Characterizing Overlay Multicast Networks and their Costs

Sonia Fahmy, *Senior Member, IEEE,* and Minseok Kwon, *Member, IEEE*

***Abstract*— Overlay networks among cooperating hosts have recently emerged as a viable solution to several challenging problems, including multicasting, routing, content distribution, and peer-to-peer services. Application-level overlays, however, incur a performance penalty over router-level solutions. This paper quantifies and explains this performance penalty for overlay multicast trees via (i) Internet experimental data, (ii) simulations, and (iii) theoretical models. We compare a number of overlay multicast protocols with respect to overlay tree structure, and underlying network characteristics. Experimental data and simulations illustrate that the *mean number of hops* and *mean per-hop delay* between parent and child hosts in overlay trees generally decrease as the level of the host in the overlay tree increases. Overlay multicast routing strategies, overlay host distribution, and Internet topology characteristics, are identified as three primary causes of the observed phenomenon. We show that this phenomenon yields overlay tree cost savings: Our results reveal that the normalized cost $\frac{L(n)}{U(n)}$ is $\propto n^{0.9}$ for small n, where $L(n)$ is the total number of hops in all overlay links, $U(n)$ is the average number of hops on the source to receiver unicast paths, and n is the number of members in the overlay multicast session. This can be compared to an IP multicast cost proportional to $n^{0.6}$ to $n^{0.8}$.**

I. Introduction

Overlay networks have recently gained attention as mechanisms to overcome deployment barriers to router-level solutions of several networking problems. Overlay solutions for multicasting [1], [2], [3], [4], [5], inter-domain routing pathologies [6], [7], content distribution [8], and content sharing [9], [10], [11] are being extensively studied. In this paper, we consider a number of *overlay* (application-layer) multicast approaches which have been proposed over the last few years. In overlay multicast, hosts participating in a multicast session form an overlay network, and only utilize unicasts among pairs of hosts (considered neighbors in the overlay tree) for data dissemination. The hosts in overlay multicast exclusively handle group management, routing, and tree construction, without any support from Internet routers.

The key advantages overlays offer are flexibility, adaptivity, and ease of deployment. Overlays, however, impose a performance penalty over router-level alternatives. While overlay multicast clearly consumes additional network bandwidth and increases latency over IP multicast, little attention has been paid to precisely quantifying this overlay performance penalty, either theoretically or experimentally. Moreover, to the best of our knowledge, there is no work on characterizing overlay multicast tree structure. Such characterization is important to gain insight into overlay properties and their causes at *both* the application layer and the underlying network layer. It is also important to compare different overlay multicast strategies to determine how to meet the goals of target applications (e.g., by balancing latency versus bandwidth tradeoffs).

In this paper, we analyze overlay multicast trees via (i) real data from Internet experiments and traceroute servers, (ii) simulations of three representative classes of overlay multicast strategies, and (iii) analytical models. We quantify the performance penalty associated with overlay multicast, with emphasis on the overlay cost (i.e., efficiency) at the network-layer. We derive and validate asymptotic forms of the overlay cost from two different tree models, constructed based upon our observations from the experiments and simulations.

Our results indicate that (i) the mean *number of hops* and *per-hop delay* between parent and child hosts generally decrease, and (ii) the degree of hosts generally decreases, as the level of the host in the overlay tree increases. We find that overlay multicast routing strategies, overlay host distribution, together with small-world and power-law Internet topology characteristics, all contribute to the observed phenomena. We extend our earlier work in [12] by isolating the impact of each of these causes, and quantifying its effect on the overlay cost. Our results reveal that the normalized overlay cost $\frac{L(n)}{U(n)} \propto n^{0.9}$ for small n, where $L(n)$ is the total number of hops in all overlay links (connections), $U(n)$ is the average number of hops on the source to receiver unicast paths, and n is the number of members in the overlay multicast session. This can be compared to an IP multicast cost proportional to $n^{0.6}$ to $n^{0.8}$ [13], [14].

The remainder of this paper is organized as follows. Section II defines overlay networks and performance metrics. Section III characterizes overlay multicast networks via Internet experimental data and simulations. Section IV proposes and validates an overlay multicast model that is based on our observations from experimental and simulation data. Section V summarizes related work. Finally, Section VI gives brief concluding remarks.

– Sonia Fahmy is with the Department of Computer Science, Purdue University, West Lafayette, IN 47907–2066, USA, Tel: +1 765 494 6183, Fax: +1 765 494 0739, E-mail: fahmy@cs.purdue.edu. Minseok Kwon is with the Department of Computer Science, Rochester Institute of Technology, Rochester, NY 14623–5608, USA. E-mail: jmk@cs.rit.edu. This research has been sponsored in part by NSF grant 0238294 (CAREER).

II. Overlay Networks: Definitions and Metrics

We consider the *underlying network* as a graph $G = (N, E)$, where N is a set of nodes, and E is a set of edges. A node $\eta_i \in N$ denotes a *router*, and an edge $(\eta_i, \eta_j) \in E$ denotes a bi-directional physical link in the underlying network. An *overlay network* superimposed on G is a *tree* $o = (s, D, N_o, E_o)$, where s is the source host, D is the set of receiver hosts, $N_o \subseteq N$ is the set of nodes in the underlying network G that are traversed by overlay links, and E_o is the set of overlay links, defined below.

The set of hosts H_o consists of s and D in o, i.e., $H_o = \{s\} \cup D$. The cardinality of set H_o is equal to n. An overlay link $e_o = (d_s, \eta_0, \ldots, \eta_{ls}, d_r) \in E_o$ comprises a host $d_s \in H_o$, followed by a sequence of routers $\eta_i \in N_o$, followed by a host $d_r \in D$. Each receiver $\in D$ appears exactly once at the *end* of any sequence denoting an overlay link, but may appear multiple times at the *beginning* of sequences for different overlay links. An overlay link is typically a UDP or TCP connection established by the overlay multicast protocol.

The number of hops in the router sequence $\eta_0, \ldots, \eta_{ls}$ in an overlay link $e_o \in E_o$ is denoted by ls. For every two routers $\eta_i, \eta_j \in N_o$ that appear consecutively in an overlay link $e_o \in E_o$, there must exist a link connecting them in the underlying network, i.e., edge $(\eta_i, \eta_j) \in E$ holds. The same router $\eta_i \in N_o$ can appear in multiple overlay links $e_o \in E_o$. Subsequences of routers $\eta_i, \ldots, \eta_j$ can also appear in multiple overlay links $e_o \in E_o$. Figure 1 illustrates an example overlay network with 6 overlay links.

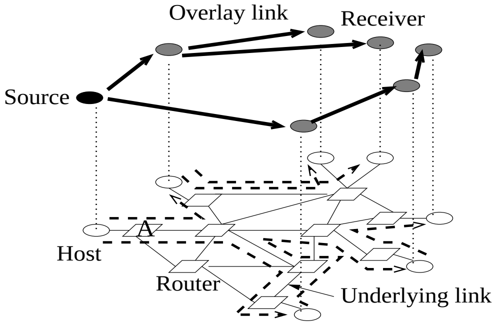


Fig. 1. An example overlay multicast tree over an underlying network

Given an overlay network o, we define the term *overlay cost* as the number of underlying hops traversed by every overlay link $e_o \in E_o$ for an overlay o. More formally, the overlay cost is: $\Sigma_{\forall e_o \in E_o}\ ls(e_o)$, where $ls(e_o)$ denotes the number of router-to-router hops between $\eta_0, \ldots, \eta_{ls}$ for the overlay link e_o (as defined above). We consider the first and last hops to/from hosts separately. This is because we must fairly compare the normalized overlay cost to the normalized IP multicast cost computed in [14], [15], [16], where the first and last hops are ignored. For example, the overlay cost for the overlay in Figure 1 is 2+3+1+1+4+2=13.

We also use the term *link stress* to denote the total number of identical copies of a packet over the same underlying link (as defined in [1], [17]). For example, the stress of the link from the source to A in Figure 1 is two. It is clear that the overlay cost defined above can be represented as $\forall i$, $\sum_i$*stress(i)* where i is any *router-to-router* link traversed by one or more overlay links $e_o \in E_o$, and *stress(i)* is the stress of link i. Prior work also used a *resource usage* metric, defined as $\forall i$, $\sum_i$*delay(i)*×*stress(i)*, where i is an underlying link traversed by one or more overlay links [1]. Our overlay cost metric is a special case of this resource usage notion, when *delay(i)*=1, $\forall i$. We have opted to evaluate delays separately from the overlay cost in this paper.

In addition to the overlay cost and link stress, we study the following overlay tree metrics: (1) degree of hosts H_o (equivalent to the host contribution to the link stress of the host-to-first-router link), (2) degree of routers $\in N_o$, and hop-by-hop delays of underlying links traversed by overlay links $\in E_o$, (3) overlay tree height, (4) per-hop delays, number of hops and total delays between parent and child hosts, (5) mean bottleneck bandwidth between the source s and receivers $\in D$, and (6) mean latency, longest latency, and relative delay penalty (RDP) from the source to a receiver. RDP was first used in [17], and is defined below.

