Measuring ISP Topologies With Rocketfuel

Neil Spring, Ratul Mahajan, David Wetherall, and Thomas Anderson

***Abstract*—To date, realistic ISP topologies have not been accessible to the research community, leaving work that depends on topology on an uncertain footing. In this paper, we present new Internet mapping techniques that have enabled us to measure router-level ISP topologies. Our techniques reduce the number of required traces compared to a brute-force, all-to-all approach by three orders of magnitude without a significant loss in accuracy. They include the use of BGP routing tables to focus the measurements, the elimination of redundant measurements by exploiting properties of IP routing, better alias resolution, and the use of DNS to divide each map into POPs and backbone. We collect maps from ten diverse ISPs using our techniques, and find that our maps are substantially more complete than those of earlier Internet mapping efforts. We also report on properties of these maps, including the size of POPs, distribution of router outdegree, and the interdomain peering structure. As part of this work, we release our maps to the community.**

***Index Terms*—Communication system operations and management, Internet, measurement, network reliability.**

I. Introduction

REALISTIC Internet topologies are of considerable importance to network researchers. Topology influences the dynamics of routing protocols [3], [11], the scalability of multicast [19], the efficacy of denial-of-service tracing and response [12], [18], [23], [24], and protocol design [20].

Sadly, real topologies are not publicly available, because ISPs generally regard their router-level topologies as confidential. Some ISPs publish simplified topologies on the Web, but these lack router-level connectivity and POP structure and may be optimistic or out of date. There is enough uncertainty in the properties of real ISP topologies (such as whether router outdegree distribution follows a power law as suggested by Faloutsos [8]) that it is unclear whether synthetic topologies generated by tools such as GT-ITM [28] or Brite [14] are representative [27].

The main contribution of this paper is to present new measurement techniques to infer high-quality ISP maps while using as few measurements as possible. Our insight is that routing information can be exploited to select the measurements that are most valuable. One technique, *directed probing*, uses BGP routing information to choose only those traceroutes that are likely to transit the ISP being mapped. A second set of techniques, *path reductions*, suppress traceroutes that are likely to yield paths through the ISP network that have been already been traversed. These two techniques reduce the number of traces required to map an ISP by three orders of magnitude compared to a brute-force, all-to-all approach, without sacrificing accuracy. We also describe a new solution to the *alias resolution* problem of clustering the interface IP addresses listed in a traceroute into routers. Our new, pair-wise alias resolution procedure finds three times as many aliases as prior techniques. Additionally, we use DNS information to break the ISP maps into backbone and POP components, complete with their geographical location.

We used our techniques to map ten diverse ISPs—Abovenet, AT&T, Ebone, Exodus, Level 3, Sprint, Telstra, Tiscali (Europe), Verio, and VSNL (India)—using over 750 publicly available traceroute sources as measurement vantage points. We summarize these maps in the paper.

Three ISPs of the ten we measured helped to validate our maps. We also estimate the completeness of our maps by scanning ISP IP address ranges for routers that we might have missed and by comparing the peering links we find with those in BGP routing tables. Our maps reveal more complete ISP topologies than earlier efforts; we find roughly seven times more routers and links in our area of focus than a recent Skitter [7] dataset.

As a second contribution, we examine properties that are of interest to researchers and likely to be useful for generating synthetic Internet maps. We characterize the distributions of router and POP outdegree, and report new results for the distribution of POP sizes and the number of connections an ISP has with with other networks. All these distributions have significant tails.

Finally, as one goal of our work and part of our ongoing validation effort, we have publicly released the ISP network maps inferred from our measurements. The entire raw measurement data is available to researchers; all our maps are constructed with end-to-end measurements and without the benefit of confidential information. The maps and data are available online [22].

The rest of this paper is organized as follows. In Sections II and III, respectively, we describe our approach and the mapping techniques. The implementation of our mapping engine, Rocketfuel, is described in Section IV. We present sample ISP maps and characterize their properties in Section V. In Section VI, we evaluate our maps for completeness and our techniques for their measurement efficiency and accuracy. We present related work in Section VII, and conclude in Section VIII.

II. Problem and Approach

The goal of our work is to obtain realistic router-level maps of ISP networks. In this section, we describe what we mean by an ISP map and the key measurement challenges we face.

An ISP network is composed of multiple points of presence or POPs, as shown in Fig. 1. Each POP is a physical location where the ISP houses a collection of routers. The ISP *backbone* connects these POPs, and the routers attached to inter-POP links are called *backbone* or *core* routers. Within every POP, *access* routers provide an intermediate layer between the ISP backbone

Manuscript received November 13, 2002; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor J. Rexford. This work was supported by the Defense Advanced Research Projects Agency under Grant F30602-00-2-0565.

The authors are with the Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195-2350 USA (e-mail: nspring@cs.washington.edu; ratul@cs.washington.edu; djw@cs.washington.edu; tom@cs.washington.edu).

Digital Object Identifier 10.1109/TNET.2003.822655

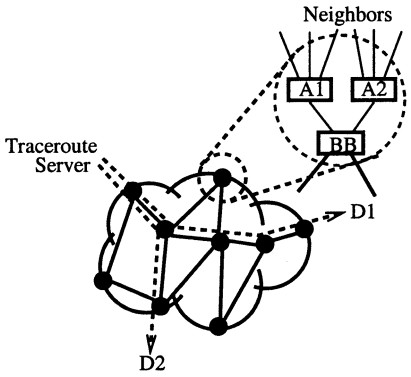

Fig. 1. ISP networks are composed of POPs and backbones. Solid dots inside the cloud represent POPs. A POP consists of backbone and access routers (inset). Each traceroute across the ISP discovers the router-level path from the source to the destination.

and routers in neighboring networks. These neighbor routers include both BGP speakers and non-BGP speakers, with most of them being non-BGP-speaking small organizations.

Our aim is to discover *ISP maps* that consist of backbone, access, and directly connected neighboring domain routers and the IP-level interconnections between them. This constitutes the interior routing region of the ISP and its boundary "peering links." ISPs are usually associated with their BGP autonomous system numbers (ASNs). The map we collect does not precisely correspond to the IP address space advertised by an AS. In particular, ISPs typically advertise the address space of non-BGP speaking customers as their own; our maps exclude such neighboring networks, consumer broadband, and dialup access networks. In the paper, we use ISP names and their AS numbers interchangeably.

Like earlier Internet mapping efforts [5], [7], [9], we discover ISP maps using traceroutes.[1] This process is illustrated in Fig. 1. Each traceroute yields the path through the network traversed from the traceroute source to the destination. Traceroute paths from multiple sources to multiple destinations are merged to form an ISP map. We use publicly available traceroute servers as sources. Each traceroute server provides one or more *vantage points*: unique traceroute sources that may be routers within the AS or the traceroute server itself.

The key challenge is to build accurate ISP maps using few measurements. We cannot burden public traceroute servers with excessive load, limiting the traceroutes we can collect from each server. A brute-force approach to Internet mapping would collect traceroutes from every vantage point to each of the 120 000 allocated prefixes in the BGP table. If public traceroute servers are queried at most once every 1.5 min,[2] this approach will take at least 125 days to complete a map, a period over which the Internet could undergo significant topological changes. Another brute-force approach is to traceroute to all IP addresses owned by the ISP. Even this approach is not feasible because ISP address space can include millions of addresses, for example, AT&T's 12.0.0.0/8 alone has more than 16 million addresses.

[1]Using traceroute has inherent well-understood limitations in studying network topology. For example, traceroute does not see unused backup links in a network, it does not expose link-layer redundancy or dependency (multiple IP links over the same fiber), and it does not discover multi-access links.

[2]This limit was provided by the administrator of one traceroute server, but is still aggressive. Traceroutes to unresponsive destinations may take much longer.

```
1.2.3.0/24          13 4 2 5
                    6 9 10 5
                    11 7 5

4.5.0.0/16          3 7 8
                    7 8
```

Fig. 2. Sample BGP table snippet. Destination prefixes are on the left, AS-paths on the right. ASes closer to the destination are to the right of the path.

Our design philosophy is to choose traceroutes that will contribute the most information to the map and omit those that are likely to be redundant. Our insight is that expected routing paths provide a valuable means to guide this selection. This trades accuracy for efficiency, though we will see that the loss of accuracy is much smaller than the gain in efficiency.

After connectivity information has been obtained through traceroutes, two difficulties remain. First, each traceroute is a list of IP addresses that represent router interfaces. For an accurate map, the IP addresses that belong to the same router, called *aliases*, must be resolved. When we started to construct maps, we found that prior techniques for alias resolution were ineffective at resolving obvious aliases. In response, we developed a new, pair-wise test for aliases that uses router identification hints such as the IP identifier, rate limiting, and TTL values.

Second, to analyze the structural properties of the collected maps, we need to identify the geographical location of each router and its role in the topology. Following the success of recent geographical mapping work [16], we leverage location hints that are typically embedded in DNS names to extract the backbone and the POPs from the ISP map.

III. Mapping Techniques

In this section, we present our mapping techniques, divided into three categories: selecting measurements, resolving aliases, and categorizing the role and location of ISP routers.

A. Selecting Measurements

We use two classes of techniques to reduce the required number of measurements. First, we select only traceroutes that we expect will transit the ISP. We use a technique called *directed probing* that interprets BGP tables to identify relevant traceroutes and prune the remainder. Second, we are interested only in the part of the traceroute that transits the ISP. Therefore, only one traceroute must be taken when two traceroutes enter and leave the ISP network at the same points. We use techniques called *path reductions* to identify redundant traceroutes.

1) Directed Probing: Directed probing aims to identify traceroutes that will transit the ISP network. Ideally, if we had the BGP routing table corresponding to each vantage point, we would know the paths that would transit the ISP being mapped. Since these tables are not available, we use RouteViews [15] as an approximation. It provides BGP views from 60 different points around the Internet.

A BGP table maps destination IP address prefixes to a set of AS-paths that can be used to reach that destination. Each AS-path represents the list of ASes that will be traversed to reach the prefix. We now show how to identify three classes of traceroutes that should transit the ISP network. In this example, we use the BGP table snippet in Fig. 2 to map AS number 7.

