A Comparative Study of Multicast Protocols: Top, Bottom, or In the Middle?

Li Lao[1], Jun-Hong Cui[2], Mario Gerla[1], Dario Maggiorini[3]

llao@cs.ucla.edu, jcui@cse.uconn.edu, gerla@cs.ucla.edu, dario@dico.unimi.it

[1] Computer Science Department, University of California, Los Angeles, CA 90095

[2] Computer Science & Engineering Department, University of Connecticut, Storrs, CT 06029

[3] Computer Science Department, University of Milano, via Comelico 39, Milano, Italy

Technical Report TR040054
January 2005
Computer Science Department
UCLA

Abstract

Multicast solutions have been evolving from “bottom” to “top”, i.e., from IP layer (called IP multicast) to application layer (referred to as application layer multicast). Recently, there are some new proposals (named as overlay multicast) using certain “infrastructure” (composed of proxies or service nodes) in the middle. Although it is well accepted that application layer multicast and overlay multicast are easier to deploy while sacrificing bandwidth efficiency compared with IP multicast, little research has been done to systematically evaluate and compare their performance. In this paper, we conduct a comparative study of different types of multicast routing protocols. We first present a qualitative comparison of three types of protocols, and then we provide a quantitative study of five representative protocols, namely, PIM-SSM, NARADA, NICE, POM and TTOM by extensive simulations. Our studies will help to answer some of the most important questions, such as which way to go: top, bottom, or in the middle?

1 Introduction

Recently, more and more group communication applications (e.g., video-conferencing, online-gaming, and long-distance education) have emerged with the increasing popularity of the Internet. To support such multi-user applications, multicast is considered as a very efficient mechanism since it uses some delivery structures (e.g., trees or meshes) to forward data from senders to receivers, aiming to reduce duplicate packets, whereas a separate delivery path is built for each sender-receiver pair when simple unicast scheme is adopted. Initially, multicast is implemented at the IP layer (i.e., IP multicast [16]), in which a tree delivery structure is usually employed, with data packets only replicated at branching nodes. In IP multicast, the multicast tree nodes are network routers ([28, 26, 17]). However, due to many technical and marketing reasons, such as the lack of a scalable inter-domain multicast routing protocol, the requirement of global deployment of multicast-capable IP routers and the lack of appropriate pricing models, etc., IP multicast is still far from being widely deployed.

To resolve the deployment issues of IP multicast, application layer multicast ([14, 20, 5, 29, 23, 25, 32, 37, 8]) and overlay multicast ([11, 22, 12, 34, 33, 6]) have been proposed as alternative solutions to realize multicast in the Internet. In both approaches, some nodes form a virtual network, and multicast trees (or other delivery structures) are constructed on top of this virtual network. Data packets are only replicated at the participating nodes, and delivered through tunnels between these nodes. However, there are significant differences between application level multicast and overlay multicast. The former approach relies only on end hosts that are part of a multicast group, whereas the latter one uses strategically deployed overlay proxy nodes (sometimes referred to as service nodes) besides end hosts in order to facilitate the management of group membership and multicast delivery structures. Comparatively, these two approaches are much easier to deploy than IP multicast since both of them neither require global multicast-capable IP routers, nor depend on an inter-domain multicast routing protocol. On the other hand, however, they sacrifice the efficiency of multicast, since some data packets might be transmitted over the same physical link multiple times.

Even though it is well accepted that application layer multicast and overlay multicast trade bandwidth efficiency for easier deployment, little research has been done to systematically evaluate and compare their performance. Answers for many important questions are still open. For example, how much worse do the upper layer alternative solutions perform compared to IP multicast? Can they serve as a long-term substitute to IP multicast, or only as a temporary solution to circumvent the difficulties of deploying IP multicast? What are the trade-offs of using extra overlay proxies vs. relying solely on end hosts? In other words, which architecture should we choose in which scenarios?

In this paper, we strive to answer these questions. We compare the performance of IP multicast, application layer multicast, and overlay multicast with respect to a variety of metrics through extensive simulation study. We also evaluate the impact of overlay structure (such as the number of proxies) on overlay multicast performance. Our goal is to provide guidelines for researchers and developers to adopt appropriate schemes under different conditions. To our best knowledge, this is the first work on vertically comparing multicast protocols implemented on three different layers of the protocol stack.

The remaining paper is organized as follows. In Section 2, we give an overview on the three multicast architectures, namely, IP multicast, application layer multicast, and overlay multicast, and provide a qualitative comparison. We describe our experimental methodologies in Section 3. Then we present the simulation results in Section 4. Finally, we briefly review related work in Section 5 and conclude our work in Section 6.

2 Multicast Routing Protocol Overview

In this section, we provide a brief overview on the multicast routing protocols and give a qualitative comparison.

We divide multicast routing protocols into three categories, namely, IP multicast, application layer multicast, and overlay multicast, depending on whether the group management and data replication are implemented in network routers, end hosts, or intermediate overlay proxies. Fig. 1 gives a high level illustration of the three multicast architectures.

2.1 IP Multicast

In IP multicast, when a host joins or leaves a group, it informs its designated multicast router in its subnetwork. Then the participating multicast routers exchange group membership information and use multicast routing protocols to establish multicast trees to deliver data. By using the tree delivery structure, IP multicast improves network efficiency and scales to large group size, since data packets are only replicated at branching routers and no redundant packets are delivered over each link. The most well-known IP multicast routing protocols are Distance Vector Multicast Routing Protocol (DVMRP) [28], Multicast Open Shortest Path First (MOSPF) [26], Core-Based Trees (CBT) [3], Protocol Independent Multicast (PIM) with Dense and Sparse Mode [17], and PIM-SSM [21]. Among these protocols, DVMRP, MOSPF, PIM-DM, and PIM-SSM construct source based trees, whereas CBT and PIM-SM utilize a core or Rendezvous Point (RP) to organize multicast trees.