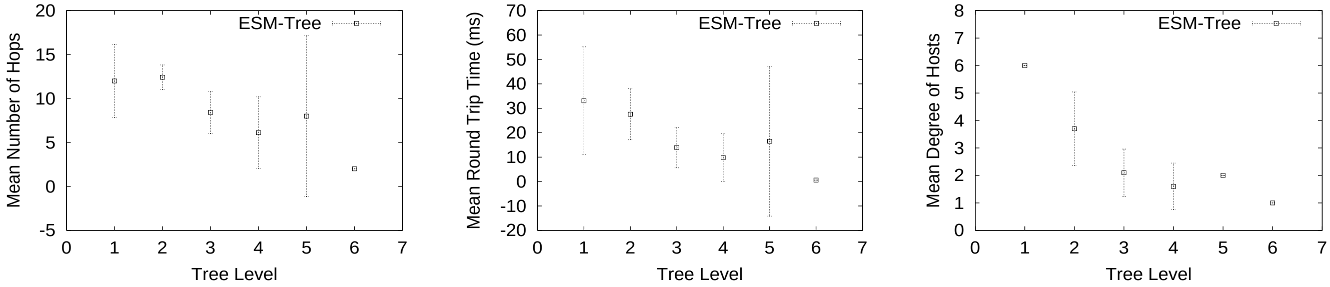
The latency *latency*(s, d_r) from the source s to $d_r \in D$ is: $delay(s, d_0) + \sum_{i=0}^{l-1} delay(d_i, d_{i+1}) + delay(d_l, d_r)$, assuming s delivers data to d_r via the sequence of hosts $(d_0, \cdots, d_l)$. Here, *delay*(d_i, d_{i+1}) denotes the end-to-end delay of the overlay link from d_i to d_{i+1}, for $d_i \in H_o$ and $d_{i+1} \in D$. Note that the RDP from s to d_r is the ratio $\frac{latency(s,d_r)}{delay(s,d_r)}$. We compute the mean RDP of all receivers $\in D$. We can also define the *stretch* as $\frac{hops(s,d_r)}{ls(s,d_r)+2}$ where $hops(s, d_r) = ls(s, d_0) + \sum_{i=0}^{l-1}(ls(d_i, d_{i+1}) + 2) + ls(d_l, d_r) + 4$. Stretch denotes the relative number of hops instead of the relative latency used in RDP. These metrics compare overlay multicast to unicast (or IP multicast using a minimum delay tree). It is clear that there is a tradeoff between the latency metrics and the stress/bandwidth metrics. Balancing this tradeoff is the key to effective overlay multicast protocol design.

III. Overlay Multicast Tree Structure

Our primary goal in this section is to understand the impact of (i) the overlay protocol, (ii) the underlying network connectivity and routing, and (iii) the overlay host distribution, on the overlay tree structure and the overlay performance. We first analyze Internet experimental data, and then conduct a set of simulations.

A. Experimental Data

In order to study the structure of *real* overlay networks in the Internet, we analyze experimental results for the End System Multicast (ESM) protocol [1], [17], the TAG protocol [4], and the NICE protocol [3]. To analyze ESM, we recorded the overlay trees constructed during experiments performed by



(a) Number of router-to-router hops between parent-child hosts versus level of host in overlay tree

(b) Mean round trip time between parent-child hosts versus level of host in overlay tree

(c) Degree of host versus level of host in overlay tree

Fig. 2. Overlay trees constructed by End System Multicast in November 2002

the ESM developers in November 2002. (Unfortunately, the ESM developers have not released the overlay tree structure in their later experiments.) We recorded the structure of 30 overlay trees. Since the overlay trees did not change significantly throughout the experiment lifetime, we selected one representative overlay tree. The tree comprises 65 hosts: 6 nodes at level 1, 22 nodes at level 2, 23 nodes at level 3, 8 nodes at level 4, 5 nodes at level 5, and 1 node at level 6.

We use *traceroute* to find the underlying path between every two hosts on the overlay tree.[1] Since we conducted our ESM analysis before PlanetLab became operational, finding the paths between two arbitrary hosts (without having accounts on either of these hosts) was non-trivial. We utilized publicly available traceroute servers [18] and our own machines to compute paths to all the hosts on the overlay tree.[2] These paths are then synthesized to approximate the paths between any two overlay hosts. For example, consider two hosts h_0 and h_1. We find the paths to both h_0 and h_1 from traceroute servers, or our local machines. If these two paths share a node, this node becomes a junction point. For example, if the path from server s_0 to h_0 is (s_0, r_0, r_1, h_0) and the path from server s_1 to h_1 is $(s_1, r_2, r_0, r_3, h_1)$, we use the approximate path $(h_0, r_1, r_0, r_3, h_1)$ between h_0 and h_1. The path synthesis task was simplified because hosts used in the experiments, with a few exceptions, were located at universities in the United States. Most university hosts are connected to the Internet2 backbone network [20], and thus the routes typically intersect at points on Internet2. These points provide the synthesis junctions used.

Number of Hops. Figure 2(a) depicts the mean number of hops between every two parent-child ESM hosts, for hosts at different levels of the overlay tree (90% confidence intervals are shown here to indicate variability). The figure shows that the number of hops typically decreases as the host level increases, though the decrease is not monotone, and there is variance among nodes at the same tree level. We now seek the causes of this phenomenon. Consider a set of routers that are connected according to the power-law [21] and small-world [22], [23] properties. The power-law property dictates that there is a larger number of low-degree routers than high-degree routers. We surmise that a high-degree high-bandwidth router is typically more likely to be traversed by overlay links near the source of the overlay tree. This is because a high-degree router has higher chances of reducing the path length and delays than a low-degree router, due to its connectivity to a larger number of routers. The high-degree router is also more likely to have high bandwidth links connected to it. Overlay multicast protocols which consider delay, path length, or bandwidth are thus likely to exploit such high-degree routers in the first few levels of the tree (unless all hosts are clustered near the source). Recall also that nearby hosts tend to be clustered by the small-world property. Accordingly, we can visualize an overlay tree where a number of high-degree routers connect the hosts at the first few levels of the tree. In addition, many hosts are connected to low-degree lower-bandwidth routers, which are clustered at lower levels of the tree. Therefore, hosts at lower levels of the overlay tree may only be a few hops away from each other. We study router degree at different levels of the tree via simulations in Section III-B.2. Overall, a significant number of hosts are within 2 or 3 hops of their parents, and many are 9–15 hops away.

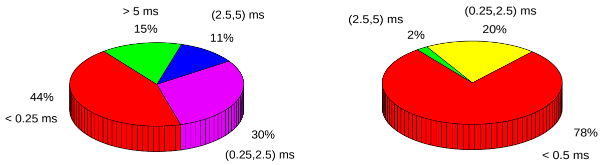
Delay Characteristics. We now study the delays between parents and children at different levels of the overlay tree. The distribution of round trip times between every two parent-child ESM hosts at different levels of the overlay tree is plotted in Figure 2(b) (with 90% confidence intervals). We use round trip time estimates obtained from traceroute. From the figure, the average round trip time generally decreases as the host level increases, confirming our intuition. The large

[1] We encountered two problems using traceroute. First, some routers do not generate ICMP Time-Exceeded packets when TTL (Time-To-Live) reaches zero. Second, many routers disable the source-route capability, primarily due to security concerns.

[2] traceroutes were not performed at precisely the same times the data was recorded, and this can slightly impact our results. However, routes do not typically change often [19].

error ranges in the figure indicate that the round trip times significantly vary at the same level of the tree. Figure 2(c) illustrates that the degree of hosts in the overlay tree grows as hosts get closer to the root of the overlay tree. This decreasing degree can be attributed ESM's goal of minimizing delay (if bandwidth is acceptable).

Figure 3 shows the distribution of per-hop delay (the delay between two consecutive routers on a path from a parent to a child ESM host) for different overlay tree levels. The per-hop delay between two consecutive routers η_i and η_j is estimated as $\frac{1}{2}rtt(\eta_i, \eta_j)$, where $rtt(\eta_i, \eta_j)$ is the time to travel from η_i to η_j and vice versa obtained via traceroute. The figure indicates that 78% of per-hop delays in lower tree levels (levels 4-6) are shorter than 0.25 ms, and only 2% are between 2.5 and 5 ms. In contrast, only 44% of per-hop delays are shorter than 0.25 ms, 11% are between 2.5 and 5 ms, and 15% exceed 5 ms, for the first level of the tree, which agrees with our discussion above.



(a) Tree level 1

(b) Tree level 4-6

Fig. 3. Distributions of per-hop delay for different overlay tree levels

Impact of Overlay Protocol. We have also conducted experiments with NICE [3] and TAG [4] on the PlanetLab testbed [24] in 2003 and 2004. We use *tracepath* [25] to find the number of hops and delay on underlying paths. We selected representative overlay trees for NICE and TAG from several experiments with 60 group members. A cluster in NICE has 2 to 5 members (see [3] for details). For TAG, we use *bwthresh*=160 kbps, *chlimit*=5, and $u = 1$ (the details of the TAG algorithm and its parameters are discussed in Section III-B.1). Our results (given in [26]) show that trees constructed by TAG exhibit similar properties to those observed with ESM, as discussed above. NICE, however, does *not* exhibit a similar decrease in number of hop as tree level increases exhibited by ESM and by TAG. This is because scalability is the primary concern of NICE, and not bandwidth or delay as in ESM and TAG. We discuss this further in Section III-B.2.

B. Simulation Experiments

We also investigate the overlay structure via session-level simulations.

1) Simulation Setup: We use two router-level topologies. The first topology contains 4000 routers connected according to power-law and small-world properties. In a power-law distribution, a complementary cumulative distribution function $cd^{-\alpha}$ is used to denote the fraction of routers with degree greater than d, where c and α are constants [27], [28]. We use $c = 1$ and $\alpha = 1.22$. These parameters mimic real Internet topologies reasonably well (refer to [28] for the rationale). Groups of routers are clustered according to the small-world property: a router connects to its closest neighbor routers with probability p, and to other routers with probability $1 - p$, according to router degree. We use $p = 0.5$. Routers are uniformly distributed on a 750×750 plane, and the Euclidean distance between two routers approximates the delay between the two routers (in ms). Hosts are connected to edge routers (which are defined as routers with degree less than 10) uniformly at random. The bandwidth from edge routers to hosts is selected according to the distribution: 40% are 56 kbps, and 15% for each of 1.5, 5, 10, 100 Mbps. All other links are assigned bandwidths ranging from 100 Mbps to 1 Gbps.[3]

The second topology we use is a Transit-Stub topology generated by the popular GT-ITM topology generator [29]. The topology contains 4040 routers which constitute 4 transit domains, 10 routers per transit domain, 4 stub domains per transit router, and 25 routers per stub domain. GT-ITM generates symmetric link delays ranging from 1 to 55 ms for transit-transit or transit-stub links. We use 1 ms to 10 ms delays within a stub. Hosts are connected to stub routers randomly and uniformly. Backbone links have bandwidths ranging from 100 Mbps to 1 Gbps, while links from edge routers to hosts have the same bandwidth range as in the first topology. In both topologies, the underlying network routes are selected to optimize *delays*. It is also worth mentioning that we have simulated smaller scale topologies and the results were similar.

