BGP churn evolution: A perspective from the core

Ahmed Elmokashfi, *Member, IEEE,* Amund Kvalbein, *Member, IEEE,* and Constantine Dovrolis, *Member, IEEE*

***Abstract*—The scalability limitations of BGP have been a major concern lately. An important aspect of this issue is the rate of routing updates (churn) that BGP routers must process. This paper presents an analysis of the evolution of churn in four networks at the backbone of the Internet over a period of seven years and eight months, using BGP update traces from the RouteViews project. The churn rate varies widely over time and between networks. Instead of descriptive "black-box" statistical analysis, we take an exploratory data analysis approach attempting to understand the reasons behind major observed characteristics of the churn time series. We find that duplicate announcements is a major churn contributor, responsible for most large spikes. Remaining spikes are mostly caused by routing incidents that affect a large number of prefixes simultaneously. More long-term intense periods of churn, on the other hand, are caused by misconfigurations or other special events at or close to the monitored AS. After filtering pathologies and effects that are not related to the long-term evolution of churn, we analyze the remaining "baseline" churn and find that it is increasing at a rate that is similar to the growth of the number of Autonomous Systems.**

I. Introduction

The deployment of the BGP routing protocol has sustained tremendous growth over the last couple of decades and it is arguably one of the main technological reasons behind the Internet's success. Lately, however, there are significant concerns about the *scalability of BGP interdomain routing*. These concerns focus either on the growing routing table size (number of routable prefixes) or on BGP dynamics and instability (also known as "churn") [17]. Both factors are important, especially for routers at the core of the Internet. The growing size of the routing table requires increasingly larger fast memory, but it does not necessarily slow down packet forwarding as long as address lookups are performed using TCAMs or constant-time longest-prefix matching algorithms [26].

Churn, on the other hand, is a more serious concern because processing BGP updates can be computationally intensive (updating routing state, generating more updates, checking import/export filters), and it can trigger a wide-scale instability. If the current best route to a destination is modified, the global RIB and the line card FIBs need to be updated.

To make things worse, routing updates are known to be very bursty, with peak rates several orders of magnitude higher than daily averages. When the rate of updates becomes too high, the fear is that there will be (or there are already) periods when routers will be unable to maintain a consistent routing table.

A. Elmokashfi and A. Kvalbein are with Simula Research Laboratory, 1364 Fornebu, Norway e-mail: ({ahmed,amundk}@simula.no).

C. Dovrolis is with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30332 USA, e-mail: (dovrolis@cc.gatech.edu).

An earlier study by Huston and Armitage reported an alarming growth in churn [10]. During 2005, the daily rate of BGP updates observed by a router in AS1221 (Telstra) almost doubled, while the number of prefixes grew by only 18%. Based on these measurements, the authors projected future churn levels and concluded that current router hardware will need significant upgrades in order to cope with churn in a 3-5 years horizon. It was this study that largely motivated our work.

Specifically, in this paper we present a longitudinal study of BGP churn spanning a longer time frame (more than seven years) and more monitors (routers at four tier-1 ISPs) than previous studies. Generally, the churn time series is very noisy, dominated by frequent large spikes and "level shifts" that last for several weeks or even months. There are periods in which churn is slowly increasing, others in which it is decreasing, and major differences between monitors. One option could be to characterize the evolution of churn using "black-box" statistical or time series analysis methods. That approach would answer questions about the correlation structure and the marginal distribution of the underlying time series, attempting to fit the data in a standard time series model. That descriptive method, however, would not be able to *explain* what causes spikes, level shifts or trends in BGP churn. We prefer, instead, to take Tukey's exploratory data analysis approach that focuses on the *causes behind the observed phenomena* and on the use of data to formulate new hypotheses instead of only testing existing hypotheses.

In more detail, we first analyze what causes some major characteristics of the "raw" BGP update time series (spikes, level shifts, etc). As a second step, after we remove pathologies or effects that are not related to the long-term evolution of churn, we apply statistical trend estimation on the remaining "baseline" churn and compare the observed growth with the growth of other routing table aspects, such as the number of routable prefixes, the number of Autonomous Systems (AS), and the number of AS routing paths. We find that duplicate announcements is a major churn contributor, responsible for most large spikes in the churn time series. Duplicate announcements are redundant and can be viewed as artifacts of some BGP implementations. Most remaining spikes in the churn time series are caused by routing incidents that affect a large number of prefixes (large events) simultaneously. These incidents show little correlation between different monitors and thus affect only a limited part of the Internet. Other intense periods of churn, which we will call level shifts, are caused by misconfigurations or other special events at or close to the monitored AS. Our observations explain why different networks and monitors experience very different churn.

After removing updates attributed to duplicates, large events, and level shifts, the remaining time series is much

smoother and shows more consistent growth across monitors. This filtered version of churn has increased during the seven years of our study period by about 100%, which is significantly less than the growth rate of routable prefixes during the same time period (about 168%). This implies that the number of updates per prefix per day is decreasing over time, indicating that, on the average, the stability of routing prefixes has improved. Interestingly, we observe that the baseline churn grows at a rate that is very similar to the growth in the number of ASes in the Internet. The average number of updates contributed by each AS per day has been almost constant at around five updates.

We examine the daily peak churn rate, measured as the busiest one-minute period in that day, and find that it would be reduced by more than an order of magnitude if updates that are redundant or caused by certain anomalies were filtered out. We also observe that the use of BGP update rate-limiting timers can play an important role in reducing certain pathological sources of churn, but it is not able to protect a router from large spikes and level shifts.

The rest of this paper is organized as follows. Sec. II gives an overview of different underlying factors that create BGP churn. Sec. III describes our dataset. Sec. IV shows the major trends in BGP routing during our study period. Sec. V focuses on the role of the rate-limiting timer and of duplicate updates on the churn time series. Sec. VI looks at the role of large routing events that affect many prefixes. Sec. VII investigates major level shifts in the churn time series. Sec. VIII analyzes the growth of the baseline churn that remains after we remove pathologies, large spikes and level shifts. Sec. IX gives an overview of related work, and Sec. X concludes this work.

II. Churn growth factors

Several different factors can influence BGP churn. First, it is expected that the rate of BGP updates a router receives will increase with the number of routable destination prefixes. Roughly speaking, each prefix corresponds to a destination network. If these destination networks fail and recover independently and with the same probability, we would expect a linear relation between the size of the routing table and churn.

The observed churn will also depend on the routing activity of individual prefixes at their origin AS. Over the past few years, it has become increasingly common for stub ASes to be multihomed to several providers [4]. Multihoming enables load-balancing by selectively announcing different prefixes to different providers. As this practice gradually becomes more common, we expect that it contributes to increasing churn when a network destination becomes unreachable.

Another source of churn is routing events taking place in or between transit ASes. Such events include link failures (physical failures, router reboots, etc), policy changes that result in new preferred routes, or changes in the IGP or iBGP configuration of a transit AS. Importantly, these operations often affect a large number of prefixes at the same time. The amount of churn observed at a router after such events will also depend on the topology and routing policies.

Topological properties of the AS-level Internet graph also affect the churn rate [6]. Increased multihoming increases the churn generated when a destination prefix is announced or withdrawn from the origin AS. On the other hand, increased connectivity can reduce the impact of failures, if a local alternative is available. Topological properties also influence the number of updates generated during the path exploration that takes place when BGP explores several paths after a routing incident before converging to a stable state [19].

Additionally, there are BGP mechanisms and parameter settings that can reduce the observed churn. Two important mechanisms are the MinRouteAdvertiseInterval (MRAI) timer and Route Flap Damping (RFD). Furthermore, the use of route reflectors in iBGP can limit or increase churn [24]. The interactions between different protocol implementations and configurations, or their impact on BGP churn, is far from well understood.

III. Dataset

Our analysis is based on BGP update traces collected by the RouteViews project [1]. RouteViews collectors run BGP sessions with several routers, referred to as *monitors*, in many networks. A monitor sends a BGP update to the collector every time there is a change in the preferred path from the monitor to a destination prefix. In addition, RouteViews dumps every two hours a snapshot of the routing table that contains the best selected paths advertised from each monitor. We use those snapshots to observe the growth of the routing table size over the last few years.

We focus on update traces from monitors at large transit networks in the core of the Internet. Specifically, we analyze the churn time series from four monitors at AT&T, Sprint, Level-3 and France Telecom (FT). The corresponding monitors belong to the *Default Free Zone* (DFZ), meaning that they do not have a default route to another provider, and so they know a route to practically all destination networks in the Internet.

RouteViews provides historical update traces spanning more than seven years for these four monitors. In some cases, the IP address of the monitor changed during our study period. We identified the corresponding IP addresses and concatenated the update time series after confirming that they correspond to the same actual monitor. Our time series cover the period from January-01-2003 to August-31-2010, giving us more than 7.5 years worth of routing updates from four backbone monitors. However, the Sprint monitor was unavailable during the last two years of our study period, while the FT monitor was unavailable after February 2009. The AT&T monitor was unavailable during 2.5 months in late 2003.

