This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

A Cost-based Evaluation of End-to-End Network Measurements in Overlay Multicast

Xing Jin Qiuyan Xia S.-H. Gary Chan
Department of Computer Science and Engineering, HKUST
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
Email: {csvenus, xiaqy, gchan}@cse.ust.hk

***Abstract*— Application-layer multicast (ALM, or overlay multicast) has been proposed to overcome limitations in IP multicast. While much measurement work (such as delay or connectivity measurement) has been conducted to build efficient ALM trees, several interesting questions therein are left unclear: (1) What are the measurement costs of the different measurement methods? (2) What is the major improvement to an ALM tree by using a certain measurement method? (3) To achieve the target tree performance, what is the most cost-efficient measurement method?**

In this paper, we study three representative measurement methods, i.e., delay measurement, connectivity measurement and available bandwidth measurement. We select six typical ALM protocols each adopting at least one of the measurement methods and evaluate their performance on Internet-like topologies. Our study shows that delay and connectivity measurements can effectively reduce end-to-end delay in overlay trees with low measurement costs. Only using delay measurement may lead to a tree consuming much network resource. The use of connectivity and bandwidth measurements can build a tree with low resource consumption.

I. Introduction

In the last few years, application-layer multicast has emerged as a promising technique to overcome limitations in IP multicast [1]–[6]. In ALM, hosts instead of routers are responsible for replicating and forwarding multicast packets. Multicast is hence achieved via piece-wise unicast connections. In contrast to IP multicast, ALM is built without the need of multicast routers. The existing solutions and functionalities of unicast protocols can be straightforwardly applied to ALM. However, ALM is not as efficient as IP multicast. Thus a major concern of ALM is how to build an efficient overlay tree for data delivery.

To achieve efficient data delivery, an ALM protocol often conducts end-to-end measurements to infer the underlay network among hosts. For example, Narada, DT, ALMI, NICE and SIM use PING to select a close parent for a new host [2], [3], [6]. TAG uses TRACEROUTE to infer the path connectivity among hosts and constructs a tree with low delay and stress [4]. Overcast measures available path bandwidth to build a high-bandwidth tree [1]. FAT uses both TRACEROUTE and bandwidth measurement tools to further improve the transmission rate [5]. However, few of these ALM protocols have considered *measurement cost* (i.e., bandwidth consumption for measurement) in system design. Considering that the measurement cost may be huge (e.g., in large scale measurements), it is important to know the relationship between the performance improvement and the measurement cost. In our study, we hence explore the following issues:

- What are the measurement costs of the different measurement methods in ALM protocols? To answer this question, we need to first quantify the cost of one measurement, and then study the number of measurements in a typical ALM protocol. We can then compute the overall cost of a certain measurement method in the protocol.
- What is the major improvement to an ALM tree by using a certain measurement method? An ALM tree can be evaluated in terms of many metrics, and a certain measurement method may improve one or multiple of them. We need to study the major functionality of each measurement method. Based on this, we can answer the third question:
- To achieve the target tree performance, what is the most cost-efficient measurement method? We need to set up selection criteria for the measurement methods based on their costs and functionalities. An application with certain performance requirements can then choose proper measurement methods.

In this paper, we provide a quantitative study to understand the above questions. We consider inferring the underlay network among a session of hosts by means of end-to-end measurements, which include: (1) delay measurement using PING; (2) connectivity measurement using TRACEROUTE, and (3) available bandwidth measurement using tools like Pathload [7]. We first analyze the cost of one measurement for each method. We then select six representative ALM protocols that adopt at least one of the measurement methods and evaluate their performance on Internet-like topologies. Our major findings are listed in Section III-D.

We briefly review the related work as follows. Many works have studied how to infer the underlay information among hosts. Max-Delta can infer a highly accurate topology (in terms of link connectivity and delay) among a group of hosts with low number of traceroutes [5], [8]. Donnet et al. note that

This work was supported, in part, by the Innovation and Technology Commission of the Hong Kong Special Administrative Region, China (GHP/045/05).

a router is often repeatedly visited in different traceroutes in large scale measurements, and propose a Doubletree algorithm to reduce the redundancy [9]. Different from these works, we are not interested in the inference of the underlay. Instead, we study the impact of underlay-awareness to the performance of ALM protocols. On the other hand, these works are orthogonal to ours and can be used in our evaluation framework. There have also been a lot of works on the performance of measurement tools. In [10], the authors study the key parameters in the PING process (such as the number of probes and the measurement interval) in order to obtain an accurate RTT on a path with the minimum bandwidth consumption. In [11], the authors compare three types of measurement-based techniques for peer selection in peer-to-peer systems. Our work does not focus on the details of any measurement tools. Instead, we select the commonly used parameters for the tools and study their impacts to the ALM protocols.

