Investigating Prefix Propagation through Active BGP Probing*

Lorenzo Colitti[†‡], Giuseppe Di Battista[†], Maurizio Patrignani[†],
Maurizio Pizzonia[†], and Massimo Rimondini[†]

[†] Dipartimento di Informatica e Automazione
Università Roma Tre
Via della Vasca Navale 79
00144 Roma, Italy
{colitti,gdb,patrigna,pizzonia,rimondin}@dia.uniroma3.it

[‡] RIPE NCC
Singel 258
1016 AB Amsterdam
The Netherlands
lorenzo@ripe.net

Abstract

For an Internet Service Provider (ISP), the knowledge of which interdomain paths could be traversed by its BGP announcements – and thus traffic flows – is essential to predict the impact of network faults, to develop effective traffic engineering and peering strategies, and to assess the quality of upstream providers. However, current methodologies do not provide this information. We present methodologies to discover how the BGP announcements for an ISP's prefix are propagated through the Internet using withdrawals and specially crafted AS-sets. The techniques allow an ISP to determine which paths could be traversed in the presence of network faults or different routing policies on the ISP's part and to deduce the routing policies of other ISPs with respect to its network. We validate our techniques through experimentation in the IPv6 and IPv4 Internet, showing that they can be safely and effectively applied in real-world situations.

1 Introduction

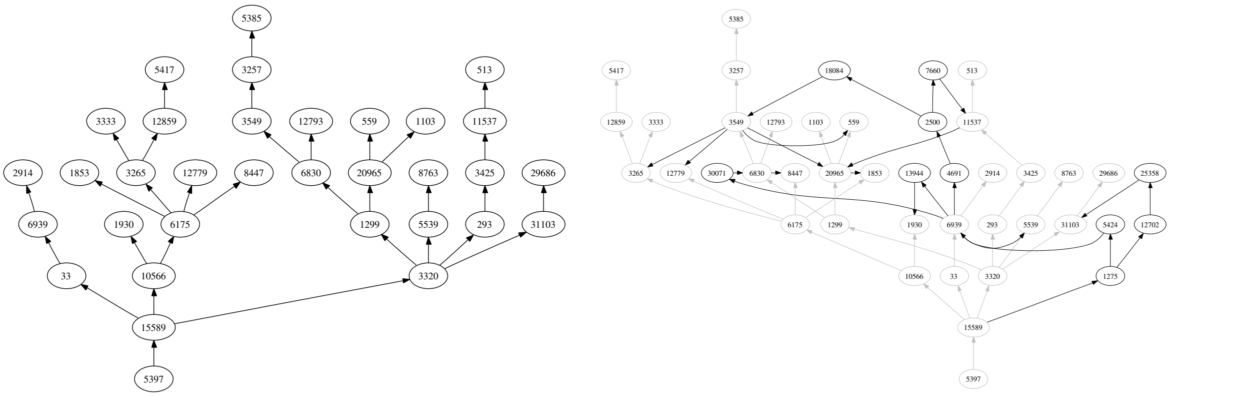
Interdomain routing in the Internet is based on the Border Gateway Protocol (BGP), which partitions the Internet into a set of Autonomous Systems (ASes). BGP advertises the reachability of destinations (prefixes) through route announcements, which originate from the AS to which the prefix belongs and are selectively propagated from AS to AS in accordance with routing policies. Traffic addressed to a certain prefix flows through the ASes that have propagated the corresponding announcement. Thus, for an Internet Service Provider (ISP), the knowledge of which interdomain paths might be traversed by its BGP announcements – and thus traffic flows – is essential to predict the impact of network faults, to perform effective traffic engineering, to develop peering strategies, and to assess the quality of connectivity provided by the ISP's upstream providers.

Interdomain topology discovery is a subject on which much has been written. The behavior of BGP has been passively observed by capturing and analyzing BGP routing tables [2] and announcements [3]; BGP beacons [13] have been used to study network dynamics; techniques have been proposed to deduce interdomain topology using probe packets [10, 8, 20, 14, 16]; and BGP data has even been used to deduce commercial relationships between ASes [7, 5].

Unfortunately, the literature does not provide methods to determine how BGP announcements are propagated. Even recent methods which make use of the passive observation of BGP dynamics [24, 6] still do not provide comprehensive information on how a specific prefix is seen by the Internet and how it might be seen in the event of link faults, changes in routing, or different traffic engineering strategies.

