A Fast Multicast IP-Routing Lookup Scheme

Keng-Ming Huang and Chung-Ju Chang, *Senior Member, IEEE*

***Abstract*—This letter proposes a fast multicast IP-routing lookup scheme, which adopts a *compression bit map* conception for forwarding information. The proposed scheme can achieve fast address lookup speed and reasonable forwarding table size.**

***Index Terms*—Address lookup, compression bit map, content addressable memory (CAM), multicast routing.**

I. Introduction

OVER THE PAST few years, the emergence of new applications with multipoint data transmissions, such as the resource discovery, multimedia conferences, or distance learning, was observed. These applications require "multicast routing"—a source sends an IP packet to a "group" so that the IP packet can reach all the members of that group without multiple "unicast routing".

There are several multicast routing protocols presented by the working groups in IETF [1], [2]. Some protocols are appropriate for the routing domain in an autonomous system (AS), while others are for inter-autonomous systems. Although these multicast routing algorithms operate with different methods in various routing environments, the *multicast address lookup scheme* should be the same. In this letter, we propose a fast multicast IP-routing lookup scheme, employing the *compression bit map* conception [3] onto which the relationship between multicast groups and sources is translated into a bit-stream, so does the output ports forwarding information. These translations can make the forwarding table size grow reasonably when the number of multicast groups or the number of multicast sources increases. This is because the scaling factor of table size is mainly proportional to the number of groups and the number of sources in bit-level. This bit-based operation is beneficial for simple hardware implementation. In the meanwhile, the number of memory accesses is always three, thus providing a feasibility of pipeline hardware architecture and obtaining faster address lookup speed.

II. Multicast IP-Routing Lookup Scheme

In the multicast IP-routing lookup scheme, it does not perform the multicast forwarding by several unicast forwardings where the *longest prefix matching* search is adopted. Every multicast source (or sender) needs to be aware of the identity (ID) of the multicast group. A multicast group ID is a multicast IP address whose first 4 bits are 1110 (Class D IP address). When a multicast source wants to send packets to the members of the group, it fills the destination address field of the packets with the multicast IP address. The intermediate routers are able to copy and forward the multicast packets by checking the multicast IP address without knowing all IP addresses of the group members, and the router will perform the *exact matching* search on the destination multicast IP address, instead of the *longest prefix matching* search. We can regard it as that the forwarding information of the route prefix carries the destination multicast IP address with prefix length equals to 32.

Furthermore, the multicast address lookup depends on not only the destination multicast IP address but also the source IP address. Multicast routing protocols, such as distance vector multicast routing protocol (DVMRP) or multicast open shortest path first (MOSPF), store each forwarding information based on the format of (group, source) pair associated with several output interfaces. The reason is that even belonging to the same multicast group, different sources may span different *multicast shortest path trees* for this group. So the forwarding information on a router for each (group, source) pair of the same multicast group would not be the same. The router needs to check over the destination IP address and the source IP address at the same time to uniquely identify the forwarding information. In order to support those above mentioned real-time services, the multicast IP-routing lookup in a router based on the multicast IP address should have two fundamental requirements: fast address lookup speed and small forwarding table size.

The design of the proposed fast multicast IP-routing lookup scheme is described by using a simple example in the following. Assume that a router has forwarding information records containing three multicast groups: G_1, G_2, G_3 and eight multicast sources: H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8. Each G_x ($1 \leqq x \leqq 3$) is a multicast IP address, while each H_y ($1 \leqq y \leqq 8$) is a unicast IP address. The group has members of sources shown below:

Group G_1: sources H_1, H_4, H_6, H_7, H_8,
Group G_2: source H_3,
Group G_3: sources H_2, H_5, H_8.

The existence of the membership for these (group, source) pairs can be represented by the cross points of a grid. As shown in Fig. 1(a), a group indicates the row index and a source decides the column index. A black point in the cross position indicates the existence of the corresponding (group, source) pair while a white point indicates the contrary case. Via the indexes of row and column, we can easily locate the cross position of the corresponding pair.

Manuscript received September 18, 2002. The associate editor coordinating the review of this letter and approving it for publication was Prof. D. W. Petr.This work was supported by the National Science Council, Taiwan, R.O.C., under Contract NSC 90-2213-E-009-051 and by Lee and MTI Center for Networking Research at National Chiao Tung University, Taiwan, R.O.C., under Grant Q.528.

K.-M. Huang is with the Computer and Communication Research Laboratory, Industrial Technology Research Institute, Taiwan 310, R.O.C. (e-mail: Lucien@itri.org.tw).