We simulate three representative overlay multicast protocols on the two topologies: ESM [1], Topology-Aware Grouping (TAG) [4], and Minimum Diameter Degree-Bounded Spanning Tree (MDDBST) [5]. The reason we select ESM is that it is the first overlay multicast protocol to be widely tested in the Internet. It was used for multicasting the SIGCOMM 2002/2003 conferences. Moreover, ESM has a unique routing mechanism. The overlay tree construction protocol of ESM is given in [26]. Each host evaluates the utility of other hosts to determine its neighbors. A host has an upper degree bound (UDB) on the number of its neighbors. We use a value of 6 for the upper degree bound. The ESM flavor used in our simulations has two discretized bandwidth levels: $>$ 100 kbps and $\leq$ 100 kbps (similar to the version used for the SIGCOMM 2002 multicast). The overlay tree is first optimized for bandwidth, and then uses delay as a tie breaker among hosts at the same bandwidth level.

The second class of protocols we investigate is topology-aware overlay multicast protocols, which includes Scribe [30], topology-aware Content-Addressable Network

[3] These numbers were synthesized from: http://www.websiteoptimization.com/bw/0509/ and FCC annual reports.

(CAN) [31], and TAG [4]. We select TAG as a representative of this group. TAG is a faithful representation of topology-based approaches, since it aligns overlay routes and underlying routes, if certain weak constraints are met. Although the TAG heuristic may not perform particularly well if inter-domain routes are of poor quality, its simplicity makes it appealing. The pseudo-code for TAG tree construction is given in [26]. A TAG host becomes the child of the host that most "matches" its path. Here, a path is defined as the sequence of routers from the source to a host. A's path matches B's path when the path from the source to A and the path from the source to B have a common prefix of length equal to the path from the source to A minus u unmatched routers. Two weak constraints are employed by TAG on the bandwidth and the number of children of a host (the bandwidth from a parent to a new member is larger than *bwthresh* and the number of children of the parent is less than *chlimit*). We use $u = 0$, *bwthresh*= 150 kbps and *chlimit*=50 in our simulations.

The third class of protocols we investigate includes protocols that seek to minimize overlay cost [32], or the longest path in an overlay network [5] (with delay or bandwidth constraints). We select MDDBST, given in [5], as a representative protocol in this class. MDDBST minimizes the cost (delay in our simulations) in the longest path, and bounds the degree of hosts. The pseudo-code for MDDBST is presented in [26]. The MDDBST protocol we use is slightly modified for use in a single-source overlay multicast scenario. We define the degree bound as *degree(v)=lastbw(v)/unitbw*, where *degree(v)* is the degree of node v, *lastbw(v)* is the last hop bandwidth of v, and *unitbw* is the desired bandwidth for a single connection. We use *unitbw*=56 kbps in our simulations. For each protocol, we run five simulations with different random number generator seeds (for topology generation and for selecting the multicast source and destinations) and average the results.

Table I compares a number of overlay multicast algorithms with respect to tree construction (mesh first, or tree, or hierarchical), tree types (source-based trees or a single shared tree), tree height, target group size, metrics used in tree construction, and control overhead.

2) Simulation Results: **Impact of Overlay Protocol and Underlying Topology on Tree Structure.** Figure 4 illustrates the mean number of hops between parent and child hosts for different host levels in the overlay tree.[4] The label "ESM-4k" denotes ESM with 4000 members; similar labels are used for the other cases. Figure 4(a) depicts the results on the power-law and small-world topology. The figure reveals that the number of hops between parent and child hosts tends to decrease as the level in the overlay tree increases, for both ESM and TAG. MDDBST does not exhibit a clear trend. The observed decrease in mean number of hops is consistent with our experimental data, and our intuition on the effect of Internet topology characteristics. We have observed similar trends with 40 and 400 members.

In order to isolate the effects of the power-law property from the small-world property, we execute the same simulations on only-power-law (but no clustering) and only-small-world (but equal degree routers) topologies. Figures 4(b) and 4(c) give the results. From both figures, we observe that the number of hops in ESM and TAG decreases with overlay tree level increase, but the decrease is not as pronounced as when both properties are combined (Figure 4(a)). Therefore, *both* clustering among closely located routers as dictated by the small-world property, and power-laws of router degrees, appear to contribute to the observed decrease in number of hops with overlay tree level increase. To confirm this, we study the results on the GT-ITM Transit-Stub topology. We find that ESM shows less noticeable and less rapid decrease in the number of hops as the level increases, compared to Figure 4(a). This is expected since GT-ITM router degrees do *not* follow a power-law. For MDDBST, the number of hops between parent and child hosts initially fluctuates and slowly decreases as the level increases [26]. This is because MDDBST does not seek the shortest path to individual hosts, but minimizes the longest path in the tree.

Two aspects of ESM contribute to the observed tree structure, which decreases tree cost: (i) the mesh optimization, and (ii) the DVMRP-based overlay tree construction. The mesh optimization chooses potentially useful nodes over unpopular nodes as intermediate hops. Most of the nodes then connect to close neighbors (many of which are at the bottom of the tree), while only a few nodes in strategic locations become intermediate hops. The routing algorithm of ESM uses shortest-path based routing (DVMRP) and hence results in a delay-balanced tree with nearby nodes clustered at lower levels. TAG also exhibits this phenomenon since its path matching algorithm aligns overlay routes with underlying routes (subject to bandwidth availability) and underlying routes are typically optimized for shortest paths (subject to routing policies). In general, the decreases are more pronounced for TAG and ESM than for MDDBST and NICE, independent of underlying topologies.

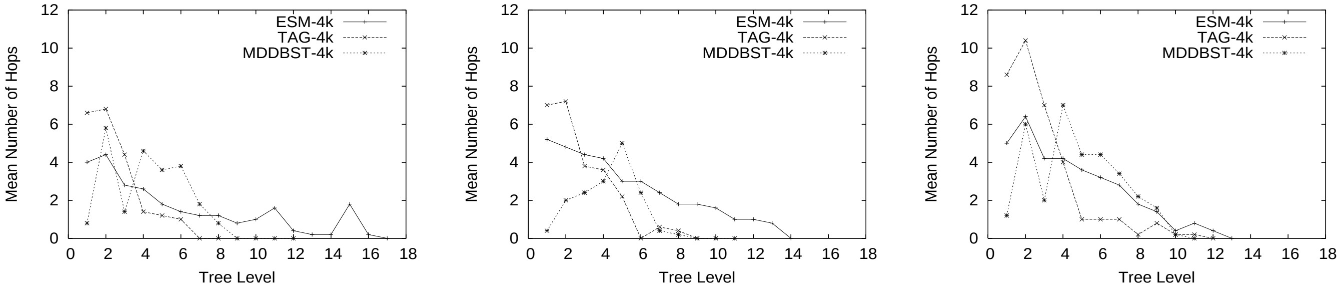
We now investigate the effects of underlying topology in more depth by varying the power-law and small-world parameters – specifically α and the probability p. In Figure 5(a), we find that the number of hops in all three protocols decreases slowly (but non-monotonically) with overlay tree level increase, when router degrees have a wide range. Relay through high-degree routers may reduce the number of hops between hosts in this case. As the range of router degrees becomes narrow (in Figure 5(c)), the number of hops tends to fluctuate more. Similarly, we have found that a stronger small-world effect yields a more smooth and more rapid decrease of the number of hops. The results were consistent

[4] We do not show confidence intervals on this and the next figures to improve readability. The standard deviation values were smallest for ESM (less than 2 for almost all tree levels) in figures 4(a), (b), and (c), followed by TAG, and then MDDBST.

