BGP Beacons

Z. Morley Mao[*], Randy Bush[†], Timothy G. Griffin[‡], Matthew Roughan[§]

ABSTRACT

The desire to better understand global BGP dynamics has motivated several studies using active measurement techniques, which inject announcements and withdrawals of prefixes from the global routing domain. From these one can measure quantities such as the BGP convergence time. Previously, the route injection infrastructure of such experiments has either been temporary in nature, or its use has been restricted to the experimenters. The routing research community would benefit from a permanent and public infrastructure for such active probes. We use the term *BGP Beacon* to refer to a publicly documented prefix having global visibility and a published schedule for announcements and withdrawals. A BGP Beacon is to be used for the ongoing study of BGP dynamics, and so should be supported with a long-term commitment. We describe several BGP Beacons that have been set up at various points in the Internet. We then describe techniques for processing BGP updates when a BGP Beacon is observed from a BGP monitoring point such as Oregon's Route Views. Finally, we illustrate the use of BGP Beacons in the analysis of convergence delays, route flap damping, and update inter-arrival times.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols—*Routing protocols*

General Terms

Measurement, Experimentation

Keywords

Network measurements, Border Gateway Protocol, convergence time

1. WHAT IS A BGP BEACON?

The Border Gateway Protocol (BGP) [1, 2, 3] is central to the stability and robustness of the Internet. Passive monitoring of BGP updates has resulted in important insights into the dynamics of BGP [4, 5, 6]. Several public sources, such as Oregon's Route Views [7] and the RIPE Routing Service [8], provide BGP updates collected from a large number of points in the Internet. Passive measurements are not sufficient for all purposes and so active techniques have also been employed in the analysis of BGP dynamics [9, 10]. With the active approach, prefixes are announced and withdrawn from the global routing domain while quantities such as convergence time are measured. The main advantage of the active approach is that the *input* to the routing system is known, which allows inferences to be made that would be difficult or impossible with purely passive measurements.

To date, the route injection infrastructure of such experiments has either been temporary in nature, or its use has been restricted to the experimenters. Mounting such an infrastructure is often beyond the means of many interested in this area of research. So we feel that the routing research community would benefit from a permanent and public infrastructure for such active routing probes. We use the term *BGP Beacon* to refer to a publicly documented prefix having global visibility and a published schedule for announcements and withdrawals. A BGP Beacon is to be used for the ongoing study of BGP dynamics, and should be supported with a long term commitment. We describe two collections of BGP Beacons that have been set up at various points in the Internet. We then describe techniques for processing BGP updates when a BGP Beacon is observed from a BGP monitoring point such as Route Views or RIPE. *Anyone* could get data from *any* public or private route monitor to study the Beacon dynamics, as the Beacon updates are globally visible.

We illustrate the use of BGP Beacons with four case studies. Each study relies on the fact that we are monitoring updates that have been generated by a Beacon event. First, we consider the impact of different implementations of BGP on the observations. Second, we investigate the potential that route flap damping [11] punishes "well behaved" routes. Simulation results in [12] have shown that it can punish "well behaved" as well as "misbehaving" routes. Here we use the BGP Beacons to validate those results in the global Internet. This is the first study of the impact of flap damping using real data from the Internet. Even though the BGP Beacons have a fairly long cycle (two hours between each announce or withdraw event), we see that *even announcements* can potentially trigger flap damping as much as 10 percent of the time at some locations on the Internet. For our third study, we present a novel analysis of the inter-arrival times of updates generated by BGP Beacons. Finally, we revisit the convergence-time issues studied in [9, 10].

[*] University of California at Berkeley, email: zmao@eecs.berkeley.edu.

[†] Internet Initiative Japan, email: randy@psg.com.

[‡] Intel Research, email: tim.griffin@intel.com. This work was conducted while Tim was with AT&T Labs–Research.

[§] AT&T Labs–Research, email: roughan@research.att.com.

IMC'03, October 27–29, 2003, Miami Beach, Florida, USA.

Table 1: The PSG Beacons

Beacon	Prefix	Period	Upstream ASN(s)	Beacon AS	Location	Beacon host	Start date	Anchor prefix
1	`198.133.206.0/24`	2 hrs.	2914, 1239 (1)	3130	WA,US	Randy Bush	Aug 10, 2002	`147.28.0.0/16`
2	`192.135.183.0/24`	2 hrs.	3701, 2914	5637	OR,US	David Meyer	Sep 4, 2002	`205.167.77.0/24`
3	`203.10.63.0/24`	2 hrs.	1221	1221	Australia	Geoff Huston	Sep 25, 2002	`165.191.0.0/16`
4	`198.32.7.0/24`	various	2914, 8001	3944	MD,US	Andrew Partan	Oct 24, 2002	`198.6.255.0/24`

2. BGP BEACONS

Currently, there are two groups of BGP Beacons that differ somewhat in implementation. There are four Beacons in the first group, called the *PSG Beacons* because the first was set up at `psg.com`. They are listed in Table 1. These Beacons were set up by Z. Morley Mao with the help of the Beacon hosts. There is a public web site for these Beacons [13] containing Beacon related scripts and Beacon data derived from public BGP monitors such as Route Views. The Beacon *period* is the time between each event at the Beacon, where a *Beacon event* is either an announcement of the Beacon prefix or a withdrawal. Two hours was picked as the period for the first three Beacons because it seemed long enough for most route flap damping to expire [1]. Furthermore, it also introduces minimal impact on the Internet routing plane. As shown later, the amount of noise introduced by the Beacons is miniscule compared the current number of updates observed. We also emphasize that traffic is not affected by the experiment as the prefixes contain no users. The Beacon prefixes must be at least `/24`'s to prevent them being filtered.

The PSG Beacons "hijack" two attributes of the announcements to serve as a timestamp and a sequence number. The aggregator IP attribute, which is an IP address, is set to have the form `10.X.Y.Z` where `0.X.Y.Z` (in binary) represents the number of seconds since the start of the month (GMT). The aggregator ASN attribute is a number that is incremented with each announcement and cycles through the values from 64,512 to 65,635. Note that values of both the Beacon timestamp and the Beacon sequence number are within "private" spaces, so these attributes will not affect routing decisions. Also note that withdrawals do not have such attributes, and therefore do not contain this extra information.

The second group of Beacons, called the RIPE Beacons [14], have been set up as a part of the RIPE Routing Information Services (RIS) following the first several PSG Beacons. Each of RIPE's nine BGP route monitors (in different geographic locations) also acts as a BGP Beacon. RIPE uses Beacon prefixes from `195.80.224.0/24` through `195.80.232.0/24`. Each RIPE Beacon also has a period of 2 hours.

The implementation of these Beacons differs in the following ways. First, PSG Beacons currently have timestamps and sequence numbers, while the RIPE Beacons do not. Second, the PSG Beacons currently have what we call *anchor prefixes* (see Section 3) associated with them, which aid in the pre-processing of update data. Third, the PSG Beacons are not associated with BGP routing monitors, as are the RIPE Beacons. Since the RIPE monitors typically reside at Internet eXchange points, the RIPE Beacons potentially have a much larger number of upstream providers directly announcing. This model arguably does not represent how a customer address block changes on the Internet.

For the rest of this study, we focus on PSG Beacons 1, 2, and 3, as Beacon 4's varying period may result in interaction between consecutive signals. We are not currently using the RIPE Beacons because of the lack of anchor prefixes needed for data cleaning. We leave the study of Beacon 4 and RIPE Beacons as our future work.

2.1 Beacon software

The PSG Beacon daemon software is based on the open-source BGP software router written in perl, and available at `http://bgpd.sourceforge.net`. The original software is purely passive and does not provide any functionality to advertise routes. We modified it to inject routing changes (triggered by a user-defined interrupt signal) to an open BGP session. A cron job is set up to regularly send an interrupt to the Beacon daemon software, so that announce and withdraw updates are sent alternately. In the beginning, the updates were sent every 30 minutes. We quickly discovered that the prefix was suppressed by route flap damping [11] at the upstream provider of the Beacon. Subsequently, the schedule was set to be 2 hours between consecutive updates to minimize the likelihood of route suppression and to allow potentially suppressed route to be unsuppressed before the next Beacon event.

2.2 Terminology for BGP update propagation

We use the term *input signal* to refer to any update generated at a BGP Beacon (either an announcement or a withdrawal). The network of BGP speaking routers can be thought of as a giant non-deterministic signal transducer [15], where each input signal *causes* various *output signals* to be generated at different locations in the Internet. Output signals generated by the Beacon input signals can vary considerably, depending on the Beacon, the monitor point, and the time observed. For example, here is a an output signal for an announcement from PSG Beacon 1, as seen from one peer at Route-Views on January 11, 2003:

Time (GMT)	Type	AS Path
05:00:11	A	8121 19151 2914 1 3130 3927
05:00:39	A	8121 16631 174 1 3130 3927
05:01:08	A	8121 3491 1 3130 3927

This output signal contains three updates. The *signal duration* is the elapsed time from the first to the last update in an output signal. For this example, the signal duration is 57 seconds. A signal containing only one update has a duration of 0 seconds.

