A New Multicast Routing Algorithm in Hierarchical Satellite Networks

Gaofeng Yue, Eylem Ekici, and Ian F. Akyildiz
Broadband and Wireless Networking Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332

***Abstract*—In this paper, a new multicast routing algorithm is introduced for multi-layered satellite networks, which include GEO, MEO, and LEO layers. This scheme aims to minimize the total cost of multicast trees in the satellite network. Multicast trees are constructed and maintained in the dynamic satellite network topology in a distributed manner. Simulation results are provided to evaluate the performance of this scheme in terms of end-to-end delay and multicast tree cost.**

I. INTRODUCTION

Many applications such as software distribution, electronic commerce, and teleconferencing rely on multicast services. The multicast routing problem in terrestrial wire-line networks has already been studied extensively in the past [1]. However, none of the existing multicast routing protocols are well-suited for the dynamic network topologies of satellite networks as discussed in [2]. The only existing multicast routing algorithm [2] developed for the satellite networks is designed primarily for LEO satellite constellations.

GEO, MEO, and LEO layers have their own advantages. A combination of different layers of satellites can provide a more efficient network with better performance than these layers individually. The so-called the *Multi-Layered Satellite Routing Algorithm* (MLSR) [3] for unicasting is designed for a satellite network that consists of satellites in three layers. In order to reduce the computational complexity in satellites and the communication load on the network, the satellite network is organized hierarchically. The hierarchical organization is used for routing table calculations. The data packets are forwarded independent of this hierarchy.Ad apting the method used by MLSR [3] algorithm to handle the mobility, we propose a new multicast routing algorithm for multi-layer satellite configuration. Our new multicast routing algorithm aims to minimize the cost of multicast trees rooted at the source.

The rest of the paper is organized as follows: In Section II, we introduce the satellite network architecture. The new multicast routing algorithm is presented in Section III. Section IV evaluates the performance of the new multicast routing algorithm. Finally, Section V concludes this paper.

II. SATELLITE NETWORK ARCHITECTURE

The hierarchical satellite network consists of three layers of satellites, namely, LEO, MEO, and GEO satellite layers. LEO and MEO satellites are moving with respect to the Earth. The mobility of the LEO satellites is captured by the logical location concept. Logical locations are fixed grid points in the space which are embodied by the nearest satellites. At any given time, a satellite is associated with only one logical location which it is closest to, and is represented by the ID of that logical location. Satellites in the same layer are connected to each other via *Inter-Satellite Links* (ISLs), while the communication between different layers is accomplished over *Inter-Orbital Links* (IOLs). An ISL from a satellite A to another satellite B is denoted by $ISL_{A\rightarrow B}$. Similarly, an IOL from A to B is represented by $IOL_{A\rightarrow B}$. The sources and destinations of information are assumed to be the gateways on the Earth. Satellites communicate with the terrestrial gateways over *User Data Links* (UDLs). A terrestrial gateway can be directly connected to multiple satellites in different layers. Each link in the network is associated with *delay* and *cost* metrics. The *delay* of a link includes processing, propagation, and queuing delays. The *cost* of a link is related to the available bandwidth and the type of the link in the satellite network.

The LEO satellites in the coverage area of a MEO satellite form a LEO group. All LEO satellites in a LEO group are managed by the MEO satellite that covers them. The period in which the LEO group memberships do not change is called a *snapshot* period. LEO groups are represented as virtual nodes in GEO satellites. GEO satellites do not know the details of the LEO satellite layer topology. The MEO satellites in the coverage of a GEO satellite form a MEO group. All MEO satellites in a MEO group are managed by the GEO satellite that covers them. A partial picture of the hierarchical satellite network is depicted in Figure 1. G_i, $i = 1, 2$, are the GEO satellites. $M_{i,j}$ are the MEO satellites and $L_{i,j}$ correspond to LEO groups, $i = 1, 2$; $j = 1, 2, 3, 4$. The LEO satellites within the LEO groups are not shown. The nodes in the satellite network are connected by the dashed lines.

III. DESCRIPTION OF THE ROUTING ALGORITHM

A. Definitions

Definition 1. **Cost of a Link:** The cost *C(l)* of a link *l* is the product of the weight of the link and the utilization of the link. The weight of a link varies with its type (ISL, IOL, or UDL).

Definition 2. **Path:** A path from one node *S* to another node *D* is denoted by $P_{S\rightarrow D}$. $P_{S\rightarrow D}$ includes all nodes along the path and the directed links from *S* to *D* along the path.