TABLE I

A COMPARISON OF OVERLAY MULTICAST ALGORITHMS

Algorithm	Mesh/Tree	Tree type	Tree height	Group size	Metrics	Control overhead
ESM	Mesh	Source	Unbounded	Small	Bandwidth, delay	O(n)
NICE	Hierarchical	Source	$O(\log n)$	Large	Delay	$O(\log n)$
Overcast	Tree	Source	Unbounded	Large	Bandwidth	O(max-degree)
CAN-multicast	Hierarchical	Source	$O(dn^{1/d})$	Large	Delay	Constant
ScatterCast	Mesh	Source	Unbounded	Large	Delay	O(max-degree)
Yoid	Tree	Shared	Unbounded	Large	Delay	O(max-degree)
ALMI	Tree	Shared	Unbounded	Small	Delay	O(max-degree)
MDDBST	Tree	Shared	Unbounded	Large	Edge cost	O(max-degree)
Scribe	Hierarchical	Source	$O(\log n)$	Large	Delay	$O(\log n)$
HMTP	Tree	Shared	Unbounded	Large	Delay	O(max-degree)
Hypercast	Mesh	Source	Unbounded	Large	Coordinate, angle	O(max-degree)
TAG	Tree	Source	Unbounded	Large	Delay, bandwidth	O(max-degree)
Bayeux	Hierarchical	Source	$O(\log n)$	Large	Delay	$O(\log n)$



(a) Power-law and small-world topology

(b) Only-power-law topology

(c) Only-small-world topology

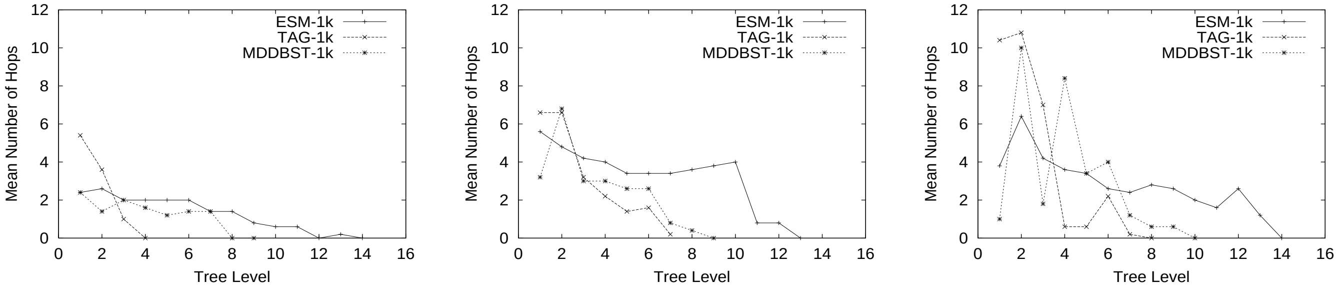
Fig. 4. Mean number of parent-child hops versus overlay tree level in power-law and small-world simulations

for ESM and TAG with different number of overlay tree members and different protocol parameters.

To validate our conjecture that high-degree routers tend to be traversed in upper levels of the overlay tree further, we examine the distribution of the router degree against the overlay tree level for the power-law and small-world topology. The router degree denotes the connectivity of the router to other *routers*. For tree level i, the routers on overlay links from hosts at level $i-1$ to i are considered. (Note that the same router may appear at different levels of the overlay tree, if traversed by overlay links at different levels). We find that the results (which we give in [26]) agree with our argument. We also find that all three protocol trees cross a significant number of high-degree routers (50+), in order to exploit their high connectivity and high bandwidth.

Impact of Member Host Distribution on Tree Structure. We also simulate the three protocols on the power-law and small-world topology with a *non-uniform* host distribution. This is the typical case with academic conference streaming, when users are clustered at a few universities. It is also common with some sporting event streaming, when there is a high concentration of viewers in the home cities of participating teams. In this case, we randomly select an edge router and then connect ω hosts to this router and its neighboring routers (one host per router), where ω is a random number between 1 and 20. Figure 6 illustrates that the number of hops between parent and child hosts decreases even more rapidly (though with some fluctuations) than uniform host distribution case (Figure 4(a)). The decrease was less pronounced when we repeated the same experiment on the Transit-Stub topology, and when we experimented with smaller ω values. Therefore, underlying topology properties as well as non-uniform host distribution are all factors that exacerbate this phenomenon. The routing features of overlay multicast protocols, such as the utility for selecting neighbors in ESM, or topology awareness in TAG, also play an important role.

Impact of Overlay Protocol, Underlying Topology, and Member Host Distribution on Tree Cost. We now compare the normalized overlay costs of different topologies and host distributions for the three protocols. Figure 7(a) and (b) show that a strong power-law (a) or small-world (b) topology achieves significantly lower costs than GT-ITM. Non-uniform host distribution also significantly reduces overlay multicast cost, as depicted in Figure 7(c). It is also clear that ESM is generally more effective in reducing cost than MDDBST, since its trees exhibit decreasing hops and delays in lower tree levels. The cost is lowest for the case of ESM in Figure 7(c), i.e., non-uniform host distribution, and small-



(a) ($\alpha = 0.5$, $p = 0.5$)

(b) ($\alpha = 1.22$, $p = 0.5$)

(c) ($\alpha = 2$, $p = 0.5$)

Fig. 5. Mean number of parent-child hops versus overlay tree level as the effect of power-law decreases

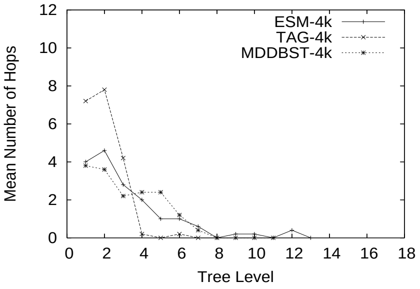


Fig. 6. Mean number of hops versus overlay tree level in simulations on the power-law and small-world topology with non-uniform (clustered) host distribution

world and power-law properties similar to Internet topologies (1.22 and 0.5 for $= \alpha$ and p respectively). These results confirm our intuition that the overlay protocol, the Internet power-law property, the Internet small-world property, and overlay host clustering all contribute to making overlay multicast effective in reducing cost and increasing bandwidth efficiency.

Host Degree and Mean Bandwidth Properties. Results of host degree and mean bandwidth, as well as results for latency, RDP, and stress can be found in [26].

IV. Overlay Multicast Tree Cost

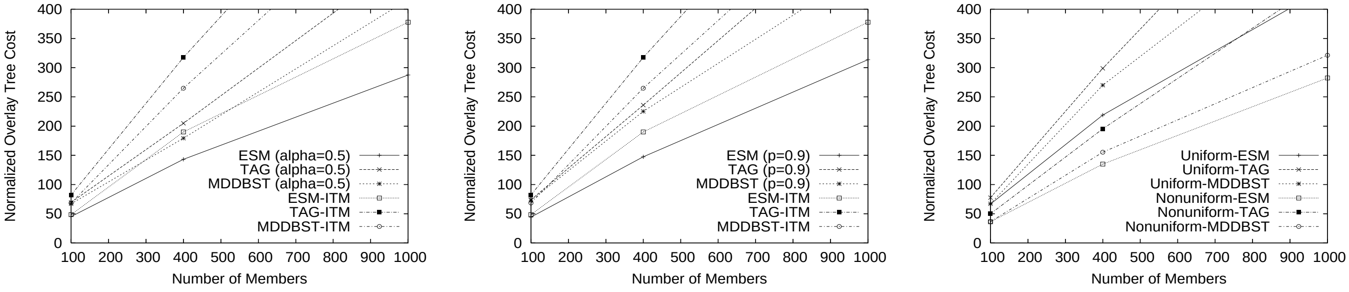
In this section, we model overlay multicast trees *based on the overlay tree structure we have observed via experiments and simulations*, and we compute the overlay costs.

A. Network Model

We model the underlying network as a graph $G = (N, E)$ and the overlay tree o as the tuple (s, D, N_o, E_o), as defined in Section II. To simplify our analysis, we assume G to be a complete k-ary tree $G = (N, E, r)$ on which o is constructed, where $r \in N$ is designated as the root router. s is the only host connected to r. Other hosts are connected to routers with equal probability in G to obtain D. The height of G is h. This assumption is not unrealistic in this context, since the overlay cost exhibited with an underlying *tree* has been shown to be *more consistent* with that exhibited with real topologies, compared to meshes or random graphs [33]. We are, however, currently investigating relaxing this assumption by computing the average costs for the set of trees covering a power-law and small-world underlying network.

We now seek to *incorporate the number-of-hops distribution properties we observed* in our Internet experimental data and simulations results (discussed in Section III). To model hops between overlay hosts, routers must be added between every two branching points in the underlying network model. Such routers are called *unary nodes*. Recall that we had observed that the number of hops between parent and child hosts approximately decreases, as the level of the host in the overlay tree increases. A similar modeling assumption to that in [16] – a *self-similar tree* – can be used to *model this observation* without making the analysis exceedingly complex. This entails that $A_i = \phi A_{i-1}$, $0 \leq \phi \leq 1$, where A_i is the number of concatenated links generated by unary nodes in the underlying network between a node at level $i - 1$ and a node at level i of the overlay tree. It is important to note that, throughout the rest of this paper, the height h refers to the height of a tree without the unary nodes. This simplifies the exposition. A number $k^{(h-i)\theta} - 1$ of unary nodes is created between adjacent nodes at levels $i - 1$ and i of the overlay tree, where $0 \leq \theta < 1$. The tree has no unary nodes when $\theta = 0$. Note that the number of hops on overlay links will not be monotonically decreasing (but will be approximately decreasing) for increasing levels of the overlay tree, since data may be disseminated up G in certain segments, as discussed below.

We assume that each receiver is connected to a router in the network uniformly and independently of other receivers. We use the term $L_o(h, k, n)$ to denote overlay cost for an overlay tree o and number of hosts $|H_o| = n$ (h and k are defined above). In [14], m, the number of distinct routers to which hosts are connected, is used instead of n in $L_o(h, k, n)$. We, however, believe that using the number of hosts n is intuitively appealing and makes analysis simpler. Note that



(a) ($\alpha = 0.5$, $p = 0.5$) versus GT-ITM

(b) ($\alpha = 1.22$, $p = 0.9$) versus GT-ITM

(c) Uniform versus non-uniform host distributions

Fig. 7. Comparisons of normalized overlay cost for different topologies and host distributions

m can be approximated by $M(1-(1-\frac{1}{M})^n)$, where M is the total number of available routers to which hosts can be connected. Therefore, $m \approx n$ when $\frac{n}{M} \ll 1$ [16].

Among all possible overlay networks that can be superimposed on G, we compute the *least cost* overlay network defined as follows.

Definition 1: Let Ω be the set of all possible overlays, connecting a particular set of n hosts, and superimposed on a network G. Let $L_\tau(h,k,n)$ be the overlay cost for $\tau \in \Omega$. Let o be the least cost overlay on G. Then, o is the overlay that satisfies $L_o(h,k,n) \leq L_\tau(h,k,n)$ for all $\tau \in \Omega$.
We consider the least cost overlay network for three primary reasons. First, modeling and analysis are simplified in this case. Second, many overlay multicast protocols optimize a delay-related metric, which is typically also optimized by underlying (especially intra-domain) routing protocols. Third, it gives a lower bound on the overlay tree cost under our assumptions.

