Measuring ISP Topologies with Rocketfuel

Neil Spring Ratul Mahajan David Wetherall

{nspring,ratul,djw}@cs.washington.edu
Computer Science and Engineering
University of Washington
Seattle, WA 98195-2350

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 directly 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, exploiting properties of IP routing to eliminate redundant measurements, 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 inter-domain peering structure. As part of this work, we release our maps to the community.

Categories and Subject Descriptors

C.2.1 [**Communication Networks**]: Architecture and Design—*topology*

General Terms

Measurement

1. INTRODUCTION

Realistic Internet topologies are of considerable importance to network researchers. Topology influences the dynamics of routing protocols [2, 10], the scalability of multicast [17], the efficacy of proposals for denial-of-service tracing and response [16, 11, 21, 22], and other aspects of protocol performance [18].

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simplified topologies on the Web, but these lack router-level connectivity and POP structure and may be optimistic or out of date.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 in [7]) that it is unclear whether synthetic topologies generated by tools such as GT-ITM [26] or Brite [12] are representative [25].

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 technique, *path reductions*, suppresses traceroutes that are likely to follow redundant paths through the ISP network. 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, and we show that the savings do not come at a high cost in terms of accuracy. We also describe a new solution to the *alias resolution* problem of clustering the interface IP addresses listed in a traceroute into their corresponding 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, Level3, Sprint, Telstra, Tiscali (Europe), Verio, and VSNL (India) – by using over 750 publicly available traceroute sources as measurement vantage points. These maps are summarized in the paper.

Three ISPs, out 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 present in BGP routing tables. Our maps reveal more complete ISP topologies compared to earlier efforts; we find roughly seven times more routers and links in our area of focus than Skitter [6].

As a second contribution, we examine several properties of the maps that are both of interest to researchers and likely to be useful for generating synthetic Internet maps. We report new results for the distribution of of POP sizes and the number of times that an ISP connects with other networks. Both distributions have significant tails. We also characterize the distribution of router outdegree, repeating some of the analysis in [7] with richer data.

Finally, as one goal of our work and part of our ongoing validation effort, we are publicly releasing the ISP network maps inferred from our measurements. We are also making the entire raw measurement data 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 at [20].

The rest of this paper is organized as follows. In Sections 2 and 3, we describe our approach and the mapping techniques respectively. The implementation of our mapping engine, Rocket-

SIGCOMM'02, August 19-23, 2002, Pittsburgh, Pennsylvania, USA.

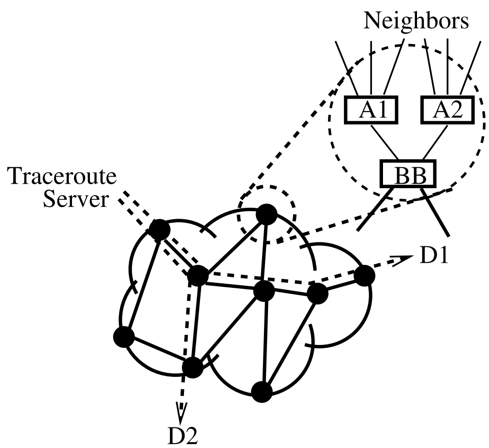


Figure 1: ISP networks are composed of POPs and backbones. Solid dots inside the cloud represent POPs. A POP consists of backbone and access routers (inset). Each traceroute across the ISP discovers the path from the source to the destination.

fuel, is described in Section 4. We present sample ISP maps in Section 5. In Section 6, we evaluate our maps for completeness, and our techniques for their measurement efficiency and accuracy. We analyze properties of the inferred maps in Section 7, present related work in Section 8, and conclude in Section 9.

2. 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 that we must address.

An ISP network is composed of multiple points of presence or POPs, as shown in Figure 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 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 along with the IP-level interconnections between them. This constitutes the interior routing region of the ISP, plus information about its boundaries. ISPs are usually associated with their BGP autonomous system numbers (ASNs). The map we collect does not precisely correspond to the address space advertised by the AS associated with an ISP. Our maps, as defined above, exclude portions of the address space that represent non-BGP speaking neighboring networks, or consumer broadband or dialup access, but they do include the boundary with the neighbors. In the paper, we use ISP names and their AS numbers interchangeably, unless the distinction is important.

Like earlier Internet mapping efforts [3, 6, 8], we discover ISP maps using traceroutes[1]. This process is illustrated in Figure 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 then merged to obtain 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 that we face is to build accurate ISP maps using few measurements. We cannot burden public traceroute servers with excessive load, so the number of traceroutes we can collect from each server is limited. 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 minutes,[2] the brute-force 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.

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 approach trades accuracy for efficiency, though we will see as part of the evaluation of these techniques that the loss of accuracy is much smaller than the gain in efficiency. That is, we make a worthwhile engineering tradeoff.

Even after the connectivity information has been obtained through traceroutes, two difficulties remain. First, each traceroute consists of 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. To address this problem, we develop a new, pair-wise test for aliases that uses a variety of router identification hints such as the IP identifier, rate-limiting, and TTL values.

Second, to be able to analyze the various structural properties of the collected maps, we need to identify the geographical location of the router and its role in the topology. Following the success of recent geographical mapping work [14], we leverage location hints that are typically embedded in the DNS name to extract the backbone and the POPs from the ISP map.

3. MAPPING TECHNIQUES

In this section, we present our mapping techniques. They are divided into three categories: selecting measurements, resolving IP addresses aliases, and identifying ISP routers, their role, and geographical information from the traceroute output.

3.1 Selecting Measurements

Based on two observations, we use two classes of techniques to reduce the required number of measurements. First, only traceroutes expected to transit the ISP need to be taken. We use a technique called *directed probing* that employs 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 needs to be taken when two traceroutes enter and leave the ISP network at the same point. We use a set of techniques called *path reductions* to identify redundant traceroutes.

---

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

Figure 2: A 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.

3.1.1 *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 transited the ISP being mapped. Since these tables are not available, we use RouteViews as an approximation [13]. It provides BGP views from 60 vantage points.

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 destinations within 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 Figure 2 to map AS number 7.

- 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.

- 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*. A traceroute from a server in AS 11 to $1.2.3.0/24$ is an up/down trace in Figure 2.

Directed probing capitalizes on the 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. In our evaluation section, we estimate the level of both these types of errors.

