Scaling Inter-domain Routing System via Path Exploration Aggregation

Xiaoqiang Wang, Peidong Zhu, Xicheng Lu, Kan Chen and Huayang Cao
School of Computer, National University of Defense Technology
Changsha, Hunan 410073 - China
[e-mail: {wangxiaoqiang,pdzhu,xclu,jeffee,hycao}@nudt.edu.cn]
*Corresponding author: Xiaoqiang Wang

Received October 15, 2012; revised February 23, 2013; accepted February 18, 2013; published March 29, 2013

Abstract

One of the most important scalability issues facing the current Internet is the rapidly increasing rate of BGP updates (BGP churn), to which route flap and path exploration are the two major contributors. Current countermeasures would either cause severe reachability loss or delay BGP convergence, and are becoming less attractive for the rising concern about routing convergence as the prevalence of Internet-based real time applications. Based on the observation that highly active prefixes usually repeatedly explore very few as-paths during path exploration, we propose a router-level mechanism, Path Exploration Aggregation (PEA), to scale BGP without either causing prefix unreachable or slowing routing convergence. PEA performs aggregation on the transient paths explored by a highly active prefix, and propagates the aggregated path instead to reduce the updates caused by as-path changes. Moreover, in order to avoid the use of unstable routes, PEA purposely prolongs the aggregated path via as-path prepending to make it less preferred in the perspective of downstream routers. With the BGP traces obtained from RouteViews and RIPE-RIS projects, PEA can reduce BGP updates by up to 63.1%, shorten path exploration duration by up to 53.3%, and accelerate the convergence 7.39 seconds on average per routing event.

Keywords: Routing scalability, BGP churn, RFD, Path Exploration, Aggregation

This work was primarily finished during Xiaoqiang's visit at Université Catholique de Louvain, Belgium, and a preliminary version had appeared in the 14th IEEE Global Internet Symposium held in conjunction with IEEE INFOCOM 2011, Shanghai, China. This version repositions the target, improves the algorithm and provides more experimental details. This research was supported in part by the NSF program of China under Grant No.61170285, No. 61100223 and No.61103189. The authors would like to thank Prof. Olivier Bonaventure (ICTEAM, Université Catholique de Louvain) for his valuable advices and comments on this work.

http://dx.doi.org/10.3837/tiis.2013.03.005

1. Introduction

The Internet has evolved from an experimental network to the very important information infrastructure of human society in recent several decades, and now is largely different in size from its original design. BGP (Border Gateway Protocol) is the defacto inter-domain routing protocol that used to exchange network reachability among ASes (Autonomous System), each of which consists of a set of routers under single technical administration. As a consequence of continuous evolution of the Internet, BGP is now facing severe scalability problem, especially the inflated route table and the rapidly increasing rate of BGP updates (BGP churn) [1]. This paper targets at the the issue of increasing churn.

Apart from the increasing entries in BGP routing table, unstable routes (also known as route flap) and BGP path exploration are the two major contributors to the rapidly increasing churn. Route flap refers to an excessive rate of BGP updates to the advertised reachability of a subset of Internet prefixes [2], and a significant fraction of BGP churn is associated to a small number of highly active prefixes [3,4]. Path exploration suggests a phenomenon that in response to routing failure or routing change, some BGP routers may try several transient routing paths before converging to the final choice. RFD (Route Flap Damping) [2] and MRAI (Minimum Route Advertisement Interval) [5] are the only two countermeasures that have been deployed in the current Internet against route flap and path exploration respectively. However, since they would either cause severe reachability loss [6] or delay BGP convergence [7], they are becoming less attractive for the rising concern about routing convergence as the prevalence of Internet-based real time applications. In 2006, RIPE Routing Working Group recommended to turn off RFD in ISP (Internet Service Provider) networks for the negative effects of RFD have become the major concern [8].

To scale the inter-domain routing system, we focus on the churn produced by highly active prefixes and amplified by path exploration, and propose a novel approach, Path Exploration Aggregation (PEA), to reduce that part of churn. BGP updates are usually caused by as-path changes [9]. In particular, we find that a highly active prefix usually produces an excessive amount of BGP updates by alternating the as-path attribute among a small set of as-paths, showing *path locality*. Based on this observation, PEA replaces the transient as-paths explored by a highly active prefix during path exploration with their aggregation as-path. In this way, the routing regarding this prefix does not have to change as long as the incoming as-path is a member of the aggregation as-path. Moreover, PEA purposely prolongs the aggregated path via as-path prepending to lower its preference in downstream routers, to avoid the use of unstable routes. The most significant point that makes PEA a different solution from RFD and MRAI is that it would neither cause prefix unreachable nor slow routing convergence. With the BGP traces from RouteViews [14] and RIPE-RIS [18] projects, PEA can reduce BGP routing updates by up to 63.1%, shorten path exploration duration by up to 53.3%, and accelerate the convergence 7.39 seconds on average per routing event, outperforming RFD [2], RFD-HT [10] and a MRAI similar method PED [11] in all these three aspects.

