INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst., **11**, 93–101 (1998)

GMNF-DVMRP: AN ENHANCED VERSION OF DISTANCE VECTOR MULTICAST ROUTING PROTOCOL

YUAN-CHENG LAI,* YING-DAR LIN AND WEI-CHE YU
Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan

SUMMARY

In this paper we propose an enhancement to the Distance Vector Multicast Routing Protocol (DVMRP), called 'Group Membership Near First DVMRP' (GMNF-DVMRP), to decrease the network cost (NC) of the multicast tree formed by DVMRP. A simulation is implemented to compare our enhanced version and the original DVMRP. We find that our method saves about 5%–7% of NC of the multicast tree formed by DVMRP. © 1998 John Wiley & Sons, Ltd.

KEY WORDS: multicast; multicast routing protocol; network cost; DVMRP

1. INTRODUCTION

A multicast routing protocol is a set of standards and parameters that two end-points of communication agree upon to route packets. The Internet Group Management Protocol (IGMP)[1] is used to carry messages between the multicast router and the end-host or between multicast routers themselves. A variety of multicast routing schemes have been developed and used. They include Distance Vector Multicast Routing Protocol (DVMRP)[2], Core-Based Trees multicast protocol (CBT)[3] and Protocol Independent Multicast protocol (PIM) Dense Mode (DM) and Sparse Model (SM).[4] A simple comparison of these schemes is given in Table I. Here we focus on DVMRP because it has been implemented in the mrouted program.

Two optimal criteria are used to evaluate the efficiency of the multicast tree formed by a multicast routing protocol, namely network cost (NC) and destination cost (DC).[5] NC is a value that measures the utilization of network resources, such as total used bandwidth. DC is a value that measures the average delay experienced by each destination. Minimizing DC is simpler than minimizing NC, because finding the minimum NC multicast tree on a given network topology is an NP-complete problem–also called a minimal Steiner tree problem.[6]

The DC of the DVMRP multicast tree is optimal because it is built on the shortest paths from the source to each destination. However, the NC of the DVMRP multicast tree may be high or low, depending on the network topology and the distribution of group members. Shared medium networks, especially the Internet, have sensitive NCs. Redundant use of one hop on the multicast tree will cost much bandwidth, especially when the volume of multicast data is large. Therefore we should try to decrease the NCs of multicast trees formed by the multicast routing protocol. In this paper we develop a new version of DVMRP as GMNF-DVMRP to decrease NC.

We first describe DVMRP in Section 2 and then propose an enhanced version to improve DVMRP in Section 3. A simulation that compares DVMRP and our enhanced version is presented in Section 4. Finally, conclusions are given in Section 5.

2. OVERVIEW OF DVMRP

DVMRP is short for Distance Vector Multicast Routing Protocol. 'Distance Vector' means that it is derived from the distance vector routing protocol—the Routing Information Protocol.[7]

DVMRP is a distance vector style of routing protocol for routing multicast datagrams through the internetwork. The DVMRP-based multicast router maintains network topology knowledge by a distance vector algorithm—also called the Ford–Fulkerson[8] or Bellman–Ford[9] algorithm.

DVMRP implements multicast based on the Reverse Path Forwarding algorithm,[10] a method of performing broadcast delivery in a store-and-forward network of point-to-point links, using only the information present in the routing table maintained by the distance vector algorithm. However, Reverse Path Broadcast (RPB) does not really perform multicast but broadcast and it just works on networks of point-to-point links. Therefore DVMRP implements a multicast forwarding algorithm called Truncated Reverse Path Broadcast—a refinement of RPB.[11]

2.1. *Truncated reverse path broadcast* (*TRPB*)

Truncated Reverse Path Broadcast is described by Deering[11] and is a refinement of Reverse Path

*Correspondence to: Y.-C. Lai, Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan. Email: gis83802@cis.netu.edu.tw

CCC 1074–5351/98/010093–9 $17.50

Received April 1997
Revised October 1997

Table I. Simple comparison of multicast routing schemes

Protocol	Tree type	Group distribution	Protocol independence	State mechanism	Scalability
DVMRP	Source-based	Dense	No	Hard	Bad
CBT	Shared-based	Sparse	No	Hard	Good
PIM-DM	Source-based	Dense	Yes	Soft	Bad
PIM-SM	Mixed	Sparse	Yes	Soft	Good

Forwarding. It selects one *dominating multicast router* among all multicast routers connected on the same subnetwork. By comparing the ‘routing updates’ from neighbouring multicast routers with its own routing table, each multicast router can determine whether itself has the shortest distance to a particular source. The multicast router that has the shortest distance back to a particular source is selected as the dominating multicast router. If more than two multicast routers on the subnetwork have the same shortest distance, the multicast router with the lowest address wins. The dominating multicast router of a subnetwork is responsible for forwarding multicast packets onto this subnetwork.

