Locating BGP Missing Routes Using Multiple Perspectives

Di-Fa Chang Ramesh Govindan John Heidemann
USC/Information Sciences Institute, Technical Report ISI-TR-2004-588

Abstract— **There have been many studies on measuring and interpreting interdomain routing dynamics. Most of them, however, are based on the approach of off-line and passive post-processing BGP routing updates. We propose a new methodology that uses real-time and active monitoring to troubleshoot various BGP routing anomalies. This paper focuses on a specific BGP routing problem — *missing routes* that occur when some ASes can reach a prefix while others can't. The idea is to periodically monitor the BGP routing status at multiple vantage points, like Route Views, and when a possible missing route event is detected issue traceroute queries from various looking glasses to learn of the packet-forwarding path status. By comparing previous and current packet-forwarding paths, we can have an idea of where the missing route event takes place. This paper examines the plausibility of this methodology and discusses preliminary experimental results.**

I. Introduction

In recent years researchers have observed and examined various problems in the inter-domain routing system. The causes of these routing problems can be understood through analytic modeling, *e.g.*, E-BGP and I-BGP route oscillation problems [1], [2], [3], [4], [5] and missing routes [6]. The dynamics of how these problems occur in Internet, however, are not well understood. For example, an ISP announces a prefix to the inter-domain routing system, but some ASes can observe the prefix while others can't. How does this *missing route* problem happen? And where does it take place? Currently, there is no automated mechanism for resolving missing routes. Nevertheless, such mechanisms are important for the originating ISP to maintain reliable global reachability to its address space.

The two questions in understanding and troubleshooting such an inter-domain routing problem are *how it happens* (assigning root cause) and *where it happens* (determining location). These questions are made challenging by the lack of information about the routing problem. This insufficiency of information results in a diversity of possible routing events that can cause the same routing problem. In general, finding the root cause is more difficult than locating the place it happens [7]. Previous research on correlating multiple BGP update streams suggests that, in today's BGP paths with average length of 4.7 peerings, only about 2 peerings can be ruled out as a suspect that causes a path change while others are undecidable [8]. Fundamentally, this arises because BGP is designed to abstract routing information in order to achieve scalability, hence the path information in BGP routing updates is too coarse to reasonably infer on where BGP path changes happen. Even if it were possible to *locate* the origin of a problem, routers support diverse mechanisms on routing configuration and different configurations may result in the same routing status, hence it is not possible to infer causality based on the change of routing status [8], [9].

This paper proposes a methodology to estimate the location of a routing event in AS peering topology. Our methodology mirrors the manual techniques that ISP operators use to diagnose their routing problems. First, we periodically monitor the BGP routing status of multiple vantage points, *e.g.*, Route Views, to detect possible missing route events by using a simple detection heuristic. The heuristic is not designed to identify every missing route event precisely, but to pick the prefixes (ranges of IP addresses) that may suffer a missing route problem for a significant period so that we can have enough time to diagnose the problem. Then we issue traceroute queries to various looking glasses to obtain current packet-forwarding paths to those suffering prefixes. By comparing the previous and current packet-forwarding paths, we can estimate the location of the problem.

We first describes the detection heuristic of missing route events in Section II. Section III describes the localization algorithms. Section IV examines whether current monitoring infrastructure of looking glasses can support this troubleshooting mechanism. And Section V provides a preliminary experimental result of our prototype of the localization algorithms.

II. Missing Route Events

This section answers the following questions:

- How can missing route events happen?
- How can we detect the missing route events?
- How frequently do the missing route events happen? If problems are rare, then operators can just ignore them.

A. How missing route events happen

Previous research has shown that prefixes originated from one ISP may not be seen by all ASes in the Internet [10], [6]. It was suggested that sometimes this is due to commercial strategies which are intentionally configured into routing policies and not a routing problem that we are interested in. What we are concerned is the "unintentional" missing route problem — for a period of time (minutes, hours, or days) an announced prefix cannot be reached by some ASes. The problem may take place in the origin AS (IBGP problems), AS peerings (EBGP problems), or the transit ASes (IBGP problems). We briefly describe the causes discovered by researchers and operators.