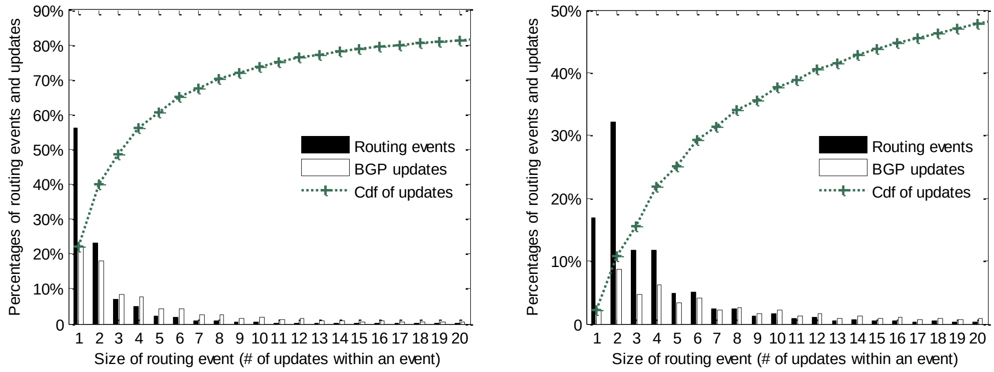
The remainder of this paper is structured as follows. Section 2 explains our motivations, and Section 3 presents the PEA algorithm and implementation details. In Section 4, taking RFD, RFD-HT and PED as references, we evaluate the reduction of BGP churn and the impact on convergence duration and convergence delay when PEA is deployed. Section 5 summarizes the related work and we conclude this paper in Section 6.

2. Motivation

PEA is motivated by two observations. The first observation highlights the important role of path exploration in producing BGP churn that the updates of the most active 1% prefixes due to path exploration are 6.48 times more than expected. The second observation reveals the path locality relevant to highly active prefixes, making the upcoming as-path changes predictable.

2.1 Path Exploration in BGP Churn

Path exploration suggests a phenomenon that in response to a routing failure or routing change, a BGP router may try several transient routing paths before converging to the final choice, and produce more than one BGP update accordingly. Based on the observation that updates sharing the same root cause often come in bursts, a time-based classification method is widely used in BGP routing dynamics studies [11,12,13] to group the multiple BGP updates caused by a trigger event into the same *routing event*. For details, two consecutive updates with inter-arrival smaller than time threshold T are assumed to be in the same event.



(a) Distribution across overall prefixes **(b)** Distribution across the most active 1% prefixes

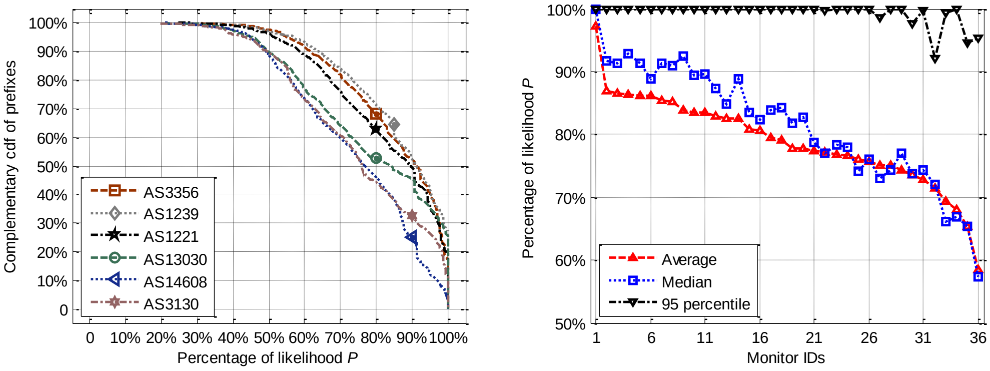
Fig. 1. Percentage of routing events/BGP updates within each class.

To understand the contribution of path exploration to BGP churn, we analyzed a ten-day BGP trace available from *rv2* (route-views2.routeviews.org) [14], ranging from Dec 1st to Dec 10th in 2010. We first identified and removed BGP data transfers due to session resets [15], and then filtered all the duplicated occurrences [16]. After that, BGP updates were classified into events per monitor and per prefix using T=5 minutes. At last, we get 40,345,948 routing events. For each routing event, only the last update reflects the converged state of BGP routing. In this sense, the others are considered to be caused by path exploration. Let n_e denote the number of routing events, and n_u the number of updates, we define **amplification factor** of path exploration to be $(n_u-n_e)/n_e$. Obviously, if no path exploration exists, n_u should be equal to n_e such that amplification factor will be zero.