The rest of the paper is organized as follows. In Section II we describe the background of end-to-end measurements and our evaluation methodology. In Section III we present and discuss our simulation results. We finally conclude in Section IV.

II. Background and Evaluation Methodology

A. End-to-End Measurements in ALM

We consider three types of end-to-end measurements in ALM, i.e., delay measurement, connectivity measurement and available bandwidth measurement. We now analyze their bandwidth consumption at the measurement initiator.

1) Delay measurement: PING program is the most frequently used tool to measure the round-trip time (RTT) between two hosts. The program sends an ICMP (Internet Control Message Protocol) echo request message to a host, expecting an ICMP echo reply to be returned. A typical PING program sends an echo request once a second. Among all the returned results, the minimum RTT is used to compute the path delay.

An ICMP echo message is formed by a header (of 8 bytes) and some optional data (often of 32 bytes). With a 20-byte IP header, one IP datagram is of 60 bytes. There is no consensus on the number of messages that are used to measure the minimum RTT of a path. According to [10], we set the number of messages in each PING to 10. As a result, the delay measurement along a path incurs $60 \times 2 \times 10 = 1200$ bytes traffic at the sender. The multiplier of 2 is because we are calculating the round trip.

2) Connectivity measurement: To obtain the router-level connectivity information between two hosts, TRACEROUTE-like tools are often used. TRACEROUTE is also implemented with ICMP messages. Each time, the source sends out an IP datagram with a certain TTL value to the destination. Each router that handles the datagram is required to decrement the TTL by one. When a router receives an IP datagram whose TTL is 1, it throws away the datagram and returns an ICMP "time exceeded" error message back to the source host. The IP datagram containing this ICMP message has the router's name, IP address and RTT to the source. In another case, if the datagram arrives at the destination with an unused port number (usually larger than $30,000$), the destination host generates an ICMP "port unreachable" error message and returns it to the source host. Therefore, in TRACEROUTE, the source host sends out a series of IP datagrams with increasing TTL to the destination and each datagram can identify one router in the path. The whole router-level path is hence identified.

An outgoing UDP datagram in TRACEROUTE contains 12 bytes of user data, 8 bytes of UDP header, 20 bytes of IP header for a total of 40 bytes. Unlike PING, however, the size of the returned datagram changes. The returned ICMP message contains 20 bytes of IP header, 8 bytes of ICMP header, 20 bytes of IP header of the datagram that caused the error, 8 bytes of UDP header for a total of 56 bytes. Given a path of N hops, the TRACEROUTE program sends out a series of ICMP messages with TTL from 1 to N, and for each TTL value three ICMP messages are sent. This incurs $3 \times N \times (40 + 56) = 288N$ bytes of bandwidth consumption at the sender.

3) Available bandwidth measurement: In this paper, we are interested in the available bandwidth on a path instead of the bottleneck bandwidth. Many tools have been proposed for such purpose, e.g., Pathload [7], PTR/IGI [12], TOPP [13], Delphi [14], Pathchirp [15] and Spruce [16]. These tools incur different measurement traffics. From [16], the measurement traffic is from 2.5MB to 10MB for Pathload, 130KB for IGI and 300 KB for Spruce. In our study, we simply take a rough average of the above values and assume that each bandwidth measurement incurs 1MB traffic.

B. Evaluation Metrics

We use the following metrics to evaluate an ALM tree:

- *Relay delay penalty (RDP)*: defined as the ratio of the overlay delay from the source to a given host to the delay along the shortest unicast path between them [2]. It is used to quantify the increase in delay as compared with IP multicast.
- *Link stress*: defined as the number of copies of a packet transmitted over a certain physical link [2]. We would like to keep the stresses of links as low as possible.
- *Resource usage*: defined as $\sum_{i=1}^{L} d_i \times s_i$, where L is the number of links active in data transmission, d_i is the delay of link i and s_i is the stress of link i [2]. Resource usage is a metric of the network resource consumed in data delivery. An implicit assumption here is that a link with higher delay tend to be associated with a higher cost.