Definition 3. **Cost of a Path:** The cost *C(P)* of the path P is the sum of the costs *C(l)* of the links *l* on the path.

Definition 4. **Least Cost Path:** The least cost path $P^*_{S\rightarrow D}$ from node S to the node D is defined as the path from S to D with the minimum cost among all the possible paths $P_{S\rightarrow D}$.

This work is supported in part by the National Science Foundation under grant number ANI-0087762, and NASA Glenn Research Center under grant number NAG3-2580.

0-7803-7632-3/02/$17.00 ©2002 IEEE

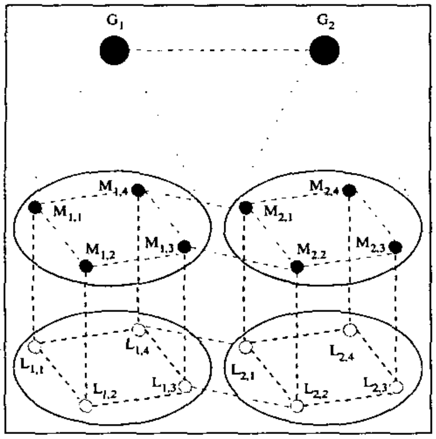


Fig. 1. The Architecture of the Hierarchical Satellite Network.

Definition 5. **Entry LEO Satellite of a LEO Group:** Let $L_{i,j}$ be a LEO group in the satellite network. The Entry LEO satellite $ENLEO_{A \to L_{i,j}}$ of $L_{i,j}$ from any node A is defined as a LEO satellite within $L_{i,j}$ which has a link from A with the least cost among all the incoming links from A to $L_{i,j}$.

Definition 6. **Exit LEO Satellite of a LEO Group:** Let $L_{i,j}$ be a LEO group in the satellite network. The Exit LEO satellite $EXLEO_{L_{i,j} \to A}$ of $L_{i,j}$ to any node A is defined as a LEO satellite within $L_{i,j}$ which has a link to A with the least cost among all the outgoing links from $L_{i,j}$ to A .

B. Overview of Multi-Layered Satellite Routing Algorithm

Multi-Layered Satellite Routing Algorithm (MLSR) considers hierarchical satellite network architecture including GEO, MEO, and LEO layers. The *logical location concept* is employed to isolate the mobility of LEO satellites from the satellites in the upper layers. LEO and MEO satellites are grouped and their management is accomplished by the corresponding MEO and GEO satellites covering them. Summary links are introduced to represent the links connecting the LEO groups and other nodes. In order to calculate routing tables, satellites measure the delay of adjacent links and encapsulate the delay measurement in a data unit called *delay measurement report (DMR)*. Satellites exchange delay measurement reports to create a picture of the topology of the network. *DMR*s are sent from lower layers to upper layers. MEO satellites create *DMR*s for the LEO groups in their coverage, and report their own *DMR*s and the *DMR*s of their LEO groups to the GEO satellites they are connected to. GEO satellites exchange the delay measurement reports to create the total topology of the network, including LEO groups rather than individual LEO satellites. Each GEO satellite calculates the routing tables for all MEO satellites and LEO groups in its coverage. Upon receiving the routing table of its LEO group from the GEO satellite, each MEO satellite generates individual routing tables for the LEO satellites in its LEO group. The details of the routing table calculation can be found in MLSR [3], consisting of a series of computation and communication events.

In our scheme, the collection and exchange of the link costs can be achieved by the method employed in MLSR [3]. The following modifications are needed for the cost exchange procedures used in our scheme:

- We use cost measurement report rather than delay measurement report.
- The gateways report the costs of *User Date Links* to the satellites they are connected to via UDLs.
- A summary link is chosen as the link with the least cost that connects the members of a LEO group with another node in the network.
- Only the first nine steps of the procedure described in MLSR [3] are adopted in the multicasting algorithm. The rest of them are not needed since the multicast trees are created on demand.

In our algorithm, the tree calculation is accomplished in a distributed manner and consists of two stages. First, the GEO satellite of the source gateway creates an initial tree in the *Initial Stage*. The initial tree includes LEO groups rather than individual LEO satellites since it is calculated by GEO satellites. The information about the initial tree is sent to MEO and GEO satellites in the initial tree, and to the MEO satellites whose LEO groups are in the initial tree. Then, the tree calculation enters the *Enhancement Stage*, where these MEO and GEO satellites expand the subtrees in their corresponding coverage areas.

