Bandwidth-Aware Routing in Overlay Networks

Sung-Ju Lee, Sujata Banerjee, Puneet Sharma, Praveen Yalagandula, and Sujoy Basu
Hewlett-Packard Laboratories
Palo Alto, CA 94304
Email: {sjlee,sujata,puneet,yalagand,basus}@hpl.hp.com

***Abstract*—In the absence of end-to-end quality of service (QoS), overlay routing has been used as an alternative to the default best effort Internet routing. Using end-to-end network measurement, the problematic parts of the path can be bypassed, resulting in improving the resiliency and robustness to failures. Studies have shown that overlay paths can give better latency, loss rate, and TCP throughput. Overlay routing also offers flexibility as different routes can be used based on application needs. There have been very few proposals of using *bandwidth* as the main metric of interest, which is of great concern in media applications. We introduce our scheme BARON (Bandwidth-Aware Routing in Overlay Networks) that utilizes *capacity* between the end hosts to identify viable overlay paths and measures *available bandwidth* to select the best route. We propose our path selection approaches, and using the measurements between 174 PlanetLab nodes and over 13,189 paths, we evaluate the usefulness of overlay routes in terms of bandwidth gain. Our results show that among 658,526 overlay paths, 25% have larger bandwidth than their native IP routes, and over 86% of ⟨source, destination⟩ pairs have at least one overlay route with larger bandwidth than the default IP routes. We also present the effectiveness of BARON in preserving the bandwidth requirement over time for a few selected Internet paths.**

I. Introduction

When end-to-end quality of service routing is not provided in the network infrastructure, routing through intermediate overlay nodes instead of using the default IP routes can alleviate performance problems. By using overlay nodes for forwarding, intermediate nodes on default IP routing paths with transient failures or congestion can be bypassed and hence overlay routing is resilient to failures and provides robustness [1]. It has also been shown that overlay routes can have better performance than native IP routes [19]. Moreover, overlay routing provides multiple routing paths to the end nodes from which they can choose based on the application requirements and the preferred metric. The potential performance gain, robustness, and flexibility make overlay routing a very attractive alternative, especially in best effort networks with unpredictable performance. Further, many of the techniques considered in overlay networking research may be integrated into the core network infrastructure in the future.

Some of the recent Internet applications require high and sustained bandwidth over time. Live high-quality video streaming, video conferencing, and graphic-intensive multi-player games are such applications. While the majority of the recent research in overlay routing focused on finding overlay routes with smaller delay, loss-rate, or higher resilience with respect to the native IP path, few studies focused on bandwidth for overlay routing. Bandwidth (both the capacity and available bandwidth) is of great importance for media applications and we turn our attention to bandwidth as our primary overlay route selection criterion. Here we distinguish between capacity and available bandwidth. Capacity is the maximum possible transfer rate of the path while available bandwidth is the residual capacity of the path one can use and is time-varying [15].

To make routing schemes bandwidth-aware, nodes need up-to-date bandwidth information. A recent work [24] uses only available bandwidth for dynamic overlay routing, and each node measures available bandwidth to a large number of nodes. However, for scalability reasons, it is not efficient or effective for each node to frequently measure the bandwidth to all other nodes. In the case of available bandwidth, it is especially true as the value of available bandwidth fluctuates rapidly and by the time new measurements are obtained, their values may be outdated, especially in large networks. Capacity values on the other hand, are relatively static, although the current tools [3], [10] require longer measurement time and larger probing overhead compared with the available bandwidth measurement tools [9], [16], [21].

We introduce our scheme BARON (Bandwidth-Aware Routing in Overlay Networks) that utilizes both capacity and available bandwidth to quickly locate alternate overlay paths that provide larger bandwidth than the direct path. In BARON, each node performs infrequent periodic capacity measurements to obtain the network capacity snapshot. When a route between two hosts is experiencing problems due to low bandwidth availability, outage, or congestion, BARON promptly finds the candidate alternate overlay paths using the latest capacity measurement values. Available bandwidth is measured for only those small number of identified viable candidate paths, and the best path is selected.

Although there is no strong evidence that there is a direct correlation between path capacity and available bandwidth, we show that using the combination of these two metrics is a viable approach. Available bandwidth best represents the current network snapshot. However, as its value is time-sensitive, its measurement should be event triggered and limited to few paths. We utilize a more stable metric in capacity to find the candidate nodes for the alternate overlay paths.

We use 174 PlanetLab [14] nodes and paths between them to study whether overlay routes that provide larger bandwidth than the native IP routes exist. We show how much capacity gain these overlay paths provide and how much latency they sacrifice. We also show performance results of our BARON

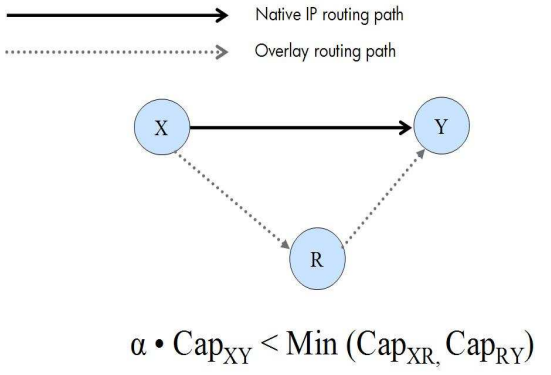


Fig. 1. Overlay path vs native IP path.

scheme.

This paper is organized as follows. The detailed operation of BARON is described in Section II. Section III presents our measurement study. Related work is surveyed in Section IV. Final remarks are made in Section V.

II. Bandwidth-Aware Routing in Overlay Networks

The main idea behind BARON is shown in Figure 1. When the path going through an intermediate overlay node has a larger bandwidth than the native IP network path, it is advantageous to use such an overlay route. But as overlay paths may have longer hops (and possibly longer latency), switching to overlay paths may not be effective when the bandwidth gain from switching to the overlay path is not substantial. Hence we introduce a system/application parameter called "path switching threshold," denoted as α ($\alpha \geq 1$), that is used to prevent unnecessary route changes. We use the following requirement for an overlay node to satisfy bandwidth gain over the native IP path:

$$\min(Cap_{XR}, Cap_{RY}) > \alpha \cdot Cap_{XY} \quad (1)$$

where X and Y are the source and the destination of the path respectively, and R is the overlay node through which the overlay path is routed. Note that $\alpha \geq 1$. When α is 1.1, the bandwidth gain of an overlay route needs to be larger than 10% to make the route switch. In our measurement dataset described in Section III, we measured 13,189 end-to-end paths, and when $\alpha = 1.0$, 11,448 paths have at least one overlay path that have larger bandwidth than the direct IP path. From this data, we surmise that overlay paths that provide larger bandwidth than native IP paths exist.