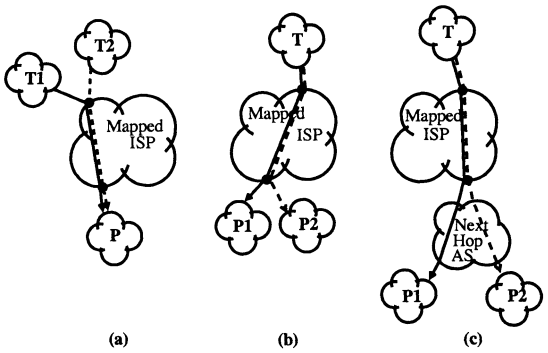

Fig. 3. Path reductions. (a) Only one traceroute needs to be taken per destination when two servers (Ts) share an ingress. (b) Only one trace needs to be taken when two dependent prefixes (Ps) share an egress router. (c) Only one trace needs to be taken if two prefixes have the same next-hop AS number.

- Traceroutes to *dependent prefixes*: We call prefixes originated by the ISP or one of its singly-homed customers *dependent prefixes*. All traceroutes to these prefixes from any vantage point should transit the ISP. Dependent prefixes can be readily identified from the BGP table: all AS-paths for the prefix would contain the number of the AS being mapped. 4.5.0.0/16 is a dependent prefix of AS 7.
- Traceroutes from *insiders*: We call a traceroute server located in a dependent prefix an insider. Traceroutes from insiders to any prefix should transit the ISP.
- Traceroutes that are likely to transit the ISP based on some AS-path are called *up/down traces*. In Fig. 2, a traceroute from a server in AS 11 to 1.2.3.0/24 is an up/down trace when mapping AS 7.

Directed probing uses routing information to skip unnecessary traceroutes. However, incomplete information in BGP tables, dynamic routing changes, and multiple possible paths lead to two kinds of errors. Executed traceroutes that do not traverse the ISP (false positives) sacrifice speed, but not accuracy. Traceroutes that transit the ISP network, but are skipped because our limited BGP data did not include the true path (false negatives), may represent a loss in accuracy, which is the price we pay for speed. Traceroutes that were not chosen may traverse the same set of links seen by chosen traceroutes, so false negatives may not always compromise accuracy. In Section VI-B1, we estimate the level of both these types of errors.

2) Path Reductions: Not all traceroute probes chosen by directed probing will take unique paths inside the ISP. The required measurements can be reduced further by identifying probes that are likely to have identical paths inside the ISP. We examine where previous traces enter and exit the ISP network to predict whether a future trace will take a new path. A fundamental assumption is that the path from entry to exit is consistent. We list three techniques based on properties of IP routing to establish entry and exit points.

Ingress Reduction. When traceroutes from two different vantage points to the same destination enter the ISP at the same point, the path through the ISP is likely to be the same. This is illustrated in Fig. 3(a). Since the traceroute from T2 to the destination would be redundant with the traceroute from T1, only one is needed. The observation is that traceroutes from a server frequently enter the ISP at only one router—other traceroute servers that enter the ISP using the same router are equivalent.

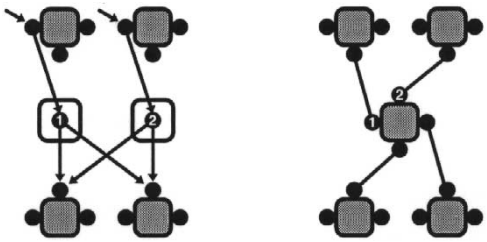

Fig. 4. Alias resolution. Boxes represent routers and circles represent interfaces. Traceroute lists input interface addresses from paths (left). Alias resolution clusters interfaces into routers to reveal the true topology. Interfaces ① and ② are aliases (right).

Egress Reduction. Conversely, if two destination prefixes are reached using the same egress router, they are equivalent: only one trace needs to be collected. This is illustrated in Fig. 3(b). Dependent prefixes are bound to egress routers in the egress discovery process described in Section IV. This prefix-to-egress-router binding would be invalid for dependent prefixes originated by the ISP that connect in multiple locations. We expect that such prefixes are few and that other prefixes are also connected to the same egress routers.

Next-hop AS Reduction. When reaching prefixes outside the ISP, the path usually depends only on the next-hop AS, and not on the specific destination prefix. Prefixes reached through the same next-hop AS are thus equivalent, as shown in Fig. 3(c). Next-hop AS and egress reductions are similar in that they apply to the end of the path through the ISP. However, they are distinct in that there may be several peering points to the next-hop AS, while we expect only one egress router for ISP prefixes. Next-hop AS reduction applies to insider and up-down traces, while egress reduction applies to traces to dependent prefixes.

Path reductions predict likely duplicates so that more valuable traces can be taken instead without sacrificing fidelity. If the prediction is false (an unexpected ingress or egress was taken), we repeat the trace using other servers.

B. Alias Resolution

Traceroute lists the source addresses of the "Time exceeded" ICMP messages; these addresses represent the link interfaces on the routers that received traceroute probe packets. A significant problem in recovering a network map from traceroutes is alias resolution, or determining which interface IP addresses belong to the same router. The problem is illustrated in Fig. 4. If the different addresses that represent the same router cannot be resolved, a different topology with more routers and links results.

The standard technique for alias resolution was introduced by Pansiot and Grad [17] and refined in the Mercator project [9]. It detects aliases by sending traceroute-like probes (to a high-numbered UDP port but with a TTL of 255) directly to the potentially aliased IP address. It relies on routers being configured to send the "UDP port unreachable" response with the address of the outgoing interface as the source address: two aliases will respond with the same source. This technique is efficient in that it requires only one message to each IP address, but we found that it missed many aliases, at least for the ISPs we studied.

Our approach to alias resolution combines several techniques that identify peculiar similarities between responses to packets sent to different IP addresses. These techniques try to collect evidence that the IP addresses are on the same router by looking for

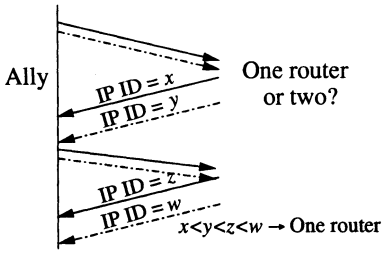

Fig. 5. Alias resolution using IP identifiers. A solid arrow represents messages to and from one IP, the dotted arrow the other.

features that are centrally applied. We look primarily for nearby IP identifiers, a counter that is stamped on responses by the host processor. The IP identifier is intended to help in uniquely identifying a packet for reassembly after fragmentation. As such, it is commonly implemented using a counter that is incremented after generating a packet. This implies that packets sent consecutively will have consecutive IP identifiers.[3] We also look for a common source IP address in responses, as in Mercator. A third feature is ICMP rate limiting, where the router's host processor responds only to the first of back-to-back probes.[4] A fourth feature that is not sufficient on its own is the TTL remaining in the response. The TTL may start at different values depending on the router operating system, and responses from routers in different locations are likely to traverse paths of different length back through the network. This makes the TTL useful for providing evidence that two addresses are not aliases, but the range of possible values is too small to show that addresses are aliases.

The procedure for resolving aliases by IP identifier is shown in Fig. 5. Our tool for alias resolution, Ally, sends a probe packet similar to Mercator's to the two potential aliases. The port unreachable responses include the IP identifiers x and y. Ally then sends a third and fourth packet to the potential aliases to collect identifiers z and w. If $x < y < z < w$, and $w - x$ is small, the addresses are likely aliases. In practice, some tolerance is allowed for reordering in the network. As an optimization, if $|x - y| > 200$, the aliases are disqualified and the third and fourth packets are not sent. In-order IP identifiers suggest a single counter, which implies that the addresses are likely aliases. The results presented in this paper were generated using a three-packet technique, without the w packet, but we believe the fourth packet should further reduce the false positive rate. We observed that different routers change their IP identifiers at different rates: the four-packet test establishes that the potentially two counters have similar value and rate of change, while the earlier three-packet test only demonstrated similar value.

Some routers are configured to rate-limit port unreachable messages. If only the first probe packet solicits a response, the probe destinations are reordered and two probes are sent again after five seconds. If, again, only the first probe packet solicits a response, this time to the packet for the other address, the rate-limiting heuristic detects a match. When two addresses appear to be rate-limited aliases, the IP identifier technique also detects a match when the identifiers differ by less than 1000.

[3]We have not observed routers that use random identifiers or implement the counter in least-significant-byte order, though some do not set the IP ID at all.

[4]We found that rate-limiting routers generally replied with the same source address and would be detected by Mercator.

Alias resolution using the IP identifier technique requires some engineering to keep from testing every pair of IP addresses. We reduce the search space with three heuristics. First, and most effectively, we exploit the hierarchy embedded in DNS names by sorting router IP addresses by their (piecewise) reversed name. For example, names like chi-sea-oc12.chicago.isp.net and chi-sfo-oc48.chicago.isp.net are lexigraphically adjacent, and adjacent pairs are tested. Second, router IP addresses whose replies have nearby return TTLs may also be aliases. Addresses are grouped by the TTL of their last response, and pairs with nearby TTL are tested, starting with those of equal TTL, then those within 1, etc. Of the 16 000 aliases we found, 94% matched the return TTL, while only 80% matched the outgoing TTL (the TTL that remained in the probe packet as it reached the router, which is included in the response). Third, "is an alias for" is a transitive relation, so demonstrating that IP_1 is an alias for IP_2, also demonstrates that all aliases for IP_1 are aliases for any of IP_2's aliases. Alias resolution is complete when all likely pairs of IP addresses are resolved as aliases, not aliases, or unresponsive.

There is a small probability that different routers will happen to pick nearby identifiers. To remove the resulting false positives, we repeat the alias resolution test to verify the alias.

C. Router Identification and Annotation

In this section, we describe how we determine which routers in the traceroute output belong to the ISP being mapped, their geographical location, and their role in the topology.