Despite the bandwidth efficiency of IP multicast, it usually suffers from a number of issues ([2, 18]), such as the lack

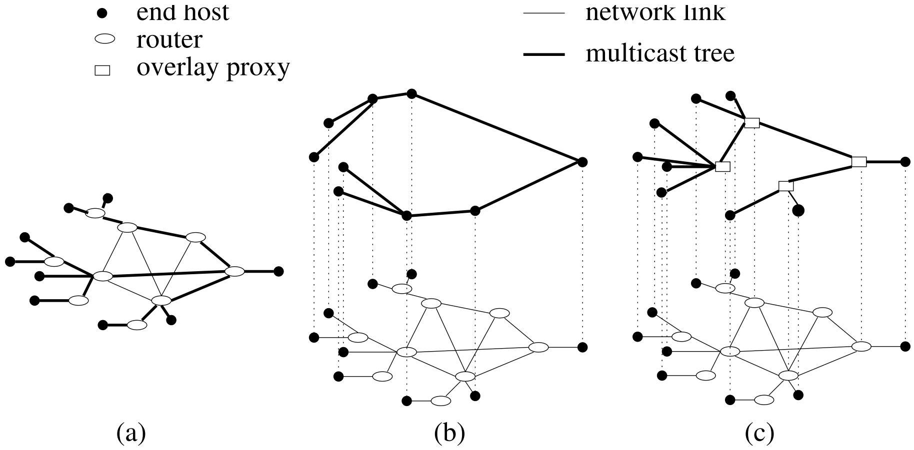


Figure 1: An illustration of (a) IP multicast, (b) application layer multicast, and (c) overlay multicast.

of a scalable inter-domain multicast routing protocol, the state scalability issue in presence of a large number of multicast groups, the difficulty of member access control and efficient reliable data transfer, and the demanding requirement of global deployment of multicast-capable routers. Thus, even till now, MBONE is the only existing while "decaying" global IP multicast testbed.

2.2 Application Layer Multicast (ALM)

Due to its ease of deployment, application layer multicast has recently gained increasing popularity in the multicast community. In this type of multicast architecture, group membership, multicast tree (or some other delivery structure) construction, and data forwarding are solely controlled by participating end hosts, thus it does not require the support of intermediate nodes (such as routers or proxies).

In general, we can divide application layer multicast protocols into two categories: structured ([32, 37, 8]), and unstructured ([14, 5, 29, 23, 25, 20]). Structured schemes leverage Distributed Hash Table (DHT)-based overlay networks and build multicast forwarding trees on top of this structured overlay. Unstructured schemes either organize end users into a mesh and establish multicast trees on top of this mesh (e.g., End System Multicast [14, 13], NICE [5], ALMI [29]), or directly construct a multicast tree (e.g., Yoid [20]) or other well-known topologies (such as hypercubes and Delaunay triangulations in HyperCast [25]).

In application layer multicast, however, the lack of knowledge about underlying network topology usually results in performance penalty compared with IP multicast, such as lower efficiency and longer end-to-end latency. Furthermore, it typically requires a large amount of control overhead to maintain group membership and multicast trees among the end users and to monitor network conditions by sending expensive probing messages. Thus, it is commonly believed that the inherent inefficiency of bandwidth usage and the difficulty of managing group membership and maintaining multicast trees pose a lot of challenges when multicast groups are large. Finally, the end hosts are fairly unstable, and they can join or leave multicast groups at will, which will affect the performance of downstream members.

2.3 Overlay Multicast (OM)

Alternatively, multicast functionalities can be supported by some additionally deployed intermediate nodes (called proxies or service nodes), which can form a "backbone overlay" network. In such an overlay network, proxies are special nodes placed inside the network. They cooperatively construct an overlay infrastructure and establish multicast trees (or other structures)

Table 1: Comparison of multicast architectures

Metrics	IP	ALM	OM
Ease of deployment	low	high	medium
Multicast efficiency	high	low	medium
Control overhead	low	high	medium
Robustness	high	low	high

among themselves for data delivery. Outside the backbone overlay, the proxies can deliver data packets to end hosts via multicast or unicast. Recently, the research community has seen increasing interests in overlay multicast, and many seminal works have been conducted, such as Scattercast [11], Overcast [22], RMX [12], AMCast [34, 33], and OMNI [6].

Compared with other two types of multicast architectures, overlay multicast has several advantages: it more or less "plays the game" in between, thus combining the benefits of both IP multicast and application layer multicast. First of all, overlay multicast takes advantage of the knowledge about the network topology by using overlay proxies: the end users can find closest overlay proxies and thus naturally form clusters around proxies without complicated measurement techniques and clustering algorithms; the overlay proxies are strategically placed and connected, so the resulting overlay network can resemble the physical network; the proxy nodes can act as additional branching points for multicasting data packets. Therefore, overlay multicast can easily improve multicast efficiency and reduce resource (such as link bandwidth) consumption. Second, the limited size of the backbone overlay and local clusters around proxies allows more efficient group membership and multicast tree management. It also reduces the scope of control message exchange. Third, unlike application layer multicast, where each multicast overlay only supports one multicast group and/or one application, backbone overlay can support multiple groups and applications. The control overhead for probing and refreshing overlay links within the backbone is bounded, and is not affected by the number of co-existing groups supported. Consequently, it is expected that when there is a large number of groups, overlay multicast tends to out-perform application-layer multicast in terms of control overhead. Last, overlay proxies are more robust than end hosts, thus it is less likely for the data transmission to be disrupted in overlay multicast (especially in POM) in comparison with application layer multicast. However, it is worth noting that these benefits are achieved at the cost of deploying and maintaining overlay proxies. Furthermore, careful design of the overlay network is necessary for achieving high performance.

A Qualitative Comparison To summarize our discussions above, we present a qualitative comparison in Table 1, illustrating the pros and cons of IP multicast, application layer multicast and overlay multicast. In Section 4, we will give a quantitative comparison by extensive simulations.

3 Experimental Methodologies

In this section, we describe our experimental methodologies, especially the topology graphs, group membership generation, and the multicast protocols of interest. Then we discuss the performance metrics used in our simulation study.

3.1 Topology Graphs

We model the network as an undirected graph $G = (V, E)$, where V is the set of nodes, and E the set of links. To quantify the trade-offs of multicast schemes on diversified networks, we use realistic Internet topologies at Autonomous System (AS) level as well as at router level, since these two kinds of graphs exhibit dramatically different characteristics such as network size and node degree distribution. Further, these two topologies allow us to evaluate the potential of using these multicast protocols in a hierarchical fashion at inter-domain and intra-domain levels.