This is because existing methods either attempt to discover the AS-level interconnections of the Internet, ignoring the effect of routing policies, or limit themselves to studying the paths to a particular destination at a given instant in time, and thus do not obtain any information on alternate paths that are permitted by routing policies, such as backup paths. As an example, Fig. 1(a) shows what the operators of AS 5397 may discover about the routing of their IPv6 prefix `2001:a30::/32` at a given time by querying the RIPE NCC RIS service [18]. An arrow from an AS A to an AS B means that an announcement of the specified prefix

* Work partially supported by European Commission – Fet Open project DELIS – Dynamically Evolving Large Scale Information Systems – Contract no 001907, by MIUR under Project ALGO-NEXT (Algorithms for the Next Generation Internet and Web: Methodologies, Design, and Experiments), and by "The Multichannel Adaptive Information Systems (MAIS) Project", MIUR Fondo per gli Investimenti della Ricerca di Base.



(a)

(b)

Figure 1. (a) What an operator of AS 5397 may discover about the routing of its prefix 2001:a30::/32 on Dec 30 2004 at 02:44:00 UTC using standard routing information services. (b) Additional topology discovered by sending one custom BGP announcement which includes ASes 33, 3320 and 10566 in an AS-set.

was made by A to B. The graph shows that AS 5397's upstream provider (AS 15589) propagates the announcement to AS 10566, AS 3320 and AS 33, but does not show, for example, that AS 15589 is propagating the announcement to AS 1275 as well. Fig. 1(b) shows the additional information that AS 5397 may obtain by sending out a single BGP announcement using the techniques presented in this paper. A more thorough application of these techniques yields almost three times the number of ASes and more than seven times the peerings visible using a standard RIS query (see Table 1).

In this paper we present active probing primitives that use standard BGP to influence how the announcements for a given prefix propagate through the Internet. We show how an AS that originates a given prefix can use these primitives (i) to discover which ASes and peerings can be traversed by the BGP announcements for that prefix, (ii) to check if a certain AS-path is permitted, for that prefix, by other ASes' routing policies, and (iii) to infer, given two AS-paths for the prefix ending in a given AS, which one is propagated by that AS to its peers.

2 Background

An Internet Service Provider (ISP) typically administers one or more *Autonomous Systems* (ASes). An AS is a portion of the Internet under a single administrative authority and is identified by an integer number. ASes exchange routing information with other ASes by means of a routing protocol called Border Gateway Protocol (*BGP*) [17, 21]. Two ASes that directly exchange routing information are said to have a *peering* between them. The ASes that have peerings with an AS A are termed the *peers* of A.

BGP operates on blocks of contiguous IP addresses known as *prefixes*. The sequence of ASes traversed by traffic sent to a particular prefix is determined by the *AS-path* attribute associated with the prefix. A $\langle$prefix, AS-path$\rangle$ pair is known as a *route*. BGP peers exchange routes using BGP *update* messages, which are either route *announcements* or route *withdrawals*.

The AS that initially announces a prefix (typically the AS to which the prefix belongs) is called the *originator* of the prefix. Routes are propagated by means of route announcements to peer ASes. A router which receives an announcement inserts the route into its *routing information base* and recalculates the *best route* to the prefix. If the best route has changed, it prepends its AS identifier to the AS-path it received and propagates the new best route to its peers; thus, the AS-path of an announcement is generally the list of ASes that the announcement has passed through. An AS-path may be optionally ended by an *AS-set* (an unordered set of AS identifiers) which is used in certain cases of route aggregation. In order to avoid routing loops, a BGP router discards announcements which include its own AS number in the AS-path.

A prefix p is selectively propagated by an AS A to its peers depending on the routing policy adopted by A and on the contents of the announcement [7, 11]. We say that an AS-path $A_n \dots A_2 A_1$, where A_1 is the origin AS, is *feasible* for a prefix p if the policies of each A_i permit A_i to announce p to A_{i+1} with AS-path $A_i \dots A_1$. Observe that the feasibility of an AS-path for a prefix does not imply that the AS-path is necessarily visible in the Internet: it only means that under certain circumstances it may be visible. The set of feasible AS-paths for a prefix p thus contains all the AS-paths that may be observed for p in the Internet. We note that the concept of feasible path has also been used in the literature on the stability of BGP with the name of *permitted* path (e.g. [9]). A peering between two ASes P and Q is feasible for p in the direction from P to Q if there exists at least one feasible path in which P immediately follows Q. In diagrams we shall represent a feasible peering from P to Q with a directed arc from P to Q. For example, in Fig. 1(a) the directed arc between 10566 and 6175 indicates that the policies of AS 10566 permit it to announce the prefix `2001:a30::/32` to AS 6175.

We name *routing state* for a prefix p at a given time the set of best routes to p of each router in the Internet at that time. We empirically say that a routing state for p is *stable* if we have observed no BGP updates for p for a sufficiently large time interval. To obtain (partial) information about the evolution of the Internet routing state, projects such as the RIPE NCC Routing Information Service (RIS) [18] and the University of Oregon's RouteViews Project [22] deploy *route collectors* in specific points of the Internet to record BGP updates from routers in a number of ASes, which we name *collector-peers*. Thus, the best routes of each collector-peer at any given time are known. The RIBs of the collectors and the updates they receive are periodically dumped, permanently stored and made publicly available over the Web.

