ATM Multicast Methods: A Comparison and Evaluation

Stephen O'Hearn
Department of Computer Science
University of Saskatchewan
57 Campus Drive
Saskatoon, SK S7N 5A9
CANADA
e-mail: sdo124@cs.usask.ca
Supervisor: Carey Williamson

Abstract

As wide area networks evolve, there is an increasing demand for multicast support. However, as networks expand to large-scale wide area networks, link resources become more scarce in comparison to richly interconnected small-scale local area networks. Distances become vast and multicast members become increasingly disparate. As a result, the efficiency of multicast distribution trees becomes very important as networks grow.

Several multicasting strategies have been devised for optimally forming multicast paths from the source(s) to the receivers. However, each method has shortcomings peculiar to it because of trade-offs that must be made in situating the fan-out nodes of the distribution tree. For example, some multicast trees yield lower average end-to-end delays but consume more bandwidth and/or require more storage for tree state information.

This paper presents a detailed survey of several multicast algorithms. These algorithms are compared and evaluated in terms of efficiency: low bandwidth consumption, low end-to-end delay, simplicity of tree computation, minimal tree state storage requirements, even traffic concentration, and network size scalability (from LAN to WAN).

Introduction

Many new applications are emerging that are contributing to an increased demand for multicast support across the Internet. Some examples include audio and video conferencing, shared whiteboard, replicated database systems, distributed systems, software update distribution facilities, distance education, pay-per-view CATV, weather maps, and many other applications (Eriksson 1994, Harrison 1995). Furthermore, the number of subscribers to these services is growing at a tremendous rate as evidenced by the widespread growth in the MBone, a virtual network running on top of the Internet that provides it with multicast support (Eriksson 1994).

It has been shown, however, that the current IP multicast architecture scales poorly with respect to computational overhead, storage requirements, and bandwidth as internetwork size and number of wide-area multicast calls increase (Ballardie, Francis, Crocroft 1995). That is, the current architecture will eventually become inadequate to efficiently support the impending wide-scale use of the current and future multicast applications. Therefore, a multicast routing protocol is needed which forms a highly efficient delivery tree - the set of paths traversed by multicast packets during their transmission from the sources of a multicast group to the receivers of that group (Ballardie, Francis, Crocroft 1995). In forming this tree, the algorithm should strive to meet as many of the following desirable properties as possible:

- low total bandwidth consumption
- low end-to-end delay
- simplicity of tree computation
- even traffic concentration
- low state information overhead
- scalability

The principal objective of multicasting is to route packets from a set of sources to a (possibly large) set of receivers of a corresponding multicast group in a way that minimizes the total consumption of network bandwidth (Harrison 1995). However, notwithstanding total bandwidth utilization, several other factors should also be considered in designing efficient multicasting techniques and routing protocols. For example, many applications like real-time video conferencing, are highly delay-sensitive in addition to being bandwidth-intensive. A scheme optimized for bandwidth may include source-to-member paths in its multicast route that are too long to yield acceptable delay times for the particular application. Moreover, overzealous attempts to multicast in the most bandwidth- conserving way can actually result in highly concentrated flows along certain network paths, leading to a poor bandwidth utilization of those paths. Some tree structures are not computable in polynomial time.

Other considerations relate to the storage requirements and to the amount of computation done by the hosts and routers of the network or internetwork in the establishment and maintenance of the multicast delivery tree(s) for each group.

Finally, a multicast routing protocol should be scalable in the number of wide-area multicast calls. Any routing algorithm or protocol not scaling well across the full range of applications will have a limited usefulness.

This paper examines several multicast routing protocols, both current and proposed, and compares their strengths and weaknesses against the desirable properties mentioned above which will serve as the evaluation criteria. In evaluating algorithms, it is important to consider the restrictions of large-scale wide area networks in the real world. It must be remembered as each method is examined that the knowledge of global topology at any network node is limited, and that routers should be able to work efficiently in a distributed manner.

Background

The problem of how to form the optimal multicast distribution path from the sources to the receivers of a group has many potential solutions involving several compromises. This path can take the form of a group of source-based, shortest path trees, spanning from each source to the corresponding group receivers, or alternatively, as a single tree shared by all sources and receivers of the group.