AS level: In AS-level topologies, each node represents a a domain and each edge is an interconnections between domains. We use the real Internet topologies provided by University of Oregon Route Views Project [1]. The data is collected from BGP routing tables of multiple geographically distributed BGP routers. For each graph, we select some nodes as group members according to the group membership generation method described in the following subsection.

Router level: We use the router-level ISP maps measured by Rocketfuel, an Internet mapping tool based on traceroutes and a few innovative techniques [35]. These maps consist of backbone routers and access routers within several major ISPs. Multicast group members are end hosts directly attached to the access routers.

For both topology types, we ran simulations on different graphs. Here we mainly present the results for an AS topology with approximately $3,000$ nodes, which was collected on January 24, 1998, and a router-level topology with approximately 300 nodes within an ISP called Ebone (Europe). In the remaining section, we denote these two graphs as 980124 and 1755 respectively.

3.2 Group Membership Generation

In this study, group members are randomly uniformly distributed in the network. For example, in AS-level topologies, nodes are randomly selected as group members; whereas in router-level topologies, additional nodes are created as group members and randomly attached to access routers. The group size is varied from 5 to 1280 to investigate the multicast performance for small and large groups. For each group, a source is randomly selected from the members. In the simulations, the members join groups at the first 400 seconds of the simulation time, and the performance metrics are collected after 1000 seconds.

3.3 Multicast Protocols for Evaluation

We compare five representative multicast routing protocols. For a fair comparison, all of these protocols use source-based tree approach (i.e., building a multicast tree for each source).

IP multicast: For IP multicast, we implement PIM-SSM protocol, which is believed to be more resource efficient than other source-based tree protocols such as DVMRP, PIM-DM and MOSPF. PIM-SSM uses reverse path forwarding (RPF) to construct shortest path trees rooted at the source, and a soft-state approach is employed by periodically refreshing the multicast forwarding states.

Application layer multicast: We evaluate NARADA, a protocol targeted at applications with small and sparse groups, and NICE, a scalable application layer multicast protocol. NARADA implements the End System Multicast architecture [14, 13], in which end hosts periodically exchange group membership information and routing information, build a mesh based on end-to-end measurements, and run a distributed distance vector protocol to construct a multicast delivery tree. NICE [5], on the other hand, recursively arranges group members into a hierarchical overlay topology, which implicitly defines a source-specific tree for data delivery. It has been shown that NICE scales to large groups in terms of control overhead and logical hops.

Overlay Multicast: We implement two prototypes of overlay multicast protocols, namely, Pure Overlay Multicast (POM) and Two-Tier Overlay Multicast (TTOM). Both of them manage overlay multicast trees among the proxy nodes inside the backbone overlay. For the communication between end users and proxies, unicast is used for POM, and application layer multicast is used for TTOM. We want to examine the benefits and overhead of using application layer multicast for users to connect to the proxies. In fact, these two protocols can be treated as simple abstractions of various overlay multicast proposals (such as OMNI, AMCast, Overcast, and Scattercast).

In our study, we use some simple heuristics to design backbone overlays. First, we choose nodes with the highest degree in the graphs as the proxies. Since the number of proxies is a parameter, we will examine the impact of this parameter on the performance of overlay multicast. After the proxies have been determined, a complete graph is maintained between the proxies, i.e., each proxy periodically exchanges link state information (obtained by measurement) with others. On top of this mesh, overlay links for data delivery are then established according to a so called "Adjacent Connection" method [27, 24] after the underlying paths have been explored: an overlay link is established between two proxies if this path does not go through other proxies.

3.4 Performance Metrics

We evaluate the performance of multicast protocols from the following aspects.

Multicast tree quality: Three metrics are used in our simulations to quantify multicast tree quality. *Multicast tree cost* is the number of physical links in the multicast tree. This is a direct measure of the resource (bandwidth) usage of multicast groups. *End-to-end delay* is defined as the number of hops between a source and a receiver. This metric reflects the timeliness of multicast data delivery. *Link stress* is the number of duplicate packets delivered over each physical link. It measures the average resource utilization in the network.

Control overhead: We use the total number of control messages incurred during the 1000-second simulation time to evaluate control overhead. The control messages include multicast tree set-up or tear-down messages, multicast tree refresh messages, and overlay link measurement (or probing) messages (for NARADA, NICE, POM, and TTOM).

Work Load: *Node stress* is the number of children on the multicast tree for a node, and it measures the load distributed on each node. We evaluate node stress for end nods and proxies (when applicable) separately.

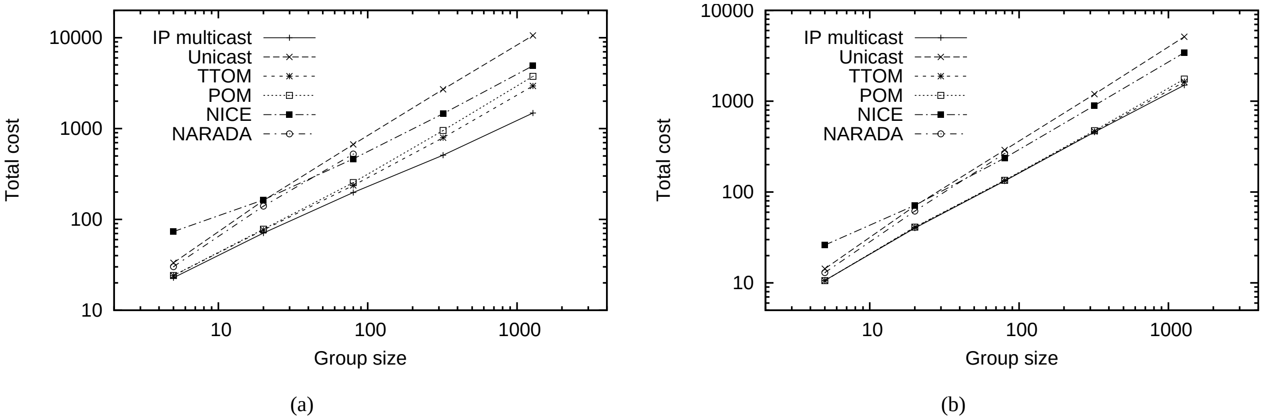
Robustness: In application layer multicast and overlay multicast, end hosts can leave the group voluntarily or due to failure. All downstream members will be affected. Thus, we use *number of descendants* to measure the robustness. It is defined for each end host as the number of downstream end hosts directly or indirectly connected to it.