We rely on the DNS to identify routers that belong to the ISP. The DNS names provide a more accurate characterization than the IP address space advertised by the AS for three reasons. First, routers of non-BGP speaking neighbors are often numbered from the AS's IP address space itself. In this case, the DNS names help to accurately locate the ISP network edge because the neighboring domain routers are not named in the ISP's domain (e.g., att.net). Some ISPs use a special naming convention for neighboring domain routers to denote the network edge. For instance, small neighbors (customer organizations) of Sprint are named sl-neighborname.sprintlink.net, which is different from Sprint's internal router naming convention. Second, edge links between two networks could be numbered from either AS's IP address space. Again, DNS names help to identify the network edge. Finally, DNS names are effective in pruning out cable modems, DSL, and dialup modem pools belonging to the same organization as the ISP, and hence numbered from the same IP address space. We resort to the IP address space criterion for routers with no DNS names (we observed very few of these), with the constraint that all routers belonging to the ISP must be contiguous in the traceroute output.

One of our goals was to understand the structure of ISP maps, including their backbone and POPs. We identify the role of each router as well as its location using the information embedded in the DNS names. Most ISPs we studied have a naming convention for their routers that helps this effort. For example, sl-bb11-nyc-3-0.sprintlink.net is a Sprint backbone (bb11) router in New York City (nyc), and p4-0-0-0.r01.miamfl01.us.bb.verio.net is a Verio backbone (bb) router in Miami, Florida (miamfl01). We discover the naming convention of the ISP by browsing through the list of router names

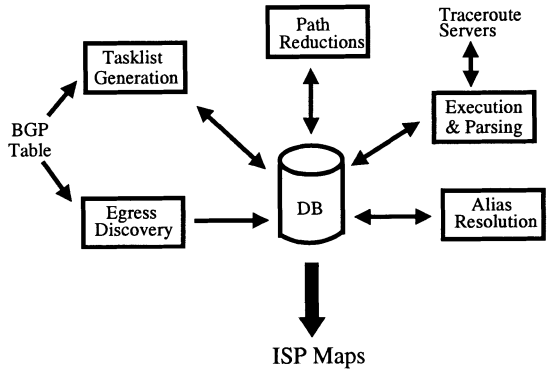

Fig. 6. Architecture of Rocketfuel. The database (DB) becomes the interprocess communication substrate.

we gather. For some ISPs, we started with city codes from the GeoTrack database [16]. Some routers have no DNS names or their names lack location information. We infer the location of such routers from that of its neighbors.

IV. Rocketfuel

In this section, we describe Rocketfuel, our ISP mapping engine. The architecture of Rocketfuel is shown in Fig. 6. A PostgreSQL database stores all information in a blackboard architecture: the database provides both persistent storage of measurement results and a substrate for interprocess communication between asynchronously running processes. The use of a database allows us to run SQL queries for simple questions and integrate new analysis modules easily.

We used 294 public traceroute servers listed at traceroute.org [10], representing 784 vantage points all across the world. A traceroute server may be configured to generate traceroutes from many routers in the same autonomous system: oxide.sprintlink.net generates traceroutes from 30 vantage points. Most (277) public traceroute servers, however, support only one source.

We now describe each module in Fig. 6. First, egress discovery is the process of finding the egress routers for dependent prefixes, which will be used for egress reduction. To find the egress routers, we traceroute to each dependent prefix from a local machine. Because dependent prefixes may be aggregated, we break them into /24's (prefixes of length 24, or, equivalently, 256 IP addresses) before probing. We assume that breaking down to /24's is sufficient to discover all ISP egress routers.

The tasklist generation module uses BGP tables from RouteViews [15] to generate a list of directed probes. The dependent prefixes in the directed probes are replaced with their egresses[5] and duplicates are removed. Tracing just to the egresses is an optimization for speed; we avoid sending probes into customer networks where they are likely to be filtered, which can slow traceroute collection.

Path reductions take the tasklist from the database, apply ingress and next-hop AS reductions, and generate jobs for execution. Information about traceroutes executed in the past is used by the path reductions module to determine, for example, which ingress is used by a vantage point. After a traceroute is taken, this module also checks whether the predicted ingress and egress were used. If so, the job is complete. Otherwise, another vantage point that is likely to take that path is tried.

[5]There may be several egresses for an aggregated prefix.

TABLE I
The Number of Routers, Links, and POPs for All Ten ISPs Studied. ISP Routers Include Backbone and Access Routers. With Customer and Peer Routers Adds Directly Connected Customer Access and Peer Routers. Links Include Only Interconnections Between These Sets of Routers. POPs are Identified by Distinct Location Tags in the ISP's Naming Convention

AS	Name	ISP		with customer & peer		POPs
		Routers	Links	Routers	Links	
1221	Telstra (Australia)	345	735	3,000	3,140	61
1239	Sprintlink (US)	471	1,337	8,280	9,022	44
1755	Ebone (Europe)	133	250	569	387	26
2914	Verio (US)	862	1,941	7,284	6,490	122
3257	Tiscali (Europe)	247	405	854	653	51
3356	Level3 (US)	624	5,299	3,446	6,741	53
3967	Exodus (US)	157	341	783	644	24
4755	VSNL (India)	11	12	120	68	11
6461	Abovenet (US)	357	914	2,249	1,292	22
7018	AT&T (US)	487	1,067	9,968	10,138	109
	Total	3,694	12,301	36,553	38,575	523

The execution engine handles the complexities of using public traceroute servers: load-limiting, load-balancing, and different formats of traceroute output. Load is distributed across destinations by randomizing the job list, implemented by sorting the MD5 hash [21] of the jobs. We enforce a five minute pause between accesses to the same traceroute server to avoid overloading it. Traceroutes to the same destination prefix are not executed simultaneously to avoid hotspots.

The traceroute parser extracts IP addresses that represent router interfaces and pairs of IP addresses that represent links from the output of traceroute servers. Often this output includes presentation markup like headers, tables, and graphics.

V. ISP Maps

We ran Rocketfuel to map ten diverse ISPs during December, 2001, and January, 2002. In this section, we present summary map information and samples of backbone and POP topology. The full map set, with images of the backbones and all the POPs of the ten ISPs, is available online [22]. We then analyze the ISP maps to report their properties, with the goal of understanding their structure and engineering. We describe the sizes and composition of POPs, degree distributions over both the router-level and backbone graph, and finally, the router-level adjacencies that make up inter-ISP peerings. We defer an evaluation of the validity of these maps to Section VI.

A. Summary Information

The aggregate statistics for all ten mapped ISPs are shown in Table I. The biggest networks, AT&T, Sprint, and Verio, are up to 100 times larger than the smallest networks we studied.

B. Backbones

Fig. 7 shows five sample backbones overlaid on a map of the U.S. Backbone design style varies widely between ISPs. We see that AT&T's backbone network topology includes hubs in major cities and spokes that fan out to smaller per-city satellite POPs. In contrast, Sprint's network has only 20 POPs in the U.S., all in

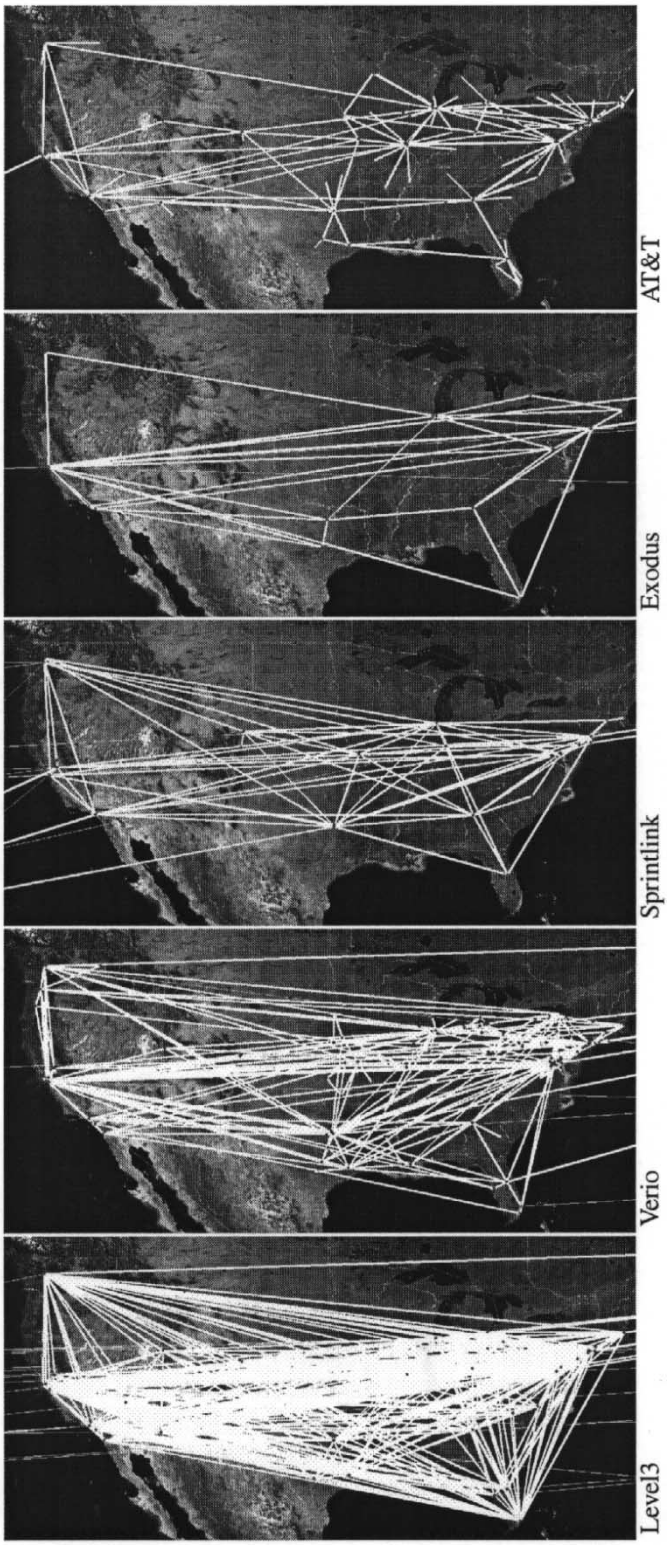

Fig. 7. Backbone topologies of U.S. ISPs, from top to bottom: AT&T, Exodus, Sprint, Verio, and Level 3. Multiple links may be present between two cities; only one is shown. Background image from NASA's Visible Earth project.

major cities and well connected to each other, implying that their smaller city customers are back-hauled into these major hubs. Level 3 represents yet another paradigm in backbone design, which is most likely the result of using a circuit technology, such as MPLS, ATM, or frame relay PVCs, to tunnel between POPs.