C. Protocols

We study three classes of ALM protocols.

- The first class includes Narada [2], GNP-based DT [17] and SIM [6]. These protocols take use of PING to construct low-delay overlay trees. According to [17], GNP-based DT has a constant measurement cost for each host. A host only needs to ping 20 landmarks and has a measure cost of around 24KB.

- The second class includes TAG [4] and FAT [5]. These protocols measure link-level connectivity to improve tree performance. In TAG, each host measures its traceroute path from the source and has a measurement cost of around 2.6KB. FAT infers the router-level topology among hosts to construct a high-bandwidth tree. As in [5], we use Max-Delta to select paths to traceroute or measure bandwidth. In our simulations, we evaluate two versions of FAT: a) Each host measures the bandwidth of the paths that have been tracerouted (denoted as *FAT-BW*). b) Each host only conducts traceroutes and does not measure path bandwidth (denoted as *FAT*). The scheme assumes that all the links have the same bandwidth and will build a tree with low link stress.
- The last class includes Overcast [1] and aforementioned FAT-BW. They measure path bandwidth to improve tree performance. Overcast aims at constructing a tree with low link stress. It does not use TRACEROUTE and only measures path bandwidth. In Overcast, the number of paths measured by a host depends on the tree depth and the bandwidth distribution, which cannot be manually controlled. In our simulations, the number of hosts in the tree is fixed (i.e., 1024) and the average measurement cost at each Overcast host is around 17MB. In the following discussion, we assume that Overcast has a constant measurement cost of 17MB for each host.

D. Simulation Setup

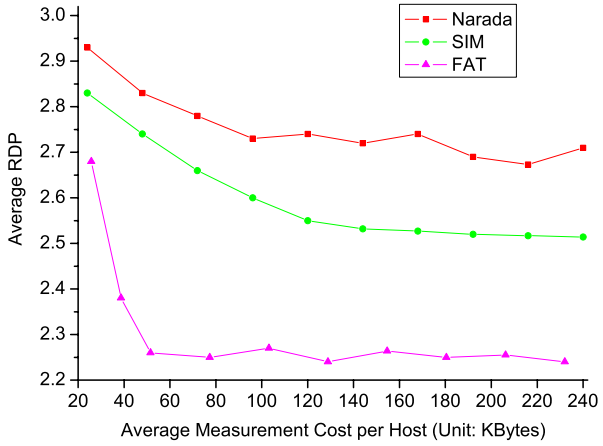
We generate 5 Transit-Stub topologies with GT-ITM [18]. Each topology is a two-layer hierarchy of transit networks (with 8 transit domains) and stub networks (with 256 stub domains). Each topology contains 3200 routers and about 20000 links. A host is randomly connected to a router with 1ms delay, while the delays of core links are given by the topology generator. The number of hosts in a session is set to 1024. Suppose that the target bandwidth is 1 unit. The bandwidth of a backbone link (at least one end point is a transit node) is set to 6, and that of a none-backbone link is uniformly distributed in $[1, 3]$.

We use shortest path routing to identify a path between a pair of hosts and assume that paths are symmetric. For protocols without bandwidth measurement (i.e., Narada, GNP-based DT, SIM, TAG and FAT), each host has a degree bound of 9. That is, a host can have at most 8 children. For protocols with bandwidth measurement (i.e., FAT-BW and Overcast), there is no such bound and the selection of overlay paths is based on path bandwidth. A specific ALM protocol may have multiple tunable parameters. To fairly compare the protocols, we use the *overall convex hull* as proposed in [19] to illustrate the results.

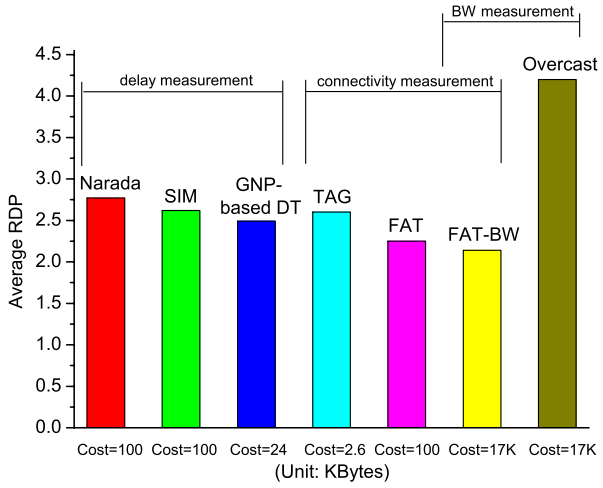
III. Results

A. Relative Delay Penalty

Figure 1 shows the average RDP achieved by different protocols. In Fig. 1(a), we show the average RDP versus the measurement cost in Narada, SIM and FAT. The results of



a) Average RDP versus average measurement cost per host.



b) Average RDP achieved by different protocols.

