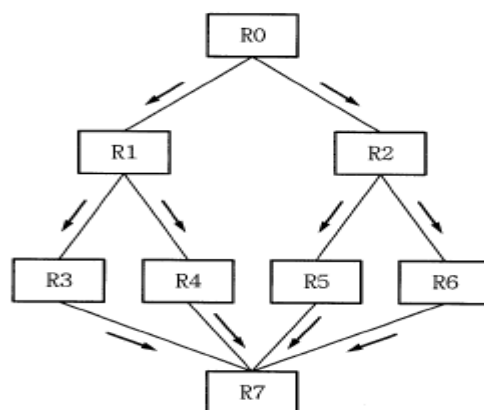


FIG. 4

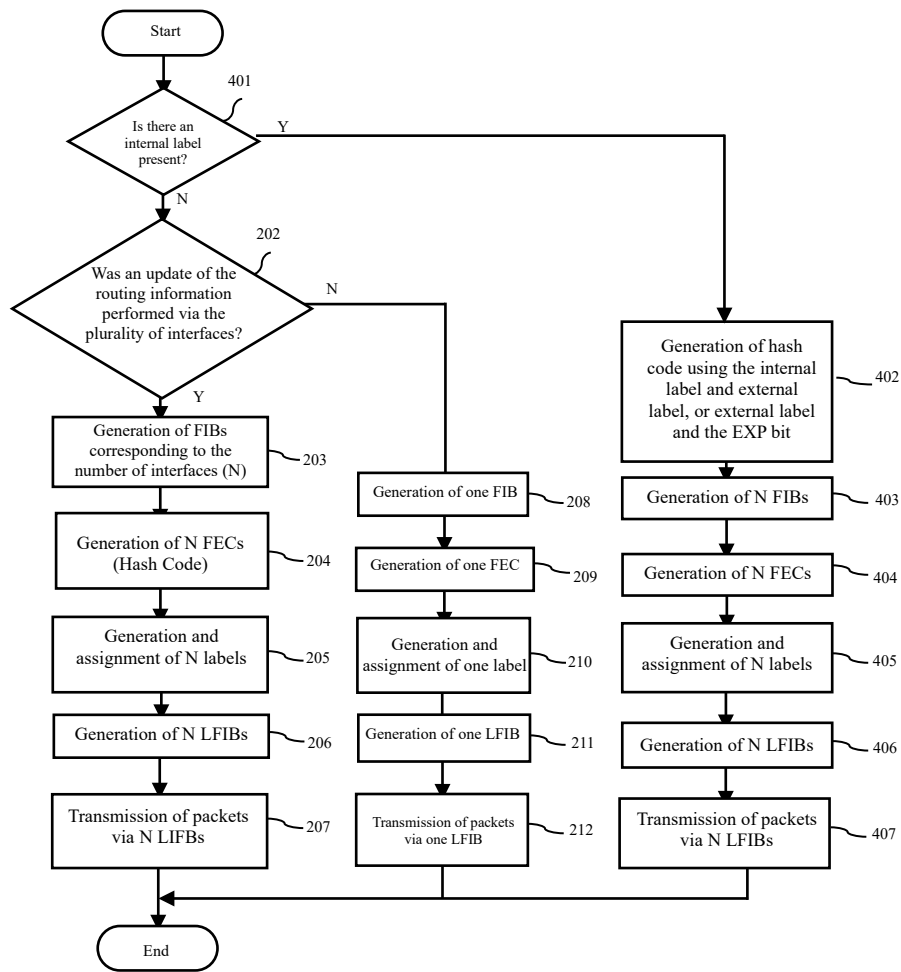


FIG. 5a

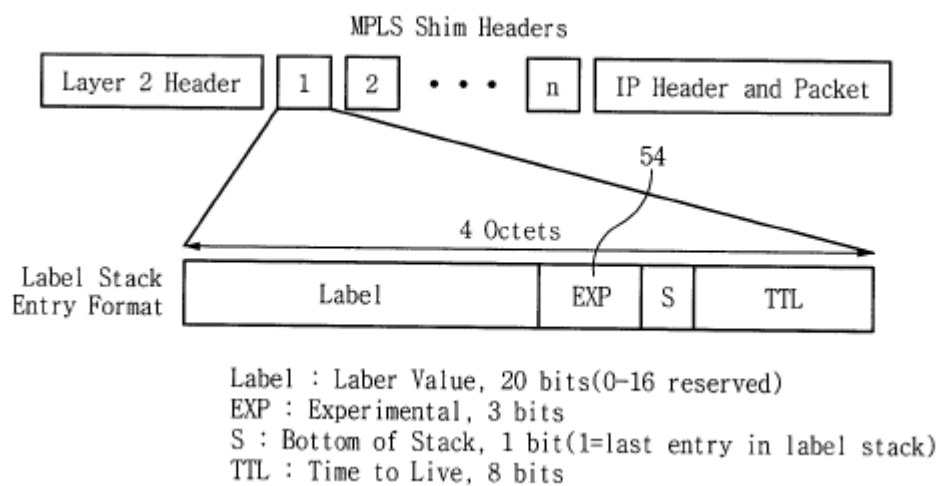
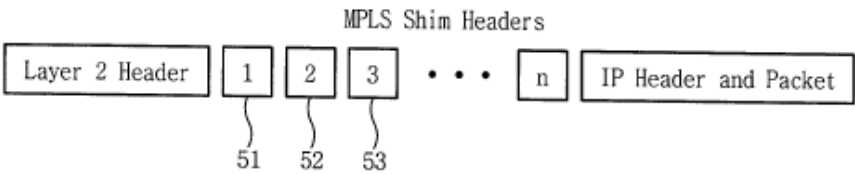


FIG. 5b



I, Corey Colling, declare as follows:

1. I am a native speaker of English and am well versed in both the Korean and English languages and have over 15 years of experience translating Korean technical documents into English.
2. The following translation of the corresponding source text from Korean into English is accurate and complete to the best of my knowledge.

I declare under penalty of perjury of the laws of the United States of America that the foregoing information is true and accurate to the best of my knowledge. Statements made herein are to the best of my knowledge true and are based on information that I believe to be true and further these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code.

Executed on: July 2, 2025, at 210 Bayside Pt SW, Airdrie, Alberta, T4B 2X6

Corey Colling

Signed by Corey Colling