4 Simulation Studies

We conduct the simulations at packet level. For fairness reasons, we set the equivalent timers in different protocols to the same value. For example, refresh timer for refreshing overlay mesh and multicast tree state is 10 seconds, and probe timer for end hosts to find better parents or add/drop overlay links is 20 seconds. Unless otherwise specified, we select 61 nodes with degree higher than 19 as proxies for AS graph 980124, and 15 nodes with degree higher than 10 as proxies for router-level graph 1755. In the remaining section, we present our simulation results.

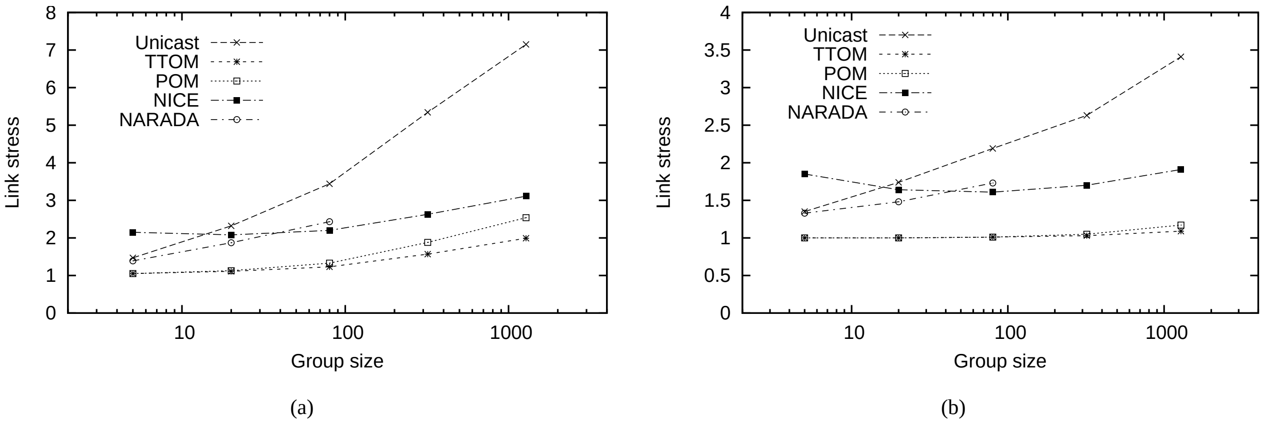
4.1 Multicast Tree Quality

Multicast Tree Cost We plot the results of multicast tree cost for different protocols in Fig. 2. Since NARADA is designed for relatively small groups and its control overhead grows rapidly with group size, we only show its performance for group size up to 80. We also give the total cost of unicast in the figure as a reference. Multicast tree cost shows a similar trend in both topology graphs: for all kinds of group size, IP multicast trees have the lowest cost, application layer multicast trees have the highest cost, and overlay multicast trees lie in between. This is consistent with our earlier discussion in Section 2. Depending on the type of graphs, as group size increases, the difference of the tree cost between application layer multicast and overlay multicast may magnify (as in the AS graph 980124) or decrease (as in the router-level graph 1755). However, even in the latter case, overlay multicast always perform significantly better than even the most scalable application layer multicast. For example, for the largest group size 1280, in 1755, the average multicast tree cost for TTOM, POM and NICE is 2938, 3751 and 4903 respectively; whereas in 980124, the cost is 1628, 1752 and 3420. Note that the tree cost of 980124



(a) (b)

Figure 2: Multicast tree cost. (a) Router-level topology; (b) AS-level topology.



(a) (b)

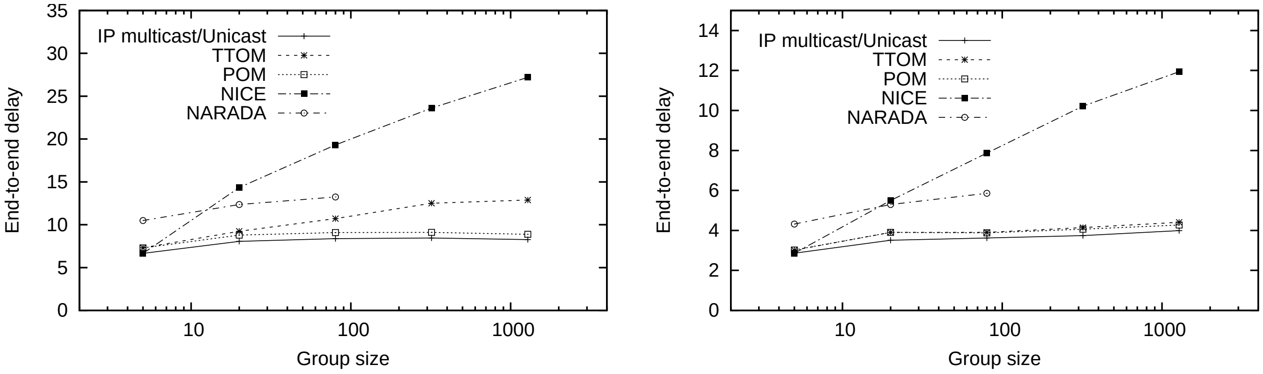
Figure 3: Average link stress. (a) Router-level topology; (b) AS-level topology.

is generally lower than that of 1755, because in 1755, we create additional nodes as members and randomly attach them to access routers. Consequently, the sizes and the diameters of the resulting graphs have been increased.

We also observe that, between the application layer multicast protocols, NARADA outperforms NICE for small groups, but not for group size larger than 100. In NARADA, each node maintains membership information about all other members and periodically improves mesh quality by probing and adding or dropping links. This feature allow NARADA to efficiently construct multicast trees when group size is smaller; but the overhead explodes when group size increases. In contrast, NICE sacrifices some efficiency but improves scalability by organizing the group into a hierarchical structure. For overlay multicast protocols, TTOM has lower tree cost than POM, since application layer multicast trees in the local clusters (i.e., the set of members clustered around each proxy) reduces the cost of unicast. In addition, in comparison with POM, TTOM has similar tree cost in large graphs such as 980124 but much lower cost in small graphs such as 1755. This is reasonable, since in the smaller graphs, every cluster spans fewer nodes and the paths between proxy and members have more overlap, so more cost can be saved by using application layer multicast.

