DINloop Based Inter-domain Multicast with MPLS

H. Q. Guo  
ECE Department  
National University of  
Singapore  
Singapore 117576  
guohq@i2r.a-star.edu.sg

L. H. Ngoh  
Institute for Infocomm Research, A*STAR  
21 Heng Mui Keng Terrace  
Singapore 119613  
lhn@i2r.a-star.edu.sg

W. C. Wong  
Institute for Infocomm Research, A*STAR  
21 Heng Mui Keng Terrace  
Singapore 119613  
lwong@i2r.a-star.edu.sg

Abstract

To overcome the existing scalability problems, DINloop (Data-In-Network loop) based multicast with MPLS (multiprotocol label switching) is proposed to optimize inter-domain multicast. In our approach, multiple DIN Nodes in the core network form the DINloop using MPLS so that multiple multicast sessions share a single DINloop. We adopt a label stack method to use fewer labels and Label Switch Paths (LSPs) are established between DIN Nodes to create the DINloop. Packet Processing Module analyzes the coming multicast packet and MPLS Manager assigns a label stack to it. Simulations show that DINloop based multicast uses the least message load needed to form the multicast architecture. In addition, the routing table size in the core router does not increase as the number of multicast group increases, and therefore DINloop based multicast increases the routing scalability for inter-domain multicast. Finally, the multicast delay in DINloop based multicast is moderate larger than other protocols.

1. Introduction

IP multicast provides efficient one-to-many or many-to-many data distribution in an Internet environment, and also provides the functionality to logically group and identify a set of hosts. However, the current infrastructure for global, Internet-wide multicast routing faces some problems.

The existing multicast routing mechanisms broadcast some information and therefore do not scale well to groups that span the Internet [1]. Multicast routing protocols like DVMRP (Distance Vector Multicast Routing Protocol) [2] and PIM-DM (Protocol Independent Multicast – Dense Mode) [3], periodically flood data packets throughout the network. MOPSF (Multicast Open Shortest Path First) [4] floods group membership information to all routers so that they can build multicast distribution trees. Protocols like PIM-SM (Protocol Independent Multicast – Sparse Mode) [5] and CBT (Core Based Tree) [6] scale better by having the members explicitly join a multicast distribution tree rooted as a Rendezvous Points (RP). However, PIM-SM is an intra-domain multicast protocol and RPs in other domains have no way of knowing about sources located in other domains [7, 8]. CBT builds a bidirectional tree rooted at a core router and the use of single rendezvous point can potentially be subjected to overloading and single-point of failure. Thus, DVMRP, PIM-DM, PIM-SM, MOSPF and CBT are mainly used for intra-domain multicast.

Multicast Source Discovery Protocol (MSDP) [9] and Border Gateway Multicast Protocol (BGMP) [10] were developed for inter-domain multicast. BGMP requires Multicast Address-Set Claim protocol to form the basis for a hierarchical address allocation architecture. However, this is not supported by the present structure of non-hierarchical address allocation architecture and it is not suitable for dynamic setup. MSDP requires a multicast router to keep forwarding state for every multicast tree passing through it and the number of forwarding states grows with the number of groups [11,
12, 13]. In addition, MSDP floods source information periodically to all other RPs on the Internet using TCP links between RPs [14]. Thus, the current inter-domain multicast protocols suffer from scalability problems.

Significant research effort has focused on the multicast scalability problem. [15] uses network-transparent multicast to completely eliminate multicast state at routers, and thus pushes the complexity to the end-points. Other architectures aim to eliminate forwarding states at routers either completely by explicitly encoding the list of destinations in the data packets, instead of using a multicast address [16] or partially by using branching node routers in the multicast tree [17]. In Aggregated multicast [11], multiple multicast groups share one aggregated tree to reduce forwarding state, and a centralized tree manager is introduced to handle aggregated tree management and matching between multicast groups and aggregated trees. Aggregated multicast is targeted for intra-domain multicast and the centralized tree manager is a weakness.

MPLS [18] presents an efficient solution for multicast scalability problem [11, 12, 13]. To overcome the scalability problems in inter-domain multicast and centralized weakness posed by Aggregation multicast and MPLS Multicast Tree (MMT) [12], DINloop based multicast with MPLS is proposed to optimize inter-domain multicast. In our approach, the core router called as DIN Node, which connects to each domain, is chosen as the RP for that domain. Within a domain, the multicast tree is formed similarly to bidirectional PIM-SM, rooted at an associated DIN Node. In the core network, multiple DIN Nodes form a DINloop using MPLS for inter-domain multicast. The remainder of this paper is organized as follows. The related works are described in Section 2. Our solution overview is outlined in Section 3. Section 4 presents the details of DINloop based multicast. We present experimental results in Section 5 followed by the conclusion in Section 6.