3 BGP Probing Primitives

Consider the case of an AS Z that originates a prefix p. In this section we present basic primitives for active BGP probing which can be used by the operators of Z to obtain information on how p is propagated in the Internet and, based on this information, to perform targeted investigation of routing policies. The primitives allow us to discover alternate paths that are feasible but are not ordinarily used, such as backup paths.

The primitives are based on sending BGP updates for p and observing the effects using route collectors or looking glasses. This implies that connectivity to p is disrupted during their use. However, p may be a test prefix which does not carry production traffic: since the vast majority of routing policies do not discriminate between different prefixes originating in the same AS based only on the prefix itself, the results obtained using p will hold for the other prefixes originated by Z as well.

Our first primitive, which we name *withdrawal observation*, consists in sending a withdrawal for p and observing all the paths that become visible during the BGP convergence process. As first noted in [12], the withdrawal of a prefix causes a potentially lengthy (usually lasting several minutes) convergence process in which BGP explores alternate paths before concluding that the prefix is unreachable. Therefore, observation of the BGP updates during a withdrawal allows us to record the alternate paths which appear during convergence and are not visible in a stable routing state. We note that the idea of observing BGP updates to discover alternate paths is not new (see, for example, [6]); however, while the approach in existing work is that of passive observation, our primitive involves the purposeful generation of withdrawals in order to observe alternate paths.

Our second primitive, which we name *AS-set stuffing*, consists in announcing p with an AS-path of $Z\{A_1, A_2, \ldots, A_n\}$, where $\{A_1, A_2, \ldots, A_n\}$ is an AS-set. Since a BGP router discards any announcement whose AS-path contains its own AS number, the ASes A_i will discard the announcement and will not propagate it to any other ASes. This effectively eliminates ASes A_i from the Internet as far as the propagation of p is concerned; we name these ASes *prohibited* ASes. The observation of the resulting routing state, and possibly of the convergence process, allows us to determine alternate feasible paths for p that do not contain the prohibited ASes. The use of an AS-set ensures that the length of the AS-path, which is one of the most important metrics used by BGP routers in the route selection process, does not depend on the number of prohibited ASes.

AS-set stuffing allows us to do more than simply observe alternate paths: it allows us to alter the interdomain routing for p in a stable state. This has two important advantages. The first is that observing alternate paths does not require the collection of BGP updates, but may be performed using any commonly available looking glass, thus greatly increasing the number of observation points that may be employed. The second is that once the routing state is altered, other tools may be used to probe network connectivity and performance, thus allowing "what-if" analyses on Internet performance to be performed.

There are intrinsic limitations to what we may observe using AS-set stuffing. Consider Fig. 2(a). If Z is the originator of p and C_1 is a collector, the use of AS-set stuffing does not guarantee that it is possible to observe the peering between B and A, although it is feasible: we cannot force the observation of the peering in a stable routing state, because the only probes that can be performed are to prohibit A and/or B, but in every case the peering will not be visible since one of its endpoints is prohibited. However, a path such as C_1DABZ may be visible during BGP convergence, depending on the unpredictable order of updates propagated in the network. In this case, our primitives will observe the peering.

4 Prefix Propagation Discovery

In this section, we present several applications of the primitives introduced in Section 3, again considering the

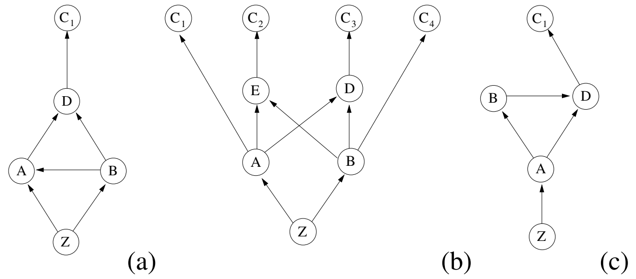


Figure 2. Z is the origin AS; C_1, C_2, C_3, and C_4 are collectors. An edge directed from x to y represents a peering that is feasible in that direction. (a) A topology where AS-set stuffing cannot guarantee the discovery of the feasible peering between B and A. (b) A topology in which level-by-level exploration cannot discover all feasible peerings. (c) A topology in which the feasibility of path $ZABDC_1$ cannot be determined using AS-set stuffing.

case of an AS Z that originates a prefix p. In the following, we use the concept of *feasibility graph*, which represents feasible peerings and how they are topologically related, and is thus a valuable starting point for further routing policy deductions. Given a set $\mathcal{S}$ of feasible paths for a prefix p, we name *feasibility graph* the directed graph whose nodes are the ASes and whose arcs are the peerings that appear in the paths of $\mathcal{S}$. Observe that, because of routing policies, not every path in the feasibility graph is feasible; however, since every arc in the feasibility graph is feasible, an arbitrary path $\mathcal{P}$ on the feasibility graph may well be feasible even if it is not in $\mathcal{S}$. We name *level* of a node X the length of the shortest directed path from Z to X.