If the multi-hop BGP session between a monitor and the collector is broken and re-established, the monitor will re-announce all its known paths, giving large bursts of updates. This is a local artifact of the RouteViews measurement infrastructure, and it does not represent genuine routing dynamics. Hence, we use the method described in [29] to identify and remove updates caused by "session resets". After filtering, our dataset consists of more than 1.8 billion updates. Note that the updates received from a monitor is not a good estimate for the total number of updates a backbone router must process. A router typically has several active BGP sessions, and so the

	Minimum	Median	Maximum
Kendall's τ	0.70	0.80	0.93
Spearman's ρ	0.84	0.92	0.98

TABLE I: Measured cross-correlation across monitors in the same AS (based on ten monitor pairs).

total load on the router is the sum of the churn from all BGP sessions.

To confirm that the method of [29] is able to identify all session resets and the subsequent routing table transfers, we applied it on BGP updates collected from three multi-hop BGP sessions. These sessions are established locally with a well-connected stub AS (AS44654). We compared table transfers detected by the previous method in a period of three months against BGP session logs from the corresponding router. The method was able to detect table transfers and their exact starting time in 11 out of 12 cases (the 12th case was also reported but with a 7-sec difference in the starting time).

Due to complex iBGP configurations using confederations or route reflectors, different edge routers of the same AS do not necessarily see the same set of paths to different destinations. In order to understand the impact of such differences on churn, we examined a set of ten pairs of monitors such that each pair consists of routers in the same AS that peer with the RouteViews Oregon-IX collector. Then, we measured the hourly churn time series in four different months (Aug'03, Mar'04, May'05, and Feb'07) for the ten monitor pairs. We calculated the cross-correlation of hourly BGP updates between the time series of each pair, using two non-parametric measures: Kendall's τ and Spearman's ρ [8]. Both measures indicated a reasonably high cross-correlation between monitors of the same AS. Table I shows the minimum, median, and maximum cross-correlation across all ten pairs (a value of 1.0 denotes maximum correlation). Even though we cannot claim that these observations are true in general, *it is reasonable to expect that two routers of the same AS produce similar (but not identical) churn*.

IV. The "raw" churn time series

We start with three important aspects of growth in the BGP routing system. The left panel in Fig. 1 shows the number of routing table entries in the four monitors, sampled on a monthly basis. The number of entries in the different monitors is very similar, which is expected since these monitors are all DFZ routers. The *number of routable prefixes increased by 168% during our study period*, from about 120K to 322K entries (the increase can be modeled as quadratic, with a coefficient of determination of 99.9%). The middle panel shows the number of ASes, sampled on a monthly basis. *This metric has increased by 143% during our measurement period*. The right panel shows the number of distinct AS paths (routing paths) in the observed BGP routing tables (after removing the effects of AS path-prepending) again on a monthly basis. *This metric has increased by 163% during our measurement period*.

One may expect that since the size of the routing table, the number of ASes, and the number of routing paths have more than doubled during the study period, BGP churn should also show a similar consistent and significant increase. This is not the case however. The left column in Fig. 2 shows the "raw" BGP churn time series, measured as the number of BGP updates received daily from each monitor. Some high-level observations are necessary before we proceed with the analysis.

The raw time series is dominated by frequent and large spikes. At all monitors, there are days with dramatically higher churn than usual. We have truncated the y-axis of these plots to make the graphs more readable (on some days the number of updates reached several millions). Large spikes are particularly frequent at the Level-3 monitor. Such spikes cannot be ignored as "statistical outliers"; instead, we need to understand what causes them.

There are several "level shifts". In addition to spikes, we observe several periods of sustained increased activity that last for weeks or months. For example, we see a period that lasted about 6 months in mid-2006 at the Level-3 monitor. Again, level shifts cannot be viewed just as incidents of statistical non-stationarity; we need to understand what causes them.

There is little correlation between monitors of different ASes. The spikes and level shifts at the four monitors do not follow the same pattern. We measured the cross-correlation between different monitors using Kendall's τ coefficient. The estimated cross-correlation coefficient is between 0.17 and 0.3, which illustrates a small correlation between the four monitors. *This indicates that churn is highly dependent on the location and configuration of the corresponding router*. So, we cannot understand the evolution of BGP churn by just looking at a single monitor.

Churn is highly bursty even at large time scales. As seen at the left column of Fig. 2, churn is highly bursty even in the relatively large time scale of a day. We also examined the churn time series in shorter time scales (5 minutes and one hour) and observed that in some cases the majority of the daily churn is produced during short periods that last for few minutes.

It can be misleading to infer long-term trends from the raw churn time series. Because of the previous issues, it is clear that the blind application of statistical trend estimation methods can fail to detect a trend or it can produce misleading results. The approach we take in this paper is to first analyze what causes some major characteristics of the raw time series (spikes, level shifts, etc) and then, after we remove pathological and effects that are not related to the long-term evolution of churn, to apply statistical trend estimation on the remaining "baseline" churn.

V. Impact of rate-limiting timers and of duplicate updates

We first analyze the deployment and impact of the rate-limiting timer during our study period. We are interested to assess the impact of rate-limiting on the observed churn, and to understand any sudden changes in churn level when the timer is toggled. Second, we examine the frequency of duplicate BGP updates in the churn time series.

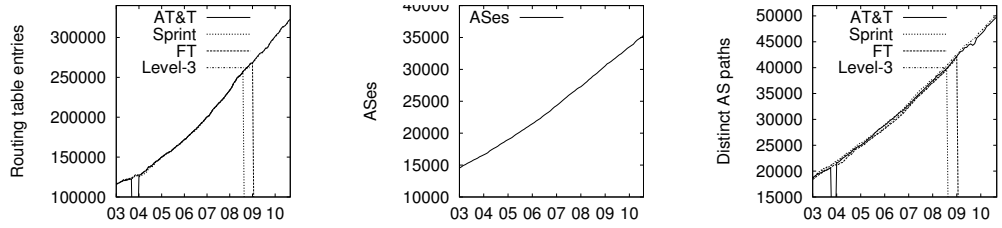

Fig. 1: Routing table size (left), number of ASes (middle), and number of observed AS paths (right) during our study period.