Another functionality of TRPB is that it can determine whether the subnetwork is a *child* or a *leaf* of the dominating multicast router on the multicast tree of a particular source. Two additional fields, *child subnetwork list* and *leaf subnetwork list*, are added at every entry of the routing table to record whether the connected subnetwork is a child or a leaf on the multicast tree rooted at the destination of this entry. The procedure used by the TRPB-based multicast router to decide whether its connected subnetwork is a child or a leaf of the multicast tree rooted at the source is listed as Procedure 1,[11] where

R = TRPB-based multicast router,

r = route to a destination,

d = destination address,

D = distance to destination,

k = subnetworks connected to R,

n = neighbouring multicast routers of R.

Procedure 1. *Finding child and leaf subnetworks*

```
WHILE R receives ‘routing updates’ entry (d,
D, next-hop-address) from n
    IF (next-hop-address<>k) AND
        (NO n such that (n.subnet=k AND
        (n.distance[d]<D OR
        (n.distance[d]=D AND n.address <
        R.address))))
    THEN k is a child of R on the multicast tree
    rooted at d
    IF (NO n such that (n.subnet=k AND
    n.distance[d]=infinity))
        THEN k is a leaf of R on the multicast tree
        rooted at d
```

After the TRPB-based multicast router has determined which subnetwork is its child or its leaf, it can forward multicast packets by the procedure listed as Procedure 2.[11]

Procedure 2. *Sending copies of multicast packets to subordinating branches*

```
WHILE R receives a multicast packet from source
s to group g via subnetwork k
  IF (the routing table of R has an entry that
  (d=s.address AND next-hop-address=k))
    FOR each connected subnetwork of R, i or F
      IF (i is not a leaf of R on the multicast tree
          rooted at s) OR
          ((i is a leaf of R on the multicast tree
          rooted at s) AND
          (there is at least a member of group g
          on subnetwork i))
      THEN send a copy of multicast packets A
      onto subnetwork i
```

We look at the partial network topology illustrated in Figure 1 as an example. Three multicast routers, MR1, MR2 and MR3, are attached to subnetwork 1. If the shortest-path distances from MR1 and MR2 to host S are five and four hops respectively, then the dominating multicast router of S1 is MR2, so that S1 is the child of MR2 on the multicast tree rooted at host S. S1 is not a child of MR1. MR3 reports to S1 that its distance to the source is ‘infinity’, so that S1 is not a leaf. MR3 is the only multicast router on S2, so that S2 is a child of MR3. However, no multicast router uses S2 to reach the source. In other words, no multicast router reports to S2 that its distance to the sources is ‘infinity’. S2 is a leaf of MR3. The partial multicast tree rooted at the source is shown in Figure 1. We assume that host H3 is a group member.

2.2. *DVMRP multicast algorithm*

DVMRP routers forward multicast packets based on their source. Each (source, group) pair forms a different multicast tree rooted at the source. DVMRP routers have no previous information about the distribution of (source, group) pairs. When receiving

