

Intersection Characteristics of End-to-End Internet Paths and Trees

Sevcan Bilir

Dept of Computer Science
University of Texas at Dallas
Richardson, TX 75083
sevcan@student.utdallas.edu

Kamil Sarac

Dept of Computer Science
University of Texas at Dallas
Richardson, TX 75083
ksarac@utdallas.edu

Turgay Korkmaz

Dept of Computer Science
University of Texas at San Antonio
San Antonio, TX 78240
korkmaz@cs.utsa.edu

Abstract

This paper focuses on understanding the scale and the distribution of “state overhead” (briefly load) that is incurred on the routers by various value-added network services, e.g., IP multicast and IP traceback. This understanding is essential to developing appropriate mechanisms and provisioning resources so that the Internet can support such value-added services in an efficient and scalable manner. We mainly consider the number of end-to-end paths or trees intersecting at a router to represent the amount of state overhead at that router. Hence, we analyze the router-level intersection characteristics of end-to-end Internet paths or trees to approximate the state overhead distribution in the Internet. For the reliability of our analysis, a representative, end-to-end router-level Internet map is essential. Although several maps are available, they are at best insufficient for our analysis. Therefore, in the first part of our work, we exert a measurement study to obtain a large size end-to-end router-level map conforming to our constraints. In the second part, we conduct various experiments using our map and shed some light on the scale and distribution of state overhead of value-added Internet services in both unicast and multicast environments.

1 Introduction

In this paper, we study the router-level intersection characteristics of the Internet paths and trees. We use these characteristics to better understand the scale and the place of “state overhead” that is incurred on the routers by value-added network services, e.g., IP multicast [4], IP traceback [24, 26, 32, 34], p2cast [23], SIFF [33, 35], IntServ [9], and Diffserv [7]. More specifically, we are seeking answers to various questions: “Does the overhead follow any known distribution?”, “How is the overhead distributed at the backbone?”, “Is there any relation between the overhead incurred on and the location (e.g., edge, border, backbone, etc.) of the routers in the network?”

Answering these questions is essential to developing appropriate mechanisms and provisioning resources so that the Internet can support aforementioned value-added services in an efficient and scalable manner. Using the observed characteristics in this study, we shed some light on

various issues related to the deployment, operation, management, and performance of value-added services. For example, how effective and scalable the existing state-reduction techniques are in reducing multicast forwarding state overhead in the routers? Where are the main choke points in terms of state accumulation for a given value-added service?

Starting from early 1990s, several value-added services have been proposed or introduced into the Internet. These include IP multicast [4]; packet marking and/or logging for IP traceback [24, 26, 32, 34]; recent proposals on receiver-controlled communication services such as p2cast [23] and SIFF [33, 35]; and IP-based QoS support such as IntServ [9] and Diffserv [7].

One common characteristic of these services is that they incur state and/or processing overhead that we briefly call “load” on the routers in the underlying paths or trees. For example, IP multicast, IntServ, and p2cast require to establish connections along the underlying end-to-end paths, resulting in state overhead on the routers in these paths. In this context, multiple simultaneous connections between the same end systems can be reduced to one by using end-to-end tunnels [13]. This simply allows us to use the number of end-to-end paths (or trees) crossing over a router as the state overhead (load) on that router. For DiffServ, we can consider the number of paths intersecting at the edge routers since DiffServ requires to maintain state information at the edge. From the foregoing discussion, we mainly consider the number of end-to-end paths intersecting at a router as the “load” on that router, and thus analyze it under various cases.

One of the key challenges in analyzing intersection characteristics of the Internet paths and trees is how to obtain a representative, end-to-end router-level Internet map. The research community has been extensively investigating various other characteristics of the Internet through measurements; thus, various maps have been collected. However, as we discuss later in detail, these maps were at best insufficient for our analysis since they were not either end-to-end, or router-level, or large enough. Therefore, one of the major tasks in our study was the collection and processing of the desired topology data as well as verifying the representativeness of the collected topology. At this end, we first conducted a relatively large-scale traceroute measurement among 153 end points located in North America and

obtained a router-level topology. We discuss the details of our measurement efforts and justify the representativeness of our topology in Section 3.

Using the topology map, we run various experiments to determine router-level intersection characteristics of end-to-end Internet paths (for unicast connections) and trees (for multicast connections). According to the experiments, the load distribution on routers follow a heavy tailed distribution where a small number of routers experience heavy load while a large number of routers experience lighter load. For unicast applications, the heavily loaded routers are backbone routers. In sparse mode multicast applications, most of the load accumulates at backbone routers. As multicast groups get denser, the overhead on exchange point routers reaches to that of backbone routers. Finally, our experiments show that the previously proposed approaches on multicast state reduction are not effective in reducing the number of forwarding states at heavily loaded branching routers, which, most of the time, correspond to border and exchange point routers. We present the details of our analysis in Sections 4 and 5.

In summary, the contributions of this paper are twofold: (1) the collection, processing and validation of a router-level topology map, and (2) experimental study on the intersection characteristics of end-to-end Internet paths and trees. Accordingly, after presenting related work in Section 2, we divide the paper into two parts. In the first part (Section 3), we describe our data collection, processing, and verification efforts. In the second part (Sections 4 and 5), we explain our experiments and results in the context of multicast and unicast scenarios, respectively. Finally, we conclude this paper and give directions for future research in Section 6.

2 Related Work

2.1 Router-Level Internet Measurements

There has been a large body of work related to Internet topology measurements. Earlier work examined routing and end-to-end path characteristics (including loss and jitter characteristics) of the Internet [11, 17, 18, 19, 21]. More recently, researchers have studied the connectivity characteristics of the Internet topology. One interesting recent finding was that the degree distribution of the nodes in the Internet follows a heavy tailed distribution. In their landmark work [25], Faloutsos et al. used Autonomous Systems (AS) and router level Internet topologies to show that power laws can be used to characterize the degree distribution of the nodes in the Internet. Later on, Broido and Claffy [10] used around 220M traceroute data (collected by the Skitter tool that we discuss in the next section) to construct a router-level Internet map and used that map to study the connectivity characteristics of the Internet. They showed that Weibull distribution can be used to approximate the outdegree distribution of the routers.