4.1 Obtaining a Feasibility Graph

There are several possible methods to obtain a feasibility graph. The simplest way is to query the route collectors for p, but the extent of such a graph is limited, as can be seen in Fig. 1(a). A more effective approach is to use withdrawal observation. Although simple, this requires the collection of BGP updates, and thus limits the number of observation points that may be used. Furthermore, the peerings discovered depend on the unpredictable order of BGP updates generated during convergence.

Another approach makes use of AS-set stuffing. By prohibiting a set of ASes, we may record one or more alternate paths. So, to obtain the most complete picture possible, Z could in theory send 2^n announcements, including in each one an AS-set prohibiting one of the 2^n subsets of the n ASes in the Internet. Such a brute force approach is infeasible both because the number of ASes that may be included in an AS-set is limited and because of the long exploration times it would require. Therefore, we adopt the following strategy: begin with the directed AS graph seen by the route collectors at a certain instant and proceed level by level, starting from level one. For each level, prohibit all the known ASes in the level. At this point, either there will be no feasible paths to the collectors, or the announcements will propagate through new, previously unknown, nodes at the same level. Each new node and arc found is added to the feasibility graph. If new nodes in the same level have been found, insert them into the prohibited set; otherwise, empty the set of prohibited ASes and proceed to the next level. As an example, Fig. 1(b) shows the new nodes and arcs discovered starting from the situation in Fig. 1(a) by announcing the AS-set $\{33, 3320, 10566\}$, which corresponds to all the known nodes at level two in the initial graph. After every BGP update, we wait a period of time to allow the network to converge and to limit the effects of route flap dampening [15]. To observe paths that are not visible in stable states, we examine all the updates received for p during the convergence period. We name this algorithm the *level-by-level* exploration algorithm.

We note that this algorithm, in addition to the intrinsic constraints of AS-set stuffing already discussed in Section 3, suffers from further limitations. For example, in Fig 2(b), the algorithm does not guarantee that all the arcs $A \rightarrow D$, $B \rightarrow E$, $A \rightarrow E$, and $B \rightarrow D$ will be discovered. Another possible algorithm based on the same primitive is as follows: once all nodes in a level l have been found, process each node in l in turn. For each one, prohibit all the other nodes in l, then progressively prohibit all visible nodes in level $l + 1$ until no new nodes in level $l + 1$ are found. Then empty the set of the prohibited nodes and advance to the next node in l. This approach overcomes the limitations of level-by-level exploration, but it requires many more updates and therefore much longer exploration times.

Finally, all these algorithms observe the network using all the route collectors simultaneously; however, in certain topologies, using only one collector at a time and merging the results at the end would discover more peerings. For example, in Fig. 2(b), exploring the topology separately using level-by-level exploration, first using only C_2 and then only C_3, would also reveal the arcs $B \rightarrow E$ and $A \rightarrow D$, while exploring the topology using both collectors simultaneously might only discover the arcs $A \rightarrow E$ and $B \rightarrow D$.

4.2 Path Feasibility Determination

Suppose that we are interested in knowing whether a certain AS-path $\mathcal{P}$ is feasible for a prefix p. For this purpose, we may use the following algorithm, which we name the *nailed-path* algorithm. Assume $\mathcal{P}$ ends at an AS A containing a collector-peer or looking glass C and consider the feasibility graph obtained as described in Section 4.1. Prohibit all the ASes in levels up to and including the level of A except for the ASes in $\mathcal{P}$. Now observe the AS-path $\mathcal{Q}$ seen by C: it is likely that either $\mathcal{Q} = \mathcal{P}$ and the path is feasible, or C does not see the prefix and the path is not feasible. If $\mathcal{Q} \neq \mathcal{P}$ (i.e., ASes or peerings that were not in the initial feasibility graph have been revealed), we repeat the above procedure after including the newly discovered ASes in the prohibited set. If $\mathcal{Q}$ reveals a shortcut between two ASes in $\mathcal{P}$, then the feasibility of $\mathcal{P}$ cannot be determined. This algorithm can be extended to deal with the case in which $\mathcal{P}$ does not end at a collector-peer: details are given in [4].

4.3 Path Preference Comparison

Given two feasible AS-paths $\mathcal{P}_1$ and $\mathcal{P}_2$ ending at the same observation point (a collector-peer or looking glass) C in AS A, we may use AS-set stuffing to determine which of the two AS-paths is preferred by C.