Table I lists the possible causes of missing-route events. We roughly divide the "unintentional" causes into two categories: misconfiguration and network instability. The causes of misconfiguration are the direct results of human errors, while network instability includes hardware and software problems. This classification is not mutually exclusive since some network instability may result from combination of misconfiguration and network failures. We first describe the types of misconfiguration which may happen in various places [6].

Conflicting RR Router ID: This is an IBGP misconfiguration

TABLE I. Causes of missing route problems. The codes for cause location are O for origin AS, P for EBGP peerings, and T for transit ASes.

Cause	Where	Description
RR-RID	OT	Router ID is duplicated.
RR-CID	OT	Cluster ID is duplicated.
IBGP-MESH	OT	The IBGP mesh is not full.
FILTER	P	Filter configuration is wrong.
FLAP	OT	Session resets due to layer 2 or hardware problems.
DAMPING	OTP	Route is falsely suppressed.

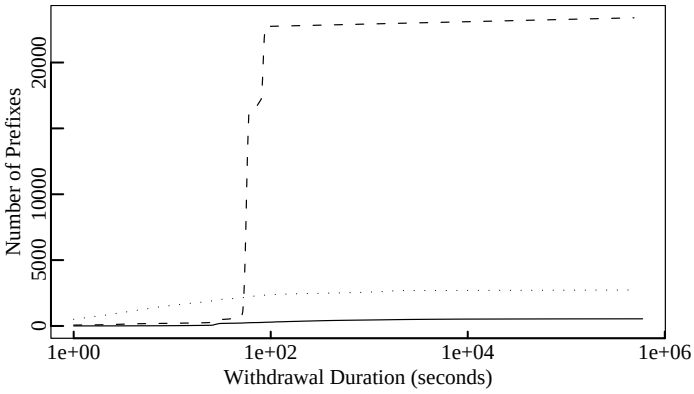

Fig. 1. Distribution of withdrawal duration for prefixes withdrawn by all observing vantage points (solid line), one vantage point (dashed), and some vantage points (dotted).

problem for route reflectors (RR). If router sees its own router ID in the Originator attribute in any received route announcement, it will reject that route. This mechanism is how a route reflector attempts to avoid routing loops. So, if operators set the routers' IDs by hand, it's possible that two routers have same ID which results in missing routes. This error can happen in origin AS or transit ASes.

Conflicting RR Cluster ID: This is also an IBGP misconfiguration problem for route reflectors. If router sees its own router ID in the Cluster-ID attribute in any received route announcement, it will reject that route. This mechanism is how a route reflector avoids redundant information. So, if Route Reflector Clients (RRC) don't peer with *all* the RR in the same cluster, missing routes can happen. This error can happen in origin AS or transit ASes.

Incomplete IBGP Mesh: BGP requires a full IBGP mesh. A lack of IBGP peering can result in incomplete route propagation *e.g.*, a route announcement received from a peer is not propagated to a downstream AS.

Filtering: The route is blocked by input or output filters. Most router implementations support various types of filter configuration, *e.g.*, filters based on prefixes, AS_PATH, and community attributes. The causes of filter misconfiguration can be cut-and-paste buffer problems, typos in configuration commands, unawareness of implicit filtering rules of router implementation, or confusion about complex policy rules set by operators themselves.

In addition to the above misconfiguration errors, there can be some network instability that causes some routes not propagated temporarily. Here are two examples.

BGP Session Resets: During a session reset, all routes going through that peering are withdrawn and re-announced. There are many events that can cause BGP session resets or peering flaps. For example, KEEPALIVE messages are lost due to some layer 2 problems, router reboots continually, rate limiting parameters are wrong, MTU is incorrectly set on links, PMTU discovery is disabled on routers, faulty MUXes, bad connectors, interoperability problems, PPP problems, satellite or radio problems, weather, etc.

Flap Damping: Route flap damping mechanism can result in convergence problem that suppresses the propagation of valid routes [11].

The duration of missing routes varies for different causes and different network configurations. We are more interested in the events that last for a significant period (at least tens of minutes) than those transient events (a few minutes). We next consider the detection mechanisms by passively monitoring BGP routing status.