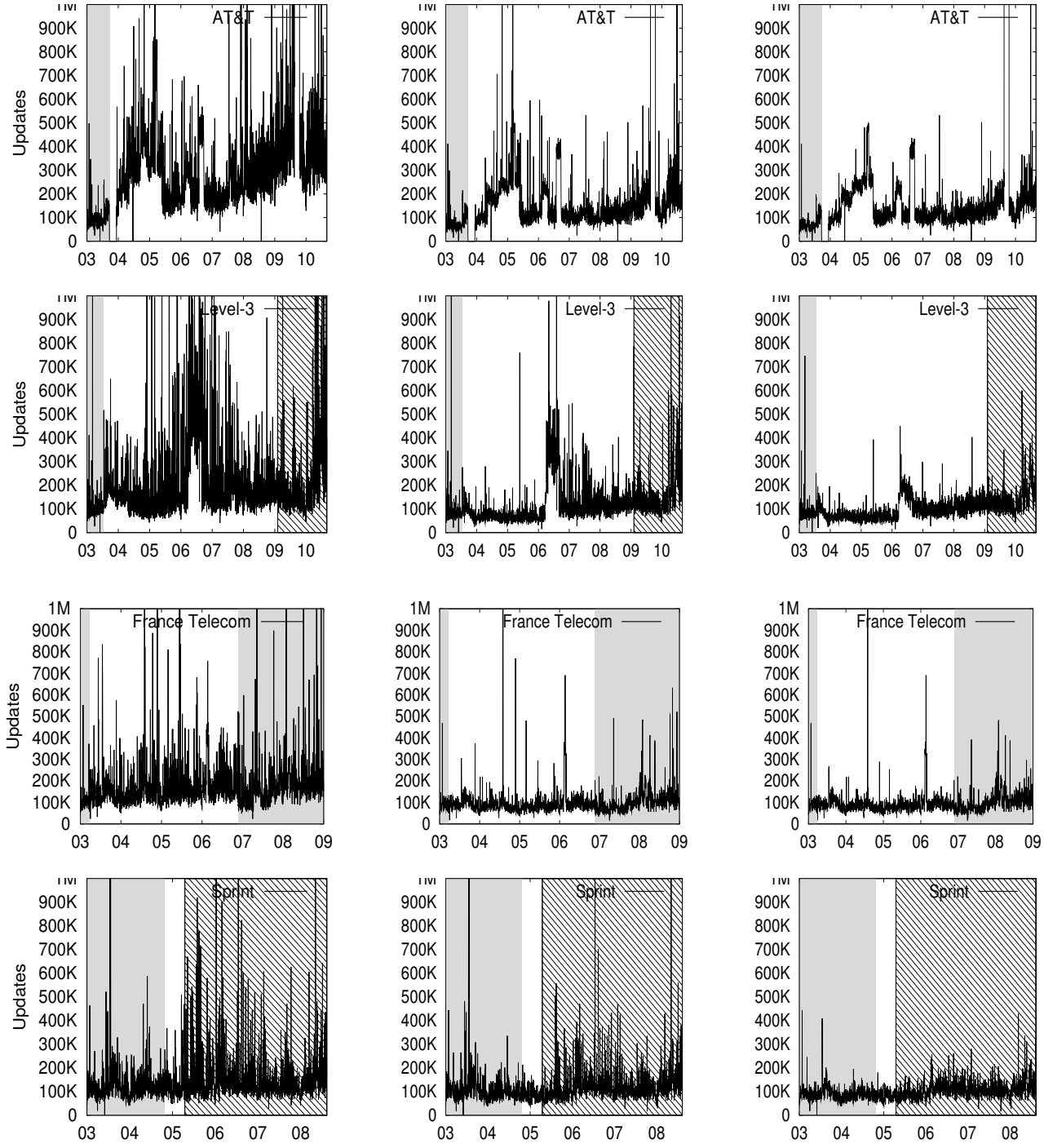

Fig. 2: Daily BGP churn: raw time series (left), after removing duplicates (middle), and after removing duplicates and large events (right) at AT&T , Level-3 , FT, Sprint from top to bottom respectively.

A. Rate-limiting timer deployment and impact

To limit the rate of BGP updates, the BGP standard [21] recommends the use of a MinRouteAdvertisementIntervalTimer (MRAI timer) which specifies the minimum time interval between sending two consecutive updates for a destination prefix. The recommended value of this timer on eBGP sessions is 30 seconds. The standard recommends jittering the MRAI timer by multiplying its value with a random number between 0.75 and 1.

Some implementations (notably Cisco IOS and the Quagga software router), implement per-session timers rather than per-prefix timers in order to reduce overhead. Another common BGP implementation (Juniper's JunOS) implements rate-limiting using the *out-delay* parameter. Unlike the MRAI timer implementation described above, this delay is added to each update for each prefix individually. When a router changes its best path to a destination prefix, it will not inform its peer about the change unless the route has been present in its routing table for the specified *out-delay*.

To determine whether a monitoring session is rate-limited, we look at the time series of updates for that monitoring session. If a monitor uses MRAI, we expect to see a pattern where updates arrive in bursts every time the timer expires. In other words, we should see very few inter-arrival periods in the range [0-22] seconds, assuming a jittered default MRAI timer value. On the other hand, if a monitor uses *out-delay* to perform rate-limiting, we do not expect the same bursty pattern of updates. Instead, we should see a pattern where updates arrive in a steady flow, but where two updates for the same prefix are always spaced by at least the *out-delay* timer value (i.e. 30 seconds). So, to detect whether a rate-limiting timer was deployed at the four monitors during the study period, we used the following two-step approach.

1. For each monitor we select one day from each week of the study period, which resulted in a sample of at least 288 days per monitor. We then calculated the distribution of update inter-arrival times for each day in the sample.

2. We calculate the fraction of update inter-arrival times across all prefixes that is less than 22.5 seconds. If that fraction is significant, MRAI was probably *not* deployed on the corresponding day. Furthermore, we calculate the fraction of update inter-arrival times for individual prefixes that is less than 30 seconds. If that fraction is significant, *out-delay* was probably *not* deployed on the corresponding day. We find that setting the threshold for the fraction of both inter-arrivals anywhere between 0.15 and 0.2 (i.e. dismissing 15% to 20% of the outliers) results in detecting the same periods.

Figure. 3 shows the fraction of update inter-arrival times for individual prefixes and across all prefixes that is less than 30 and 22.5 seconds respectively, for the AT&T and Sprint monitors. Note that in AT&T the fraction of update inter-arrival times across all prefixes started at about 5% during the first nine months of 2003, and then it increased steeply to about 99% in the rest of the study period. The fraction of update inter-arrival times for individual prefixes remained relatively large during the study period. In Sprint, the fraction of update inter-arrival times across all prefixes was about 10%

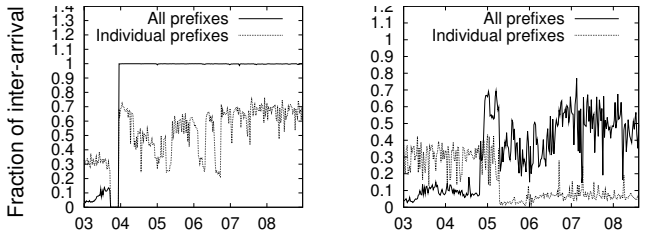

Fig. 3: Identification of rate-limited periods (left: AT&T, right: Sprint).

in the period between Jan'03 to Oct'04, and after Oct'04 it increased to about 40%. The fraction of update inter-arrival times for individual prefixes was about 30% between Jan'03 and Apr'05, then it decreased steeply to less than 10% in the rest of our study period.

The above observations suggest that the MRAI timer at the AT&T monitor was active initially, but it was then turned off. In addition, the observations indicate that the MRAI timer at the Sprint monitor was active between Jan'03 to Oct'04, it was then turned off for six months, and then *out-delay* was used from Apr'05 until the end of our study period. Switching between MRAI and *out-delay* suggests that the router hardware of the Sprint monitor was replaced. Using a similar analysis we inferred the rate-limiting deployment periods for the two other monitors.[1] In Figs. 2 and 8, we use *grey shaded areas in the time series to indicate periods in which the MRAI timer was deployed, and a diagonally shaded areas to indicate the same for the out-delay timer*.

To investigate the impact of the rate-limiting timer (MRAI or *out-delay*), we measured the median daily churn rate in a three month period before and after each identified deployment transition. For each transition, we calculated the ratio (median churn without rate-limiting) over (median churn with rate-limiting). *We found that the churn level increases when the rate-limiting timer is turned off, while it decreases when it is turned on*. For example in Level-3, churn increased by a factor of 1.8 when the timer was turned off. A recent study has illustrated that *out-delay* limits more churn than MRAI [5].

B. Duplicate updates

The conventional wisdom is that BGP implementations generate a large number of duplicate updates, which imposes an unnecessary processing load on routers. It has been pointed out that one reason for the large number of redundant updates is stateless BGP implementations that do not keep track of the last update sent to a peer [12].

We identified all duplicate updates (announcements and withdrawals) in our dataset. By "duplicate announcement" we mean an announcement that is identical to the last seen announcement for the same prefix, i.e., no change in either the AS-path or in any of the transitive route attributes. These announcements are redundant and can be viewed as a pathology of the BGP implementation at the corresponding monitor.

[1] To confirm the robustness of the aforementioned rate-limiting inference we have investigated the deployment of rate-limiting with smaller timer values (i.e. between 5 and 30 seconds). We did not detect rate-limiting configurations that use non-standard timer values.

A recent measurement study [20] attributed BGP duplicate updates to interactions between iBGP and eBGP.

To our great surprise, we measured that, across all four monitors, duplicate announcements are responsible for about 40% of the churn during the study period! On the other hand, duplicate withdrawals are close to zero (except Level-3, where they account for about 1% of the updates). It is interesting that almost half of the observed churn is not really necessary. This number is higher than the 16% of the duplicate announcements "AADupType1" reported earlier in [14]; that study looked at monitors located in ASes of different sizes during a 6-month period in 2006. Our estimate is also higher than what is reported in [20].

The number of duplicate updates per day is highly variable, and shows no correlation across monitors. It is also difficult to identify any consistent long-term trend in the number of duplicates. These results indicate that the specific implementation of BGP and local configuration details can greatly influence the amount of redundant updates.

There is still much to be gained by deploying improved BGP implementations that avoid sending redundant updates to the global routing system. Such improvements would require, however, per-neighbor state at BGP routers to keep track of what was sent to each peer earlier, so that duplicate updates can be detected before they are transmitted. There is a trade-off between allowing the generation of duplicate updates (that will be filtered at the receiving router) versus more heavy weight processing at the sending router that would also eliminate duplicate updates [23]. Arguably, these changes may not be worth doing, given the lightweight handling of duplicates.

The second column in Fig. 2 shows the four time series after filtering out duplicate updates[2]. Note that removing duplicate updates has the additional benefit that most of the spikes are also removed. *This indicates that redundant updates are not only responsible for a large fraction of churn, but they are also responsible for generating large bursts of churn*. There is no measurement work that quantifies the actual processing burden on routers caused by duplicates, hence the impact of these bursts on routers' CPUs is not clear at this point.