We further classified those routing events according to the number of updates within each of them. **Fig. 1** shows the percentage of routing events and updates falling into each category. As seen in **Fig. 1(a)**, where overall Internet prefixes are considered, 56.1% of the routing events undergo no path exploration at all, i.e., each of them consists of one update, and they account for 22% of the observed updates. We can derive an equation from these two figures that $56.1\%\times n_e=22\%\times n_u$, making amplification factor 56.1%/22%-1=1.55. The amplification factor increases to 16.9%/2.26%-1=6.48 in **Fig. 1(b)**, where only the most active 1% prefixes are considered.

2.2 Path Locality

Path exploration is rooted in routing diversity that there are usually multiple routes between pairwised ASes [17]. An AS would explore its alternatives one by one until the final choice is reached when its primary route fails. In this sense, the number of as-paths that an AS can employ to reach a destination network is limited by the routing diversity in between. Moreover, BGP uses a “one-route-fits-all” model that only the most preferred route is selected for data forwarding and propagated to neighboring ASes. The bounded routing diversity and the preferential routing of BGP together may yield unbalanced routing selection that a small set of as-paths are selected as best path at a higher frequency than other paths, showing *path locality*.



(a) Path locality observed from 6 sample ASes **(b)** Path locality observed from 36 monitor ASes

Fig. 2. Path locality of the most active 1% prefixes. Likelihood *P* of a prefix is the percentage of the three most frequent as-paths’ occurrences in all the as-path occurrences for this prefix.

To confirm this conjecture, we conducted the following estimation of path locality based on the same BGP trace aforementioned. For each prefix observed from a monitor AS, we first extracted the set of as-paths that this AS had ever used to reach this prefix. Then we sorted those as-paths according to the number of their occurrences in the update stream. For each prefix, we defined a likelihood *P* as the probability of that a received BGP update contained an as-path among the three most frequent as-paths for this prefix over the ten-day study period. Since a small number of highly prefixes account for an out of portion number of BGP updates [3,4], we consider the most active 1% prefixes only (around 3,400 in number). **Fig. 2(a)** plots the ccdf of this likelihood across prefixes for six ASes. As we can see, more than 70% of the most active 1% prefixes in all those 6 ASes have this likelihood higher than 60%. Moreover, more than 25% of these prefixes are found to have explored ≤3 as-paths during our ten-day study period in AS1239 and AS13030. We want to emphasize that the likelihood *P* in a smaller time window may be higher, since the flapping routes per prefix may vary as time goes on [4].

We then extended the selected observation points to the overall 36 living *rv*2 monitors. The average, median and 95th percentile of the likelihood *P* regarding the most active 1% prefixes are shown in **Fig. 2(b)**, where monitors are sorted according to the averaged likelihood *P* observed from them. We do not show the maximum since it is always 100% across all the 36 monitors. **Fig. 2(b)** proves the wide and stable existence of path locality that even in the worst case, the averaged likelihood *P* can reach 58.5%.

3. Path Exploration Aggregation (PEA)

For a highly active prefix, PEA maintains a set of as-paths that are most frequently used to reach this prefix in recent period. Every as-path in this set will be replaced by the aggregation of all as-paths from this set before being propagated to downstream routers. In this way, as-path changes are no longer changes since their initial and ending state have become identical now. In addition, PEA prolongs the aggregated path via *as-path prepending* to indirectly inform downstream routers the instability of this prefix.

Table. 1 Route Aggregation vs. Path Exploration Aggregation(PEA)

Item		**Route Aggregation**	**PEA**
Involved objects		routes $r_1r_2\ldots r_n(n\geq2)$	Route r and m as-paths $p_1p_2\ldots p_m(m\geq0)$
NLRI(*prefix*)		A less specific prefix that covers all the NLRI attribute of $r_1r_2\ldots r_n$	r.NLRI
Path Attributes	*origin*	Prefers INCOMPLETE over EGP, then over IGP among the *origin* attribute of $r_1r_2\ldots r_n$	*r.origin*
	as-path	Aggregation of the as-path attribute of $r_1r_2\ldots r_n$	Aggregation of *r.as-path* and $p_1p_2\ldots p_m$
	next-hop	Either the next-hop when they have the same *next-hop* attribute, or the interface on the BGP speaker that performs this route aggregation	*r.next-hop*
	med	Routes with different MED attributes shall not be aggregated	*r.med*
	local-pref	Recalculated according to local policy	*r.local-pref*
	atomic-aggregate	Aggregated route will has this attribute *iff* one of the aggregating routes has this attribute.	*r.atomic_aggregate*
	aggregator	'AS number+Router IP' of the router where route aggregation is performed	'AS number+Router IP' of PEA router if m>0