B. How can we detect the missing route events

ISP operators usually make use of various BGP vantage points like Route Views and RIPE RIS to check whether their prefixes appear in global BGP tables. Thus, a simple heuristic to detect missing routes is that as long as there is at least one vantage point unable to reach a prefix while others can, then a missing route event is said to happen. This heuristic, however, will sensitively detect those reachability problems that are caused by transient session failures, which is not what we want. We want to restrict the detection criteria to pick the missing route events that last for a significant period, *e.g.*, 30 minutes.

From the BGP updates collected by Route Views from Jan. 1 to 31, 2004, we calculate the duration of each possible missing route event. Since it is impossible to know how many missing route events occurred within that period of time, we assume that each missing route affects only one prefix and we count the number of prefixes that experience the missing route problem. Figure 1 shows the average number of prefixes withdrawn at most t seconds per hour, that is, those prefixes are withdrawn from the BGP tables of some vantage points for at most t seconds. To understand whether the number of vantage points that observe the missing route events is correlated with the event duration, we classify the missing route events into three groups. In the figure, the solid line shows the number of prefixes (representing missing route events) withdrawn by all vantage points that ever observe them. The dashed line is for prefixes withdrawn by one vantage point while at least one other vantage point has routes to them. And the dotted line is for prefixes withdrawn by more than one vantage points when at least one other vantage point has routes to them.

For the dashed line, there are two spikes: one in 50–60 seconds and another in 78–90 seconds, which are roughly two and three times of the default MinRouteAdver timer, respectively. We find that the many single-vantage-point withdrawals occur simultaneously for prefixes from various origin ASes. This observation indicates that single-vantage-point withdrawals (dashed line) are likely caused by transient events on the

peering sessions near to vantage points. Also, since all-vantage-point withdrawals (solid line) are not missing route events according to the definition, it is only the some-vantage-point withdrawals (dotted line) that we are interested in.

Accordingly, our detection heuristic is to select the prefixes such that the routes to them are withdrawn by at least two vantage points. Intuitively these events likely take place in some transit peerings instead of the peerings adjacent to vantage points. Also, this event likely lasts long enough for us to localize it.

C. How frequently do the missing route events happen

Past research showed that up to 5% of Internet routing table was unreachable by some providers and 2,000 origin ASes not globally visible [10]. Figure 1 shows that there are, on the average, 3.5 prefixes per hour suffering the missing route problem according to our detection criteria, namely, at least two vantage points lose the routes to them for more than 30 minutes while other vantage points still have routes to them during that period of time.

III. Localization of Missing Route Events

This section is to answer these questions: *How can we localize the missing route events? What are the costs of these localization methods?*

The localization algorithm is to locate the suspect peerings that stop the route propagation. We first describe an ideal algorithm for this localization problem and why we can't use this algorithm in current stage. Then we propose other algorithms that we used in our experiments.

A. Prefix-Level Localization

The basic element for BGP policy configuration is prefix. Different prefixes from the same origin AS may have different routes and suffer from different routing events.

Figure 2 shows an example of missing route events. Prefix p_1 is originated by AS-d. Th vantage point in AS-x reaches p_1 via the path (x, a, b, c, d), and the vantage point in AS-y uses (y, b, c, d). Suppose an unknown error takes place in the peering (a, b) and results in a withdrawal of the path (x, a, b, c, d). Ideally, if we know that looking glasses l and m previously reached the prefix p_1 via the routes (l, r_a, r_b, r_c, r_d) and (m, r_b, r_c, r_d), respectively (where r_a is a router in AS-a), then we can use the following algorithm to locate the suspect peerings. We ask looking glasses l and m to traceroute to prefix p_1. If m can reach p_1 but l only traceroutes to r_a and stops, then the peering (a, b) is a suspect peering.

In order to use this algorithm, we need to know the route from any looking glass to any prefix. Assuming there are 140,000 prefixes in default-free BGP tables and we query a looking glass for the route to a prefix every 60 seconds, then it takes 97.22 days to obtain the routing snapshot for each looking glass. If we increase the query rate to one prefix per one second, we can obtain the routing snapshot within two days, but it will impose a huge computational burden on the looking glass. So, this algorithm is not a practical solution for its high cost of routing status acquisition. Thus, we propose other algorithms.