VI. Large Events

After removing duplicates, we focus on "large routing events", or simply *large events*, loosely defined as events that affect a large number of prefixes at about the same time. The intuition is that incidents in the core of the Internet have the potential to introduce instability to a large number of prefixes simultaneously, causing major churn spikes. Such incidents may be link failures in or between transit ASes, or internal routing or policy changes in an AS. For instance, a large number of prefixes may change their BGP next hop following an adjustment of IGP link weights due to hot-potato routing [25]. Large events may also be triggered by changes in geographical community attributes that determine the preferred exit POP when two ASes peer at more than one location. Large events can potentially impose a high burden on a router's CPU, because they affect a large number of prefixes simultaneously. It is important to characterize large events in order to understand the extent and evolution of their impact.

When an underlying incident triggers a routing change, it often results in several updates for each affected prefix. We define a *prefix event* as a sequence of updates for a given prefix that are likely generated by the same underlying incident. The updates of a prefix event typically have short inter-arrival times. Here, we adopt the definition given in [28] for identifying prefix events:

Definition 1: Two consecutive updates for the same prefix belong to the same *prefix event* if they are no more than 70 seconds apart. The maximum duration for a prefix event is set to 10 minutes. Events with duration longer than 10 minutes are considered to be flapping.

The previous thresholds were determined based on measuring the convergence times for beacon prefixes [16]. The authors in [28] showed that over 98% of the updates received after a beacon prefix announcement/withdrawal had shorter inter-arrival times than 70 seconds.

Some routing incidents affect several prefixes. We group prefix events that occur at about the same time into *events*.

Definition 2: Starting with a prefix event p, the *event* that follows p consists of all prefix events that start no later than t seconds after the start of p.

The intuition is that when a routing incident affects multiple prefixes, the first updates for these prefixes should arrive in a burst. The "event grouping threshold" t should therefore be set to a low value, to minimize the risk of erroneously grouping prefix events that are caused by different underlying incidents into the same event. To find a suitable value for t, we investigated how the event size (i.e. the number of prefix events included in an event) varies for different values of t. We looked at different time periods in all our four monitors, and found that the 99-th percentile of event sizes shows small variations when t is between 1 and 20 seconds. We use $t = 5$ seconds in the rest of this paper.

Next, we define a *large event* as an event that affects many prefixes. To choose an appropriate threshold for classifying an event as a large event, we identified all events that took place during the month of January in each year of our study period, for all four monitors. Fig. 4 shows the distribution of the number of affected prefixes per event - each curve represents the events during the period of one month and for one monitor. Our objective here is not to analyze the differences between monitors or months, but to observe the "typical" distribution of event sizes. We only show the tail of the distribution - the full CDF shows that half of all events affect less than 10 prefixes, while more than 90% of events affect less than 40 prefixes. Based on this graph, we use a threshold of 2000 prefixes. Note that all CDFs flatten out after this threshold. With this definition, at most 0.2% of all events are considered to be large events.

Definition 3: A *large event* is an event that includes at least 2000 prefix events.

The number of large events over our study period varies significantly across monitors (from 1554 for France-Telecom to 15054 for AT&T). Note that 12265 of the large events

[2]Raw and filtered datasets are available at http://vefur.simula.no/bgp-churn/.

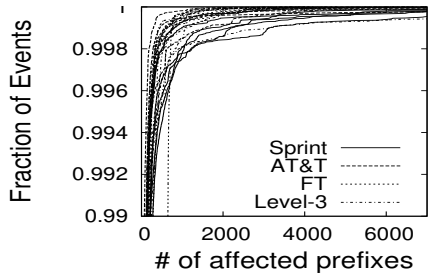

Fig. 4: Distribution of the number of affected prefixes per event.

at the AT&T monitor were observed during a period of two days between June-29-2010 and July-01-2010. These events resulted in over 156 million updates. A closer look indicated that this high activity is caused by continuous flapping of about 217K prefixes that were originated from 25000 different ASes. The flapping involved 1937 different next hops. The involvement of a large number of prefixes and the high diversity in terms of next hops and origins show that the cause of those events was probably local to the monitor AS. In the rest of this section, we focus on the remaining 2789 large events in the AT&T time series.

Next, we characterize large events with respect to their type, and describe the evolution of their size, frequency, and duration. We also investigate the correlation of large events across monitors.

A. Types of large events

Given that the number of large events is significant, it is difficult to examine them individually in order to pin-point their root causes. Instead, we categorize large events into different *Path Transition Signatures* (PTS) based on the most dominant routing change (i.e. path instabilities, path changes, or path withdrawals) in the underlying single prefix events that constitute them. We classify a single-prefix event into different types, depending on the *best known path before the event* and the *best known path after the event* [13], [19]:

- WA: Starts with no known path to the affected prefix, and ends up with a path.
- AW: Starts with a path and ends with no path.
- AAC: Starts with a path P and ends with a different path P'.
- AAD: Starts with a path P and ends with the same path P, but at least one different path P' was seen during the convergence process.
- AAS: Starts with a path P and ends with the same path P, and no other path was seen during the convergence process.

If a large event was caused by a certain routing incident, the majority of involved prefixes are likely to show the same PTS. To verify this, we group the single-prefix events that constitute a large event based on their PTS, and define the dominant PTS as the PTS of the largest of these groups. Across all monitors, we observe that the dominant PTS normally covers most of the underlying prefix events: in 99% of large events, more than 50% of the prefix events have the same PTS.

Monitor	AT&T	Level-3	FT	Sprint
AW	21.1%	1.8%	4.4%	1.6%
WA	21.7%	1.6%	4.1%	1.6%
AAC	46.1%	23.9%	66.9%	6.4%
AAD	8.9	32.8%	15.6%	1.5%
AAS	0.7%	36.6%	0.3%	70.1%
ND	1.2%	3.1%	8.8%	18.7%

TABLE II: Classification of large events at the four monitors.

M \ **M'**	AT&T	FT	Level-3	Sprint	Uncond-Prob
AT&T	-	0.05	0.11	0.11	0.003
FT	0.05	-	0.09	0.09	0.003
Level-3	0.02	0.02	-	0.03	0.018
Sprint	0.03	0.03	0.05	-	0.010

TABLE III: Conditional probability that a large event is seen at monitor M given a large event at a monitor M' in the same 10-min interval. The corresponding unconditional probability is shown at the rightmost column.

To simplify our analysis, we proceed to classify large events based on the dominant PTS of the corresponding prefix events. We say that a large event is of type A if at least 70% of the involved single prefix events are of type A. As seen in Tab. II, most large events can be assigned to one of the event classes using this definition. *We observe that the dominant classes differ from one monitor to another. AAC, AW, and WA are dominant at AT&T. At Sprint, the majority of large events are of AAS type. AAD, AAC, and AAS dominate at Level-3, while AAC and AAD dominate at FT.*

B. Temporal correlations of Large Events

In this subsection we investigate whether large events tend to happen at the same time at different monitors, and whether prefixes that are affected by a large event at monitor M also tend to be active at other monitors at the same time.

Large events across monitors. To calculate how large events are related across monitors, we divide each time series into bins of 10 minutes. Then, we construct a new binary time series such that a bin will be assigned a value of one if we record at least one large event during the time covered by that bin, and zero otherwise. Further, we estimate the probability that a large event is seen at monitor M given that a large event is seen at monitor M' in the same 10-min bin (i.e. the conditional probability $P(M|M')$). We calculate this probability for all pairs of monitors at lags 1, 0, and -1. Table III shows the conditional probability $P(M|M')$ and the unconditional probability $P(M)$ at lag 0 (the probabilities at lag 1 and lag -1 are smaller than at lag 0). Each row corresponds to a monitor M, while each column corresponds to a monitor M'. We observe that the conditional probabilities are markedly higher than the corresponding unconditional probabilities in most monitors, showing that at least some large events affect multiple monitors. Note, however, that the absolute probabilities are quite low. In other words, *it is not very likely that we observe a large event at monitor M' even if there is a large event at monitor M. This indicates that most large events affect only a limited part of the Internet.*

The propagation of large events. We can now investigate whether a large event at monitor M is visible (perhaps not as large event) at monitor M'. To do so, we start by identifying

prefixes that are affected by a large event at monitor M. For each such prefix we check if it was active within a time window of width W in the update traces of monitor M'. Then, we calculate the fraction of prefixes in a large event that shows such temporal correlations. This analysis is performed for all observed large events and between all pairs of monitors. We experimented with several correlation window sizes. Increasing the correlation window size from 5 to 10 minutes does not affect the results significantly; in the following this threshold is set to 5 minutes.

