Design of a Scalable Multicast Scheme With an Application-Network Cross-Layer Approach

Xiaohua Tian, *Student Member, IEEE*, Yu Cheng, *Senior Member, IEEE*, and Bin Liu, *Member, IEEE*

***Abstract*—This paper develops an efficient and scalable multicast scheme for high-quality multimedia distribution. The traditional IP multicast, a pure network-layer solution, is bandwidth efficient in data delivery but not scalable in managing the multicast tree. The more recent overlay multicast establishes the data-dissemination structure at the application layer; however, it induces redundant traffic at the network layer. We propose an application-oriented multicast (AOM) protocol, which exploits the application-network cross-layer design. With AOM, each packet carries explicit destinations information, instead of an implicit group address, to facilitate the multicast data delivery; each router leverages the unicast IP routing table to determine necessary multicast copies and next-hop interfaces. In our design, all the multicast membership and addressing information traversing the network is encoded with bloom filters for low storage and bandwidth overhead. We theoretically prove that the AOM service model is loop-free and incurs no redundant traffic. The false positive performance of the bloom filter implementation is also analyzed. Moreover, we show that the AOM protocol is a generic design, applicable for both intra-domain and inter-domain scenarios with either symmetric or asymmetric routing.**

***Index Terms*—Application oriented networking, bloom filter, cross-layer design, multicast, scalability.**

I. Introduction

PROVISIONING high-quality multimedia services (e.g., online multiplayer games, IPTV, and video conferencing) to a large number of subscribers, possibly located in a vast geographic area, requires a scalable and efficient multicast scheme to disseminate the shared data to a group of destinations. In this paper, we propose a protocol-independent multicast scheme based on an application-network cross-layer design, which is scalable in routing, forwarding, and address allocating.

The traditional multicast solutions are implemented at the *network layer*, where the IP routers need to communicate with each other to construct and maintain a tree structure according to a distributed multicast routing algorithm [1]–[5]. Although various multicast protocols, e.g., *dense mode* protocols [3], [6], [7], *sparse mode* protocols [4], [8], and inter-domain protocols [9], [10], have been proposed to reduce the messaging overhead and the amount of states at routers for enabling a single group, the messaging overhead and the memory cost grow linearly with the number of multicast groups being supported by the router, leading to the *scalability issue*[10]. The unscalable implementation hinders IP multicast to be an efficient transport scheme for delivering multimedia applications over the Internet, where a huge number of groups need to be supported.

The emergence of overlay networks provides another alternative multicasting approach, where trees or other delivery structures are constructed at the *application layer* [11], [13], [14]. Each link in the overlay network is an end-to-end logic connection between two end hosts. Overlay multicast is increasingly popular as the underlying unicast infrastructure needs no modification. Nevertheless, overlay multicast performs much less efficiently than IP multicast in bandwidth utilization, as it is not a rare case that separate overlay links pass through common physical links in the underlying transport network.

The long-lasting issue that neither the network-layer nor the application-layer approach itself can achieve a generic scalable multicast solution reveals that multicasting by nature incurs an *application-network cross-layer* design problem. Specifically, identifying the users associated with a multicast group requires application-layer membership management, while delivering data to the proper destinations needs network-layer support according to the application-layer membership information.

Some multicast studies in the literature implicitly take the cross-layer approach. The recursive unicast approach (REUNITE) [15], [16] maintains some destinations information at the branching nodes of the multicast tree, so that the forwarding table size in these routers can be significantly reduced. However, REUNITE still requires per-group forwarding entries, which leads to the scalability issue, and induces large message overhead to refresh the destination information maintained in the branching nodes. The free ride multicast (FRM) [17] takes a source-routing manner, where the inter-domain multicast tree is coded into the packet header for protocol-independent multicasting. Although the FRM scheme well exploits the existing unicast infrastructure, the transmission overhead and the forwarding false positive rate will dramatically increase along with the size of the multicast tree.

In this paper, we interpret the application-network cross-layer design as *incorporating application intelligence into the network*, based on which we propose an application-oriented multicast (AOM) protocol. The basic idea of AOM is to make the packet carry the explicit destination addresses in its header, so that the routers (with application intelligence) can retrieve the addresses and leverage the unicast IP routing table to determine necessary multicast copies and the corresponding forwarding

Manuscript received October 01, 2008; revised May 27, 2009. Current version published September 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Aggelos K. Katsaggelos.

X. Tian and Y. Cheng are with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL 60616 USA (e-mail: xtian3@iit.edu; cheng@iit.edu).

B. Liu is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: liub@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TMM.2009.2026104

interface for each copy, without establishing and maintaining any separate multicast tree. However, the fundamental issue is that we must limit the bandwidth overhead for such explicit addressing; it is impractical to attach all the destination addresses to each packet. We have presented a preliminary service model of AOM in [18]. In this paper, we develop a bloom filter based design to make the AOM practical. We further prove theoretically that the AOM service model is loop-free and incurs no redundant traffic. The false positive performance of the bloom filter implementation is also analyzed. Moreover, we show that the AOM protocol is a generic design, applicable for both intra-domain and inter-domain scenarios with either symmetric or asymmetric routing.

In addition, the proposed AOM incorporates the feature of *localized group identifier (ID) allocation*, which can further enhance its scalability and management flexibility in supporting a large number of multimedia communication groups. In most of the existing multicast protocols, an active group is uniquely identified by an IP address, i.e., a class-D address according to IPv4. Such a global address allocation scheme is neither scalable nor flexible when the number of active groups grows significantly. The global address allocation implies that the total number of active multicast groups is limited by the class-D address space. Even if the address space may not be a problem under IPv6 [12], hooking up the group ID with routing is very inflexible. For example, if a multimedia provider tends to expand its channel list or adjust the multicast addresses allocated to some channels, all the routing information within the network has to be reestablished. In contrast, AOM enables a source-specific, localized group ID allocation scheme, and the group ID is further decoupled with multicast routing and forwarding for maximum management flexibility and scalability.

The remainder of this paper is organized as follows. In Section II, we further elaborate our perspective on application-network cross-layer design. Section III describes the service model of AOM. Section IV presents the bloom-filter based implementation for AOM. Section V gives theoretical investigations on the loop-free property, traffic redundancy, and false positive performance. Section VI presents some simulation and numerical results to demonstrate the performance of AOM. Section VII gives the conclusion remarks.

II. Application-Network Cross-Layer Perspective: Application-Oriented Networking

Integration of application intelligence into the network is not a brand new idea, which has been taken as an efficient approach, implicitly or explicitly, to implement some basic networking functionalities, develop new network protocols, or facilitate upper-layer applications. In the recent decade, enhancing network nodes with application-specific intelligence has become one of the mainstream ideas to design the next-generation Internet [19], [20], stimulated by various applications, including firewalls, Web proxies/caches, mobile gateways, service-oriented architectures, in addition to the multicast. However, all of these application-oriented solutions have been implemented in an ad hoc manner.