As will be discussed, traditional IP multicast protocols have incorporated shortest path trees exclusively into their algorithms due in part to their simplicity of computation (Wei, Estrin 1993). For example, Dijkstra's algorithm can be used to find the shortest path to the receivers from each source in turn. However, most of the research on multicast tree types has been directed towards group shared trees, due to their enormous bandwidth-saving potential. One commonly researched shared tree class includes the Steiner minimal tree (Wei, Estrin 1993), and the near-optimal approximations to the Steiner tree, including the KMB algorithm (Kou, Markowsky, Berman 1981) and the Doar and Leslie Naive algorithm (Wei, Estrin 1993). As will be discussed, the main objective of these algorithms is to build a cost-optimal shared tree between the sources and receivers of a group; in this context, cost refers to bandwidth utilization.

Another group shared tree type under extensive consideration is the core-based tree (or center-based tree) designed to simplify the computational requirements of the Steiner trees and their approximations, to reduce the amount of state information set-up (Ballardie, Francis, Crowcroft 1993), and to allow for the design of protocols using rendezvous mechanisms (Deering, Estrin, Farinacci, Jacobson, Liu, Wei 1994). In this paper, different variants of core-based tree structures are examined.

Existing IP Multicast Architecture (Shortest Path Trees)

The existing IP multicast architecture is not limited exclusively to IP networks, but is often considered to be the multicasting solution in many types of networks (from LANs, to extended LANs, to ATM WANs). The IP multicast architecture is the building block of two currently used multicast routing schemes, namely the Distance-Vector Multicast Routing Protocol (DVMRP) and the Link- State Multicast Routing Protocol.

For each multicast group, and for every source in each group, the current architecture builds a shortest-path tree (SPT) rooted at the source and extending to the corresponding receivers. The algorithms that accomplish this are tightly bound to the particular algorithm used to route unicast packets in the internetwork; such multicast routing protocols are said to be *protocol dependent.* Various ad hoc means must be used to construct the shortest-path trees for any group that extends beyond domain boundaries where different multicast algorithms are used and must interface.

Some form of multicast addressing is supported in the hardware of most broadcast LANs, such as Ethernet, FDDI, and ATM LANs (Ballardie, Francis, Crocroft 1995, Partridge 1994). As a result, the hosts and routers on these LANs can distinguish multicast packets (or cells) from unicast packets based on their address type. IP, for example, uses class D addresses, which comprise a portion of the IP address space reserved for multicast group addresses. Each address within this class refers to a particular multicast group currently in session (Harrison 1995). In both DVMRP and link-state multicast routing, routers receive all multicast packets in the internetwork. Any host attached to the subnet of that router can then receive these packets if it requests to become a member of the corresponding multicast group. Routers are made aware of requests of attached hosts to join a multicast group by means of an extension to the IP protocol called the *Internet Group Management Protocol*

(IGMP). In IGMP, a particular router on the LAN subnet, called the *designated router*, polls the LAN for host memberships to the multicast group at regular intervals.

DVMRP

DVMRP, developed by Deering (Deering, Cheriton 1990), establishes source-based shortest-path trees through a pruning algorithm based on a method used for internetwork broadcasting, called *Truncated Reverse Path Forwarding* (Dalal, Metcalfe 1978). The first few multicast packets from a source are broadcast throughout the network. Eventually packets reach routers having no child subnets nor subordinate routers with multicast member hosts on them. These routers then send a prune message back along the reverse path towards the source. The router on the reverse path closer to the source, in turn, also sends prune messages along the reverse path if it receives prune messages from all of its subordinate routers and if its attached subnet has no host members of the group. The trees are thus built from information about group member *absence* propagating in the reverse direction of the trees along all branches not leading to receivers. Packets originating from the same sources are then blocked from traversing branches of the trees not leading to group members.

Link-State Multicast Routing Protocol

Link-State Multicast Routing is accomplished by having routers include, in their link-state tables for each outgoing link, a list of groups having multicast members on that link. Whenever a new group appears on a link of the internetwork, a router on that link broadcasts the new link-state entry to all of the other routers on the internetwork.

Consequently, each router has complete knowledge of group membership for each link and is thus able to formulate a shortest path multicast tree using Dijkstra's algorithm; the tree construction information ends up being disseminated across the internetwork. A router can thus determine whether it is situated on any shortest path trees and convey multicast packets along its appropriate incident links.

Evaluation of Existing Shortest Path Tree IP Multicast Protocols

Algorithms that build source-based delivery trees perform poorly in conserving total bandwidth because multiple delivery trees must exist for a group rather than a single shared tree. Furthermore, DVMRP routers must intermittently transmit prune messages to maintain the correct set of delivery trees across the internetwork. The Link-State Multicast Routing Protocol requires routers to flood the internetwork with group membership reports to maintain the disseminated tree state information.