The observations in [25] have generated a significant debate on whether the node degree distribution can be modeled by power laws or not. During this debate, researchers questioned many aspects of the methodology that is currently used in Internet measurements studies: some pointed out the marginal utility of using additional vantage points in topology collection [6]; some discussed the difficulties

of inferring the topological attributes from the collected data [5, 36]; some pointed out the potential of sampling biases in topology collection [15]; and some questioned the validity of using degree distribution as the only (or the main) metric to characterize the Internet topology [16].

2.2 Value-added Services

IP Multicast

IP multicast [4] is one of the first value-added network services that is developed and partially deployed in the Internet. In IP multicast, source data propagates on a multicast distribution tree toward the receivers. Each router on a multicast tree maintains group specific forwarding state. As the number of multicast groups increase, the state overhead in the network increases. To reduce this overhead, researchers have proposed several state-reduction approaches that can be divided into two groups: (1) state aggregation and (2) tunneling approaches.

The main idea in state aggregation is to combine multiple multicast forwarding state entries into one single entry [29, 22]. Tunneling proposals, on the other hand, focus on reducing the number of multicast states by using unicast- or multicast-based tunnels [13, 8, 28, 30].

Although much work has been done on state-reduction techniques, little has been done on understanding the characteristics of multicast state overhead or evaluating the effectiveness of the proposed state-reduction techniques under realistic settings. In this end, Wong and Katz conducted an experimental study to analyze the nature of the multicast state scalability problem in [31]. By using several AS level Internet maps, a snapshot of Multicast Backbone (MBone), and a set of synthetic router level topologies, they provided a comprehensive analysis of state scalability problem and demonstrated the effects of (1) ISP peering relationships and (2) application and session characteristics on the distribution of multicast states on the routers. They also observed a power law relation between the growth of multicast states and the receiver size in the session. Finally, they studied the effect of non-branching state elimination on state scalability problem. They showed that non-branching state elimination helps reduce the state load considerably.

Unicast

In addition to IP multicast, several value-added network services have been proposed for unicast environments [7, 9, 23, 24, 26, 35, 33]. Researchers have informally discussed the scalability problems associated with these unicast value-added services. However, to the best of our knowledge, there is no work that experimentally evaluates the load incurred by value-added unicast services. In this end, our analysis in Section 5 seem to be the first.

3 Data Set

In this section we first describe our methodology in collecting and processing a new router-level Internet map, which is essential to our analysis. We then discuss the representativeness of our map in the light of the recently proposed sampling bias test [15]. We also offer some insight into the effectiveness of this test in finding sampling biases.

3.1 Data Collection

In order to study the router-level state and processing overhead distribution of value-added services, it is deemed

necessary to use a realistic Internet map that should have the following properties:

- Vantage points should be end points (or close to end points) in the Internet. This is so that the analysis represents the *end-to-end* load distribution characteristics.
- The map should include the path information among all vantage points as much as possible. Topology maps are often obtained based on traces from a single vantage point to a large number of subnet prefixes. Naturally, the resultant maps become a tree-oriented topology. Such a topology is not sufficient for our analysis because it excludes significant amount of path information (path traces) among all the end points (leaves of the underlying trees).
- The map should include path traces in both directions between two end points and should not use path symmetry assumption [20].
- The vantage points should be carefully selected so as to avoid any topological imbalance that may cause bias during our experiments. As an example, having a single vantage point in, say, Japan or Australia along with a large number of vantage points in North America may result in heavy load accumulation on the routers toward this remote vantage point. This may introduce potential bias for our experiment results, and, hence, should be avoided.

Having stated the properties that we want to have in our Internet map, we can now look at the **available Internet maps** and discuss why they are insufficient for our analysis. There have been several topologies collected and used in other measurement studies. For example, Pansiot and Grad collected end-to-end routes in order to construct representative multicast tree topologies [19]. Their data set includes 11 Internet-wide tree topologies which are collected by running traceroutes from these 11 vantage points to over 5000 subnets on the Internet. Due to its tree nature, this data set is not suitable for our analysis. Paxson collected end-to-end path information among 37 vantage points and used this data set to study end-to-end routing behavior in the Internet [20]. Paxson's data set is suitable for our study but is of limited size. Finally, Spring et al. used traceroute queries from 750 publicly available traceroute vantage points [27] most of which are not end points. These traces aim at discovering internal topologies of ISP networks and are not necessarily complete end-to-end traces. Therefore, they are of limited use for our purposes.

In addition to the above studies, there are three well-known collaborative efforts that provide Internet measurements support in large scale. They are Skitter [18] project of CAIDA, PlanetLab [3], and Active Measurement Project (AMP) [17] of NLNR. Skitter has 30 publicly available monitors that provide traffic measurements services for researchers. PlanetLab is a Internet-wide measurement test bed that has around 170 monitors at 70 different locations world wide at the time of our data collection (late 2004). AMP has 150 measurement monitors most of which are deployed in the United States. AMP monitors provide the infrastructure to take site-to-site measurements on high-speed research networks for monitoring and/or debugging purposes for the networking community.

Our Data Set

To obtain a representative end-to-end router-level Internet map having the aforementioned properties, we sig-

nificantly benefit from the resources of the above projects. Specifically, the vantage points that we use in our work include 120 measurements sites used by AMP project of NLNR and 33 traceroute servers that are listed at www.traceroute.org web site. We observed that most of our vantage points are located at universities or research institution in North America and most of them are connected to Internet2. We also observed a significant overlap between our vantage points and the active measurement sites of the PlanetLab.

While choosing the vantage points, we paid attention to choose the sites located at the periphery of the network. For this, we first used *ipas* tool [2] to map IP addresses of the vantage points to their AS numbers. Then, by consulting an AS level Internet map from [31], we classified these ASes as stub ASes and others. Most of the resulting candidate vantage points were located in North America and there were several others from Europe and Far East. Given the significant difference in the number of vantage points in North America and other parts of the world, we decided to use the vantage points located in North America only. At the end of this process, we were left with 153 vantage points that are located at stub ASes in North America. Finally, we used traceroute tool and collected end-to-end paths (153*152 traces) between all vantage points. After eliminating incomplete path traces and paths with loops, we had 19,739 path traces in our data set, based on which our topology is formed as discussed next.