C.-J. Chang is with the Department of Communication Engineering, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: cjchang@cc.nctu.edu.tw).

Digital Object Identifier 10.1109/LCOMM.2003.809988

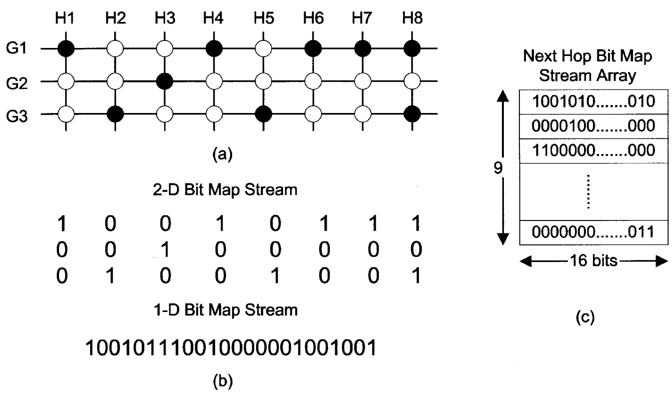

Fig. 1. (a) The grid to represent the existence information of the (group, source) pairs, (b) The pair bit map streams, (c) The next hop bit map stream array.

Translate the black point by bit '1' and the white point by bit '0', which forms a *two-dimensional bit map stream*, and arrange these bits to become a *one-dimensional bit map stream* by combining the seriated bit streams of rows from top to down together, as shown in Fig. 1(b). On the other hand, adopt the same translation conception on the multiple output ports information. For example, if the output ports information for a (group, source) pair includes port 1, 4, 8, 15, and the total output port number of the router is 16, the multiple output ports information can be converted into a *next hop bit map stream* with 16-bit width expressed as 1 001 000 100 000 010. The first, fourth, eighth and fifteenth bits are marked by bit '1' to indicate that they are the output ports where the packet should be forwarded. Next, we arrange these *next hop bit map streams* associated with these (group, source) pairs by left to right order in the *one-dimensional bit map stream* to form a *next hop bit map stream array*. As shown in Fig. 1(c), the *next hop bit map stream array* totally has nine forwarding information of (group, source) pairs, which were arranged as the order of (G_1, H_1), (G_1, H_4), (G_1, H_6), $(G_1, H_7$, (G_1, H_8), (G_2, H_3), (G_3, H_2), (G_3, H_5), (G_3, H_8). This output port bit map technique can be used in the design of layer-2 switch fabric because of its simple hardware implementation on the operation of bit comparison.

For an incoming multicast packet, the router uniquely locates the cross position in the grid by the destination and source IP addresses. Two content addressable memories (CAM) can be adopted to do the *exact matching* search on the destination and source IP addresses. The result after the *exact matching* search on the destination IP address is the value of x, while that on the source address is the value of y. The pair (x, y) indicates the cross position in the grid. If a nonmatching case exists in either one CAM, it indicates that the forwarding information of the (group, source) pair does not exist and the network processor must be triggered to compute the forwarding information for this pair. We adopt the CAM method without any special reasons except the fast search speed of $O(1)$. Then the cross position of the pair (x, y) can be transformed to be the bit position in the *one-dimensional bit map stream* by a simple calculation: $(x-1) \times Y + y$, where Y is the total number of sources.

Next, the bit value in the located position is checked. If the bit in that position is bit '0', it means that the forwarding information for this (group, source) pair does not exist and will trigger the network processor to compute the forwarding information for this pair. If the bit in the position is bit '1', then we count the number of bit '1' from the first bit to this bit in the *one-dimensional bit map stream*. The counting result is the array index of the entry, which will be used in the *next hop bit map stream* array. After reading out the corresponding *next hop bit map stream*, the address lookup operation completes.

Since there may have many multicast groups or sources, it takes too much time to count the number of bit '1' if the located bit position is far from the first bit. This problem can be solved by partitioning the *one-dimensional bit map stream* into several parts and appending each part with a field to record the accumulated number of bit '1' in previous parts. We call this field as pre-accumulated-value (PAV) and the partitioned parts as pair-bit-map (PBM). Consequently, the structure of the multicast forwarding table can be figured out as in Fig. 2, which includes the CAMs, PBM, PAV, and the *next hop bit map stream* array.