2.3 Convergence time

A BGP route monitor may be receiving updates from more than one neighbor. For example, Route Views currently has around 30 peers. For any input signal from a Beacon, a particular route monitor will receive a first update from some peer. For each peer of the route monitor, the *relative convergence time* is the time between the first update at the route monitor (not necessarily associated with the peer in question), and the last output update from that peer. For example, if the first update received from any peer for the announcement event above was 05:00:06, then the example signal has a relative convergence time of 62 seconds. The *End-to-end convergence time* is the time between the sending time of the input signal

[1] The maximum suppress time for route flap damping is by default one hour (Table 5).

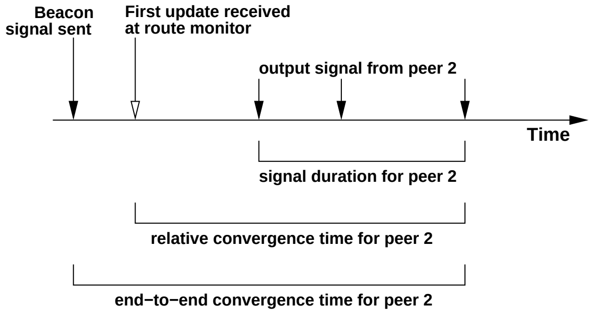

Beacon signal sent
First update received at route monitor
output signal from peer 2
Time
signal duration for peer 2
relative convergence time for peer 2
end−to−end convergence time for peer 2

Figure 1: An illustration of the difference between signal duration, and relative and end-to-end convergence times.

based on the Beacon timestamp in the aggregator field and the last update in a signal. Figure 1 illustrates the difference between signal duration, and relative and end-to-end convergence time.

Ideally, we would like to know the end-to-end convergence times, but this requires clock synchronization. Both the Beacon machine and the monitoring sites should be NTP synchronized according to people who administer these machines. However, to our surprise, we discovered many instances where the receiving timestamp of a Beacon update is smaller than the sending timestamp of the update due to the problem with clock synchronization. We do not have control over the machines at monitoring sites, nor the machines that run the Beacon software, so we sometimes use relative convergence times and signal durations to understand the convergence delays.

2.4 Beacon location terminology

Beacon ASN (Autonomous System Number) as shown in Table 1 refers to the origin AS in the AS path of the updates associated with the Beacon prefix. It is the Autonomous System which the Beacon prefix belongs to. Anchor prefixes are statically nailed down prefixes associated with the same AS and are not affected by Beacon input signals. The *Upstream AS* is the closest tier-1 ISP that provides connectivity to the Beacon. Note, a Beacon can have multiple Upstream ASes. For instance, Beacon 1 is currently multihomed to AS2914 (Verio) and AS1239 (Sprint). Previously it was single-homed to only Verio and multihomed to Verio and Genuity (AS1).

2.5 Public monitoring points

There are several public monitoring points where BGP data is collected from multiple ISPs. Route Views [7] is one such monitoring point, that peers with about 30 different networks and receives default-free updates from these peers. The BGP Beacon owners have arranged that their upstream provider announces the Beacon prefix unaggregated, and so (unless deliberately filtered) they will be globally visible, in particular in all the BGP feeds available at these monitoring points.

3. DATA CLEANING AND SIGNAL IDENTIFICATION

For the Beacon analysis, it is important to clearly identify the output signals by associating the observed BGP updates with a single input signal. We describe in detail a novel methodology to achieve this goal. Not all observed updates related to the Beacon prefix are caused by our input signals. Some of them, for instance, may be caused by routing changes in an upstream AS from the observation point. Given a BGP feed at Route Views, for example, there are typically always some routing updates observed every second. Take a typical day, May 2, 2003, there are more than 5,200,000 updates observed from about 35 peers at Route Views. Such routing changes can be due to a variety of reasons such as routing policy change, link failure, or congestion. In contrast, there are only 760 updates associated with Beacon 1.

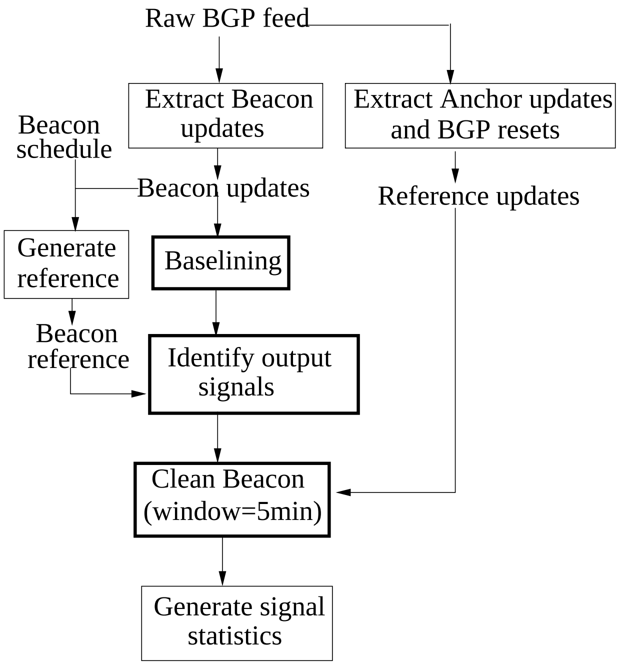


Figure 2: The process of cleaning Beacon data and identifying signals

We use the following example to illustrate the importance of data cleaning. A BGP session requires keep-alive messages to be exchanged between the two neighbors. If a timeout occurs due to congestion or link failure, the BGP session is reset. Upon session reestablishment, the entire routing table is exchanged resulting in a large number of updates observed. Such session resets can occur locally between the local router and the route monitoring software. In that case, such updates do not reflect actual routing changes that affect the forwarding plane of the data traffic. Work by Wang *et al.* [16] demonstrates the importance of filtering out updates due to local BGP session resets in analysis. Very different conclusions are drawn if such updates are not properly filtered.

We generalize the cleaning of session resets to eliminating all routing changes not caused by our measurement input signals. We take a sequence of steps as shown in Figure 2 to clean the Beacon data to identify output signals and compute several signal statistics. This process consists broadly of three steps: baselining, signal grouping, and noise filtering or cleaning. Next we describe each step in detail.

3.1 Baselining

The goal of this step is to process the BGP data such that we can make fair comparisons between peers. We first extract from the raw data the updates associated with the Beacon prefixes, *i.e.*, the Beacon updates. We found some peers at Route Views send out updates related to the BGP community and MED (Multi-Exit-Discriminator) attribute changes (even though policies can be set to prevent such attribute changes from being sent). These peers tend to send more updates, revealing more of the internal dynamics

Table 2: Effect of cleaning on observed *announcement* signals (Route Views): signal count, average duration, delay, and length

Beacon	Before cleaning				After cleaning			
	count	avgDur (sec)	avgDelay (sec)	avgSigLen	count	avgDur (sec)	avgDelay (sec)	avgSigLen
1	33536	27.13	50.60	1.47	33318 (99.35%)	19.36	41.89	1.47
2	34522	9.13	29.56	1.20	33726 (97.69%)	6.75	25.21	1.17
3	32504	10.82	34.99	1.22	32188 (99.03%)	5.77	28.40	1.21
4	39044	41.95	63.66	1.52	37970 (97.25%)	22.79	43.16	1.46

Table 3: Effect of cleaning on observed *withdrawal* signals (Route Views): signal count, average duration, delay, and length

Beacon	Before cleaning				After cleaning			
	count	avgDur (sec)	avgDelay (sec)	avgSigLen	count	avgDur (sec)	avgDelay (sec)	avgSigLen
1	33443	37.88	100	2.07	33261 (99.46%)	32.98	90.09	2.07
2	33860	45.24	109.23	2.19	33344 (98.48%)	42.94	94.38	2.19
3	32379	59.16	120.64	2.55	31182 (96.30%)	56.36	114.40	2.55
4	36633	96.33	139.63	3.43	35776 (97.66%)	75.65	115.90	3.41

of the AS, since these updates typically reflect the BGP dynamics within the last-hop AS. We also found that some peers of Route Views send out consecutive updates that are identical. About a third of the peers at Route Views send either duplicates or updates that differ only by these two attributes.

To make comparisons between peers more fair, any updates that are either identical to the previous update or differ only in community or MED attribute values are eliminated during the baselining step. This reduces about 15% of the updates for all Beacons based on Route Views data. We verified that eliminating such updates has little or no effect on our analysis of inter-arrival update time (Section 6) and convergence delay (Section 7). However, as a result of baselining, the route flap damping analysis (Section 5) provides only a lower bound, and the analysis of signal length may also be an underestimate.

3.2 Signal identification

For the ease of analysis, we do additional processing to group updates together according to the input signals. To achieve that, we create *Beacon references* that identify the starting time of each output signal for a given input signal. In principle, any output signal received will have a timestamp greater than or equal to the Beacon reference timestamp, which we know from the Beacon schedule, or the timestamp in the BGP update aggregator field in case of announcement event. However, as mentioned before, the clocks of the Beacon machines and the monitoring machines must be synchronized for these timestamps to be used in unison.

We thus resort to several algorithms to generate the Beacon references. If the monitoring sites receive the BGP feed directly from the AS that hosts the Beacon, the Beacon AS, then the timestamps of the messages from that AS can be used in the references. A BGP monitoring site, such as Route Views, usually establishes multihop EBGP sessions with several different ISPs. The output signal coming from the Beacon AS almost always arrives first, as it typically travels through the fewest number of routers and smallest distance. Two of the Beacons (Beacon 1 and 3) fall into this category. Typically, the Beacon AS produces very clean output signals that consists of a single announcement given an input announcement signal, and similarly a single withdrawal given an input withdrawal signal.