Our scheme executes the following steps:

1. Each node periodically (with large interval) measures bandwidth capacity to every node in the network.
2. When the native IP path is not providing enough bandwidth between nodes X and Y,
   (a) Check if there exist overlay paths that satisfy Eq. (1).
   (b) If such paths exist, select the top k overlay paths that provide the maximal bandwidth capacity.
   (c) Node X measures available bandwidth to the intermediate nodes of these k overlay paths, and also requests the same intermediate nodes to measure available bandwidth to node Y.
   (d) Node X switches to a path that gives the maximal available bandwidth between nodes X and Y.

We utilize the combination of capacity and available bandwidth. We use the capacity in the initial step as the capacity values are more stable than the available bandwidth. We use available bandwidth for the actual alternate route selection as it better represents the current bandwidth status. Moreover, the probing overhead and the estimation time for available bandwidth is much less than for capacity. Although there has not been a strong evidence of a direct correlation between capacity and available bandwidth, we believe utilizing them together will promptly locate high bandwidth overlay routes.

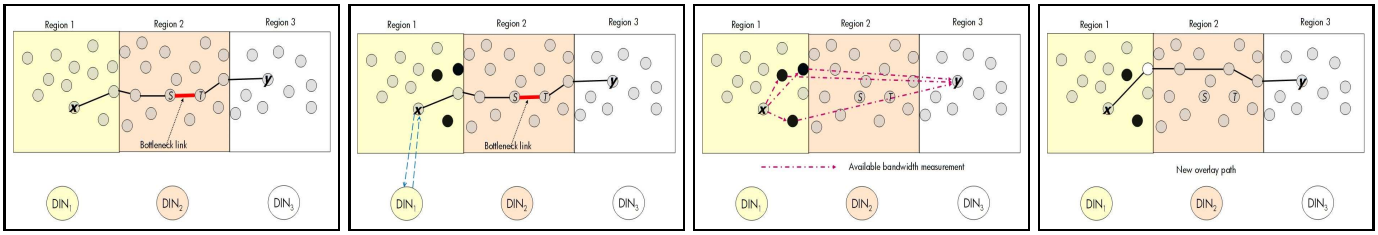
Although measuring capacity to all nodes is inefficient, our scheme performs the network-wide measurement only in the network initiation stage and subsequent capacity measurements are made only when route changes occur. Nodes detect route changes by periodically performing traceroutes. Performing periodic traceroutes incurs less measurement overhead and returns estimates more quickly than periodic capacity measurements.

Note that for simplicity, the above algorithm only uses one-hop intermediate overlay node, but the algorithm can easily be expanded to consider multiple intermediate hops. We evaluate two-hop relays as well as one-hop relay overlay paths in Section III. The key in step 2(b) is to quickly identify a small number (k) of candidate nodes when searching for a new route. We propose two different algorithms, which we describe next.

A. Using Distributed Information Nodes

Making all nodes store and maintain the path information from every node to every other node in the network is not scalable and feasible. Therefore, having an infrastructure node that has a database of all node and path information is advantageous. A node queries this infrastructure node when it needs certain path information. Having a centralized system however, creates a single point of failure. Moreover, due to traffic concentration, node update frequency will be limited, and depending on the location of this infrastructure node, some nodes will have higher latency in acquiring information from the infrastructure node. Replication is one solution, but it increases the network update traffic. Partitioning the node information database across a set of DINs (Distributed Information Nodes) is used in [5] and we adopt this technique in our study. Instead of storing only network position information, the DINs in our system also store bandwidth capacity information. We use the *closest partitioning* [5] approach for its simplicity. In closest partitioning, a node gets assigned to a DIN that has the smallest latency to it.

Let's use Figure 2 (a) as an example. The DIN_1 has all the bandwidth capacity information of the paths that are sourced from the nodes in the Region 1. In Figure 2 (b), a bottleneck



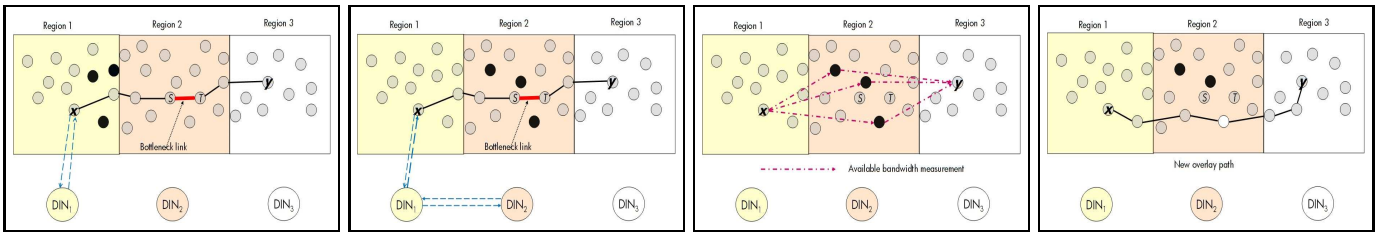
(a) Native IP routing path with a bottleneck link.

(b) Communication with the Distributed Information Node.

(c) Bandwidth measurement.

(d) New overlay route.

Fig. 2. Finding the new overlay route with the DINs.



(a) Communication with a DIN.

(b) Communication with the neighboring DIN.

(c) Bandwidth measurement.

(d) New overlay route.

Fig. 3. Finding the new overlay route with the neighboring DINs.

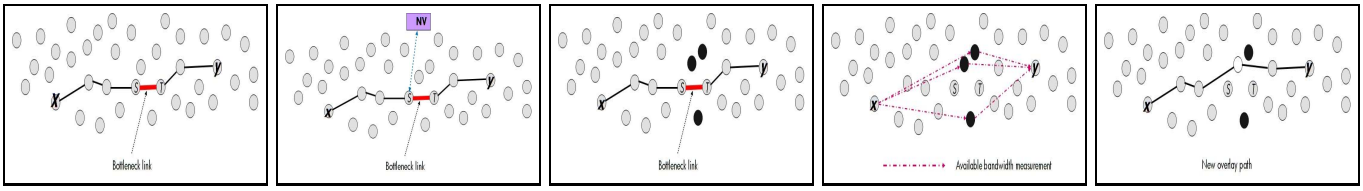
link on the path causes a drop in the end-to-end bandwidth, and the source node X consults the infrastructure node of its region, DIN_1 to find k candidate nodes that satisfy Eq. (1). Note that since DIN_1 has all the required bandwidth capacity information (i.e., from node X to all the nodes in Region 1 and from nodes in Region 1 to the destination node Y), it can easily perform this operation. The source of the path, node X then measures available bandwidth to those k nodes, shown in black, and the k nodes measure available bandwidth to the receiver of the path, node Y, as shown in Figure 2 (c). Node X selects and switches to the new path among those paths that has the largest available bandwidth. This new route is shown in Figure 2 (d).