3.1.2 *Path Reductions*

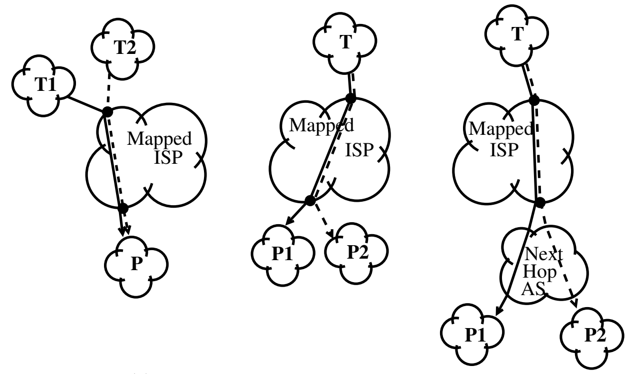
Not all the traceroute probes identified by directed probing will take unique paths inside the ISP. The number of measurements required can be reduced further by identifying probes that are likely to have identical paths inside the ISP. We list three different techniques here that exploit common properties of IP routing to cut down on redundant measurements. Although described separately, these techniques compose to bring about an even greater reduction in the number of required measurements.

Ingress Reduction. The path taken by a packet through a network is usually destination-specific. 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 Figure 3a. Since the traceroute from T2 to the destination would be redundant with the traceroute from T1, only one is needed. This redundancy can also be exploited to balance load between traceroute servers.

Egress Reduction. Similarly, traces from the same ingress to any prefix behind the same egress router should traverse the same path. Such traces are redundant, so only one needs to be collected. This is illustrated in Figure 3b.

Next-hop AS Reduction. The path through an ISP usually depends only on the next-hop AS, not on the specific destination prefix. This means that only one trace from ingress router to next-hop AS is likely to be valuable, as illustrated in Figure 3c. Unlike egress reduction, Next-hop AS reduction does not assume that there is only one egress per destination: the next-hop AS may peer in several places, while there is likely only one egress for a customer's prefix.

Ingress and egress predictions remove likely duplicates so that more valuable traces can be taken instead without sacrificing fidelity. If we find that our ingress-router prediction was incorrect, we repeat the trace using other servers that share the predicted ingress.



(a) **(b)** **(c)**

Figure 3: Path reductions. (a) Only one traceroute needs to be taken per destination when two servers (T's) share an ingress. (b) Only one trace needs to be taken when two dependent prefixes (P's) share an egress router. (c) Only one trace needs to be taken if two prefixes have the same next-hop AS number.

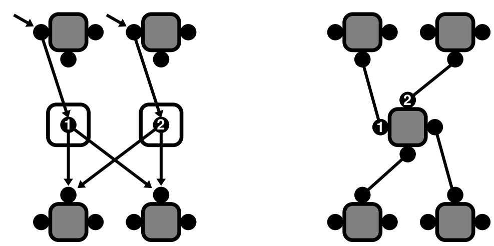


Figure 4: Alias resolution. Boxes represent routers and circles represent interfaces. Traceroute lists input interface addresses from paths (left). Alias resolution clusters interfaces into routers to reveal the true topology. Interfaces ❶ and ❷ are aliases (right).

3.2 Alias Resolution

Traceroute lists the source addresses of the "Time exceeded" ICMP messages; these addresses represent the interfaces 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 Figure 4. If the different addresses that represent the same router cannot be resolved, we get a different topology with more routers and links than the real one.

The standard technique for alias resolution was developed as part the Mercator project [8]. It detects aliases by sending traceroute-like probe (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. Mercator's technique is efficient in that it requires only one message to each IP address, but we found that it missed many aliases.

Our approach to alias resolution combines several techniques that identify peculiar similarities between responses to packets sent to different IP addresses. We include Mercator's IP address-based method, which detects an alias when both responses have the same source address. We compare the TTLs in responses to add confidence to an alias match, as well as to choose candidate address pairs to test, although comparing TTLs is not accurate by itself. We test for ICMP rate limiting, as described below.[3] However, none of these techniques were as successful as comparing the IP identifier field of the responses.

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 sending a packet. This implies that packets sent consecutively will have consecutive IP identifiers.[4]

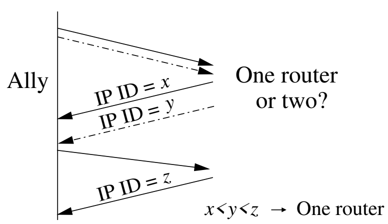


Figure 5: Alias resolution by IP identifiers. A solid arrow represents messages to and from one IP, the dotted arrow the other.

The procedure for resolving aliases by IP identifier is shown in Figure 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 sends a third packet to the address that responded first. If $x < y < z$, and $z - x$ 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 packet is not sent. This establishes a range: in-order IP identifiers suggest a single counter, which implies that the addresses are likely aliases. This three-packet approach compensates for routers that increment the IP id counter at varying speeds, and reduces the likelihood of a false positive.

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.

There is a small probability that two response packets will have nearby identifiers, without the IP addresses actually being aliases. To remove false positives, we repeat the alias resolution test on unverified aliases at a later time.

3.3 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 address space advertised by the AS for three reasons. First, routers of non-BGP speaking neighbors are often numbered from the AS's 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 ISPs domain (att.net, sprintlink.net, etc.). In some cases, the directly connected neighboring domain routers have a special naming convention that helps locate 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 address space. Again, DNS names help to identify the network edge correctly if they are assigned correctly. 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 address space. We resort to the 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 would be contiguous in the traceroute output.

One of our goals was to understand the structure of ISP maps, which includes their backbone and POPs. To do this we identify the role of each router as well as its location. We again use the information embedded in the DNS names for this purpose. Most ISPs have a naming convention for their routers. 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 we gather. For some ISPs, we started with city codes from the database in [14]. Some routers have no DNS names or their names lack location information. We infer the location of such routers from that of its neighbors.

4. ROCKETFUEL

In this section, we describe Rocketfuel, our mapping engine that infers maps using the above techniques. The architecture of Rocketfuel is shown in Figure 6. A PostgreSQL database stores all information in the 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 enables us to run SQL queries for simple questions, and integrate new analysis modules easily.

We used 294 public traceroute servers listed by the traceroute.org Web page [9], representing 784 vantage points all across the world. One traceroute server can potentially generate traceroutes from many routers in the same autonomous system. For example, `oxide.sprintlink.net` can generate traceroutes from 30 different van-

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

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

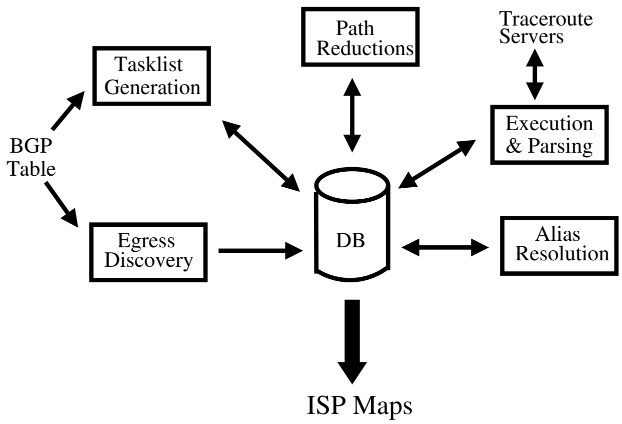


Figure 6: Architecture of Rocketfuel. The Database becomes the inter-process communication substrate.

tage points. 277 of the public traceroute servers only support one source. The BGP tables are taken from RouteViews [13].