The left panel in Fig. 5 illustrates the activity of prefixes at AT&T during large events at the other three monitors. The CCDF plots show the fraction of large events (on the y-axis) where at least $x\%$ of the prefixes affected by the large event are active. We observe that for most large events at remote monitors, only a small percentage of the affected prefixes are active at AT&T at the same time; in 95% of the cases, less than 5% of affected prefixes are also active at AT&T. *If we look closer at those large events where many prefixes are active also at remote monitors (the tail of the plot in Fig. 5), we see that they are mostly of type AW or WA*. This is intuitive, since events that affect the reachability of prefixes will often be propagated widely across the Internet. The corresponding plots for other combinations of monitors show similar results.[3]

To summarize, *the time series of large events show little correlation between different monitors. In addition, large events that are observed at one monitor have mostly negligible impact on other monitors. Therefore, the number and magnitude of observed large events are highly dependent on the monitoring point*.

C. Evolution of Large Events

Next, we turn to exploring the evolution of large events and their characteristics. More specifically, we investigate how the size, intensity, duration, and frequency of large events have changed over time. For each of these metrics, we use the Mann-Kendall statistical test for trend detection to determine if we can identify a trend at a 90% significance level.

The first observation is that for all these metrics, there are significant variations, both over time and across monitors. This is illustrated in the right panel in Fig. 5 showing the number of large events per month at Level-3 and AT&T. We observe that this number can vary by several orders of magnitude from one month to the next. *This highlights the importance of using sufficiently long measurement periods when looking for trends in the evolution of large events*.

Keeping this in mind, *we find few clear trends in the evolution of large events*. The exception is Level-3, where we see a clear increase in the impact of large events, as shown in the right panel in Fig. 5. Looking closer at the large events in Level-3, we note that this increase can be attributed to events of type AAS. In particular, after 2006 we see a large number of large events where the updates contain changes in a COMMUNITY attribute, related to the geographic exit point in the Level-3 network. Hence, we believe that this increase

[3]We see a somewhat higher correlation at the Sprint monitor during large events at France Telecom.

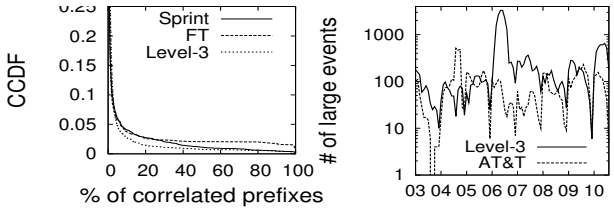

Fig. 5: The impact of remote large events at AT&T (left), Large events per month (right).

is caused by a local configuration change at Level-3, and does not represent a general trend in the Internet.

Another interesting observation is that the occurrence of large events in AT&T tends to be more temporally clustered than in the other monitors. While we typically observe several days in each month without any large events, the median inter-arrival time for large events in AT&T is in the order of tens of seconds. For the other monitors, the median inter-arrival time is in the order of hours. The low inter-arrival times at AT&T are probably caused by the fact that almost 50% of the large events at AT&T are of types AW and WA as shown in Tab. II. It is likely that a failure of a large number of routes will shortly be restored if it is caused by a transient loss of reachability.

Unlike duplicates discussed in the previous section, *updates caused by large events are necessary for correct routing, and they cannot be viewed as artifacts of the protocol implementation. However, they are less important for the long-term evolution of Internet-wide churn*. The third column in Fig. 2 shows the churn, after removing updates due to large events. Comparing this time series with the churn after removing duplicates, we see that most remaining large spikes in the duplicate-free churn are related to large events. Even though the remaining time series, after excluding the impact of duplicate updates and large events, are much smoother, they still show several significant level shifts; they are the subject of the next section.

VII. Analyzing level shifts

The time series (for the AT&T and Level-3 monitors in particular) are still dominated by level shifts where the magnitude of churn changes substantially and abruptly. The presence of these level shifts makes it difficult to reliably detect long-term trends. Instead of trying to automatically identify a plausible root cause for every level shift (a difficult and error-prone task), we make an in-depth "manual" analysis of few major level shifts. We focus our analysis on the AT&T and Level-3 monitors.

AT&T: The AT&T time series involves several clear level shifts, in addition to a long period of increased activity spanning 1.5 years from Jan'04 to Jun'05. Overall, we identified five distinct level shifts at that network. The first level shift is the long period of increasing activity from December-11-2003 to March-01-2005. The second level shift started immediately after the first period and lasted for one month. The third and fourth level shifts took place from February-15-2006 to March-31-2006 and from July-31-2006 to September-

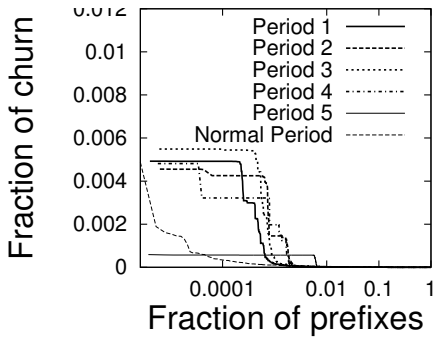

Fig. 6: Churn contribution from the most active prefixes during five level shifts at the AT&T monitor.

25-2006, respectively. Finally, the fifth level shift took place from August-19-2009 to October-16-2009.

Fig. 6 shows the fraction of total churn contributed by each prefix during our five activity periods, sorted by the activity level of each prefix. *We observe in the first four periods that there is a very small set of prefixes that contributed the majority of churn*. In the fifth period on the other hand, a relatively larger set of prefixes was responsible for the majority of churn. For comparison, we also include a curve for the churn in 2008, which does not contain any level-shifts. This clearly shows the abnormality during the level shift periods.

During period-1, a small set of 148 prefixes (i.e 0.1% of the total number of prefixes) contributed 49.8% of the total churn. We investigated the activity patterns of these prefixes by examining the inter-arrival times of their updates. The prefixes can be classified into three groups based on their median updates inter-arrival times.

The first group consists of prefixes with median update inter-arrival time at 58 seconds. When investigating their update patterns we find that these prefixes belong to AS21617 and are reached through the path {7018,701,21617} where 7018 is the monitor AS. During this period the previous group of prefixes flapped up and down almost in every minute. It is reasonable to believe that this long-lasting and high-frequency flapping pattern is caused by a flapping link or misconfiguration.

The prefixes that fall into the second group have a median update inter-arrival time of 65 seconds. We find that they are originated by either the monitor AS (i.e. 7018) or its direct customers. During this period this group of prefixes exhibited a change which is either a withdrawal or a re-announcement approximately every minute. Although the prefixes in the first and second groups have nearly identical median inter-arrival times, the second group stands out with a very regular activity pattern. The 90-th percentile of the update inter-arrival times is approximately equal to the median, which confirms a strict periodicity in these updates. This implies that these updates are caused by an anomaly that changes the path selection at regular intervals, rather than a flaky link or some adaptive load balancing method that would give a more irregular pattern.

The last group includes prefixes with a median update inter-arrival time at 196 seconds. We find that these prefixes belong to AS1938, and their AS path was switching between {7018 10888 24 11537 20965 2200 1938} and {7018 10888 11537 20965 2200 1938}. It is difficult to spot the root cause in this case. However, in the FT and Level-3 datasets we observe similar flapping patterns that involve switching some prefixes' next hop from AS24 (NASA) to other ASes. Therefore, this activity might be caused by some instability in or near AS24 that lasted for a long time.

Period-2 started immediately after the end of period-1, and lasted for one month. There is a small set of 170 prefixes that generated 71.7% of the total churn during this period. The main cause of this level shift is a small set of prefixes belonging to General Electric's AS (AS80). These prefixes continuously flapped between the direct route {7018 80} and a longer route with AS1239 (Sprint) as a next hop, i.e.{7018 1239 80}. Note that AS80 is a stub AS and does not announce many prefixes. Still, the frequency of route changes is high enough to create this radical increase in churn.

In Period-3 and Period-4, we find that the level shifts are caused by leaking of private AS numbers into the global routing system. Private AS numbers (ranging from 64512 to 65535) are used to divide large ASes into multiple smaller domains connected by eBGP, or they can be assigned to stub ASes that want to use BGP with their upstream provider but do not want to be part of the global routing system. Private AS numbers should be removed from routing updates that are sent to the global BGP system. During these two level shifts, updates containing private AS numbers are responsible for 54.3% and 70.5% of total churn respectively.

Period-5 is different from the other four periods in two respects. First, it involves a larger number of prefixes (2030). Second, the daily churn is much higher during the shift period, about 1.8 million updates per day. We observed that a set of prefixes reached by AT&T through AS7132 (SBIS-AT&T Internet service) and AS2685(AT&T Global Network Services) flapped with a high frequency during the shift period – approximately every 80 seconds. A discussion in the NANOG mailing list pointed to this level shift and observed the same flapping behavior [18].