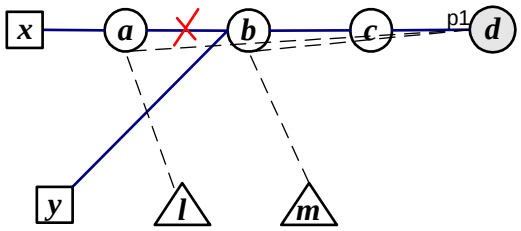

Fig. 2. Missing Route: vantage point x can't reach prefi x p_1 while y can, where a, b, c, d represent ASes, x, y are the ASes having BGP vantage points, and l, m are looking glasses.

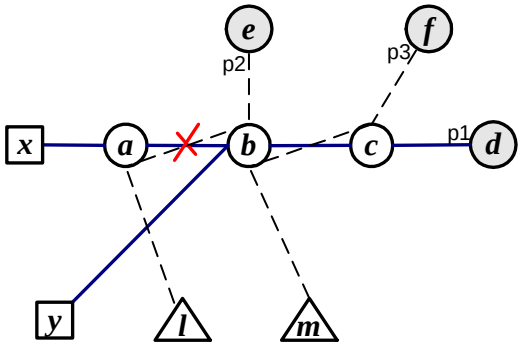

Fig. 3. Missing Route: vantage point x can't reach prefi x p_1 while y can.

B. Peering-level Localization

Figure 3 shows the same example of missing route events. But here we only know that looking glass l could reach prefix p_2 via the route (l, r_a, r_b, r_e) and m could reach p_3 via (m, r_b, r_c, r_f). Suppose the error that takes place in peering (a, b) is a peering failure (*e.g.*, BGP session reset) that causes withdrawal of the paths (a, b, c, d) for prefix p_1 and (a, b, e) for prefix p_2. Now, if we ask looking glass l to traceroute to p_2, l will answer that it stops at AS-a. Based on this answer, we can only say the peering (a, b) is a suspect peering *if the cause of this missing route event is a peering failure*. Consider otherwise: if the cause is a filter misconfiguration and affects only prefix p_1, then no matter whether looking glass l can reach p_2 we can't say anything about which peering is a suspect peering for this missing route event of p_1.

The cost of initial routing status acquisition for this algorithm is relatively small since we only need to know whether a looking glass can reach a specific peering. One way of obtaining that information is to ask each looking glass for its route to every AS. Assuming there are 14,000 ASes in the Internet, and we query a looking glass for the route to an AS every 60 seconds, then it takes 9.7 days to obtain the routing snapshot for each looking glass, which is acceptable.

C. Approximate Prefix-level Localization

The peering-level localization algorithm is not suitable for localizing missing routes caused by problems other than peering failures, so we suggest using prefix-level localization with less knowledge of initial routing status. The idea is to let each looking glass be in charge of a subset of all prefixes, hence the overhead of initial probing can be reduced. For example, if there