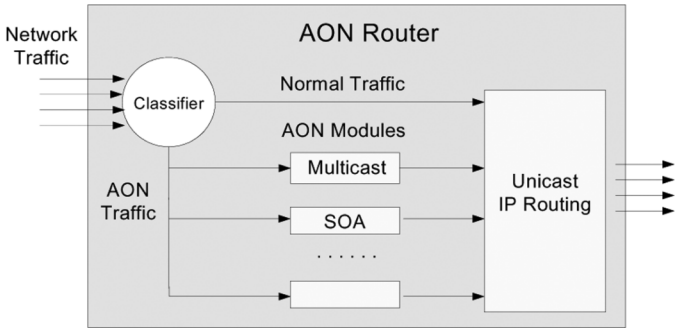

Fig. 1. Generic architecture for an AON router.

The *active network* [20], [21] was proposed in the mid-1990s as a generic architecture to provision programmability within the network, instead of those ad hoc approaches. In an active network, packets are replaced with *capsules*, which are program segments (possibly with embedded data) executable by an active network node. The active network has never been widely deployed; the main reasons include the large bandwidth overhead of carrying programs, lack of a common capsule program language, and the security issue due to users' active control capability.

The current trend of service consolidation over Internet protocol (IP) requires a more intelligent networking infrastructure that is able to respond quickly and cost-effectively to new market demands, which is one of the motivations leading to the Cisco application-oriented networking (AON) technology [23]. An AON-based network can transparently intercept the content and context of application messages, conduct operations on those messages according to business-driven policies and rules; all these are achieved by enhancing IP routers with application-layer intelligence.

We would like to emphasize that although the term AON was initiated by Cisco as a vendor-specific solution, we take a generic interpretation of AON from the application-network cross-layer perspective: *the IP devices can intercept and process application messages*. How to systematically exploit the AON capability to streamline the design of network functionalities is currently obscure in both industry and academia. In this paper, we initiate the study in this area by proposing an AON-based multicast scheme.

The generic architecture for an AON router can be designed as illustrated in Fig. 1. The incoming traffic will be first classified as *normal traffic*, which does not need application-level processing and is directly forwarded against the IP routing table, and *AON traffic*, which requires application-level processing before forwarded. The AON traffic will be further categorized and dispatched to different application-specific AON modules. We can select 1 bit in the IP header, e.g., one of the type-of-service (TOS) bits in the IPv4 header or one of the traffic class (TC) bits in the IPv6 header, to behave as the *normal/AON traffic indicator flag*. The flag is set to "1" for indicating the AON traffic. Although more TOS bits and TC bits may be used to further identify the AON modules, we prefer the fine-grained classification information to be carried in the payload, for higher scalability and flexibility.

III. AOM Service Model

The proposed AOM protocol adopts a source-based service model, which comprises the components of *membership management* and *forwarding protocol*. We use the inter-domain case for illustration purpose; the service model is also applicable to the intra-domain scenario.

A. Membership Management

For membership management, a border router of a stub autonomous system (AS) domain is selected as the *designated router* (DR). For convenience, we use RDR (SDR) to denote the DR of a receiver-side (source-side) AS domain. When multicast routing/forwarding is considered, we use RDR to represent the prefix associated with the corresponding receiver domain. The meaning of RDR will be clear in the context. The data source node in the domain of SDR is denoted as SRC.

The RDR basically needs to implement the Internet group management protocol (IGMP) [24] to discover the active groups within its domain. Each RDR may periodically, or triggered by special events, send *membership updating messages* (MUMs) to the SRC in the format as $(\mathrm{RDR} : \mathrm{GID}_1, \mathrm{GID}_2, \ldots, \mathrm{GID}_n)$, where RDR represents a domain prefix and GID represents the group identifier. The MUM will be delivered along the shortest path between the RDR and the SRC, indicated by the unicast routing table.

The SRC aggregates the MUM messages it received and maintains a *multicast group list* (MGL). For each group provisioned by the source, the MGL establishes a record in the format as $(\mathrm{GID} : \mathrm{RDR}_1, \mathrm{RDR}_2, \ldots, \mathrm{RDR}_n)$, where each RDR again indicates a domain prefix. When the SRC sends data over a certain group, it will insert the corresponding MGL into the packet as the destination information in the format of a shim header. The multicast packets are then forwarded to the SDR for inter-domain multicasting.

We use an example as shown in Fig. 2 to illustrate the membership management. Router A is the SDR; B and C are border routers of the transit domains (TBRs); D, E, and F are three RDRs. The MUM messages from the three RDRs will be propagated via unicast to the SRC, where the information is aggregated to a MGL as shown in the figure.

B. Multicast Forwarding Protocol

The multicast forwarding is facilitated by the AON technique. All routers (including SDRs, RDRs, and TBRs) are assumed to be AON routers. When receiving a multicast packet, each AON router will extract the MGL record from the packet. With the list of destination RDRs available from the MGL record, the AON router will check its IP routing table to determine the output interface to each RDR and make necessary aggregation.

As in Fig. 2, the IP routing table of the SDR A tells that the output interface 1 is on the path to both RDR D and RDR E, so only one copy is necessary to be forwarded via interface 1. The IP routing table also shows that another copy should be forwarded via interface 2 to reach RDR F. When the input multicast packet is replicated and put onto each output interface, the MGL record attached to each copy is updated correspondingly to include only the destination RDRs that can be reached via that interface. For example, the MGL record in the packet

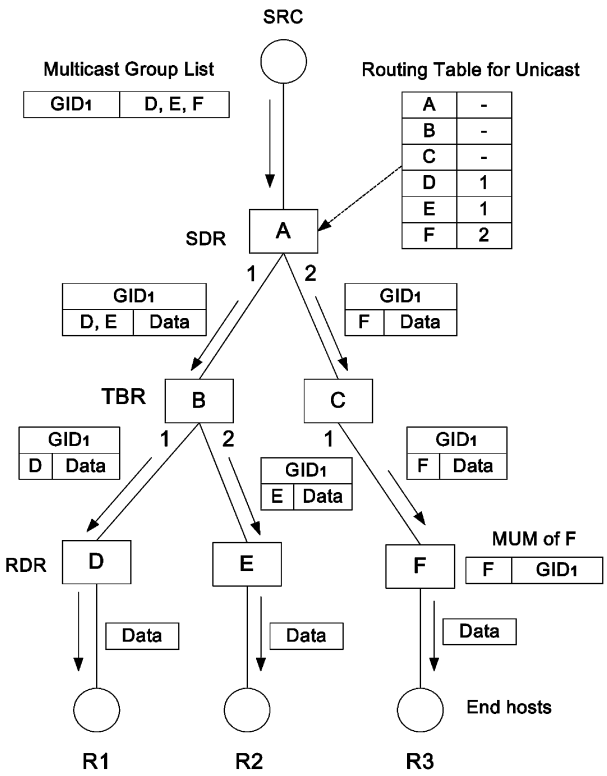

Fig. 2. AOM service model.

delivered over A's interface 1 includes only RDRs D and E. *By removing unnecessary addresses from the MGL record, the downstream router will not generate unnecessary packet copies for those destinations that have been delivered over other sibling subtrees.* Each AON router will execute the same operations of aggregation, replication, and MGL record updating, until one multicast packet reaches an RDR.