To determine which path C prefers, we obtain a feasibility graph as described in Section 4.1 and attempt to ensure that the only announcements received by C for p have the paths $\mathcal{P}_1$ and $\mathcal{P}_2$. Namely, we prohibit all the ASes in all levels up to the level of A except the ASes in $\mathcal{P}_1 \cup \mathcal{P}_2$. Usually, this is enough for C to see either $\mathcal{P}_1$ or $\mathcal{P}_2$. If not, the announcement may lead to the discovery of new ASes which are not in $\mathcal{P}_1$ or in $\mathcal{P}_2$ and were not previously visible in the feasibility graph; in this case it is sufficient to prohibit the new ASes and repeat the announcement until no new ASes are discovered.

The announcement may also lead to the observation of another path made up exclusively of ASes which belong to either $\mathcal{P}_1$ or $\mathcal{P}_2$; in this case it is not possible to determine whether C prefers $\mathcal{P}_1$ or $\mathcal{P}_2$. This may occur only if there is a feasible peering between an AS in $\mathcal{P}_1$ and an AS in $\mathcal{P}_2$. Furthermore, if $\mathcal{P}_1$ and $\mathcal{P}_2$ have ASes in common in addition to A, it is not possible to determine which AS-path is preferred by C, since routers in one of the common ASes may have chosen between the two paths and only re-announced one of them, resulting in only one of them reaching A. Finally, since this technique requires us to determine whether $\mathcal{P}_1$ and $\mathcal{P}_2$ are feasible, it cannot be applied if it is not possible to determine the feasibility of the paths as described in Section 4.2.

An example of this technique is in Fig. 4(b), where we aim to determine whether the collector-peer in AS 8468 prefers the path through AS 6461 or AS 9044. The graph was obtained using withdrawal observation. Since all the ASes not part of the two paths (shown in gray) are prohibited, the preferred path is the one seen by AS 8468. For details of the experiment, see Section 6.3.

5 Applicability Considerations

In this section, we discuss the technical and operational factors that limit the applicability of our techniques – including their impact on the interdomain routing infrastructure – and show how they are unlikely to be an issue given current operational practices.

First, the number of ASes that may appear in an AS-set is limited: the BGP specification limits the length of an AS-set to 255 ASes, and the limits posed by commercial router BGP implementations are lower. This was not an issue in any of the topologies we tested, and should not pose a problem for all but the very largest ISPs with hundreds of peers. Our tests on network equipment from various vendors showed that they correctly handle long AS-sets [4]. This is is not surprising, as AS-sets are constantly present in the Internet and AS-sets of unusual length have been observed before [1, 23]; we know of no reports of adverse affects on the network caused by these announcements.

A second constraint is that posed by route flap dampening, which limits the propagation of frequent updates for the same prefix. The exact effects of route flap dampening depend on the topology, but [15] suggests that the maximum length of time a route can be suppressed by dampening in today's Internet is approximately one hour; therefore, dampening can be avoided by rate-limiting BGP probes to less than one every hour [4]. Even at this rate, all the experiments we performed, including level-by-level explorations of nodes up to four levels away, were completed in a few hours. Rate-limiting the updates also has the effect of limiting the load placed on routers by the explorations, although this is negligible in the face of the order of the ∼15,000 updates per hour seen by Tier-1 routers.

As regards possible impact on router memory, we expect a large AS-set to consume about 200 extra bytes per prefix. This is negligible compared to the several megabytes of memory used by a full BGP table; also, since prefixes used for AS-set stuffing suffer connectivity problems, we expect its use to be restricted to test prefixes and limited to temporary experiments. The use of AS-sets including other ASes should not cause confusion or hamper debugging, since the identity of the origin AS is visible in the path (it is the AS immediately before the AS-set), and is also documented in the Internet registries. On a more philosophical note, it has been suggested that BGP was not designed with AS-set stuffing in mind and that it is not appropriate to include the numbers of other ASes in an announcement. We address these issues, which are not technical problems, in [4].

6 Experimental Results

Due to the innovative nature of our techniques, we first tested them in the IPv6 Internet, in order to limit the ex-

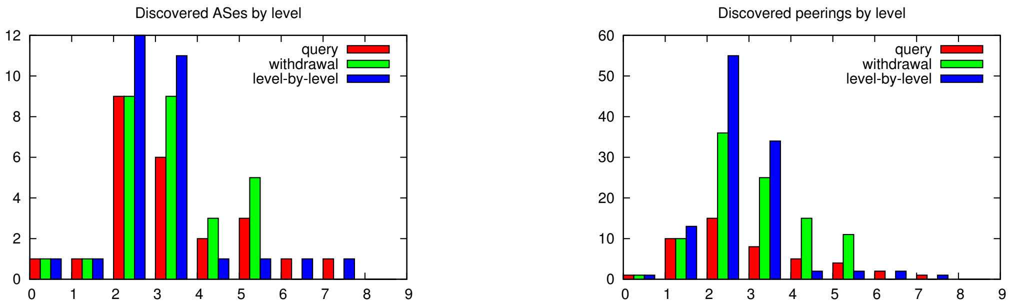


Figure 3. ASes and peerings found by a standard RIS query, by withdrawal observation, and by level-by-level exploration, sorted by level (IPv4). Every peering is counted twice, once at the level of one endpoint and once at the level of the other endpoint.