Since these protocols build trees that give the shortest paths from each source to the receivers, they have the lowest possible end-to-end delay values. They also offer the simplest delivery tree computation (using Dijkstra's algorithm in polynomial time). Furthermore, traffic concentration is kept relatively even because each source has its own distribution tree.

However, shortest path tree approaches incur large state information overhead because several trees must be maintained per group. This drawback severely retards scalability with number of sources as well. If G multicast groups exist with S sources per group on average, then the scaling factor is $S \times G$. Another scaling problem relates to routers that are not involved in a multicast group. Such routers must still process prune messages for every group on the internetwork.

Group-Shared Trees

Steiner Minimal Tree

A Steiner Minimal Tree (SMT) is defined as the minimum cost subgraph (i.e. tree) spanning a given subset of nodes in a graph (Winter 1987). In the context of multicasting, a SMT can be defined as the shared tree of minimum total link cost that links the sources of a multicast group to the receivers.

The SMT approach to multicasting would be very efficient in terms of low total bandwidth consumption because only one tree is formed for the group. Furthermore, state information overhead would be very low for the same reason. However, the computation of the SMT is NP-complete and is not expected to have a polynomial time solution (Winter 1987). Thus the tree computation is too complex to allow this approach to used in practice. This computational complexity also prohibits this algorithm from being scalable to internetworks of any reasonable size.

Another problem is the unbounded delay that is theoretically possible with the SMT; the maximum source-to-receiver path length can be equal to the longest acyclic path in the original network (Wei, Estrin 1993). (Note that finding the longest acyclic path in a graph is also an NP-complete problem, causing delays in extremely large networks to be virtually unbounded.)

Steiner Minimal Tree Approximation Algorithms

A number of sub-optimal approximations to the Steiner Minimal Tree have been devised that are considered acceptable for use in real networks.

Kou, Markowsky and Berman invented what is now called the *KMB algorithm* to approximate SMTs (Kou, Markowsky, Berman 1981). The cost of the KMB tree averages only 5 percent more than the cost of the SMT (Wei, Estrin 1993). The original KMB algorithm requires complete network topology information, making it unsuitable for use on wide area networks. However, distributed KMB algorithms exist where each router would only need partial network topology information. But it is believed that the distributed KMB algorithm would entail enormous message complexity making protocol design extremely difficult (Wei, Estrin 1993).

Another SMT approximation algorithm by Doar and Leslie (Wei, Estrin 1993) called the *Naive Tree Algorithm* first computes a shared tree based on finding the shortest path trees across the initial multicast group and then combining them. New members are then joined to the nearest attachment point on the tree (Wei, Estrin 1993). Through simulation, it was found that the average cost of the Naive trees was only about 50 percent higher than the average cost of the KMB trees (Wei, Estrin 1993).

Evaluation of SMT and Sub-optimal SMT Approximations

Because SMT and approximations involve a single shared tree, bandwidth utilization and link costs are optimized. Indeed, KMB tree simulations on random graphs yielded the lowest cost of multicast distribution over all tree types considered in this paper (Wei, Estrin 1993).

However, these trees are optimized for link cost rather than for delay, and hence, the other criteria are poorly met. As already mentioned, end-to-end delay can be unbounded (for the SMT), and tends to be significantly higher than for any other tree type, especially as network size increases (Wei, Estrin 1993). The ratio of worst-case KMB tree delay to SPT delay was as high as 2.4 in random graph simulations (Wei, Estrin 1993). Simplicity of tree computation, state information overhead, and scalability have already been addressed above. Unfortunately no simulations to date have examined KMB tree traffic concentration. It might be speculated, however, that since bandwidth utilization is minimized, certain subsets of the multicast tree links would be highly concentrated with traffic flow leading to an uneven traffic concentration across the tree.

Center-Based Tree Approaches

Due to the shortcomings of the Steiner tree and its approximations, and to the desire to have a single bandwidth- conserving shared tree that could be computed relatively easily, Wall proposed several center-based (or core-based) tree algorithms (Wall 1983). Two practical protocols were created based on Wall's center-based tree algorithms, namely the Core Based Tree Interdomain Multicast Routing Protocol (Ballardie, Francis, Crowcroft 1993), and Protocol Independent Multicast (PIM) (Deering, Estrin, Farinacci, Jacobson, Liu, Wei 1995).

