

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

Li Lao<sup>1</sup>, Jun-Hong Cui<sup>2</sup>, Mario Gerla<sup>1</sup>, Dario Maggiorini<sup>3</sup>

<sup>1</sup> Computer Science Department, University of California, Los Angeles, CA 90095

<sup>2</sup> Computer Science & Engineering Department, University of Connecticut, Storrs, CT 06029

<sup>3</sup> Computer Science Department, University of Milano, via Comelico 39, Milano, Italy

**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) 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 four representative protocols, namely, PIM-SSM, NARADA, NICE, and POM 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?

## I. 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. Initially, multicast is implemented at the IP layer (i.e., IP multicast [13]), in which a tree delivery structure composed of network routers is usually employed, with data packets only replicated at branching nodes ([26], [24], [14]). 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 practical pricing models, it is still far from being widely deployed [2], [15].

To resolve the deployment issues of IP multicast, application layer multicast ([11], [17], [4], [27], [20], [23], [29], [34], [7]) and overlay multicast ([9], [19], [10], [31], [30], [5]) have been proposed. In both approaches, certain nodes form a virtual network, and multicast delivery structures are constructed on top of this virtual network. Data packets are replicated at participating nodes and delivered through tunnels. However, there are significant differences between these two approaches. 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. 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 upper layer architecture should we choose in which scenario?

In this paper, we strive to answer these questions. We compare their performance with respect to a variety of metrics through extensive simulation studies. 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 II, we give an overview on the three multicast architectures and provide a qualitative comparison. We describe our experimental methodologies in Section III. Then we present the simulation results in Section IV. Finally, we briefly review related work in Section V and conclude our work in Section VI.

## II. MULTICAST ROUTING PROTOCOL OVERVIEW

In this section, we provide a brief overview on the multicast routing protocols and give a qualitative comparison. Fig. 1 gives a high level illustration of the three multicast architectures, namely, IP multicast, application layer multicast, and overlay multicast.

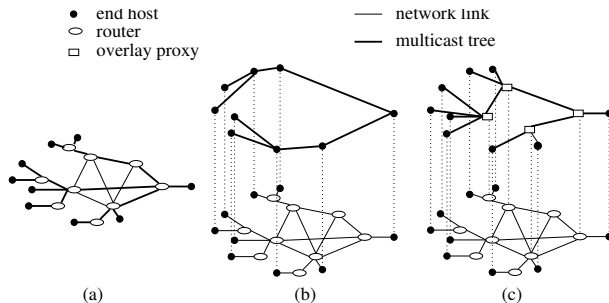


Fig. 1. An illustration of (a) IP multicast, (b) application layer multicast, and (c) overlay multicast.

### A. 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. The most well-known IP multicast routing protocols are Distance Vector Multicast Routing Protocol (DVMRP) [26], Multicast Open Shortest Path First (MOSPF) [24], Core-Based Trees (CBT) [3], Protocol Independent Multicast (PIM) with Dense and Sparse Mode [14], and PIM-SSM [18]. Among these protocols, DVMRP, MOSPF, PIM-DM, and PIM-SSM construct source-based trees (i.e., building a multicast tree for each source), whereas CBT and PIM-SM utilize a core or Rendezvous Point (RP) to organize multicast trees.

As mentioned earlier, despite its bandwidth efficiency, IP multicast usually suffers from a number of deployment issues. Thus, even till now, MBONE is the only existing while “decaying” global IP multicast testbed.

### B. 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 delivery structure construction, and data forwarding are solely controlled by participating end hosts, thus it does not require the support of intermediate nodes (e.g., routers or proxies).

In general, we can divide application layer multicast protocols into two categories: structured ([29], [34], [7]), and unstructured ([11], [4], [27], [20], [23], [17]). 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 [11], NICE [4], ALMI [27]), or directly construct a multicast tree (e.g., Yoid [17]) or other well-known topologies (such as hypercubes and Delaunay triangulations in HyperCast [23]).

TABLE I  
COMPARISON OF MULTICAST ARCHITECTURES

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

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 delivery structures among the end users and to monitor network conditions. Thus, it is commonly believed that these issues pose a lot of challenges when multicast groups are large.

### C. Overlay Multicast (OM)

Alternatively, multicast functionalities can be supported by specially deployed intermediate proxies (or service nodes), which cooperatively construct a “backbone overlay” infrastructure and establish multicast trees (or other structures) among themselves for data delivery. Outside the backbone overlay, the proxies can deliver data packets to end hosts via multicast or unicast. Recently, many seminal works have been conducted, such as Scattercast [9], Overcast [19], RMX [10], AMCast ([31], [30]), and OMNI [5].