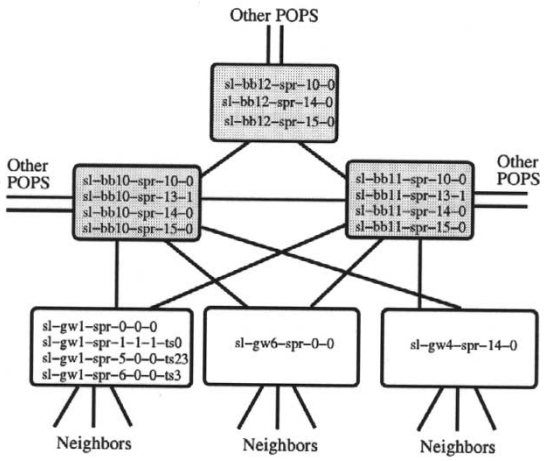

Fig. 8. Sample POP topology from Sprint in Springfield, Massachusetts. The names are prefixes of the full names, without sprintlink.net. Aliases for the same router are listed in the same box. Most POPs in Sprint are larger and too complex to show, but exhibit a similar structure.

C. POPs

Unlike the backbone designs, we found POP designs to be relatively similar. Each POP is a physical location where the ISP houses a collection of routers. A generic POP has a few backbone routers in a densely connected mesh. In large POPs, backbone routers may not be connected in a full mesh. Backbone routers also connect to backbone routers in other POPs. Each access router connects to one or more routers from the neighboring domain and to two backbone routers for redundancy. It is not necessary that all neighboring routers are connected to the access router using a point-to-point link. Instead, a layer-2 device such as a bridge or a multi-access medium such as a LAN may aggregate neighboring routers that connect to an access router. A limitation of our study is that traceroute cannot differentiate these scenarios from point-to-point connections.

As an example of a common pattern, Fig. 8 shows our map of Sprint's POP in Springfield, MA. This is a small POP; large POPs are too complex to show here in detail. In the figure, names of the aliases are listed together in the same box. The three backbone nodes are shown on top, with the access routers below. Sprint's naming convention is apparent: sl-bb*n* names backbone routers, and sl-gw*n* names their access routers. Most directly connected neighboring routers (not shown) are named as sl-neighborname.sprintlink.net. These are mainly small organizations for which Sprint provides transit. The value of DNS names for understanding the role of routers in the topology is clear from this naming practice.

D. POP Composition

The distribution of POP sizes, aggregated over the ten ISPs, is shown in Fig. 9. Most POPs are small, but most routers are in big POPs. In earlier work [25], we presented a sample of the variation by ISP: some have more small POPs or a few larger POPs. Small POPs may be called by other names within the ISP; we do not distinguish between exchange points, data centers, and private peering points.

In Fig. 10, we show the number of backbone routers relative to the total number of routers in the POP. Backbone routers

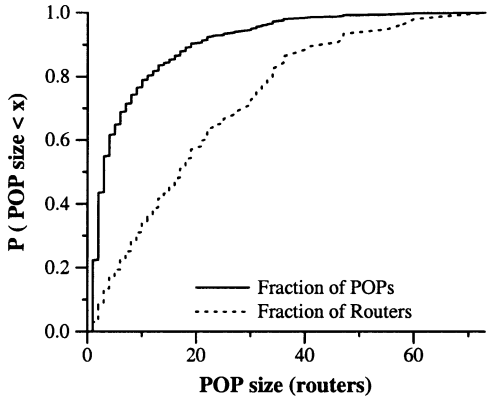

Fig. 9. Cumulative distribution of POP sizes (solid), and the distribution of routers in POPs of different sizes (dotted). The mean POP size is 7.4 routers, and the median is 3 routers.

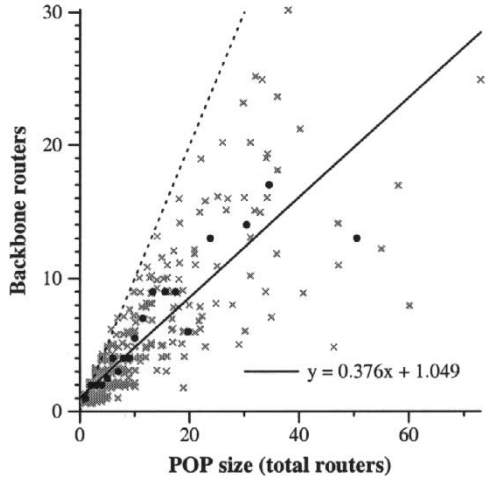

Fig. 10. Backbone routers in a POP relative to its size. A small random jitter was added to the data points to expose their density. Circles represent the median of at least ten nearby values: fewer medians are present for the few large POPs. The dotted line follows $x = y$, where all routers in a POP are backbone routers. The solid line traces a linear regression fit with $R^2 = 0.69$. This is an aggregate graph over the ten ISPs.

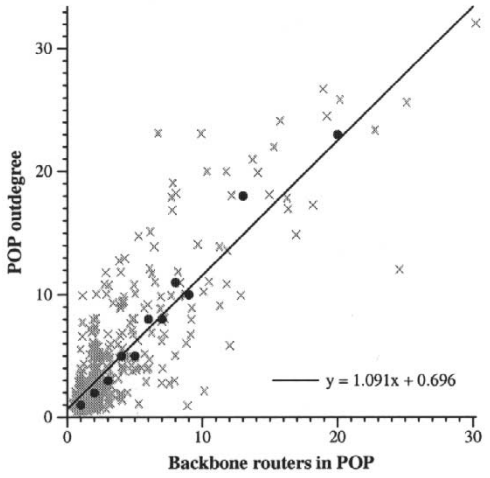

Fig. 11. POP outdegree versus backbone routers in the POP. A small random jitter was added to the data points to expose their density. Circles represent the median of at least ten nearby values: fewer medians are present for the few large POPs. The solid line traces a linear regression fit, with $R^2 = 0.70$. This is an aggregate graph over nine ISPs, excluding Level 3 due to its logical mesh topology that gives POPs very high outdegree.

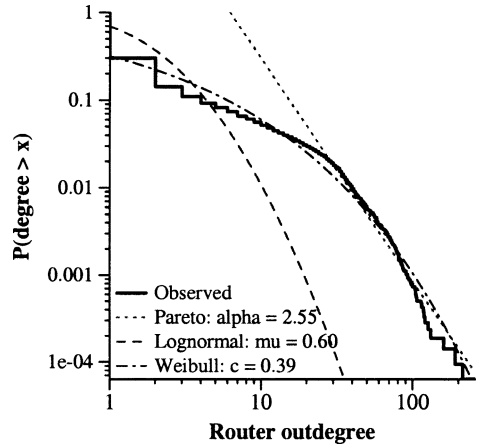

Fig. 12. Router outdegree CCDF. The Pareto fit is only applied to the tail. 65% of all routers have only a single link within the ISP; the mean outdegree is 3.0. This is an aggregate over nine of the ISPs: Level 3 is excluded due to its logical mesh topology.

are those that connect to other POPs, and the routers we consider are limited to those identifiable by DNS name and IP address as being part of the ISP. We define "backbone" in this ISP-independent way because DNS tags that represent the ISP's idea of a router's role in the topology are not universally used. Unsurprisingly, we find that most of the routers in small POPs are used to connect to other POPs, likely to the better connected core of the network. However, while we expected that as POPs became larger, a smaller fraction backbone routers would be required, instead we found that this is not always the case: POPs with more than 20 routers vary widely in the number of backbone routers used to serve them. We conclude from this graph that the smallest POPs have multiple backbone routers for redundancy, while larger POPs vary widely in the number of backbone routers present.

In Fig. 11, we show the outdegree of a POP as a function of the number of backbone routers present. We were surprised to find a roughly linear relationship. In general, the median tracks a line where the outdegree of a POP is equal to the number of backbone routers present. However, there are POPs where one or two backbone routers connect to several other POPs, and conversely there are POPs where several backbone routers provide redundancy in connecting to just a few other POPs. We conclude that there is no standard template for how backbone routers are connected to other POPs.

E. Router Degree Distribution

To describe the distribution of router outdegree in the ISP networks we use the complementary cumulative distribution function (CCDF). This plots the probability that the observed values are greater than the ordinate. We consider all routers, regardless of their role in the ISP.

The CCDF of router outdegree is shown in the aggregate over all ISPs in Fig. 12. We fit the tails of these distributions using Pareto (power-law), Weibull, and lognormal distributions. The α parameter for the Pareto fit is estimated over the right half of the graph to focus on the tail of the distribution. The Weibull scale and shape parameters are estimated using a linear regression over a Weibull plot. The lognormal line is based on the mean μ and variance of the log of the distribution.

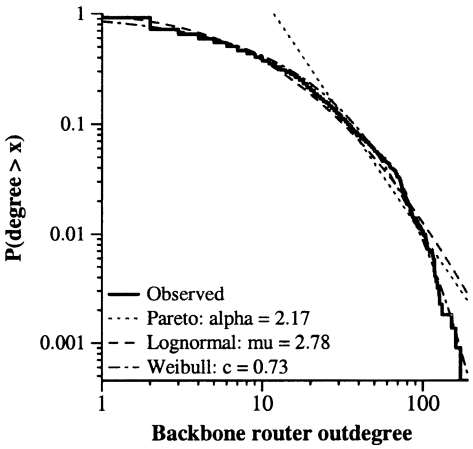

Fig. 13. Backbone router outdegree CCDF. The Pareto fit is only applied to the tail. The mean outdegree is 11.7, the median is 5. This is an aggregate over nine of the ISPs: Level 3 is excluded due to its logical mesh topology.