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

The tasklist generation module uses BGP tables to generate a list of directed probes. The dependent prefixes in the directed probes are replaced with their (possibly multiple) egresses, and the duplicates are removed. This enables us to trace just to the egresses, and not beyond.

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 perform the reductions. For example, past traceroutes would tell us which ingress is used by a vantage point. After a traceroute is taken, this module also checks whether the predicted ingress or egress was used. If so, the job is complete and can be taken off the list. Otherwise, another vantage point that is likely to take that path is tried.

The execution engine handles the complexities of using publicly available traceroute servers: load-limiting, load-balancing, and different formats of traceroute output. Load distribution across destinations is achieved by randomizing the job list, which is done by sorting the MD5 hash [19] 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 hot-spots.

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 mark-up like headers, tables and graphics.

Alias resolution using the IP identifier technique in Section 3.2 requires some engineering to keep from testing every pair of IP addresses. We reduce the search space with three simple 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 IPs whose replies have nearby return TTLs may also be aliases. IPs 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. 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 either as aliases, not aliases, or unresponsive.

5. ISP MAPS

We ran Rocketfuel to map ten diverse ISPs. 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 at [20].

5.1 Summary Information

The names and aggregate statistics for all ten mapped ISPs are shown in Table 1. We see a large range in the sizes of the ISPs, with the biggest networks, AT&T, Sprint, and Verio depending on the metric, 100 times larger than the smallest networks.

5.2 Backbones

It is evident that the style of backbone design varies widely between ISPs. Figure 7 shows three sample backbones overlaid on a map of the United States [23]. We see that the 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 USA, all in major cities, and well connected to each other, implying that their smaller city customers are backhauled into these major hubs. Most major providers still have the AT&T type network, and are in various stages of transition to this newer design [4]. Level3 represents yet another paradigm in backbone design. Its highly connected backbone is most likely the result of using a circuit technology, such as MPLS, ATM or frame relay PVCs, to tunnel between POPs.

5.3 POPs

Unlike the backbone designs, we found POP designs to be relatively similar. 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. One cannot differentiate these scenarios from point-to-point connections using traceroute.

As anexample of a common pattern, Figure 8 shows our map of Sprint's POP in Springfield, IL. 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. 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.

AS	Name	ISP		with customer & peer		POPs
		Routers	Links	Routers	Links	
1221	Telstra (Australia)	355	700	2,796	3,000	61
1239	Sprintlink (US)	547	1,600	8,355	9,500	43
1755	Ebone (Europe)	163	300	596	500	25
2914	Verio (US)	1,018	2,300	7,336	6,800	121
3257	Tiscali (Europe)	276	400	865	700	50
3356	Level3 (US)	624	5,300	3,446	6,700	52
3967	Exodus (US)	338	800	900	1,100	23
4755	VSNL (India)	11	12	121	69	10
6461	Abovenet (US)	367	1,000	2,259	1,400	21
7018	AT&T (US)	733	2,300	10,214	12,500	108

Table 1: The number of routers, links, and POPs for all ten ISPs. ISP routers include backbone and gateway routers. With customer and peer routers adds directly connected customer access and peer routers. Links include only interconnections between these sets of routers, and are rounded to the nearest hundreds. POPs are identified by distinct location tags in the ISP's naming convention.

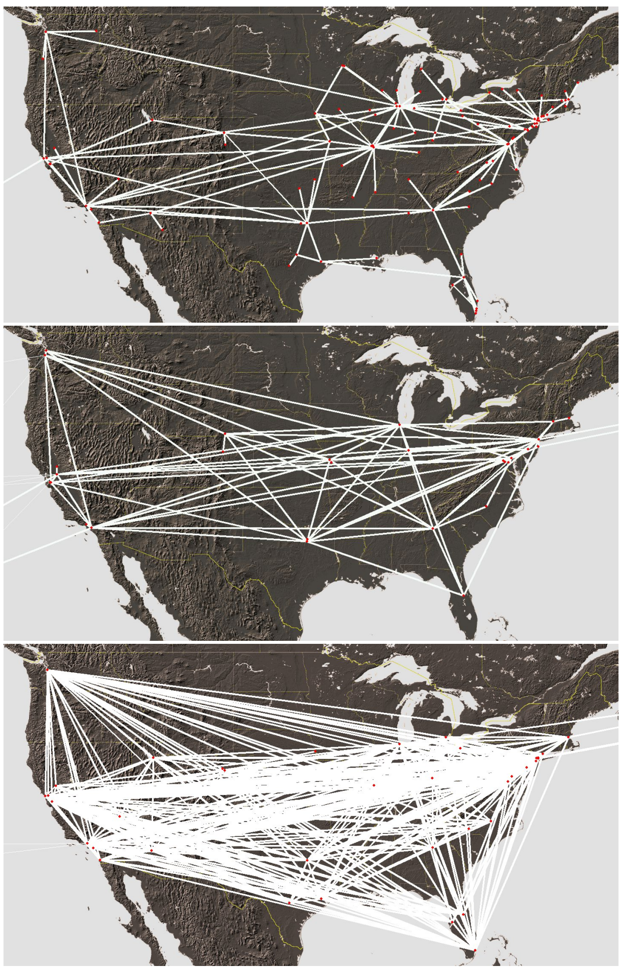


Figure 7: Backbone topologies of AT&T (top), Sprint (middle), and Level 3 (bottom). Multiple links may be present between two cities; only one link is shown for clarity. Shaded relief background image ©1995 Ray Sterner, Johns Hopkins University Applied Physics Laboratory, used with permission.

6. EVALUATION

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.

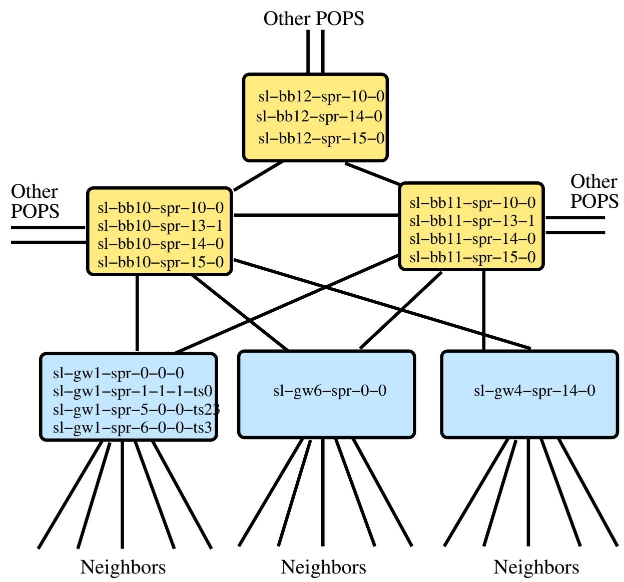


Figure 8: A sample POP topology from Sprint in Springfield, Illinois. The names are prefixes of the full names, without sprintlink.net. Most POPs in Sprint are larger and too complex to show, but retain the same design.