Center-based tree methods form a single, group-shared shortest path tree rooted at a "center node" in the internetwork. Wall gives two ways to choose this center node: (1) *Delay-Optimal CBT*: Choose any network node (group member or non-member) as a center node giving the center-based tree having minimal delay (maximum or average); (2) *Delay-Optimal MSPT (Member- /sender-rooted Shortest Path Tree)*: Choose from amongst the multicast group members a center node giving the center-based tree having minimal delay (maximum or average). A third way to choose the center from any network node is called *Cost- optimal CBT*. This tree type minimizes aggregate tree-link costs. If the choice of center is made from amongst the multicast group members, the method is called *Cost-optimal MSPT*.

However, not all of these center-based trees can be used in practice. Finding the center for the delay-optimal CBT is NP- complete (Wei, Estrin 1993). Practical CBTs can be found using heuristics for center placement (Ballardie, Francis, Crowcroft 1993). Alternatively, delay-optimal MSPTs can be used in practice. There is also no efficient distributed algorithm to compute a cost-optimal CBT (Wei, Estrin 1993). Therefore, cost-optimal MSPTs can be used instead.

Evaluation of Center-Based Trees

Included in this evaluation is a report on the findings of an extensive simulation performed by Wei and Estrin at the University of Southern California on the various multicast tree types already discussed (Wei, Estrin 1993). These simulations were performed on a graph modelling the early ARPAnet topology and then again on large random graphs. The results were consistent across both types of simulation.

All of the center-based tree types discussed above were simulated and their performance was compared with the source-based shortest path trees discussed earlier in terms of delay, cost, and traffic concentration distribution. The purpose of this comparison was to determine how much better center-based trees perform in terms of cost (bandwidth), and how much worse they perform in terms of delay and evenly distributed traffic concentration, and to show that center-based trees are, in fact, superior. KMB tree costs and delays were also included in the comparison.

The results for link cost comparisons revealed the following: KMB tree average cost was less than cost-optimal CBT average cost, which, in turn, was less than both delay-optimal CBT and delay-optimal MSPT average costs (which were roughly equal), which was less than SPT average cost. In short, SPTs exhibit a relatively high cost (or bandwidth consumption); the relatively easily computed delay-optimal MSPT has a reasonably low cost.

The results for delay basically showed the opposite trend: SPTs, of course, have the lowest delay, followed by delay-optimal CBT average delay, followed by delay-optimal MSPT average delay, followed by cost-optimal CBT average delay, followed by KMB tree average delay as the highest. In short, SPTs exhibit the lowest average delay; again the delay-optimal MSPT gives a reasonably low delay.

The SPTs, as would be expected, exhibited the most even traffic distribution. The distribution was found to become more evenly distributed as the average node degree was increased. This observation occurs with SPTs presumably because higher degree implies the existence of more incident links along which the SPTs can be situated. The CBTs, on the other hand, exhibited uneven traffic concentration as was expected; the concentration failed to become more even with increased degree.

It is apparent that simultaneous optimization of cost (bandwidth utilization), end-to-end delay, and traffic concentration is impossible. However, a good trade-off exists: The author of this paper believes that delay-optimal MSPT gives the best overall performance since it gives reasonable delay performance and relatively low cost. Furthermore, delay-optimal MSPTs are relatively easy to compute. The major drawback of the MSPT approach is the potential for unevenly distributed concentration of traffic flow.

Center-based trees, maintaining a single shared group tree, exhibit a low state information overhead. Simplicity of tree computation can also be realized using MSPTs, although more computation is involved here than in the computation of SPTs.

Center-based trees in general, and particularly the delay-optimal MSPTs, should allow the construction of highly scalable multicast protocols from local area networks to wide area internetworks. This scalability is made possible by a combination of its simplicity, relatively low cost, and relatively low delay. MSPT- based protocols should be sufficient, then, for both delay-sensitive, and bandwidth-intensive applications.

Protocol Independent Multicast (PIM)

PIM is a newly proposed protocol designed primarily to facilitate scaling of multicast services from LANs to WANs (Deering, Estrin, Farinacci, Jacobson, Liu, Wei 1995). PIM differentiates between two different modes of service, namely *Dense Mode (DM)* and *Sparse Mode (SM)* operations, which can be used in tandem to establish optimal multicast distribution trees from the sources of a group to their receivers. In this context, density implies plentiful group membership within a particular region of the internetwork, or a condition where link and bandwidth resources in the area spanned by the members are rich; sparsity implies a large spread of group members across regions of the internetwork. Thus, PIM can establish a highly efficient set of multicast distribution trees for any multicast group topology, even when the group contains sources or receivers situated far away from the majority of the group members. This is accomplished by having routers judiciously choose from amongst the two modes of operation as the tree is formed.