IV. Bloom Filter Based Implementation

The protocol-independent AOM model cannot be directly applied without elaboration. It is obvious that the MUM and the MGL will become impractically long, when a large number of groups are active in a large number of destination domains. This section presents a streamlined bloom filter based design to achieve the AOM with reasonable cost. We take the assumption of symmetric unicast routing for the convenience of demonstration, and discuss later how to extend AOM for operating in asymmetric routing scenarios.

A. Bloom Filter Data Structure

We are to describe the bloom filter based data structure for AOM according to the upstream procedure and downstream procedure, as illustrated in Fig. 3, where bloom filters are illustrated as shadowed areas. The group source node address is in the destination IP field for the upstream packet header, and in the source IP field for the downstream packet header. Due to the explicit context, in all the figures, we ignore the header field containing the group source IP address and the default multicast data payload for not cluttering the illustration.

The left part of Fig. 3 shows how upstream messages are processed. To reduce the bandwidth overhead for membership

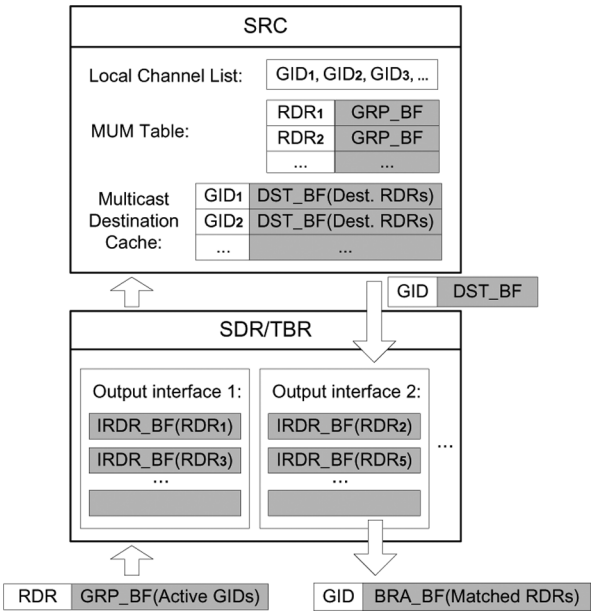

Fig. 3. Bloom filter data structure for AOM.

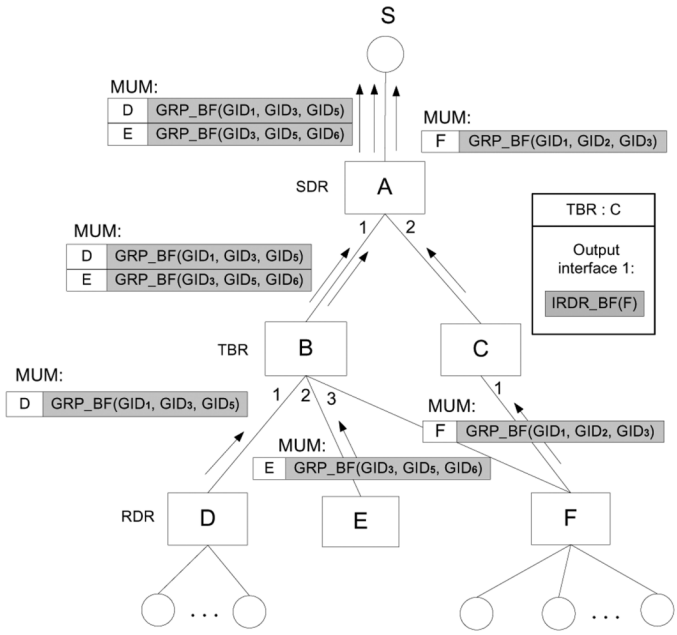

Fig. 4. Membership updating process of AOM.

registration or updating, the list of active groups in the MUM message is encoded with a *group bloom filter* (GRP_BF). When an MUM message reaches an upstream TBR/SDR router, the router will retrieve the RDR prefix, and store it as a local forwarding state that leads to a reverse path of the MUM incoming interface. By continuously observing the MUMs, each related interface of the TBR/SDR will memorize all the destination domains that can be reached through it. At the interface, each RDR is stored as a separate bloom filter, termed as *interface RDR bloom filter* (IRDR_BF). The IRDR_BF will be used to facilitate multicast forwarding.

The upstream MUM messages will finally reach the SRC node, and each message will be stored as a record of an MUM table. The SRC node should have a *local channel list* indicating the multicast groups it provisions. By checking each GID against the MUM table and identifying the matched GRP_BF, the SRC can detect the destination prefixes for a given group. The destinations information under the group ID will be encoded into a *destination bloom filter* (DST_BF) and stored into the multicast destination cache. Note that the DST_BF in fact encodes the MGL according to the AOM service model.

The right part of Fig. 3 illustrates how downstream multicast packets are forwarded. At the SRC node, the DST_BF for a group will be inserted as the destination information into each multicast packet. At the SDR and each downstream TBR, the DST_BF will be checked against the IRDR_BFs at each output interface to implement the aggregation, replication, and MGL record updating operations defined in the AOM service model. The subset of prefixes associated with each output interface are determined and re-encoded into the *branch bloom filter* (BRA_BF). The BRA_BF will be inserted into the packet copy delivered through that interface, serving as the destination information DST_BF for further downstream forwarding.

B. Membership Updating and Forwarding

We here present the membership updating procedure and the downstream data forwarding procedure, based on which the theoretical analysis of loop-free property, traffic redundancy, and false positive rate in next section can be understood.

1) Membership Updating: The membership updating procedure is illustrated in Fig. 4. The membership updating messages are periodically sent from RDRs to sources by leveraging the unicast mechanism. The periodic updating is normally used to refresh the states of active groups. The MUM updating can also be triggered when new active groups show up. We will evaluate the control message overhead at different updating frequencies in Section VI. In Fig. 4, the path *F-C-A-S* shows the joining process of RDR *F*.

2) Downstream Data Forwarding: The downstream message processing is illustrated in Fig. 5, which captures the scenario that a multicast packet for group 3 is at router *B*. Note that although the DST_BF carries the destinations information, we cannot just directly check the unicast forwarding entry against DST_BF to determine the output interfaces. The reason is that the unicast routing table normally applies the route aggregation and the longest prefix matching, particularly in the inter-domain context, where the accurate RDR prefix information may not be available to query the DST_BF. The local IRDR_BFs stored at each interface are important components to enable the AOM.

The AOM routing/forwarding is implemented through comparing the DST_BF against the local IRDR_BFs, which requires that the SRC node and the network agree on the size of the bloom filters and the set of hush functions involved. For example, given that 5 hush functions are used, if the comparison between the DST_BF and an IRDR_BF confirms 5 matched "1" bits, we define that a matched IRDR_BF is identified. Each interface containing matched IRDR_BF(s) represents a tree branch according to the *reverse-path-forwarding* concept, over which a separate copy of the data packet will be delivered.