3.2 Data Processing

After collecting the end-to-end paths, the next step is to process the data set to build an Internet map. This task involves two steps: (1) alias resolution for the routers that have multiple IP addresses and (2) resolving the identities of the unresponsive routers, i.e., routers causing traceroute program to print a "*" during the trace.

Alias Resolution

The first step in data processing is to resolve IP aliases of the routers. A router may respond to different traceroute queries with different interface IP addresses. This results in a situation where traceroute returns a list of interface IP addresses but does not group these interfaces into routers. Alias resolution refers to the process of checking if two (or more) given IP addresses belong to the same router.

Currently, there are two well-known IP alias resolution tools: *mercator* [14] and *ally* [1]. *Mercator* resolves aliases by using source IP addresses of ICMP PORT UNREACHABLE responses. *Mercator* sends an ICMP probe to each of the two IP addresses that are in question. If two probes with two different IP addresses result in ICMP responses with the same source IP address, then these two probed IP addresses are assumed to be aliases for the same router. *Ally* extends *mercator* by including a second step where it checks the IP identifier field values in the IP protocol header of the returning ICMP response messages. The intuition in this approach is that even if the ICMP responses for two alias probes have different source IP addresses, they can still belong to the same router if they have close IP identifier values. Given two IP addresses, *ally* tool returns three possible answers: "alias", "not alias", or "unknown" followed with an explanation. "Unknown" is returned when at least one of

the probes does not result in a response. This can happen as some ISPs configure their routers to ignore probes directed to themselves.

Since *ally* is an improvement over *mercator*, we used *ally* to resolve IP aliases in our data set. Using *ally*, we detected 1536 IP alias pairs corresponding to 435 unique routers. The maximum number of aliases that a router has in our data set is 23. On the other hand, 79% of IP addresses (around 5900 addresses) did not have any alias. At first look, this result suggests that 5900 IP addresses represent 5900 different routers in our data set. But, after carefully studying the output of *ally* probes, we noticed that *ally* probes to 3122 (out of 7073) IP addresses did not return any response. This suggests that some of the IP addresses among 5900 addresses may correspond to the same router. However, the current state-of-the-art techniques in alias resolution cannot help us find them.

Resolving Unresponsive Routers

The second step is to identify unresponsive routers causing traceroute to display ‘*’s during the trace. This task is important because more than half of the traces contain at least one ‘*’ corresponding to an unresponsive router. Routers that are configured not to respond TTL expiration event cause traceroute to display ‘*’ in its output. Note that the simplistic approach that assigns a unique IP address to each unresponsive router would not be suitable as some of these unresponsive routers may in fact be the same router.

To resolve unresponsive routers in different traces, we compare all path pairs (say p_1 and p_2) with unresponsive routers and give them the same IP address as follows:

- Suppose p_1 and p_2 contain one ‘*’ between two known routers. If the corresponding ‘*’ entries have the same upstream router and the same downstream router while both p_1 and p_2 have the same final destination, then we consider such unresponsive routers as the same router and assign a unique name, e.g., *ur.1*, to it. This case is illustrated in Figure 1(a). Since the A-to-D and B-to-

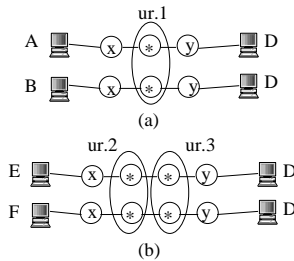


Figure 1. Resolving unresponsive routers.

D traces include an unresponsive router which has the same upstream router x , the same downstream router y , and the same destination D , we assign *ur.1* to it.

- Suppose p_1 and p_2 contains two consecutive ‘*’s between two known routers. Similar to previous procedure, we first cluster these routers and give the same name to routers in the same cluster if the cluster has the same upstream, the same downstream routers, and the same destination, as illustrated in Figure 1(b).
- Discard traces having more than two consecutive ‘*’s.

This way, we mapped 2748 unresponsive routers (i.e., 2748 occurrences of ‘*’s) to 406 distinct routers. Finally, we obtained our router-level map with 6,058 unique nodes, 13,873 links and 19,739 paths. Compared to topologies that have been collected by using (k,m) -traceroute queries ($k \ll m$), we expect our (n,n) -traceroute based topology to be more suitable for studying the end-to-end path intersection characteristics.

3.3 Representativeness of Our Data Set

Qualitatively speaking, it is clear that as the sample size increases, the collected data (our map) will be more and more representative for the sample space (the Internet). With this in mind, we conducted our measurement study. We believe that we maximally utilized the resources publicly available to us and obtained a large size end-to-end router-level map conforming to our constraints as outlined in Section 3.1.

Having said that, we now look at some quantitative evidence by comparing the similarities between the topological characteristics of our data set and that of those in other recent Internet topology measurement studies. We mainly consider the degree distribution characteristics from [25] and sampling bias issue from [15].

Power Law Conformance Test

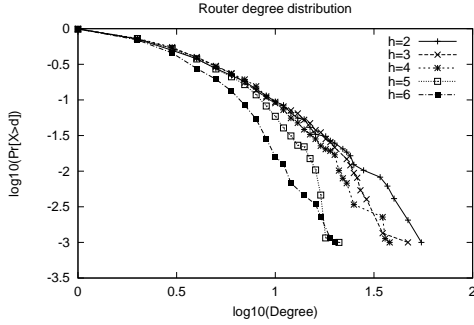
In [25], Faloutsos et al. showed that several characteristics of the Internet topology follow a power law distribution. In our work, we analyzed the degree distribution (power law 1) and degree rank distribution (power law 2) characteristics of our topology map. For the first power law, we found the rank exponent to be -0.48 and for the second power law we found the outdegree exponent to be -2.3. Both figures are consistent with the rank exponents suggested for the Internet.