Level-3: The data shows a clear level shift in the Level-3 time series from March-01-2006 to August-31-2006. Following a similar analysis as in AT&T, we find that the increased activity can be attributed to a set of flapping prefixes, which changed their AS-PATH continuously from {3356 3561 4134 X} to {3356 1239 4134 X} or vice versa, where X represents the rest of the AS path. Here we see how AS3356 (Level-3) alternated between two different neighboring ASes, AS3561 (Savvis) and AS1239 (Sprint) to reach AS4134 (China-Backbone). Note here that Savvis is owned by Level-3 and hence the route through Savvis is preferred. When this route is lost, Level-3 selects the backup route through Sprint. The frequency of this flapping for each prefix is between once every 10 minutes and once every 20 minutes. However, China-Backbone is a major transit provider, and Level-3 selects it as the preferred path for more than 2000 destination prefixes. Hence, a single change will trigger a large number of updates.

We also identified a second level shift in the Level-3 time series, that took place from June-15-2010 to 31-July-2010. We find that this level shift is caused by persistent flapping in reaching prefixes originated by AS9808 (Guangdong Mobile Communication) and its customers.

The previous analysis shows that level shifts are usually caused by specific failures or misconfigurations in or at the border of the monitored AS. The left column in Fig. 8 shows the churn time series after filtering out all updates attributed to the level shift events previously described.

VIII. The growth of baseline churn

In this section, we analyze the growth of the churn time series after removing duplicate updates, large events, and the level shifts of the previous section. We refer to this time series as the "baseline churn". We also analyze the time series of peak churn, measured from the busiest 1-minute period of each day.

A. Baseline churn

Compared to the raw time series, the baseline churn is much smoother and shows more correlation across monitors (see Fig. 8). The Kendall's τ rank correlation coefficient between the AT&T, Level-3, and Sprint baseline time series is around 0.5, which is almost double the highest value observed in the raw time series (0.25). *This increase suggests that our approach has filtered out many of the effects that affect only a limited part of the Internet*. The cross-correlation between the three North American monitors and FT is lower (around 0.4). This is likely caused by differences in geographical presence.

Next, we use statistical methods to characterize the evolution of baseline churn. The application of linear regression on the baseline time series results in a low Pearson's correlation coefficient (0.03 to 0.42, depending on the monitor), since even the baseline churn contains some spikes and small level shifts. Therefore, we rely on non-parametric statistics and in particular on the Mann-Kendall statistical test for trend detection. The Mann-Kendall test reports that there is a statistically significant increasing trend in the baseline time series in all four monitors at a 90% significance level. Actually, both the non-parametric and parametric (linear regression) tests give similar estimates for the slope of the increasing trend. Table IV presents the estimated slopes in additional updates per day.

The same Table also shows the estimated relative churn increase during the study period. This figure is calculated based on the estimated slope and the median daily churn rate during the first 3 months as starting point. The two estimation techniques are in reasonable agreement with each other. Note that the estimated increase covers a period of six years for FT and Sprint, while it spans seven years and eight months for AT&T and Level-3. During the first six years the daily churn grew by about 50% at AT&T and 69% at Level-3, which indicates a faster growth than at FT and Level-3. Interestingly, the estimated increase reported in Table IV shows a faster growth at both AT&T and Level-3 during the last 20 months (about 30%) than during the first six years. *The significant differences between monitors are not surprising, since different monitors have different sets of customers and peers and different internal configuration*.

Next, we compare the growth in baseline churn to the growth in different measures of the global routing system. The left panel in Fig 7 shows the average daily number of baseline updates per prefix at the AT&T monitor.[4] Similarly, the middle panel shows the average daily number of updates per AS, while the left panel shows the same per distinct AS-PATH. These values are sampled once per month in the period from Jan'04 to Sep'10. We calculate the average number of daily (baseline) updates in each month and divide by the corresponding comparison metric. In all plots we also show the linear regression estimate. The Mann-Kendall statistical test for trend detection identifies a decreasing trend in the average number of updates per prefix and per distinct AS-PATH, at a 90% significance level. However, no significant trend is detected in the case of updates per AS. *The average number of updates contributed by each AS is almost constant at around 5 updates*.

In other words, the increase in the baseline churn is slower compared to the growth of the routing table size and the number of distinct AS-PATHs. This is in agreement with the data presented in Fig. 1. During our study period, the number of routable prefixes and distinct AS-PATHs increased by 168% and 163% respectively, while the baseline churn has increased by about 100%.

We also observe that baseline churn growth is similar to the increase in the number of ASes; the middle panel in Fig. 7 illustrates no change in the average number of updates per AS as the number of ASes increases. The data in Fig. 1 indicates an increase in the number of ASes by 143%. However, after Jan-2004 the number of ASes has increased by about 112%, which is close to the growth of the baseline churn (100%). This observation suggests that *the growth in baseline churn is mainly driven by the growth in the number of ASes, rather than the number of prefixes*. Most of the growth in the number of ASes occurs at the periphery of the Internet, in the form of stub ASes. The stable relationship between baseline churn and the number of stub ASes suggests that the latter generate updates at a stable rate, even though the average number of prefixes per AS has increased. A deeper investigation of the relationship between churn and other Internet-wide metrics requires further research.

B. Daily peak activity

The churn rates presented so far are daily averages. The peak churn rate in shorter time scales may be more important in terms of the processing load imposed on routers. Here, we examine the growth of the peak daily churn rate, measured as the *maximum 1-minute churn on each day*. We refer to this time series as the "daily 1-minute peak churn".

The plots in the second and third columns of Fig. 8 show the daily 1-minute peak churn in the raw time series and in the baseline time series, respectively. *A first observation is that the daily peak activity in the raw time series is much higher than in the baseline time series: on average, there is an order of magnitude difference between the two time series across all monitors, and on some days the difference can reach up to two orders of magnitude*.

The Mann-Kendall test reports an increasing trend in the raw and baseline daily peak churn across all four monitors. The

[4] Other monitors show similar trends.

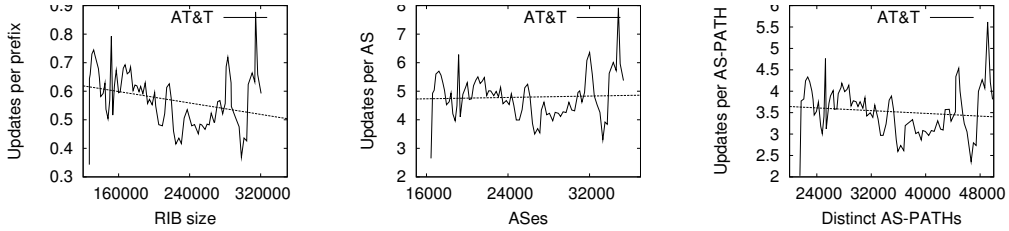

Fig. 7: Churn evolution with respect to different size measures of the global routing system.