PEA is not completely new, and it is a variation of Route Aggregation (RA), which is designed to reduce the amount of information that a BGP speaker must store and exchange with other BGP speakers [5]. As shown in **Table. 1**, these two aggregation technologies are very similar but still different. (1) While RA consists of several sub-aggregations separately applied to NLRI and path attributes of the same type, such as *origin*, *as-path*, *next-hop*, *med*, *local-pref*, *atomic-aggregate*, and *aggregator*, PEA involves only *as-path* aggregation. Therefore, PEA has lower complexity for fewer attributes are involved. (2) RA aggregates a group of routes destined for different prefixes, which are usually covered by a less specific prefix, but PEA aggregates routes regarding the same prefix. (3) The routes involved in RA are assured to be simultaneously available while the member as-paths from the buffered history in PEA may be outdated. For instance, PEA may aggregate several paths learned from the same BGP session into a new path, and those as-paths are obviously not simultaneously available under current BGP logic.

3.1 PEA Terminology

PEA can be simply modeled as a black box with an input and an output of routes. For each prefix d observed from input, PEA maintains a prefix specific auxiliary data structure, as shown in **Fig. 3**. For details, this structure includes a *penalty* value P_d (or a figure of merit value in [2]), the latest *time* t_d(UTC format) when this prefix specific structure is updated, the history of as-paths recently explored by routes regarding *d* (denoted as H_d), and the latest input and output route with regard to prefix *d*, denoted as $r^{(i)}_d$ and $r^{(o)}_d$ respectively. Since $r^{(i)}_d$ and $r^{(o)}_d$ are not exclusive to PEA and in fact they can be retrieved from Adj-Rib-In and Adj-Rib-Out [5], we list these two routes just for conceptual completence.

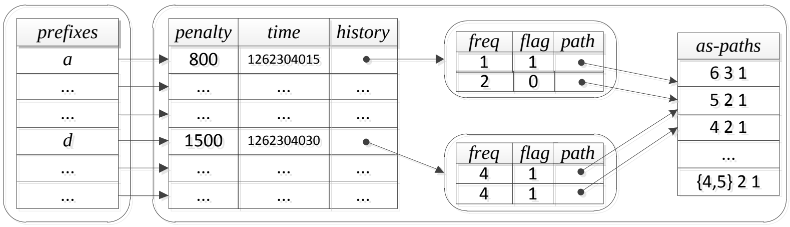


Fig. 3. Auxiliary data structure that PEA maintains for each prefix

Table. 2 PEA terminology

Terms	Definition
$r^{(i)}_d$ $r^{(o)}_d$	Latest route received from input regarding prefix *d* Latest route written to the output regarding prefix *d*
P_d t_d H_d $f_{d,p}$ $g_{d,p}$	Prefix penalty of prefix *d* Latest time that the prefix specific date structure of prefix *d* is updated Set of as-paths observed from input relevant to prefix *d* Frequency of path *p* of being observed from the routes related to *d*, $p \in H_d$ Flag bit, indicating whether $p(p \in \mathrm{H}_d)$ is an aggregation member of $r^{(o)}_d$
λ,*Halftime* Th_c Th_r	Decay factor, λ is defined together with H*alftime* that $e^{-(\lambda \times \mathrm{Half}time)}=1/2$ Cutoff threshold above which a prefix's transient route should be aggregated Reuse threshold below which a prefix's transient route should not be aggregated

Each element in H_d is a tuple $(p, f_{d,p}, g_{d,p})$, where *p* is an *as-path*, $f_{d,p}$ the occurrence frequency of *p* and $g_{d,p}$ the indicator that whether *p* is involved in the latest output path, i.e., $r^{(o)}_d$*.as-path*, which may or may not be an aggregation path. Considering that an as-path usually carries several prefixes, a global as-path sharing is used to improve storage efficiency. **Table. 2** summarizes the PEA terminology, where λ, *Halftime*, Th_c and Th_r are notions cited from RFD. For details, *Halftime* refers to the time for a penalty or path frequency to decay to half its value, λ is the decay factor defined together with *Halftime*, and Th_c/Th_r refers to the threshold above/below which a prefix's transient routes should be/not be aggregated.

3.2 PEA Algorithm

PEA algorithm consists of three phases, triggering, damping and releasing, as shown in **Fig. 4**. The first two phases are event-driven that they can be only activated by the received BGP routes from the input, while the third phase is periodically scheduled to release the resources relevant to dampend routes.

3.2.1 PEA Triggering Phase

In this phase, PEA first updates the auxliary data structure relevant to the received route, and then determines whether it is necessary to enter the next phase, i.e., damping phase.