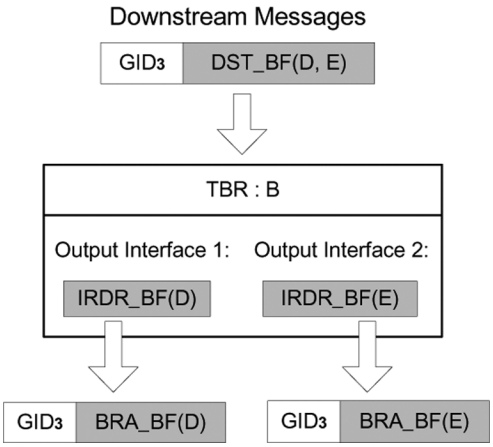

Fig. 5. Multicast forwarding processing at router B.

Moreover, a BRA_BF will be generated by aggregating all the matched IRDR_BFs (through an "OR" operation) over that interface. As mentioned in Section IV-A, the BRA_BF will be used as a DST_BF for further downstream forwarding.

We would like to emphasize that replacing the incoming DST_BF with the corresponding BRA_BF over each output interface achieves the aggregation, replication, and MGL record updating operations defined in the AOM service model, which will successfully avoid redundant traffic according to Section III-B. In Fig. 5, the incoming DST_BF contains both D and E as destinations. With interfaces 1 and 2 being determined as forwarding interfaces, the BRA_BF over each interface removes the destination processed by the other branch. It would be remiss not to mention that matching the DST_BF against the IRDR_BFs may generate false positives, which will be analyzed in detail in the next section.

C. Generic Properties

1) Uniformed Intra-/Inter- Domain Solution: The AOM protocol can be applied in a uniformed manner for both intra-domain and inter-domain multicast. For a transit or stub domain, no matter what intra-domain routing protocols are running, the MUM messages entering the domain may label the IRDR_BFs not only at the border routers but also at the corresponding core routers within the domain. According to our forwarding design presented above, it can be seen that the downstream packets will find the end-to-end path according to the reverse-path-forwarding operation. As a comparison, the FRM [17] only codes the inter-domain multicast tree in the packet header, so it requires additional intra-domain multicast scheme to achieve a complete multicast solution. If a transit domain is not equipped with an intra-domain multicast protocol, then N-unicast or broadcast has to be used to handle the transit-domain FRM traffic, which will lead to significant amount of redundant traffic.

2) Asymmetric Routing Scenario: The AOM implementation can also be readily extended to asymmetric routing scenarios. As the inter-domain routing is normally policy based [26], a border router will be aware of the asymmetric routing policy applied to a given destination domain. In case an SDR receives an MUM in the asymmetric case, in addition to forwarding the MUM to the source node, the SDR will further pass the MUM along the downstream path to the destination RDR. The local IRDR_BFs will be installed at the corresponding output interfaces along the path from the SDR to the RDR. The MUM returned back to the destination RDR also behaves as a notification signal that asymmetric routing is the policy and the multicast forwarding states have been established successfully. As an extra step, the RDR then sends a new type MUM (F: asymmetric) to the SRC to remove the local IRDR_BFs that had been established under the symmetric assumption. As long as the local IRDR_BF states are correctly established, other AOM implementation details apply to both the symmetric and asymmetric cases.

3) Services Decoupled From Routing: The bloom filter based AOM implementation demonstrates that AOM readily supports a flexible and scalable group ID allocation in the form of a two tuple (source node address, source-specific channel ID) [15], which breaks the address space limitation and brings significant management flexibility. It even allows a logical local channel ID rather than an IP-address based channel ID. Furthermore, the AOM decouples the membership management component from the multicast forwarding component. Specifically, with AOM, the forwarding component at a router just requires RDR related information. Group IDs are only used for labeling groups at the SRC and RDRs to establish the service relationship as shown in Fig. 3. In case that an SRC updates its channel list or wants to upgrade the services of existing channels, it just sends related service information to each destination domain over the established multicast tree. As long as the RDRs and active members tune to the new channel list, service starts immediately. In other words, the service upgrading or rearrangement could be implemented seamlessly, where the established multicast infrastructure does not need any extra operation.

V. Theoretical Analysis

A. Loop-Free Forwarding Without Redundant Traffic

The AOM protocol based on the service model presented in Section III ensures loop-free forwarding and incurs no redundant traffic. We have the following theorems.

Theorem 1: The AOM downstream forwarding is free from directed cycles if the following conditions hold: 1) the domains associated with the SDR and RDRs are stub domains of the multicast group under consideration; 2) the unicast routing in the network is stable; and 3) the bloom filter implementation does not incur false positive (the false positive performance is to be analyzed separately).

Proof: Consider a network supporting multiple multicast groups. We assume that the AOM may lead to directed cycles, and then derive the contradictions. Due to the uniformed solution of AOM for both intra- and inter- domain cases, we here give the proof regarding the border routers for convenience. We can see that for a given multicast group, a directed cycle or a forwarding loop can only appear as two cases.

Case 1) The directed cycle takes the form $(\mathrm{TBR}_i \rightarrow \mathrm{RDR}_j \rightarrow \cdots \mathrm{TBR}_k \cdots \rightarrow \mathrm{RDR}_l \rightarrow \mathrm{TBR}_i)$. In this case, after a packet reaches the destination RDR, it will be further forwarded to other RDRs or TBRs and result in the

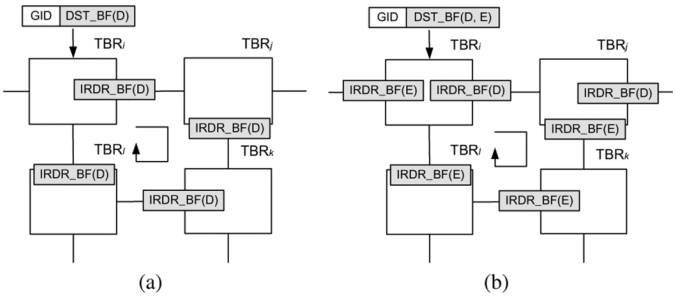

Fig. 6. Illustration of directed cycles.

loop, which obviously conflicts with the condition 1) that the RDRs are stub domains of the multicast group under consideration where further packet delivery for this group should stop.

Case 2) The directed cycle takes the form $(\mathrm{TBR}_i \rightarrow \mathrm{TBR}_j \rightarrow \cdots \mathrm{TBR}_l \rightarrow \mathrm{TBR}_i)$. In this case, the directed cycle forms among TBRs within the network before reaching the destination domain. This case further includes two subcases.

For the first subcase, as illustrated in Fig. 6(a), all the interfaces consisting of the forwarding loop are labeled with the same IRDR_BF, which are supposed to be along the path to the destination. According to our reverse-path-forwarding design, the directed cycle in the downstream path implies that the upstream path taken by the MUM messages also contains directed cycle. However, in the AOM design, the upstream MUM messages exploit the existing unicast path, which is loop-free in stable state, and we get a contradiction.