Overlay multicast more or less “plays the game” in between IP multicast and application layer multicast, thus combining the benefits of both approaches. First of all, it implicitly gains knowledge about the network topology by using proxies: end users can naturally form clusters around closest proxies without complicated measurement techniques and clustering algorithms; the backbone overlay can be strategically dimensioned to resemble the physical network. Therefore, it can easily improve multicast efficiency and reduce resource (such as bandwidth) usage. Second, the limited size of the backbone overlay and local clusters 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 overlay only supports one group and/or one application, backbone overlay can support multiple groups and applications simultaneously, and the control overhead for probing and maintaining the backbone is not affected by the number of co-existing groups supported. Consequently, when there are a large number of groups, overlay multicast is expected to out-perform application-layer multicast in terms of control overhead. Nevertheless, 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 discus-

sions above, we present a high-level qualitative comparison in Table I, illustrating the pros and cons of IP multicast, application layer multicast and overlay multicast. In Section IV, we will give a quantitative comparison by extensive simulations.

### III. 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 studies.

#### A. Topology Graphs

We use router-level ISP maps measured by an Internet mapping tool called Rocketfuel [32] and AS (Autonomous System) level network topologies obtained by Route Views Project [1], since AS- and router-level graphs exhibit different characteristics such as network size and node degree distribution, and they allow us to evaluate the potential of using multicast protocols in a hierarchical fashion at both inter- and intra-domain levels. Due to space limitation, however, in this paper, we mainly present the results for a router-level topology with approximately 300 nodes within an ISP called Ebone. Interested readers are referred to our technical report [21] for results on AS graphs. For simplicity while without affecting the fairness of comparison, we assume all links have same delay and infinite bandwidth capacity.

#### B. Group Membership Generation

In this study, group members are randomly uniformly distributed in the network<sup>1</sup>. 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, and a source is randomly selected from members for every group. Thus, multicast traffic is mainly determined by the location of group members and source. The members join groups at the first 400 seconds of the simulation time, and the performance metrics are collected after 1000 seconds. The results for each group size are averaged over multiple groups (i.e., from 20 to 100 groups depending on group size).

#### C. Multicast Protocols for Evaluation

We compare four representative multicast routing protocols. All of them use source-based tree approach, since core-based tree approach requires additional parameters (such as the selection of cores) and most application layer multicast protocols use source-based tree approach. However, it is worth investigating the impact of core selection on the performance of different multicast protocols in the future work.

<sup>1</sup>It has been shown in [12] that in the Internet, group members may be uniformly or non-uniformly distributed depending on applications. The impact of non-uniform group member distribution on multicast performance is currently under investigation.

**IP multicast:** For IP multicast, we implement PIM-SSM protocol, which is believed to be more resource efficient and scalable 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 [11], 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 [4], 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. Both NARADA and NICE require a special Rendezvous host to bootstrap incoming hosts into the mesh structure.

**Overlay Multicast:** We implement a simple while generic overlay multicast protocol, called Pure Overlay Multicast (POM). It manages overlay multicast trees among the proxy nodes inside the backbone overlay. End users connect to proxies via unicast connections. In fact, POM can be treated as a simple abstraction of various overlay multicast proposals, such as OMNI, AMCast, Overcast, and Scattercast. The techniques employed in these proposals could improve POM's performance significantly. In this paper, our goal is to explore the basic trade-offs between different types of multicast protocols. Thus, we tend to make the overlay multicast protocol as simple and generic as possible. Note that POM also requires a bootstrap mechanism for users to join group.

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. Then each proxy periodically exchanges link state information (obtained by measurement) with each other. Finally, on top of this complete graph, overlay links for data delivery are then established according to a so-called "Adjacent Connection" method ([25], [22]): an overlay link is established between two proxies if this path does not go through other proxies. Since the number of proxies is a parameter, we will also examine the impact of this parameter on the performance of POM.

It is worth pointing out that though PIM-SSM, NARADA, NICE, and POM all need some kind of mechanisms to discover sources or bootstrap nodes for users to join the group, in this paper, we do not compare these mechanisms (which also deserve further research). Instead, we focus our efforts on several key aspects of different multicast routing protocols such as multicast tree quality and control overhead.

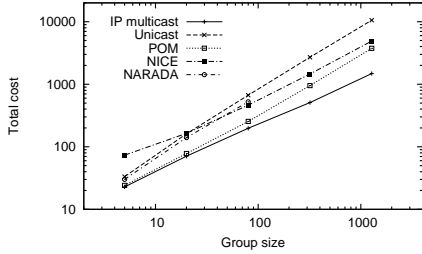


Fig. 2. Multicast tree cost with varying group size.