Method	IPv6		IPv4	
	ASes	Peerings	ASes	Peerings
Stable state	32	31	24	23
Withdrawal	94 (2.9x)	211 (6.8x)	28 (1.2x)	49 (2.1x)
Level-by-level	97 (3.0x)	222 (7.2x)	29 (1.2x)	55 (2.4x)

Table 1. ASes and feasible peerings found by a standard RIS query, by withdrawal observation, and by level-by-level exploration.

tent of any possible problems they might cause. Once we were confident of their reliability and effectiveness, we performed more limited testing on the IPv4 Internet.

All BGP updates were generated using custom software developed by the authors [19], and the effect of each BGP update sent was observed by means of the RIS database, which provides BGP data collected in real-time. IPv6 BGP announcements originated in AS 5397 using the prefix `2001:a30::/32`, while IPv4 BGP announcements originated in AS 12654 using the prefixes `84.205.73.0/24` and `84.205.89.0/24`. These IPv4 prefixes are reserved for BGP experiments and are announced by the RIS route collectors, currently present in 13 locations around the world. In order to approximate the situation of a small to medium-sized ISP, each prefix was announced by one route collector at a time.

6.1 Prefix Propagation Discovery

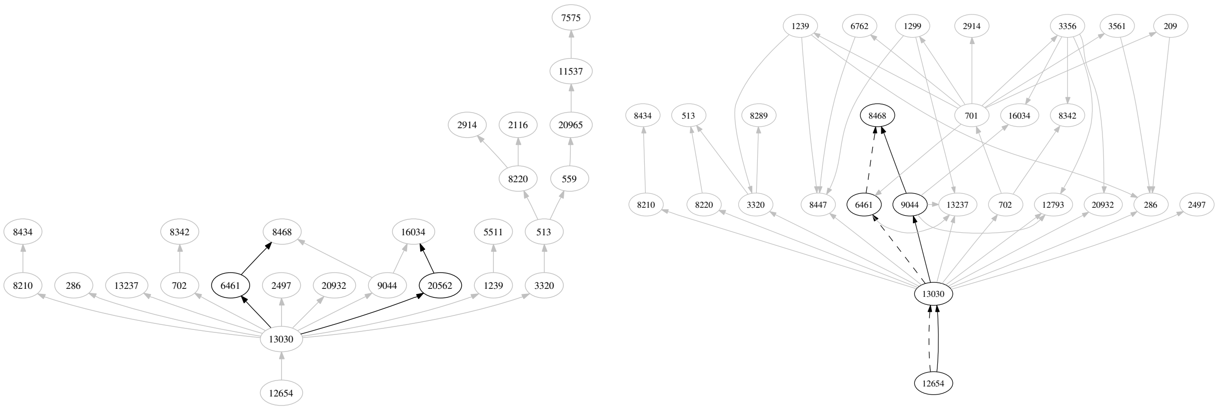
To evaluate the effectiveness of our topology discovery strategies, we compared the feasibility graphs they generated to graphs obtained from the collectors in stable routing states. The results are in Table 1. As can be seen, our techniques observe between 20% and 200% more ASes and between 110% and 620% more feasible peerings than when active probing is not used. There is a difference in effectiveness between IPv6 and IPv4, which we believe to be due to the much more restrictive routing policies employed in the IPv4 Internet.

The results obtained using AS-set stuffing are slightly better than those produced by withdrawal observation: although the results are similar in terms of the number of nodes and peerings discovered, the topologies discovered are different. The graphs in Fig. 3 show the level at which the new ASes and peerings are discovered. As can be seen from the graph, the topology produced by level-by-level exploration is more concentrated in the lower levels of the feasibility graph. Since the ASes that were discovered by the two methods are mostly the same, this means that certain ASes were discovered at a lower level by level-by-level exploration than by withdrawal observation. Therefore, by definition of level of a node, the topologies produced by level-by-level exploration are more accurate.

Fig. 4 shows an example of how AS-set stuffing may be used to discover the peers of an ISP's upstream provider. Fig. 4(a) shows a stable state feasibility graph for prefix `84.205.73.0/24` announced from RRC11 and observed at 14:49:59 UTC on July 5 2005. The graph shows that AS 13030 is the only upstream of the origin AS (AS 12654) and has 9 visible peers. A single announcement with an AS-path of 12654 {8210, 20932, 286, 13237, 702, 2497, 9044, 1239, 3320} at 14:55:35 UTC allowed the discovery of two previously unknown peers of AS 13030 which propagate the announcements for the prefix (in black). We note that a previously performed withdrawal observation had not discovered either of these ASes as peers of AS 13030.