Suppose that in Figure 2 (b) above (repeated in Figure 3 (a)), DIN_1 cannot identify any node that satisfies Eq. (1). In that case, DIN_1 consults with its neighboring DINs (DIN_2 in this example). Since DIN_2 has all the bandwidth capacity information of the paths from nodes in the Region 2 to the destination node Y, and DIN_1 has all the information of the paths from the nodes in Region 1 to the nodes in Region 2, the k candidate nodes can be found, shown as black nodes in Figure 3 (b). Note the filtering of the candidate nodes are performed in this process. When contacting DIN_2, DIN_1 indicates the set of nodes S within DIN_2 that satisfy the condition of $Cap_{XR} > \alpha \cdot Cap_{XY}$, where $R \in S$. DIN_2 in turn selects nodes in S that satisfy $Cap_{RY} > \alpha \cdot Cap_{XY}$ and returns those nodes and their Cap_{RY} to the source. The source of the path node X selects the top k nodes and measures available bandwidth to those k nodes, and the k nodes measure available bandwidth to node Y, as shown in Figure 3 (c). Node X switches to the new path with the largest available bandwidth, as shown in Figure 3 (d). In this description, the alternate path search process started from querying the DIN of the source node. We can also start with the DIN of the destination of the path, or the DIN of the bottleneck link.

B. Using Bottleneck Link Avoidance

This scheme involves locating the bottleneck link and bypassing that link for the new route, and is illustrated in Figure 4. Bottleneck links can be located by using tools such as multiQ [11], pathneck [7], or STAB [17]. As seen in Figure 4 (a), the path between nodes X and Y has a bottleneck link from router S to router T. As the bottleneck link is identified, node X finds the k closest nodes to the router S using a network proximity estimation tool such as Netvigator [20] or Meridian [22]. This process is illustrated in Figure 4 (b). We see from Figure 4 (c) that k (= 3) closest nodes to S are identified (black nodes). Node X measures available bandwidth to those k nodes, and the k nodes measure available bandwidth to node Y, as shown in Figure 4 (d). Node X selects the largest available bandwidth path. This new route is highlighted in Figure 4 (e) with the new overlay node (white node).

The two schemes described in Sections II-A and II-B are utilized to limit the number of candidate paths so that a new route can be found quickly without having to probe a large number of network nodes. However, if the above schemes do not return any satisfying candidate path, probing all possible alternate candidate nodes must be performed to find a new path that fulfills the bandwidth requirement.



(a) Default path with a bottleneck link.

(b) The use of Netvigator.

(c) Identification of candidate overlay nodes.

(d) Bandwidth measurement.

(e) New overlay route.

Fig. 4. Finding the new overlay route with the bottleneck link avoidance.

III. Measurement Results

We use PlanetLab [14] as our measurement testbed. Most bandwidth measurement tools require running on both the source and the receiver nodes. Since PlanetLab gives login access to all its machines, it is an attractive platform for our study. We selected 174 nodes from 174 sites. We did not use more than one node per site as the path between the nodes in the same site typically have large bandwidth and hence skew the results. Among 174 nodes we use, 93 are located in the Americas (North and South Americas), 66 are in Europe, and 15 are in Asia and Australia.

We use pathrate [3] for capacity and pathchirp [16] for available bandwidth measurements. We selected these tools as they are known to be one of the most accurate estimation tools and they work well under the current PlanetLab platform. We use the measurement data from the S^3 (Scalable Sensing Service) [18], [23] collected on June 15th, 2007. We performed evaluation with data from different time periods and obtained similar results. Hence we present our analysis with the recent data set. Although we ran pathrate for all pairs (174×173 = 30,102), we could only get and use 13,189 measurements. Pathrate could not be run on certain pairs when nodes are down or experiencing high loads. In addition, we filter out measurement runs that terminate with a high Coefficient of Variation (CoV value reported by pathrate) in the measured estimates. For the 13,189 pairs, the average capacity is 59.69 Mb/s and the average round-trip time is 113.88 msec. For a detailed bandwidth measurement study on PlanetLab, refer to [12].

A. Capacity Gain of Overlay Routes

Figure 5 shows the percentage of overlay routes with capacity larger than their respective default path capacity. We evaluate overlay paths of one-hop relay and two-hop relays. Let V be the set of nodes used in our measurement, x and y be the source and the destination for the end-to-end pairs, and r be the relay node between x and y. Let us define $C_{i,j}$ as the capacity from node i to j where $i, j \in V$, $i \neq j$, and $C_{i,j} > 0$. Note that if there was no capacity estimate from i to j, $C_{i,j} = 0$. We define the set of overlay paths with one hop relay as follows:

$$A_1 = \{(x, y, r) \in V^3 : C_{x,r} \cdot C_{r,y} \cdot C_{x,y} > 0\} \quad (2)$$

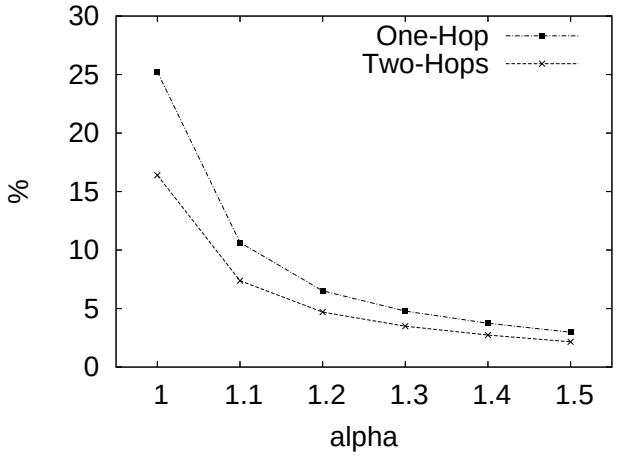


Fig. 5. Percentage of overlay routes with larger capacity than their native IP path.