It is important to point out that if there are no issues with time synchronization, the Beacon timestamps in the aggregator field provide very accurate timestamps for the references that can be used for all monitoring sites. The timestamps based on the Beacon AS are specific to the monitoring site and require that there is a BGP feed from the Beacon AS to the monitor. If other sites use such Beacon references generated from a different site, there may be offsets from the reference timestamps for the output signals, *i.e.*, some output signals may start earlier than specified due to clock differences.

If the BGP feed from the Beacon AS is not available at the monitoring sites, heuristics are used based on the Beacon schedule to determine the start of a new Beacon signal. The period of Beacon announcement is purposely set to be two hours for the first three Beacons. We therefore expect that most sites converge on the final route long before the next input signal is injected. One simple heuristic is to look for large gaps between updates and use the Beacon schedule as a reference to identify the starting times of the output signals. It is easy to distinguish the start of a new announcement signal with the help of the sequence number in the aggregator field. The start of a withdrawal output signal is identified using the timing heuristic. As a heuristic, we eliminate from the Beacon output signal any Beacon updates that occur 4000 seconds after its preceding update. This eliminates some of the updates caused by routing events other than our injected Beacon signals. The next noise filtering step will also take care some of these updates separated by large inter-arrival delay from the previous updates. This decision is also justified by the fact that the default route flap damping setting should delay convergence by at most one hour (Section 5).

3.3 Noise filtering/cleaning

To differentiate the updates caused by our injected routing changes from updates caused by other effects, we propose the use of an *anchor prefix* to detect routing changes other than those induced by the Beacon itself. As mentioned before, an anchor prefix is a statically nailed down prefix belonging to the Beacon AS. An anchor prefix could be an unused prefix purposely announced for the use with the Beacon prefix, or it could be a prefix containing live hosts — the important thing is that it is expected to be fairly stable. Anchor prefixes serve as calibration points to identify non-Beacon routing changes. When there are no such routing changes, we will observe no routing updates associated with the anchor prefix. Anchor prefixes originate from the same AS as the Beacon, so that any routing changes experienced by the anchor prefix most likely are also experienced by the Beacon prefix and are therefore not caused by our purposely injected routing changes. In general, we found Beacons hosted by larger ISPs, e.g., AS1221 in the case

Table 4: A comparison between a Cisco and a Juniper router. The table shows average statistics (including the average signal length, or number of updates, the average duration, the average time between updates during a sequence of events, and the percentage of inter-arrival times less than 26 seconds), for announcement 'A', and withdrawal 'W' events, for the two known last hop routers at Route Views.

		signal length		duration		inter-arrival		% of short inter-arrivals	
Peer	Type	A	W	A	W	A	W	A	W
147.28.255.1	Cisco	1.20	2.07	6.79	48.4	34.8	45.4	1.56	0.44
147.28.255.2	Juniper	1.50	2.49	7.13	44.3	14.2	29.6	12.76	4.37

of Beacon 3 to be more stable as they have more redundant network connectivity. This is evidenced by the observation that Beacon 3's anchor prefix files are typically one third smaller than other Beacons.

The Beacon signals are cleaned by deleting signals that can be affected by unexpected routing changes as experienced by anchor prefixes. Such cleaning is done on a per Beacon and per peer basis, as each Beacon has a different anchor prefix and each peer experience different routing changes. For each anchor prefix update, we construct a window starting W minutes before the update time and ending W minutes after the update time. Any Beacon prefix output signal that has an overlap with the window is ignored, as it is likely affected by external routing changes. Based on empirical observation of the Beacon study, it takes on the order of 2-3 minutes for a routing change to become globally visible. We tried several values of W larger than this interval, and settled on 5 minutes because we found that the total number of output signals remained almost constant for $W > 5$ (based on Route Views data). We also apply such a window around certain BGP STATE messages that indicate BGP session resets between the route monitor and its peers. Note, we do not need a larger window to encompass the entire duration of the table exchange following the session reset [16], because we only care about changes to the Beacon prefix. The cleaning process deletes on average 2 to 3 percent of updates.

Tables 2 and 3 show the effect of cleaning on observed signals in terms of signal count, average signal duration, delay, and signal length for Route Views data. They demonstrate the importance of cleaning. For all four Beacons, less than 5% of the signals have been deleted after cleaning. Overall, the average signal delay and signal duration have decreased for both announcement and withdrawal signals after cleaning. In some cases, the decrease is as much as 50% of the original value. However, the signal length remains mostly the same after cleaning. This means that cleaning tends to remove a few outliers with large inter-arrival time or long signal duration.

For each of the four PSG Beacons, we perform this sequence of steps and finally generate a set of signal statistics such as relative convergence time, signal duration, end-to-end convergence time, and number of updates. The following sections present the analysis we have performed on these results.

4. BGP IMPLEMENTATION IMPACT: CISCO VS JUNIPER

One question of interest is how much impact the different BGP implementations have on BGP dynamics. The BGP specification (RFC 1771) [1] defines the protocols to be used, but not how they should be implemented, and in some cases BGP implementations have contained bugs, resulting in nonconformance to the specification. There are implementation discrepancies between different router vendors, even between models, and software versions from the same company. In addition, there are configurable parameters which may impact behavior (*e.g.,* the MinRouteAdverTimer). As specified by the BGP RFC, MinRouteAdverTimer specifies "the minimum amount of time that must elapse between advertisement of routes to a particular destination from a single BGP speaker." In an ideal world, these differences would have little impact on the operation of BGP, but studies (*e.g.,* [17]) have shown that at least some of these differences may have a big impact.

In this section we consider one example of the impact that differences in implementation may have on the behavior of BGP. Namely, we consider the difference between Cisco and Juniper implementations of BGP. The decision to compare these two (out of all the possibilities) is also motivated by the fact that we know the make and model of two last hop routers (as seen by Route Views). In addition, Cisco and Juniper are currently the two dominant router types in the core of the Internet. In general, this information is not publicly available, but the last hop routers corresponding to Route Views Peer 147.28.255.1 and 147.28.255.2 are known to be Cisco and Juniper routers, respectively. They belong to the same network and are both located in Seattle, WA.

Table 4 presents a comparison between Juniper and Cisco routers (as seen from Beacon 2). The table shows that the Juniper router sends about 25% more updates, has a similar update duration, and a substantially smaller average inter-arrival time for updates (around 60% of that for Cisco routers). The most startling difference, though, is in the number of short (< 26 second [2]) inter-arrival times, which is much greater for the Juniper router.

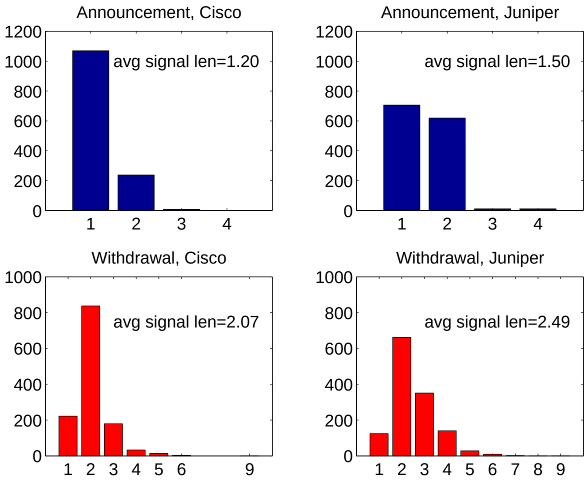


Figure 3: Distribution of signal lengths of Beacon 2: Comparison of the numbers of updates per output Beacon signal for two known Cisco and Juniper routers from the same peer AS.

[2] 26 seconds is used here rather than 30 seconds (the default setting of MinRouteAdver Timer in Cisco routers), because jitter is typically applied to such timer values. We observe that many inter-arrival values cluster around 26 seconds.

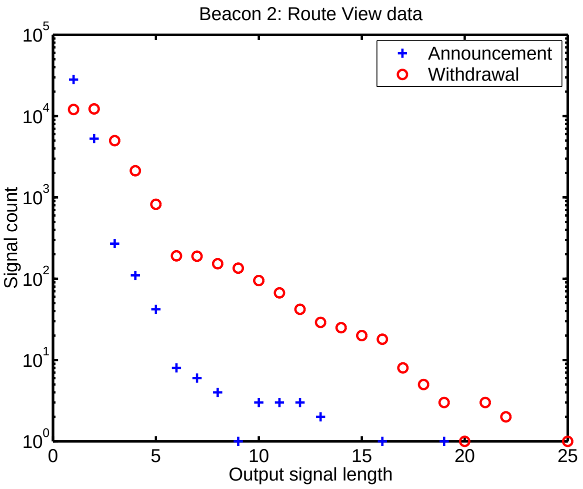


Figure 4: Beacon 2's signal length distribution: withdrawal input signals clearly produce longer output signals.