B. Receivers at Leaf Nodes

We first consider a network in which receivers can only be connected to leaf nodes in the underlying network. Figure 8(a) shows a model of such a network. One host, which is the current source of the overlay multicast session, is connected to the root r of the tree. All other hosts are connected to leaf nodes, selected independently and uniformly. We define ρ to be the lowest level with branching nodes above or at *half* of the total tree height. Since $\sum_{i=\rho+1}^{h} k^{(h-i)\theta}$ indicates the height from ρ to the lowest tree level, ρ can be computed as:

$$2\sum_{i=\rho+1}^{h} k^{(h-i)\theta} \leq \sum_{i=1}^{h} k^{(h-i)\theta}. \tag{1}$$

Thus,

$$\rho = \left\lceil h - \frac{1}{\theta}\log_k \frac{k^{h\theta}+1}{2} \right\rceil. \tag{2}$$

Figure 8(a) shows that the cost incurred when communicating from a receiver to another receiver, both connected to descendants of node σ at level ρ, is bounded by the total tree height. Otherwise, the source would send another copy directly to the receiver at a cost equal to the tree height. For this reason, we group together all receivers connected to descendants of σ in a subtree rooted at σ. Similar subtrees are created for every node at level ρ.

We divide the computation of $L_o(h,k,n)$ into two terms. The first term is the minimum cost to send to the subtrees rooted at σ, and the second term is the minimum cost of data dissemination within the subtrees. To compute the first term, we observe that there are k^ρ nodes at level ρ in the tree. The probability that a link connecting to level ρ is traversed by overlay o is $1-(1-k^{-\rho})^n$. Thus, the cost to transmit to all nodes at level ρ, without unary nodes, is simply $k^\rho(1-(1-k^{-\rho})^n)$. Since $k^{(h-i)\theta}$ is additionally incurred by a node at level i if the tree is extended with unary nodes, the first term becomes:

$$\sum_{i=1}^{h} k^{(h-i)\theta} k^\rho (1-(1-k^{-\rho})^n) = \frac{k^{h\theta}-1}{k^\theta-1} k^\rho (1-(1-k^{-\rho})^n). \tag{3}$$

To compute the second term of $L_o(h,k,n)$, we consider the subtree rooted at σ. This subtree and potential overlay links are shown in Figures 8(a) and (b). Consider a node α_l at branching point level l, where $\rho \leq l < h$. Let α_{l+1}^0 and α_{l+1}^1 be two children of α_l at the next branching point level $l+1$. Suppose that A is a receiver connected to a descendant of α_{l+1}^0, and B is a receiver connected to a descendant of α_{l+1}^1. Sending data from A to B across (up and then down) α_l costs:

$$2\sum_{i=l+1}^{h} k^{(h-i)\theta} \approx 2k^{(h-l-1)\theta}. \tag{4}$$

The probability that data is transmitted via a branching point at branching point level $l+1$ in o is $1-(1-k^{-(l+1)})^n$. Node α_l has k children in o, so we multiply this factor by k, which yields $k(1-(1-k^{-(l+1)})^n)$. Since overlay links for data transmission are created between children of α_l across α_l, we modify the factor to $k(1-(1-k^{-(l+1)})^n)-1$.

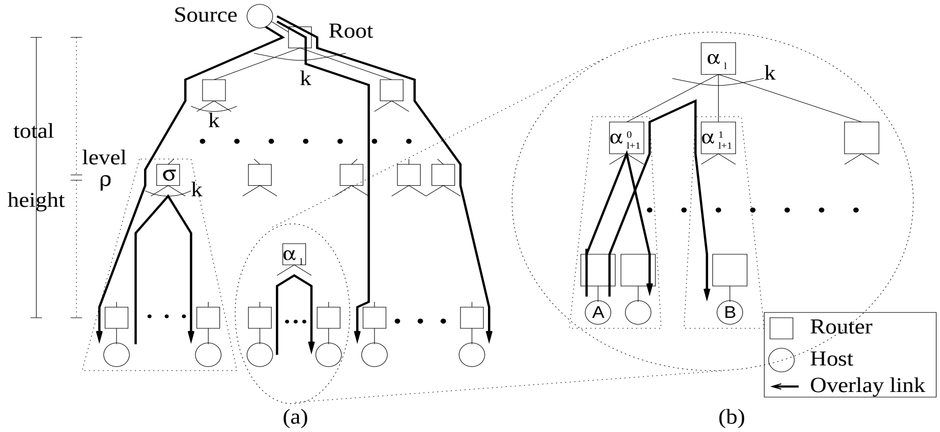


Fig. 8. An overlay tree model with receivers located only at leaf nodes (for simplicity, unary nodes are not shown)

Multiplying Equation (4) by this factor yields the total cost for data transmission from leaves (to other leaves) across all branching points at branching point level l in the subtree: $g(l) = 2k^{(h-l-1)\theta}(k(1-(1-k^{-(l+1)})^n)-1)$.

Consequently, the second term of of $L_o(h,k,n)$ becomes:

$$\sum_{l=\rho}^{h-1} k^l g(l). \tag{5}$$

$L_o(h,k,n)$ is the sum of (3) and (5):

$$L_o(h,k,n) = \frac{k^{h\theta}-1}{k^{\theta}-1}k^{\rho}(1-(1-k^{-\rho})^n)+\sum_{l=\rho}^{h-1} k^l g(l). \tag{6}$$

We prove that this tree is indeed the least cost overlay tree on this underlying network in [26]. Since the average number of hops on the source to receiver unicast paths $U_o^\theta(h,k)$ is $\sum_{i=1}^{h} k^{(h-i)\theta} = \frac{k^{h\theta}-1}{k^{\theta}-1}$, the normalized overlay cost becomes:

$$R_o^\theta(h,k,n) = \frac{L_o(h,k,n)}{U_o^\theta(h,k)}. \tag{7}$$

A power-law is observed in (7), where the exponent of n is $1-\theta$ (see Lemma 2 in the Appendix for details). Figure 9(a) depicts the normalized overlay cost $R_o^\theta(h,k,n)$ against the number of overlay group members n.[5] The figure shows that $R_o^\theta(h,k,n) \propto n^{0.92}$, for $0 < a < 1$. Saturation occurs as $a \to \infty$ ($n \to \infty$).

C. Receivers at Leaf or Non-leaf Nodes

We now relax the restriction that receivers are only connected to leaf nodes in the underlying network, as illustrated in Figure 10. A non-leaf node with receiver(s) connected receives data from an ancestor, and relays this data to its descendants. In contrast, descendants of a non-leaf node which has no receivers connected must receive data from other non-ancestor nodes.

We use the same underlying network model as in Section IV-B. We now assume that receivers are uniformly and independently distributed over the entire tree *with the exception of unary nodes*. This implies that the probability that a node (other than the root) has at least one receiver connected is:

$$p = 1-(1-\frac{1}{M})^n \tag{8}$$

for n receivers, where

$$M = k+\cdots+k^h = \frac{k^{h+1}-k}{k-1}. \tag{9}$$

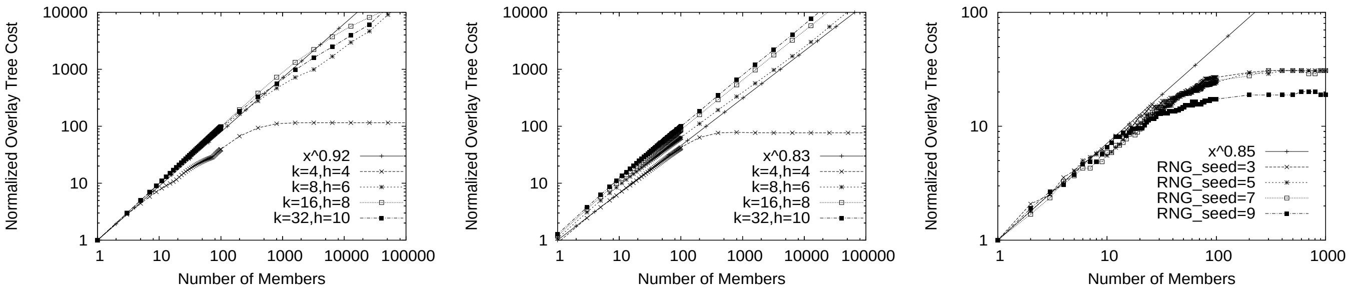
On the average, among the k children of a non-leaf node, kp children have receivers connected, while $k(1-p)$ children have no receivers connected. Let $L_\nu(h,k,n)$ be the overlay cost of an overlay network ν. The computation of $L_\nu(h,k,n)$ is split into two components: (i) cost for kp children of the root with receivers, and (ii) cost for $k(1-p)$ children of the root without receivers. In the first component, one of the kp children incurs $k^{(h-1)\theta}$ from the root and $L_\nu(h-1,k,n)$ for its descendants. Thus, the cost for the kp children of the root is:

$$kp(k^{(h-1)\theta}+L_\nu(h-1,k,n)). \tag{10}$$

Now, consider one of the $k(1-p)$ children of the root without receivers. We again have kp children with connected receivers, and $k(1-p)$ children without connected receivers. A recurrence relation based on this pattern computes the second part of $L_\nu(h,k,n)$ for the $k(1-p)$ children of the root. Consider node σ at branching point level l which does not have receivers connected (refer to Figure 10). There may be receivers at the descendants of σ that use the link from the parent of σ to σ with probability:

$$1-\left(1-k^{-l}\frac{k^h-k^l}{k^{h+1}-k}\right)^n, \tag{11}$$

[5] The total number of routers including unary nodes is 356 for ($k=4, h=4$), 309,819 for ($k=8, h=6$), 4.6 billion for ($k=16, h=8$) and more than 4.6 billion for ($k=32, h=10$).