where $x \neq y \neq r$. Now using the path switching threshold $\alpha(\geq 1)$, we define the set of overlay paths that satisfy Eq. (1):

$$A_1^* = \{(x, y, r) \in A_1 : \min(C_{x,r}, C_{r,y}) > \alpha \cdot C_{x,y}\}. \quad (3)$$

Similarly for overlay paths with two-hop relays, let r_1 and r_2 be the relay nodes between x and y, and we have:

$$A_2 = \{(x, y, r_1, r_2) \in V^4 : C_{x,r_1} \cdot C_{r_1,r_2} \cdot C_{r_2,y} \cdot C_{x,y} > 0\} \quad (4)$$

where $x \neq y \neq r_1 \neq r_2$, and also have:

$$A_2^* = \{(x, y, r_1, r_2) \in A_2 : \min(C_{x,r_1}, C_{r_1,r_2}, C_{r_2,y}) > \alpha \cdot C_{x,y}\}. \quad (5)$$

Figure 5 plots $\frac{|A_1^*|}{|A_1|} \times 100(\%)$ and $\frac{|A_2^*|}{|A_2|} \times 100(\%)$. Observe that more than 25% of the 658,526 one-hop relay overlay routes and more than 16% of 53,771,605 two-hop relay routes have larger capacity than the corresponding native IP path capacity. However, as we increase the value of α, the portion of overlay paths that satisfy the bandwidth gain requirement decreases. With α = 1.1, less than 11% of the paths gain capacity over the default IP path. Note however that as we keep increasing α, the drop of the fraction is not as steep.

Now we consider what fraction of end-to-end paths (source, destination pairs) has at least one overlay path with a non-zero capacity gain. We define:

$$B = \{(x, y) \in V^2 : C_{x,y} > 0\} \quad (6)$$

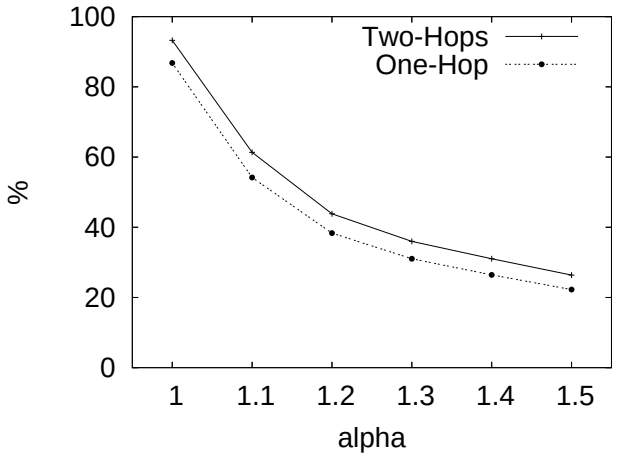


Fig. 6. Percentage of <source, destination> pairs that have overlay routes with larger capacity than the native IP paths.

and

$$B_1^* = \{(x,y) \in B : \exists r, (x,y,r) \in A_1^*\} \quad (7)$$

for one-hop relays and

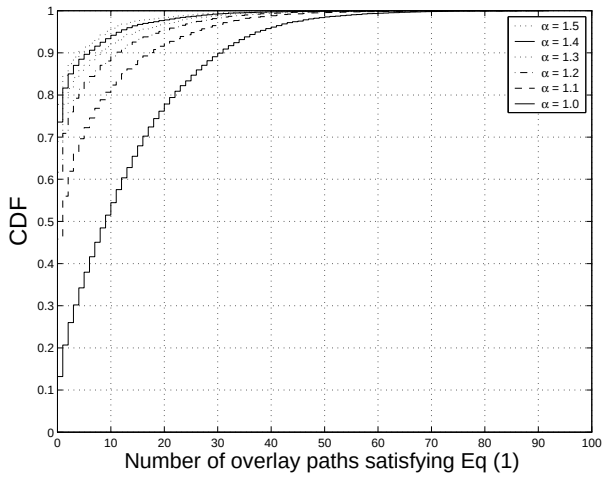
$$B_2^* = \{(x,y) \in B : \exists r_1, r_2, (x,y,r_1,r_2) \in A_2^*\} \quad (8)$$

for two-hop relay overlay paths.

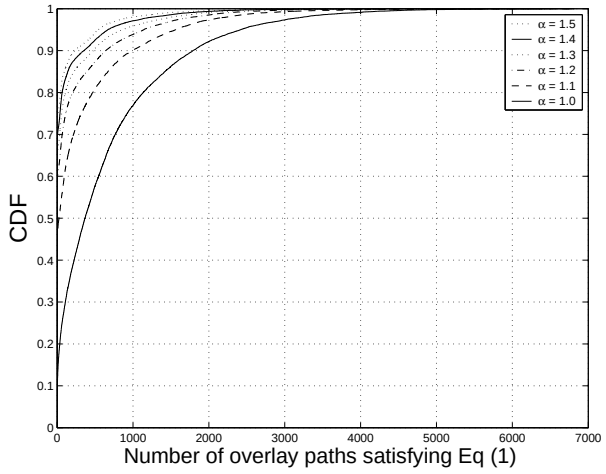
Figure 6 shows $\frac{|B_1^*|}{|B|} \times 100(\%)$ and $\frac{|B_2^*|}{|B|} \times 100(\%)$. More than 86% of pairs have one-hop relay overlay paths that provide larger capacity. For the two-hop case, more than 93% of the pairs can benefit from capacity gain by using overlay routes. Even with the increase of the α value to 1.2, nearly 40% of pairs have overlay paths that satisfy Eq. (1). It is interesting to note that although a larger portion of one-hop relay paths yield capacity gain from the default path, the <source, destination> pairs are more likely to have two-hop overlay paths that provide capacity increase. This stems from the fact that there are more two-hop relays than one-hop relays. When α = 1.2, there are 2,526,512 two-hop relay overlay paths that satisfy Eq. (1) while there are 42,931 such one-hop relays.

Figure 7 shows the cumulative distribution function of the number of one-hop and two-hop relay overlay paths that satisfy Eq. (1) for each ⟨source, destination⟩ pair, for various α values. The plots confirm the observation that there are more two-hop relays that provide bandwidth gain. It should also be noted that care must be taken when selecting the value of α. Having a larger value would minimize the route switching overhead and enable larger bandwidth gain. However, only a small number of alternate paths may be available with a large α. When α = 1.5, nearly 80% of ⟨source, destination⟩ pairs have no one-hop relay paths that satisfy the route switch requirement.