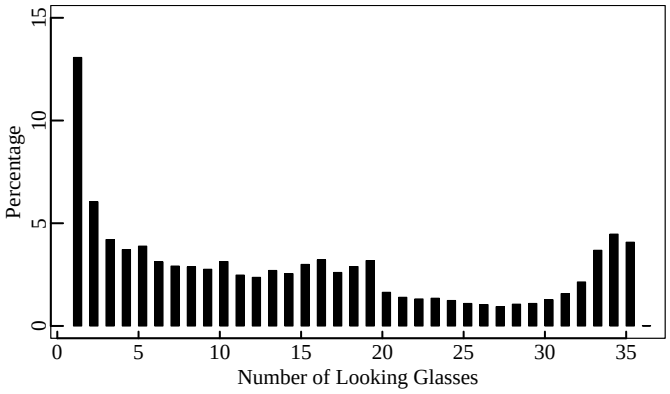

Fig. 4. Peering Visibility

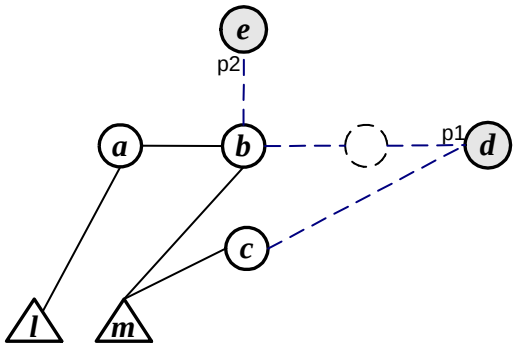

Fig. 5. Failed traceroutes to destinations in prefixes p_1 and p_2: $traceroute(l, p_1) = (l, r_a, r_b, *)$, $traceroute(l, p_2) = (l, r_a, r_b, *)$, $traceroute(m, p_1) = (m, r_c, *)$, $traceroute(m, p_2) = (m, r_b, *)$.

are 140,000 prefixes and 100 looking glasses, then each looking glass traceroutes to 5,000 prefixes one by one which takes 3.5 days. For each prefix, we have the initial routing status from 3–4 looking glasses. Based on this routing status, we can apply the prefix-level localization algorithm.

IV. Looking Glasses Coverage

This section describes the experimental results of probing initial routing status. We are interested in knowing how much of the topology is covered by a single looking glass since this extent of coverage affects what area of Internet our localization algorithms can work. Table II lists the looking glasses we used to estimate the coverage. The experiment is to make each looking glass issue traceroute probes to 16,575 ASes. The table shows that each looking glass discover 31–42 thousand links (in router level) and 4–7 thousand AS peerings. The coverage is quite small — about 12% of AS peerings are covered by each looking glass.

The last column in the table shows the marginal information provided by the i-th looking glass when the first i looking glasses are used. That is, when we include one more looking glass into our monitor infrastructure, we obtain less than 1% more coverage of the entire peering topology. This observation suggests that it is more feasible to localize routing problems occurred in Internet core than in the edge which consists of large portion of peering topology that are not tracerouted from looking glasses.

The localization algorithms require diverse routing knowledge from multiple looking glasses. That is, the more looking glasses can reach a peering, the more likely we can detect the routing problem occurred in that peering. Figure 4 shows the number of visible peerings that are visible to exactly L looking glasses. Only 17,303 (30.01%) of total 57,672 peerings are visible to our looking glasses. Among them, 13,997 peerings are visible to at least three looking glasses.

A. Mapping Routers to ASes

The traceroutes return the IP addresses of the routers in the packet forwarding paths to the specified destination. To know what AS peerings included in the packet forwarding path, we need to map the router addresses to AS numbers. Previous research proposed a computationally intensive algorithm for IP-to-AS mappings based on dynamic programming [12]. Since our tool is designed to work in real time, we'd like to use simpler and faster approaches. There are two such methods. One is based on operational BGP tables, the other is based on IRR registry.

In the first method, we obtain the BGP tables from Route Views vantage points. Given a router address, we find the longest matched prefix. If the path to this prefix ends with AS-o, then we say the router is located in AS-o. As of Mar. 27, 2004, there are 198,238 prefixes in these BGP tables.

In the second method, we obtain the IRR databases from 59 IRRs (*e.g.*, RIPE, RADB, ARIN, APNIC). The databases record that which prefix is allocated to which AS. Among the 198,238 prefixes in operational BGP tables, this method is able to decide the origin ASes of 171,013 (86%) prefixes.

We first use BGP tables to do the mappings, then if there is some router addresses not mapped, we make use of the IRR data. According to [12], this mapping has 73% accuracy.

B. What Routers Block Traceroutes

As described previously, our experimental result shows that each looking glass has a small traceroute coverage of peering topology. There are two possible explanations for it. One is that many routers block the traceroute probes and cause downstream routers untraceable. Another is that the peerings which are not "seen" are on the edge of Internet.

This section discusses the first explanation. We want to know whether the routers in some transit-ASes tend to block traceroutes. Our method is to compute for each transit AS-T the number of prefixes (or origin ASes) that can't be tracerouted because the traceroutes stopped at some router r in T. That is, router r is the last identified router for the traceroute to some origin AS. The actual blocking router is the next router to r, which cannot be identified (*i.e.*, traceroute shows '*' for this router), hence we do not know which AS it belongs to. Currently, we assume this blocking router resides in the same AS (*i.e.*, AS-T) as the router r. Figure 5 shows an example.