C. The Initial Stage

The tree calculation is initiated by the source gateway. The source *S* creates an *Init* message, which contains the source and the group members of the multicast group. If the source *S* has a UDL to a GEO satellite, it sends the *Init* message to the GEO satellite. Otherwise, it sends the *Init* message along the shortest delay path to its GEO satellite. Receiving the *Init* message, the GEO satellite follows the steps below to compute an initial tree rooted at the source according to the topology information at the GEO satellite, spanning all destinations.

(a) The initial tree (T_i) only has the source node, i.e., $T_i = \{S\}$.
(b) The GEO satellite uses Dijkstra's algorithm [4] to determine the least cost paths from the source to the destinations. Assuming N destinations $D_1, \cdots, D_N$, the GEO satellite calculates $P^*_{S \to D_i}$ $(Definition\, 4)$, for $i = 1, 2, \cdots, N$.
(c) The minimum cost path among all the paths obtained above is added to the initial tree. Select $P = \{P^*_{S \to D_j} | \min_{i \in \{1,2,\dots,N\}} C(P^*_{S \to D_i})\}$, and extend the tree as $T_i = T_i \bigcup P$.
(d) A destination is selected from the destinations not included in the tree, such that the added cost is minimum when the least cost path from a node in the tree to this destination is added to the tree. In other words, the destination which is closest to the tree is connected to the tree. The destination D to be added and the node t in the tree from which the tree will be expanded to D are selected as follows: $(t, D) = \{(t, D_j) | minC(P^*_{t \to D_i})\}, D_i \notin T_i, t \in T_i$. T_i is updated as $T_i = T_i \bigcup P^*_{t \to D}$.
(e) Step (d) is repeated until all destinations are included in the multicast tree.

This process is depicted in Figure 2. The links that are not part of the multicast tree are omitted for clarity. The source

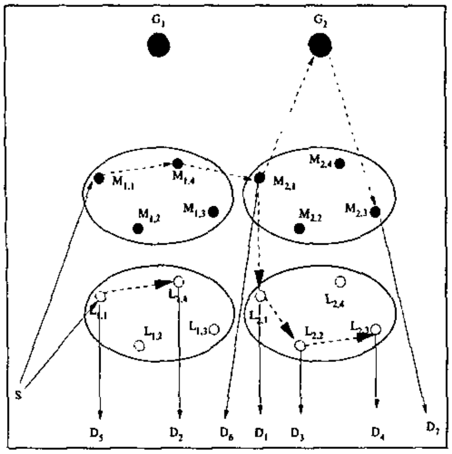


Fig. 2. The Setup of the Initial Tree.

S sends an *Init* message to the GEO G_1. The *Init* message is composed of the source (S) of the multicast group and the group members $D_i \quad i = 1, \cdots, 7$. The procedure described above is used by the GEO G_1 to build the initial tree. The path from S to D_5 is added to the tree first. Then, D_2, D_3, D_1, D_6, D_4, and D_7 are sequentially appended to the tree. The links between satellites are represented by dashed lines, and UDLs by solid lines.

D. The Enhancement Stage

The GEO satellite uses the *Connectivity* message to download the necessary information about the initial tree to the source, MEO and GEO satellites in the initial tree, and MEO satellites of the LEO groups in the initial tree via the direct links or the shortest delay paths. The *Connectivity* message includes an *ownerFlag* field, an *upstream* node field, a *downstream* nodes field, and a field consisting of *destinations* connected to the node receiving the *Connectivity* message. The *ownerFlag* field tells whether the *Connectivity* message is for the node receiving it (*ownerFlag*=1) or for the LEO group of the node receiving this message (*ownerFlag*=0). If both a MEO satellite and its LEO group are in the initial tree, the MEO satellite will receive two *Connectivity* messages, one for the MEO satellite, and the other for its LEO group.

After the *Connectivity* messages are sent, the source gateway sends a *Setup* message to its downstream nodes, which triggers the tree setup and calculation of missing tree segments in the LEO groups. If a LEO group is a downstream node, the *Setup* message is delivered to the managing MEO satellite. The *Setup* message has only one field, *ownerFlag* which has the same meaning as in the *Connectivity* message. The satellite receiving the *Setup* message adds corresponding forwarding entries to its routing table to reach the downstream nodes and the destinations connected to it. Then it sends *Setup* messages to the downstream nodes, as the source gateway does. The MEO satellites use the procedure used by the GEO satellites to calculate the subtrees for their LEO groups or for both themselves and their LEO groups at the same time.