For the second subcase, as illustrated in Fig. 6(b), the interfaces consisting of the forwarding loop are labeled with different IRDR_BFs. The IRDR_BF configuration in Fig. 6(b) is possible if the destination domains D and E submit to different groups through different path. According to the AOM design, if two neighboring links (termed as one upstream and one downstream) along a path are labeled with different IRDR_BFs, it means that the node containing the downstream link is a branching node that intersects different paths to different destinations. Note that the AOM protocol incorporates the MGL record updating operation, that is, after the branching node processing, the destinations associated with other branches will be removed from the DST_BF for those downstream packets. The MGL updating operation ensures that the forwarding loop as shown in Fig. 6(b) is impossible, if condition 3) holds.

Summarizing case 1 and case 2, the theorem is proved. ■

Theorem 2: The AOM downstream forwarding generates no redundant traffic under the same conditions as applied in Theorem 1.

Proof: Theorem 1 has proved that the directed cycle does not exist, so there is no redundant traffic caused by forwarding loops. Consider two additional cases which may potentially generate redundant traffic. Case 1 is that multiple copies of the same packet reach the same RDR from different paths. Case 2 is that some packets reach a TBR but will be dropped due to nonexistence of matching output interfaces at that TBR. According to the reverse-path-forwarding principle of AOM, case 1 implies that some MUM messages have labeled more than one paths, which is impossible under condition 2). Moreover, based on the MGL record updating operation and the condition 3), we can see that the redundant traffic indicated in case 2 is impossible either. ■

B. False Positives in Forwarding

1) False Positive on an Interface: In the AOM forwarding process, bit matching between the in-packet DST_BF and the local IRDR_BF may incur false positives. We can analyze the bit matching in a more general context. Assume two bloom filters BF_1 and BF_2 both are represented as m-bit arrays and generated by the same k hash functions. BF_1 and BF_2 contain n_1 elements and n_2 elements, respectively.

The bit-matching false positive may happen in three cases: 1) an element in BF_1 but not in BF_2 is positively detected in BF_2; 2) an element in BF_2 but not in BF_1 is positively detected in BF_1; 3) an element neither in BF_1 nor in BF_2 is positively detected in both of them.

The bloom-filter theory [28] tells that the false positive probability associated with BF_1 and BF_2 are $f_{n_1} = \left(1-(1-1/m)^{n_1 k}\right)^k$ and $f_{n_2} = \left(1-(1-1/m)^{n_2 k}\right)^k$, respectively. It is not difficult to see that the total probability for bit-matching false positive due to either case 1or case 2 can be expressed as

$$f_1 = 1-(1-f_{n_2})^{n_1} \cdot (1-f_{n_1})^{n_2}. \tag{1}$$

The bit matching false positive due to case 3 is

$$f_2 = f_{n_1} \cdot f_{n_2}. \tag{2}$$

Thus, the total bit-matching false positive rate is

$$f(n_1, n_2) = f_1 + f_2. \tag{3}$$

In AOM forwarding, the false positive rate can be computed with n_1 equal to the number of elements in the DST_BF and $n_2 = 1$.

2) False Positive Along a Path: In AOM, if a false positive, say for destination RDR_i, happens at a certain output interface along a path, it will persist until it reaches the destination. The reason is that the MGL updating operation will not remove the RDR_i from the DST_BF, although it is confirmed by false positive, and thus the packet will finish the path labeled by the IRDR_BF (RDR_i). Moreover, new false positives may happen in a downstream node due to other RDRs. It is noteworthy that the MGL updating operation has a side benefit to reduce the false positive along a path, because the items contained in the DST_BF or BRA_BF continuously become less. We consider that the inter-domain multicast tree can be modeled as a binary tree with height H [29]. For a given destination RDR, an upper

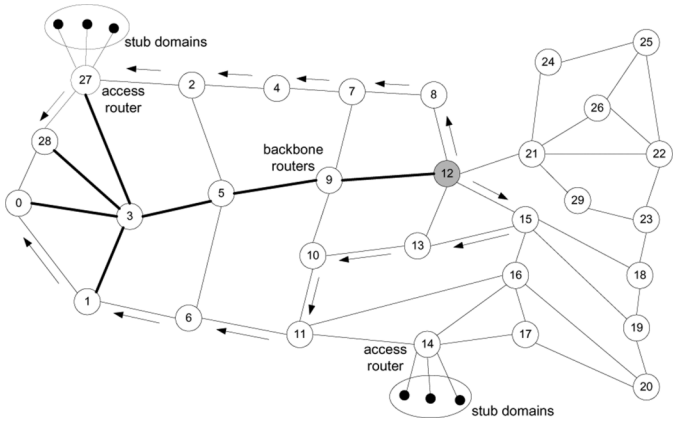

Fig. 7. Simulation topology.

bound of the probability that the RDR receives traffic by false positive, denoted as F_p, can be expressed as

$$F_p(H) = \sum_{h=1}^{H} f\left(\frac{n_1}{2^{h-1}}, 1\right) \tag{4}$$

where n_1 is the number of elements contained in the DST_BF generated by the SDR, and $n_1/2^{h-1}$ is the number of RDRs in the updated BRA_BFs at the layer-i TBR. We present the specific numerical analysis in Section VI-B to show the efficiency of AOM in terms of false positive.

VI. Performance Evaluation

In this section, we present some NS2 [30] simulation results and numerical analysis to demonstrate the efficiency of AOM on bandwidth utilization and small false positives.

The network topology for simulation is given in Fig. 7, which is widely used in the literature to approximate the U.S. backbone network [25]. In our model, each transit domain is represented as a backbone router or backbone node, and the backbone routers where stub domains are connected are termed as *access routers* or *access nodes*. The multicast source is located at node 12. We simulate multiple multicasting scenarios where the multicast group involves different numbers of access routers, and evaluate the bandwidth consumption across the backbone network. The scenario involving four access routers includes nodes $\{2, 4, 6, 8\}$. The node sets $\{11, 14, 17, 18\}$, $\{1, 19, 23, 27\}$, $\{0, 10, 24, 25\}$, and $\{13, 20, 28, 29\}$ will be added in turn to form the scenarios involving 8, 12, 16, and 20 access routers, respectively.

A. Bandwidth Overhead

The AOM bandwidth overhead is incurred by both membership updating and data forwarding. Since we adopt the method suggested by FRM [17] to update the group membership, we here mainly focus on the bandwidth overhead in data forwarding.

The AOM forwarding incurs bandwidth overhead due to two reasons. 1) Each packet needs to carry the destination information, i.e., the DST_BF, using a shim header. 2) When the destination entities are too many to be encoded in a single shim header, redundant packet copies have to be incurred, each of which carries a subset of destinations in its own shim header. The similar approach has been adopted by FRM to use multiple packets to carry a big source-routing tree. We fix the size of the shim header ($< 10\%$ of the packet size) for AOM and FRM, and measure the overhead in terms of the number of packets transmitted.