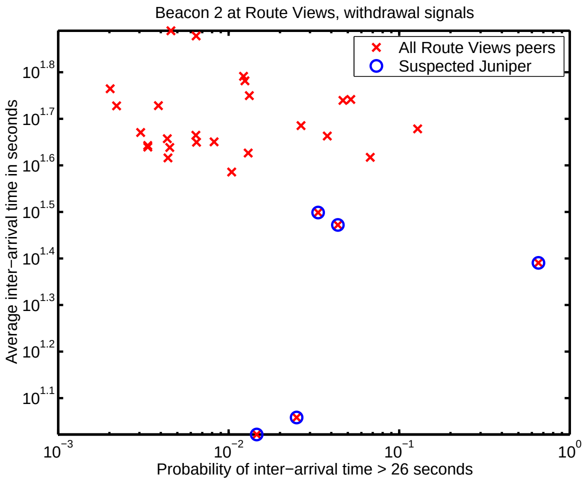


Figure 5: A scatter plot showing the routers classified as Juniper-like. Although there is not a completely clear distinction in the plot, detailed examination of the update sequences shows that the marked peer routers show similar characteristics to the known Juniper router.

The differences can be explained by the fact that, by default, the MinRouteAdverTimer is turned off in Juniper routers [18]. It is common practice for users to leave default settings alone, unless they have particular reasons to do otherwise. We know from discussion with the administrator of these routers that the defaults are used. Confirming the results in [17] we observe that having a small, or zero MinRouteAdverTimer can result in large numbers of additional updates. Figure 3 shows the distributions of signal lengths for the two known routers, for announcement and withdrawal events, to further illustrate the difference. Similar to observations made in [9], the MinRouteAdverTimer spaces out the updates, thereby potentially increasing the convergence time, but reducing the number of updates. Withdrawal input signals on average introduce a larger number of updates compared to announcement as shown in Figure 4.

Given the obvious difference between the two router types above, the question naturally arises, "can we distinguish the other last hop routers as being of one type or the other?" Figure 5 suggests that we can separate the two, though not perfectly. We could identify candidates from this figure, but needed to do a final verification by examining the update sequences in detail to confirm the findings. The routers that appear to be Juniper routers are marked on the plot.

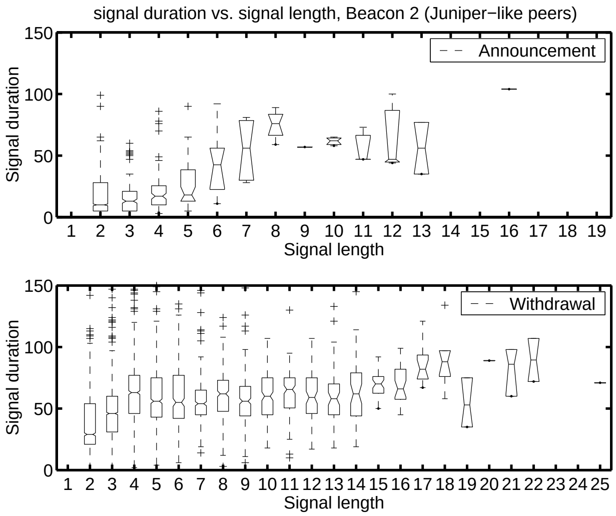


Figure 6: Beacon 2's signal duration distribution for each signal length for Juniper-like peers (top shows announcements, bottom shows withdrawals)

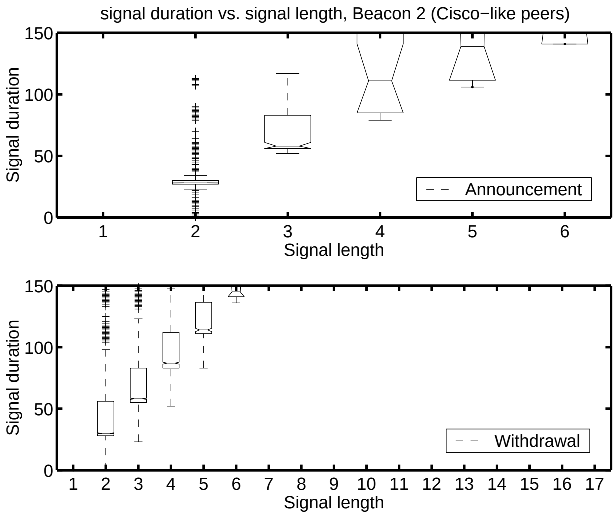


Figure 7: Beacon 2's signal duration distribution for each signal length for Cisco-like peers (top shows announcements, bottom shows withdrawals)

Using this separation, we examine in more detail some of the observed behavior of these router types. The box plots in Figures 6 and 7 demonstrate the difference in signal duration distribution as the signal length increases. For Juniper-like routers (Figure 6), there is no clear dependence between the signal duration and signal length. As the signal gets longer, the duration increases only

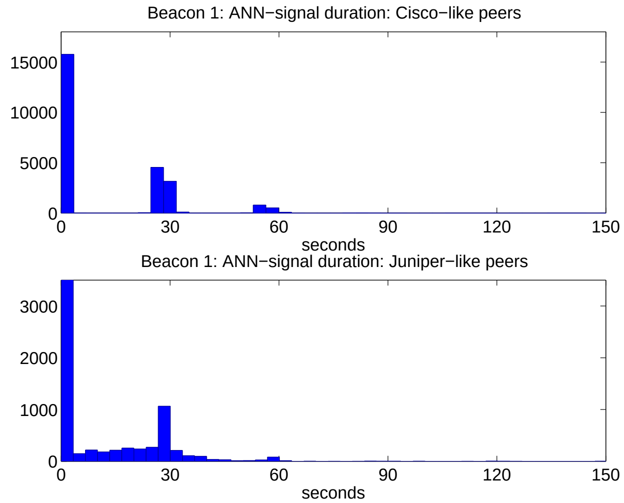


Figure 8: Comparing Beacon 1's announcement signal duration distribution for Cisco- like peers with that for Juniper-like peers

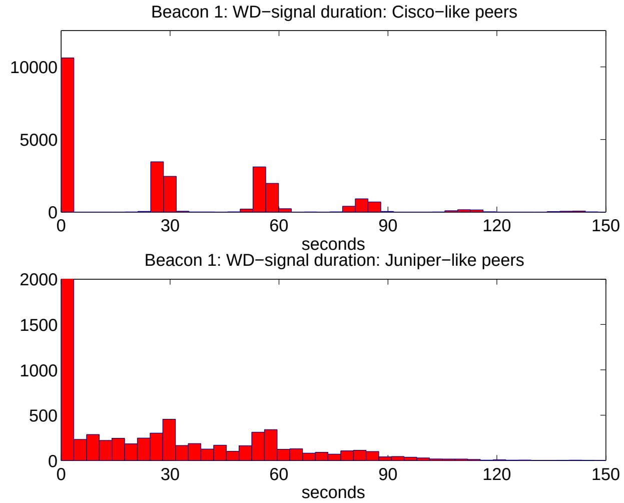


Figure 9: Comparing Beacon 1's withdrawal signal duration distribution for Cisco- like peers with that for Juniper-like peers

slightly and keeps the values around 40 to 60 seconds. This is because Juniper's default rate-limiting algorithm allows updates to be sent in bursts. In clear contrast, Cisco-like routers (Figure 7) show the 30 second rate-limiting behavior: there is a linear increase with slope of around 30 seconds in the duration as the signal gets longer. Furthermore, signals generated from Cisco-like peers have shorter signals compared to that of Juniper-like peers.

Beacon 1's signal duration distribution is shown in Figures 8 and 9. Again we differentiate between Cisco-like and Juniper-like peers. Announcement signals have in general shorter durations compared to withdrawal signals. In addition, we again see the signal duration for Cisco-like peers to be multiples of 30 seconds. The distribution for Juniper-like peers is much more spread out, with a smaller peak at 30 seconds that could be easily caused by upstream Cisco-like routers. A large number of signals converge within 30 seconds for both types of routers.

Table 5: Default route flap damping parameter settings. The differences between Cisco and Juniper are shown in boldface.

RFD parameter	Cisco	Juniper
Withdrawal penalty	1000	1000
Readvertisement penalty	**0**	**1000**
Attributes change penalty	500	500
Cutoff threshold	**2000**	**3000**
Half-life (min)	15	15
Reuse threshold	750	750
Max suppress time (min)	60	60

5. ROUTE FLAP DAMPING ANALYSIS

Route flap damping [11], abbreviated as RFD, is one of the two mechanisms in BGP aimed at achieving routing stability. Flap damping punishes unstable routes or routes that change frequently by suppressing them. It is designed to deal with routes that are unstable on a long time scale. In contrast, the other mechanism, the minimum route advertisement timer (MinRouteAdverTimer) is designed to act on routes that are unstable on a short time scale. specifies The reason for delaying the updates is to allow consecutive updates to be batched together to reduce update traffic. These two mechanisms can interact: the timer determines the number of updates that are propagated during the convergence process, and this number directly affects the likelihood that route flap damping is triggered.