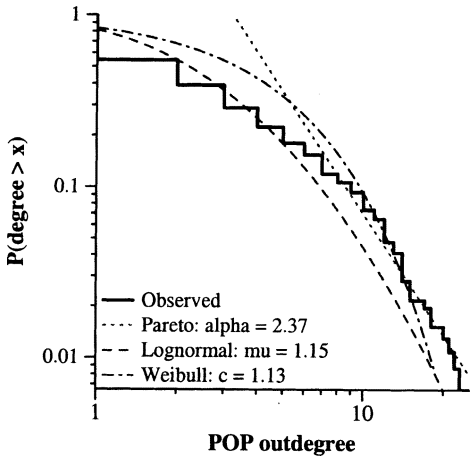

Fig. 14. POP outdegree CCDF, which represents the node degree distribution over the backbone graph where each node is a POP. The mean outdegree is 3.5, the median outdegree is 2. This is an aggregate over nine of the ISPs: Level 3 is excluded due to its logical mesh topology.

We observe that, unlike the measured degree in AS graphs [8], router outdegree has a small range in our data; it covers only two orders of magnitude over the ten ISPs. Physical size and power constraints naturally limit the underlying router outdegree. However, our data can include undetected layer-2 switches and multi-access links, which would inflate the observed router outdegree.

We next look closely at the distribution of outdegree for backbone routers. When we apply the same outdegree analysis over only those routers that we classify as "backbone," in that they connect to other POPs, we extract a visually different distribution in Fig. 13. This distribution of backbone router outdegree is more easily fit by the lognormal curve. While most ISP routers are "leaves" in that they connect to only one other ISP router, (over 65% as shown in Fig. 12), most backbone routers have high outdegree. We conclude that the backbone routers serve a noticeably different purpose in the topology—providing rich connectivity. Other routers in the network, while they may connect widely externally, are more likely to act as stubs within the ISP network.

F. Pop Degree Distribution

We now step back from the router-level topology to look at the POP-level topology. This topology is represented by the backbone graph: POPs are the nodes, and bidirectional backbone links connect them. Multiple links between POPs are collapsed into a single link. Fig. 14 shows the POP outdegree distribution. We find that this distribution is similar to that of routers, though over a smaller range. Nearly half of the POPs are stubs that connect to only one other POP. On the right-hand side of the graph, we can see that there are several POPs that act as hubs. We do not include Level 3 in Fig. 14: it creates a large mode at backbone outdegree around 50.

G. Peering Structure

Our maps are collected using traceroutes that enter and exit our ISPs at diverse points, giving us the unique opportunity to study the link-level peering structure between ASes. Adjacencies exposed in BGP tables show only that pairs of ASes connect somewhere. Using Rocketfuel topologies, however, we can infer where and in how many places our measured ISPs exchange traffic. For example, while BGP tables show that Sprint and AT&T peer, they do not show where the two ISPs exchange traffic.

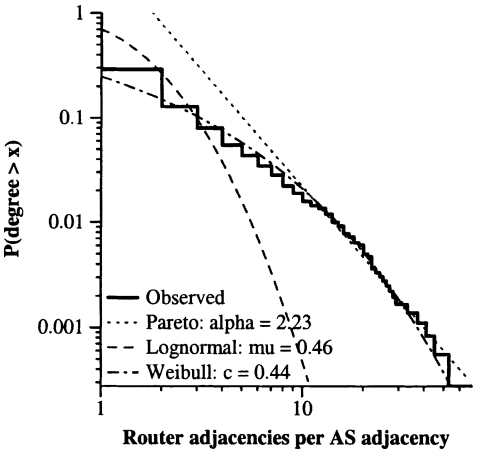

Fig. 15. CCDF of the number of router-level adjacencies seen for each AS-level adjacency. AS adjacencies include both peerings with other ISPs and peerings with customers that manage their own AS.

We summarize the link-level peering structure by showing the number of locations where the mapped ISP exchanges traffic with other ASes. The other ASes may represent other ISPs, whether in a transit or peer relationship, as well as customers running BGP, e.g., for multihoming. We use the same CCDF plot style for simplicity. Fig. 15 plots this CCDF, aggregated over the mapped ISPs. The Pareto, lognormal, and Weibull fits are calculated as before.

We see that the data is highly skewed for all the ISPs. Each ISP is likely to peer widely with a few other ISPs, and to peer in only a few places with many other ISPs. These relationships are perhaps not surprising given that the distribution of AS size and AS degree are heavy tailed [26].

We also see that the data has a small range, covering only one to two orders of magnitude. Some of the "peers" with many router-level adjacencies are actually different ASes within the same organization: AS 7018 peers with AS 2386 in 69 locations and with AS 5074 in 45 locations, but all three represent AT&T.

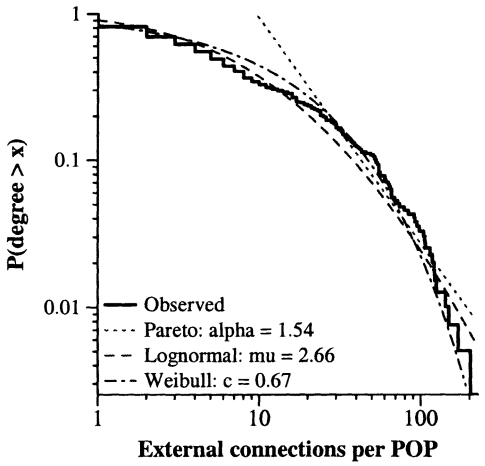

Fig. 16. CCDF of the number of external adjacencies per POP. Some POPs are particularly important, while most have at least a few external connections.

Discounting these outliers, the graphs show that it is rare for ISPs to peer in more than 30 locations.

In Fig. 16, we show a CCDF of the number of peering connections per POP. This graph relates to the outdegree graphs previously presented in that this shows the outdegree of a POP in terms of the number of its external connections. There are a handful of cities that are central, in which our ISPs connect to hundreds of other ASes. However, most cities house only a few external connections.

H. Summary

In this section, we have shown several attributes of the ISP maps that exhibit skewed or highly variable distributions. These include peering degree, POP-external connection degree, POP outdegree, router outdegree, backbone router outdegree, and POP size. While the best-fit functions and parameters for each of these distributions vary, the theme is consistent: skewed distributions are endemic to network topologies at every level. We looked at the structural breakdown of POPs into backbone routers and other routers, and found that large POPs vary widely in the number of backbone routers present, and that while the number of backbone routers tends to be dependent on the outdegree of the POP, it may vary widely for small POPs that may have special roles within the topology. However, distributions alone do not characterize the design of these networks. We found that the ISPs differ in how they engineer their POP interconnections, and observed that backbone routers differ from the rest in how they are internally connected.

VI. Validation

In this section, we evaluate the effectiveness of our techniques along two axes: the fidelity of the resulting maps and the efficiency with which they were constructed.

A. Completeness

We used four independent tests to estimate the accuracy and completeness of our maps. First, we asked the ISPs that we mapped to help with validation. Second, we devised a new technique to estimate the completeness of an ISP map using IP address coverage. Third, we compared the BGP peerings we found to those present at RouteViews. Finally, we compared our maps with those obtained by Skitter [7], an ongoing Internet mapping effort at CAIDA.

1) Validating With ISPs: Three out of ten ISPs assisted us with a partial validation of their maps. We do not identify the ISPs because the validation was confidential. Below, we list the questions we asked and the answers we received.

1) *Did we miss any POPs*? All three ISPs said *No*. In one case, the ISP pointed out a mislocated router; the router's city code was not in our database.
2) *Did we miss any links between POPs*? Again, all three said *No*, though in two cases we had a spurious link in our map. This could be caused by broken traceroute output or a routing change during the trace, as we expected (Section II).
3) *Using a random sample of POPs, what fraction of access routers did we miss*? One ISP could not spot obvious misses; another said all backbone routers were present, but some access routers were missing; and the third said we had included routers from an affiliated AS.
4) *What fraction of customer routers did we miss*? None of the ISPs were willing to answer this question. Two claimed that they had no way to check this information.
5) *Overall, do you rate our maps: poor, fair, good, very good, or excellent*? We received the responses: "Good," "Very good," and "Very good to excellent."

We found these results encouraging, as they suggest that we have a nearly accurate backbone and reasonable POPs. This survey and our own validation attempts using public ISP maps also confirms to us that the public maps are not authoritative sources of topology. They often have missing POPs and optimistic deployment projections, and show parts of partner networks managed by other ISPs.

2) IP Address Space: As an estimate of the lower bound of the completeness of these maps, we randomly searched prefixes of the ISP's address space for additional responsive IP addresses. Finding new routers by scanning the ISP's IP address space would tell us that our traceroutes have not covered some parts of the topology. We randomly selected 60 /24 prefixes from each ISP that included at least two routers from our measured maps to search for new routers. Most ISPs appear to assign router IP addresses in a few blocks; this simplifies management.[6] New IP addresses are those that both respond to ping and have names that follow the ISP's router naming convention, though they may or may not participate in forwarding. Prefixes were chosen to make sure that both backbone and access routers were represented.

The criteria we chose for this test provide a lower bound on completeness. First, any new address found through IP address scanning need only have a name that follows the ISP convention, while those found through traces have demonstrated that they are attached to routers that participate in forwarding. Second, the percentage comparison applies to addresses and not routers. We use alias resolution in this test only to remove aliases for already known routers, which means this completeness estimate is independent of the performance of our alias resolution tool, but unknown addresses may belong to just a handful of routers.

[6]We select only prefixes with at least *two* routers because many prefixes used to connect ISPs will have only one router from the mapped ISP: our coverage of such a prefix would be 100%, providing little information.