PIM was motivated primarily by a need for efficient SM multicasting (again SM stands for sparse-mode). However, it also specifies a new DM protocol similar to DVMRP but which is more efficient, as will be described below.

Unlike DVMRP, PIM-DM can make use of existing unicast routing tables independent of the unicast routing table that created them. Yet PIM is not dependent on the unicast routing method. Furthermore, PIM-DM does not require storage of its own unicast routing information. Another innovation of PIM is the integration of PIM-DM control message processing and data packet forwarding with the PIM-SM operations. This innovation allows a single router to run different modes as deemed appropriate for different groups. Control message overhead can be limited to a fixed percentage of the bandwidth available by adjusting the control message transmission frequency on a per link basis.

PIM-SM strives to constrain data distribution to a minimal number of routers. PIM-SM differs from existing IP multicast approaches in two ways:

- Routers with local members must send explicit join messages to join a group rather than having membership assumed until explicit prune messages are sent.

- PIM-SM establishes Rendezvous Point(s) for receivers to "meet" new sources rather than using a data driven approach wherein the first few multicast packets cause tree construction

Using this approach, PIM-SM effectively builds a CBT across the internetwork.

Evaluation of PIM

PIM promises to be the most scalable multicast architecture because it combines ideas from other trade-off optimal methods in the formation of a delivery tree across the network or internetwork. This delivery tree is optimized by topological area of the group as explained above. Thus, in keeping with the group topology, it should be possible to optimize the factor that is most important for that particular area and multicast application. For example, if certain links become highly concentrated, it should be possible for PIM routers near the concentrated area to use a DM operation, allowing the concentrated links to be abridged by links nearby. If a particular application is highly delay-sensitive, the PIM protocol can use a DM operation to ensure that SPTs are generated.

However, the proposed PIM protocol appears to be fairly complicated. Certain multicast group topologies might fall somewhere in between DM and SM. It might have been easier to simply use MSPTs.

Conclusion

This paper has considered the relative merits of using several available and proposed multicast distribution tree forming architectures. It was shown that existing IP architectures which form source-based shortest path trees are acceptable for small- scale internetworks. But these were shown to scale poorly to the wide-area or to large multicast group sizes. Several alternative center-based trees were then examined; some being only theoretically useful, while others were practical.

It appears as though simultaneous optimization of all criteria used to evaluate efficiency in this paper is not possible. The principal goal of future research, then, should not be to try to accomplish this impossible feat, but rather, to find the optimal trade-off multicast tree for the given group topology.

References

Ballardie, T., Francis, P. and Crowcroft, J. 1993. Core Based Trees (CBT): An Architecture for Scalable Inter- Domain Multicast Routing. In *Proceedings of ACM SIGCOMM '93*, pp. 85-95

Dalal, Y. and Metcalfe, R. 1978. Reverse Path Forwarding of Broadcast Packets. In *Communications of the ACM*, Vol. 21, No. 12, pp. 1040-1048.

Deering, S. and Cheriton, D. 1990. Multicast Routing in Datagram Internetworks and Extended LANs. In *ACM Transactions on Computer Systems*, Vol. 8, No. 2, pp. 85-110.

Deering, S., Estrin, D., Farinacci, D., Jacobson, V., Liu, C. and Wei, L. 1995. Protocol Independent Multicast (PIM): Motivation and Architecture. (http://www.usc.edu/dept/cs/tech.html/95-608.ps)

Eriksson, H. 1994. MBONE: The Multicast Backbone. In *Communications of the ACM*, Vol. 37, No. 8, pp. 54-60.

Gilbert, E. and Pollak, H. 1968. Steiner Minimal Trees. In *SIAM Journal of Applied Mathematics*, Vol. 16, No. 1, pp. 1-29.

Harrison, T. 1995. Performance Evaluation of Routing Algorithms for Multicast Traffic. M.Sc. Thesis, University of Saskatchewan.

Kou, L., Markowsky, G. and Berman, L. 1981. A Fast Algorithm for Steiner Trees. In *Acta Informatica*, Vol. 15, pp. 141-145.

Partridge, C. 1994. *Gigabit Networking*, Addison-Wesley, Reading MA.

Wall, D. and Owicki, S. 1983. Construction of Centered Shortest-Path Trees in Networks. In *Networks*, Vol. 13, No. 2, pp. 207-230.

Wei, L. and Estrin, D. 1993. The Trade-offs of Multicast Trees and Algorithms. (http://www.usc.edu/dept/cs/tech.html/93-560.ps)

Winter, P. 1987. Steiner Problem in Networks: A Survey. In *Networks*, Vol. 17, pp. 129-167.