We now briefly describe how route flap damping works. The router keeps track of a penalty value for each route received from its EBGP neighbors. The value is kept on a per route and per neighbor basis. Whenever the route changes, the penalty value is incremented for the corresponding neighbor, with the increment depending on the type of change. Both Juniper and Cisco routers have their own specific *penalty increments* as shown in Table 5. The penalty value decays exponentially over time, with the decay rate given by the *half-life* parameter (also shown in Table 5), which determines the amount of time it takes for the penalty to decrease to half of its original value. If the penalty ever exceeds the *cutoff threshold*, the route is considered suppressed, that is, no longer eligible as a usable route to forward traffic. A withdrawal is sent out if the route was previously used in the forwarding table. Subsequently, if any other updates for this route are received, they will not be propagated. There is a limit on how long the route can be suppressed given by the *max suppress time*. Once the penalty value decays below the *reuse threshold*, the route is considered usable again, and if it would be the best route, a new announcement is sent.

It has been shown [12] in simulations and a commercial router testbed that in certain topologies, no matter how large the MinRouteAdverTimer is, there are sufficient updates induced by a single route change to trigger route flap damping. This means that a *single* router reboot, which translates to a withdrawal message followed by an announcement message, can cause the route to be suppressed somewhere on the Internet. As there is no feedback in the flap damping mechanism, it is difficult to determine which router suppresses the route. If the route suppressed is the only route to reach a destination prefix, then the destination becomes unreachable from the network that suppresses the route. It is thus very important to understand how likely this occurs in today's Internet.

It is very difficult to understand the extent at which route flap damping can suppress well-behaved or stable routes on today's Internet. The difficulty arises due to the complexity in inferring the root causes of BGP updates observed in passive measurements [15]. The Beacon infrastructure provides a perfect medium for doing such a study, as routing changes are injected at known times and locations. Assuming the Beacon prefix routes are not suppressed, we can simulate how likely it is for a single routing change to cause the route to be suppressed. We implemented the route flap damping algorithm using the Cisco and Juniper default parameters and calculated the percentage of observed signals that could trigger route suppression at the monitoring sites [3]. In fact, this is an underestimate of how often a single route change can trigger route suppression, as some networks may suppress the route and reduces the amount of updates propagated. Based on Route Views data, we observe that about 5% of input signals would be suppressed measured across all Route Views peers as shown in Figure 10. Some peers are much more likely to suppress the route than others because the large number of updates generated. The columns labeled with "peer max" in the Figure indicate the maximum percentage of suppressed signals on a per peer basis. Some peers would suppress close to 45% of all signals received.

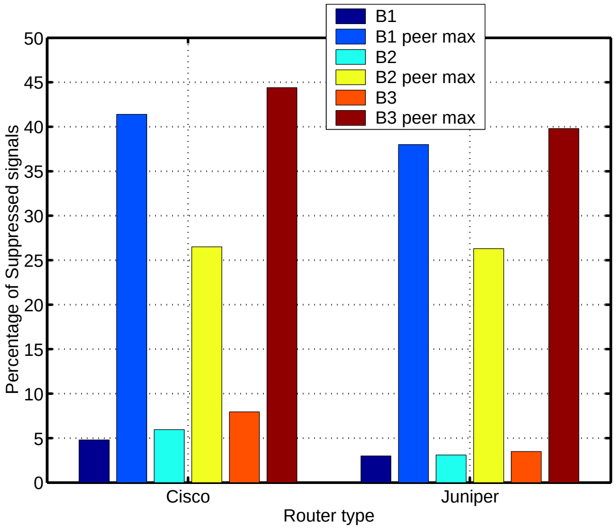


Figure 10: Overall percentage of suppressed signals due route flap damping for each Beacon and on a per peer basis for Cisco and Juniper.

Figure 11 breaks down the suppressed signals between announcements and withdrawals. Since withdrawals typically generate more updates, a much higher percentage of withdrawal signals are suppressed compared to announcement signals. In fact, at some peers, close to 90% of all withdrawal signals can trigger route suppression using Cisco's default setting. Overall, Cisco is more aggressive in suppressing routes than Juniper based on Route Views data for our three Beacon prefixes for both announcement and withdrawal signals. In fact, this may not always be the case. Although Cisco's cutoff threshold is lower than Juniper's, it does not punish a route readvertisement or an announcement that is preceded by a withdrawal. Consequently, Cisco is more likely to suppress routes with the following update patterns: "AAAW" (A: announcement, W: withdrawal). There is no readvertisement in such patterns; therefore, Cisco's lower threshold will increase the probability of route suppression. Juniper is more likely to suppress routes with update patterns such as "AWAWA" where there are readvertisements. From our data, the former pattern is much more prevalent; therefore, Cisco is overall more aggressive in route suppression than Juniper using the default RFD setting.

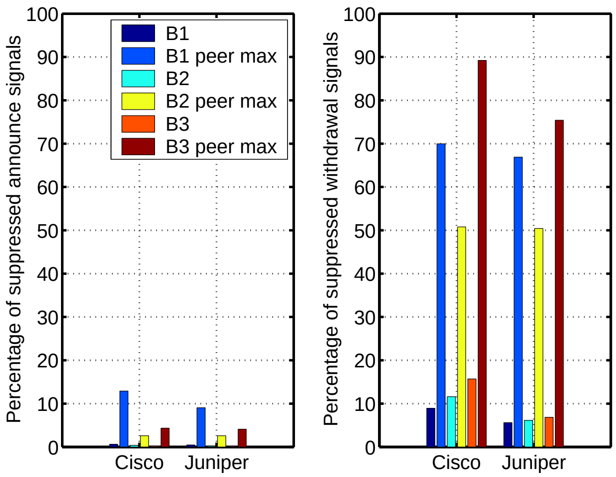


Figure 11: Percentage of suppressed Beacon signals due to announcement and withdrawal

Our analysis only provides a lower bound for the percentage of suppressed signals, as some routers in the network may already have suppressed the Beacon prefix, resulting in fewer updates observed at the monitoring site. Furthermore, the data cleaning step eliminated updates that differ only in community and/or MED attributes from previous updates. Certain update patterns also indicate the presence of route suppression by some intermediate routers from a given monitoring point. For an announcement input signal, we sometime observe inter-arrival time between 1000 and 3600 seconds. After the long timeout, the update sequence always ends with an announcement. Such a long timeout is extremely unlikely to be caused by propagation delay, router processing delay, BGP path vector effects, or MinRouteAdverTimer values, even if they are accumulated along each router hop. Flap damping suppression duration is typically on the order of tens of minutes which matches well with the timeout values we observe. In fact, based on the default Cisco parameter setting, the minimum suppression duration is about 30 minutes. For Juniper, this value is about 21 minutes.

If the data are cleaned properly and there are no missing updates, such timeout values provide a good indication that route suppression has occurred. The final announcement can be caused by the re-announcement of the route after its penalty has decayed below the reuse threshold. And the long break indicates the duration during which the route is suppressed. We show two such examples below (Tables 6 and 7). In the first one, the last update before the long break is an announcement. In the second example, a withdrawal occurs before the long break. In both cases, the timeout value is about 40 minutes or 2400 seconds. We observe three transient routes from peer 216.18.31.102 in case 1. The final route goes through AS701, and it is very likely that the long break is due route flap damping occurring along the AS path "6539 701 1 3130 3927". ASes 6539, 701 or 1 may have suppressed the route originated by AS3927. AS3130 could not have suppressed the route, because the alternate path still goes through it. As soon as this route is un-

[3] The route monitor itself typically does not implement route flap damping algorithm.

suppressed, it is chosen as the preferred route by AS6539, which apparently has at least three alternate routes to reach the destination AS3927 (the source ASN). It very likely only suppresses the route learned from AS701; therefore, its connectivity to the Beacon is not affected. In general, if the last update before the timeout is an announcement, it indicates that there is an alternate path available from the monitoring point. In the case of withdrawal, flap damping has affected all the available paths from the monitoring location.

The latter is exemplified by case 2 shown in Table 7. The route with AS path "11608 2914 3130 3927" appears to be preferred over the alternate route with AS path "11608 2914 1239 3130 3927". In this example, it is very likely that AS11608 suppresses the route received from AS2914. AS2914 is less likely to suppress the routes coming from both ASes 3130 and 1239, as this requires a router AS2914 to have received sufficient updates from both these neighbors. AS3130 is also unlikely to have suppressed the route from the origin AS 3927, as it is very close to the Beacon source and gets fewer updates. In general, ASes farther away from the origin AS are more likely to experience more updates due to richer network connectivity to the source and thus more likely to suppress the route.

Table 6: Case 1: observation from peer 216.18.31.102 on Apr 3, 2003 for Beacon 1:

Time (GMT)	Type	AS Path
23:00:17	A	6539 3561 1 3130 3927
23:00:44	A	6539 701 1 3130 3927
23:01:14	A	6539 3602 16914 852 1 3130 3927
23:42:46	A	6539 701 1 3130 3927

Table 7: Case 2: observation from peer 207.246.129.14 on Sep 17, 2002 for Beacon 1:

Time (GMT)	Type	AS Path
22:38:45	A	11608 2914 1239 3130 3927
22:39:13	A	11608 2914 3130 3927
22:39:40	W	
23:24:26	A	11608 2914 3130 3927

In our data, close to 1% of update sequences were indicative of route suppression in an intermediate router. We purposely separated Beacon input signals by two hours, as the maximum suppress time is one hour. Therefore, for an announcement input signal, we should always expect to observe an terminating announcement unless data are missing or the anchor prefix is unstable. Similarly, the output signal of a withdrawal input signal should always terminate with a withdrawal.