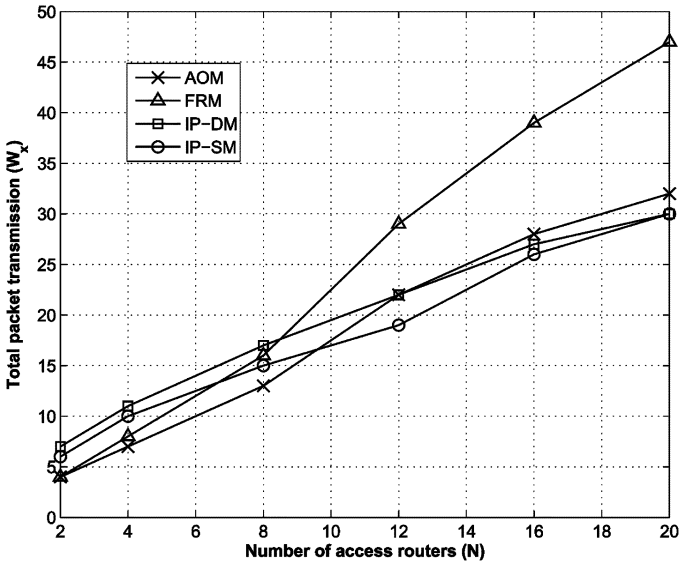

Fig. 8. Total number of packet transmissions, W_x.

1) Total Packet Transmission: The total number of packet transmissions W_x is defined as

$$W_x = \frac{T}{C} \tag{5}$$

where T denotes the total number of transmissions over the whole backbone network within the simulation duration, and C the total number of packets generated by the source. W_x represents the average number of transmissions over the whole network to multicast a single packet from the source to all the access routers.

Fig. 8 shows the values of W_x versus N, the number of access routers involved in multicasting, with different multicast schemes compared. The case $N = 2$ (node 4 and 8 involved) represents a multicast scenario with very sparse node distribution. We observe that, among all kinds of multicast schemes considered, AOM achieves the best performance when the number of access routers N is small or moderate. When N grows large, AOM still performs close to IP multicast. Except for the case of $N = 2$, AOM always outperforms FRM, especially when N is large and the multicast tree has many branches. Moreover, W_x increases along with the number of access routers; because each multicast packet has to travel over more links to reach more destinations, which incurs more transmissions across the network.

Compared with IP multicast dense mode (IP-DM), AOM in fact implicitly establishes a source-based tree from the IP routing table (referring to Section IV-B), which is the same as the tree constructed in IP-DM. However, AOM does not use the "broadcast-and-prune" [10] method to construct the multicast

tree as used by IP-DM. In the sparse scenarios with small N, the larger values of W_x are due to unnecessary transmissions during the periodical "broadcast-and-prune" operation. When N becomes large, IP-DM becomes favorable since multicast in a densely-populated network approaches broadcast, and few number of broadcast packets are wasted.

The reason for better performance of AOM over FRM is that AOM needs to encode less elements than FRM does [17]. For example, consider the case that node 12 multicasts data to access routers $\{0, 1, 27, 28\}$; the multicast tree is highlighted in Fig. 7 with thick lines. For illustration purpose, consider that the shim header can only encode four elements. In such a scenario, one shim header can encode all the four destinations under AOM. Under FRM, the seven-branch tree needs to be encoded, which exceeds the capacity of one shim header. Therefore, four shim headers over four packets have to be used, each containing the tree branches to one of the destinations. Three of such four packets are counted as redundant traffic compared to AOM. For the case of $N = 2$, the shim header is capable of containing the entire forwarding information for both AOM and FRM, so they have the same W_x. In addition, when the number of access routers grows, the size of the multicast tree increases at a faster rate; FRM then needs to use more redundant packets than AOM to encode the forwarding information, which explains why the performance of FRM increasingly deviates from that of AOM.

The performance of IP multicast sparse mode (IP-SM) is closely related to the selection of rendezvous point (RP). In the simulation, we select node 10 as the RP for IP-SM scheme. The efficiency of IP-SM compared to IP-DM in scenarios with sparse node distribution is clearly demonstrated in Fig. 8. In the mean time, IP-SM has higher W_x values than AOM and FRM in the sparse cases. The reason is that the data packets are first unicast to the RP and then disseminated to access routers from there, which causes some redundant transmissions.

2) Link Packet Transmission: The link packet transmission, L_x, is defined as

$$L_x = \frac{T_l}{C} \tag{6}$$

where T_l denotes the total number of data transmissions over link l within the simulation duration, and C the total number of packets generated by the source. L_x represents the average number of transmissions over a given link required to multicast a single packet from the source node to all the receiver domains, which is a good indicator of traffic load due to multicast and may be exploited for admission control [5], [27].

The distribution of L_x over those backbone links traversed by multicast packets is computed in two scenarios, with the number of access routers $N = 12$ and $N = 20$, respectively. The results are summarized in Table I for different multicast schemes. Specifically, the total number of links involved in multicasting, the L_x values, and the corresponding percentage of links (%) are listed. In both scenarios, about 90% of links see exact one transmission under AOM, and rest of the links observe two transmissions; the bandwidth efficiency is very close to that under the IP-SM scheme. The redundant traffic in AOM is incurred by splitting the destination set into smaller sub-sets to fit into the shim header. In FRM, the largest value of L_x is up to four and seven, respectively in the two scenarios, which is due to the multicast tree splitting for shim header coding.

TABLE I
LINK PACKET TRANSMISSION DISTRIBUTION

N	AOM		FRM		IP-DM		IP-SM	
	29 links		29 links		50 links		29 links	
	%	L_x	%	L_x	%	L_x	%	L_x
12			5.0	4				
			10.0	3				
	10.0	2	10.0	2	36.0	1	11.1	2
	90.0	1	75.0	1	64.0	< 1	88.9	1
20			3.4	7				
			3.4	5				
			3.4	4				
			6.9	3				
	10.3	2	3.4	2	54.0	1	7.1	2
	89.7	1	79.5	1	46.0	< 1	92.9	1

The L_x value under IP-DM may not be integers, since IP-DM periodically executes the "broadcast-and-prune" operation to establish the tree, and those pruned links only see the tree-constructing broadcast messages but no data packets. Table I shows that when the access routers become more densely populated, more links see data transmissions. Moreover, Table I shows that for the 29-link multicast tree, the "broadcast-and-prune" operation involves 50 links, which results in redundant traffic. In IP-SM, only the link between node 12 and 9 and that between node 9 and 10 observe two packet transmissions, since every packet should be unicast to the RP at node 10, before being multicast.

3) Control Message Overhead: The AOM control message overhead is caused by the MUM message, which is used to update membership as well as establish the routing states along the path from the access router to the source node. The control message overhead is determined by the MUM updating frequency, and extra overhead is incurred by the asymmetric inter-domain routing. As discussed in Section IV-C2, for asymmetric routing, the MUM message needs to travel upstream and then downstream to install IRDR_BFs on appropriate routers, and the RDR also needs to send an extra upstream message to remove the routing states constructed in the first upstream travel under the symmetric routing assumption. In the symmetric routing case, one run of upstream travel is enough.