The result shows that there are 2,039 transit ASes where traceroutes from some looking glasses terminated. Table III shows 10 transit ASes that blocks traceroutes to most origin ASes. Most of these ASes are in the Internet core, so it is possible that some routers in these ASes are configured to block

TABLE II. Estimated traceroute coverage of looking glasses.

Looking Glass	AS	Country	Links	Peerings	Coverage (%)	Marginal (%)
sow.isi.edu	226	USA	41536	7230	12.84	12.84
tcruskit.telstra.net	1221	Australia	41473	7033	12.49	3.23
proxy1.syd.connect.com.au	2764	Australia	38775	6533	11.60	2.09
cgi.cs.wisc.edu	59	USA	40392	6878	12.22	1.00
www.research.compaq.com	33	USA	39609	6838	12.15	0.86
www.telcom.arizona.edu	1706	USA	39179	6900	12.26	1.27
www.vineyard.net	2914	USA	41917	7072	12.56	0.37
voa.his.com	3491	USA	42274	7114	12.64	0.43
home.acadia.net	3561	USA	42787	7139	12.68	0.54
www.gip.net	4005	USA	42208	6998	12.43	0.23
traffi c.stealth.net	8002	USA	40654	6931	12.31	0.28
www.univ-st-etienne.fr	1939	France	41275	6891	12.24	0.14
www.eu.org	5410	France	42196	6969	12.38	0.24
www.helios.de	517	Germany	42410	6970	12.38	0.81
ppewww.ph.gla.ac.uk	786	United Kingdom	31316	4614	8.19	0.21
traceroute.colt.net	8220	United Kingdom	40828	6970	12.38	0.28
www.net.cmu.edu	9	USA	44626	7504	13.01	2.23
www.net.berkeley.edu	25	USA	45720	7684	13.32	0.59
noc.net.umd.edu	27	USA	41949	7338	12.72	0.29
www.sdsc.edu	1227	USA	44487	7875	13.65	0.21
www.psychosis.net	1784	USA	43381	7548	13.08	0.66
www.t1shopper.com	2914	USA	44417	7648	13.26	0.10
www.slac.stanford.edu	3671	USA	42959	7500	13.00	0.19
www.fmp.com	3796	USA	45488	7576	13.13	0.14
unixvirt-svca.www.conxion.com	4544	USA	42630	7298	12.65	0.13
www.socket.net	4581	USA	42921	7346	12.73	0.08
noc.informationwave.net	5042	USA	38631	6681	11.58	0.29
www.abs.net	5641	USA	47170	7790	13.50	0.13
www.ntplx.net	6062	USA	43738	7231	12.53	0.07
www.getnet.net	6091	USA	44000	7377	12.79	0.08
www.spftl.com	6172	USA	32081	5959	10.33	0.02
zeus.bintec.com	6283	USA	44167	7279	12.62	0.13
lava.net	6435	USA	44090	7401	12.83	0.04
www.above.net	6461	USA	44338	7637	13.24	0.20
www.undergroundpalace.com	6517	USA	33843	6246	10.83	0.01

TABLE III. Traceroute-blocking Transit-ASes. $O_{blocked}$ is the number of origin ASes which looking glasses can't traceroute to because it stops at this AS.

AS	Name	$O_{blocked}$
10844	Vastnet	14987
701	Alternet	2117
1239	SprintLink Backbone	2002
7018	AT&T WorldNet Services	1552
3356	Level 3 Communications	1408
209	Qwest Communications	1353
2914	Verio	896
4323	Time Warner Telecom	637
3561	Cable & Wireless	586
6461	Metromedia Fiber Network	582

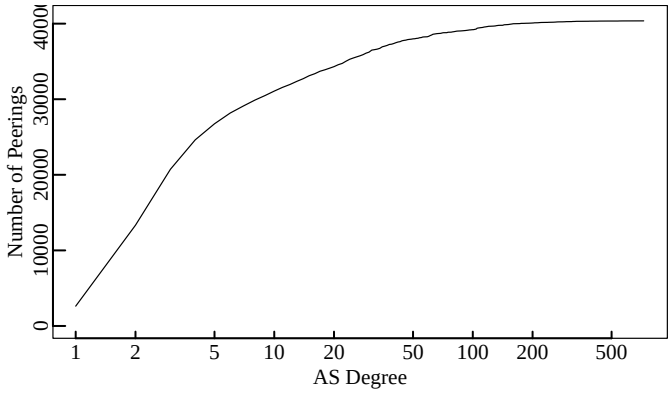

Fig. 6. Distribution of the number of peerings that are not tracerouted and have x degree.

traceroute probes. However, we are unable to determine where these routers reside.