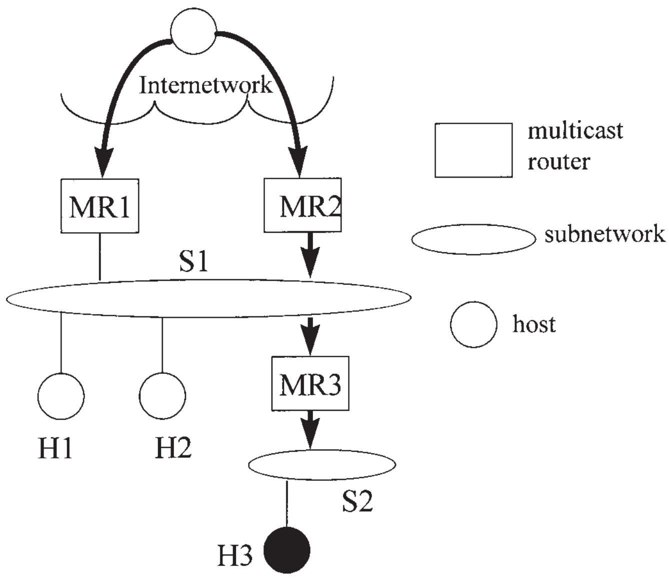

Figure 1. TRPB on multi-access subnetwork

a multicast packet sent from a particular sender to a particular group, DVMRP routers just deliver this multicast packet according to the Truncated Reverse Path Broadcast algorithm described in the previous subsection. However, the Truncated Reverse Path Broadcast algorithm only guarantees that multicast packets are broadcast on the shortest path from the source to all reachable destinations. Multicast packets may be delivered to some subnetworks that are not required to receive these packets. An example of a multicast tree formed by TRPB is shown in Figure 2. Multicast packets are delivered on the shortest path from the source to all reachable destinations. However, it is evident that subnetworks S5, S6 and S8 are not required to receive these packets, because no group member exists on these branches. Such multicasting is called *truncated multicasting*.

If there is no group member on a branch of the multicast tree, then no multicast packet should come over this branch. That is, this branch has to be pruned from the multicast tree.

A *prune message* for a particular (source, group) pair is first generated by the router which has no child subnetwork or which owns child subnetworks but all of them are leaf subnetworks without group members on them. The router sends this prune message back to its predecessor router on the multicast tree. If the predecessor router receives prune messages from all its subordinate routers, the predecessor route goes on sending the prune message back to its predecessor. Eventually the multicast tree will be pruned back so that all the leaf subnetworks of the pruned multicast tree have group members. Such multicasting is called *pruned multicasting* or *true multicasting*. The truncated and pruned multicast tree of the previous example is shown in Figure 3.

If a group member on the pruned subnetwork wants to join the group again, the dominating router of this subnetwork stops sending the prune message to its predecessor. Instead, the router sends a *graft message* to its predecessor to cancel its previous