The AOM control message overheads in both symmetric and asymmetric routing scenarios are examined in the simulation. Fig. 9 illustrates the total number of control messages across the network per second versus the MUM updating intervals, with the number of access routers $N = 20$. We change the weights of links to realize the asymmetric routing configuration. For example, the paths connecting node $\{0, 1, 27, 28\}$ with the source node 12 are marked in Fig. 7. In the symmetric routing case, the MUM messages go up and multicast data stream down along the links highlighted with the thick lines; in the asymmetric routing case, the downstream forwarding paths are marked with arrows. Use C_{up} to denote the number of MUM transmissions in the upstream path to the source node and C_{down} the downstream transmissions in the downstream forwarding path. According to the discussions in the previous paragraph, the ratio of control overhead in the asymmetric routing case to that in the symmetric

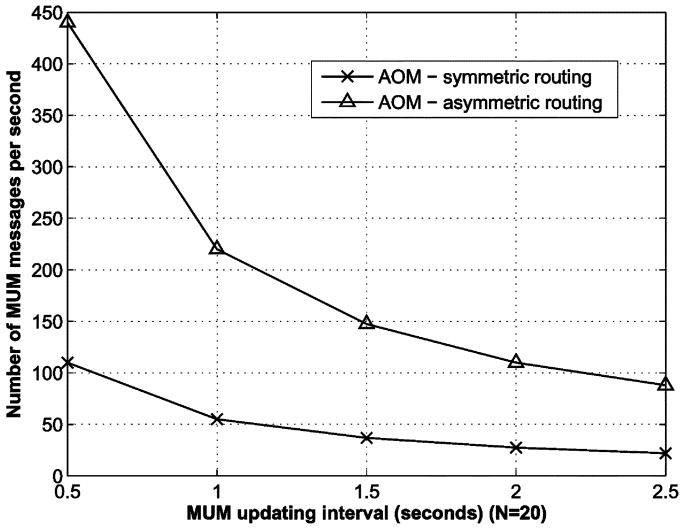

Fig. 9. Control message overhead.

routing case can be expressed as $(2 \cdot C_{up} + C_{down})/C_{up} = 2 + C_{down}/C_{up}$, which is independent of the MUM updating interval. In our simulations generating the results of Fig. 9, with $N = 20$, $C_{down} = 220$, and $C_{up} = 110$ and leads to a control overhead ratio of 4 determining the distance between the two curves.

By adopting the MUM updating technique suggested in [17], the size of the MUM message can be controlled small as 15 bytes, leading to a total packet size of 73 bytes with TCP/IP/MAC headers taken into account. For the asymmetric routing configuration, Fig. 9 shows that 440 control messages are running over the network per second when the updating interval is 0.5. For a large scale Internet backbone with 20 000 access routers, the control overhead can be estimated as $440 \times 73 \times 20\,000/20 = 32.12$ MBps, which occupies only a small fraction of the bandwidth of the backbone network.

B. Forwarding False Positive

We here demonstrate the efficiency of AOM in reducing the false positives induced by the bloom-filter scheme, with comparison to the FRM scheme.

In FRM, the bloom filter in each packet header encodes the whole multicast tree. When a packet arrives at a TBR, all the output interfaces connecting the TBR to its neighboring TBRs will be checked to detect the tree branch. Use n_t to denote the number of tree branches encoded in the packet bloom filter, the false positive in identifying a tree branch is $(1-(1-1/m)^{n_t \cdot k})^k$. The property of FRM is that the false positives at different hops are independent, and thus the probability that an upstream false positive is propagated to downstream nodes and further to the destination is negligibly small. Therefore, the probability F_p that a given RDR receives traffic by false positive under FRM is mainly determined by the last-hop positive as $F_p \approx (1 - (1 - 1/m)^{n_t \cdot k})^k$.

We compare the F_p probability obtained under AOM with that under FRM in Table II. The binary tree topology [29] with different heights is considered. The bloom filter size in the packet header is set as 100 bytes and 20 hash functions are used. For each tree, Table II lists both the numbers of elements to be encoded and the corresponding false positive rates.

TABLE II
FORWARDING FALSE POSITIVE COMPARISON ($m = 800$, $k = 20$)

H	AOM		FRM	
2	4	2.6934×10^{-53}	6	5.4875×10^{-50}
3	8	1.4433×10^{-15}	14	2.5713×10^{-11}
4	16	2.3242×10^{-10}	30	2.8203×10^{-6}
5	32	6.6281×10^{-6}	62	0.0085
6	64	0.0111	126	0.4172
7	128	0.4358	254	0.9658

Compared with FRM in terms of forwarding false positive rate, AOM is more efficient, since AOM packets only need to remember where to go (RDRs) instead of how to get there (the whole multicast tree) as FRM does. The efficiency of AOM over FRM will be particularly significant, when the multicast tree has dispersive branches, and each branch needs to travel a long distance in terms of hops to reach a destination domain. Such kind of tree requires a small number of RDRs to be encoded in the DST_BF and thus reduces the false positive rate in the bit matching operation. In contrast, as FRM is designed to encode the entire tree in each packet, it is more suitable for "short" multicast trees with fewer branches.

VII. CONCLUSIONS

In this paper, we propose and analyze a scalable and efficient multicast protocol from the perspective of an application-network cross-layer design. We particularly exploit the network-embedded application intelligence or application-specific computation provisioned by the emerging application-oriented networking technologies. The proposed application-oriented multicast has the following properties: 1) It leverages the unicast routing infrastructure, eliminating the need for maintaining an extra multicast routing table. 2) The forwarding bandwidth efficiency of AOM is very close to that of the IP multicast; 3) The computation/memory cost incurred at each router is limited and independent of the number of groups. 4) Multicast addresses can be allocated at each source locally and decoupled with routing and forwarding.

REFERENCES

[1] S. Deering and D. Cheriton, "The PIM architecture for wide area multicasting," *ACM Trans. Comput. Syst.*, vol. 8, no. 2, pp. 85–110, May 1990.
[2] S. Deering, D. Estrin, D. Faranacci, V. Jacobson, C. Liu, and L. Wei, "The PIM architecture for wide area multicasting," *IEEE/ACM Trans. Netw.*, vol. 4, pp. 153–162, Apr. 1996.
[3] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicasting routing protocol," in *IETF RFC 1075*, Nov. 1988.
[4] A. Ballardie, "Core based trees (CBT) multicast routing architecture," in *IETF RFC 2201*, Sep. 1997.
[5] S. Bhattacharyya *et al.*, "An overview of source-specific multicast (SSM)," in *IETF RFC 3569*, Jul. 2003.
[6] J. Moy, "Multicasting extensions to OSPF," in *IETF RFC 1584*, Mar. 1994.
[7] A. Adams *et al.*, "Protocol independent multicast-dense mode (PIM-DM): Protocol specification (revised)," in *IETF RFC 3973*, Jun. 1998.
[8] D. Estrin *et al.*, "Protocol independent multicast-sparse mode (PIM-SM): Protocol specification," in *IETF RFC 2362*, Jun. 1998.
[9] S. Kumar, P. Radoslavov, D. Thaler, C. Alaettinoglu, D. Estrin, and M. Handley, "The MASC/BGMP architecture for inter-domain multicast routing," *Proc. ACM SIGCOMM*, vol. 28, no. 4, Oct. 1998.