C. What Peerings Are Not Tracerouted

To examine the second explanation, we show the relation between the visibility of the peerings and the AS degree (we assume the ASes of low degree are on the edge of Internet). Specifically, the degree of peering (X, Y) is $\min\{degree(X), degree(Y)\}$.

Figure 6 shows the CDF of the peerings that are not tracerouted. Figure 7 shows the CDF's of the peerings that are tracerouted by L looking glasses, where L is labeled in the end of the CDF curve. It appears that there is no significant correlation between the degree of peering and its visibility. Accordingly, we prefer the first explanation that some routers in Internet core ASes tend to block traceroutes.

V. Localization Experiments

Based on the initial routing status obtained in Section IV we conduct experiments of localizing missing route events using

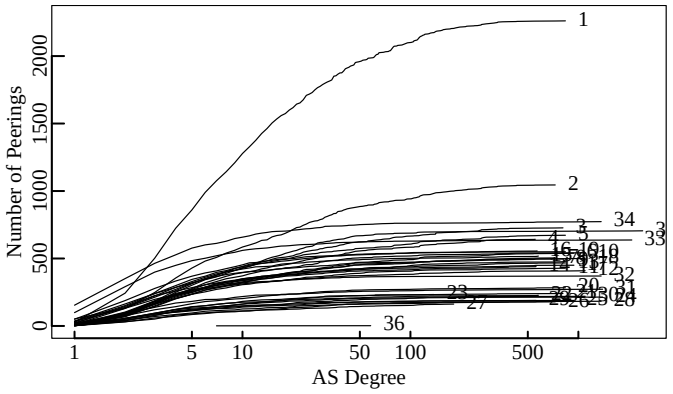

Fig. 7. Distribution of the number of peerings that are tracerouted by L looking glasses and have x degree. L is labeled in the end of the distribution.

two algorithms. Our preliminary results are described below.

A. Peering-level Localization

This experiment uses the algorithm described in Section III-B to determine how many missing route events are caused by peering failures. First, we detect the possible missing-route events by monitoring vantage points' BGP tables. Specifically, we examine the BGP updates from Route Views vantage points every 15 minutes and choose *one* prefix p that matches the following conditions.

1. The routes to this prefix are withdrawn by at least two vantage points. So this event likely takes place in some transit peerings instead of the peerings adjacent to vantage points. Also, this event likely lasts long enough for us to localize it.
2. The AS peerings along those routes were all reachable by some looking glasses according to the initial routing status obtained in Section IV.
3. If multiple prefixes meet the above two conditions during past 15 minutes, select the prefix whose event occurrence time is latest. So this event likely lasts long enough for us to localize it.

Then, we determine whether this event is caused by peering failure by checking the reachability of the common peerings along the withdrawn routes. Specifically, supposed there are n common peerings $\{peering_i\}$ in the withdrawn routes, we check the peering reachability by the following procedure:

1. For $peering_i$, we find the looking glasses $\{l_{i,j}\}$ that traceroutted through $peering_i$ to reach some IP address $\{a_{i,j}\}$.
2. Send traceroute queries $\{a_{i,j}\}$ to looking glasses $\{l_{i,j}\}$. If at least one looking glass returns that it can reach $peering_i$, then $peering_i$ is reachable, *i.e.*, it doesn't suffer peering failure.
3. If there is one peering unreachable, then we say this missing route event is caused by peering failure occurred in that peering.

We ran the experiment for 4 days from Feb. 10 to 13, 2004. During the experiment, our monitor examines the vantage points' BGP tables 381 times (once per 15 minutes). The result shows that we detect 378 possible missing-route events, of which 11 events (2.91%) have some peerings unreachable.