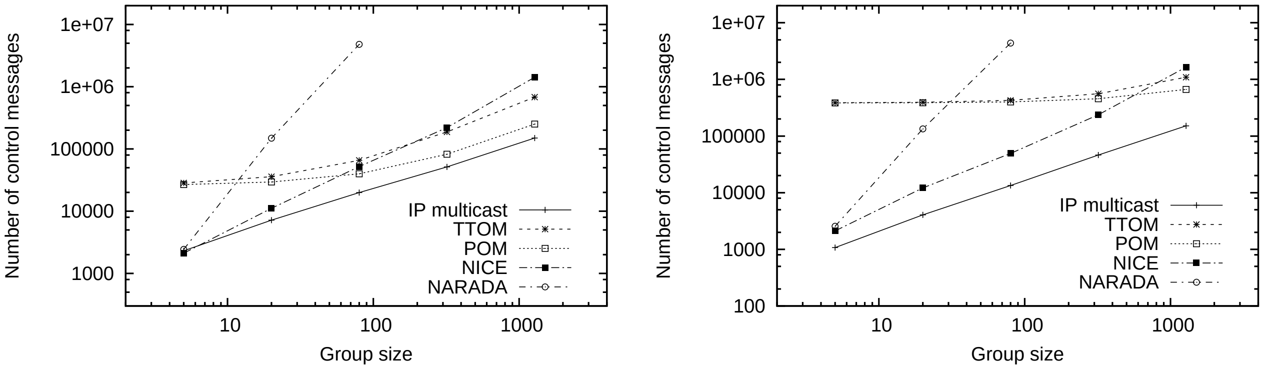
Link Stress We also show the results for link stress in Fig. 3, from which we observe very similar trends to tree cost in Fig. 2 (Note that IP multicast is not plotted since it has unit link stress): the performance of overlay multicast lies between IP multicast and application layer multicast; NARADA is more efficient than NICE only when the group size is small; TTOM consistently has smaller link stress than POM.

End-to-End Delay Fig. 4 depicts the end-to-end delay of multicast trees. IP multicast yields the same delay as unicast,



(a) (b)

Figure 4: Average end-to-end delay. (a) Router-level topology; (b) AS-level topology.



(a) (b)

Figure 5: Control overhead for a single group. (a) Router-level topology; (b) AS-level topology.

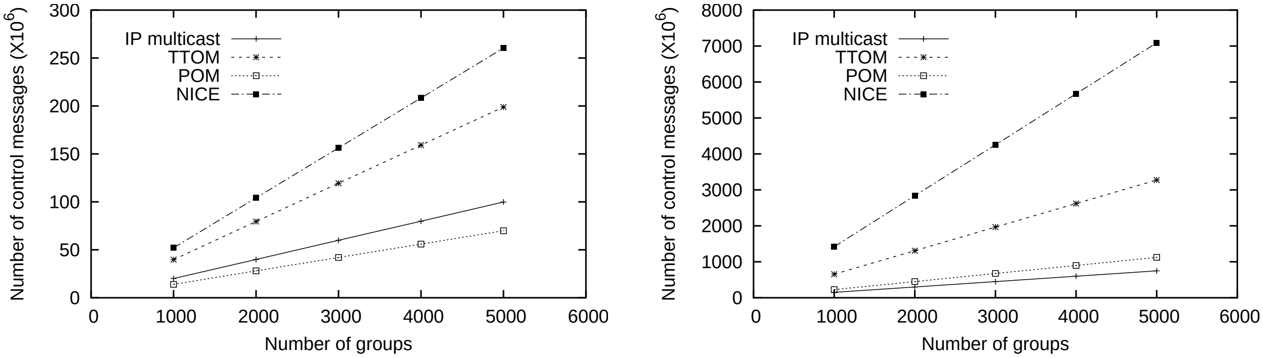
as shortest path trees are used in our simulations. The delay of overlay multicast (especially POM) trees is slightly higher than that of IP multicast. The main reason is that the proxies are high-degree nodes located in the core of the network, and the overlay connections created by “Adjacent Connection” resemble the underlying network. Therefore, data packets do not need to go through unnecessarily long paths. In comparison, application layer multicast has the highest delay. As group size increases, the delay of NARADA trees remains fairly constant, but the delay of NICE trees increases rapidly, due to the fact that the number of clusters and thus the number of logical hops increase with group size.

Comparing Fig. 4 with Fig. 2, a trade-off between multicast tree efficiency and end-to-end delay can be found. For example, NICE begins to have a lower tree cost than NARADA when the group size is around 100, but its delay also exceeds NARADA at the same time. Unicast has the highest cost, while it achieves the lowest end-to-end delay. As to IP multicast and overlay multicast, they comparatively sit at better trade-off points between the tree cost and end-to-end delay.

4.2 Control Overhead

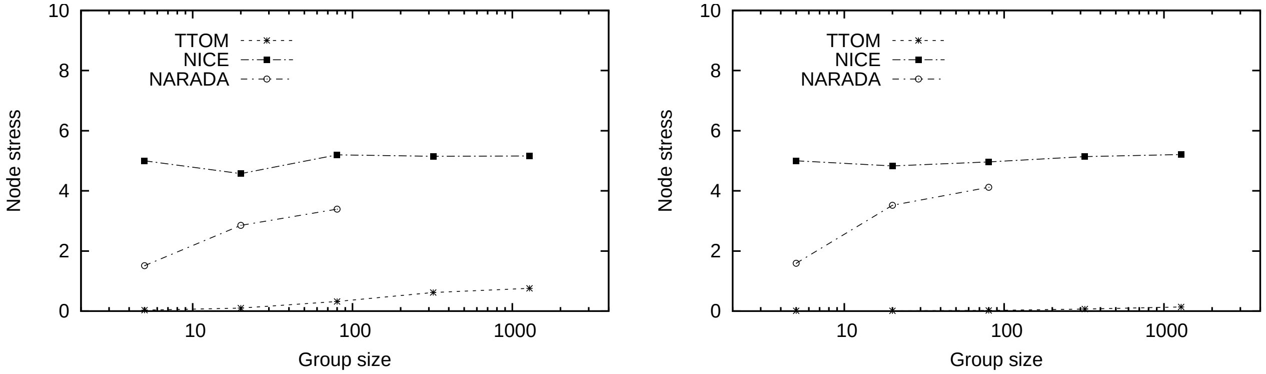
To evaluate the control overhead of different multicast protocols, we carry out two sets of simulation experiments. In the first set, we study the control overhead for single groups with varying group size. In the second set, we examine the control for multiple groups by changing the number of groups.

