A Hierarchical Multicast Routing Protocol

Ian F. Akyildiz and W. Yen

School of Electrical and Computer Engineering
Georgia Institute of Technology; Atlanta, GA 30332
Tel: 404-894-5141; Fax: 404-853-9410
E-mail: ian@armani.gatech.edu; wei@eecom.gatech.edu

1 Introdction

Multimedia applications will become more and more important and popular in the future networks [5]. Many multimedia applications including electronic newspaper and remote education require point to multipoint communication instead of point to point communication. The point to multipoint communication is refered as multicast in this paper. The problem of multicast routing is considered as a graph theoretical optimal delivery tree problem. Several highly complex multicast routing algorithms have been proposed along these lines in recent years. These algorithms require intensive computation and message exchanges; thus they are unsuitable for practical protocol implementation. Existing multicast routing protocols were designed to trade optimality for time efficiency [2, 3, 6]. Still, the existing protocols have either scalability or performance problems when applied in the wide area network environment [1, 4]. In this paper, a hierarchical multicast routing (HMR) algorithm is proposed which uses the concepts of clusters and local cores to address these problems. This algorithm is simple, yet flexible enough to provide various routing modes to satisfy different users' delay and scalability requirements. A set of protocols is designed to implement the algorithm. These protocols (a) are independent of the underlying unicast protocol; (b) exhibit low complexity and good scalability; (c) provide the upper bound for the delay performance; and (d) support dynamic membership change. Through simulations, the performance of the new protocol is compared to that of existing protocols and its advantages are demonstrated.

2 The Hierarchical Multicast Routing Algorithm

The hierarchical multicast routing (HMR) algorithm constructs delivery trees connecting a multicast group. Member and non-member sources can use these trees to send multicast packets to the group. A network can be modeled as a connected graph (V,E), where V is the set of nodes and E is the set of edges. The routers of the network are nodes in V and the links between routers are edges in E. We assume that each node is labeled by its IP address and each edge is bi-directional and symmetric. The delay between two nodes is defined as the number of edges, i.e., the number of hops, on the path connecting the nodes. A multicast group consists of a set of destinations which may be in different routing domains. Each destination is attached to its designated router which is responsible for communication with other routing domains. The designated routers of these destinations are elements in the set V and we call them the members of the group. The principle of the HMR algorithm can be explained as in what follows.

Given a multicast group, we decompose its members into several mutually exclusive subsets, i.e., the intersection of any two subsets is a null set, called *clusters.* Basically, members 'close' to each other are assumed to be in the same cluster. Moreover, a local core is selected for each cluster among its members. We refer to the number of members in a cluster as *the size of that cluster.* Another important parameter used to characterize the clusters is *the order of the clusters* which is the maximum delay between the local core and members of its cluster. After decomposing the multicast group into several clusters, two shortest path trees (SPT's) rooted at the local core are established for each cluster. The first SPT, called *the intra-cluster SPT*, spans the members within the cluster. The second SPT called *the inter-cluster SPT* spans all the other local cores.

2.1 Procedures of the HMR Algorithm

The HMR algorithm generates clusters and their local cores. First, we need to determine the neighbor set for each member in the multicast group. Suppose a particular multicast group selects o as *the order of the clusters.* The selection of o is determined based on the multicast group's preference. The neighbor set of a member, i, is then defined as a set of all members which are less than or equal to o edges away from i. Hence, the minimum delay between i and any element in the neighbor set is less than or equal to o. We denote n_i as the neighbor set of member i. In this case, member i is called the *center* of that neighbor set. After determining the neighbor set, each member i sends its neighbor set n_i to all the other members in the group. Let now N be the set of all n_i. Then, upon receiving N, each member can determine the clusters and local cores due to the following steps:

1. Assume the maximum size among all the neighbor sets to be s_{max}. Select neighbor sets that have size s_{max}.

2. If two or more neighbor sets are selected, compare the IP addresses of the centers of these neighbor sets. Select the neighbor set that has center with largest IP address.

3. Assign the selected neighbor set to be a cluster, denoted by C_i, and the center of the neighbor set to be the local core of the cluster, denoted by c_i.

4. Remove the neighbor sets that have centers which belong to the cluster.

5. Remove the members of the already selected cluster from other neighbor sets.

6. Repeat the above steps until all clusters and local cores are determined.

3 The Hierarchical Multicast Routing Protocol

We introduce three protocols to implement the HMR algorithm: i) the protocol for determining the neighbor sets and ii) the tree construction protocol. The three protocols are briefly explained next. In the first protocol, the group members broadcast the so-called 'probe' messages in order to determine their neighbor sets. The tree construction protocol are used to set up routing information at the routers. Note that we assume that there already exist independent protocols which deal with the group initiation and multicast group address assignment. We also assume that each member in the group knows all addresses of the members in the group and the assigned group address. However, it is not necessary for each member to know the topology of the mulitcast group, i.e., the shortest paths among members. In addition to these assumptions, the reliable transmissions of the routing messages are also required.