6.1 Completeness

We use four independent tests to estimate the accuracy and completeness of our maps. First, we ask the ISPs we mapped to help with validation. Second, we devise a new technique to estimate the completeness of an ISP map using IP address coverage. Third, we compare the BGP peerings we found to those present at RouteViews. Finally, we compare our maps with those obtained by Skitter [6], an on-going Internet mapping effort at CAIDA.

6.1.1 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 POP?* 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?* 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 while the traceroute was being taken.

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 three 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, optimistic deployment projections, and show parts of partner networks managed by other ISPs.

6.1.2 IP address space

As another completeness estimate, we searched prefixes of the ISP's address space for additional responsive IP addresses. New routers found by scanning 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 to search for new routers. 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. New routers are those IP addresses that both respond to ping and have names that follow the ISP's router naming convention, though they may not participate in forwarding. Prefixes were chosen to make sure that both backbone and access routers were represented.

Table 2 shows the estimated percentage coverage for each ISP. This percentage 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.

6.1.3 Comparison with RouteViews

Another estimate for completeness is the number of BGP adjacencies seen in our maps compared to the number observed in the BGP tables from RouteViews [13]. 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.

Figure 9 compares the number of adjacencies seen by Rocketfuel and RouteViews. The worst case for Rocketfuel is Sprint (1239), where we still find more than 70% of the neighbors. It is interesting that Rocketfuel discovers neighbors that are not present in the BGP table, a result consistent with [5]. We studied the adjacencies found by both approaches, and found that the BGP tables find more small (low degree in the AS-graph) neighbors, while Rocketfuel finds more large neighbors.

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%

Table 2: Estimate of Rocketfuel's coverage of router-like named IP addresses. Aliases of known routers are not counted. "n/a" implies that the ISP's naming convention doesn't differentiate between backbone and access routers.

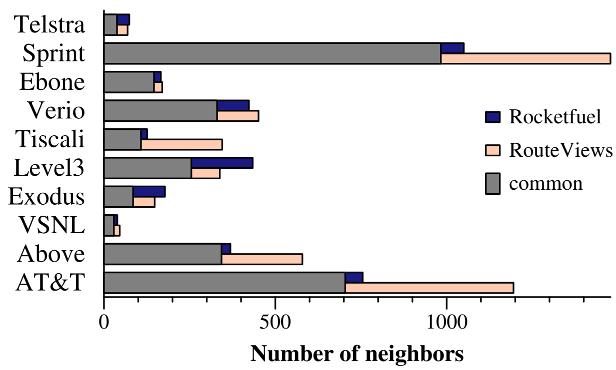


Figure 9: Comparison between BGP adjacencies seen in our maps and those seen in the BGP tables from RouteViews.

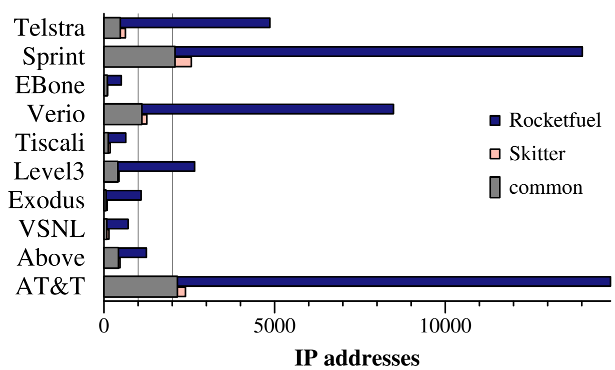


Figure 10: Comparison between Rocketfuel and Skitter for each ISP.

6.1.4 Comparison with Skitter

Skitter is a traceroute-based mapping project run by CAIDA [6]. We analyze Skitter data that was collected on 11-27-01 and 11-28-01. We compare the IP addresses, routers after alias resolution, and links seen by Skitter and Rocketfuel for each mapped AS. We also count the number of routers and links seen in only one dataset. The IP address statistics are presented for each AS in Figure 10 and all three statistics are summarized in Table 3.

Rocketfuel finds roughly seven times as many links, IPs 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 (Skitter was 11/01, and Rocketfuel was 1/02), most of it 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.

6.2 Impact of Reductions

This section evaluates the directed probing and path reductions described in Section 3. We evaluate these techniques for both the extent of reduction in measurements that they bring, and potential loss of accuracy that they cause. Most of the results presented here are aggregated over all the ten ISP's we map; individual results were largely similar.

6.2.1 Directed Probing

We consider three aspects of directed probing: the fraction of traces it is able to prune; the amount of pruned traces that would have transited the ISP and should have been kept; and the traces that should not have been taken because they did not in fact transit the ISP.

The effectiveness of directed probing is shown in Table 4. The brute-force search from all vantage points to all BGP-advertised prefixes plus the broken down ISP prefixes (/24s) requires 90-150 million traceroutes. With directed probing only between 1-8% of these traces are required.

To estimate how many useful traces, which would have traversed the ISP, were pruned, we ran an experiment using Skitter data. Assuming that the only vantage points available to us were those used by Skitter, we calculate the traces that would have been selected by directed probing. Using these and the Skitter traces, which are collected through brute-force mapping, we calculate the fraction of Skitter traces that traversed the ISP but were not identified by directed probing. We find that this fraction of useful but pruned traces varies by ISP from 0.1 to 7%. It is low for non-US ISPs like VSNL (4755) and Tiscali (3257), and high for the big US ISPs like AT&T and Sprint. This variation can be attributed to the difference in the likelihood of a trace from a vantage point to a randomly selected destination traversing the ISP. Even when the fraction of useful traces is 7%, it means that in absence of extra information such as BGP tables of the traceroute server itself, we would have to carry out 100 extra measurements to get 7 potentially useful ones. We did not explore how many of these potentially useful traces were actually useful in that they yielded paths not already seen by the chosen traces.

To determine how many traces we took that were unnecessary, we were able to tally directly from our measurement database. Of all the traces we took, roughly 6% did not transit the ISP.

These numbers are very encouraging because taken together they mean that not only does directed probing help cut the number of traces to 1-8%, but that there is very little useful work in what is pruned out, and very little useless work in what is done.

6.2.2 Ingress Reduction

In this section, we evaluate ingress reduction for its effectiveness in discarding unnecessary traces. Overall, ingress reduction kept only 12% 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.