We plot the results of control overhead for single groups in Fig. 5(a). We find out that IP multicast has the least control overhead overall. Application layer multicast has smaller control overhead than overlay multicast for smaller groups and/or



(a) (b)

Figure 6: Control overhead for multiple groups in router-level topology. (a) 80 group members; (b) 1280 group members.



(a) (b)

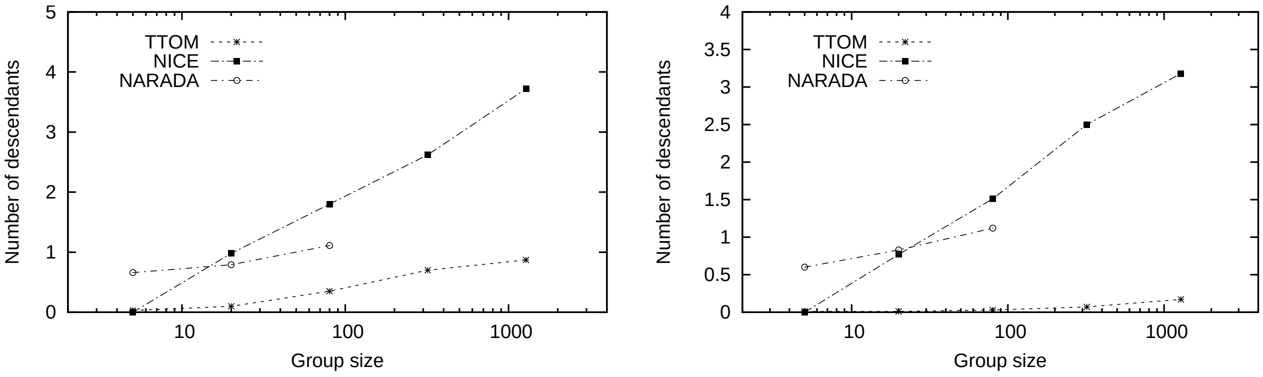
Figure 7: Average node stress of end users. (a) Router-level topology; (b) AS-level topology.

larger graphs; but its overhead exceeds that of overlay multicast when group size increases beyond a certain point. The control overhead of overlay multicast consists of two parts: backbone overlay and cluster maintenance. The former part is mainly determined by the number of proxies and is independent of group size. Only the latter part is related to group size. In small graphs, the number of proxies is smaller, so the backbone overlay maintenance overhead is also lower. On the other hand, in application layer multicast protocols, the control overhead is mainly determined by group size. Consequently, the slopes of the overlay multicast (POM and TTOM) curves are much smaller than that of application layer multicast curves in the figure.

Let us consider the scenario when there are multiple concurrent groups in the network. The backbone overlay maintenance overhead is constant irrespective of the number of groups maintained by the backbone overlay. The control overhead of application layer multicast is proportional to the number of groups, since each group independently establishes its multicast tree. Hence, we expect that overlay multicast gains more benefits when there is a large number of groups. In Fig. 6, we plot the control overhead in 1755 for different group sizes when the number of groups varies. It is clear that when there are numerous groups, overlay multicast out-performs application layer multicast even at small group size. This gap becomes more distinct as group size increases.

4.3 Work Load

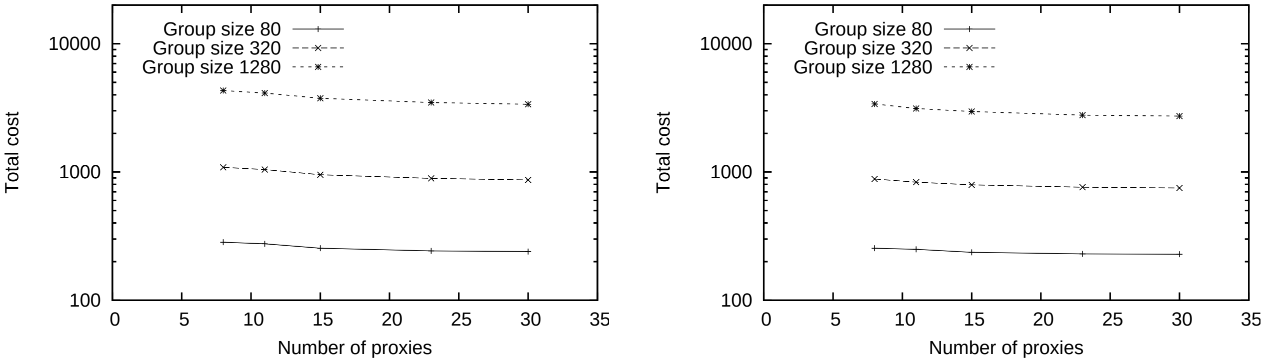
Fig. 7 presents the average node stress for nodes with non-zero stress. IP multicast and POM are not included since their members do not need to forward multicast packets. From this figure, NICE and NARADA have higher load on end hosts than



(a)

(b)

Figure 8: Average number of descendants. (a) Router-level topology; (b) AS-level topology.



(a)

(b)

Figure 9: Total cost vs. number of proxies. (a) POM; (b) TTOM.