(a) Receivers at leaf nodes

(b) Receivers at leaf or non-leaf nodes

(c) Traceroute-based overlay multicast simulations using minimum spanning trees

Fig. 9. Normalized overlay cost versus number of members from $R_o(h,k,n)$ for (a) and from $R_\nu(h,k,n)$ for (b) ($\theta = 0.1$) and from simulations for (c) (log-log scale)

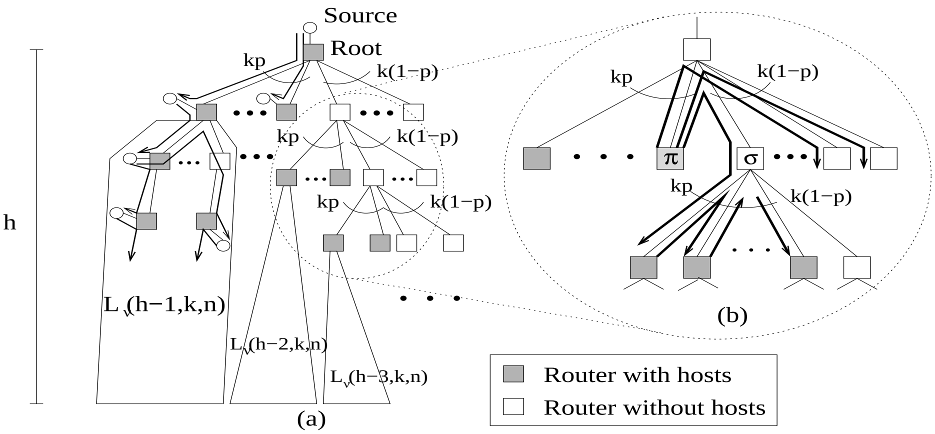


Fig. 10. An overlay tree model with receivers located at leaf or non-leaf nodes (for simplicity, unary nodes are not shown)

where k^{-l} is the probability that a receiver is located below σ, and $\frac{k^h-k^l}{k^{h+1}-k}$ is the probability that the receiver is connected to a non-leaf node at branching point i, $l < i < h$. The latter probability is based on the fact that the total number of branching points except the root is $k+\cdots+k^h = \frac{k^{h+1}-k}{k-1}$ and the number of nodes at branching point i is $\frac{k^h-k^l}{k-1}$. We use $1-(1-k^{-l})^n$ as an approximation of Equation (11) for large values of h.

Let $T(l)$ denote the cost required to deliver data to the descendants of σ at branching point level l. As illustrated in Figure 10, at least one of the kp children must receive data from nodes other than σ and the descendants of σ. A sibling node of σ which has receivers (π in the figure) would minimize the cost to one of these children to $2k^{(h-l)\theta} + k^{(h-l-1)\theta}$. An additional cost of $2k^{(h-l-1)\theta}(kp-1)$ is required to relay the data among the kp children of σ. Thus, $B(h-l-1) = k^{(h-l-1)\theta}(2k^\theta + 2kp - 1)(1-(1-k^{-l})^n)$ is incurred for the kp children of σ. Also, $kpL_\nu(h-l-1,k,n)$ is incurred by the descendants of the kp children of σ. For the $k(1-p)$ children of σ without receivers, $k(1-p)T(l+1)$ is incurred. Hence, $T(l)$ can be computed as:

$$\begin{aligned} T(l) &= B(h-l-1) \\ &\quad + kpL_\nu(h-l-1,k,n) + k(1-p)T(l+1) \\ &= \sum_{i=l}^{h-1} k^{i-l}(1-p)^{i-l} \\ &\quad \times\{B(h-i-1) + kpL_\nu(h-i-1,k,n)\}. \end{aligned} \tag{12}$$

The cost for the $k(1-p)$ children of the root at branching point level $l = 1$ is:

$$\begin{aligned} k(1-p)T(l=1) &= \sum_{i=1}^{h-1} k^i(1-p)^i \\ &\quad \times\{B(h-i-1) + kpL_\nu(h-i-1,k,n)\}. \end{aligned} \tag{13}$$

Therefore,

$$\begin{aligned} &L_\nu(h,k,n) = kp(k^{(h-1)\theta} + L_\nu(h-1,k,n)) \\ &+ \sum_{i=1}^{h-1} k^i(1-p)^i\{B(h-i-1) + kpL_\nu(h-i-1,k,n)\}. \end{aligned} \tag{14}$$

Lemma 1: Solving the recurrence relation in Equation (14) with a fixed ratio $a = \frac{n}{M}$ $(0 < a < \infty)$ (M is as defined in Equation (9)) yields:

$$\begin{aligned} L_\nu(h,k,n) &= k^{(h-1)\theta+1}p + (k^h + k^{h\theta}\sum_{i=2}^{h-1} k^{(1-\theta)i})p^2 \\ &+ k^{(h-2)\theta+1}(1-p)(2k^\theta + 2kp - 1)\sum_{i=0}^{h-2} k^{(1-\theta)i} \\ &- k^{h-\theta}(1-p)(2k^\theta + 2kp - 1)c_2(a,\theta), \\ &+ O(1) \end{aligned} \tag{15}$$

where $c_2(a,\theta) = \sum_{i=0}^{\infty} k^{-(1-\theta)i}e^{-ak^{i+1}}$.

The proof of Lemma 1 can be found in the Appendix. (The proof that $L_\nu(h,k,n)$ is the minimum cost overlay tree when receivers are located at any node except the root can be found in [26].) The average number of hops on the source to receiver unicast paths, $U_\nu^\theta(h,k)$, can be computed as:

$$U_\nu^\theta(h,k) = \frac{1}{M}\sum_{l=1}^{h} k^l \sum_{i=1}^{l} k^{(h-i)\theta}. \tag{16}$$

The normalized overlay cost $R_\nu^\theta(h,k,n) = \frac{L_\nu(h,k,n)}{U_\nu^\theta(h,k)}$ does not exhibit a power-law (see [26]). However, Figure 9(b) demonstrates that $R_\nu^\theta(h,k,n)$ behaves asymptotically similar to a power-law when $0 < a < 1$. The total numbers of routers is the same as in Figure 9(a). In the figure, $R_\nu^\theta(h,k,n) \propto n^{0.83}$. The factor 0.83 is *smaller* than the 0.92 for the case when hosts are only connected at leaves, since many additional hops can be saved in this case. *It is also important to note that our decreasing unary node distribution leads to a lower tree cost (0.83 versus an 0.87 factor for this same model with uniformly distributed unary nodes).* The cost provides a useful notion for comparing and designing overlay multicast protocols to optimize loads. The 0.8 to 0.9 factor can be also compared to a factor ≈ 0.7 for IP multicast [13], [14].

D. Simulation and Experimental Validation

We validate our analytical results using a traceroute-based simulation topology. (Our methodology for synthesizing the routes is discussed in Section III-A.) We simulate hosts connected to edge routers by randomly connecting 1000 hosts to the edge routers connected to 60 selected traceroute servers. [6] We first construct an overlay that is a complete graph among these 1000 hosts. In order to be consistent with our modeling assumption that the least cost overlay tree is used, we compute the minimum spanning tree on that graph. An important difference, however, is that a host in the overlay tree enforces an upper degree bound (UDB) on the maximum number of children, to simulate bandwidth constraints.[7]

Figure 9(c) shows the normalized overlay cost versus the number of members with UDB=6. Four different random number generator seeds (RNG_seed=3,5,7,9) are used for the assignment of hosts. We observe that the results are consistent with our modeling results. The normalized overlay cost is asymptotically close to $n^{0.85}$ or so, for a small number of members (< 100). The value was higher ($n^{0.87}$) when we repeated the same experiment with UDB=1. The tree cost saturates at around 36, when the number of members is ≈ 100, which is earlier than the curves in Figure 9(b). This can be attributed to the usage of only 60 routers to which hosts are connected in the simulation, versus a much larger number of underlying routers used in Figure 9(b).

We have also examined the normalized overlay cost via simulations of the three overlay protocols on the topologies described in Section III-B. The results reveal that ESM and MDDBST behave asymptotically close to $n^{0.8}$ to $n^{0.9}$ or so, before they saturate, which is consistent with our analytical results. TAG has a slightly higher cost than ESM and MDDBST. Partial path matching in TAG may incur higher costs due to the u unmatched routers allowed with high *bwthresh* values. We also found that the normalized cost was higher for the GT-ITM topologies than for the power-law and small-world topologies, since router degree and clustering properties are exploited by overlay protocols to reduce stress and cost.

To further validate our results, we compute the stress and overlay cost for the real ESM tree used in Section III-A. We find that the maximum stress is 12, the total stress is 696, and the overlay tree cost is 568. Since the average unicast path length is $\approx$ 12.01, the normalized overlay cost is $\frac{568}{12.01} \approx$ 47.3. Since $n = 59$ (we only use hosts for which we could obtain underlying routes), the normalized tree cost $\approx n^{0.945}$.

V. Related Work

The objectives of our work are similar to those of work evaluating IP multicast efficiency. Chuang and Sirbu [14] were first to investigate the efficiency of IP multicast in terms of network traffic load. They found that the ratio between the total number of multicast links and the average unicast path length exhibits a power-law with respect to the number of distinct sites with multicast receivers ($m^{0.8}$). Their conclusion was based on real and generated network topologies. Chalmers and Almeroth [13] subsequently investigated the efficiency of IP multicast over unicast experimentally. They carefully analyzed numerous real and synthetic Internet data sets. They argue that the normalized tree cost is closer to $n^{0.7}$ than to $n^{0.8}$. In addition, their results indicate that multicast trees typically include a high frequency (70 to 80%) of unary nodes.