Sampling Bias Test

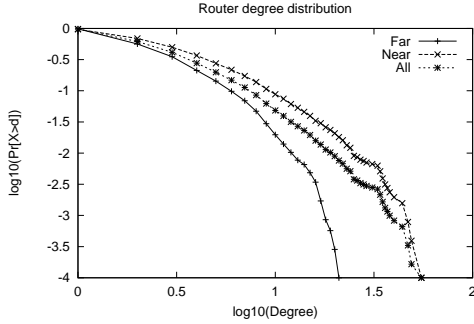
We also checked for potential sampling bias in our topology in the light of the previously mentioned work by Lakhina et al. [15]. The main idea in [15] is that the degree of a router should be independent of the distance (or hop count h) from the vantage point (i.e., traceroute source) to the router. Therefore, if the measurement procedure used in collecting a topology map is not biased, the statistical distribution of node degrees should not change with the distance h from the vantage point. To test for sampling bias, the paper first divides the set of routers V in the collected topology into two subsets, namely N (Near Set) and F (Far Set). While N includes the nodes closer to the vantage point than the median distance, F includes the nodes that are at least the median distance to the vantage point. The paper then concludes that there exist sampling biases if the following two tests are positive.

- C1: check if the 1% highest-degree vertices tend to appear mostly in N rather than in F .
- C2: check if the degree distribution of the routers in N (or F) is different than that of all routers in V .

According to these tests, our topology seems to present sampling bias. For example, we considered the degree distribution by hop distance from the vantage point(s). As we see in Figure 2-a, the closer the routers to vantage points,



(a) Degree distribution by hop count from vantage point(s).



(b) Degree distribution of nodes in *Near*, *All* and *Far* sets.

Figure 2. Degree distribution where h is the minimum distance to a vantage point.

the higher the degree distribution. In other words, routers with small h values have larger values on the y-axis when having the same value on the x-axis, especially for high degrees on the x-axis. This results in a confirmative answer to the first sampling bias test (C1), which questions if the highest-degree nodes tend to be near the vantage point(s) or not. For the second sampling bias test, we determined the degree distributions for N , F , and V . As seen in Figure 2-b, the routers in N (or F) have different distribution than those in V . Specifically, they can be ordered based on their sets as $Near Set > All > Far Set$, again suggesting that our topology shows sampling bias.

Actually these findings were somewhat surprising to us because our intuition suggests that a topology based on the (n,n) -traceroute approach should properly represent the degree distribution characteristics of the underlying network topology. We now present some explanations for the sampling bias suggested by C1 and C2 while also questioning the validity of such tests.

Comments on Sampling Biases

At this point, without knowing the underlying network topology, it is difficult to comment on the (lack of) representativeness of our topology. But at least, we can comment on the results of the sampling bias tests as follows. We claim that the differences between our expectations and the test results are mainly due to two reasons: (1) imperfect IP alias

resolution and (2) the procedure used to choose “hop count” (h) values for the routers during the sampling bias tests.

We suspect that the observed difference in the degree distribution of the routers in *Near Set* and in *Far Set* is partly due to imperfect IP alias resolution. As we mentioned previously, about 44% of the routers in our topology did not respond to *ally* queries which might have caused some of the aliases to remain undiscovered. The likely consequence of this limitation can be explained as below. Consider a router R (see Figure 3) that is in the *Near Set* with respect to a vantage point V_1 . Assume that R responds traceroute queries

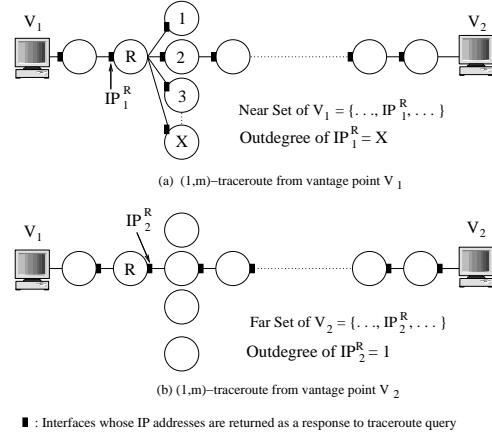


Figure 3. Effect of alias resolution on perceived sampling bias.

from V_1 with its IP address IP_1^R . Assume that the outdegree of R in the tree topology with respect to V_1 is $X > 1$. That is R is a major branching point in the tree topology rooted at V_1 and outdegree of IP_1^R is X . Now, assume that the same router R appears in the *Far Set* with respect to another tree topology rooted at a vantage point V_2 . Based on the observations in [12], the outdegree of R in this tree topology will likely be small, say it is 1, assuming that traceroute query from V_2 causes R to send a response with its IP address IP_2^R . At this point, we have R that appears in two different trees and in one it appears in *Near Set* with its address IP_1^R and with an outdegree of X . In the other tree, it appears in *Far Set* with its address IP_2^R and with an outdegree of 1. Now, if we cannot resolve the IP addresses for R and cannot detect that IP_1^R and IP_2^R in fact belong to the same router R , in our topology we will have two different nodes, one (IP_1^R) in the final *Near Set* with a high outdegree and the other (IP_2^R) in *Far Set* with a low outdegree (see Figure 3). But, in fact, these IP addresses belong to the same node and according to the procedure the node should only exist in the *Near Set* in the final topology. This discrepancy is not because of the potential bias in the sampling but rather because of the failure of the IP alias resolution. This observation also suggests that no matter how the topology sampling is done, due to the failure in IP alias resolution, the resulting topology will always look like it is biased. Hence, this deteriorates the validity of the sampling bias tests unless a perfect IP alias resolution can be achieved on the data.

In order to verify our claim, we generated a 10,000 node transit-stub network (1,000 nodes being transit nodes and 9,000 being stub nodes) using Georgia Tech Internet Topology Modeler (GT-ITM). We extracted a subtopology by collecting end-to-end shortest paths between 150 stub nodes from the main topology. This subtopology represents an (n, n) -traceroute based topology with no IP alias resolution problems. Using this subtopology, we looked at the degree distribution of the nodes in Near Set and Far Set. As we show in Figure 4, the degree distributions in Near Set has smaller CCDF values. As a result, this experiment suggests

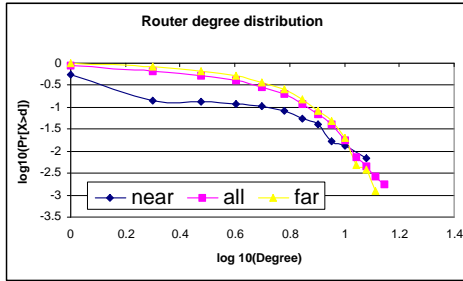
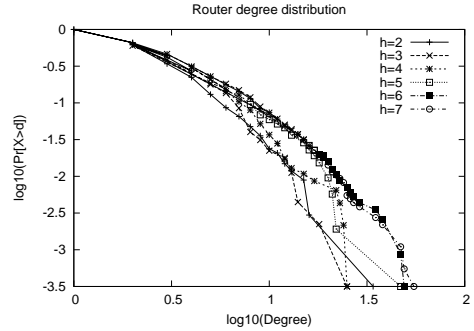


Figure 4. Degree distribution on GT-ITM data set.