The number of vantage points that share an ingress is given in Figure 11. 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 a large number of 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.

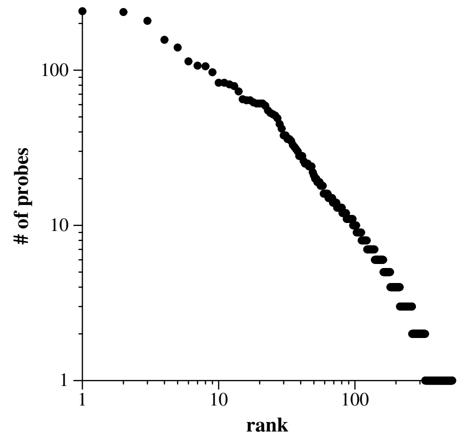


Figure 11: The number of vantage points that share an ingress, by rank, aggregated across ASes. 232 vantage points share the same ingress at left, while 247 vantage points have unique ingresses.

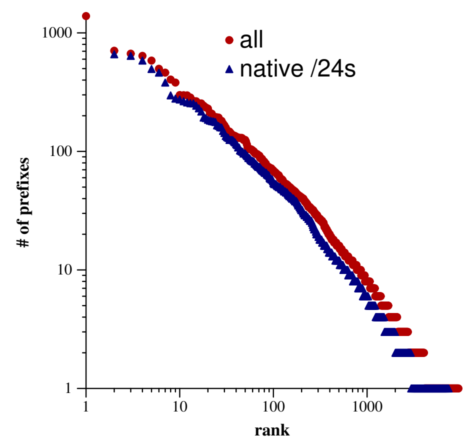


Figure 12: The number of dependent prefixes that share an egress, by rank, and aggregated across all ASes. */24s* refer to broken down ISP prefixes, and *all* also includes the dependent prefixes not originated by the ISP.

6.2.3 Egress Reduction

Overall, egress reduction kept only 18% of the dependent prefix traces chosen by directed probing. Figure 12 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, even for broken down /24s. This shows the necessity of breaking down large CIDR prefixes into smaller units; mapping using one address per prefix from BGP tables, as performed by existing BGP based mapping techniques, would miss out on many routers inside the ISP.

To test our hypothesis of /24 being a sufficiently fine granularity for egress discovery, we randomly chose 100 /24s (half of these were ISP prefixes) from the set of dependent prefixes and broke them down into /30s. We then traced to each /30 from a

	Links		IPs		Routers	
	Total	Unique	Total	Unique	Total	Unique
Rocketfuel	92723	84317	57528	49389	50787	45720
Skitter	12643	4237	9533	1392	7245	1323

Table 3: Comparison between Rocketfuel and Skitter aggregated over all 10 ISPs.

ASN	Name	Brute Force	Directed Probes	Remote Traceroutes	Egress Discovery
1221	Telstra (Australia)	105 M	1.5 M	20 K	20 K
1239	Sprintlink (US)	132 M	10.3 M	144 K	54 K
1755	Ebone (Europe)	91 M	15.3 M	16 K	1 K
2914	Verio (US)	118 M	1.6 M	241 K	36 K
3257	Tiscali (Europe)	92 M	0.2 M	6 K	2 K
3356	Level3 (US)	98 M	5.0 M	305 K	10 K
3967	Exodus (US)	91 M	1.2 M	24 K	1 K
4755	VSNL (India)	92 M	0.5 M	5 K	2 K
6461	Abovenet (US)	92 M	0.7 M	111 K	3 K
7018	AT&T (US)	152 M	4.5 M	150 K	80 K

Table 4: The effectiveness of directed probing, along with a summary of the number of traceroutes taken, including those from remote, public traceroute servers, and those taken locally to determine egresses for dependent prefixes.

local machine. The ratio of previously unseen egresses to the total discovered is an estimate of accuracy loss in exploring 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%. The wide range in fraction of newly discovered egresses suggests that our assumption, while valid for some ISPs (two of them had 0% 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. Techniques such as binary search can be used effectively for this purpose.

6.2.4 Next-Hop AS Reduction

Next-hop AS reduction reduces the number of up/down and insider traces (those that leave the ISP to enter another AS) to 5% of those chosen by directed probing. In Figure 13, we show the number of prefixes 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 directed probing, would otherwise traceroute to all 120,000 prefixes in the BGP table. Next-hop AS reduction allows insiders to instead trace to only 1,000 external destinations.

We also evaluate how commonly the early exit assumption underlying the reduction is violated. We used Verio as a test case by conducting 600K traces without the reduction. In all, the traces contained 2500 (ingress, next-hop AS) pairs. We found that in only 7% of the cases did an ingress see more than one egress when crossing over to the same AS. It is likely that different ISPs have different policies regarding early-exit routing, but nevertheless this result is encouraging.

6.2.5 Overall Impact

Our reductions are mostly orthogonal to each other and they compose to give multiplicative benefit. The overall impact of the reductions can be seen in Table 4, which shows the total number of traceroutes that we took 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 for ISPs varied between 0.3% (Level3) to 0.05% (VSNL).

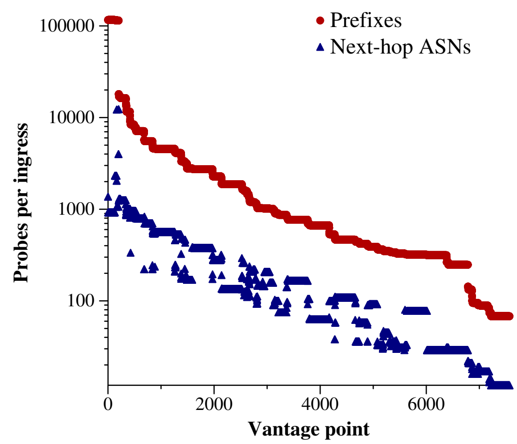


Figure 13: The number of prefixes and unique next-hop ASes for vantage points.

Our mapping techniques also scale with the number of vantage points. If we are given more vantage points, we would be able to generate better maps more quickly, but would not increase the number of traceroutes unnecessarily. Extra vantage points bring one of the two things to the table – speed or accuracy. The speed is improved when the new vantage point shares an ingress with an existing vantage point. Accuracy is improved if the new vantage point has a unique ingress to the ISP.

6.3 Alias Resolution

The effectiveness of both the IP address based approach and our new approach to alias resolution is shown in Table 5. The table shows how many aliases, which are additional IP addresses for the same router beyond the first, were found by each technique. Ally finds almost three times more aliases than the earlier address-based approach. Moreover, we found aliases resolved by Ally to be a superset of those resolved by an address-based technique. This means that using only Ally suffices for alias resolution.

As a test 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 iden-

Tool	Aliases Recognized
Mercator	2,656
Ally	7,423

Table 5: Ally's IP identifier-based technique finds almost three times as many aliases as an address-based technique.