6. INTER-ARRIVAL TIME ANALYSIS

In this section, we explore an aspect of BGP dynamics not considered in any previous work. A BGP update sequence has been typically considered (within this paper as elsewhere) to be a sequence of N updates, over some duration, but little attention has been given to the distribution of updates within this time interval. However, within this sequence the updates are spaced according to an *inter-arrival* time distribution. We provide a brief examination of some of the properties and ramifications of this distribution as another example of how BGP Beacon data may be used in the analysis of Internet routing dynamics.

As noted above, the inter-arrival times for Juniper-like and Cisco-like routers are different, and so we shall consider these separately here. First, consider the Cisco-like cases. Figure 12 shows log-log plots of the Complimentary Cumulative Distribution Function (CCDF) of the inter-arrival times for updates (including announcements and withdrawals), for each of the three Beacons. The x-axis is the time between updates (in seconds), and the y-axis is the probability that an interval exceeds this time. Note that in the results here, intervals are rounded up to the nearest second, so that all intervals less than 1 second will appear as one second.

Despite the difference between Figure 12 (a) and (b), we see two regimes in both. This is most clear in the distributions for the Cisco-like routers. For each of the Beacons we see a *body* region of step like decrease at slightly less than 30 second intervals (the vertical dashed line are drawn at exactly 30 second intervals). The 30 second intervals seem to match well the default value of the MinRouteAdverTimer described above. The second *tail* region seems to level out the distribution, followed by a sharp decrease, before the distribution is truncated at around 3500-3600 seconds. The cut off between the two regions appears to be around 100 seconds. The Juniper-like routers show a similar division of the distribution, though the body part is less step like.

A natural hypothesis to make is that the two components of this distribution arise from different basic causes. We shall examine this hypothesis by trying to understand what two processes might produce these results. Our first approach is to do some distribution fits to the data, to gain an understanding of what we are seeing.

Figure 13 shows one such fit (on log-log and semi-log axes), done by eye for the Cisco-like routers. The fit combines three components. First, we model the step function taking the number of steps to be given by a geometric distribution, with the length of the steps to be 28 seconds (plus a small Gaussian jitter). The second component is a small mass at 1 second, because of the discretization of the interval times, in particular all times below one second are rounded to one. The third component is a shifted exponential distribution, which matches the tail of the distribution. The figure shows the fitted curve as the dashed line. Note that the fit on the log-log plot is very visually satisfying, as it is on both of the semi-log graphs.

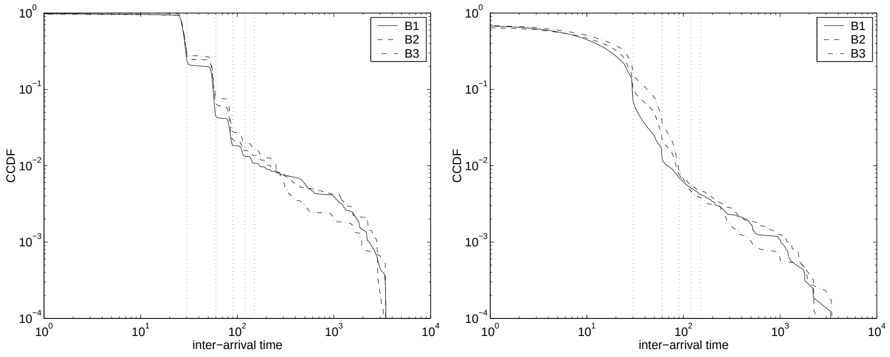
Note that, in this fitting we are not seeking to gain a precise knowledge of the parameters involved. In fact, given the number of parameters we have to play with here, the data are not sufficient to achieve a precise parameterization (one can always fit a sufficiently complex curve to any data set). The aim is to gain an understanding of the processes that might be involved by seeing what type of distributions they generate. However, for the benefit of the reader we provide a precise definition of the distributions and parameters used to produce the fitted distribution.

The distribution is generated using:

$$X = \begin{cases} 28 * (1 + \text{Geom}(0.81)), & \text{with probability } 0.9524, \\ 1, & \text{with probability } 0.0381, \\ 90 + \text{Exp}(970), & \text{with probability } 0.0095, \end{cases}$$

where $\text{Exp}(970)$ refers to an exponentially distributed random variable with mean 970 (seconds), and $\text{Geom}(0.81)$ refers to a geometric distribution with parameter $p = 0.81$, and therefore mean $(1 - p)/p = 0.2346$.

Rather than trying to consider the exact nature of the distribution above, let us try to understand the implications of this form of



(a) Cisco-like last-hop routers.

(b) Juniper-like last-hop routers.

Figure 12: The inter-arrival time distribution for each of the three Beacons as seen from Cisco-like and Juniper-like routers. The vertical dotted lines are drawn at 30 second intervals.

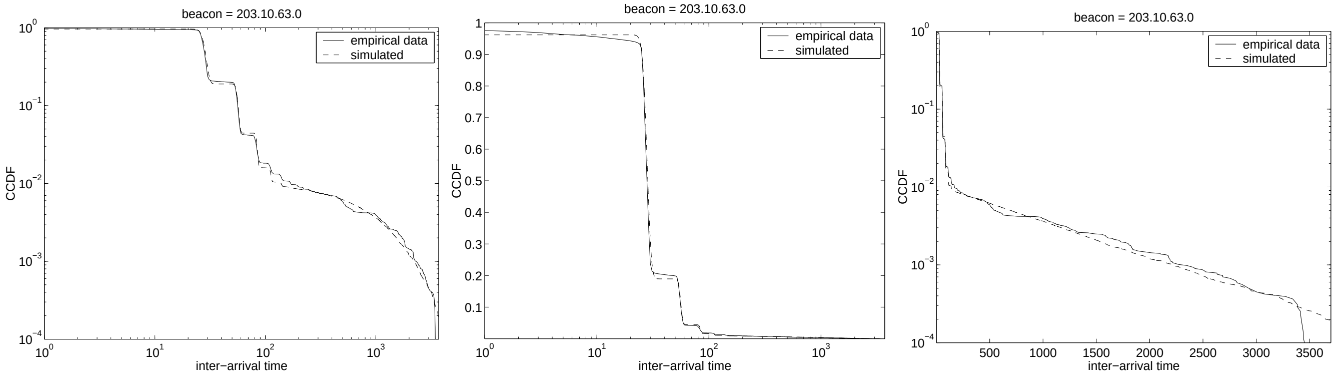


Figure 13: Empirical inter-arrival time distribution for Cisco-like routers compared with simulated values of Beacon 3

distribution. This type of distribution could arise as a result of a random mixing between three different random variables:

- **geometric distribution:** The first part of the distribution is generated by a series of steps, each nearly 30 seconds apart. The action of the MinRouteAdverTimer would certainly explain the first step – this timer prevents a router from sending an update within some time of the last prior update of the prefix. The typical default (rarely changed in practice) of a Cisco router is 30 seconds, and the router adds jitter to this amount to prevent any possible synchronization effects. Hence, one would naturally expect a delay of around 30 seconds between announcements – hence the first step in the distribution. The second and further steps can then be explained as multiple MinRouteAdverTimer intervals. We can suggest a simple reason why one might see such gaps: Cisco's implementation of the MinRouteAdverTimer is not 'per prefix', but rather 'per peer'. That is, the router will send a series of announcements to a peer, and then wait for the MinRouteAdverTimer. Hence, an announcement which arrives 'late' due to delays in prior transmission (through a series of AS's) can miss the next batch of transmissions, and be delayed for a step. This can happen multiple times as an announcement traverses the Internet, and so we see multiple missed steps. The interesting thing is that this process can be so simply modelled by a geometric distribution, in which the probability of missing the next step does not depend on how many steps have already been missed.

- **mass at one:** The discretization of timestamps to integer seconds results in discretization of the inter-event times – hence times between zero and one will tend to be lumped into a point at one. Thus we need to include a probability mass at one, which is indicative of the number of very short inter-arrival times.

- **shifted exponential:** This is the most puzzling part of the distribution, partly because we have the least data in this region (less than 1% of the distribution falls into the tail). The lack of a large data set, and the fact that these are truncated (by only using data sets for which the total time is roughly less than one hour), means that one can model the tail almost as accurately using a power-law distribution. However, of the known BGP mechanisms, the most likely source of these delays is route flap damping, discussed above. This seems even more likely because of the truncation of the distribution near one hour, the maximum suppress time.