When a satellite *W*, which is a GEO satellite or a MEO satellite whose LEO group is not adjacent to itself in the initial tree, receives a *Setup* message for itself, it adds the corresponding forwarding entries to its routing table to reach the downstream nodes and the destinations directly connected to *W*. If a downstream node is a LEO group (*L*), the corresponding forwarding entry should be from *W* to the $ENLEO_{W \to L}$. Then, *W* sends *Setup* messages to its downstream nodes.

If the LEO group of a MEO satellite is in the initial tree, then the MEO satellite handles the *Setup* message differently according to its position relative to its LEO group in the initial tree. The MEO satellites use the procedure used by the GEO satellites to calculate the subtrees for their LEO groups or for both themselves and their LEO groups at the same time. The difference is the selection of the source and the destinations for different cases, the selection of links involved the tree calculation. After the subtree calculation is completed, the MEO satellites inform the LEO satellites to add corresponding forwarding entries to their routing tables, and send *Setup* messages to their own downstream nodes and/or the downstream nodes of their LEO groups. The following illustrates different cases.

1) If $M_{i,j}$ and $L_{i,j}$ are not adjacent in the initial tree:
   The Entry LEO satellite from the upstream of $L_{i,j}$ (or the source gateway if the source is the upstream node of $L_{i,j}$) is taken as the "source". The destinations connected to $L_{i,j}$ and the Exit LEO satellite to the downstream nodes of $L_{i,j}$ are treated as the "destinations". Only the links within $L_{i,j}$, and the links going from $L_{i,j}$ to the destinations connected to the LEO group are involved in the subtree calculation.
2) If $M_{i,j}$ is the upstream node of its LEO group $L_{i,j}$ in the initial tree:
   $M_{i,j}$ is taken as the "source". The destinations connected to $M_{i,j}$ and $L_{i,j}$, and the Exit LEO satellite to the downstream nodes of $L_{i,j}$ are treated as the "destinations". Only the links within $L_{i,j}$, the links from $M_{i,j}$ to $L_{i,j}$, and the links connecting the destinations are involved in the subtree calculation.
3) If $M_{i,j}$ is the downstream node of $L_{i,j}$ in the initial tree:
   The Entry LEO satellite from the upstream of $L_{i,j}$ (or the source gateway if the source is the upstream node of $L_{i,j}$) is taken as the "source" in the procedure. The destinations connected to $M_{i,j}$ and $L_{i,j}$, and the Exit LEO satellite to the downstream nodes of $L_{i,j}$, and $M_{i,j}$ are treated as the "destinations". Only the links within $L_{i,j}$, the links from $L_{i,j}$ to $M_{i,j}$, and the links connecting the destinations are involved in the subtree calculation.

The resulting complete tree is shown in Figure 3. When each node in the initial tree calculates its subtree, it sends a *Setup Ack* message to the GEO of the source gateway. After receiving all *Setup Ack* messages from the nodes in the initial tree, the GEO satellite sends a *Setup Complete* message to the source to start multicast session.

E. Dynamic Group Membership

When a terrestrial gateway wants to join a multicast group, it sends a *Join Request* message to the source of the multicast group. The source forwards the *Join Request* message to its GEO satellite in charge of constructing the initial tree. Based

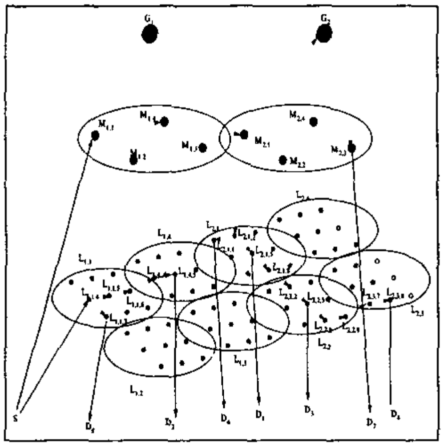


Fig. 3. The Complete Multicast Tree.

on the existing initial tree, the GEO satellite calculates the least costs path from the nodes in the tree to this joining gateway and chooses the minimum cost path. The satellites in the newly added minimum cost path use the method described in III-D to extend the multicast tree.

A gateway on the Earth intending to leave will send a *Prune* message to its upstream satellite, which deals with *Prune* message depending on the number of entries for this multicast session.