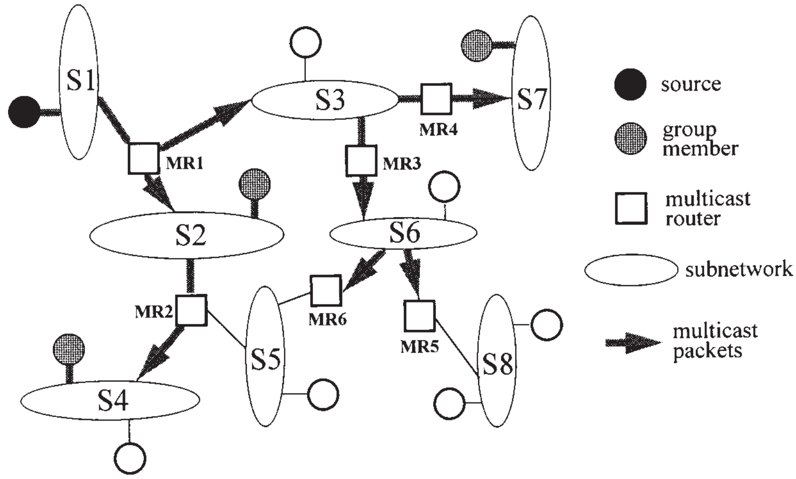

Figure 2. Example of truncated multicast tree formed by TRPB

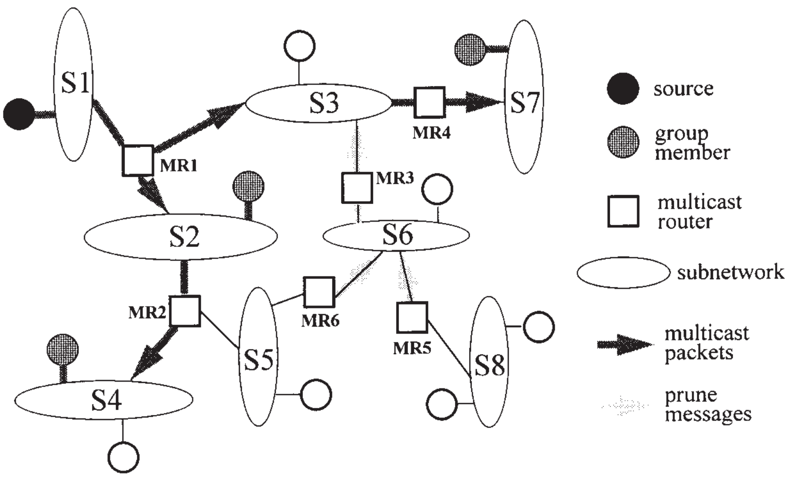

Figure 3. Example of truncated and pruned multicast tree

prune message. The graft message is propagated back to the source as far as the prune message is. After the predecessor receives the graft message from a subordinate, it will send a copy of multicast packets to this subordinate when multicast packets come. Consider the example of a truncated and pruned multicast tree illustrated in Figure 3. Assume that the host on subnetwork S6 wants to join the group. MR3, the dominating multicast router of S6, will send a graft message back to its predecessor MR1 and then MR1 will restart sending multicast packets to MR3, as shown in Figure 4.

3. ENHANCED VERSION OF DVMRP

3.1. *An example*

Consider an example of building a multicast tree of DVMRP on the network topology illustrated in Figure 5. Multicast packets are sent from the source to multicast router MR1. MR1, the designated multicast router of subnetworks S2 and S3, then delivers these multicast packets onto S2 and S3. Assume that the addresses of MR2 and MR3 are 140.113.136.110 and 140.113.136.111 respectively. MR2 becomes the designated multicast router of subnetwork S2 with a lower address than MR3, though they have the same distance of two hops to the source. As a result, MR2 is responsible for delivering these multicast packets onto S4 but MR3 is not. MR3 will send a prune message back to its predecessor MR1. We consider NC as the total hop number used by the multicast tree. Thus the NC of this multicast tree is four. If, however, MR3 wins to be the designated multicast router, MR3 will be responsible for delivering multicast packets onto S4 rather than sending a prune message back. On the contrary, MR2 will send a prune message to its predecessor MR1 and then MR1 will stop delivering multicast packets onto S2. The NC of the new multicast tree is three, resulting in one hop saving. Since MR2 and MR3 have the same distance to the source, the DC of the new multicast tree is equivalent to that of the previous multicast tree. The new multicast tree is shown in Figure 6.

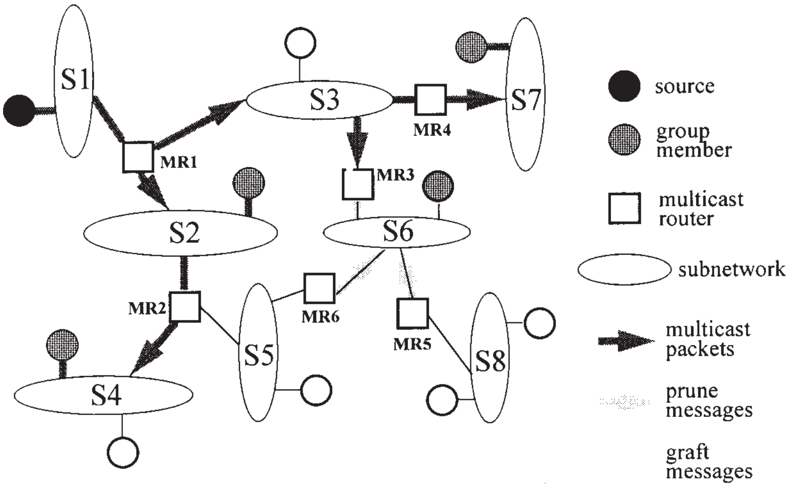

Figure 4. Example of truncated and pruned multicast tree with graft

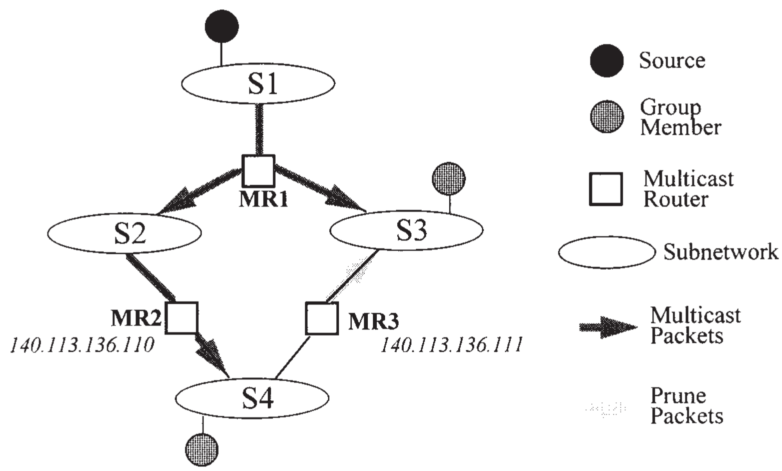

Figure 5. Example of building multicast tree of DVMRP where MR2 is designated router of S4