2. Related works

Overview of IP Multicast in a Multi-Protocol Label Switching (MPLS) Environment is proposed in [19]. [20]
explains how to use PIM to distribute MPLS labels for multicast routes. Edge Router Multicasting (ERM) [21] limits branching point of multicast delivery tree to only the edges of MPLS domains. ERM is related to MPLS traffic engineering and it is an intra-domain routing scheme.

MMT [12] utilizes MPLS LSPs between multicast tree branching node routers in order to reduce forwarding states and enhance scalability. In MMT, each domain should contain a network information manager system (NIMS) to collect join messages from all group members and have a complete overview about the multicast network. For inter-domain, the border router contacts border routers in other domains with a normal (S, G) join message. These border routers contact NIMS routers in their domain. Therefore, the centralized NIMS is a weakness too and the number of forwarding states in inter-domain routers grows with the number of groups.

To demonstrate the effectiveness of DINloop, we have used video as an application example of DINloop and present a new video distribution service [22]. The looping has been implemented in the network layer to achieve data persistence within the network [23]. In addition, the DINloop has also been implemented in the application-level within a P2P overlay network and results in a hop efficient dynamic multicast infrastructure [24]. The DINloop in [24] is suitable to optimize a single application level multicast tree. In this paper, DINloop based multicast extends the DINloop concept to inter-domain network layer multicast using MPLS.

3. Solution overview

In DINloop based multicast, the idea is that, in order to reduce multicast forwarding state, instead of constructing a tree for each individual multicast session in the core network, one can have multiple multicast sessions share a single DINloop formed by multiple DIN Nodes. The multicast packet is fast forwarded in the core network using label rather than address matching to determine the next hop. Compared with other inter-domain multicast protocols, our solution shows that DINloop based multicast has below distinct advantages:
• In inter-domain, multiple multicast sessions sharing a single DINloop reduces multicast state, and correspondingly reduces tree maintenance overhead.

• DINloop based multicast consumes fewer labels to improve routing scalability, as well as reduce the routing look-up time for fast routing.

• DINloop based multicast architecture has the fast setup with least control message load for many-to-many multicast crossing multiple domains.

The details of our solution are shown in next.

4. DINloop based multicast

An inter-domain multicast flow typically starts from its source, travels an access network into a core network, and finally enters another access network to reach the receivers. Our scheme uses MPLS to set up the DINloop in the core network by the way that LSPs are established between DIN Nodes. One example of DINloop based multicast shown in Fig. 1. DIN Nodes A, B, C, D and E are the core routers and function as RPs for the associated intra-domains (e.g., circle areas) respectively. DIN Nodes A, B, C, D and E form the DINloop (thick arrow line in Fig. 1) in the core network.

4.1. Label

To avoid a large size for the routing table and gain the speed in table look-ups, we adopt a method of label stack. We assign all multicast groups a same top label in a label stack and a different bottom label in the label stack used to differentiate the multicast messages. The label stack is organized as last-in, first-out. The forwarding is always based on the top label. The purpose of the label stack is to improve routing scalability by reducing the amount of information that has to be handled by OSPF and/or IS-IS.

The advantages of our approach include:
- Fewer labels for routing so as to reduce the routing look-up time.
- Easy to retrieve data from the DINloop since when the top label is popped off, the bottom label is sufficient to unambiguously identify the multicast packet.
- Also reduce the amount of label distribution control traffic needed.

The decision to bind a particular label stack to a particular Forwarding Equivalence Class (FEC), which is typically a destination address for multicast, is made by the DIN Node. Then the DIN Node informs the neighbour DIN Node in the DINloop about label/FEC binding that it has made. The adjacent DIN Nodes use LDP to exchange label/FEC binding.
4.2. Assign labels to multicast messages

When a multicast packet arrives in the ingress DIN Node, Packet Processing Module is activated. Packet Processing Module looks the IP header of the multicast packet and identifies the source address and destination address (D class for multicast). Then Packet Processing Module reports the information to MPLS Manager. MPLS Manager determines the packet’s FEC and looks up the Label Table to assign a label stack with two level labels. The top label corresponding to the inter-domain multicast is same for all multicast groups and the bottom label corresponding to the destination address differentiates the multicast messages. In order to transmit the label stack along with the packet, the label stack is encoded as a “shim” between the data link layer and network layer headers. The DIN Node also defines the lifetime of packet to circuit in the DINloop and adds this information in the header. When the lifetime of packet is expired, the packet is moved from the DINloop.