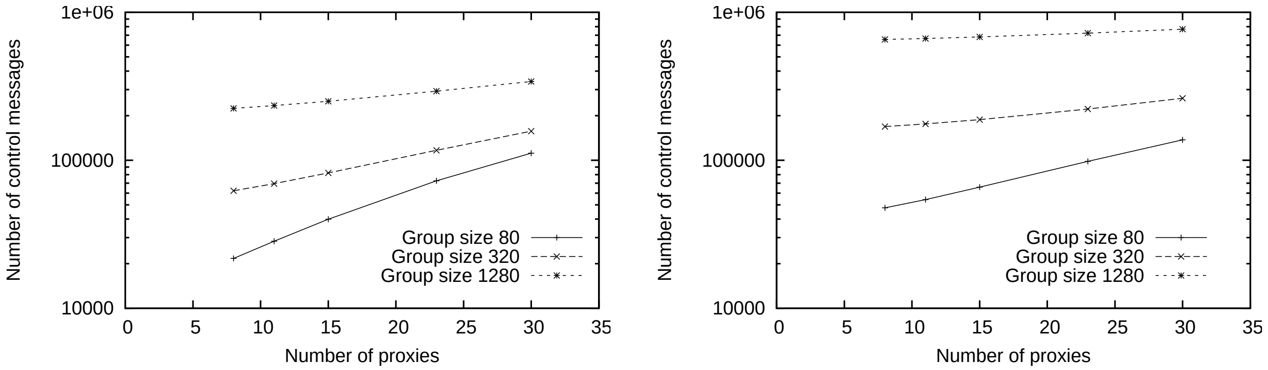
TTOM on average. The reason is simple: In TTOM, the overlay proxies reduce the load distributed on end hosts because they deliver data packets to some members directly. We also examined the average node stress for proxy nodes in TTOM and POM, and we found that POM proxies has higher stress than TTOM proxies, because proxies forwards data to members directly in POM and directly/indirectly in TTOM.

4.4 Robustness

As mentioned earlier, robustness is an important aspect for application layer multicast and overlay multicast. Fig. 8 demonstrates that TTOM is more robust than NARADA and NICE with respect to the number of descendants each member has. This is again due to the fact that proxy nodes deliver packets directly to some members and help to reduce the average number of descendants for end hosts. In NICE, cluster leaders in higher layers of the hierarchy have a very large number of descendants. If one of these nodes fail, then a large portion of the members will be affected. From this point of view, TTOM and POM are more robust than NICE, albeit at the expense of deployment and management cost of overlay proxies.

4.5 Impact of the Number of Overlay Proxies

In previous experiments, we fix the number of overlay proxies in each topology. Now we investigate the impact of this parameter on the performance of overlay multicast protocols. Fig. 9 and 10 depict multicast tree cost and control overhead of



(a) (b)

Figure 10: Control overhead vs. number of proxies. (a) POM; (b) TTOM.

POM and TTOM as the number of proxies increases for different group sizes in 1755. It is clear that a larger number of proxies helps to improve the multicast tree efficiency and reduce the multicast tree cost. At the same time, it induces a larger amount of control overhead due to backbone overlay maintenance. Additional simulation results (not shown here) indicate that the deployment of more proxies can also decrease average end-to-end delay and link stress. Accordingly, network designers can tune this parameter to balance the application performance and overhead.

4.6 Summary and Discussions

In this section, we have provided a quantitative comparison of four representative multicast routing protocols in the three types of multicast architectures. Our observations can be summarized as follows: 1) IP multicast and overlay multicast (POM and TTOM) have better trade-offs between multicast tree cost and end-to-end delay than application layer multicast (NARADA and NICE); 2) NARADA is more resource efficient than NICE when the group size is small (less than 100 in our simulations), while it incurs relatively higher end-to-end delay; on the other hand, NICE is more scalable to large groups in terms of tree cost, but the end-to-end delay increases rapidly when there are more group members; 3) By using application layer multicast trees in local clusters (such as TTOM), the overall multicast tree efficiency and quality of overlay multicast can be improved at the cost of higher control overhead and more complicated tree management; 4) for single groups, overlay multicast has higher control overhead than IP multicast and application layer multicast when the groups size is small; however, the control overhead is significantly decreased when there are multiple multicast groups since the backbone overlay maintenance overhead does not increase with the number of groups; 5) the performance of overlay multicast is significantly affected by the number of proxies, and thus the overlay design is very critical for overlay multicast.

Based on our studies, it seems that overlay multicast is a promising solution to multicast wide deployment, considering its comparable performance to IP multicast and easier deployment. On the other hand, however, we need to point out its additional cost: proxy placement and overlay dimensioning, which are not involved in application layer multicast at all. Thus, overlay multicast is more suitable for an ISP to adopt in order to balance the “price” and “value” (for which obviously a proper pricing model is needed), while application layer multicast is good for immediate deployment, especially for small groups.

5 Related Work

In the literature, there have been many seminal works on multicast performance evaluation. Early works, such as [36], [7], [15], [30], and [10], mainly focus on comparison of IP multicast protocols and characterization of IP multicast trees. More recent works study the performance of application layer multicast protocols. For example, Radoslavov et. al. compare application-level and router-assisted hierarchical schemes for reliable multicast [31]. Fahmy and Kwon characterize application level multicast trees, such as NARADA and TAG, using numerical analysis and simulations [19]. Castro et. al. evaluate DHT (Distributed Hash Table) based application level multicast protocols, including CAN and Pastry [9]. Banerjee and Bhattacharjee provided a brief overview and a high-level comparison of several representative application level multicast protocols in [4], but no simulation results have been given. Clearly, most previous works either investigate the performance of multicast protocols implemented in the same layer (e.g., IP multicast or application-level multicast) in a horizontal way, or focus on a specific type of multicast protocols (such as reliable multicast). Our work, to the best of our knowledge, reports the first vertical comparison of all three types of multicast protocols implemented on different layers of the protocol stack.

6 Conclusions

In this paper, we conduct a comparative study of different types of multicast routing protocols: IP multicast, application layer multicast, and overlay multicast. We first give an overview of the protocols, comparing them qualitatively, then by simulations, we provide a quantitative study of four representative multicast protocols, namely, PIM-SSM, NARADA, NICE, POM and TTOM. Based on our studies, we could answer the three important questions posed in Section 1: 1) overlay multicast could achieve comparable performance to IP multicast; 2) compared with application layer multicast, overlay multicast is a good choice for large numbers of groups; 3) a good indication (from our analysis and simulation results) is that application layer multicast is a suitable solution for immediate deployment, while overlay multicast could serve as a long-term solution, which deserves significant amount of further research efforts on different aspects, such as overlay design and pricing model.

References

[1] *University of Oregon Route Views Project*. http://antc.uoregon.edu/route-views/.