B. Approximate Prefix-level Localization

This experiment is to use the algorithm of Section III-C. Similarly, we examine the BGP updates from Route Views vantage points every 15 minutes and choose *one* prefix p that matches the following conditions.

1. The routes to this prefix are withdrawn by at least two vantage points.
2. The initial routing status shows that we have more than three looking glasses (denoted by $\{l_i\}$) that had routes $\{r_{1,i}\}$ to this prefix.
3. The occurrence time of the event is latest.

When such a prefix is found, we ask looking glasses $\{l_i\}$ to traceroute to it and obtain the new routes $\{r_{2,i}\}$. Then we compare the two set of routes $\{r_{1,i}\}$ and $\{r_{2,i}\}$ to infer the suspect peerings using the heuristics described in Section III-A. We ran the experiment for a week from Mar. 21 to 27, 2004. During the experiment, our monitor examines the vantage points' BGP tables 669 times. The result shows that 534 possible missing-route events are detected, of which 54 events (10.11%) can be localized based on our heuristics.

VI. Future Work

We have shown that missing routes are a problem in the Internet (about 3.5 occur every hour) and that existing monitoring infrastructure of looking glasses can provide us a rough location of the problem ASes (about 10% of the time using our prototype monitor). There is still much work to be done. First, we need to build query interface to more looking glasses so that we can reduce the query load for each looking glass and have more diverse initial routing status. Second, we'd like to know if there is a pattern of what routers are more likely to block traceroute queries. Finally, there is a problem of validation, *i.e.*, how can we know our localization algorithms really catch the suspect peerings. We are cooperating with our ISP to setup a controlled experiment to validate these results.

References

[1] Timothy G. Griffin and Gordon Wilfong, "An analysis of BGP convergence properties," in *Proceedings of the ACM SIGCOMM*, 1999, pp. 277–288.
[2] Timothy G. Griffin, F. Bruce Shepherd, and Gordon Wilfong, "The stable paths problem and interdomain routing," *IEEE/ACM Transactions on Networking*, vol. 10, no. 2, pp. 232–243, Apr. 2002.
[3] Danny McPherson, Vijay Gill, Daniel Walton, and Alvaro Retana, *BGP Persistent Route Oscillation Condition*, rfc 3345 edition, Feb. 2002.
[4] A. Basu, L. Ong, B. Shepherd, A. Rasala, and Gordon Wilfong, "Route oscillations in I-BGP with route reflection," in *Proceedings of the ACM SIGCOMM*, 2002.
[5] Timothy G. Griffin and Gordon Wilfong, "Analysis of the MED oscillation problem in BGP," in *Proceedings of the IEEE International Conference on Network Protocols*, 2002, pp. 90–99.
[6] Philip Smith, "Troubleshooting BGP," in *NANOG 29*, Oct. 2003.
[7] Timothy G. Griffin, "What is the sound of one route flapping," in *IPAM Workshop on Large-Scale Communication Networks: Topology, Routing, Traffic, and Control*, Mar. 2002.
[8] D.-F. Chang, R. Govindan, and J. Heidemann, "The temporal and topological characteristics of BGP path changes," in *Proceedings of the IEEE International Conference on Network Protocols*, 2003, pp. 190–199.
[9] Matthew Caesar, Lakshminarayanan Subramanian, and Randy H. Katz, "Root cause analysis of Internet routing dynamics," Tech. Rep. UCB/CSD-04-1302, University of California, Berkeley, 2003.
[10] Craig Labovitz and Abha Ahuja, "Shining light on dark Internet address space," in *NANOG 23*, Oct. 2001.
[11] Zhuoqing Mao, Ramesh Govindan, Randy Katz, and George Varghese, "Route flap damping exacerbates Internet routing convergence," in *Proceedings of the ACM SIGCOMM*, 2002.
[12] Zhuoqing Mao, David Johnson, Jennifer Rexford, Jia Wang, and Randy H. Katz, "Scalable and accurate identification of AS-level forwarding paths," in *Proceedings of the IEEE INFOCOM*, Mar. 2004.