We now investigate the amount of absolute capacity gain that can be achieved by the overlay paths. Figure 8 shows the data with the varying value of α. "AVG" is the average bandwidth gain made by all overlay routes in A_1^* for one-hop and A_2^* for two-hop relay routes. "MAX_BW" and "MIN_RTT" denote the average bandwidth increase by using the maximum capacity overlay path and the minimum round-trip delay overlay path for each ⟨source, destination⟩ pair in B_1^* and B_2^*. We see that the capacity gain is over 35 Mb/s



(a) One-hop relays.



(b) Two-hop relays.

Fig. 7. CDF of the number of overlay routes that satisfy Eq. (1) for each <source, destination> pair.

for MAX_BW paths when α is 1.5. The increase of capacity in utilizing two-hop relays over one-hop is not significant. In fact, for AVG and MIN_RTT, one-hop relay paths have larger bandwidth than two-hop relays.

Overlay routes usually have more hops and higher latency than the native path. We plot the round-trip time increase of overlay paths in Figure 9. One-hop relay overlay paths on average have more than 80 msec delay increase compared against the default Internet routes, which could be unacceptable for some delay-sensitive applications. As the overlay relay hops increase to two, the delay increase on average is nearly 200 msec. Similar trend can be found for the MAX_BW paths where the delay increase of two-hop relays double that of one-hop relay paths. MIN_RTT paths on the other hand, show little latency increase. When α = 1.0, the MIN_RTT paths yield latency merely 6 msec larger than native IP paths. With the

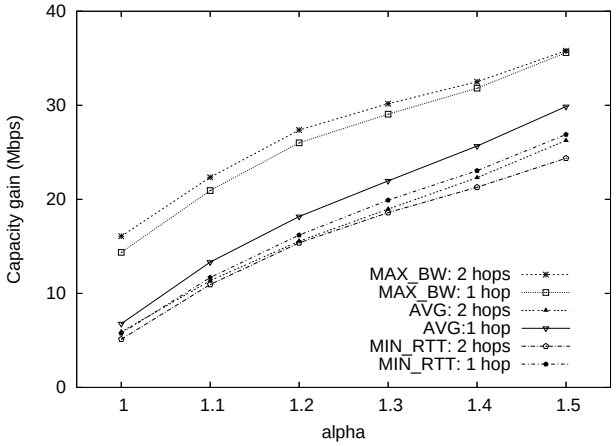


Fig. 8. Average capacity increase of overlay routes.

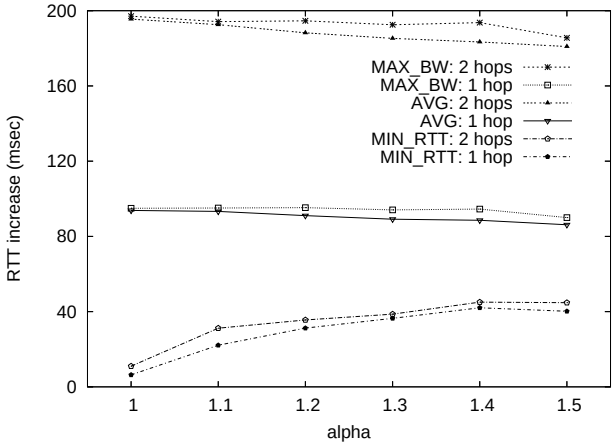


Fig. 9. Average RTT increase of overlay routes.

larger α values, MIN_RTT paths for both one-hop and two-hop relays show around 40 msec delay increase. With the capacity gain similar to the average paths as shown in Figure 8, and with little round-trip time increase, using MIN_RTT overlay paths may be a good compromise for the bandwidth and delay tradeoff.

For all 658,526 one-hop relay overlay paths in our measurement, we plot each overlay path's capacity gain and delay increase from its respective default IP path in Figure 10. An ideal overlay path would have a capacity increase and a delay decrease (i.e., reside within the highlighted box in the figure). Less than 3% of one-hop relay paths falls into the category however, and more than 60% of paths have smaller capacity and larger delay than the default path. For the MAX_BW and MIN_RTT paths, 9% and 15% are within the highlighted box respectively. We learn from this result that although there are many overlay paths available in the network, few paths provide advantage over the native IP path for both bandwidth and delay metrics. The number gets smaller for two-hop relays, as only 0.6% of all overlay paths are in this box; MAX_BW has 3%, and MIN_RTT has 10%.

For the rest of this section, we focus on the evaluation

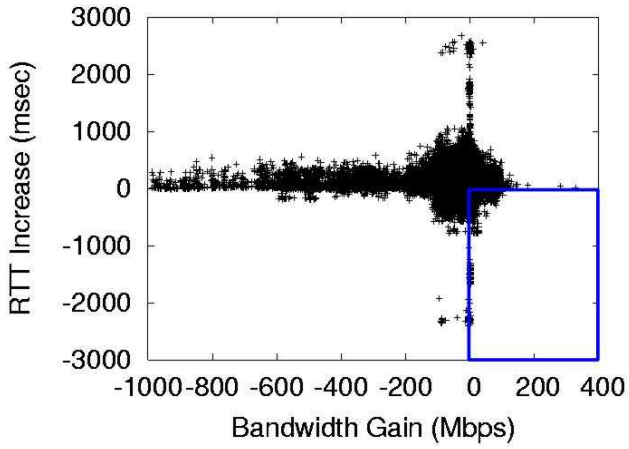


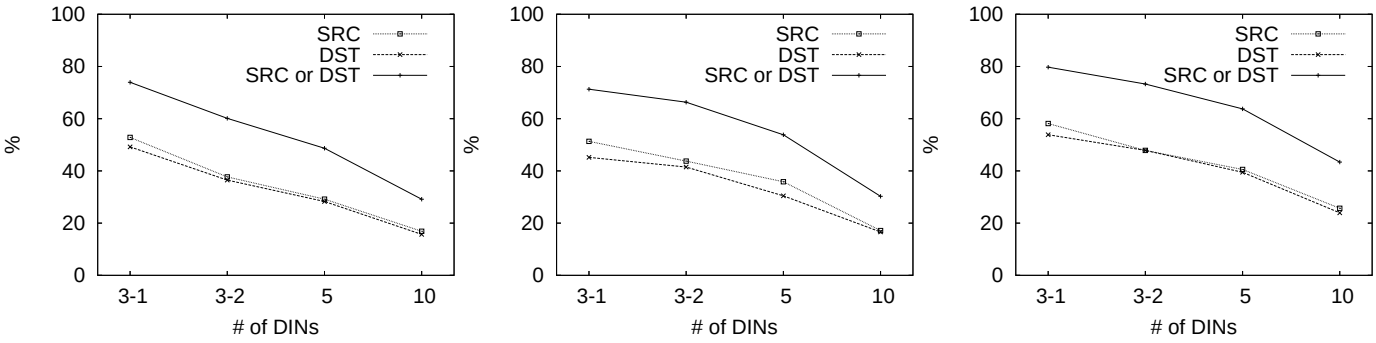
Fig. 10. Capacity gain and RTT increase of overlay routes.