[10] K. C. Almeroth, "The evolution of multicast: From the mbone to interdomain multicast to Internet2 deployment," *IEEE Netw.*, vol. 14, no. 1, pp. 10–20, Jan.–Feb. 2000.
[11] S. Fahmy and M. Kwon, "Characterizing overlay multicast networks and their costs," *IEEE/ACM Trans. Netw.*, vol. 15, pp. 373–386, Apr. 2007.
[12] S. Deering, "Internet protocol, version 6 (IPv6) specification," in *IETF RFC 2460*, Dec. 1998.
[13] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana, "Internet indirection infrastructure," *IEEE/ACM Trans. Netw.*, vol. 12, pp. 205–218, Apr. 2004.
[14] Y. Chu, S. Rao, S. Seshan, and H. Zhang, "Enabling conferencing applications on the Internet using an overlay multicast architecture," in *Proc. ACM SIGCOMM*, Aug. 2001, pp. 55–67.
[15] I. Stoica, T. S. E. Ng, and H. Zhang, "REUNITE: A recursive unicast approach to multicast," in *Proc. IEEE INFOCOM*, Mar. 2000, vol. 3, pp. 1644–1653.
[16] L. Costa, S. Fdida, and O. Duarte, "Incremental service deployment using the HOP-by-HOP multicast routing protocol," *IEEE/ACM Trans. Netw.*, vol. 14, pp. 543–556, Jun. 2006.
[17] S. Ratnasamy, A. Ermolinskiy, and S. Shenker, "Revisiting IP multicast," in *Proc. ACM SIGCOMM*, Sep. 11–15, 2006.
[18] X. Tian, Y. Cheng, K. Ren, and B. Liu, "Multicast with an application-oriented networking (AON) approach," in *Proc. IEEE ICC*, May 19–23, 2008, pp. 5646–5651.
[19] D. Clark *et al.*, New Arch: Future Generation Internet Architecture, Tech. Rep., 2003. [Online]. Available: http://www.isi.edu/newarch/iDOCS/final.finalreport.pdf.
[20] D. L. Tennenhouse and D. J. Wetherall, "Towards an active network architecture," in *Proc. DARPA Active Networks Conf. Expo.*, 2002, pp. 2–15.
[21] D. L. Tennenhouse, J. M. Smith, W. D. Sincoskie, D. J. Wetherall, and G. J. Minden, "A survey of active network research," *IEEE Commun. Mag.*, vol. 35, no. 1, pp. 80–86, Jan. 1997.
[22] J. Pasley, "How BPEL and SOA are changing Web services development," *IEEE Internet Comput.*, vol. 9, pp. 60–67, May-Jun. 2005.
[23] Cisco application-oriented networking, Cisco Systems. [Online]. Available: http://www.cisco.com/application/pdf/en/us/guest/products/ps6438/c1650/cdccont_0900aecd802c1f9c.pdf.
[24] B. Cain, S. Deering, I. Kouvelas, B. Fenner, and A. Thyagarajan, "Internet group management protocol, version 3," in *IETF RFC 3376*, Oct. 2002.
[25] R. Doverspike, G. Li, K. Oikonomou, K. K. Ramakrishnan, and D. Wang, "IP backbone design for multimedia distribution: Architecture and performance," in *Proc. IEEE INFOCOM*, May 6–12, 2007.
[26] V. Paxson, "End-to-end routing behavior in the Internet," *IEEE/ACM Trans. Netw.*, vol. 5, pp. 601–615, Oct. 1997.
[27] C. Cetinkaya, V. Kanodia, and E. W. Knightly, "Scalable service via egress admission control," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 69–81, Mar. 2001.
[28] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Math.*, vol. 1, no. 4, pp. 485–509, May 2004.
[29] R. Chalmers and K. Almeroth, "On the topology of multicast trees," *IEEE/ACM Trans. Netw.*, vol. 11, pp. 153–165, Feb. 2003.
[30] The Network Simulator—ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns.
[31] S. Egger and T. Braun, "Multicast for small conferences: A scalable multicast mechanism based on IPv6," *IEEE Commun. Mag.*, vol. 42, no. 1, pp. 121–126, Jan. 2004.

Xiaohua Tian (S'08) received the B.E. and M.E. degrees in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2003 and 2006, respectively. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer engineering, Illinois Institute of Technology, Chicago.

His current research interests include application-oriented networks, multicast protocols, cross-layer design for multimedia networking, and peer-to-peer networks.

Yu Cheng (S'01–M'04–SM'09) received the B.E. and M.E. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2003.

From September 2004 to July 2006, he was a Postdoctoral Research Fellow in the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. Since August 2006, he has been with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, as an Assistant Professor. His research interests include service and application oriented networking, autonomic network management, Internet measurement and performance analysis, wireless networks, and wireless/wireline interworking.

Dr. Cheng received a Postdoctoral Fellowship Award from the Natural Sciences and Engineering Research Council of Canada (NSERC) in 2004, and a Best Paper Award from the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine'07), Vancouver, BC, Canada, in August 2007. He served as a Technical Program Co-Chair for the Wireless Networking Symposium of IEEE ICC 2009. He is an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.

Bin Liu (M'04) received the M.S. and Ph.D. degrees in computer science and engineering from Northwestern Polytechnical University, Xi'an, China, in 1988 and 1993, respectively.

From 1993 to 1995, he was a Postdoctoral Research Fellow with the Beijing University of Posts and Telecommunications, Beijing, China. In 1995, he joined the Department of Computer Science and Technology, Tsinghua University, Beijing, as an Associate Professor, where he became a full Professor in 1999. He had led his teams to prototype numerous equipments such as large capacity of ISDN/ATM switches and high speed routers and transferred these prototypes to industries. His current research areas include network processors, traffic measurement and management, service aware routers, and high speed network security. He coauthored a book *High Performance Switches and Router* (New York: Wiley, 2007).

Dr. Liu has received numerous awards from China including the Distinguished Young Scholar of China. He got the Best Paper Award of the 16th ICCC among over 800 accepted papers. He had served as the Co-Chair of Advances in Networks and Internet Symposium, ICC 2008, the Guest-Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS Special Issues on High Speed Network Security, the Panel Chair of HPSR 2005, and TPC member of many conferences as INFOCOM. He is now an Associate Editor for the journal *Security and Communication Networks*, Wiley, and also a member of the Communications and Information Security Technical Committee (CISTC).