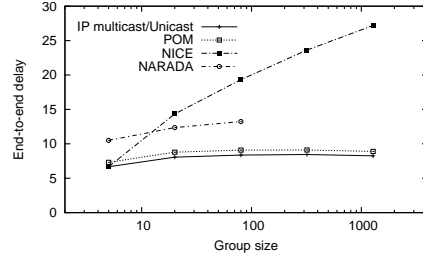


Fig. 3. Average end-to-end delay with varying group size.

#### D. Performance Metrics

We evaluate the performance of multicast protocols from the following aspects (additional simulation results on load distribution and robustness can be found in [21]).

**Multicast tree quality:** Two 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.

**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, and POM).

### IV. 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 15 nodes with degree higher than 10 as proxies for POM. In the remaining section, we present our simulation results.

#### A. Multicast Tree Quality

We plot the results of multicast tree cost and unicast cost in Fig. 2. Since NARADA is designed for small groups, we only show its performance for group size up to 80. In Fig. 2, for all kinds of group size, IP multicast has the lowest tree cost, application layer multicast has the highest cost, and overlay multicast lies in between. This is consistent with our earlier discussion in Section II. 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, every node maintains global group membership information and periodically improves mesh quality by probing and adding or dropping links. This feature allows it to construct multicast trees with high quality; 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 POM, when the group size becomes larger, multicast tree cost increases faster than IP multicast, and also a little bit faster than NICE. This is because unicast is used in POM for the communication between end users and proxies. The scalability of POM can be improved by employing IP or application layer multicast outside the backbone overlay.

Fig. 3 depicts the end-to-end delay of multicast trees. IP multicast yields the same delay as unicast, as shortest path trees are used in PIM-SSM. The delay of 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. 3 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 POM, they comparatively sit at better trade-off points between the tree cost and end-to-end delay.

#### B. Control Overhead

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

We plot the results of control overhead for single groups in Fig. 4. 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; but its overhead exceeds that of overlay multicast when group size increases beyond a

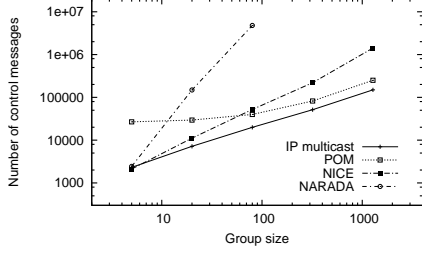


Fig. 4. Control overhead with varying group size.

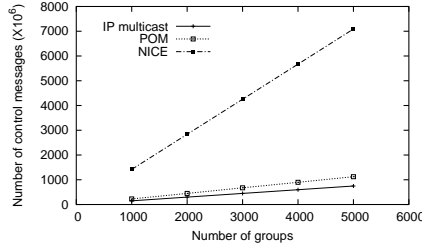


Fig. 5. Control overhead for multiple groups with group size 1280.

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. On the other hand, in application layer multicast protocols, the control overhead is mainly determined by group size. Consequently, the slope of the POM curve is much smaller than that of application layer multicast curves.

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 are a large number of groups. In Fig. 5, we plot the control overhead for group size of 1280 when the number of groups varies. It is clear that when there are numerous groups, overlay multicast out-performs application layer multicast by a large margin.

### C. Impact of the Number of Overlay Proxies

In previous experiments, we fix the number of overlay proxies. Now we investigate the impact of this parameter on the performance of overlay multicast protocols. Fig. 6 and 7 depict multicast tree cost and control overhead of POM as the number of proxies increases for different group sizes. It is clear that a larger number of proxies help 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

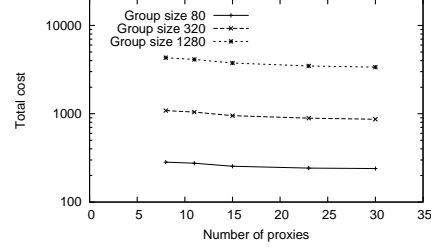


Fig. 6. Total cost of POM vs. number of proxies.

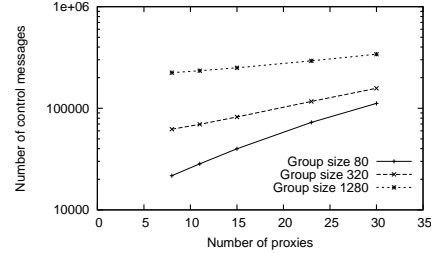


Fig. 7. Control overhead of POM vs. number of proxies.

more proxies can also decrease average end-to-end delay. Accordingly, network designers can tune this parameter to balance the application performance and overhead.