In order to precisely understand the causes of IP multicast traffic reduction, several mathematical models have been devised. Phillips *et al.* [15] were first to derive asymptotic forms for the power-law in k-ary trees and more general networks.

[6] The total number of routers including unary routers is approximately 18,957.

[7] Hosts connected to the same router are not considered in the UDB check.

Their models, however, are approximate and cannot precisely explain the 0.8 (or 0.7) power-law. Adjih *et al.* [16] obtained more accurate asymptotic forms of the power-law. They show that the essence of the problem is the modeling assumption. To prove this, the simple k-ary tree used in [15] is abandoned, and a k-ary *self-similar* tree is used. The authors argue that the self-similar tree provides a plausible explanation of the power-law. However, no experimental data is given to support that IP multicast trees are indeed self-similar, i.e., the number of unary nodes decreases as the tree level increases. Mieghem *et al.* [34] have also analyzed the Chuang and Sirbu result. The expected number of joint hops in a shortest-path multicast tree is used to compute the expected number of links.

We consider the case of overlay multicast, not IP multicast, in this paper. A number of overlay multicast protocols have been proposed over the last three years. ESM (or Narada) [1], [17] was one of the earliest approaches. ESM hosts exchange group membership and routing information to build a mesh, and then execute a DVMRP-like protocol to construct a forwarding tree. A hierarchical approach to improve scalability is proposed in [3]. In [5], the authors utilize host degree constraints and diameter bounds to centrally compute an optimal overlay multicast network. TAG [4] uses route overlap as a heuristic for constructing a low-delay overlay tree in a distributed manner.

Perhaps the work that comes closest to ours is presented in [33] and [28]. Radoslavov *et al.* [33] characterized real and generated topologies with respect to neighborhood size growth, robustness, and increase in path lengths due to link failure. They briefly analyzed the impact of topology on two heuristic overlay multicast strategies, in terms of stretch (the ratio of the number of links in overlay multicast to that in IP multicast) and maximum link stress. Jin and Bestavros [28] have shown that both Internet AS-level and router-level graphs exhibit small-world behavior, due to power-law degree distributions and preference to local connections. They also outlined how small-world behavior affects the overlay multicast tree size.

VI. Conclusions and Future Work

We have characterized overlay multicast trees via experimental data and simulations of three overlay multicast protocols. We also have derived an expression for the overlay cost, defined as the total number of hops in all overlay links. Based on our results, we can make the following observations. First, the experimental data and simulations illustrate that both the mean number of hops, per-hop delay, and total delay between parent and child hosts tend to decrease as the level of the host in the overlay tree increases. Our analysis suggests that routing strategies in overlay multicast protocols, along with power-law and small-world Internet topology characteristics, play a key role in explaining these phenomena. Non-uniform multicast host distribution reinforces them. Second, our models behave asymptotically close to power-laws, ranging from $n^{0.83}$ to $n^{0.92}$ for n hosts. Simulations and experimental data validate our models, and show the tradeoffs in overlay trees constructed via three different protocols. We can quantify potential bandwidth savings of overlay multicast compared to unicast since $n^{0.9} < n$, and the bandwidth penalty of overlay multicast compared to IP multicast ($n^{0.9} > n^{0.8}$). The overlay protocol routing and Internet topology characteristics, in addition to host distribution, contribute to further reducing the overlay costs. This sheds light on the effectiveness of various overlay protocol design methodologies. We plan to conduct larger-scale experiments to better understand overlay tree properties, and their correlations with underlying network characteristics.

References

[1] Y. Chu, S. Rao, S. Seshan, and H. Zhang, "Enabling Conferencing Applications on the Internet using an Overlay Multicast Architecture," in *Proc. of ACM SIGCOMM*, August 2001, pp. 55–67.

[2] J. Jannotti, D. Gifford, K. Johnson, M. Kaashoek, and J. O. Jr., "Overcast: Reliable multicasting with an overlay network," in *Proc. of OSDI*, October 2000.

[3] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable Application Multicast," in *Proc. of ACM SIGCOMM*, August 2002.

[4] M. Kwon and S. Fahmy, "Topology-Aware Overlay Networks for Group Communication," in *Proc. of ACM NOSSDAV*, May 2002, pp. 127–136.

[5] S. Shi, J. Turner, and M. Waldvogel, "Dimensioning server access bandwidth and multicast routing in overlay networks," in *Proc. of ACM NOSSDAV*, June 2001, pp. 83–91.

[6] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: a Case for Informed Internet Routing and Transport," *IEEE Micro*, vol. 1, no. 19, pp. 50–59, January 1999.

[7] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient Overlay Networks," in *Proc. of ACM SOSP*, October 2001, pp. 131–145.

[8] J. Byers, J. Considine, M. Mitzenmacher, and S. Rost, "Informed Content Delivery Across Adaptive Overlay Networks," in *Proc. of ACM SIGCOMM*, August 2002.

[9] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications," in *Proc. of ACM SIGCOMM*, August 2001, pp. 149–160.

[10] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," in *Proc. of ACM SIGCOMM*, August 2001, pp. 161–172.

[11] A. Rowstron and P. Druschel, "Pastry: Scalable, Decentralized Object Location and Routing for Large-scale Peer-to-Peer Systems," in *Proc. of ACM/IFIP Middleware*, 2001.

[12] S. Fahmy and M. Kwon, "Characterizing Overlay Multicast Networks," in *Proceedings of the IEEE ICNP*, November 2003, pp. 61–70.

[13] R. Chalmers and K. Almeroth, "Modeling the Branching Characteristics and Efficiency Gains in Global Multicast Trees," in *Proc. of IEEE INFOCOM*, April 2001, pp. 449–458.

[14] J. Chuang and M. Sirbu, "Pricing Multicast Communications: A Cost-Based Approach," in *Proc. of Internet Society INET*, July 1998.

[15] G. Phillips, S. Shenker, and H. Tangmunarunkit, "Scaling of Multicast Trees: Comments on the Chuang-Sirbu scaling law," in *Proc. of ACM SIGCOMM*, 1999, pp. 41–51.

[16] C. Adjih, L. Georgiadis, P. Jacquet, and W. Szpankowski, "Multicast Tree Structure and the Power Law," in *Proc. of SODA*, 2002.

[17] Y. Chu, S. Rao, and H. Zhang, "A Case for End System Multicast," in *Proc. of ACM SIGMETRICS*, June 2000, pp. 1–12.

[18] "Traceroute.org," http://www.traceroute.org.

[19] Y. Zhang, V. Paxson, and S. Shenker, "The stationarity of internet path properties: Routing, loss, and throughput," ACIRI technical report, May 2000, http://www.icir.org/vern/papers.html.

[20] D. V. Houweling, "Internet 2," http://www.internet2.edu.

[21] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On Power-Law Relationships of the Internet Topology," in *Proc. of ACM SIGCOMM*, August 1999, pp. 251–262.

[22] D. Watts and S. Strogatz, "Collective Dynamics of Small-world Networks," *Nature*, vol. 363, pp. 202–204, 1998.

[23] A. Barabasi and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, pp. 509–512, 1999.

[24] L. Peterson, T. Anderson, D. Culler, and T. Roscoe, "A Blueprint for Introducing Disruptive Technology into the Internet," in *Proceedings of the HotNets-I*, October 2002.

[25] A. Kuznetsov, "Tracepath," ftp://ftp.inr.ac.ru/ip-routing/iputils-current.tar.gz.

[26] S. Fahmy and M. Kwon, "Characterizing overlay multicast networks and their costs," Purdue University, Tech. Rep. CSD-TR-04-007, 2004.

[27] J. Winick and S. Jamin, "Inet-3.0: Internet Topology Generator," Univ. of Michigan, Tech. Rep. UM-CSE-TR-456-02, 2002.

[28] S. Jin and A. Bestavros, "Small-World Internet Topologies: Possible Causes and Implications on Scalability of End-System Multicast," Boston University, Tech. Rep. BUCS-TR-2002-004, 2002.

[29] E. Zegura, K. Calvert, and S. Bhattacharjee, "How to Model an Internetwork," in *Proc. of IEEE INFOCOM*, vol. 2, March 1996, pp. 594 –602.

[30] 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*, vol. 20, no. 8, October 2002.

[31] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Topologically-Aware Overlay Construction and Server Selection," in *Proc. of IEEE INFOCOM*, vol. 3, June 2002, pp. 1190–1199.

[32] N. Malouch, Z. Liu, D. Rubenstein, and S. Sahu, "A Graph Theoretic Approach to Bounding Delay in Proxy-Assisted, End-System Multicast," in *Proc. of IWQoS*, May 2002.

[33] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin, "On Characterizing Network Topologies and Analyzing Their Impact on Protocol Design," Dept. of Computer Science, University of Southern California, Tech. Rep. USC-CS-TR-00-731, February 2000.

[34] P. Mieghem, G. Hooghiemstra, and R. Hofstad, "On the Efficiency of Multicast," *IEEE/ACM Transactions on Networking*, vol. 9, no. 6, December 2001.

[35] W. Szpankowski, *Average Case Analysis of Algorithms in Sequences*. John Wiley & Sons, New York, 2001.