For a newly received route *r* with NLRI attribute *d* and *as-path* attribute *p*, PEA first decays the prefix penalty P_d over time (line 4), and then increases it according to the determined change type (procedure DETERMINECHANGETYPE, line 5) and the parameter setting in **Table. 3**.

After that, PEA decays the frequency of each path in H_d, and increases $f_{d,p}$ by 1 (procedure RENEWHISTORY, line 7). At last, t_d is set to be t (line 8).

```
PEA TRIGGERING AND DAMPING PHASE
C: candidate set of as-paths that to be aggregated
ct: type of routing change
identity: local AS number plus IP address
1. Receiving a route r from the input at time t
2.   $d \leftarrow r.prefix, p \leftarrow r.as\text{-}path$, $\Delta t \leftarrow t - t_\mathrm{d}$
3.      #PEA TRIGGERING PHASE
4.      $P_d \leftarrow P_d \times e^{-\lambda\times\Delta t}$
5.      $ct \leftarrow$ DETERMINECHANGETYPE($r^{(i)}_d$, $r$)
6.      increase $P_d$ according to ct
7.      RENEWHISTORY ($H_d$, $\Delta t$, $p$)
8.      $t_\mathrm{d} \leftarrow t$
9.      if $P_d < Th_\mathrm{c}$ or r is 'WITHDRAWAL' then
10.        $r' \leftarrow r$
11.     else if $r^{(o)}_\mathrm{d} \in Repr(r)$ then
12.        $r' \leftarrow \emptyset$
13.     else
14.        #PEA DAMPING PHASE
15.        $C \leftarrow$ PATHSELECTION($H_d$)
16.        if $p \notin C$ then
17.           $r' \leftarrow r$
18.        else
19.           $r' \leftarrow$ PATHAGG (r, C)
20.        endif
21.     endif
22.  $r^{(i)}_d \leftarrow r$
23.  if $r' \neq r^{(o)}_d$ then
24.     $r^{(o)}_d \leftarrow r'$# propagate only route changes
25.  endif
```

```
PROCEDURE DETERMINECHANGETYPE ($r_{old}$, $r_{new}$)
ct: type of route change from $r_{old}$ to $r_{new}$
1.  if $r_{new}.as\text{-}path=\emptyset$ then
2.       $ct \leftarrow$'WITHDRAWAL'
3.  else if  $r_{old}.as\text{-}path=\emptyset$ then
4.       $ct \leftarrow$'PATHCHANGE'
5.  else if $r_{new}=r_{old}$ then
6.       $ct \leftarrow$'READVERTISEMENT'
7.  else if $r_\mathrm{new}.as\text{-}path \neq r_\mathrm{old}.as\text{-}path$ then
8.       $ct \leftarrow$'PATHCHANGE'
9.  else
10.       $ct \leftarrow$'OTHERCHANGE'
11.  endif
12.  return ct
```

```
PEA RELEASING PHASE
Q: family of PEA prefixes
t: the time relasing phase is scheduled
1. for d in Q
2.   $\Delta t \leftarrow t - t_d$
3.   $P_d \leftarrow P_d \times e^{-\lambda\times\Delta t}$
4.   if $P_d < Th_r$ then
5.      remove p, $\forall p \in H_d$
6.      if not $r^{(o)}_d = r^{(i)}_d$ then
7.         $r^{(o)}_d \leftarrow r^{(i)}_d$
8.      endif
9.   else
10.     for p in $H_d$
11.        $f_{d,p} \leftarrow f_{d,p} \times e^{-\lambda\times\Delta t}$
12.     endfor
13.  endif
14. $t_d \leftarrow t$
15. endfor
```

```
PROCEDURE PATHAGG(r, C)
C: the set of as-paths to be aggregated
1.  $r' \leftarrow r$
2.  $p \leftarrow$aggregated path from C
3.  $dis \leftarrow max\{len(x)|x \in C\}-len(p)$
4.  set r'.community according to dis
5.  set $g_{d,k}$ for each k in C # set flag bit
6.  $r'.as\text{-}path \leftarrow p$
7.  if $p \neq r.as\text{-}path$ then
8.       $r'.aggregator \leftarrow identity$
9.  endif
10.  return r'
```

```
PROCEDURE RENEWHISTORY($H_d$, $\Delta t$, $p$)
1.  for each path k in $H_\mathrm{d}$
2.    $f_{d,k} \leftarrow f_{d,k} \times e^{-\lambda\times\Delta t}$
3.  endfor
4.  if $p=\emptyset$ then
5.     return
6.  endif
7.  if $p \notin H_\mathrm{d}$ then
8.     insert p into $H_\mathrm{d}$
9.     $f_{d,p} \leftarrow 0$, $g_{d,p} \leftarrow false$
10. endif
11. $f_{d,p} \leftarrow f_{d,p}$ +1
```

Fig. 4. PEA algorithm details

PEA considers it necessary to enter damping phase if and only if the following three requirements are met, otherwise the original route r would be propagated.