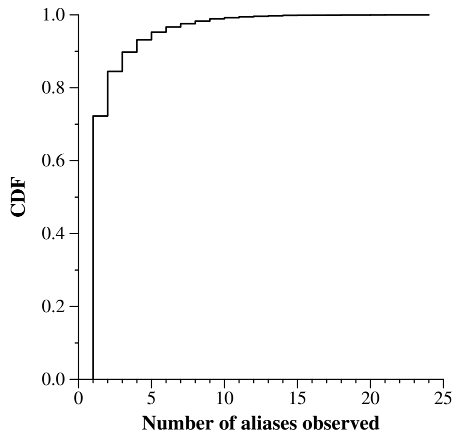


Figure 14: The number of aliases observed for routers within the mapped ISPs.

tifiers. 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. 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, 5 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 6,000 that never responded to our alias resolution queries.

We plan to investigate why there were 30 Sprint and 15 Ebone routers that were responsive, but were not completely and correctly resolved. Potential causes include implementation flaws in our alias resolution tool, routers that were temporarily unresponsive, considering stale DNS entries authoritative, and routers with multiple IP stacks (and thus two IP identifier counters).

Figure 14 plots a CDF of how many aliases we saw for routers within the ISPs we mapped. It shows that we saw only one IP for 70% of the routers, and 2 IPs for another 10% of them. 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 routers have since it is likely that we do not see all possible IPs for a router.

7. ANALYSIS

To demonstrate the utility of our maps, we also include in this paper a preliminary analysis of their properties. We report on router outdegree distributions, repeating the analysis in [7] over our more detailed data, and present POP and peering statistics, not previously reported, as may be useful for synthetic topology generation.

7.1 POP sizes

The distribution of POP sizes is shown in Figure 15 for Level3, AT&T, and Sprint, both as a cumulative fraction of the total number of POPs (top) and the total number of routers (bottom). To determine POP size, we count only backbone and access routers, and exclude the customer and peer routers.

The distributions are all skewed, though their details differ, with AT&T having both the largest POPs and most skewed distribution, and Level3 having the least skew and variation in POP size. Most of the routers are present in ten largest POPs for all three networks, and all three ISPs have a significant number of small POPs. For example, more than 60% of Sprint POPs have fewer than 15 routers and these POPs account for less than 20% of all Sprint routers.

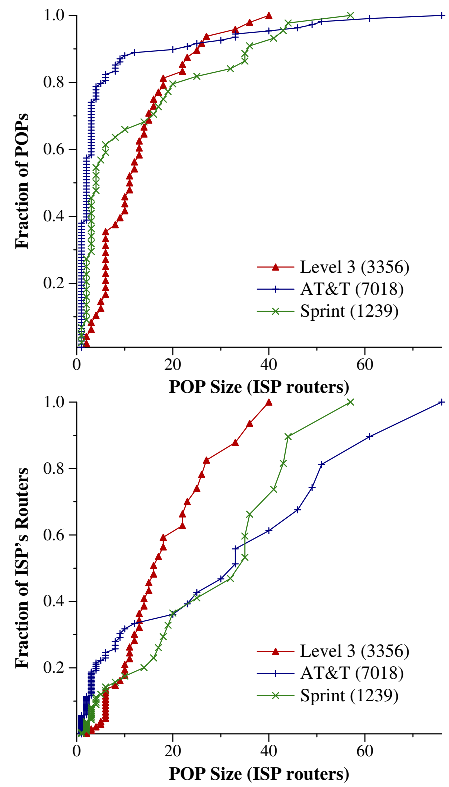


Figure 15: The distribution of POP size for AT&T, Sprint, and Level 3 by POPs (top) and by routers (bottom)

The largest POPs are Chicago and New York for AT&T, Fort Worth and Chicago for Sprint, and New York and San Jose for Level 3. The smallest POPs for these networks are in Europe, maintained by the ISPs for trans-continental connectivity, or smaller cities in the United States. Small POPs may be called by other names within the ISP; we do not distinguish between backbone network nodes, data centers or private peering points.

7.2 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 complementary cumulative distribution function (ccdf) of router is shown in the aggregate over all ISPs in Figure 16 and for individual ISPs in Figure 18 (at the end of the paper). 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.

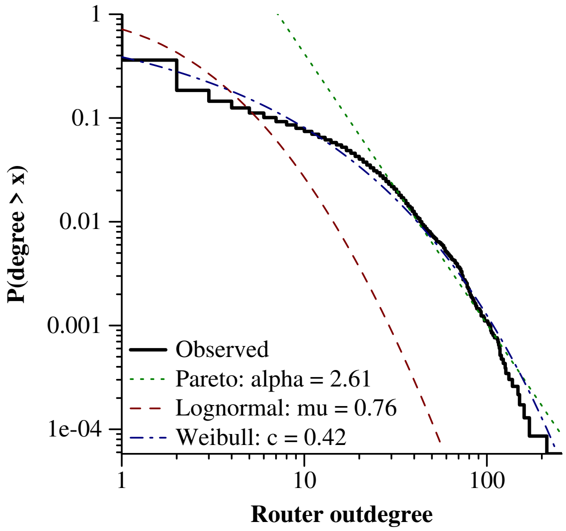


Figure 16: Router outdegree ccdf. The Pareto fit is only applied to the tail.

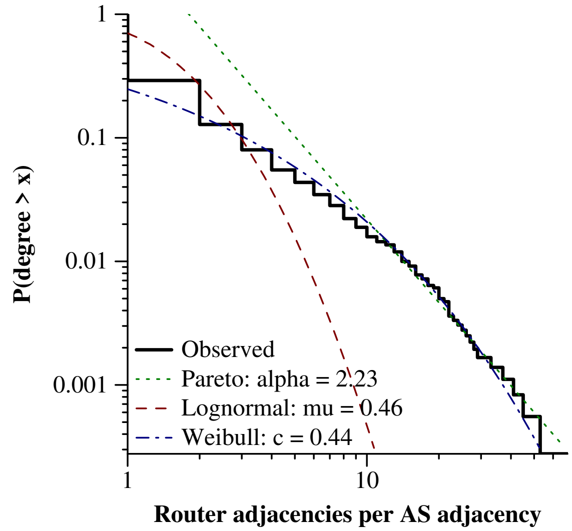


Figure 17: A ccdf of the number of router-level adjacencies seen for each AS-level adjacency. AS adjacencies include both peerings with other ISPs and peerings with customers that manage their own AS.

We observe that, unlike the measured degree in AS graphs, router outdegree has a small range in our data; it covers only two orders of magnitude over the ten ISPs. This is despite the fact that, while physical size and power constraints naturally limit the underlying router outdegree, our data can include undetected layer two switches that would inflate the observed router outdegree, perhaps substantially.