APPENDIX

Proof of Lemma 1

$$\begin{aligned}
&L_\nu(h-1,k,n) = kp(k^{(h-2)\theta} + L_\nu(h-2,k,n)) \\
&+\sum_{i=1}^{h-2} k^i(1-p)^i\{B(h-i-2) \\
&+kpL_\nu(h-i-2,k,n)\}. \\
&L_\nu(h,k,n) = kp(k^{(h-1)\theta} + L_\nu(h-1,k,n)) \\
&+\sum_{i=1}^{h-1} k^i(1-p)^i\{B(h-i-1) + kpL_\nu(h-i-1,k,n)\} \\
&= k^{(h-1)\theta+1}p + k^2p^2(k^{(h-2)\theta} + L_\nu(h-2,k,n)) \\
&+kp\sum_{i=1}^{h-2} k^i(1-p)^i\{B(h-i-2) \\
&+kpL_\nu(h-i-2,k,n)\} \\
&+\sum_{i=1}^{h-1} k^i(1-p)^i\{B(h-i-1) + kpL_\nu(h-i-1,k,n)\}
\end{aligned}$$

$$\begin{aligned}
&= k^{(h-1)\theta+1}p + k^2p^2k^{(h-2)\theta} + k^2pL_\nu(h-2,k,n) \\
&+\sum_{i=2}^{h-1} k^i(1-p)^{i-1}\{B(h-i-1) \\
&+kpL_\nu(h-i-1,k,n)\} + k(1-p)B(h-2) \\
&= k^{(h-1)\theta+1}p + p^2(k^2k^{(h-2)\theta} + k^3k^{(h-3)\theta}) \\
&+k^3pL_\nu(h-3,k,n) + \sum_{i=3}^{h-1} k^i(1-p)^{i-2}\{B(h-i-1) \\
&+kpL_\nu(h-i-1,k,n)\} + k(1-p)(B(h-2) + kB(h-3)),
\end{aligned}$$

where

$$B(h-i-1) = k^{(h-i-1)\theta}(2k^\theta + 2kp - 1)(1-(1-k^{-i})^n). \tag{17}$$

Repeating this process yields

$$\begin{aligned}
&L_\nu(h,k,n) = k^{(h-1)\theta+1}p + p^2\sum_{i=2}^{h-1} k^ik^{(h-i)\theta} \\
&+k^{h-1}pL_\nu(1,k,n) + k^{h-1}(1-p)B(0) \\
&+k(1-p)\sum_{i=0}^{h-3} k^iB(h-i-2) \\
&= k^{(h-1)\theta+1}p + k^{h\theta}p^2\sum_{i=2}^{h-1} k^{(1-\theta)i} + k^hp^2 \\
&+k^{(h-2)\theta+1}(1-p)(2k^\theta + 2kp - 1)\sum_{i=0}^{h-2} k^{(1-\theta)i} \\
&-k^{(h-2)\theta+1}(1-p)(2k^\theta + 2kp - 1)\sum_{i=1}^{h-1} k^{(1-\theta)i}(1-k^{-i})^n,
\end{aligned}$$

where $L_\nu(1,k,n) = kp$ and $L_\nu(0,k,n) = 0$. Since $j = h-1-i$, we have

$$\sum_{i=1}^{h-1} k^{(1-\theta)i}(1-k^{-i})^n = k^{(1-\theta)(h-1)}\sum_{j=0}^{h-2} k^{-(1-\theta)j}\left(1-\frac{k^j}{k^{h-1}}\right)^n. \tag{18}$$

As analyzed in the Appendix A.1 of [16],

$$\sum_{j=0}^{h-2} k^{-(1-\theta)j}\left(1-\frac{k^j}{k^{h-1}}\right)^n = c_2(a,\theta) + O(1), \tag{19}$$

where $c_2(a,\theta) = \sum_{i=0}^{\infty} k^{-(1-\theta)i}e^{-ak^{i+1}}$. Thus,

$$\sum_{i=1}^{h-1} k^{-(1-\theta)i}(1-k^{-i})^n = k^{(1-\theta)(h-1)}c_2(a,\theta) + O(1). \tag{20}$$

Finally,

$$\begin{aligned}
&L_\nu(h,k,n) = k^{(h-1)\theta+1}p + (k^h + k^{h\theta}\sum_{i=2}^{h-1} k^{(1-\theta)i})p^2 \\
&+k^{(h-2)\theta+1}(1-p)(2k^\theta + 2kp - 1)\sum_{i=0}^{h-2} k^{(1-\theta)i} \\
&-k^{h-\theta}(1-p)(2k^\theta + 2kp - 1)c_2(a,\theta) + O(1).
\end{aligned}$$

■

Lemma 2: For a fixed ratio $a = \frac{n}{k^h}$, when $0 < a < \infty$, $L_o(h,k,n)$ has the following asymptotic expansions:
(i) When $\log_k(1-(1-\frac{1}{k})^{\frac{1}{n}})^{-1} - 1 < \rho$,

$$L_o(h,k,n) = \frac{k^{h\theta}-1}{k^{\theta}-1}k^{\rho}(1-(1-k^{-\rho})^n), \tag{21}$$

(ii) Otherwise, that is, when n is large,

$$\begin{aligned} L_o(h,k,n) &= \frac{k^{h\theta}-1}{k^{\theta}-1}k^{\rho} + 2(k^h - k^{(h-\rho)\theta+\rho}) \\ &\quad \times \left(\frac{k-1}{k-k^{\theta}} - c_1(a,\theta)\right) + O(1), \end{aligned} \tag{22}$$

where $c_1(a,\theta) = \sum_{i=0}^{\infty} k^{(-1+\theta)i}e^{-ak^i}$.

Proof: The result in (i) is easily obtained when $g(l) = 0$. In (ii), we only need to compute the following.

$$\begin{aligned} \sum_{l=\rho}^{h-1} k^l g(l) &= 2\sum_{l=\rho}^{h-1} k^l k^{(h-l-1)\theta}(k(1-(1-k^{-(l+1)})^n) - 1) \\ &= 2k^{-\theta}\{\sum_{l=\rho}^{h-1} k^{(h-l)\theta}k^{l+1}(1-(1-k^{-(l+1)})^n) \\ &\quad - \sum_{l=\rho}^{h-1} k^{(h-l)\theta}k^l\}. \end{aligned} \tag{23}$$

Since $i = l+1$, the first term in Equation (23) is computed as follows.

$$\begin{aligned} &\sum_{l=\rho}^{h-1} k^{(h-l)\theta}k^{l+1}(1-(1-k^{-(l+1)})^n) \\ &= k^{\theta}\sum_{i=\rho+1}^{h} k^{(h-i)\theta}k^i(1-(1-k^{-i})^n) \\ &= k^{\theta}\{\sum_{i=1}^{h} k^{(h-i)\theta}k^i(1-(1-k^{-i})^n) \\ &-k^{(h-\rho)\theta}\sum_{i=1}^{\rho} k^{(\rho-i)\theta}k^i(1-(1-k^{-i})^n)\}. \end{aligned}$$

This can be rewritten as

$$\begin{aligned} &k^{h+\theta}\left(\frac{k^{1-\theta}}{k^{1-\theta}-1} - c_1(a,\theta)\right) \\ &-k^{(h-\rho)\theta+\rho+\theta}\left(\frac{k^{1-\theta}}{k^{1-\theta}-1} - c_1(a,\theta)\right) + O(1). \end{aligned} \tag{24}$$

Using the analysis in Appendix A.1 of [16],

$$\begin{aligned} &\sum_{i=1}^{h} k^{(h-i)\theta}k^i(1-(1-k^{-i})^n = \\ &\qquad k^h\left(\frac{k^{1-\theta}}{k^{1-\theta}-1} - c_1(a,\theta)\right) + O(1), \end{aligned} \tag{25}$$

where

$$c_1(a,\theta) = \sum_{i=0}^{\infty} k^{(-1+\theta)i}e^{-ak^i}. \tag{26}$$

The second term in Equation (23) is

$$\sum_{l=\rho}^{h-1} k^{(h-l)\theta}k^l = \frac{k^{h+\theta} - k^{(h-\rho+1)\theta+\rho}}{k-k^{\theta}}. \tag{27}$$

Now, $\sum_{l=\rho}^{h-1} k^l g(l)$ becomes

$$\begin{aligned} &\sum_{l=\rho}^{h-1} k^l g(l) = 2k^{-\theta}((k^{h+\theta} - k^{(h-\rho)\theta+\rho+\theta}) \\ &\times\left(\frac{k^{1-\theta}}{k^{1-\theta}-1} - c_1(a,\theta)\right) - \frac{k^{h+\theta} - k^{(h-\rho+1)\theta+\rho}}{k-k^{\theta}}) + O(1) \\ &= 2(k^h - k^{(h-\rho)\theta+\rho})\left(\frac{k-1}{k-k^{\theta}} - c_1(a,\theta)\right) + O(1). \end{aligned} \tag{28}$$

From equation (28), when n is large,

$$\begin{aligned} &L_o(h,k,n) = \frac{k^{h\theta}-1}{k^{\theta}-1}k^{\rho} \\ &+2(k^h - k^{(h-\rho)\theta+\rho})\left(\frac{k-1}{k-k^{\theta}} - c_1(a,\theta)\right) + O(1). \end{aligned} \tag{29}$$

■

AUTHOR BIOGRAPHIES

Sonia Fahmy [SM] is an associate professor at the Computer Science department at Purdue University. She received her PhD degree from the Ohio State University in 1999. She is currently investigating Internet tomography, overlay networks, network security, and wireless sensor networks. She received the National Science Foundation CAREER award in 2003, and the Schlumberger technical merit award in 2000. She is a member of the ACM. For more information, please see: http://www.cs.purdue.edu/∼fahmy/

Minseok Kwon [M] is an assistant professor in the Department of Computer Science at Rochester Institute of Technology (RIT). He received his Ph.D. degree in Computer Science from Purdue University in 2004. His main research interests are in peer-to-peer networks, network security, and wireless networks. He is a co-chair of the IEEE Communications and Aerospace Joint Chapter in Rochester, NY. He is a member of the ACM. For more information, please see: http://www.cs.rit.edu/∼jmk/