TABLE II
ESTIMATE OF ROCKETFUEL'S COVERAGE OF IP ADDRESSES NAMED LIKE ROUTERS. ALIASES OF KNOWN ROUTERS ARE NOT COUNTED. "n/a" IMPLIES THAT THE ISP'S NAMING CONVENTION DOES NOT DIFFERENTIATE BETWEEN BACKBONE AND ACCESS ROUTERS

AS	Backbone	Access	Total
Telstra (1221)	64.4%	78.1%	48.6%
Sprint (1239)	90.1%	35.0%	61.3%
Ebone (1755)	78.8%	55.1%	65.2%
Verio (2914)	75.1%	60.6%	57.5%
Tiscali (3257)	89.1%	n/a	41.5%
Level3 (3356)	78.6%	77.4%	55.6%
Exodus (3967)	95.4%	59.8%	53.6%
VSNL (4755)	n/a	n/a	48.4%
Abovenet (6461)	83.6%	n/a	76.0%
AT&T (7018)	65.4%	91.6%	78.9%

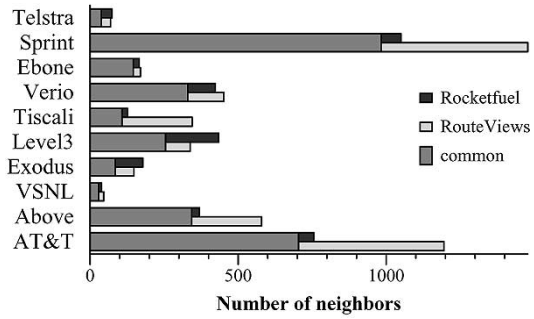

Fig. 17. Comparison between BGP adjacencies seen in our maps and those seen in the BGP tables from RouteViews.

Table II shows the estimated percentage coverage for each ISP. This is calculated as the number of known IP addresses relative to the total number of addresses seen in the subnets, not counting additional aliases of known routers. If the ISP has a consistent naming convention for backbone routers and access routers, the total is broken down into separate columns, otherwise, "n/a" is shown. The table suggests that we find from 64%–96% of the ISP backbone routers. The access router coverage is fair, and in general less than backbone coverage. We plan to investigate the differences between the routers found by Rocketfuel and address range scanning.

3) Comparison With RouteViews: Another estimate for completeness is the BGP adjacencies seen in our maps compared to those in the BGP tables from RouteViews [15]. For each adjacency in the BGP table, a complete router-level map should include at least one link from a router in the mapped AS to one in the neighboring AS.

Fig. 17 compares the number of adjacencies seen by Rocketfuel and RouteViews. The worst case for Rocketfuel is AT&T (7018), where we still find more than 63% of the neighbors. Rocketfuel discovers some neighbors that are not present in RouteViews data, a result consistent with that found by Chang *et al.* [6]. We studied the adjacencies found by both approaches and found that RouteViews contains more adjacencies to small (low degree in the AS graph) neighbors, while Rocketfuel finds more adjacencies with large neighbors. The intuition is that BGP is more likely to expose the preferred routes through customer networks (smaller neighbors) while Rocketfuel is more likely to traverse edges between large ISPs.

4) Comparison With Skitter: Skitter is a traceroute-based mapping project run by CAIDA [7]. Skitter has a different goal: to map the entire Internet, and a different approach: many traceroutes from tens of dedicated servers. Although using public traceroute servers is unlikely to scale to the whole Internet, we show that there is additional detail to be found. We analyzed Skitter data collected on November 27 and 28, 2001. (Rocketfuel collected data primarily during January, 2002.) We compare the IP addresses, routers after alias resolution, and links seen in each mapped AS by both techniques and by only one. The IP address statistics are presented for each AS in Fig. 18 and all three statistics are summarized in Table III.

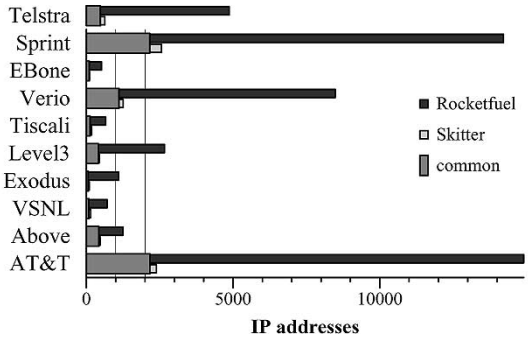

Fig. 18. Comparison between unique IP addresses discovered by Rocketfuel and Skitter for each ISP we studied.

TABLE III
COMPARISON OF LINKS, IP ADDRESSES, AND ROUTERS DISCOVERED BY ROCKETFUEL AND SKITTER, AGGREGATED OVER ALL TEN ISPS. UNIQUE FEATURES ARE THOSE THAT ARE ONLY FOUND IN ONE OF THE MAPS. UNIQUE ROUTERS ARE THOSE THAT HAVE NO ALIASES IN THE OTHER DATA SET

	Links		IP addresses		Routers	
	Total	Unique	Total	Unique	Total	Unique
Rocketfuel	69711	61137	49364	42243	41293	36271
Skitter	10376	1802	8277	1156	5892	870

Rocketfuel finds six to seven times as many links, IP addresses, and routers in its area of focus. Some routers and links were only found by Skitter. While some of this difference is due to the different times of map collection, most corresponds to routers missed by Rocketfuel. We investigated and found that the bulk of these were neighboring domain routers and some were access routers. That both tools find different routers and links underscores the complexity of Internet mapping.

B. Impact of Reductions

This section evaluates directed probing and path reductions described in Section III. We evaluate these techniques for both the efficiency gained through reduction and the accuracy that may be lost. Most results presented here are aggregated over all ten ISPs we map; individual results were largely similar. We first present directed probing, followed by each of the three path reductions, then describe their combined impact.

1) Directed Probing: We consider three aspects of directed probing: the fraction of traces it can prune; the number of pruned traces that would have transited the ISP and should have been kept; and the traces that should have been discarded because they did not transit the ISP.

The effectiveness of directed probing is shown in Table IV. The brute-force search from all vantage points to all BGP-advertised prefixes (using /24's within the ISP) would require

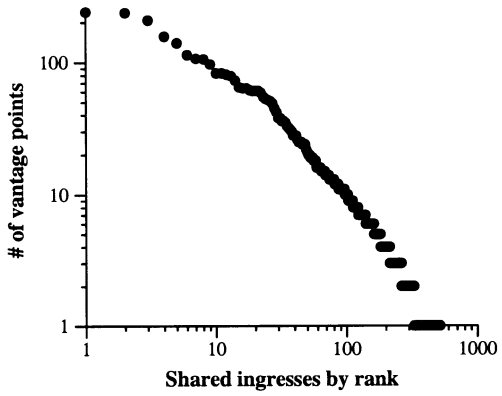

Fig. 19. Number of vantage points that share an ingress, by rank, aggregated across ASes. 232 vantage points share the same ingress at left, while 247 vantage points have unique ingresses. The area under the curve represents the number of vantage points we used times the ten ISPs we mapped.

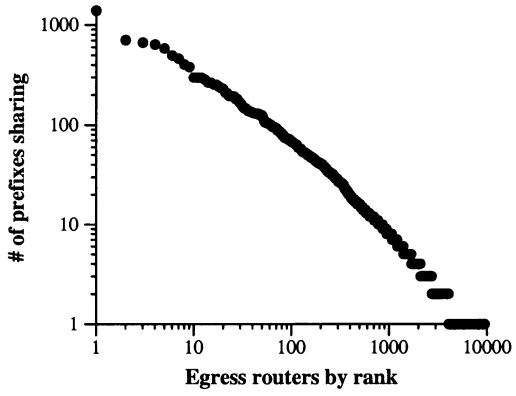

Fig. 20. Number of dependent prefixes that share an egress, by rank, and aggregated across all ASes.

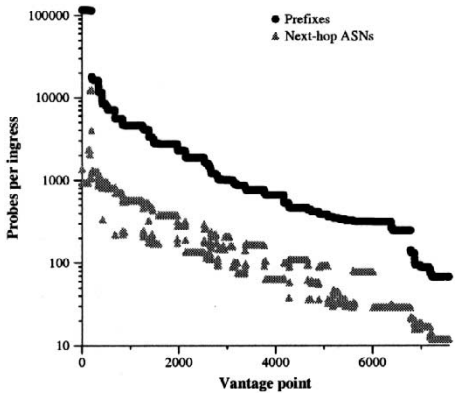

Fig. 21. Number of prefixes and unique next-hop ASes for vantage points. A vantage point is counted once for each mapped ISP.

90–150 million traceroutes. With directed probing only between 0.3%–17% of these traces are chosen by Rocketfuel.

We used Skitter data to estimate how many useful traces, which would traverse the ISP, are pruned by directed probing. We use directed probing to select traces for Skitter vantage points to collect in mapping our ISPs, then calculate the fraction of actual Skitter traces, collected through brute-force mapping, that did traverse the ISP but were not selected. This fraction of useful but pruned traces varies by ISP from 0.1% to 7%. It is low for non-U.S. ISPs like VSNL (4755) and Tiscali (3257), and high for the big U.S. ISPs like AT&T and Sprint. This variation can be attributed to the difference in the likelihood that a trace from a vantage point to a randomly selected destination will traverse the ISP. Even when the fraction of useful traces is 7%, without extra information, such as BGP tables collected at the traceroute server itself, we would have to carry out 100 extra measurements to get seven potentially useful ones. We did not explore how many of these potentially useful traces would traverse new paths.

To determine how many traces we took that were unnecessary, we tally directly from our measurement database. Roughly 6% of the traces we took did not transit the ISP.

These numbers are encouraging: not only does directed probing cut the number of traces dramatically, but little useful work is pruned out, and little useless work is done.

2) Ingress Reduction: In this section, we evaluate ingress reduction for its effectiveness in discarding unnecessary traces. Ingress reduction kept 2%–26% (12% overall) of the traces chosen by directed probing. For VSNL, ingress reduction kept only 2% as there were only a few ingresses for our many vantage points. In contrast, it kept 26% of the traces chosen by directed probing of Sprint.