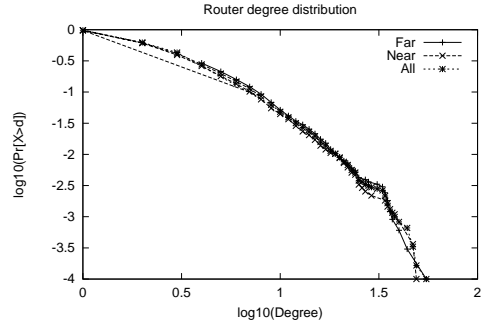
that Internet topologies that are collected by using (n, n) -traceroutes are likely to be free from sampling bias.

Our second observation is related to the way the hop count value h is computed for each router in the final topology. According to [15], if a router appears on topology trees of multiple vantage points, the h value is chosen to be the hop distance of the router to the closest vantage point. Since our topology consists of data generated by (n, n) -traceroutes, a significant majority of the routers appear on the topology trees collected by several different vantage points. As an example, (3, 5, 5, 5, 5, 5, 10, 11) represents the distances of a router in our topology to nine different vantage points. According to the above procedure, for this router we choose $h=3$ as the hop count to the vantage points and perform sampling bias tests. In this case, we feel that choosing h as the minimum distance to a vantage point may introduce bias within the methodology by itself and shows a router closer to the periphery of the network than its actual location. As an alternative approach, we tried the *median* hop count to set h ($h=5$ in the above example) and ran the tests. In this case, the distribution of the node degrees in Near and Far Sets is similar, as seen in Figure 5-b. In addition, Figure 5-a shows that the outdegree distribution of the routers is random at different hop counts, i.e., there is not a strong order in the form of $h=2 > h=3 > h=4 > \dots$

These observations may not necessarily give us a definitive answer on the existence or the lack of sampling biases in our topology. But at least they experimentally show that (n, n) -traceroute based topologies that do not have IP alias resolution problems are likely to be free from sampling bias. In addition, the discussion in this section points out the limitations of the procedure used for check-



(a) Degree distribution by hop count from vantage point(s).



(b) Degree distribution of nodes in Near, All and Far sets.

Figure 5. Degree distribution where h is the median distance to a vantage point.

ing sampling bias and helps us realize the importance of IP alias resolution in Internet measurement studies.

4 Load Distribution in Multicast Context

In this section, we present our analysis on multicast state scalability problem at the router level. We first investigate the effects of two important parameters, namely *usage rate* and *session density*, on multicast state distribution in the network¹. By considering scenarios with different usage rates and different session density values, we examine state distribution characteristics under various cases. We also examine state distribution at backbone and exchange point routers as they constitute potential scalability bottleneck points. Finally, we revisit the effectiveness of multicast state elimination approaches that focus on improving multicast state scalability. In this end, our analysis extends the previous work by Wong and Katz [31] who studied the problem mainly at the AS level.

4.1 Effect of Usage Rate and Session Density

In this set of experiments, we use different combinations of session density (trees with 2, 15, and 50 receivers) and usage rate (2, 5, 10, 15, and 50 trees) levels. In each experiment, we form multicast trees by choosing the sources and

¹Usage rate refers to the number of multicast groups in the network and session density refers to the number of receivers in a multicast group.

receivers according to usage rate and session density values respectively. Then, for each experiment, we count the number of states at backbone and exchange point routers. We ran several experiments for each session density and usage rate case. The results of the experiments are shown in Tables 1 and 2 as the average overhead for each experiment.

According to the first rows (2 receiver tree case) in Tables 1 and 2, at low session densities, the backbone routers have relatively more load than the exchange point routers especially at high usage rates. On the other hand, as session density increases, the load at exchange point routers get closer to the load at the backbone routers (see the third rows in Tables 1 and 2).

	Usage Rate (Num Trees)				
Session Density	2	5	10	15	50
2 Receiver Trees	0.36	1.27	2.28	3.18	10.32
15 Receiver Trees	1.27	2.90	6.68	8.90	27.86
50 Receiver Trees	1.64	3.82	8.41	11.90	35.73

Table 1. Average load at backbone routers (w.r.t. usage rate)

	Usage Rate (Num Trees)				
Session Density	2	5	10	15	50
2 Receiver Trees	0.33	1.00	0.92	1.83	6.58
15 Receiver Trees	1.00	1.83	4.34	5.83	15.00
50 Receiver Trees	1.50	3.33	7.33	11.16	32.75

Table 2. Average load at exchange point routers (w.r.t. usage rate)

We believe that this behavior is an expected behavior. That is, at low session densities, the first receiver will incur state on at most two exchange point routers. This happens when the path from receiver to sender crosses over the backbone. The additional receivers will then incur state on at most one exchange point router (assuming a single backbone domain). On the other hand, these receivers may incur state overhead on more than one backbone routers. Therefore, at low session densities, the backbone routers are likely to get more states than the exchange point routers. While we increase the session density, the probability that each exchange point leading toward a receiver (or a multicast sender) will increase, and, therefore, most of the exchange points will incur state overhead for many multicast trees. On the other hand, since it is a low probability for a backbone router to be on *all* end-to-end paths, the load on backbone routers will be limited. As we increase the usage rate and session density levels to 50%, the average load on exchange point routers exceed the average load on backbone routers (results not shown).