6.2 Comparison with the Full AS Graph

We also compared the results of our per-prefix discovery strategies with more conventional interdomain topology discovery techniques. At a given time, we simultaneously obtained a feasibility graph W using withdrawal observa-



(a)

(b)

Figure 4. Examples of use of our techniques. (a) Using targeted announcements to discover the peers of an upstream ISP: feasibility graph obtained by making a single announcement with AS-path 12654 {8210, 20932, 286, 13237, 702, 2497, 9044, 1239, 3320}. The two newly-discovered peers of AS 13030 are in black. (b) Path preference comparison: the paths to be compared are 8468 6461 13030 12654 (dashed) and 8468 9044 13030 12654 (solid). The ASes in gray are prohibited.

Date	Protocol	I	W	I only	W only
2005/02/23 09:54	IPv6	312	158 (51%)	175	21 (13%)
2005/02/25 10:03	IPv6	334	168 (50%)	189	23 (14%)
2005/02/27 15:18	IPv6	302	154 (51%)	174	26 (17%)
2005/07/05 00:00	IPv4	241	61 (25%)	181	1 (2%)

Table 2. Comparison between the arcs in the graph W generated by withdrawal observation and those in the graph I induced by W in the global AS-graph C.

tion and a full AS graph C for all the prefixes announced on the Internet. We then compared W with the graph I induced by the nodes of W in C.

The results, in Table 2, show that our per-prefix graphs only have about 50% of the arcs of the induced graphs for IPv6 and 25% for IPv4. This shows that there is a substantial difference between existing topology discovery methods and our active probing discovery methods. The topology captured by the former is much richer; however, the topology discovered by our techniques consists of only those ASes and peerings that may actually be traversed by BGP announcements from p (and thus traffic flows to p) and is thus much more valuable from an ISP's point of view.

Finally, we note that between 13% and 17% of the arcs in the IPv6 per-prefix graphs were not visible in the graphs induced in the full AS graph. The IPv4 figure is only 2%. We suspect that this is due to the much greater redundancy of the IPv6 network, in which the widespread use of tunnels makes it easy to create a much denser connectivity mesh.

6.3 Path Feasibility Determination and Path Preference Comparison

To verify our path feasibility determination techniques, we obtained an initial feasibility graph using withdrawal observation, chose arbitrary paths on the graph starting from the origin AS and ending in a route collector, and applied the nailed-path algorithm to determine which of the paths were feasible. Examples are in Table 3. The "Prefix" column shows the prefix we tested and thus whether the test was performed in the IPv6 or IPv4 network. The "Path" column shows the path we tested. The "UTC Time" column contains the time at which the BGP announcement was sent and the "AS-set" column shows the AS-set announced. The "Observed Path" column shows the path that was observed after BGP propagation and the "Feasible" column shows whether the path was feasible. Note that if no AS-path was observed then we can affirm that a path is not feasible; if another AS-path was observed, it is not possible to determine feasibility (see Section 4.2).

We tested our path preference comparison technique on various paths ending in route collectors. Table 4 shows some of our results, and Fig. 4(b) shows a graphical representation of the experiment in the third row. The second example in the table is interesting because it shows an AS preferring a longer path over a shorter one: between 1103 2607 1275 15589 5397 and 1103 20965 1299 3320 15589 15589 5397, AS 1103 (SURFnet, the Dutch research network), prefers the latter. This suggests that AS 1103 is explicitly configuring its routers to prefer paths coming from AS 20965 (the Géant European research network).

Prefix	UTC Time	Path	AS-set	Observed Path	Feasible
84.205.89.0/24	2005-07-05 11:31:44	8468 6461 701 702 13030 12654 12654	{8210, 8220, 3320, 286, 8447, 20932, 9044, 12793, 13237, 2497, 8434, 513, 8289, 8342, 16034, 1239, 209, 3561, 6762, 2914, 1299, 3356}	8468 6461 13030 12654 12654	Unknown
84.205.89.0/24	2005-07-05 11:36:01	8468 6461 13030 12654 12654	{8210, 8220, 3320, 286, 8447, 20932, 9044, 12793, 13237, 2497, 8434, 513, 8289, 8342, 16034, 1239, 209, 3561, 6762, 2914, 1299, 3356, 701, 702}	8468 6461 13030 12654 12654	Yes
84.205.73.0/24	2005-07-05 13:29:29	2116 1299 1239 13030 12654 12654	{209, 286, 513, 701, 702, 2497, 2914, 3320, 3356, 3561, 6461, 6762, 8210, 8220, 8289, 8342, 8434, 8447, 8468, 9044, 12793, 13237, 16034, 20932}	–	No

Table 3. Path feasibility determination results.