The distribution of vantage points that share an ingress is given in Fig. 19. The number of vantage points sharing an ingress is sorted in decreasing order, and plotted on a log-log scale. From the right side of the curve, we see that the approach of using public traceroute servers provides many distinct ingresses into the mapped ASes. At the left, many vantage points share a small number of ingresses, which implies that ingress reduction significantly reduces the amount of work necessary, even after directed probing.

3) Egress Reduction: Overall, egress reduction kept only 18% of the dependent prefix traces chosen by directed probing. Fig. 20 shows the number of dependent prefixes that share an egress router. The x-axis represents each egress router, and the y-axis represents the number of prefixes that share that egress. The left part of the curve depicts egresses shared by multiple prefixes, and demonstrates the effectiveness of egress reduction. The right part shows that many prefixes had unique egresses.

To test our hypothesis that breaking larger prefixes into /24's is sufficient for egress discovery, we randomly chose 100 /24's (half of these were ISP prefixes) from the set of dependent prefixes and broke them down further into /30's. We then traced to each /30 from our machine. The ratio of previously unseen egresses to the total discovered is an estimate of accuracy lost in the ISP boundaries due to not breaking down more finely. Overall, 0–20% of the egresses discovered during this process were previously unseen, with the median at 8%. This wide range suggests that our assumption, while valid for some ISPs (two had virtually no new egresses), is not universally applicable. This is perhaps because the minimum customer allocation unit used by some ISPs is smaller than a /24. In the future, we intend to dynamically explore the length to which each dependent prefix should be broken down to discover all egresses.

4) Next-Hop AS Reduction: Next-hop AS reduction selects only 5% of the up/down and insider traces (these two classes leave the ISP and proceed to enter another AS) chosen by directed probing. In Fig. 21, we show the number of prefixes

TABLE IV
THE EFFECTIVENESS OF DIRECTED PROBING, ALONG WITH A SUMMARY OF THE NUMBER OF TRACEROUTES TAKEN. ROCKETFUEL EXECUTES BOTH THE REMOTE TRACEROUTES, CHOSEN AFTER PATH REDUCTIONS ARE APPLIED TO THE DIRECTED PROBES, AND THE EGRESS DISCOVERY TRACEROUTES. THE TOTAL COLUMN FOR THE BRUTE-FORCE TRACES IS OMITTED: IT WOULD BE CHEAPER TO GENERATE A WHOLE-INTERNET MAP

ASN	Name	Brute Force	Directed Probes		Remote Traceroutes	Egress Discovery	Overall Reduction
1221	Telstra (Australia)	105 M	1.5 M	(1.4%)	20 K	20 K	0.04%
1239	Sprintlink (US)	132 M	10.3 M	(7.8%)	144 K	54 K	0.15%
1755	Ebone (Europe)	91 M	15.3 M	(16.8%)	16 K	1 K	0.02%
2914	Verio (US)	118 M	1.6 M	(1.3%)	241 K	36 K	0.23%
3257	Tiscali (Europe)	92 M	0.2 M	(0.2%)	6 K	2 K	0.01%
3356	Level3 (US)	98 M	5.0 M	(5.1%)	305 K	10 K	0.32%
3967	Exodus (US)	91 M	1.2 M	(1.3%)	24 K	1 K	0.03%
4755	VSNL (India)	92 M	0.5 M	(0.5%)	5 K	2 K	0.01%
6461	Abovenet (US)	92 M	0.7 M	(0.7%)	111 K	3 K	0.12%
7018	AT&T (US)	152 M	4.5 M	(2.9%)	150 K	80 K	0.15%
	Total		40.8 M		1022 K	209 K	

chosen for each vantage point (the upper line), and the number of next-hop ASes that represent jobs after reduction. Next-hop reduction is effective because the number of next-hop ASes is consistently much smaller than the number of prefixes. It is particularly valuable for insiders who, with only directed probing, would otherwise traceroute to all 120 000 prefixes in the RouteViews BGP table. Next-hop AS reduction allows insiders to instead trace to only the 1000 or so external destinations that cover the set of possible next hops.

Next-hop AS reduction achieves this savings by assuming that routes are chosen based solely on the next-hop AS, and not differently for each prefix it advertises. Commonly, this is equivalent to whether the ISP uses "early exit" routing. However, the reduction preserves accuracy as long as the traces from each ingress to randomly chosen prefixes in the next-hop AS are sufficient to cover the set of links to that AS.

We used Verio to test how frequently this assumption is violated by conducting 600K traces without the reduction. The traces contained 2500 (ingress, next-hop AS) pairs, of which only 7% included more than one egress, violating the assumption. Different ISPs have different policies regarding per-prefix interdomain routing, but nevertheless, this result is encouraging.

5) Overall Impact: Our reductions are mostly orthogonal and they compose to give multiplicative benefit. Table IV shows the total number of traceroutes that we collected to infer the maps. We executed less than 0.1% of the traces required by a brute-force technique, a reduction of three orders in magnitude. The individual reductions varied between 0.3% (Level 3) to 0.01% (VSNL and Tiscali).

Our mapping techniques also scale with the number of vantage points. Extra vantage points contribute either speed or accuracy. Speed is increased when the new vantage point shares an ingress with an existing vantage point because more traceroutes can execute in parallel. Accuracy is improved if the new vantage point has a unique ingress to the ISP.

C. Alias Resolution

The effectiveness of both the IP-address-based approach and our new approach to alias resolution is shown in Table V. As mentioned in Section III-B, we used the three-packet version of Ally. The table shows how many aliases, which are additional IP addresses for the same router beyond the first, were found by each technique. Ally's IP identifier-based technique finds almost three times more aliases than the earlier address-based approach. Moreover, we found aliases resolved using the IP identifier to be a superset of those resolved by an address-based technique. This means that using only Ally suffices for alias resolution.

TABLE V
ALLY'S IP IDENTIFIER-BASED TECHNIQUE FINDS BETWEEN 1.5 TO 4 TIMES AS MANY ALIASES AS AN ADDRESS-BASED TECHNIQUE. DIFFERENT ISPS MAY PREFER DIFFERENT ROUTERS FROM DIFFERENT VENDORS, ACCOUNTING FOR THE DIFFERENCE BY ISP, AND THESE RESULTS MAY CHANGE OVER TIME

ISP	Alias resolution method		Ratio
	IP identifier	IP address	
Telstra	1,142	483	2.36
Sprint	4,406	2,357	1.87
Ebone	869	590	1.47
Verio	2,332	747	3.12
Tiscali	631	354	1.78
Level3	1,537	465	3.31
Exodus	1,390	352	3.95
VSNL	191	123	1.55
Abovenet	1,557	491	3.17
AT&T	2,966	1,182	2.51
Total	17,021	7,144	2.38

To build confidence that the resolved aliases were correct and complete, we compare the aliases found by Ally to those predicted by DNS names. We chose two ISPs, Ebone and Sprint, that name many of their routers with easily recognized unique identifiers. This provides a reference for estimating how many aliases our technique missed. Of the DNS predicted aliases for Sprint, 240 backbone and gateway routers were correctly resolved. However, 63 routers did not resolve correctly: 30 of these routers had at least one interface address that never responded. We correctly resolved 119 of 139 Ebone routers, five of which failed from unresponsive addresses.

This suggests that a problem for even the most effective alias resolver is how to handle unresponsive IP addresses. Out of 56 000 IP addresses in our maps, we found nearly 6000 that never responded to our alias resolution queries.

We plan to investigate why there were 33 Sprint and 15 Ebone routers that were responsive, but were not completely and correctly resolved. Potential causes include temporarily unrespon-

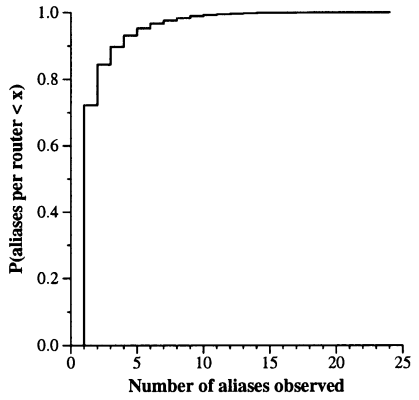

Fig. 22. Number of aliases observed for routers within the mapped ISPs.

sive routers, stale or incorrect DNS entries, and routers with multiple IP stacks (and thus multiple IP identifier counters).

Fig. 22 plots a cumulative distribution function (CDF) of how many aliases we saw for routers within the ISPs we mapped. We saw only one IP address for 70% of the routers, and two IP addresses for another 10%. The maximum number of aliases observed was 24, for an AT&T router in New York. This graph is an underestimate of the number of aliases that routers have, since it is likely that we do not see all IP addresses for a router.

D. Summary

To assess the mapping techniques in Rocketfuel, we checked the resulting maps for completeness and accuracy, and estimated the effectiveness of these techniques at reducing workload. Network operators informed us that our maps were good, though imperfect. We found them to be substantially more detailed in the ISP networks we studied than earlier Internet-wide maps, uncovering six to seven times more routers and links. To obtain a weak lower bound on the completeness of the maps, we scanned the IP address space of the ISPs and found that we have at least half of the routers in the real topology. Similarly, a comparison with RouteViews data shows that we find at least two-thirds of the peerings for all maps, and typically much more.

Compared to a naive all-to-all measurement scheme, directed probing and path reductions reduced the number of measurements to map the ISPs by three orders of magnitude on average. We used test cases to estimate both how many useful measurements were omitted and how many uninformative measurements we took. These evaluations yielded encouraging results: for instance, using directed probing, 7% of the traceroutes we omitted might have been of use, while 6% of those taken were not.

We also evaluated the effectiveness of the new IP-identifier-based alias resolution tool. We found it performed well, but incompletely resolved roughly 10% of the IP addresses because they did not respond to measurement probes. On average, our tool found three times as many aliases as the earlier method, of which the aliases found by the latter were essentially a subset.