Another observation from the analysis is that multicast usage rate seems to be a more effective parameter for state scalability at backbone and exchange point routers. According to Table 1, as we increase the usage rate from 2 trees

to 50 trees at a session density of 2-receiver trees (the first row of the table), the average state overhead at backbone routers increases from 0.36 to 10.32. On the other hand, if we fix the usage rate at 2 trees and increase session density from 2 receiver trees to 50 receiver trees (the first column of the table), the average state overhead at backbone routers increases from 0.36 to 1.64. Since the increase in the first case is more, we conclude that usage rate is a more effective parameter for state scalability at backbone routers. A similar conclusion can be reached for exchange point routers in the same way by using Table 2.

4.2 Multicast State Elimination - Revisited

We now consider the effectiveness of multicast state elimination approaches. In general, researchers evaluate the effectiveness of state elimination approaches by looking at the number/ratio of non-branching states that are eliminated from the network. The work in [31] studies the state elimination on per-node resolution on AS level Internet maps and concludes that except for a negligible number of nodes, state elimination techniques are effective in reducing the number of states by removing the non-branching states from the nodes at the AS level.

In our work, we look at the effectiveness of state elimination approaches on a router level Internet map. For our evaluations, we consider several scenarios by choosing different number of multicast usage rates (i.e., 10, 25, and 50 trees) and session densities (i.e., 10, 25, and 50 receivers per tree). After constructing multicast trees, we count both the total number of states (both branching and non-branching states) and the number of branching states on each router in our network. Figure 6 presents the ratio of branching states on 50 most loaded routers (most loaded with respect to total number of states). According to the figure, we see that in each experimental scenario there are a number of routers whose branching ratio is significantly high. In other words, most of the states these routers are maintaining are branching states and non-branching state elimination techniques cannot help reduce the state overhead on these routers much. After a close look at the results, we observe that most of these routers are backbone or exchange point routers in our data set. This observation suggests that, contrary to previous conclusions, state elimination approaches are not necessarily effective in removing multicast states at bottleneck routers (e.g., backbone and exchange point routers). Since such bottleneck routers are potential performance choke points, eliminating non-branching states at other routers will not likely provide a practical solution to state scalability problem.

5 Load Distribution in Unicast Context

In this section, we study the worst-case load distribution characteristics of value added unicast services. Due to space limitation, average case analysis requiring to consider different load patterns is left as a future work.

For the worst-case analysis, we consider *all* unicast paths (19,739 paths) among end points in our data set, and count the number of end-to-end paths passing over a router as the load (i.e., state overhead) incurred on that router. Figure 7-a presents the worst-case load distribution in a decreasing order (i.e., rank distribution) in log-log scale. In addi-

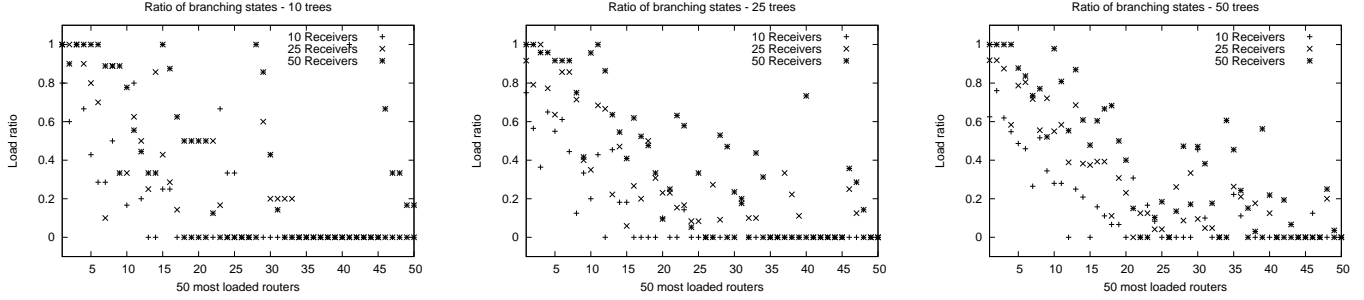
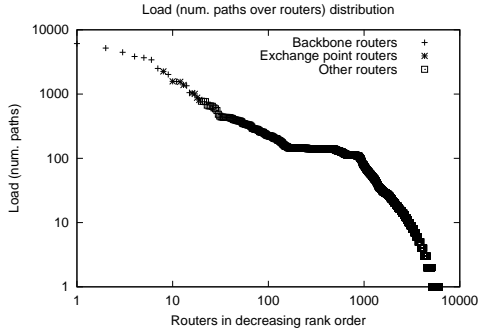
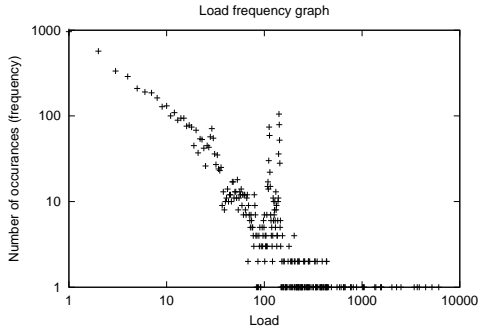


Figure 6. Nonbranching state ratio on routers.



(a) Load distribution.



(b) Load frequency distribution.

Figure 7. Worst-case load distribution.

tion, Figure 7-b presents the frequency distribution of the load. These figures show that a small number of routers have large numbers of paths passing over them and remaining significant majority of the routers appear on smaller number of paths. More specifically, there are 7 routers in the range [2503-6095], i.e., the router that is loaded highest appears on 6095 paths and 7th highest loaded router appears on 2503 paths. According to our data set, all of these routers belong to Abilene backbone which corresponds to a significant portion of the backbone network in our data set. Then, we list 10 routers in [2503-879] range and these routers are mostly exchange point routers. Finally, 6,000+ routers fall in the range [879-1].