Fig. 1. Overall convex hulls for average RDP achieved by different protocols.

GNT-based DT, TAG and Overcast are not shown, since they have constant measurement costs for each host. The result of FAT-BW is also not shown, for it measures path bandwidth and incurs a much higher measurement cost.

The average RDP achieved by Narada decreases at the beginning as the number of PINGs increases. When the measurement cost reaches a certain value (around 96KB), its average RDP does not change much and fluctuates around 2.7. Note that Narada uses inconsistent criteria for adding and dropping paths in the mesh, and it is hard to form a stable mesh. SIM shows a much smooth curve than Narada. In SIM, the more hosts a new host pings, the closer parent it can find. From the figure, the improvement in RDP becomes subtle when the measurement cost is larger than 120KB in SIM. FAT shows a significant reduction in RDP when the measurement cost increases from 25KB to 50KB (i.e., the

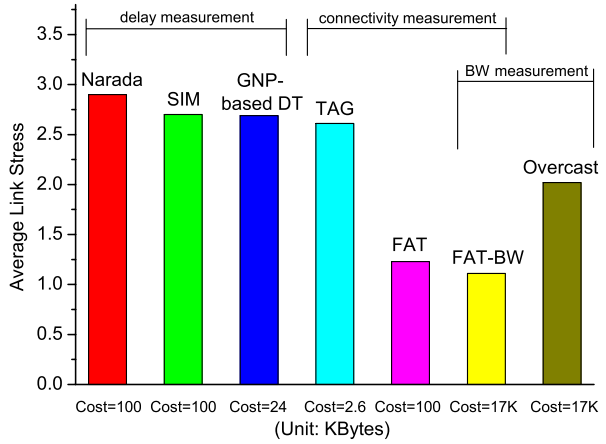


Fig. 2. Overall convex hulls for link stress achieved by different protocols.

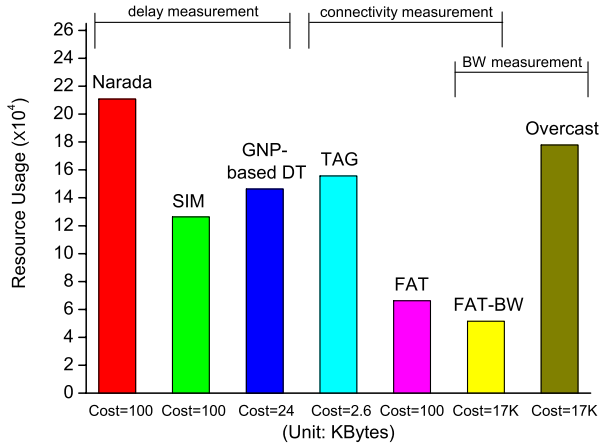


Fig. 3. Overall convex hulls for resource usage achieved by different protocols.

number of traceroutes increases from 10 to 20). When the measurement cost continues increasing, the RDP tends to stabilize around 2.3. In our simulations, when the numbers of traceroutes at each host are 10 and 20, Max-Delta can discover a topology with 83% and 93.5% links of the complete underlay topology, respectively. Clearly, in the latter case, the knowledge of the underlay is much more accurate and the tree constructed shows much lower RDP.

From Fig. 1(a), we can see that the delay and connectivity measurements incur low bandwidth consumption at each host (at most several hundred KB). This is because (1) PING and TRACEROUTE are lightweight tools and can be easily performed, and (2) Each host only needs to ping or traceroute a few hosts to achieve low enough RDP.