### D. 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 POM have better trade-offs between multicast tree cost and end-to-end delay than application layer multicast; 2) NARADA is more resource efficient than NICE when the group size is small, 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 delay increases rapidly with group size; 3) for single groups, POM 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; 4) 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. Nonetheless, 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”, while application layer multicast is good for immediate deployment, especially for small groups.

## V. RELATED WORK

In the literature, there have been many seminal works on multicast performance evaluation. Early works, such as [33] and [6], mainly focus on the comparison of IP multicast protocols. 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 [28]. Fahmy and Kwon characterize application level multicast trees, such as NARADA and TAG, using numerical analysis and simulations [16]. Castro et. al. evaluate DHT (Distributed Hash Table) based application level multicast protocols, including CAN and Pastry [8]. 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 routing protocols implemented on different layers of the protocol stack.

## VI. CONCLUSIONS

In this paper, we conduct a comparative study of different types of multicast routing protocols: IP multicast, application layer multicast, and overlay multicast, both qualitatively and quantitatively. Based on our studies, we could answer the three important questions posed in Section I: 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 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, C. Kommareddy, and B. Bhattacharjee. Scalable application layer multicast. In *Proceedings of ACM SIGCOMM*, Aug. 2002.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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. 2003.
- [9] 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/>.
- [10] Y. Chawathe, S. McCanne, and E. A. Brewer. RMX: Reliable multicast for heterogeneous networks. In *Proceedings of IEEE INFOCOM*, Mar. 2000.
- [11] Y.-H. Chu, S. G. Rao, and H. Zhang. A case for end system multicast. In *Proceedings of ACM Sigmetrics*, June 2000.
- [12] J.-H. Cui, M. Faloutsos, D. Maggiorini, M. Gerla, and K. Boussetta. Measuring and modelling the group membership in the Internet. In *Proceedings of IMC 2003*, Oct. 2003.
- [13] S. Deering. Multicast routing in a datagram internetwork. *Ph.D thesis*, Dec. 1991.
- [14] 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*, Apr. 1996.
- [15] C. Diot, B. Levine, J. Lyles, H. Kassem, and D. Balensiefen. Deployment issues for the IP multicast service and architecture. *IEEE Network*, Jan. 2000.
- [16] S. Fahmy and M. Kwon. Characterizing overlay multicast networks. In *Proceedings of IEEE ICNP*, Nov. 2003.
- [17] P. Francis. *Yoid: Extending the Multicast Internet Architecture*. White paper, <http://www.aciri.org/yoid/>.
- [18] H. Holbrook and D. Cheriton. IP multicast channels: EXPRESS support for large scale single-source applications. *Proceedings of SIGCOMM'99*, Sept. 1999.
- [19] 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.
- [20] M. Kwon and S. Fahmy. Topology aware overlay networks for group communication. In *Proceedings of NOSSDAV*, May 2002.
- [21] L. Lao, J.-H. Cui, M. Gerla, and D. Maggiorini. A comparative study of multicast protocols: Top, bottom, or in the middle? Technical report, UCLA CSD TR040054. <http://www.cs.ucla.edu/NRL/hpi/papers.html>, Jan. 2005.
- [22] Z. Li and P. Mohapatra. The impact of topology on overlay routing service. In *Proceedings of IEEE INFOCOM*, Mar. 2004.
- [23] 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.
- [24] J. Moy. Multicast routing extensions to OSPF. *RFC 1584*, Mar. 1994.
- [25] A. Nakao, L. Peterson, and A. Bavier. A routing underlay for overlay networks. In *Proceedings of SIGCOMM*, Aug. 2003.
- [26] C. Partridge, D. Waitzman, and S. Deering. Distance Vector Multicast Routing Protocol. *RFC 1075*, 1988.
- [27] D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel. ALMI: An application level multicast infrastructure. In *Proceedings of the 3rd USENIX Symposium on Internet Technologies and Systems*, Mar. 2001.
- [28] 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.
- [29] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Application-level multicast using content-addressable networks. In *Proceedings of NGC*, Nov. 2001.
- [30] S. Shi and J. S. Turner. Routing in overlay multicast networks. In *Proceedings of IEEE INFOCOM*, June 2002.
- [31] S. Shi, J. S. Turner, and M. Waldvogel. Dimensioning server access bandwidth and multicast routing in overlay networks. In *Proceedings of NOSSDAV'01*, June 2001.
- [32] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *Proceedings of ACM SIGCOMM*, Aug. 2002.
- [33] L. Wei and D. Estrin. The trade-offs of multicast trees and algorithms. *Proceedings of ICCCN*, 1994.
- [34] 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.