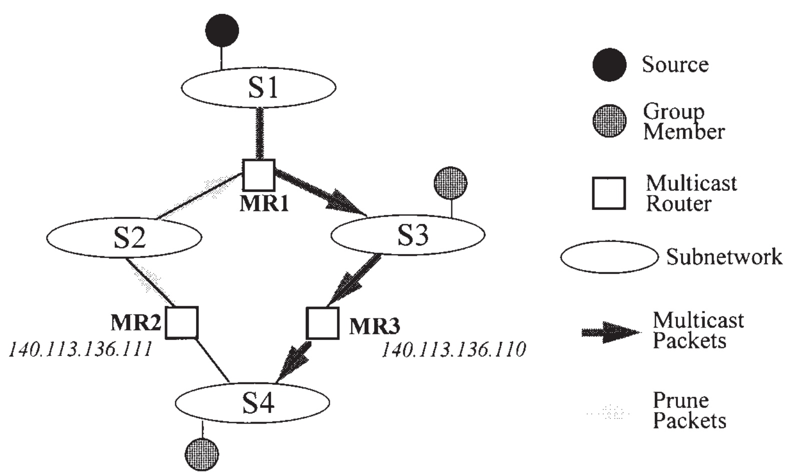

Figure 6. Example of building multicast tree of DVMRP where MR3 is designated router of S4

3.2. *Idea*

The idea of our enhanced version of DVMRP is simple. If two or more shortest paths exist between a source and its destination, choose the path where the first predecessor has one group member. Whenever a host joins a group, the interrogating multicast router of the host (MR_A) sends this group address to its neighbouring multicast routers (MR_B). Multicast router MR_B now knows that a member of such group address exists on the subnetwork of MR_A. From now on, whenever MR_B receives multicast packets, it checks whether the multicast address of these packets is the group address it received from its neighbouring multicast routers, e.g. MR_A. If it matches, then *GMN(source address)* of MR_B is set to be true, where *source address* is the source address of these multicast packets. *GMN(source address)* means that group membership of the sender with source address exists on the neighbouring subnetwork. Now MR_B has higher priority to become the designated multicast router under the situation that other multicast routers on the same subnetwork have the same shortest distance to this source. Consider the example shown in Figure 5 again. MR1, the interrogating multicast router of S3, sends a message to MR3 to announce that a member of a particular group address exists on S3. When MR3 receive multicast packets of such group address, it retrieves the source address from the header of the first packets and sets *GMN(source address)* to be true. Now MR3 has higher priority to become the designated multicast router for forwarding multicast packets from that source even though its address is not the lowest.

3.3. *Algorithm*

We denote our enhanced version of DVMRP as GMNF-DVMRP. GMNF is short for 'Group Membership Near First'. It means that if there is a member on the neighbouring network, the algorithm should consider the router attached to this neighbouring subnetwork first to become the designated multicast router of its subnetwork.

Each interrogating multicast router of a subnetwork sends the message of group membership to its neighbouring multicast routers periodically. The packet format of the message is as follows:

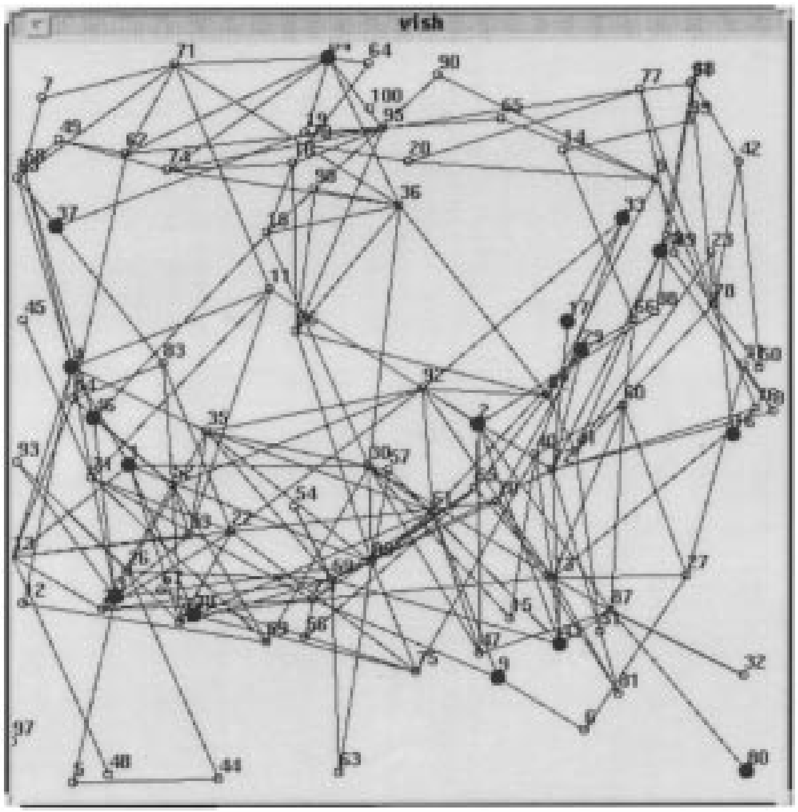

Figure 7. Example of 100-node graph

multicast group multicast group multicast group …
address1 address2 address3

After receiving this message, the receiving multicast router will record these multicast group addresses in a table. Whenever multicast packets from a particular source come, this multicast router will compare the multicast address in the packet header with those in the small table. If the multicast address of the incoming packets exists in the small table, *GMN(source address)* of the multicast router is set to be true. Procedure 1 of the DVMRP algorithm is enhanced as Procedure 3.

Procedure 3. *Enhancement of Procedure* 1
WHILE R receives 'routing updates' entry (d, D, next-hop-address) from n
IF (next-hop-address <> k) AND
(NO n such that (n.subnet=k AND
(n. distance[d] < D OR
(n.distance[d] = D AND (
(NOT R.GMN(d) AND n.GMN(d)) OR
(R.GMN(d) AND n.GMN(d) AND n.address < R.address) OR
(NOT R.GMN(d) AND NOT n.GMN(d) AND n.address < R. address))))))
THEN k is a child of R on the multicast tree rooted at d