In the second step of our analysis, we take a closer look at the load distribution on the routers as we go from the

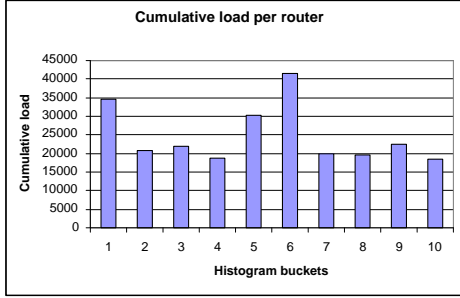
periphery of the network toward the core. We again consider all end-to-end paths corresponding to the worst case scenario. Due to the heterogeneity in path lengths in our data set and the difficulties in mapping routers to a precise location in one dimension, we calculate the load distribution after normalizing the path lengths as follows. First we create a histogram with 10 buckets². For each router on an end-to-end path, we use the location l of the router and the path length $pathlen$ to compute the ratio $l/pathlen$. Since a router may appear on multiple end-to-end paths, we use the *median* ($l/pathlen$) ratio to identify a bucket that this router maps to in our histogram as

$$bucket = round(\text{median}(\frac{l}{pathlen}) * 10). \quad (1)$$

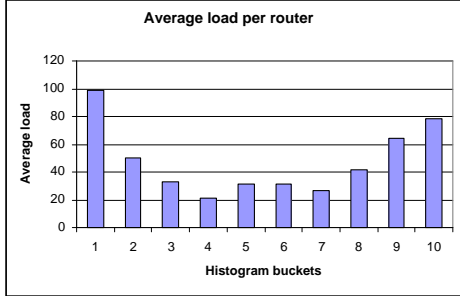
Then, we increment the load corresponding to this bucket by the amount of the load at this particular router and repeat the procedure for each router in our topology. At the end, the overall load is distributed in a one-dimensional space where the buckets at the two edges of the histogram correspond to the periphery of the network and the buckets at the center of the histogram correspond to the core of the network.

Figure 8-a shows the *cumulative* load distribution at each bucket. By ignoring information loss due to the normalization, this figure shows that most of the load is accumulated at the backbone (i.e., buckets around the center of the x-axis) at buckets 5 and 6. Then, the bucket corresponding to the edge of the network (i.e., buckets 1) bears high load. Figure 8-b shows the *average* load distribution at each bucket. Average load is defined as the ratio of the cumulative load on a bucket divided by the number of routers corresponding to that bucket. According to this figure, the average load at backbone routers is significantly less than that of edge routers. This result is surprisingly different than the cumulative load distribution. One possible explanation of this outcome is that not all routers that map to the buckets at the middle of the histogram are highly loaded routers. As we have seen in Figure 7, some of the highly loaded individual routers are backbone and exchange point routers. However, not all backbone routers in our data set are highly loaded. Please note that this may not necessarily reflect the

²We tried histograms with different number of buckets including 10-, 20-, and 30-bucket histograms and 10-bucket histogram seems to provide a good representation of the load distribution.



(a) Cumulative load distribution.



(b) Average load distribution.

Figure 8. Histogram based load distribution.

actual load on such routers in the Internet but rather indicates the load accumulated on those routers with respect to our experiments. We plan to further investigate the reasons for this outcome in our future work.

Next we check for a potential correlation between load and degree distribution. Figure 9 depicts the relation between the degree of the routers and their load. Let D be a random variable denoting the degree of routers and L be a random variable denoting the load on routers. The correlation of D and L is defined by

$$\rho_{L,D} = \frac{E[DL] - E[D]E[L]}{\sqrt{\text{VAR}(D)\text{VAR}(L)}}.$$

From this, we compute the correlation of D and L as 0.3, which indicates that there is a positive but not a significant correlation between the degree of a router and its load. This is also seen from the figure. Specifically, the highest degree for a router in our data set is 55. However, the load on that router is not as high as that on others. In addition, the degree of the router which has the highest load in the worst case is 40. And, the average degree for the *backbone* routers is 17. This observation makes perfect sense when we consider the hierarchical structure of the Internet topology. That is, in the Internet, it is the exchange point and border routers that have a large number of peers. Core routers, on the other hand, bear a large load (i.e., appears on a lot of end-to-end paths) but do not peer with a large number of other nodes. Note that previous studies that use AS level topology maps cannot reflect this observation.

In summary, in unicast environments, load accumulation at the core is significantly more than other parts of the

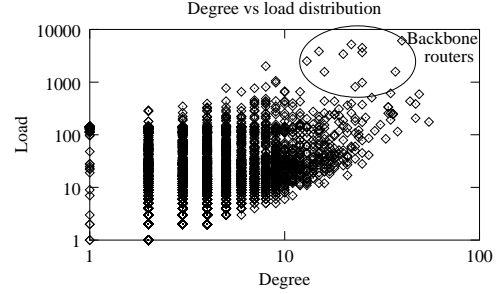


Figure 9. Degree vs load distribution.

network. Most of the highly loaded individual routers are backbone and exchange point routers. However, our observations on the average load distribution suggest that not all backbone routers are highly loaded in our data set. In addition, the results indicate that there is a positive but not so significant correlation between the degree of a router and its load.

6 Conclusions and Future Work

In this study, we first collected an end-to-end router-level Internet topology to study the scale and the distribution of state overhead on the routers. In contrast to (k, m) -traceroute approaches, we used (n, n) -traceroute approach and justified its representativeness. We then used this topology and analyzed the distribution of state overhead incurred by value-added services in both multicast and unicast environments. Specifically, we have shown that usage rate (i.e., number of trees) of multicast services is more important than session density in increasing the overall state overhead in the network. This suggests that tunneling mechanisms (e.g., aggregated multicast) that combines multiple multicast trees into one single tree is an effective approach in reducing the overall state overhead in the backbone. On the other hand, we have observed that such an approach is not always effective in reducing the state overhead at some of the heavily loaded border and exchange point routers. In the context of unicast services, we have shown that the backbone and the exchange point routers bear a heavy load. Therefore, it is deemed necessary to develop mechanisms that can reduce the state overhead not only at the core of the network (as done in DiffServ) but also at the exchange point routers.

Our work in this paper can be improved in several directions. One future work item is to collect and process a larger scale (n, n) -traceroute based Internet topology for studying the load distribution in the Internet. The recent DIMES project (www.netdimes.org) of Tel Aviv University uses (n, n) -traceroute approach to collect a very large scale Internet map. The topology collected by this project would be an excellent data set to apply and extend the work presented in this paper. Another future work item is related to the observation we made about the importance of IP alias resolution in constructing a representative topology from the collected traces. At this end, we plan to investigate IP alias resolution and identification of unresponsive routers to improve the representativeness of the collected

topologies. Finally, we plan to extend our study by considering various average-case load distribution characteristics under different traffic patterns.