[2] K. Almeroth. The evolution of multicast: From the MBone to inter-domain multicast to Internet2 deployment. *IEEE Network*, Jan./Feb. 2000.

[3] A. Ballardie. Core Based Trees (CBT version 2) multicast routing: protocol specification. *IETF RFC 2189*, Sept. 1997.

[4] S. Banerjee and B. Bhattacharjee. *A Comparative Study of Application Layer Multicast Protocols*. Under submission.

[5] S. Banerjee, C. Kommareddy, and B. Bhattacharjee. Scalable application layer multicast. In *Proceedings of ACM SIGCOMM*, Aug. 2002.

[6] S. Banerjee, C. Kommareddy, K. Kar, B. Bhattacharjee, and S. Khuller. Construction of an efficient overlay multicast infrastructure for real-time applications. In *Proceedings of IEEE INFOCOM*, Apr. 2003.

[7] T. Billhartz and et al. Performance and resource cost comparisons for the CBT and PIM multicast routing protocols. *IEEE Journal on Selected Areas in Communications*, 1997.

[8] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron. Scribe: A large-scale and decentralized application-level multicast infrastructure. *IEEE Journal on Selected Areas in Communications*, 20(8):1489 – 1499, Oct. 2002.

[9] M. Castro, M. B. Jones, A.-M. Kermarrec, A. I. T. Rowstron, M. Theimer, H. J. Wang, and A. Wolman. An evaluation of scalable application-level multicast built using peer-to-peer overlays. In *Proceedings of IEEE INFOCOM*, apr.

[10] R. Chalmers and K. Almeroth. Modeling the branching characteristics and efficiency gains of global multicast trees. *Proceedings of the IEEE INFOCOM’ 2001*, Apr. 2001.

[11] Y. Chawathe, S. McCanne, and E. A. Brewer. *An Architecture for Internet Content Distributions as an Infrastructure Service*, 2000. Unpublished, http://www.cs.berkeley.edu/yatin/papers/.

[12] Y. Chawathe, S. McCanne, and E. A. Brewer. RMX: Reliable multicast for heterogeneous networks. In *Proceedings of IEEE INFOCOM*, Mar. 2000.

[13] Y.-H. Chu, S. G. Rao, S. Seshan, and H. Zhang. Enabling conferencing applications on the internet using an overlay multicast architecture. In *Proceedings of ACM SIGCOMM*, Aug. 2001.

[14] Y.-H. Chu, S. G. Rao, and H. Zhang. A case for end system multicast. In *Proceedings of ACM Sigmetrics*, June 2000.

[15] J. Chuang and M. Sibru. Pricing multicast communications: A cost-based approach. *Proceedings of the Internet Society INET’98 Conference*, July 1998.

[16] S. Deering. Multicast routing in a datagram internetwork. *Ph.D thesis*, Dec. 1991.

[17] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, and L. Wei. The PIM architecture for wide-area multicast routing. *IEEE/ACM Transactions on Networking*, 4(2):153–162, Apr. 1996.

[18] C. Diot, B. Levine, J. Lyles, H. Kassem, and D. Balensiefen. Deployment issues for the IP multicast service and architecture. *IEEE Network*, Jan. 2000.

[19] S. Fahmy and M. Kwon. Characterizing overlay multicast networks. In *Proceedings of IEEE ICNP*, Nov. 2003.

[20] P. Francis. *Yoid: Extending the Multicast Internet Architecture*. White papar, http://www.aciri.org/yoid/.

[21] H. Holbrook and D. Cheriton. IP multicast channels: EXPRESS support for large scale single-source applications. *Proceddings of SIGCOMM’99*, Sept. 1999.

[22] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O. Jr. Overcast: Reliable multicasting with an overlay network. In *Proceedings of USENIX Symposium on Operating Systems Design and Implementation*, Oct. 2000.

[23] M. Kwon and S. Fahmy. Topology aware overlay networks for group communication. In *Proceedings of NOSSDAV’02*, May 2002.

[24] Z. Li and P. Mohapatra. The impact of topology on overlay routing service. In *Proceedings of IEEE INFOCOM*, Mar. 2004.

[25] J. Liebeherr, M. Nahas, and W. Si. Application-layer multicasting with delaunay triangulation overlays. *IEEE Journal on Selected Areas in Communications*, 20(8):1472–1488, Oct. 2002.

[26] J. Moy. Multicast routing extensions to OSPF. *RFC 1584*, Mar. 1994.

[27] A. Nakao, L. Peterson, and A. Bavier. A routing underlay for overlay networks. In *Proceedings of ACM SIGCOMM*, Aug. 2003.

[28] C. Partridge, D. Waitzman, and S. Deering. Distance Vector Multicast Routing Protocol. *RFC 1075*, 1988.

[29] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel. ALMI: An application level multicast infrastructure. In *Proceedings of the 3rd USNIX Symposium on Internet Technologies and Systems*, Mar. 2001.

[30] G. Philips and S. Shenker. Scaling of multicast trees: comments on the Chuang-Sirbu scaling law. In *Proceedings of ACM SIGCOMM'99*, pages 41–51, Sept. 1999.

[31] P. Radoslavov, C. Papadopoulos, R. Govindan, and D. Estrin. A comparison of application-level and router-assisted hierarchical schemes for reliable multicast. In *Proceedings of IEEE INFOCOM*, 2001.

[32] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Application-level multicast using content-addressable networks. In *Proceedings of NGC*, Nov. 2001.

[33] S. Shi and J. S. Turner. Routing in overlay multicast networks. In *Proceedings of IEEE INFOCOM*, June 2002.

[34] S. Shi, J. S. Turner, and M. Waldvogel. Dimensioing server access bandwidth and multicast routing in overlay networks. In *Proceedings of NOSSDAV'01*, June 2001.

[35] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. 2002.

[36] L. Wei and D. Estrin. The trade-offs of multicast trees and algorithms. *Proceedings of the International Conference on Computer Communications and Networks (ICCCN)*, 1994.

[37] S. Q. Zhuang, B. Y. Zhao, A. D. Joseph, R. H. Katz, and J. D. Kubiatowicz. Bayeux: An architecture for scalable and fault-tolerant wide-area data dissemination. In *Proceedings of NOSSDAV'01*, June 2001.