The ccdfs of router outdegree from different ISPs are fit best by different distributions: some fit Weibull, a few fit the simpler lognormal, and most have tails that are consistent with the Pareto fit. It appears that Weibull often fits both tail and body of the distribution. Although these distributions have significant tails, the α parameter is high for a heavy-tailed distribution.

7.3 Peering Structure

Since our maps are collected using traceroutes that enter and exit the mapped ISP from various ingresses and egresses, they give us the unique opportunity to study the peering structure between ASes. From paths in BGP tables, one can only infer from an adjacency between two ASes in the AS-level graph that the two ASes connect somewhere. However, using Rocketfuel topologies, we can infer where and in how many places these two ASes exchange traffic. For example, while BGP tables expose the fact that Sprint and AT&T peer, they do not show the different locations at which the two ISPs exchange traffic. By *peering structure*, we refer to this important level of detail not present in the AS-level graphs collected from BGP tables.

We summarize the peering structure by showing the number of locations at which the mapped ISP exchanges traffic with each other AS. 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. Figure 17 plots this ccdf, aggregated over the mapped ISPs, while Figure 19 (at the end of the paper) shows plots for individual ISPs. Both figures include Pareto, lognormal and Weibull fits calculated as before.

We see that the data is highly skewed for all of 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 [24].

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

8. RELATED WORK

An early attempt [15] to infer a router-level map of the Internet started with a list of 5000 destinations, and used traceroutes from a single network node. Mercator [8] is also a map collection tool run from a single host. 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 [3] use BGP tables to find destination prefixes. They source their traceroute from a single machine, but improve their coverage by using tunnels to some other machines on the network, similar in effect to source-routing. Skitter [6] uses BGP tables and a database of Web servers. Skitter monitors probe the network from about 20 different locations worldwide. Our mapping technique differs fundamentally from all of these efforts. Instead of trying to collect the router level map of the whole Internet, we do focused probing of an ISP to recover its map. The result is an ISP map that is more complete than that obtained by other mapping efforts.

In [1], the authors analyze the marginal utility of adding vantage points and destinations to discover the Internet backbone topology. Our work is similar in that we also try to minimize the number of measurements needed, but we use routing knowledge instead of reducing the number of vantage points.

9. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented new techniques for mapping the router-level topology of focused portions of the Internet, such as an ISP, using only end-to-end measurements. We have shown that routing information can be exploited in several ways to select only those traceroutes that are expected to be useful. The result is that we are able to reduce the mapping workload by three orders of magnitude compared to a brute-force all-to-all approach while losing little in the way of accuracy. This in turn enabled us to use the more than 750 public traceroute servers for our measurement sources, providing us with many more vantage points than other mapping efforts. We have also presented a new alias resolution technique that discovered three times more aliases than the current approach based on return addresses. This increases 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 at [20]. To evaluate the maps, we compared them with i) the true map as understood by the ISP operators; ii) the total number of routers found by scanning sampled subnets; iii) the peerings known to exist from BGP tables; and iv) previous 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.

The above notwithstanding, it is clear to us that this work has only scratched the surface of ISP map construction and analysis. It can be 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 though samples are small. Second, we can extract further properties from the traceroutes that can be used to annotate the maps and improve their usefulness. 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. Finally, it seems likely that we can improve our basic techniques further, perhaps substantially, leading to mapping that is both more efficient and more accurate.

10. ACKNOWLEDGEMENTS

We thank Tom Anderson for his guidance. We are grateful to the administrators of the traceroute servers whose public service enabled our work, and operators who provided feedback on the quality of our maps. We thank Lakshminarayanan Subramanian for scripts from [14], and CAIDA for skitter data. Ramesh Govindan provided independent verification of our alias resolution technique, and helpful mapping advice. We also thank Steve Bellovin, Christophe Diot, and Randy Bush for early insights into ISP backbone and POP topologies. Allen Downey provided lognormal distribution analysis tools and guidance. Walter Willinger provided helpful feedback on the implications of our analysis results.

This work was supported by DARPA under grant no. F30602-00-2-0565.

11. REFERENCES

[1] P. Barford, A. Bestavros, J. Byers, and M. Crovella. On the Marginal Utility of Network Topology Measurements. In *ACM SIGCOMM Internet Measurement Workshop*, November 2001.

[2] A. Basu and J. Riecke. Stability issues in OSPF routing. In *ACM SIGCOMM*, August 2001.

[3] H. Burch and B. Cheswick. Mapping the Internet. *IEEE Computer*, 32(4):97–98, 102, 1999.

[4] R. Bush. Private communication, November 2001.

[5] H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. On Inferring AS-Level Connectivity from BGP Routing Tables. Technical Report UM-CSE-TR-454-02, 2002. http://topology.eecs.umich.edu/.

[6] k. claffy, T. E. Monk, and D. McRobb. Internet tomography. In *Nature*, January 1999.

[7] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *ACM SIGCOMM*, 1999.

[8] R. Govindan and H. Tangmunarunkit. Heuristics for Internet map discovery. In *IEEE INFOCOM*, 2000.

[9] T. Kernen. traceroute.org. http://www.traceroute.org.

[10] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. Delayed Internet Routing Convergence. In *ACM SIGCOMM*, September 2000.

[11] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker. Controlling high-bandwidth aggregates in the network (extended version). http://www.aciri.org/pushback/, July 2001.

[12] A. Medina, I. Matta, and J. Byers. BRITE: A flexible generator of Internet toplogies. Technical Report BU-CS-TR-2000-005, Boston University, 2000.

[13] D. Meyer. RouteViews Project. http://www.routeviews.org.

[14] V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for Internet hosts. In *ACM SIGCOMM*, August 2001.

[15] J. Pansiot and D. Grad. On routes and multicast trees in the Internet. In *ACM SIGCOMM Computer Communication Review*, pages 41–50, 1997.

[16] K. Park and H. Lee. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets. In *ACM SIGCOMM*, August 2001.

[17] G. Philips, S. Shenker, and H. Tangmunarunkit. Scaling of multicast trees: Comments on the Chuang-Sirbu scaling law. In *ACM SIGCOMM*, August 1999.

[18] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin. On characterizing network topologies and analyzing their impact on protocol design. Technical Report CS-00-731, USC, 2000.

[19] R. Rivest. The MD5 message-digest algorithm, 1992. Networking Working Group Requests for Comment, MIT Laboratory for Computer Science and RSA Data Security, Inc., RFC-1321.

[20] Rocketfuel maps and data. http://www.cs.washington.edu/research/networking/rocketfuel/.

[21] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical network support for IP traceback. In *ACM SIGCOMM*, August 2000.

[22] 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 *ACM SIGCOMM*, August 2001.

[23] R. Sterner. Color landform atlas of the United States. http://fermi.jhuapl.edu/states/.