of one-hop relay overlay paths as two-hop relays generate only a small capacity advantage over one-hop overlays while incurring much larger latency. Moreover, the processing of two-hop overlays involves larger overhead as they require more computation and there are larger number of paths.

B. Evaluation of the DIN Scheme

We evaluate the effectiveness of the DIN-based approach described in Section II-A. The objective of the DIN scheme is to quickly identify overlay nodes that provide large bandwidth capacity. Here we investigate which regions such nodes belong to. We vary the number of DINs from 3 to 10. The selected DINs and the number of nodes assigned to each DIN are shown in Table I. For the cases with three DINs, we have two different sets; one set (3-1) is geographically distributed by different continents while the other set (3-2) is selected based on the distribution of the nodes in each continent. Using closest partitioning, a node gets assigned to the region of a DIN whose latency to it is the smallest among the DINs. If a node does not have delay measurement to any of the DINs, it is left unassigned.

Figure 11 (a) shows the distribution of which region overlay nodes in A_1^* belong to. "SRC" denotes that the node is in the same region as the source of the path, "DST" the destination, and "SRC or DST" is the case when the overlay node is in the same region with the source or the destination. Note that there are instances where the source and the destination are in the same region. We observed similar trends with the varying α, and we only present results when α = 1.2. We see that overlay nodes that satisfy Eq. (1) do exist in the same region with the source or the destination, and the numbers are greater than the statistical average (33% for 3 DINs, 20% for 5 DINs, 10% for 10 DINs). Obviously, having less DINs and regions will increase the chance of finding the desired overlay nodes in the same region. However, when there are a small number of regions, each DIN will be overloaded. Even when there



(a) Region distribution of overlay nodes with larger capacity over native IP paths.

(b) Region distribution of the maximal capacity overlay node of each native IP path.

(c) Region distribution of the minimal RTT overlay node of each native IP path.

Fig. 11. Evaluation of the DIN scheme, α = 1.2.

TABLE I
DIN ASSIGNMENTS.

DINs	DIN nodes	# of nodes
3-1	planetlab1.cse.nd.edu	81
	edi.tkn.tu-berlin.de	68
	pub1-s.ane.cmc.osaka-u.ac.jp	11
3-2	planlab1.cs.caltech.edu	60
	planetlab1.cs.columbia.edu	50
	edi.tkn.tu-berlin.de	55
5	planlab1.cs.caltech.edu	34
	planetlab1.cse.nd.edu	38
	planetlab1.cs.columbia.edu	33
	edi.tkn.tu-berlin.de	53
	pub1-s.ane.cmc.osaka-u.ac.jp	10
10	planet1.scs.stanford.edu	12
	planlab1.cs.caltech.edu	17
	planetlab1.csres.utexas.edu	7
	planetlab1.cse.nd.edu	9
	planetlab1.cs.unc.edu	27
	planetlab1.cs.columbia.edu	24
	planetlab1.xeno.cl.cam.ac.uk	33
	edi.tkn.tu-berlin.de	30
	pub1-s.ane.cmc.osaka-u.ac.jp	6
	planetlab1.netmedia.gist.ac.kr	7

are the same number of regions, we see that the selection of DINs affects the performance greatly, as we observe from the difference between 3-1 and 3-2.

Figures 11 (b) and (c) show which region the "MAX_BW" and "MIN_RTT" nodes belong to, respectively. It is interesting that the maximal bandwidth nodes tend to be in the same region as the source. Hence, when a new path with a large capacity is needed, the source consulting its DIN to find the new route is a viable option. We also see that the percentage of finding the minimal delay node in the source or the destination regions is higher (more than 70% with 3 DINs and 60% with 5 DINs) compared with what we observed from Figure 11 (a).

C. Available Bandwidth with BARON

In this experiment, we evaluate the available bandwidth gain our BARON scheme brings. We measure the available bandwidth of the default path of the <source, destination> pair and k (= 5) candidate overlay paths that provide the most capacity gain.

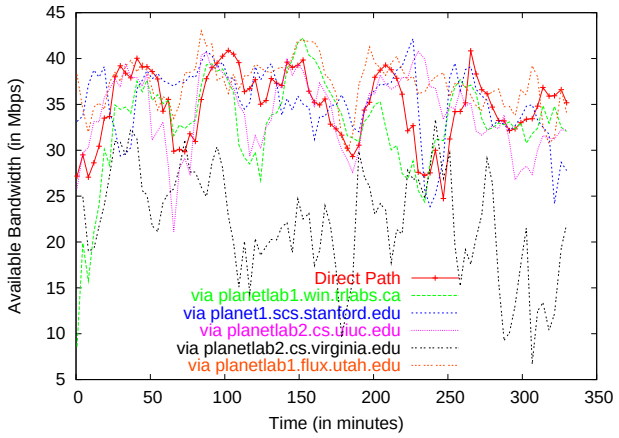
TABLE II
TRADEOFF WITH α VALUES.