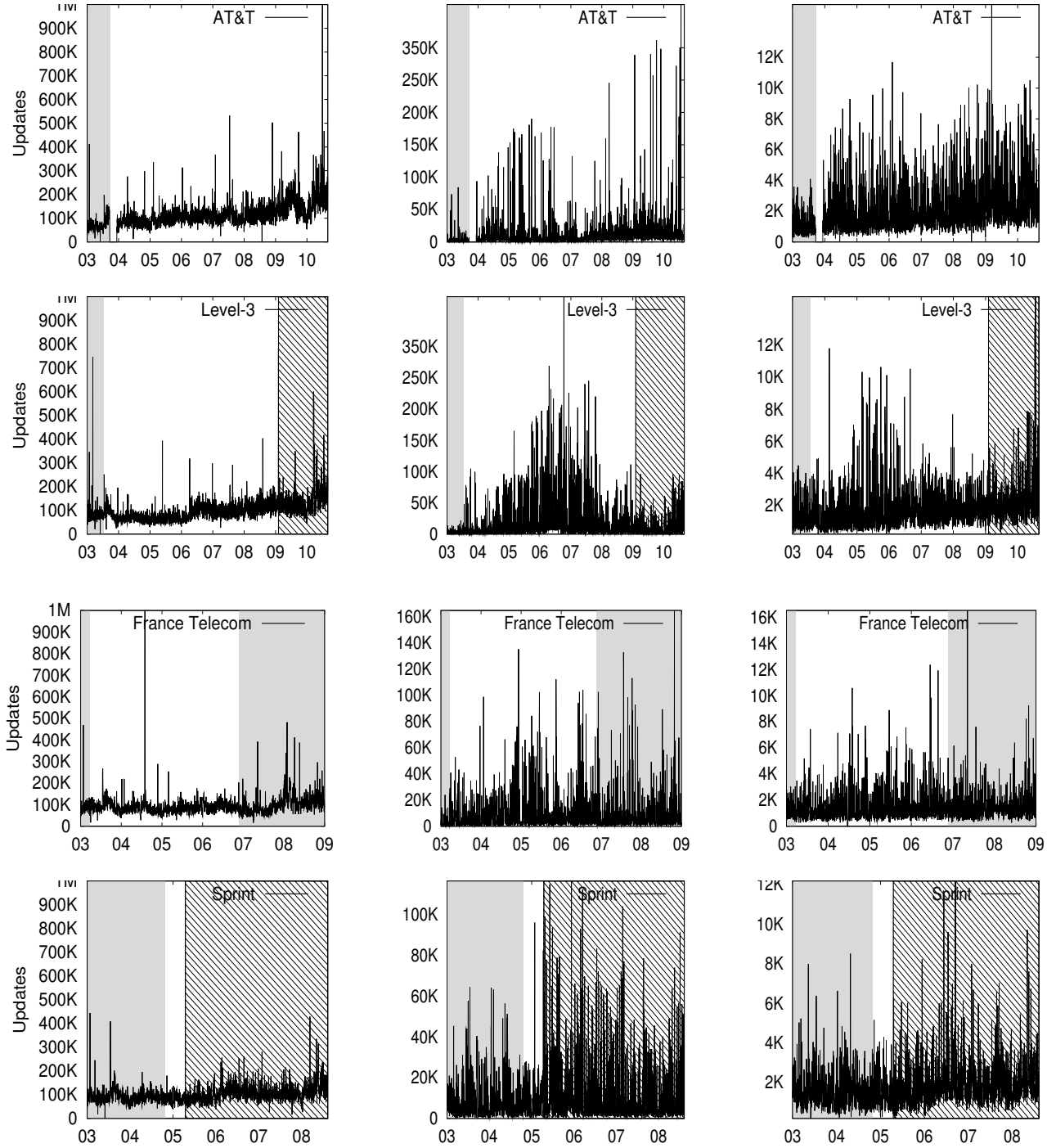

Fig. 8: Baseline daily total churn (left), 1-minute peak churn per day in the raw time series (middle), and 1-minute peak churn per day in the baseline time series (right) in AT&T , Level-3 , FT, Sprint, from top to bottom respectively.

TABLE IV: Baseline churn growth: Mann-Kendall slope estimate in updates per day, and the estimated relative churn increase during our study period. The parametric estimates are also shown.

Monitor	AT&T	Level-3	FT	Sprint
M-K slope	33.82	28.65	7.79	14.38
Est. increase	101.2%	103.7%	20.0%	29.4%
Lin. regr. slope	40.06	31.55	6.50	16.42
Est. increase	119.9%	114.2%	15.5%	33.3%

TABLE V: Daily 1-minute peak churn growth.

Monitor	AT&T	Level-3	FT	Sprint
Raw peak churn				
M-K slope	2.22	0.81	0.27	-
Est increase	168.4%	171.3%	50.5%	-
Baseline peak churn				
M-K slope	0.50	0.42	0.10	0.29
Est increase	100.0%	113.4%	20.9%	39.3%

exception is the raw time series at the Sprint monitor, where no trend could be detected. Table V presents the slope and the relative estimated increase at each monitor. In order to compare all four monitors, we compute the M-K slope of the raw and baseline peak churn during the first six years [5]. During this period, the M-K slope of the AT&T baseline peak churn (0.32) was about 1/3 of the M-K slope of the AT&T raw peak churn (1.03). We observe a similar trend in the Level-3 monitor with the M-K slope of the baseline peak churn (0.43), while the M-K slope of the raw peak churn is 1.57. The modest growth at the FT and Sprint monitors is probably due to the use of rate-limiting timers. The noisy nature of the raw time series makes it difficult to get accurate growth trends, and so these numbers should be viewed only as rough estimates.

We observe that the estimated relative growth in the daily 1-minute peak churn rate is somewhat higher for the raw time series than for the baseline. *This indicates that the impact, in terms of peak churn, of duplicates and effects that are not related to the long-term evolution of churn increases with time*. For the baseline time series, *the increase in the daily 1-minute peak level is comparable to the increase in the total daily churn*.

Finally, we investigate to what extent the daily 1-minute peak churn is influenced by the use of rate-limiting timers. We compare the median daily 1-minute peak churn calculated in a three-month window immediately before and after each change in the rate-limiting configuration at the FT, Level-3, and Sprint monitors. Fig. 9 shows the churn in the 3-month period before and after the MRAI timer was turned on in late 2006 at the FT monitor, for the raw and baseline time series (the horizontal lines in the plots show the median level of churn).*We find that the rate-limiting timer has no clear effect on the daily 1-minute peak churn in the baseline time series. However, in the raw time series, there is a clear increase in the peak churn when the rate-limiting timer is off*. The peak churn increases by a factor 1.1 and 1.2 during the first and second transitions at Sprint, 2.0 and 0.0 during the first and second transitions at Level-3, and 3.7 and 2.8 during the first and second transitions at FT.

[5]We do not include the last 1.5 years because the Sprint and FT monitors were unavailable.

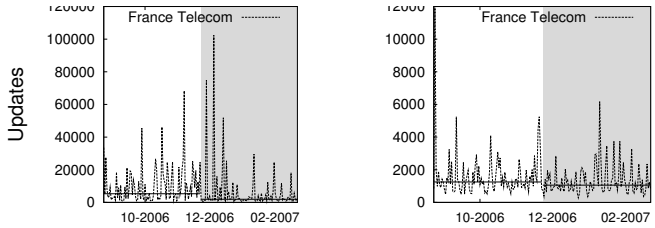

Fig. 9: Churn before and after MRAI timer is turned on at the FT monitor (left: raw 1-min peak churn, right: baseline 1-min peak churn).

These findings show that the effect of the rate-limiting timer is much stronger on the raw time series than on the baseline. *This implies that the rate-limiting timer is mostly effective at filtering out some duplicate updates and effects that are not related to the long-term evolution of churn.*

IX. Related work

Interdomain routing dynamics and scalability have been active topics of research during the last decade or so. In the following we only review the most relevant related work.

Labovitz et al.[12] were the first to show that BGP suffers from excessive churn caused by pathological protocol behavior and he suggested practical ways to fix broken BGP implementations. In follow-up work [13], they found that better router implementations had reduced churn by an order of magnitude, but that duplicate announcements still contributed much unnecessary churn. Our findings confirm that this is still the case, and that this type of updates is responsible for most large spikes. Mahajan et al. reported [15] that BGP misconfigurations are pervasive and cause an increase in the processing load of routers. A recent measurement study [14] concluded that the state of BGP routing is now "healthier" than it was a decade ago.

The phenomenon of *path exploration* was first discussed by Labovitz et al. [11]. A later study by Griffin and Premore examined the effectiveness of the MRAI timer to limit path exploration [7].

Network topology plays a role in the observed churn as well. In a recent measurement study, it was shown that path exploration is less severe in the core of the Internet than at its periphery [19]. It has also been shown that events at the edge of the network affect a larger number of ASes than those at the core [30]. In a recent study [6], we investigated the role of various topological factors, including multihoming, hierarchy and peering links, as well as the role of the rate limiting timer on BGP churn growth.

Another set of studies analyzed the contribution of different ASes and prefixes to the observed churn. Broido et al. [2] showed that a small fraction of ASes is responsible for most of the churn seen in the Internet. Similarly, several other papers [22], [27] reported that a small subset of prefixes are responsible for a large percentage of churn. Recently, Cittadini et al. [3] investigated the impact of prefix de-aggregation on BGP dynamics and showed that the de-aggregated prefixes do not generate a large number of updates in comparison to their number. In addition, they concluded that the increase in BGP

dynamics is caused by growth at the periphery of the AS-level topology.

An earlier study by Huston and Armitage [10] reported that BGP churn increases at a much faster pace than the routing table size. During 2005, the daily rate of update messages almost doubled, while the size of the routing table grew by only 18%. Our study, based on a much longer study period and a larger number of monitors, gives a more optimistic view for the churn growth rate. However, a recent study by Huston [9] concluded that BGP churn increases at a much slower pace than the routing table size, in agreement with the findings in our work.

X. Conclusions

This study has investigated the evolution of churn at four monitors located in the core of the Internet during a period of up to seven years and eight months. The corresponding time series are very bursty, with large spikes and level-shifts. We have performed an in-depth analysis of the time series in order to identify and explain the main sources of churn.

We have found that up to 40% of route announcements are redundant and they are not needed for correct protocol behavior. These duplicate announcements are also responsible for most large spikes in the churn time series. The remaining spikes are caused by large routing events that affect 2000 or more prefixes simultaneously. The impact of large events is mostly confined to a single monitor; we see little correlation in large events between different monitors. There are no clear trends in the size, intensity, duration or frequency of large events during our study period. We have also identified the underlying reasons for the most severe churn level-shifts. These are normally caused by configuration mistakes or other anomalies in or at the border of the monitored AS. Our findings suggest that the most effective short-term solutions for limiting churn are BGP implementation improvements that filter out redundant updates, and methods that can detect (long-lasting) configuration mistakes and other anomalies that result in sustained high churn.