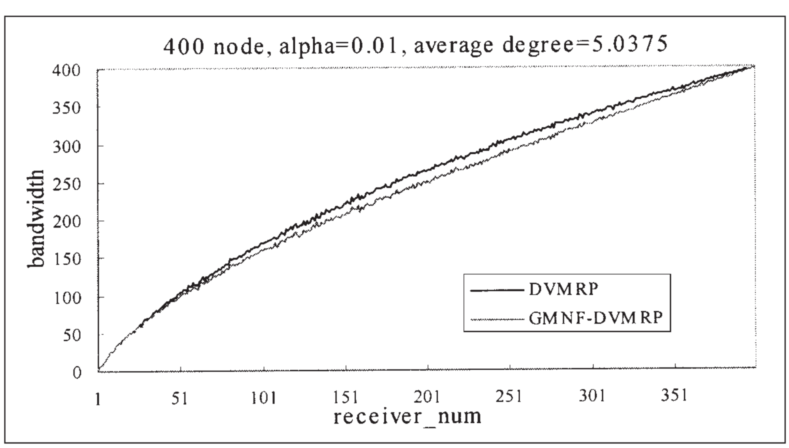

Figure 8. Bandwidth consumption in 400-node graphs

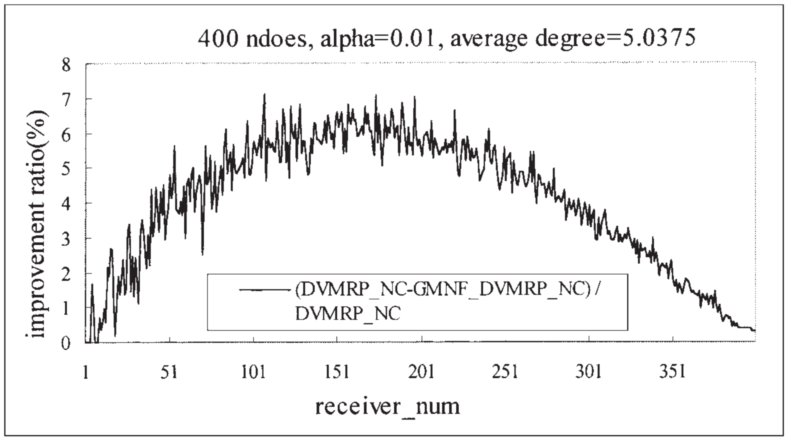

Figure 9. Improvement ratio of bandwidth in 400-node graphs

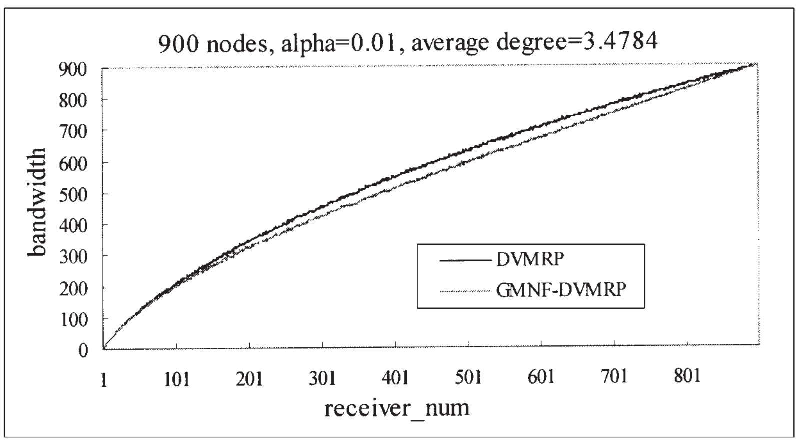

Figure 10. Bandwidth consumption in 900-node graphs

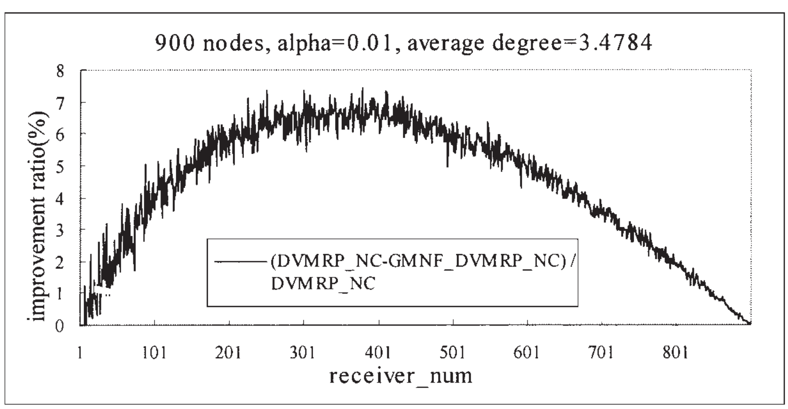

Figure 11. Improvement ratio of bandwidth in 900-node graphs

IF (NO n such that (n.subnet=k and n.distance[d]=infinity))

THEN k is a leaf of R on the multicast tree rooted at d

4. SIMULATION

We have written programs to simulate the above two algorithms. We measure the NC of a multicast tree in a simple way, namely by its total hop counts.

4.1. *Network model*

We model the network as an undirected graph where nodes represent multicast routers and edges represent links or subnetworks connecting routers. Precisely speaking, a node represents a multicast router and its connected hosts. A node, where one of its connected hosts is the sender of multicast packets, is the source and a set of nodes, where one or more of its connected hosts are the receiver of multicast packets, excluding the source node, represents group members.

To create such a network with N nodes, each node is assigned a random co-ordinate in the unit square. The edge between two nodes is generated depending on the probability function of its length. We adopt the following probability distribution function to generate edges:[12]

$$\text{Probability}(d) = \begin{cases} \alpha & \text{if } d \geq 0{\cdot}3 \\ \alpha \dfrac{\sqrt{2} - \mathrm{d}}{\sqrt{2} - 0{\cdot}3} & \text{if } d > 0{\cdot}3 \end{cases}$$

where d is the Euclidean distance between two nodes and α is a constant that affects the average degree of the graph. If the distance between two nodes is no more than 0·3 unit, the probability of generating an edge between these two nodes is fixed to be α. If the distance is more than 0·3, the probability decreases linearly with the distance. In order not to create any isolated node in the graph, we connect nodes one-by-one at the beginning; that is, node 1 to node 2, node 2 to node 3, and so on. An example of such a graph with 100 nodes is illustrated in Figure 7.