When the number of destinations added to or removed from the multicast tree exceeds a threshold value, or when the group memberships change at the beginning of each snapshot period, the *Tree Update* operation is activated by the source gateway to construct a new multicast tree with the updated destinations.

IV. SIMULATION RESULTS

A. System Description

In this simulation, satellite positions and orbits are taken from the GEO constellation *Inmarsat-3*, the MEO constellation *MEONET*, and the LEO constellation *Iridium*. The interconnection structure of the satellites in the hierarchical network is as follows: GEO, MEO, and LEO satellites are connected to their two adjacent neighbors in the same plane via Intra-plane ISLs. LEO satellites in a plane have an inter-plane ISL to each of the adjacent co-rotating planes. We assume that ISLs crossing the seam are not considered for multicast tree generation and there are no inter-plane ISLs in the area above the latitude 70° and below the latitude $-70°$. In the MEO constellation, each MEO satellite can establish an Inter-plane ISL with each of the two adjacent planes. A satellite in a lower layer has an IOL with a satellite which provides the longest coverage service time in each of upper layers. IOLs in reverse directions are also established to provide duplex communications.

Here, we consider two types of source and destination distribution: uniform and non-uniform distribution. For non-uniform distribution, We have adopted the voice traffic distribution from existing literature ([5], [6]) by tailoring it to data traffic distribution. We analyze two cases of network background traffic. The first case is a lightly loaded network, where the utilization of each link is between 10% and 50%. The second case is a heavily loaded network, where the utilization of each link is between 50% and 95%.

B. The Comparison with Shortest Path Tree Algorithm

In the first set of experiments, we compare the performance of the trees created by our algorithm with SPT [4]. An SPT is composed of the shortest delay paths from the source to the destinations. Figure 4(a) shows the delay percentage increase of multicast trees created by our scheme over SPT. The multicast tree cost percentage increase by SPT over our scheme is shown in Figure 4(b). It can be seen that the performance of the SPT protocol and our algorithm vary with group size, member distribution. For a group size, 1000 multicast groups are produced for networks with heavy and light background traffics, respectively. These simulations are executed independently and the comparison results are averaged over the corresponding simulations. In both member distributions, the delay and cost difference have similar curves. However, these differences for non-uniform member distribution are slightly smaller than for uniform member distribution. A noticeable observation is that the delay increase by our algorithm is much smaller than the cost increase by SPT protocol. This indicates that we can sacrifice a small delay loss to achieve a higher bandwidth gain.

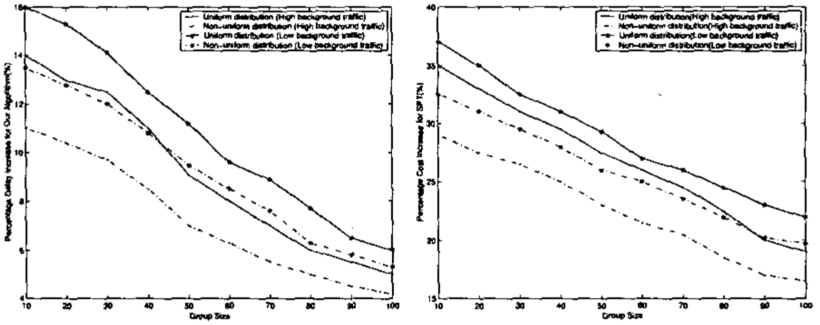
C. The Comparison with Core Based Tree Protocol

In this set of experiments, we compare the performance of the trees created by our algorithm with CBT [7]. The core of the multicast group is selected as follows: the location of each terrestrial gateway is represented as a location vector in a Cartesian coordinate system. The vector of the location center of terrestrial gateways in one multicast group is assumed to be the sum of the location vectors of these terrestrial gateways. The vector of the location center is converted into spherical coordinates. The LEO satellite which is closest to this position is selected as the core. The motivation of using LEO satellites rather than MEO or GEO satellites as cores is that the selection of LEO satellites as cores can incur smaller end-to-end delays.

Figure 5 shows the delay, cost percentage increase of core based trees with respect to the multicast trees generated by our scheme. The delay and cost increase curves have the similar shape for uniform and non-uniform member distribution. However, the increase in cost and delay for non-uniform distribution is higher than for uniform distribution.