α values	1.0	1.1	1.2	1.3	1.4	1.5
Overlay path usage (%)	77.37	77.2	64.39	40.35	36.14	16.32
Bandwidth gain (Mb/s)	3.4	2.91	2.18	1.55	1.08	0.55
Route switches	246	151	80	43	23	11

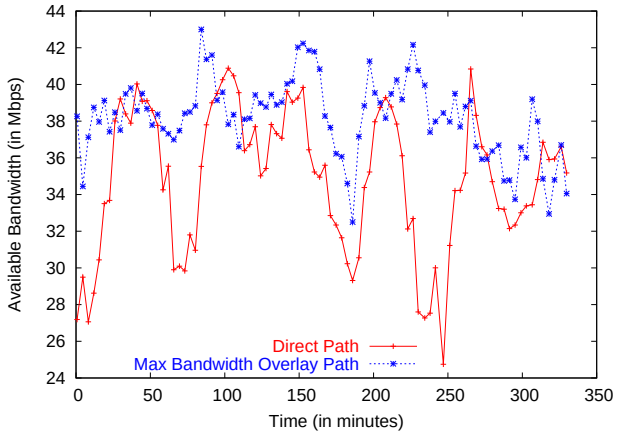
We performed the measurements for multiple sets of paths spanning different continents, but we present the ones that are the most representative and provide us with insights. Figure 12 (a) plots the available bandwidth of the native path from planetlab-1.cs.colostate.edu to planetlab1.cs.pitt.edu, and five overlay paths that provide the largest capacity. The measurement data for six hours is used so that we don't draw any conclusions from any possible temporary network problem. We see that available bandwidth fluctuates for all paths, and it is difficult to predict which path will provide the largest available bandwidth at a given time. In Figure 12 (b), we show the maximum available bandwidth among the five overlay paths for each measurement point and compare it with that of the default path. Most of the time, an overlay path provides available bandwidth gain over the direct path.

We now analyze the performance of BARON from the data set of the above path. Suppose we have a video conferencing application that requires 40 Mb/s of bandwidth. We initially use the default path and when its available bandwidth falls below the required bandwidth, we search for overlay paths that satisfy Eq. (1). If such overlay paths exist, we switch to the path that provides the maximum available bandwidth at that instance. This process corresponds to step 2 of our algorithm presented in Section II. When the new path fails to sustain the required bandwidth, the algorithm is executed again.

The results are provided in Table II. Here we use the data measured for 40 hours, although we used only six hours of data for Figure 12 for clarity. For different values of path switching threshold (α), we present the portion of time overlay routes are used, the average gain in available bandwidth over the default path, and the number of route switches performed. As

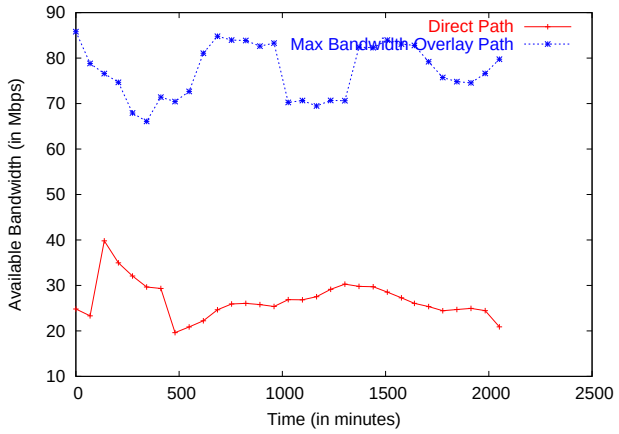


(a) Direct path and k (= 5) overlay paths.

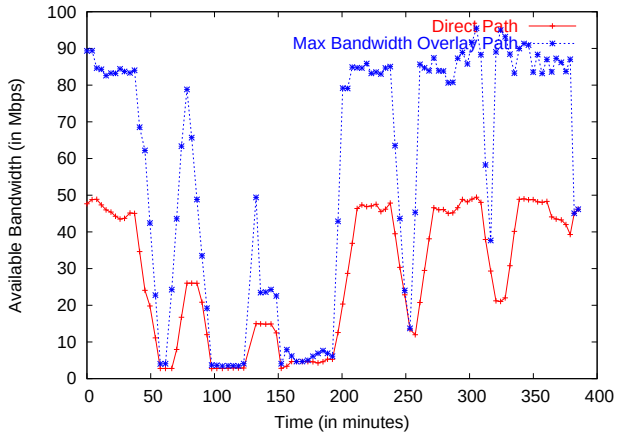


(b) Direct path and the maximum of overlay paths.

Fig. 12. Available bandwidth from planetlab-1.cs.colostate.edu to planetlab1.cs.pitt.edu.



(a) From planetlab1.sfc.wide.ad.jp to planetlab-1.cmcl.cs.cmu.edu.



(b) From planetlab1.cs.columbia.edu to planetlab1.netmedia.gist.ac.kr.

Fig. 13. Available bandwidth of other paths.

the α value increases, there are less number of route switches and overlay routes are used less. Although one may expect the bandwidth gain will grow with the increase of α, we observe a reverse trend. With a greater α value, we make less route switches although there may exist other paths with larger bandwidth. Hence, there is a tradeoff between bandwidth gain and route switching overhead.

We also present available bandwidth plots for different paths. Figure 13 (a) shows the case when overlay paths clearly give significant bandwidth gain over the default IP path at all times. An interesting case is observed in Figure 13 (b). Overlay paths generally provide larger available bandwidth than the native path. However, when there are sharp bandwidth drops by the default path, overlay paths also have similar decreases. When the bottleneck link is located near the source or the destination [7], most of, if not all, paths will have similar amount of bandwidth as the paths inevitably share the bottleneck link. Note that whether BARON makes the route switch depends on the required minimum bandwidth and the value of α. When other alternate paths do not provide bandwidth gain at the time of a new route search, no route change will be made. In the scenario in Figure 13 (b), if the minimum bandwidth required is 10 Mb/s, few route switches will be made as overlay paths do not give large gain when the default path's bandwidth is below 10 Mb/s. When the required bandwidth is larger, say 40 Mb/s, a switch to overlay paths will be made, and overlay paths will continue to be utilized as other paths, including the default path, cannot provide larger bandwidth when there are sharp decreases.

We have learned from our measurement study that by probing available bandwidth to a small number of large capacity paths, BARON provides overlay routes that sustain bandwidth advantage over the default path. We are in the process of evaluating other methods such as measuring available bandwidth to the paths that provide capacity gain but with least latency increase, and avoiding the bottleneck link.

IV. Related Work

Use of relay nodes to overcome the failures and performance issues of direct routing has been suggested in several papers [1], [4], [6], [13], [19]. Earlier work [1], [19] focused on leveraging the relay nodes to route around routing failures. The driving heuristic in most of these approaches [4], [13] is to increase the path diversity in the alternate paths. Bypassing the router failures only indirectly impacts the performance improvement and does not address the QoS requirements of the applications.