Figure 1(b) shows the average RDP achieved by different protocols given a certain measurement cost. For Narada, SIM and FAT, we set the measurement cost to 100KB. As shown in Fig. 1(a), these protocols achieve stable RDP values at such a measurement cost. On the other hand, GNP-based DT, TAG and Overcast have constant measurement costs at each host, which cannot be manually controlled. We hence accordingly show their performance at their own measurement costs. Furthermore, since both FAT-BW and Overcast measure path bandwidth, we set the measurement cost of FAT-BW to that of Overcast (i.e., 17MB) in order to fairly compare them. From the figure, Overcast has a much higher RDP than all the others. This is because it does not optimize the tree in terms of end-to-end delay. Furthermore, to reserve bandwidth for future hosts, each Overcast host is inserted into the tree as far from the source as the bandwidth constraint allows. It hence performs poorly in terms of RDP. All the other protocols have similar RDP values, ranging from 2.14 to 2.77. Among them, FAT-BW achieves the lowest RDP.

In summary, delay measurement can help construct a low-delay overlay tree at a low measurement cost. Connectivity measurement can further reduce RDP to a certain degree. In a session with 1024 hosts, each host only incurs around 100KB traffic for delay or connectivity measurement. This is small as compared to the control overhead (e.g., DONet averagely incurs 200KB control traffic at each host for transferring 10MB data [20]).

B. Link Stress

Figure 2 shows the link stress achieved by different protocols. From the figure, TAG and the three protocols only using PING achieve high stresses of around $2.6 - 2.9$. Overcast has a lower stress of 2.0. FAT and FAT-BW further reduce the stresses to around $1.1 - 1.2$. As shown, the protocols only using PING do not know the router-level connectivity and cannot intelligently select paths to evenly distribute the delivery loads to links. In TAG, a host only measures its traceroute path from the source. Its knowledge of the underlay as well as its path selection mechanism is limited. It hence has a high stress. Overcast does not infer any router-level connectivity information. However, while selecting paths with high available bandwidth, it avoids repeatedly crossing the same links and hence reduces the average stress. FAT has inferred a highly accurate underlay topology and can accordingly build a tree by minimizing the stress. FAT-BW makes further reduction as compared to FAT. With the knowledge of both connectivity and bandwidth, FAT-BW achieves the lowest stress.

Although FAT and FAT-BW achieve the lowest average stress, it does not mean that their trees are the best. A protocol trying to distribute the delivery loads to unused links can also achieve low stress. However, such a tree uses a large number of links and consumes much network resource. We hence continue evaluating the resource usage of the protocols.

C. Resource Usage

Resource usage quantifies the network resource consumed by an overlay tree for the delivery of unit data. From Fig. 3,

Narada has the highest resource usage. This is not surprising since Narada has the second highest RDP and the highest stress among the protocols. SIM, GNP-based DT, TAG and Overcast achieve lower resource usage. Note that although Overcast shows low stress in Fig. 2, its resource usage is the second highest. This is because Overcast often uses long paths and has high end-to-end delay (as shown in Fig. 1(b)). FAT and FAT-BW have the lowest resource usage. This is because they have the lowest RDP and lowest stress among the protocols. Therefore, their trees are much more efficient than the others.

D. Discussion

From the above results, we can draw the following conclusions.

1) Delay and connectivity measurements have low measurement costs while available bandwidth measurement has a high measurement cost.

Delay and connectivity measurements generate a little network traffic (often on the scale of several hundred KB at a host) and have a short measurement duration. On the contrary, available bandwidth measurement generates much more traffic (often hundreds of times) than delay or connectivity measurement and has a long measurement duration.

2) Delay measurement can effectively reduce end-to-end delay. However, if only delay measurement is used, the resultant tree often consumes much network resource.

For applications only focusing on end-to-end delay, we can use PING to reduce the delay. Typical examples are radio broadcasting, news-on-demand and online voice meeting. These applications do not require high transmission rate, but they do require real-time data delivery. From our simulations, delay measurement is sufficient to achieve low delay. However, a tree only based on delay measurement often consumes more network resource than that based on connectivity measurement. This will make inefficient use of the network resource.

3) Connectivity measurement can reduce both end-to-end delay and resource consumption. In a network with homogeneous bandwidth distribution, it can also build a low-stress tree.

As compared to delay measurement, connectivity measurement incurs a similar cost and achieves similar delay. A major advantage of connectivity measurement is that it can significantly reduce the resource usage, which allows an ALM tree to make more efficient use of the network resource. Furthermore, in a network with homogeneous bandwidth distribution, it is possible to build a low-stress tree with only connectivity measurement.