- C_1: $P_d > T_{cutoff}$, since PEA handles only highly active prefixes.

- C_2: *r* is an BGP ANNOUNCEMENT,since PEA transparently propagates reachability.
- C_3: $r^{(o)}_d \notin Repr(r)$. That is, $r^{(o)}_d$ can not *represent* *r*, where *Repr*(*r*) denotes the set of routes that can represent route *r*.

The C_3 roots in the state based nature that BGP propagates only routing changes. For example, considering two consecutive routes regarding the same prefix, the latter one is necessary only when the previous one cannot *represent* it. Conventional BGP implies exact match logic that a route can represent another only when their route attributes are the same. PEA sticks to the exact match logic as BGP except for the *as-path* attribute. For details, in PEA a route *x* can represent another route *y* if and only if the following two conditions hold true: (1) *x.attr*=*y.attr* for any $attr \in \{prefix, origin, next\text{-}hop, med, atomic\_aggregate, local\text{-}pref\}$ and (2) *x.as-path* is an aggregation of *y.as-path*. Following this logic, an *as-path* does not have to change as long as the incoming as-path is one of its aggregation members. With the flag bit maintained for each as-path and each prefix in the auxiliary data structure, PEA can quickly tell whether *p* is the aggregation member of $r^{(o)}_d$*.as-path* or not.

3.2.2 PEA Damping Phase

Damping phase first determines the set of as-paths (denoted as *C*) that would be aggregated together (procedure PATHSELECTION, line 15). If as-path *p* is not covered by this set ($p \notin C$,line 16), as-path aggregation is unnecessary since the aggregation of *C* cannot represent *p*, and PEA propagates the original route in this case, otherwise the output route of PATHAGG (line 19) would be propagated.

Procedure PATHSELECTION plays a critically important role in BGP churn reduction. In theory, the percentage of reduced BGP updates is the probability of encountering an as-path change whose initial and ending as-path are covered by the as-path set determined by PATHSELECTION. To define a proper criterion for path selection, we had ever tried the following three polices, each with its own pros and cons.

- **Constant Top *k* Paths(CTP-*k*)**.

As the name implies, **CTP-*k*** selects the *k* paths with the highest frequencies. This is a typical MFU (Most Frequently Used) policy. Intuitively, a bigger *k* can improve the aggregation efficiency, at the cost of more path detail loss.

- **Self-Adaptive Top *k* Paths**(**SATP**-*k*).

Preliminary research [13] reveals that prefixes originated from edge ASes usually explore more as-paths in the pursuit of convergence than those originated from transit or core ASes.Therefore, prefixes that explored more as-paths should be associated with a bigger *k* value. Under this policy, PEA keeps monitoring for each prefix *d* the size of its history path set, i.e., $|H_d|$, and uses the mean value as the *k* of this prefix.

- **Maximum Path Similarity**(**MPS**).

Path similarity between as-path *a* and *b* is defined to be the number of AS segments in their longest common subsequence, divided by the number of unique AS appeared in *a* and *b*. This policy selects the set of as-paths that can maximize the overall path similarity, in order to preserve as much as path details.

To select a strategy most suitable for PEA, we conducted a comparison among the three polices. This comparison was based on BGP trace of *rv*2 in Dec 2010 observed from AS1239 (Tier1), AS1221 (Transit) and AS14608 (Stub). For the limit of space, we summarize only conclusions here. (1) The memory cost of buffering history as-paths and the reduction on BGP updates are proportional to the *k* in use under CTP-*k* policy, but the increase of this reduction becomes very limited when *k* is higher than 5. (2) SATP-*k* policy can achieve a performance as good as CTP-*5* while the incurred memory cost is between CTP-*3* and CTP-*4*. (3) The reduction on BGP updates under MPS policy is very limited, since MPS aggregates only two as-paths in the majority of cases. Finally, we adopted SATP-*k* policy in PATHSELECTION.

Procedure PATHAGG replaces the *as-path* attribute of input route *r* with path aggregated from candidate set *C*, which is determined in PATHSELECTION, and then tags the new route as a route resulted from aggregation by setting the *aggregator* attribute. Apart from the criterion of as-path aggregation defined in RFC4271 [5] (Section 9.2.2.1), PEA introduces some changes to assure the following properties.

Property 1: Non-loss of path(NLP),$\forall p \in C$ and $\forall a \in p \rightarrow a \in Z$, where Z is the aggregated path of as-paths in set C.