We have also shown that *there is a long-term increasing trend in the identified baseline churn*, but at the same time, *the growth rate is relatively low*. We find that the *churn rate increases more slowly than the number of prefixes in the routing table*. While the routing table grew by about 168% during our study period, the baseline churn rate grew at most by about 100%. We have also observed that *the growth of the baseline churn is close, in magnitude, to the growth in the number of ASes*.

There are several reasons why we only see a slow increase in the baseline churn compared to the growth of the routing table size. On one hand, configuration management systems and operational experience are improving. Also, the observed increasing connectivity in the Internet [4] can play a positive role, since more failures can be handled locally if an alternate route is known.

We have also investigated the daily 1-minute peak churn rate, and found that this is an order of magnitude higher in the raw time series compared to the baseline. These time series are very noisy, but they appear to be slowly growing with time.

In future work, we want to further investigate the slow growth of the baseline churn and its close relation to the number of ASes.

Acknowledgment

We would like to thank Samantha (Sau Man) Lo for her inputs and help during the early stages of this work. We also thank Tarik Cicic for his help with measurement data. We are very grateful to the RouteViews project because this study would not be possible without their long-term efforts.

References

[1] Routeviews project page. http://www.routeviews.org.

[2] Andre Broido, Evi Nemeth, and kc claffy. Internet expansion, refinement, and churn. In *European Transactions on Telecommunications*, January 2002.

[3] Luca Cittadini, Wolfgang Muhlbauer, Steve Uhlig, Randy Bush, Pierre Francois, and Olaf Maennel. Evolution of Internet address space deaggregation: Myths and reality. *IEEE Journal on Selected Areas in Communications*, 2010.

[4] Amogh Dhamdhere and Constantine Dovrolis. Ten years in the evolution of the Internet ecosystem. In *In the Proceedings of ACM SIGCOMM/USENIX Internet Measurement Conference (IMC)*, 2008.

[5] Ahmed Elmokashfi, Amund Kvalbein, and Tarik Cicic. On update rate-limiting in BGP. In *To appear in IEEE ICC*, 2011.

[6] Ahmed Elmokashfi, Amund Kvalbein, and Constantine Dovrolis. On the scalability of BGP: the roles of topology growth and update rate-limiting. In *Proceedings ACM CoNEXT*, 2008.

[7] Tim Griffin and Brian Premore. An experimental analysis of BGP convergence time. In *Proceedings ICNP*, 2001.

[8] Myles Hollander and Douglas A. Wolfe. *Nonparametric statistical methods*. Wiley, second edition, 1999.

[9] Geoff Huston. BGP in 2009 (and a bit of 2010). Presentation at ARIN XXV meeting, TORONTO, Canada, April 2010.

[10] Geoff Huston and Grenville Armitage. Projecting future IPv4 router requirements from trends in dynamic BGP behaviour. In *Proceedings ATNAC*, Australia, dec 2006.

[11] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed Internet routing convergence. In *Proceedings ACM SIGCOMM*, pages 175–187, 2000.

[12] Craig Labovitz, G Robert Malan, and Farnam Jahanian. Internet routing instability. In *Proceedings ACM SIGCOMM*, Cannes, France, 1997.

[13] Craig Labovitz, G. Robert Malan, and Farnam Jahanian. Origins of Internet routing instability. In *Proceedings IEEE INFOCOM 1999*, New York, NY, March 1999.

[14] Jun Li, Michael Guidero, Zhen Wu, Eric Purpus, and Toby Ehrenkranz. BGP routing dynamics revisited. *Computer Communications Review*, April 2007.

[15] Ratul Mahajan, David Wetherall, and Tom Anderson. Understanding BGP misconfiguration. In *Proceedings ACM SIGCOMM*, pages 3–16, New York, NY, USA, 2002. ACM.

[16] Z. Morley Mao, Randy Bush, Timothy G. Griffin, and Matthew Roughan. BGP beacons. In *IMC '03: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 1–14, New York, NY, USA, 2003. ACM.

[17] David Meyer, Lixia Zhang, and Kevin Fall. Report from the IAB workshop on routing and addressing. http://tools.ietf.org/id/draft-iab-raws-report-02.txt, apr 2007.

[18] http://www.mail-archive.com/nanog@nanog.org/msg15962.html, October 2009.

[19] Ricardo Oliveira, Beichuan Zhang, Dan Pei, Rafit Izhak-Ratzin, and Lixia Zhang. Quantifying path exploration in the Internet. In *Proceedings IMC*, Rio de Janeiro, Brazil, oct 2006.

[20] Jong Han Park, Dan Jen, Mohit Lad, Shane Amante, Danny McPherson, and Lixia Zhang. Investigating occurrence of duplicate updates in BGP announcements. In *PAM*, pages 11–20, 2010.

[21] Yakov Rekhter, Tony Li, and Susan Hares. A border gateway protocol 4 (BGP-4). RFC4271, January 2006.

[22] Jennifer Rexford, Jia Wang, Zhen Xiao, and Yin Zhang. BGP routing stability of popular destinations. In *IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*, pages 197–202, New York, NY, USA, 2002. ACM.

[23] http://www.mail-archive.com/rrg@irtf.org/msg02714.html, March 2010.
[24] Virginie Schrieck, Pierre Francois, Cristel Pelsser, and Olivier Bonaventure. Preventing the unnecessary propagation of BGP withdraws. In *Proceedings of the 8th International IFIP-TC 6 Networking Conference*, NETWORKING '09, pages 495–508, Berlin, Heidelberg, 2009. Springer-Verlag.
[25] Renata Teixeira, Aman Shaikh, Tim Griffin, and Geoffrey M. Voelker. Network sensitivity to hot-potato disruptions. In *Proceedings of SIGCOMM*, pages 231–244, Portland, Oregon, USA, August 2004.
[26] George Varghese. *Network Algorithmics*. Morgan Kaufmann, San Fransisco., 2004.
[27] Ricardo V.Oliveira, Rafit Izhak-Ratzin, Beichuan Zhang, and Lixia Zhang. Measurement of highly active prefixes in BGP. In *Proceedings IEEE Globecom*, 2005.
[28] Jian Wu, Zhuoqing Morley Mao, Jennifer Rexford, and Jia Wang. Finding a needle in a haystack: pinpointing significant BGP routing changes in an IP network. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation - Volume 2*, NSDI'05, pages 1–14, Berkeley, CA, USA, 2005. USENIX Association.
[29] Beichuan Zhang, Vamsi Kambhampati, Mohit Lad, Daniel Massey, and Lixia Zhang. Identifying BGP routing table transfers. In *MineNet '05: Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data*, pages 213–218, New York, NY, USA, 2005. ACM.
[30] Xiaoliang Zhao, Beichuan Zhang, Andreas Terzis, Daniel Massey, and Lixia Zhang. The impact of link failure location on routing dynamics: A formal analysis. In *Proceedings ACM SIGCOMM Asia Workshop*, apr 2005.

Dr. Ahmed Elmokashfi is a postdoctoral fellow at Simula Research Laboratory in Oslo, Norway. He received the B.Sc. degree in Electrical and Electronics Engineering from the University of Khartoum in 2003, the M.Sc degree in Telecommunication form Blekinge Institute of technology in 2007, and the Ph.D. degree from University of Oslo in 2011. The focus of his Ph.D. thesis has been on the scalability of BGP inter domain routing with respect to churn. In particular, he has worked on how the Internet topology and different protocol mechanisms and implementations influence the observed update rate. Elmokashfi research interests include IP routing, routing scalability, measurements, and network science.

Dr. Amund Kvalbein is a Senior Research Scientist at Simula Research Laboratory. He received the M.S. degree from the University of Oslo in 2003, and the Ph.D. degree from the same institution in 2007. He spent one year as a visiting Post Doc at Georgia Institute of Technology in 2007-2008. At Simula, he is the leader of Resilient Networks, an activity focusing on increased network resilience in wired and wireless networks. His main research interests are network layer protocols, in particular issues related to fault tolerance, scalability and robustness under unexpected operational environments.

Dr. Constantine Dovrolis is an Associate Professor at the College of Computing of the Georgia Institute of Technology. He received the Computer Engineering degree from the Technical University of Crete in 1995, the M.S. degree from the University of Rochester in 1996, and the Ph.D. degree from the University of Wisconsin-Madison in 2000. He joined Georgia Tech in August 2002, after serving at the faculty of the University of Delaware for about two years. He has held visiting positions at Thomson Research in Paris, Simula Research in Oslo, and FORTH in Crete. His current research focuses on the evolution of the Internet, Internet economics, and on applications of network measurement. He is also interested in network science and in applications of that emerging discipline in the understanding of complex systems. Dr. Dovrolis has been an editor for the IEEE/ACM Transactions on Networking, the ACM Communications Review (CCR), and he has served as the Program co-Chair for PAM'05, IMC'07, CoNEXT'11 and as the General Chair for HotNets'07. He received the National Science Foundation CAREER Award in 2003.