There is a natural test for the cause of the step like body of

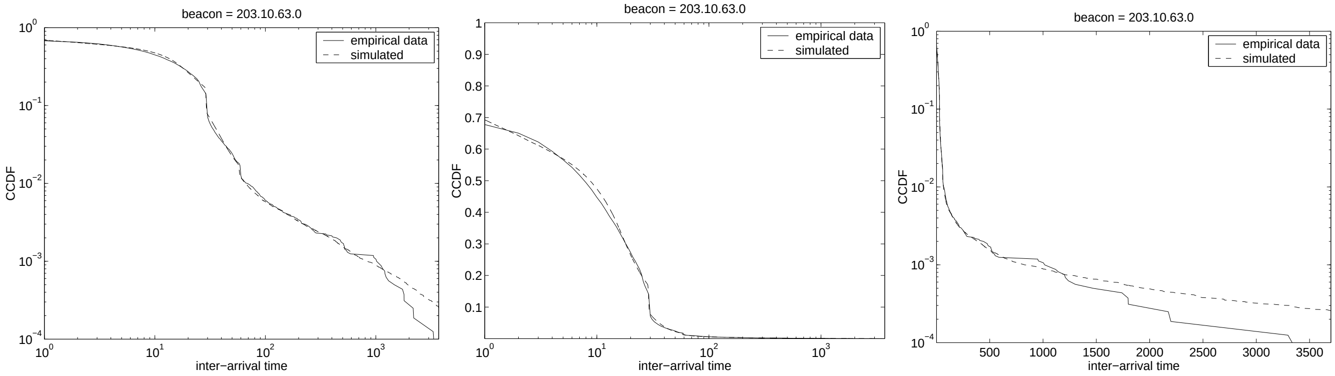
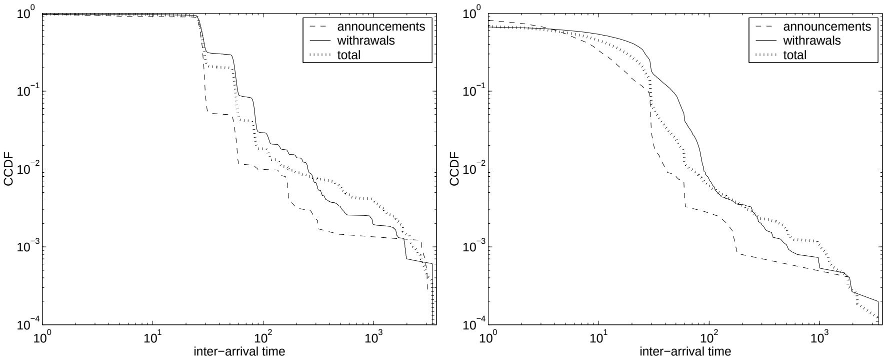


Figure 14: Empirical inter-arrival time distribution for Juniper-like routers compared with simulated values of Beacon 3



(a) Cisco-like last-hop routers.

(b) Juniper-like last-hop routers.

Figure 15: Inter-arrival time distribution for Juniper-like and Cisco-like routers separated out by announcement and withdrawal signals

the distribution above. That is to consider the Juniper-like last hop routers, for whom we suspect the MinRouteAdverTimer is not used.

Figure 14 shows the plots of the inter-arrival time distribution for the Juniper last hop routers (on log-log and semi-log axes). This appears to be somewhat different from Figure 13. We can see much less evidence of a step-like decrease, and significantly more of the mass of the distribution appears before 30 seconds. Figure 14 also shows a fit to the distribution, though this time the fit has four components:

- a geometric distribution (step size ~ 30),
- mass at one (probability 0.1765);
- a convolution of a uniform (over $[0, 16)$) and exponential distribution (mean 10),
- a power-law tail (with $\alpha = 0.85$.).

In this case the much smaller (probability 0.0882) residual geometric component of the distribution can be easily explained by Cisco routers earlier in the path of the updates. The first part of the distribution, formed from the convolution of uniform and exponential distributions appears to be the fundamental difference between the two. It seems clear from these results the the MinRouteAdverTimer is responsible for the inter-arrivals times typically being multiples of approximately 30 seconds.

We display the power law fit to the tail for the Juniper-like routers, as in this case it appears to be better than an exponential fit, but note that it is not really possible to rigorously distinguish the two given the small amount of data in the tail, and the narrow range of scales across which it traverses. It will be interesting to study this as more data becomes available, to determine which model is better, as well as better determining the cause of this tail.

Finally, the above graphs lump two components together, the inter-arrivals for announcement, and withdrawal signals. It might be possible that the two components seen are actually derived from these two separate types of events. In Figure 15 we show the separate distributions for announcement and withdrawal signals separated (and compared to the overall distribution), for Juniper-like and Cisco-like routers. Note that for both Cisco-like and Juniper-like routers the three curves all retain the same basic characteristics, though the curves for the announcement events both drop more sharply, and level off more than those for the withdrawals. The overall curve is more like the withdrawal curve, largely because withdrawal events generate more updates, and therefore more inter-arrival measurements.

The above analysis provides us with one more insight that might not be immediately obvious. The Juniper-like routers appear to have a classic "heavy-tailed" distribution. This is important because it immediately explains our earlier finding that the convergence times are not well correlated with the number of updates seen.

The sum of a series of heavy-tailed random variables is well known to also have a heavy-tail, but a less well known result is the fact that the heavy-tail of the sum arises not from a sum of medium size events, but from single large events. The intuitive explanation for this effect is that "rare events happen in the most likely way". In this context, we may interpret this to mean that the longer convergence times are not typically the result of a long series of updates that take a long time to converge, but rather the result of a single long inter-arrival time between updates. The data seem to validate this intuition. If removing the heavy-tail from convergence times is considered important, then one must first concern oneself with the causes of the heavy-tail in the inter-arrival time (for instance flap damping), rather than trying to reduce the number of updates.

The Cisco-like routers also have a somewhat heavy-tailed distribution (even if it cannot be modeled as power-law, there is a significant probability of an inter-update time several orders of magnitude larger than the typical time – thousands of seconds as opposed to around 30). However, in this case the low probability of small events and large chance of 30 inter-update times masks the previous results, so that there is a direct correlation between the number of updates, and the signal duration.

7. CONVERGENCE ANALYSIS

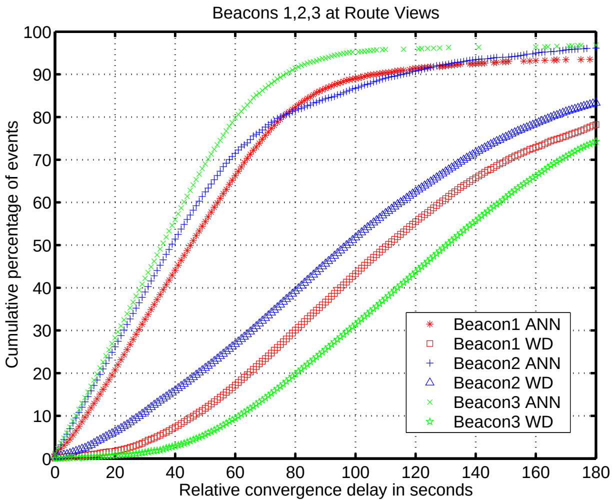


Figure 16: Cumulative distribution of relative convergence times for all three Beacons for both announcement and withdrawal signals

Given the insight we gained from our previous three sections, we now revisit the work by Labovitz *et al.* [9], conducted about three years ago. [9] analyzes BGP convergence behavior of four types of events: announcement (Tup), withdrawal (Tdown), fail-over to a shorter route (Tshort), and fail-over to a longer route (Tlong). We only focus on the first two cases and leave it to future work to study the latter two cases. To study Tshort and Tlong, we need to modify the Beacon setup to inject a withdrawal to only one of the upstream ASes rather than to both ASes in the case that the Beacon is multihomed. As pointed out by Labovitz, the observed behavior of Tshort is very similar to Tup and that of Tlong is quite similar to Tdown. In fact, we just recently completed setting up a fifth Beacon with such capability. We leave the analysis to future work. Please refer to the PSG Beacon web page for details of the schedule for the new Beacon.

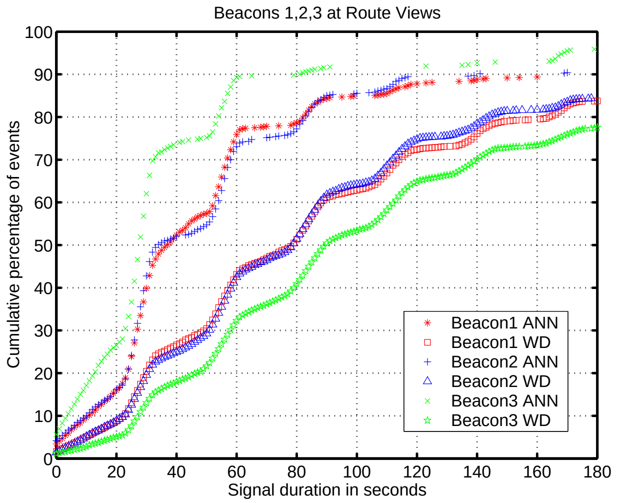


Figure 17: Cumulative distribution of signal duration for all three Beacons for both announcement and withdrawal signals

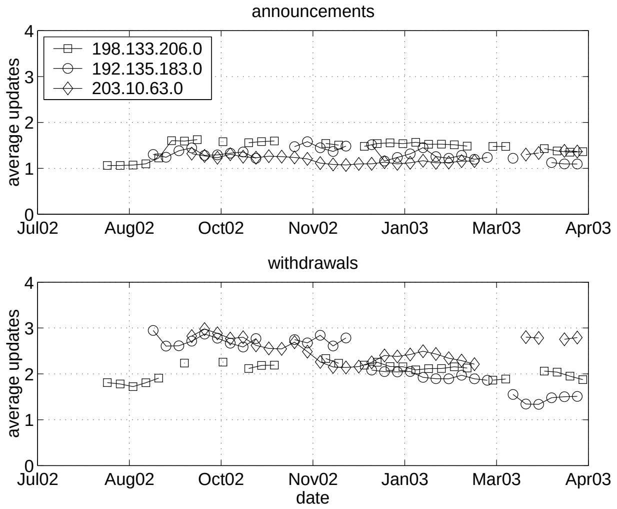


Figure 18: Variation in average signal length over time (Beacons 1,2,3).