That is, all the ASes appeared in the member paths will appear in the aggregated path as well. This property is obvious if the basic aggregation method in RFC 4271 is fully respected. However, since RFC 4271 allows network operators to remove *AS_SET* segments formed in as-path aggregation if the loop-free property can be guaranteed by operators themselves, the NLP property may not always hold true in route aggregation.

Property 2: Longer aggregated path(LAP),$\forall p \in C \rightarrow len(Z) > len(p)$,where Z is the as-path aggregated from as-paths in set C. The symbol $len(X)$ denotes the length of as-path X in BGP decision process, where each *AS_SEQ* AS counts as 1 while an *AS_SET* as 1 no matter how many ASes are in the set.

This property can be fulfilled via *as-path prepending* that the deploying AS can prepend more than one instance of its own AS number in the aggregated as-path attribute before exporting a route to eBGP peers (BGP routers that are in a different AS than the deploying AS). Unfortunately, PEA cannot arbitrarily perform this operation since as-path prepending cannot be applied to iBGP peers (BGP routers that are in the same AS as the deploying AS), otherwise the iBGP routers would discard the received routes for the local AS number has already appeared in the as-path attributes. Instead of prepending the aggregated as-path immediately, PEA computes the number of AS number instances that should be prepended on the as-path attribute, and tags that route with a predefined community string. Then BGP routers within the deploying AS would perform the as-path prepending task on behalf of the PEA deploying router when this route traverses across AS boundary. Since the length of as-path is an important metric in route selection that the shorter, the more preferred, by prolonging the length of aggregated path, PEA purposely lowers the preference of unstable routes in downstream routers.

3.2.3 PEA Releasing Phase

For the event-driven nature of PEA, the auxiliary data structure relevant to a prefix would only be updated when there is a BGP update regarding this prefix. Considering a previously active prefix that has already stabilized, the outdated history information will result in low memory efficiency. Even worse, the majority of highly active prefixes are usually transient, lasting only a few days, while only a small number of them are persistently active over long period of time [4]. Thus to aggressively release resources relevant to dampened prefixes is necessary.

PEA RELEASING PHASE is scheduled as a background thread to clean the history of those prefixes whose prefix penalties have fallen beyond Th_r. During this process, de-aggregation may be triggered since the involved prefix is no longer highly active, and the original route would be propagated again (line 6-8). In this paper, this phase is invoked every 4 hours, and the caused router performance degradation is believed to be negligible for its low frequency.

3.3 Analysis of Reachability and Convergence Delay

The most significant point that makes PEA a different solution from RFD and MRAI is that PEA would neither cause prefix unreachable nor slow routing convergence. We make this conclusion by directly comparing the input and output route vector of PEA as follows.

Without loss of generality, we consider the routing regarding a particular prefix d. Let R_d (n is the size of R_d and $n\geq1$) denote the original route vector received from the input in a given period, among which each route r_i ($1\leq i\leq n$) is received at time t_i. Analogously, we define R'_d (with size m and $m\geq1$) and t'_i for the output route vector. Since BGP propagates only routing changes, both R_d and R'_d should meet the compactness requirement that for each $1\leq i\leq n$-1, $r_i\neq r_{i+1}$ (or for each $1\leq i\leq m$-1, $r'_i\neq r'_{i+1}$).

To ease our analysis, we firstly establish an one to one relationship between R_d and R''_d that $r''_i=f(r_i)$, where $r_i\in R_d$, $r''_i\in R''_d$, and $|R_d|=|R''_d|=n$. After that, we enforce the compactness rule on R''_d to obtain R'_d. For details, every two consecutive routes are compared and the latter one is preserved only when they are different.

Observed over a long time, the lifetime of r_i ranges from t_i to t_{i+1}, so does the r''_i. For any time range $[t_i, t_{i+1}](1\leq i\leq n\text{-}1)$, R_d and R''_d indicate the same reachability and converged state since r_i and r''_i have the same type(ANNOUNCEMENT or WITHDRAWAL), the same *next-hop* attribute, and are propagated at the same time($t_i= t''_i$). Meanwhile, compared R'_d with R''_d, only the duplicated occurrences are removed, thus R'_d implies the same reachability and converged state to R''_d. Finally, by creating this bridge R''_d between R_d and R'_d, we can conclude that R'_d causes neither extra reachability loss nor extra convergence delay to R_d.