It can be seen that the size of the multicast forwarding table structure is dominated by the memory sizes of PBM and PAV. Assume that there are X multicast groups and Y multicast sources, and the number of bits in PBM and PAV is M and N, respectively. The table sizes needed for PBMs and PAVs can be computed by

$$\text{Total memory size of PBM} = \left\lceil \frac{X \times Y}{M} \right\rceil \times M \text{ (bits)} \quad (1)$$

$$\text{Total memory size of PAV} = \left\lceil \frac{X \times Y}{M} \right\rceil \times N \text{ (bits)} \quad (2)$$

This forwarding table size is scalable because the scaling factor, $X \times Y$, is on bit-level, especially for applications which contain a large number of multicast sources per group such as video conference.

It can be found from Fig. 2 that the number of memory access in the proposed multicast IP-routing lookup scheme is always three. With the current ASIC technology, one address lookup can be accomplished within time less than 40 ns. Therefore, the router can perform up to 25 mega-packets per second (pps). Moreover, because of the operation independence between each other, the hardware pipeline architecture can be considered and the faster lookup speed can be achieved.

We can form the PBM and PAV by simply implementing two-dimensional (2-D) link lists for multicast groups and sources in the network processor. Therefore the forwarding table can be constructed quickly without special construction algorithm. If there is any variation on the forwarding information for some (group, source) pairs, the router only needs to modify the corresponding *next hop bit map stream*. If there is any multicast source leaving the group (network deletion), it does not need to update the forwarding table right away because this obsolete information will not affect the accuracy of the multicast address lookup operation. Only when there is a multicast source joining a group (network addition) or re-joining another group (network updation), we must update the forwarding table. However, the network updation, at the column of multicast source in the 2-D bit map stream as in Fig. 1(b), is just to clear the bit in the row

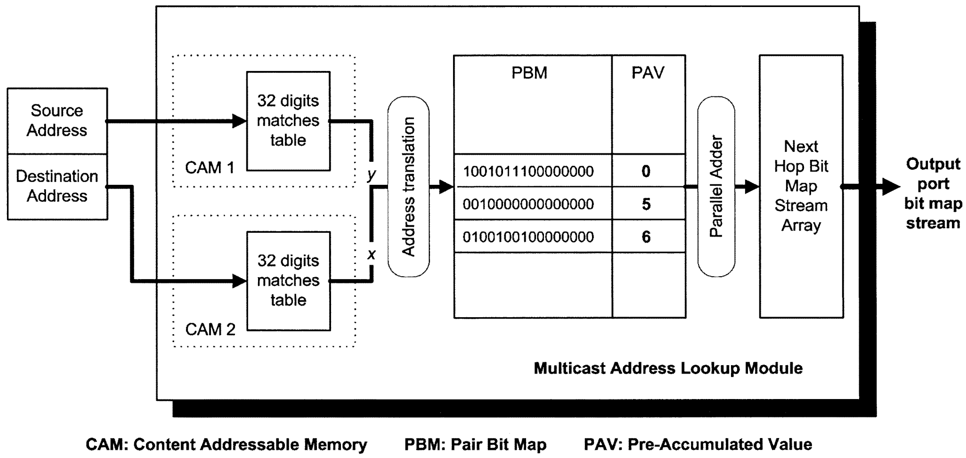

Fig. 2. The multicast forwarding table structure.

of leaving group and set the bit in the row of re-joining group. The network addition is to append a new column into the 2-D bit map stream. The above two hardware operations are simple and not necessary to rebuild the forwarding table but just a little reconfiguration. On the other hand, the frequency of changing on multicast groups has been suggested to occur as few as possible, and consequently the number of modifications for the forwarding table decreases.

III. Conclusion

This letter proposes a fast multicast IP-routing lookup scheme, with which the forwarding table size can scale well and an address lookup operation needs only three memory accesses. Moreover, when implemented in hardware pipeline architecture, the operation speed can be even up to one memory access per address lookup. The forwarding table can be constructed quickly without complex construction algorithm because of the straightforward compression bit map process, thus providing the simplicity to rebuild the forwarding table whenever necessary.

Acknowledgment

The authors thank the anonymous reviewers for their comments to improve the presentation of the paper.

References

[1] A. Adams, J. Nicholas, and W. Siadak, “Protocol independent multicast – Dense mode (PIM-DM): Protocol specification (Revised),” *Internet Draft*, Feb. 2002.

[2] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, “Protocol independent multicast – Sparse mode (PIM-SM): Protocol specification (Revised),” *Internet Draft*, Mar. 2002.

[3] N. F. Huang and S. M. Zhao, “A novel IP routing lookup scheme and hardware architecture for multi-gigabit switch routers,” *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1093–1104, June 1999.