4.2. *Simulation result*

Two combinations of size and connectivity are used: a 400-node graph with an average degree of 5·0375 and a 900-node graph with an average degree of 3·47. We generated 10 graphs of each combination.

The simulation results of the two multicast algorithms under these two combinations are shown in Figures 8–11. Our GMNF-DVMRP algorithm has the best effect when the number of receiver nodes is about half of the total node number. It saves about 7% of NC in the 900-node graph. Although 7% is a small number, saving one hop on a multicast tree may result in saving a lot of network bandwidth, because the volume of multicast packets is large.

5. CONCLUSIONS

In this paper we propose an enhancement to the Distance Vector Multicast Routing Protocol, called GMNF-DVMRP, to decrease the NC of the multicast tree formed by DVMRP. A simulation has been implemented to compare our enhanced method and the original DVMRP. We find that our method saves about 5%–7% of NC of the multicast tree formed by DVMRP.

In this paper we focus on DVMRP because it is implemented in the mrouted program. However DVMRP is considered as a first-generation multicast routing protocol. Other multicast protocols have been developed that are considered to be better than DVMRP. In future research we are going to study these second-generation multicast protocols and try to find a better solution.

REFERENCES

1. S. Deering, 'Host extensions for IP multicasting', *RFC 1112*, 1989.
2. D. Waitzman, C. Partridge and S. Deering, 'Distance vector multicasting routing protocol', *RFC 1075*, 1988.
3. T. Ballardie, P. Francis and J. Crowcroft, 'Core based trees (CBT) an architecture for scalable inter-domain multicast routing', *ACM SIGCOMM '93, 85–95, 1993.*
4. S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu and L. Wei, 'An architecture for wide-area multicast routing', *ACM SIGCOMM '94*, 126–135, 1994.
5. K. Bharath-kumar and J. M. Jaffe, 'Routing to multiple destinations in computer networks', *IEEE Trans. Commun*., **COM-31**, (3), 343–351 (1983).
6. E. N. Gilbert and H. O. Pollak, 'Steiner minimal trees', *SIAM J. Applied. Math*., **16**, (1), 1–29 (1968).
7. C. Hedrick, 'Routing information protocol', *RFC 1058*, 1988.
8. L. R. Ford Jr. and D. R. Fulkerson, *Flows in Networks*. Princeton University Press, Princeton, 1974.
9. R. E. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, 1957.
10. Y. K. Dalal and R. M. Metcalfe, 'Reverse path forwarding of broadcast packets', *Commun. ACM*, **21**, (12), 1040–1048 (1978).
11. S. Deering, 'Multicast routing in a datagram internetwork', *PhD Thesis*, 1991.
12. K. Calvert, R. Madhavan and E. W. Zegura, 'A comparison of two practical multicast routing schemes', *GIT-CC-94/25*, 1994.

Authors' biographies:

Yuan-Cheng Lai received BS and MS degrees in computer science and information engineering from National Taiwan University in 1988 and 1990 respectively. From 1992 to 1994 he was an associate researcher in the Computer & Communication Research Labs of the Industrial Technology Research Institute, Taiwan. He is currently finishing his PhD in the Department of Computer and Information Science at National Chiao Tung University. His research interests are in design, management and analysis of ATM networks, queueing theory and multimedia communications.

Ying-Dar Lin received a Bachelor's degree in computer science and information engineering from National Taiwan University in 1988 and MS and PhD degrees in computer science from the University of California, Los Angeles in 1990 and 1993 respectively. He joined the faculty of the Department of Computer and Information Science at National Chiao Tung University in August 1993 and is now an associate professor. His research interests include traffic characterization, service and network management, distributed multimedia applications and high-speed LANs/MANs/WANs. Dr Lin is a member of the ACM. He can be reached at ydlin@cis.nctu.edu.tw.

Wei-Che Yu received BS and MS degrees in computer and information science from National Chiao Tung University in 1994 and 1996 respectively. His research interests

include high-speed LANs and multimedia communications.
He is now on 2 years' duty in the army of Taiwan.