References

- [1] <http://www.cs.washington.edu/research/networking/rocketfuel/>.
- [2] *IPAS Tool*. <http://mna.nlanr.net/Software/IPAS/docs/index.html>.
- [3] Planetlab. <http://www.planet-lab.org>.
- [4] K. Almeroth. The evolution of multicast: From the Mbone to inter-domain multicast to Internet2 deployment. *IEEE Network*, 14(1):10–20, January/February 2000.
- [5] L. Amini, A. Shaikh, and H. Schulzrinne. Issues with inferring internet topological attributes. In *Proceedings of SPIE ITCOM*, Boston, MA, USA, July/August 2002.
- [6] P. Barford, A. Bestavros, J. Byers, and M. Crovella. On the marginal utility of network topology measurements. In *ACM Internet Measurements Workshop*, San Francisco, CA, USA, November 2001.
- [7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. Informational RFC 2475, IETF, December 1998.
- [8] L. Blazevic and J. Boudec. Distributed core multicast (dcm): a multicast routing protocol for many groups with few receivers. In *Networked Group Communication Workshop*, Pisa, ITALY, November 1999.
- [9] R. Braden, D. Clark, and S. Shenker. Integrated services in the internet architecture: an overview. Internet Engineering Task Force (IETF), RFC 1633, June 1994.
- [10] A. Broido and k. claffy. Internet topology: Connectivity of IP graphs. In *Proceedings of SPIE ITCOM Conference*, Denver, CO, USA, August 2001.
- [11] R. Caceres, N. Duffield, H. J., and D. Towsley. Multicast-based inference of network-internal loss characteristics. *IEEE Transactions on Information Theory*, 45(7), November 1999.
- [12] R. Chalmers and K. Almeroth. On the topology of multicast trees. *IEEE/ACM Transactions on Networking*, 11(1):153–165, January 2003.
- [13] J. Cui, J. Kim, D. Maggiorini, K. Boussetta, and M. Gerla. Aggregated multicast — a comparative study. *The Journal of Networks, Software and Applications*, 2003.
- [14] R. Govindan and H. Tangmunarunkit. Heuristics for internet map discovery. In *IEEE INFOCOM*, Tel Aviv, ISRAEL, March 2000.
- [15] A. Lakhina, J. Byers, M. Crovella, and P. Xie. Sampling biases in IP topology measurements. In *IEEE INFOCOM*, San Francisco, CA, USA, March 2003.
- [16] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the internet’s router-level topology. In *Proceedings of ACM SIGCOMM*, Portland, OR, USA, August 2004.
- [17] A. McGregor, H.-W. Braun, and J. Brown. The NLANR network analysis infrastructure. *IEEE Communications Magazine*, 38(5):122–128, May 2000.
- [18] D. McRobb, K. Claffy, and T. Monk. *Skitter: CAIDA’s macroscopic Internet topology discovery and tracking tool*, 1999. Available from <http://www.caida.org/tools/skitter/>.
- [19] J. Pansiot and D. Grad. On routes and multicast trees in the Internet. *ACM Computer Communication Review*, 28(1), January 1998.
- [20] V. Paxson. End-to-end routing behavior in the Internet. In *ACM SIGCOMM*, Stanford, CA, USA, August 1996.
- [21] V. Paxson, J. Mahdavi, A. Adams, and M. Mathis. An architecture for large-scale internet measurement. *IEEE Communications*, August 1998.
- [22] P. Radoslavov, D. Estrin, and R. Govindan. Exploiting the bandwidth-memory tradeoff in multicast state aggregation. Technical report, University of Southern California, July 1999.
- [23] K. Sarac. SSM-based receiver-controlled communication in the Internet. In *Proceedings of South Central Information Security Symposium*, Denton, TX, USA, April 2003.
- [24] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical network support for IP traceback. In *Proceedings of the ACM SIGCOMM*, Stockholm, SWEDEN, August 2000.
- [25] G. Siganos, M. Faloutsos, P. Faloutsos, and C. Faloutsos. Power-laws and the as-level internet topology. *IEEE/ACM Transactions on Networking*, 11(4):514–524, August 2003.
- [26] A. Snoeren, C. Partridge, L. Sanchez, C. Jones, F. Tchakountio, B. Schwartz, S. Kent, and W. Strayer. Single-packet ip traceback. *IEEE/ACM Transactions on Networking*, 10(6), December 2002.
- [27] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring ISP topologies using rocketfuel. *IEEE/ACM Transactions on Networking*, 12(1):2–16, February 2004.
- [28] I. Stoica, T. Ng, and H. Zhang. Reunite: A recursive unicast approach to multicast. In *IEEE INFOCOM*, Tel Aviv, ISRAEL, March 2000.
- [29] D. Thaler and M. Handley. On the aggregatability of multicast forwarding state. In *IEEE INFOCOM*, Tel Aviv, ISRAEL, March 2000.
- [30] J. Tian and G. Neufeld. Forwarding state reduction for space mode multicast communication. In *IEEE INFOCOM*, San Francisco, CA, USA, March 1998.
- [31] T. Wong and R. Katz. An analysis of multicast forwarding state scalability. In *IEEE International Conference on Network Protocols*, Osaka, JAPAN, October 2000.
- [32] A. Yaar, A. Perrig, and D. Song. Pi: A path identification mechanism to defend against DDoS attacks. In *Proceedings of IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2003.
- [33] A. Yaar, A. Perrig, and D. Song. SIFF: A stateless Internet flow filter to mitigate DDoS flooding attacks. In *IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2004.
- [34] A. Yaar, A. Perrig, and D. Song. FIT: Fast internet traceback. In *IEEE INFOCOM*, Miami, FL, USA, March 2005.
- [35] X. Yang, D. Wetherall, and T. Anderson. A DoS-limiting network architecture. In *Proceedings of the ACM SIGCOMM*, Philadelphia, PA, USA, August 2005.
- [36] B. Yao, R. Viswanathan, F. Chang, and D. Waddington. Topology inference in the presence of anonymous routers. In *IEEE INFOCOM*, San Francisco, CA, USA, March 2003.