Cha *et al.* [2] propose algorithms for the placement of relay nodes to improve the path diversity. Controlled relay placement is not possible in certain conditions such as peer-to-peer networks. In such scenarios, it is important to efficiently select the best alternate path from a large set of options. To limit the computation and measurement overhead, Gummadi *et al.* [6] proposed selecting a random subset of nodes as k potential relay nodes. The best performing node of these k nodes is then used for relaying the data. Since the selection of initial nodes is random, some of the better relays might get ruled out. Another scalable mechanism to choose an alternate path in peer-to-peer networks is studied by Fei *et al.* [4]. Their approach is to reduce the overlap between the two paths by selecting the path diverging the earliest from the current path. It selects a path that is highly disjoint from the original path to avoid the link experiencing congestion, loss or failure.

None of the above mentioned approaches take the bandwidth on a path into consideration while choosing the alternate paths. A recent work [24] uses available bandwidth for dynamic overlay routing, and each node measures available bandwidth to a large number of nodes, which is not scalable. Moreover, available bandwidth fluctuates, and by the time new measurements are obtained, their values may have very little implication on the current available bandwidth, especially in large networks. Similarly, Jain *et al.* [8] use a link-state like protocol for distribution of bandwidth, loss, delay and other overlay link properties. Link state protocols do not scale as the size of the peer-to-peer network grows.

V. Conclusions

We presented Bandwidth-Aware Routing in Overlay Networks (BARON). Using PlanetLab measurements, we first analyzed the availability and characteristics of overlay paths. We showed that 25% of one-hop relay overlay paths have larger capacity than their respective default IP paths and over 86% of source-destination pairs have overlay routes with larger capacity than the respective native IP paths. With multiple overlay paths available for default IP paths, promptly finding the overlay path with bandwidth gain (and minimal latency increase) is a challenge.

We proposed scalable mechanisms to select alternate overlay paths to meet the bandwidth requirement of the applications, without incurring large measurement overhead. Our schemes use a combination of capacity and available bandwidth measurements to quickly identify high bandwidth routes. Our measurement-based evaluation showed that by utilizing the alternate overlay paths our scheme provides, sustained bandwidth gain over the default path are made.

References

[1] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of the ACM SOSP 2001*.

[2] M. Cha, S. Moon, C.-D. Park, and A. Shaikh. Placing relay nodes for intra-domain path diversity. In *Proceedings of the IEEE INFOCOM 2006*.

[3] C. Dovrolis, P. Ramanathan, and D. Moore. Packet-dispersion techniques and a capacity-estimation methodology. *IEEE/ACM Trans. Netw.*, 12(6):963–977, December 2004.

[4] T. Fei, S. Tao, L. Gao, and R. Guerin. How to select a good alternate path in large peer-to-peer systems? In *Proceedings of the IEEE INFOCOM 2006*.

[5] R. Fonseca, P. Sharma, S. Banerjee, S.-J. Lee, and S. Basu. Distributed querying of internet distance information. In *Proceedings of the IEEE Global Internet 2005*.

[6] K. P. Gummadi, H. Madhyastha, S. D. Gribble, H. M. Levy, and D. J. Wetherall. Improving the reliability of internet paths with one-hop source routing. In *Proceedings of the USENIX OSDI 2004*.

[7] N. Hu, L. Li, Z. M. Mao, P. Steenkiste, and J. Wang. Locating internet bottlenecks: Algorithms, measurements, and implications. In *Proceedings of the ACM SIGCOMM 2004*.

[8] M. Jain and C. Dovrolis. Path selection using available bandwidth estimation in overlay-based video streaming. In *Proceedings of the IFIP Networking 2007*.

[9] M. Jain and C. Dovrolis. Pathload: A measurement tool for end-to-end available bandwidth. In *Proceedings of the PAM 2002*, Fort Collins, CO, March 2002.

[10] R. Kapoor, L.-J. Chen, L. Lao, M. Gerla, and M. Y. Sanadidi. CapProbe: A simple and accurate capacity estimation technique. In *Proceedings of the ACM SIGCOMM 2004*, Portland, OR, August 2004.

[11] S. Katti, D. Kitabi, C. Blake, E. Kohler, and J. Strauss. MultiQ: Automated detection of multiple bottleneck capacities along a path. In *Proceedings of the ACM IMC 2004*.

[12] S.-J. Lee, P. Sharma, S. Banerjee, S. Basu, and R. Fonseca. Measuring bandwidth between PlanetLab nodes. In *Proceedings of the PAM 2005*.

[13] J. M. Opos, S. Ramabhadran, A. Terry, J. Pasquale, A. C. Snoeren, and A. Vahdat. A performance analysis of indirect routing. In *Proceedings of the IEEE IPDPS 2007*.

[14] PlanetLab, http://www.planet-lab.org.

[15] R. Prasad, C. Dovrolis, M. Murray, and kc claffy. Bandwidth estimation: Metrics, measurement techniques, and tools. *IEEE Netw.*, 17(6):27–35, Nov/Dec 2003.

[16] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell. pathChirp: Efficient available bandwidth estimation for network paths. In *Proceedings of the PAM 2003*, La Jolla, CA, April 2003.

[17] V. J. Ribeiro, R. H. Riedi, and R. G. Baraniuk. Locating available bandwidth bottlenecks. *IEEE Internet Comput.*, 8(6):34–41, September/October 2004.

[18] S^3: Scalable Sensing Service, http://networking.hpl.hp.com/s-cube.

[19] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The end-to-end effects of Internet path selection. In *Proceedings of the ACM SIGCOMM'99*.

[20] P. Sharma, Z. Xu, S. Banerjee, and S.-J. Lee. Estimating network proximity and latency. *ACM Computer Communications Review*, 36(3):41–50, July 2006.

[21] J. Strauss, D. Katabi, and F. Kaashoek. A measurement study of available bandwidth estimation tools. In *Proceedings of the ACM IMC 2003*, Miami, FL, October 2003.

[22] B. Wong, A. Slivkins, and E. G. Sirer. Meridian: A lightweight network location service without virtual coordinates. In *Proceedings of the ACM SIGCOMM 2005*.

[23] P. Yalagandula, P. Sharma, S. Banerjee, S.-J. Lee, and S. Basu. S^3: A Scalable Sensing Service for Monitoring Large Networked Systems. In *Proceedings of ACM SIGCOMM Workshop on Internet Network Management 2006*.

[24] Y. Zhu, C. Dovrolis, and M. Ammar. Proactive and reactive bandwidth-driven overlay routing: A simulation study. *Computer Networks*, 50(6):742–762, April 2006.