Prefix	UTC Time	Collector	AS-path $\mathcal{P}_1$	AS-path $\mathcal{P}_2$	Observed path	Preferred
2001:a30::/32	2005-02-21 16:30:28	3333	3333 3265 6175 13944 6939 15589 5397	3333 1103 3425 293 3320 15589 5397	3333 1103 3425 293 3320 15589 5397	$\mathcal{P}_2$
2001:a30::/32	2005-04-19 12:38:09	1103	1103 2607 1275 15589 5397	1103 20965 1299 3320 15589 5397	1103 20965 1299 3320 15589 15589 5397	$\mathcal{P}_2$
84.205.89.0/24	2005-07-05 12:31:01	8468	8468 6461 13030 12654 12654	8468 9044 13030 12654 12654	8468 9044 13030 12654 12654	$\mathcal{P}_2$

Table 4. Path preference comparison results.

Acknowledgements

The authors thank Maurizio Goretti and Gabriele Barbagallo from CASPUR for access to IPv6 routing infrastructure and Henk Uijterwaal and the RIPE NCC RIS team for access to IPv4 routing infrastructure and the RIS database. We would also like to thank Monica Cortes from RIPE NCC for providing us with the opportunity to run lab tests on routing equipment. Thanks go to Tim Griffin, Geoff Huston, Ruediger Volk, and Randy Bush for useful discussion.

References

[1] A. Antony. RIS observations. In *RIPE 38*, Jan. 2001.

[2] H. Chang, R. Govindan, S. Jamin, S. J. Shenker, and W. Willinger. Towards capturing representative AS-level Internet topologies. In *Proc. SIGMETRICS '02*, June 2002.

[3] L. Colitti, G. Di Battista, F. Mariani, M. Patrignani, and M. Pizzonia. Visualizing interdomain routing with BGPlay. *Journal of Graph Algorithms and Applications*, 9(1):117–148, Nov. 2005.

[4] L. Colitti, G. Di Battista, M. Patrignani, M. Pizzonia, and M. Rimondini. Active BGP probing. Technical Report 102, Roma Tre University, Dipartimento di Informatica e Automazione, Nov. 2005.

[5] G. Di Battista, M. Patrignani, and M. Pizzonia. Computing the types of the relationships between autonomous systems. In *Proc. INFOCOM '03*, Apr. 2003.

[6] X. Dimitropoulos, D. Krioukov, and G. Riley. Revisiting Internet AS-level topology discovery. In *Proc. PAM 2005*, Apr. 2005.

[7] L. Gao. On inferring autonomous system relationships in the Internet. *IEEE/ACM Transactions on Networking*, 9(6):733–745, Dec 2001.

[8] R. Govindan and H. Tangmunarunkit. Heuristics for Internet map discovery. In *Proc. INFOCOM '00*, March 2000.

[9] T. G. Griffin, F. B. Shepherd, and G. Wilfong. The stable paths problem and interdomain routing. *IEEE/ACM Trans. Netw.*, 10(2):232–243, 2002.

[10] B. Huffaker, D. Plummer, D. Moore, and k claffy. Topology discovery by active probing. In *Symposium on Applications and the Internet (SAINT)*, Jan. 2002.

[11] G. Huston. Interconnection, peering and settlements. *Internet Protocol Journal*, 2(1–2), 1999.

[12] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. Delayed Internet routing convergence. In *Proc. SIGCOMM '00*, Sept. 2000.

[13] Z. Mao, R. Bush, T. G. Griffin, and M. Roughan. BGP beacons. In *Proc. IMC '03*, Oct. 2003.

[14] Z. Mao, J. Rexford, J. Wang, and R. Katz. Towards an accurate AS-level traceroute tool. In *SIGCOMM '03*, Aug. 2003.

[15] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz. Route flap damping exacerbates Internet routing convergence. In *Proc. SIGCOMM '02*, Aug. 2002.

[16] Z. M. Mao, D. Johnson, J. Rexford, J. Wang, and R. Katz. Scalable and accurate identification of AS-level forwarding paths. In *Proc. INFOCOM '04*, Mar. 2004.

[17] Y. Rekhter. A border gateway protocol 4 (BGP-4). RFC 1771, Mar. 1995.

[18] RIPE NCC. Routing Information Service.
http://www.ripe.net/ripencc/pub-services/np/ris/.

[19] Roma Tre Computer Networks research group. BGP probing. http://www.dia.uniroma3.it/~compunet/bgp-probing/.

[20] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *SIGCOMM '02*, Aug. 2002.

[21] J. W. Stewart. *BGP4: Inter-Domain Routing in the Internet*. Addison-Wesley, Reading, MA, 1999.

[22] University of Oregon. RouteViews project.
http://www.routeviews.org/.

[23] J. Xia. Weird aspath in update message. RIPE Routing Working Group list archive, June 2002.

[24] B. Zhang, R. Liu, D. Massey, and L. Zhang. Collecting the internet AS-level topology. *SIGCOMM Comput. Commun. Rev.*, 35(1):53–61, 2005.