[24] H. Tangmunarunkit, J. Doyle, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. Does AS Size Determine Degree in AS Topology? In *ACM Computer Communication Review*. October 2001.

[25] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. Network topology generators: Degree-based vs structural. In *ACM SIGCOMM*, August 2002.

[26] E. W. Zegura, K. Calvert, and S. Bhattacharjee. How to model an internetwork. In *Proceedings of IEEE INFOCOM*, 1996.

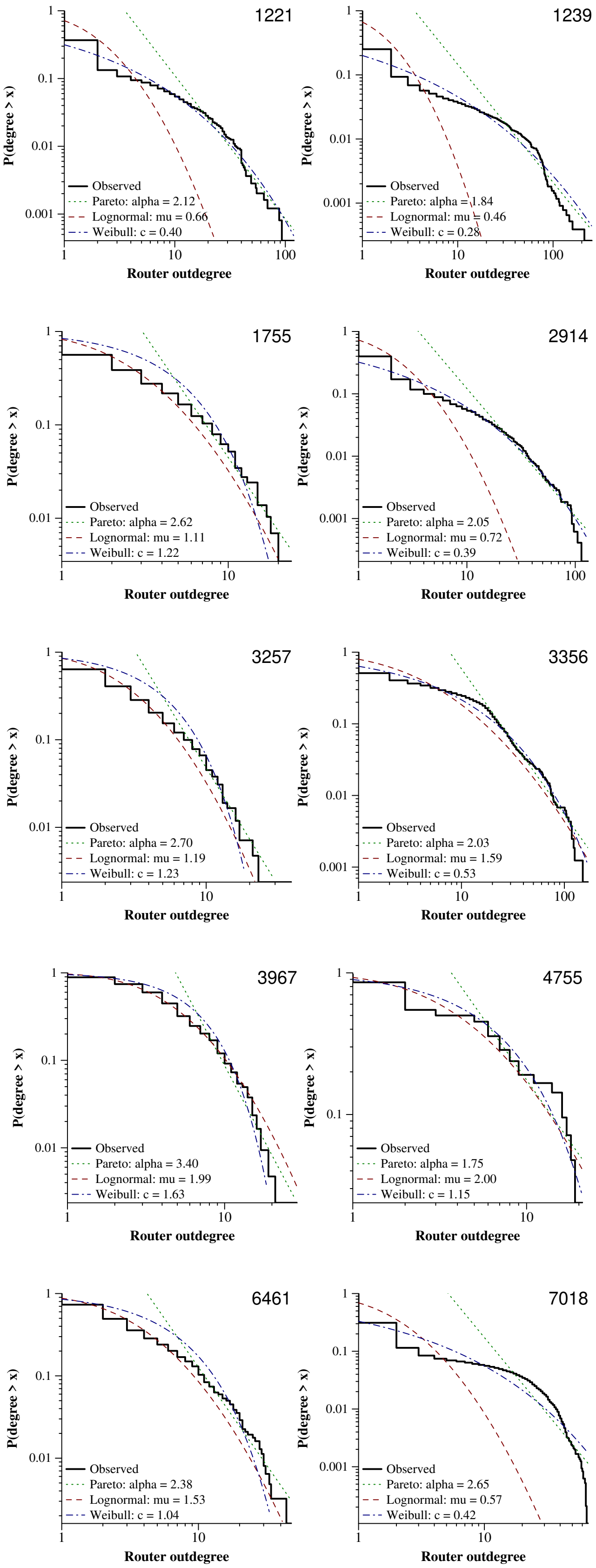


Figure 18: Router outdegree ccdf by ISP.

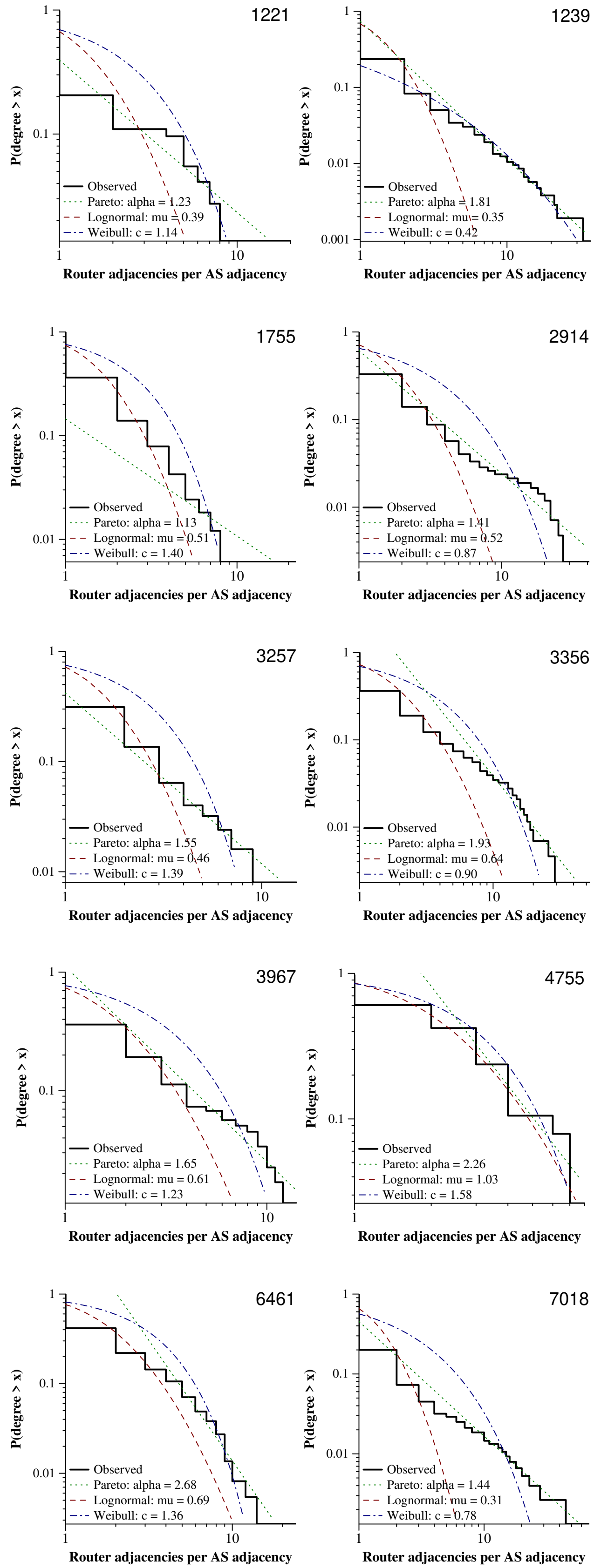


Figure 19: A ccdf of the number of router-level adjacencies seen for each AS-level peering, broken down by ISP.