3.1 The Protocol for Determining the Neighbor Sets

The protocol for determining the neighbor sets is implemented at routers. Each member broadcasts a 'probe' message to the network with a data field, denoted by f. The initial value of f is set to the order of the clusters, o. The order of the clusters is pre-defined for some services, i.e., all multicast groups demand a certain service must use a pre-defined o which is selected off-line by taking into account the network efficiency and the group's preference for this service. Or, it can be determined by some negotiation process among group members to characterize the need, such as delay, bandwidth, and scalability, of the multicast group. In general, we would like to have a large o to achieve better scalability. However, in some applications, the maximum delay is contrained, i.e., it can not exceed a certain upper bound. In this case, the largest o which can satisfy the delay contraint might be the best choice. Note that if a multicast group chooses the single cluster topology, then instead of infinity, we assign any value which is not less than the maximum delay between the nodes in the network to o. Upon receiving the probe message, each router will take four steps:

1. It checks if its address is contained in the probe message. If not, then it appends its address to the probe message. Otherwise, the router ignores the probe message.[1]

2. It substracts f by one.

3. It checks if there is any group member in its attached network. If yes, the router sends the probe message to the members in its attached network.

4. It checks the data field f. If f is greater than zero, then the router broadcasts the probe message to all its outgoing interfaces except the one leading to the source. However, if f is equal to zero, the router will not broadcast the probe message.

The probe messages help members to decide on their neighbor sets. If a member i receives a probe message from member j, then i knows that j is in its own neighbor set. In other words, all members whose probe messages are received by i belong to the neighbor set n_i and the member i

[1] This step prevents the probe message from being passed through a router twice, i.e., cycles do not occur.

is the center of n_i. Since the probe message contains the appended addresses of the intermediate routers, member i can obtain the shortest path from the probe message. Moreover, the data field f contained in the probe message can be used to calculate the minimum delay $d(i,j) = o - f$. In conclusion, the center of a neighbor set knows the shortest paths and the minimum delays between itself and all other members in the neighbor sets. These informations will benefit the tree construction and the local core adjustment protocols later on.

3.2 The Tree Construction Protocol

This protocol provides the construction of two types of shortest path trees, the intra-cluster SPT and the inter-cluster SPT. An intra-cluster SPT rooted at a local core spans all the other members in the same cluster. In the first protocol given in Section 3.1, the local core obtains the shortest paths between itself and all other members in the same cluster from the probe messages it receives. Therefore, the local core can use these informations about of the shortest paths to setup routing tables at the routers on the intra-cluster SPT. These routers, then, transmit multicast packets according to the source address contained in the headers of these packets. If the address is that of the local core, then the routers multicast the packets downstream to the members in their child or attached networks; otherwise, the routers send the packets to the local core via the shortest path. In other words, the routing information stored at the routers on the intra-cluster SPT is very small. For routers on the inter-cluster SPT's, explicit joining messages need to be sent among local clusters to setup the routing paths. Moreover, the clusters need to send 'refresh' messages periodically to prevent being pruned from the inter-cluster SPT's. Each local core takes two steps when it receives a multicast packet.

1. The specific local core checks the source address of the multicast packet. If the source address is one of the other local cores, then it multicasts the packet to all members in its own cluster. If the source address does not belong to any other local core, then it multicasts the packets to all other local cores and to all members in its own cluster.

2. Before multicasting the packet, the specific local core replaces the source address in the multicast packet by its own address.

References

[1] T. Ballardie, P. Francis, and J. Crowcroft,"Core Based Tree (CBT)," *Proc. of ACM Sigcomm '93 Conference*, San Francisco, CA, pp. 85-95, Sept. 1993.

[2] S. Deering, "Host Extensions for IP Multicasting," Internet RFC 1112, Aug. 1989.

[3] S. E. Deering and D. R. Cheriton,"Multicast Routing in Datagram Internetworks and Extended LANs," *ACM Transactions on Computer Systems*, Vol. 8, No. 2, May 1990.

[4] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, and L. Wei,"An Architecture for Wide-Area Multicast Routing," *Proc. of ACM Sigcom '94 Conference*, London, pp. 126-135, Aug. 1994.

[5] R. Frederick, "Ietf Audio & Videocast," *Internet Society News*, Vol. 1, No.4, pp. 19, 1993.

[6] J. Moy, "MOSPF: Analysis and Experience", *Internet Draft*, July 1993.