Figures 16 and 17 present cumulative distributions of the relative convergence times and signal durations, for PSG Beacons 1, 2 and 3. These results are entirely consistent with the results of [9], showing that these characteristics appear not to have changed significantly in the last few years. Our analysis is consistent also with a preliminary study of the RIPE Beacons, recently presented [19].

Figure 18 presents the variations of average signal length over time. These averages mask the wide variations seen among peers, which can be observed in Figure 19. As in [9], we see more updates in signals associated with withdrawals than announcements.

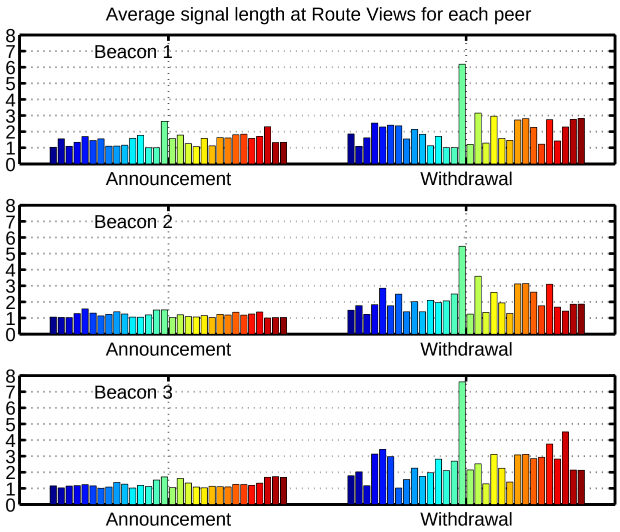


Figure 19: Average signal length for each peer

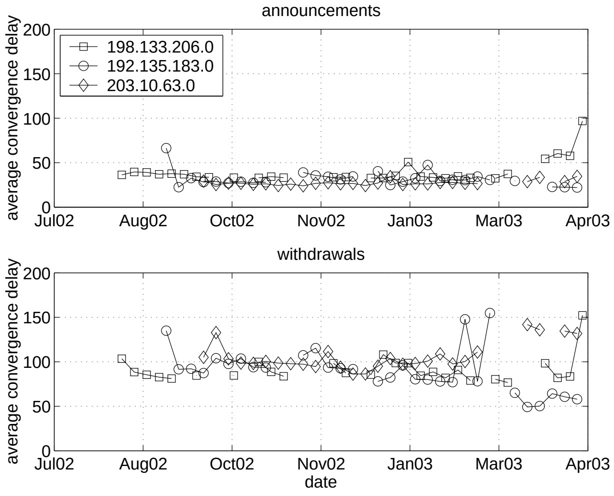


Figure 20: Variation in average relative convergence delay over time (Beacons 1,2,3).

Figure 20 shows the variation in the relative convergence delay over time — there is a large amount of variation but no clear trend.

Figure 21 illustrates an interesting time series of Beacon 1's signal duration during the course of our study to demonstrate the effect of upstream connectivity on the amount of BGP noise. The gaps indicate the time periods during which the Beacon was down due to network problems. In the month of August 2002, the Beacon was single-homed: with only one upstream provider AS2914. In September 2002, it became multi-homed to AS2914 and AS1. There is little change for withdrawal signal duration; however, the duration for announcement signals doubled from around 30 seconds to 60 seconds. Upon further analysis, we found that the average announcement signal length also doubled from around 1 to 2. After Beacon 1 becomes multihomed, the announcement signal is very likely to explore the alternate less preferred route first before settling on the final route. There is another interesting change occurred in April 2003, during which one of Beacon 1's upstream providers is changed from AS1 to AS1239. Apparently AS1239 seems to be better connected than AS1, resulting in shorter announcement signal durations.

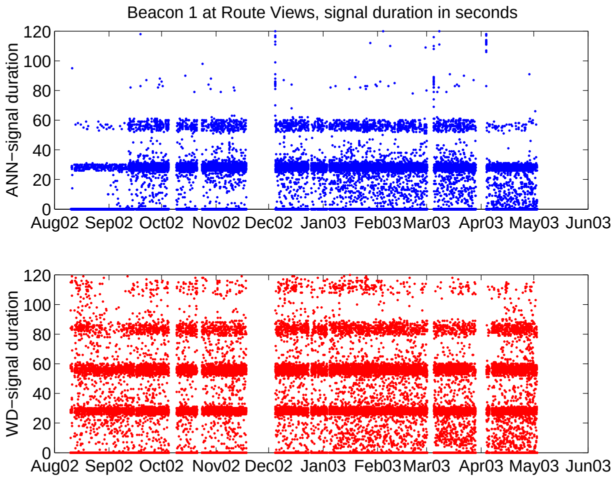


Figure 21: Beacon 1's signal duration variation over time, cutoff at 120 seconds. Max duration is 3525 seconds.

8. CONCLUSION

The paper describes a set of BGP Beacons that have been set up for public use, along with techniques for obtaining clean and useful data from these Beacons. Used in conjunction with public route monitors they provide a mechanism for performing controlled experiments with the global Internet routing system. We present several examples of how data from such experiments may be used to understand BGP routing dynamics.

Many interesting questions remain to be investigated, for instance, can we use this data to form a realistic and robust model of the dynamics of BGP? Such a model should, at the least, allow one to engineer BGP settings such as the MinRouteAdvertTimer, or route flap damping parameters to optimize global BGP convergence and stability.

To achieve such a model, we must gain a better understanding of some of the results discussed here – for instance, we need to better comprehend the processes that lead to the tail of the inter-arrival time. The results above suggest that the large inter-arrival times might be the best place to start improving BGP performance, but the data so far are not conclusive as to the cause of these gaps, though route flap damping is the likely culprit.

We have limited our attention to the PSG Beacons and Route Views monitoring data. We expect that using the RIPE Beacons will provide invaluable additional insights for this investigation, particularly because of the different connectivity and geographic locations of the RIPE Beacons. Furthermore, the on-going nature of the Beacons create a historical view of Internet routing dynamics that will allow measurement of the impact of changes to BGP.

Acknowledgments

We would like to thank all the PSG Beacon hosts for their cooperation. Without their patience and encouragement, this work would have been possible. We also would like RIPE Information Service for providing Beacons as well as monitors. Finally, we thank the people at Route Views for their continuous support.

9. REFERENCES

[1] Y. Rekhter and T. Li, “A Border Gateway Protocol,” RFC 1771 (BGP version 4), March 1995.

[2] S. Halabi and D. McPherson, *Internet Routing Architectures*, Cisco Press, Indianapolis, Indiana, second edition, 2000.

[3] C. Huitema, *Routing in the Internet*, Prentice Hall, 2000.

[4] C. Labovitz, R. Malan, and F. Jahanian, “Internet Routing Stability,” in *Proceedings of ACM SIGCOMM 1997*.

[5] C. Labovitz, R. Malan, and F. Jahanian, “Origins of Internet Routing Instability,” in *Proceedings of INFOCOM 1999*.

[6] C. Labovitz, A. Ahuja, and F. Jahanian, “Experimental Study of Internet Stability and Wide-Area Network Failures,” in *Proceedings of FTCS 1999*.

[7] “University of Oregon Route Views Archive Project,” `www.routeviews.org`.

[8] Ripe NCC, “Routing Information Service Raw Data,” .

[9] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, “Delayed Internet Routing Convergence,” in *Proceedings of ACM SIGCOMM 2000*.

[10] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja, “The Impact of Internet Policy and Topology on Delayed Routing Convergence,” in *Proceedings of INFOCOM 2001*.

[11] C. Villamizar, R. Chandra, and R. Govindan, “BGP Route Flap Damping,” RFC 2439, 1998.

[12] Zhuoqing Morley Mao, Ramesh Govindan, George Varghese, and Randy H. Katz, “Route flap damping exacerbates internet routing convergence,” in *Proceedings of ACM SIGCOMM 2002*.

[13] “PSG BGP Beacons,” `http://www.psg.com/~zmao/BGPBeacon.html`.

[14] “RIPE BGP Beacons,” `http://www.ripe.net/ris/beacon.html`.

[15] Tim Griffin, “What is the sound of one route flapping?,” Network Modeling and Simulation Summer Workshop at Dartmouth, July 2002.

[16] Lan Wang, Xiaoliang Zhao, Dan Pei, Randy Bush, Daniel Massey, Allison Mankin, S. Felix Wu, and Lixia Zhang, “Observation and Analysis of BGP Behavior under Stress,” in *Proceedings of Internet Measurement Workshop 2002*.

[17] Timothy G. Griffin and Brian J. Premore, “An Experimental Analysis of BGP Convergence Time,” in *Proceedings of ICNP 2001*.

[18] Pedro Roque Marques, “BGP route advertisement interval,” Talk at RIPE 45 Meetings, available at `http://www.ripe.net/ripe/meetings/ripe-45/presentations/`.

[19] Henk Uijterwaal, “Routing Beacons,” Available at `http://www.potaroo.net/iepg/november2002/`, November 2002.