VII. Related Work

Several research efforts have attempted to infer the router-level topology of the Internet. An early attempt started with a list of 5000 destinations and used traceroutes from a single network node [17]. Mercator is also a map collection tool run from a single host [9]. Instead of a list of hosts, it uses *informed random address probing* to find destinations. Both these efforts explore the use of source routing to discover cross-links to improve the quality of the network map. Burch and Cheswick use BGP tables to find destination prefixes [5]. They source traceroutes from a single machine, but improve coverage by using tunnels to other machines on the network, similar in effect to using multiple vantage points. Skitter, a topology collection project at CAIDA, uses BGP tables and a database of Web servers to find destination prefixes [7]. Skitter monitors probe these networks from about 20 different locations worldwide. Our mapping goal differs fundamentally from all of these efforts. Instead of trying to collect the router-level map of the whole Internet, we focus probes on individual ISP networks. The result is an ISP map that is more complete than that obtained by other mapping efforts.

Barford *et al.* have analyzed the marginal utility of adding vantage points and destinations to discover the Internet backbone topology [2]. Our work is similar in that we also try to minimize the number of measurements needed, but while we use routing knowledge to eliminate individual traces, Barford *et al.* try to find the minimal set of vantage points.

While our focus is on router-level topologies, measurement and characterization of AS-level topologies has been the subject of much work [4], [6], [8]. Recently, Andersen *et al.* [1] have inferred the internal logical topology of two ISPs by observing correlations between BGP interdomain routing update messages. Correlated update messages imply that some prefixes attach to the network at the same point or nearby [1].

VIII. Conclusion and Future Work

In this paper, we presented new techniques for mapping the router-level topology of focused portions of the Internet, such as an ISP network or an exchange point, using only end-to-end measurements. We have shown that routing information can be exploited in several ways to perform only those measurements that are expected to be useful, reducing the mapping workload by three orders of magnitude compared to a brute-force all-to-all approach with little loss in accuracy. This enabled us to use nearly 300 public traceroute servers as measurement sources, providing us with nearly 800 vantage points, which is many more than other mapping efforts. We also presented a new alias resolution technique that discovered three times more aliases than the current approach based on return addresses. This increased the accuracy of our maps compared to earlier efforts.

We used our new techniques to map ten diverse ISPs, and are releasing both the composite maps and raw data to the community [22]. We find that all ISPs are structured as POPs connected by backbone routers but that ISPs differ noticeably in the design of their networks. In all cases, skewed distributions are endemic to network topologies at every level, from router outdegree to POP size and number of peerings. To validate the maps, we compared them with 1) the true map as understood by the ISP operators; 2) the total number of routers found by scanning sampled subnets; 3) the peerings known to exist from BGP tables; and 4) maps extracted from Skitter. Our maps stack up well in these comparisons. They contain roughly seven times as many nodes and links in the area of focus as Skitter, and are sufficiently complete by the other metrics that we believe they are representative models for ISP networks.

Our work can readily be extended in several dimensions. First, the data we are releasing can be used to study properties of Internet topology. We reported new results for the distribution of POP sizes and the number of times that an ISP connects with other networks, finding that both distributions have significant tails. Second, we can extract other kinds of properties such as routing and failure models from the traceroutes. This can be used to annotate the ISP maps and improve their utility. As an example, we have recently devised a method for inferring approximate link weights to characterize the routes that are taken over the underlying topology [13]. Finally, improvements to these techniques could lead to high quality mapping that is efficient enough to perform on demand.

Our efforts with Rocketfuel to date have greatly increased the availability of network topologies as well as deepened their characterizations. At the same time, it is clear to us that we have only scratched the surface of what is possible in terms of understanding models of the Internet.

ACKNOWLEDGMENT

The authors are grateful to the administrators of the traceroute servers whose public service enabled their work and the operators who provided feedback on the quality of their maps. They also thank S. Bellovin, C. Diot, and R. Bush for early insights into ISP backbone and POP topologies. L. Subramanian provided Geotrack scripts. CAIDA provided skitter data. R. Govindan helped test the alias resolution technique and provided mapping advice. H. Hagerstrom assisted in some analyses. A. Downey provided lognormal distribution analysis tools and guidance. W. Willinger provided helpful feedback on the implications of our analysis results.

REFERENCES

[1] D. G. Andersen, N. Feamster, S. Bauer, and H. Balakrishnan, "Topology inference from BGP routing dynamics," in *Proc. ACM SIGCOMM Internet Measurement Workshop (IMW)*, Nov. 2002, pp. 243–248.

[2] P. Barford, A. Bestavros, J. Byers, and M. Crovella, "On the marginal utility of network topology measurements," in *Proc. ACM SIGCOMM Internet Measurement Workshop (IMW)*, Nov. 2001, pp. 5–17.

[3] A. Basu and J. Riecke, "Stability issues in OSPF routing," in *Proc. ACM SIGCOMM*, Aug. 2001, pp. 225–236.

[4] T. Bu and D. Towsley, "On distinguishing between Internet power law topology generators," in *Proc. IEEE INFOCOM*, vol. 2, June 2002, pp. 638–647.

[5] H. Burch and B. Cheswick, "Mapping the Internet," *IEEE Computer*, vol. 32, pp. 97–98, Apr. 1999.

[6] H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Toward capturing representative AS-level Internet topologies," in *ACM SIGMETRICS*, June 2002, pp. 280–281.

[7] K. Claffy, T. E. Monk, and D. McRobb, "Internet tomography," *Nature*, Jan. 1999.

[8] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the Internet topology," in *Proc. ACM SIGCOMM*, Sept. 1999, pp. 251–262.

[9] R. Govindan and H. Tangmunarunkit, "Heuristics for Internet map discovery," in *Proc. IEEE INFOCOM*, vol. 3, Mar. 2000, pp. 1371–1380.

[10] T. Kernen, Traceroute.org, [Online.] http://www.traceroute.org.

[11] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," in *Proc. ACM SIGCOMM*, Sept. 2000, pp. 175–187.

[12] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high-bandwidth aggregates in the network," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 3, pp. 62–73, July 2002.

[13] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson, "Inferring link weights using end-to-end measurements," in *Proc. ACM SIGCOMM Internet Measurement Workshop (IMW)*, Nov. 2002, pp. 231–236.

[14] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: an approach to universal toplogy generation," in *Proc. MASCOTS*, Aug. 2001.

[15] D. Meyer, RouteViews Project, [Online.] http://www.routeviews.org.

[16] V. N. Padmanabhan and L. Subramanian, "An investigation of geographic mapping techniques for Internet hosts," in *Proc. ACM SIGCOMM*, Aug. 2001, pp. 173–185.

[17] J. Pansiot and D. Grad, "On routes and multicast trees in the Internet," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 28, no. 1, pp. 41–50, Jan. 1998.

[18] K. Park and H. Lee, "On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets," in *Proc. ACM SIGCOMM*, Aug. 2001, pp. 15–26.

[19] G. Philips, S. Shenker, and H. Tangmunarunkit, "Scaling of multicast trees: Comments on the Chuang-Sirbu scaling law," in *Proc. ACM SIGCOMM*, Aug. 1999, pp. 41–52.

[20] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin, "On Characterizing network topologies and analyzing their impact on protocol design," Univ. of Southern Calif., Tech. Rep. CS-00-731, 2000.

[21] R. Rivest, "The MD5 Message-Digest Algorithm," Network Working Group, RFC 1321, Apr. 1992.

[22] Univ. of Washington, Rocketfuel Maps and Data, [Online.] http://www.cs.washington.edu/research/networking/rocketfuel/.

[23] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for IP traceback," in *Proc. ACM SIGCOMM*, Aug. 2000, pp. 295–306.

[24] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer, "Hash-based IP traceback," in *Proc. ACM SIGCOMM*, Aug. 2001, pp. 3–14.

[25] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with Rocketfuel," in *Proc. ACM SIGCOMM*, Aug. 2002, pp. 133–146.

[26] H. Tangmunarunkit, J. Doyle, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Does AS size determine degree in AS topology?," *ACM Comput. Commun. Rev.*, vol. 31, no. 4, pp. 7–10, Oct. 2001.

[27] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Network topology generators: degree-based vs. structural," in *Proc. ACM SIGCOMM*, Aug 2002, pp. 147–160.

[28] E. W. Zegura, K. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. IEEE INFOCOM*, Mar. 1996, pp. 594–602.

Neil Spring received the B.S. degree in computer engineering from the University of California, San Diego, in 1997, and the M.S. degree in computer science from the University of Washington, Seattle, in 2000. He is currently working toward the Ph.D. degree at the University of Washington.

His research interests include congestion control, network performance analysis, distributed operating systems, adaptive scheduling of distributed applications, and operating system support for networking.

Mr. Spring has been a student member of the Association for Computing Machinery since 2001.

Ratul Mahajan received the B.Tech. degree from the Indian Institute of Technology, Delhi, India, and the M.S. degree from the University of Washington, Seattle. He is currently working toward the Ph.D. degree at the University of Washington.

Mr. Mahajan has been a student member of the Association for Computing Machinery since 2000.

David Wetherall (M'89) received the B.E. degree from the University of Western Australia in 1989 and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, in 1998. He is currently a member of the Faculty of Computer Science and Engineering at the University of Washington. His research interests span the range from distributed systems, to internetworking, to programming languages. Dr. Wetherall's thesis research helped to pioneer the field of Active Networks, in which flexible network infrastructures are used to enable rapid service innovation.

Dr. Wetherall has been a member of the Association for Computing Machinery since 1995.

Thomas Anderson received the A.B. degree in philosophy from Harvard University, Cambridge, in 1983 and the Ph.D. degree in computer science from the University of Washington, Seattle, in 1991.

He taught at the University of California at Berkeley from 1991 to 1996. He is currently a Professor in the Department of Computer Science and Engineering, University of Washington. His research interests include most systems research topics, from multiprocessor and uniprocessor operating system design, scalable and reliable cluster file systems, high-performance network switch design, wide area distributed systems, compiler-assisted fault isolation, processor-in-memory computer architectures, to his most recent focus on Internet measurement, reliability, and evolvability. Many of his papers have received awards at various systems research conferences.