4.3. Forward multicast message for inter-domain

The Next Hop Label Forwarding Entry (NHLFE) is used when forwarding a labelled packet. It contains the following information:

1. the packet’s next hop that towards the next DIN Node in the DINloop
2. the operation to perform on the packet’s label stack that replace the label at the top of the label stack with a specified new label that is meaningful locally.

For DINloop based multicast, an explicitly routed LSP is used that each DIN Node chooses the adjacent DIN Node in the DINloop as the next hop and put the information in the NHLFE. Fig. 3 shows the LSP for DINloop. When forwarding the labelled packet, the DIN Node examines the label at the top of the label stack and uses Incoming Label Map (ILM) to map this label to an NHLFE. Using the information in the NHLFE, it determines to forward the packet to next DIN Node, and performs an operation on the packet’s label stack. It then encodes the new label stack into the packet, and forwards the packet.

If two adjacent DIN Nodes in the DINloop are not the adjacent LSRs, then LSP tunnel is built up between adjacent DIN Nodes.

4.4. Retrieve multicast message

DINloop in the core network allows DIN Nodes to share information about active sources. Each DIN Node knows about the receivers in their local domain respectively. For example in Fig. 4, when DIN Node B in remote domains receives the join message for multicast group G from Receiver H_b, it posts the join message in the DINloop. When DIN Node A in the source domain receives this join message and knows that there is the receiver in the remote domain, DIN Node A will forward the multicast message into the DINloop. When DIN Node B receives the packets, it pops the top label off the label stack and uses the bottom label to examine the multicast packet. If it finds the required packet, DIN Node B duplicates the packet and forwards one copy to next DIN Node and one copy to its local receiver H_b. Similarly, Receiver H_c1, H_c2 and H_d in other domain can also join the multicast group G and receive the multicast information from Source S.

5. Experimental results

This section comprises three parts to compare with the conventional inter-domain multicast protocols. First, we describe the simulations to demonstrate that DINloop based multicast reduces the message load needed to form the multicast architecture. Second, we compare the
forwarding state size kept in core routers. Finally, we describe the performance in multicast delay.

Fig. 4. Retrieve multicast message

5.1. Message load

We use the message load to evaluate the performance of DINloop based multicast versus conventional inter-domain multicast protocols, i.e., MSDP and BGMP. The metric of message means the signaling message that is needed to form the multicast architecture. The message load is counted in this way that when a signaling message passes one link, the number is added by one.

In MSDP, the RP in each domain establishes an MSDP peering session using a TCP connection with the RPs in other domains. MSDP floods source information periodically to all other RPs using TCP links between RPs. MSDP also allows receivers to switch to shortest-path tree to receive packets from sources and we refer it as MSDP+source. BGMP builds bidirectional shared tree rooted at the root domain and is referred as BGMP+shared. In addition, BGMP also allows source-specific branches to be built by the domains. We refer to the tree built by BGMP consisting of the bidirectional tree and the source-specific branches as BGMP+source.

The simulations ran on the network topologies, which were generated using the Georgia Tech [25] random graph generator according to the transit-stub model [26].

We used the graph generator to generate different network topologies. The number of nodes in transit domain was fixed, i.e., 25 that means 25 DIN Nodes for 25 domains. The number of nodes in stub domains was varied from 500 to 6000 that spread in 25 domains. The number of sources is 25 and one domain has one source. Each transit node is the RP for its associated stub domain respectively. The results are shown in Fig. 5.

From Fig. 5, it is clear that DINloop based multicast uses the least message load to form the multicast architecture. The message load in MSDP is the largest and the message load in BGMP is in between those of DINloop based multicast and MSDP. As the node number increases, the message load increases linearly for all protocols.

Fig. 5. Message load comparison on the effect of node number

We then evaluate the effect of the source number. In Fig. 6, the number of nodes in transit domain was also fixed, i.e., 25 and the number of nodes in stub domains was fixed to 1000 that spread in 25 domains. The number of sources is varied from 2 to 25 and the sources spread in 25 domains.

In Fig. 6, the message loads in MSDP and BGMP increase linearly as the source number increases. The slope in MSDP is higher than BGMP. However, the message load in DINloop based multicast is not affected by the source number since all sources share the same DINloop.

From the above simulations, we can obtain the conclusion that the message load in DINloop based multicast is the smallest, and correspondingly reduces tree maintenance overhead at core network.
5.2. Router forwarding state size

In this sub-section, forwarding state size kept in core routers is compared with conventional multicast protocols. The current inter-domain multicast protocols require a core router to keep forwarding state for every multicast tree passing through it. We refer the current inter-domain multicast protocols as non-DINloop scheme.