4) If connectivity measurement and available bandwidth measurement are used together, it is possible to build a tree with low delay, low stress and low resource consumption.

IV. Conclusion

ALM protocols often use end-to-end measurements to infer the underlay network in order to build an efficient overlay tree. However, few of them have considered the cost for measurements. In this paper, we quantitatively study the relationship between the performance improvement and the measurement cost. We compare three representative measurement methods and evaluate six ALM protocols that adopt at least one of the measurement methods. Our results show that PING and TRACEROUTE can reduce end-to-end delay with low measurement costs. Available bandwidth measurement has a high measurement cost. With the combination of TRACEROUTE and available bandwidth measurement, it is possible to build a tree with low end-to-end delay, low stress and low resource usage.

References

[1] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O'Toole, "Overcast: Reliable multicasting with an overlay network," in *Proc. OSDI'00*, Oct. 2000, pp. 197–212.

[2] Y. H. Chu, S. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE JSAC*, vol. 20, no. 8, pp. 1456–1471, Oct. 2002.

[3] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," in *Proc. ACM SIGCOMM'02*, Aug. 2002, pp. 205–217.

[4] M. Kwon and S. Fahmy, "Topology-aware overlay networks for group communication," in *Proc. ACM NOSSDAV'02*, May 2002, pp. 127–136.

[5] X. Jin, Y. Wang, and S.-H. G. Chan, "Fast overlay tree based on efficient end-to-end measurements," in *Proc. IEEE ICC'05*, May 2005, pp. 1319–1323.

[6] X. Jin, K.-L. Cheng, and S.-H. G. Chan, "SIM: Scalable island multicast for peer-to-peer media streaming," in *Proc. IEEE ICME'06*, July 2006, pp. 913–916.

[7] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput," in *Proc. ACM SIGCOMM'02*, Aug. 2002, pp. 295–308.

[8] X. Jin, W.-P. K. Yiu, S.-H. G. Chan, and Y. Wang, "Network topology inference based on end-to-end measurements," *IEEE JSAC*, vol. 24, no. 12, pp. 2182–2195, Dec. 2006.

[9] B. Donnet, P. Raoult, T. Friedman, and M. Crovella, "Efficient algorithms for large-scale topology discovery," in *Proc. ACM SIGMETRICS'05*, June 2005, pp. 327–338.

[10] Z. Wang, A. Zeitoun, and S. Jamin, "Challenges and lessons learned in measuring path RTT for proximity-based applications," in *Proc. PAM'03*, April 2003.

[11] T. S. E. Ng, Y. Chu, S. Rao, K. Sripanidkulchai, and H. Zhang, "Measurement-based optimization techniques for bandwidth-demanding peer-to-peer systems," in *Proc. IEEE INFOCOM'03*, April 2003, pp. 2199–2209.

[12] N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," *IEEE JSAC*, vol. 21, no. 6, pp. 879–894, Aug. 2003.

[13] B. Melander, M. Bjorkman, and P. Gunningberg, "A new end-to-end probing and analysis method for estimating bandwidth bottlenecks," in *Proc. IEEE GLOBECOM'00*, Nov. 2000, pp. 415–420.

[14] V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, and R. Baraniuk, "Multifractal cross-traffic estimation," in *Proc. ITC Specialist Seminar*, Sept. 2000.

[15] V. Ribeiro, R. Riedi, R.Baraniuk, J.Navratil, and L. Cottrell, "pathChirp: Efficient available bandwidth estimation for network paths," in *Proc. PAM'03*, April 2003.

[16] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools," in *Proc. ACM SIGCOMM IMC'03*, Aug. 2003, pp. 39–44.

[17] W.-C. W. Wong and S.-H. G. Chan, "Improving Delaunay triangulation for application-level multicast," in *Proc. IEEE GLOBECOM'03*, Dec. 2003, pp. 2835–2839.

[18] E. Zegura, K. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. IEEE INFOCOM'96*, March 1996, pp. 594–602.

[19] J. Li, J. Stribling, R. Morris, M. F. Kaashoek, and T. M. Gil, "A performance vs. cost framework for evaluating DHT design tradeoffs under churn," in *Proc. IEEE INFOCOM'05*, 2005, pp. 225–236.

[20] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, "CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming," in *Proc. IEEE INFOCOM'05*, March 2005, pp. 2102–2111.