D. The Effect of Dynamic Group Membership

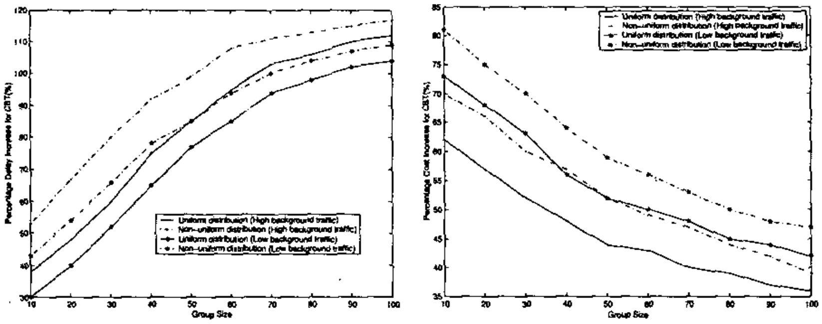
The terrestrial gateways can freely join or leave a multicast group. The joining and leaving of multicast members may make the multicast tree lose its characteristic. When this happens, the tree update procedure should be activated to recalculate the tree. In this experiment, we show how well our algorithm can accommodate dynamic membership. Here, we produce a large amount of multicast groups with non-uniform member distribution and perform the tree update after $1, \cdots, 8$ dynamic operations, respectively. Figure 6 exhibits that the tree cost increase goes up with the number of dynamic operations before the tree update procedure operates. Also, the trees with smaller group member size are more subject to the dynamic operations than larger size groups.



(a) End-to-End Delay Comparison

(b) Cost Comparison

Fig. 4. Comparison with SPT



(a) End-to-End Delay Comparison

(b) Cost Comparison

Fig. 5. Comparison with CBT

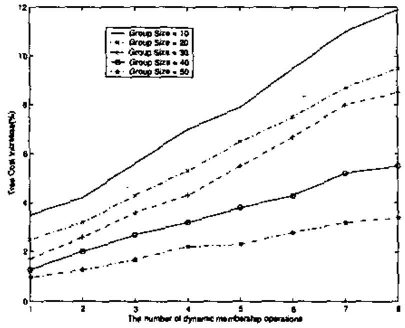


Fig. 6. Effect of Dynamic Group Membership

V. CONCLUSIONS

In this paper, we proposed a multicast routing algorithm for hierarchical satellite networks. Our proposed scheme utilizes the approach used in MLSR [3] to capture the dynamics of the satellite network, where the mobility of the LEO satellites is captured using the logical location concept and the mobility of the MEO satellites is captured with snapshots. The objective of our multicast routing algorithm is to create and maintain multicast trees for which the cost is minimized. The simulation results demonstrate that our algorithm generates multicast trees with lower tree costs at the expense of a small delay increase when compared with shortest path trees. With respect to core based trees, our algorithm has better performance in terms of both delay and cost. The simulations have also shown that our algorithm can support dynamic multicast group membership efficiently.

REFERENCES

[1] L. H. Sahasrabuddhe and B. Mukherjee, "Multicast Routing Algorithms and Protocols: A Tutorial,"*IE EE Network*, vol. 14, no. 1, pp. 90–102, January–February 2000.

[2] E. Ekici, I. F. Akyildiz, and M. D. Bender, "A Multicast Routing Algorithm for LEO Satellite IP Networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 2, pp. 183-192, April 2002.

[3] I. F. Akyildiz, E. Ekici, and M. D. Bender, "MLSR: A Novel Routing Algorithm for Multi-Layered Satellite IP Networks," to appear in *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 411-424, June 2002.

[4] E. W. Dijkstra, "A Note on Two Problems in Connection with Graphs," *Numer. Math.*, vol. 1, pp. 269–271, 1959.

[5] M. Werner. "Analysis of system connectivity and traffic capacity requirements for LEO/MEO S-PCNs." In E. Del Re, Editor, *Mobile and Personal Communications, Proceedings 2nd Joint COST 227/231 Workshop*, pages 183-204, Florence, Italy,A pr.1 995. Elsevier.

[6] M. Werner and G. Maral. "Traffic Flows and Dynamic Routing in LEO Intersatellite Link Networks." *Proceedings of the 5th International Mobile Satellite Conference(IMSC'97)*, pp. 283-288,1997

[7] A. Ballardie, B. Cain, and Z. Zhang, "Core Based Trees (CBT)," *in Proc. of ACM SIGCOMM '93*, pp. 85–95, September 1993.