In non-DINloop scheme, the core router looks up IP header of every packet and identifies the destination address. Then, the core router looks up the routing table and forwards the packet to the correct interface (Fig. 7). Therefore, the routing table size is increased linearly with the number of multicast groups.

From Fig. 8, we obtain the results that the routing table size does not increase as the number of multicast group increases, and therefore the DINloop based multicast increases the routing scalability for inter-domain multicast, as well as reduces the routing look-up time for fast routing.

We also want to point out that DINloop based multicast pays the price for the high routing scalability. That is DINloop based multicast adds the additional MPLS header to the multicast packets.

5.3. Multicast delay

In this sub-section, we compare the multicast delay on the different types of inter-domain multicast architectures constructed by above protocols, i.e., unidirectional shared tree, bidirectional shared tree, and DINloop-based multicast. Multicast delay is from sources sending out data to receivers receiving the data. We used the graph generator to generate different network topologies. The number of nodes in transit domain was varied, i.e., from 1 to 20 that means domain number
changes from 1 to 20. The number of nodes in stub domains was fixed as 6000 that spread in all domains. The legend in Fig. 9 is explained as below.

\[ \text{Unidirectional Tree (max)} = \frac{\text{Maximum Delay in Unidirectional Tree}}{\text{Maximum Delay in Bidirectional Tree}} \]

\[ \text{Unidirectional Tree (ave)} = \frac{\text{Average Delay in Unidirectional Tree}}{\text{Average Delay in Bidirectional Tree}} \]

\[ \text{DINloop (max)} = \frac{\text{Maximum Delay in DINloop Based Multicast}}{\text{Maximum Delay in Bidirectional Tree}} \]

\[ \text{DINloop (ave)} = \frac{\text{Average Delay in DINloop Based Multicast}}{\text{Average Delay in Bidirectional Tree}} \]

Fig. 9. Delay ratio comparison on the effect of domain number

From Fig. 9, the delay in DINloop based multicast is close to the delay in bidirectional tree and is lower than that in the unidirectional tree when the domain number is small. When the domain number is higher than 7, the delay in DINloop based multicast is highest. The reason is that because of random network topologies, the delay in the DINloop is high if the topology is not a ring while the other multicast protocols benefit from the cut through. If the core network is the ring topology, then the delay in DINloop is close to others. Although the delay in DINloop based multicast is high when the domain number is large, the average delay ratio is less than 2.4 and the maximum delay ratio is less than 3.4 when the domain number is less than 20. Therefore, the delay in DINloop based multicast can be accepted.

6. Conclusion

Current inter-domain multicast protocols suffer from a number of scalability problems. To overcome the existing problems, DINloop based multicast with MPLS is presented to optimize inter-domain multicast. In the core network, multiple DIN Nodes form the DINloop using MPLS for inter-domain multicast. Instead of constructing a tree for each individual multicast session in the core network, multiple multicast sessions share a single DINloop. DINloop based multicast uses MPLS to set up the DINloop in the core network and adopts label stack method to use fewer labels. LSPs are established between DIN Nodes to create the DINloop. When a multicast packet arrives to the ingress DIN Node of core network, Packet Processing Module analyzes the packet and MPLS Manager determines the packet’s FEC and looks up the Label Table to assign a label stack with two level labels. The top label corresponding to the inter-domain multicast is same for all multicast groups and the bottom label corresponding to the destination address differentiates the multicast messages. Then the multicast packet is fast forwarded in the core network using label rather than address matching to determine the next hop. The egress DIN Node pops the top label off the label stack and uses the bottom label to locate the multicast data and forward it to its associated receivers.

There are some results based on simulations and analyses. First, simulations show that DINloop based multicast uses the least message load needed to form the multicast architecture compared with the conventional inter-domain multicast protocols. Second, the routing table size in the core router does not increase as the number of multicast group increases, and therefore the DINloop based multicast increases the routing scalability for inter-domain multicast, as well as, reduces the routing look-up time for fast routing, at the price of adding additional MPLS header to the multicast packets. Finally, the multicast delay in DINloop based multicast is close to that in bidirectional tree and is lower than that in the
unidirectional tree when the domain number is small while the multicast delay in DINloop based multicast is high when the domain number is large. However, the delay ratio is small, and therefore the multicast delay in DINloop based multicast can be accepted.

In the future work, we will evaluate the performance of DINloop when it is used as a network cache.

Reference