4. Evaluation

PEA scales BGP by reducing the churn produced by highly active prefixes and amplified by path exploration. RFD [2] and MRAI [5] are the only two built-in countermeasures of BGP routers against route flap and path exploration respectively. To this end, we select RFD, RFD-HT (RFD with Higher Threshold) [10] and PED (Path Exploration Damping) [11] as the references in our evaluation. RFD-HT is the latest variation of RFD, and PED is shown by Huston et al. to perform better than MRAI in the reduction of churn due to path exploration [11]. For implementation details, RFD-HT raises the cutoff threshold to 12,000 or higher, in order not to penalize well-behaved prefixes. PED works in a quite different way that for each prefix learned over each BGP session, PED keeps comparing the received as-path with previous one to categorize the type of as-path changes, and a longer as-path is considered as the indicator of path exploration, and delayed at most a PEDI (PED interval) accordingly.

RFD, PED and PEA are implemented as standalone modules, in each of which an update is associated with a unique ID so that we are able to distinguish one from another even after the route manipulation of these mechanisms. By feeding these modules with the real BGP update streams observed from several monitor ASes, we simulated a deployment scenario that these monitor ASes had deployed these four mechanisms to protect their internal peers or customers from the churn caused by flapping prefixes.

4.1 Dataset and Parameter Setting

Our dataset included two ten-day BGP traces, ranging from 1st Dec, 2010 to 10th Dec, 2010, collected from *rrc*03 of RIPE-RIS [18] and *rv*2 of RouteViews [14] respectively. These two collectors were selected for their peers had provided a fair coverage of major transit ASes and Tier 1 ASes, which are believed to have more severe scalability issues. BGP traces were pre-processed to remove the updates due to BGP table transfers [15]. Then all the duplicated occurences were removed. After that, traces were split into several update streams according to the observation points, which were futher fed into PEA and reference mechanisms.

Table. 3 Parameter setting of RFD, RFD-HT and PEA

Parameter	RFD	RFD-HT	PEA
Withdrawal	1000	1000	0
Readvertisement	0	0	0
Attribute change	500	500	/
As-path change	/	/	1000
Other attributes change	/	/	0
Cutoff threshold	2000	12000	3000
Halftime (min)	15	15	30
Reuse threshold	750	750	750
Max suppress time (min)	60	60	/

Table. 4 Summary of evaluation results

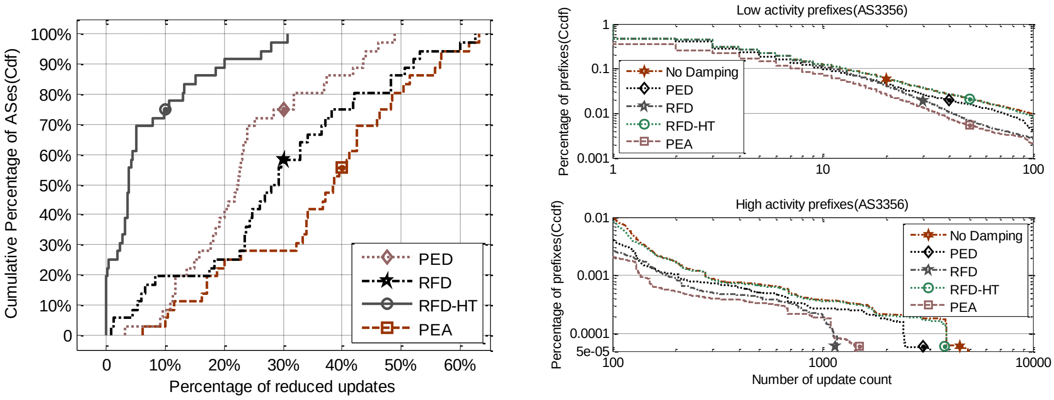
Metrics		RFD	RFD-HT	PED	PEA
Reduction of updates (Percentage)	Max	62.5%	30.7%	48.9%	63.1%
	Min	0.8%	0	3.1%	6.2%
	Avg	29.2%	7.1%	23.6%	36.2%
	Std	16.6%	8.4%	11.5%	15.6%
Convergence duration (*vs*. original one)	Max	4.93	1.01	1.03	0.99
	Min	0.62	0.86	0.81	0.47
	Avg	1.75	0.98	0.94	0.74
	Std	0.78	0.02	0.05	0.16
Caused Convergence delay (in seconds)	Max	530	39.1	23.1	-0.16
	Min	14.1	0	3.66	- 23.3
	Avg	213	5.22	13.1	-7.39
	Std	126	9.01	4.39	6.07

RFD and RFD-HT were configured with *Cisco* default parameters, as shown in **Table. 3**, except that RFD-HT raised its cutoff threshold value from 2000 to 12000. PEA inherited most of its parameters from RFD, but still made two changes. Firstly, it divided *attribute change* into two sub-classes: *as-path change* and *other attributes change*, and assigned different penalties to them. In particular, the penalty related to *other attributes change* was zero in our setting since the BGP updates that PEA can reduce is restricted to those caused by as-path changes. Secondly, the Halftime was prolonged to 30 min. PEA used a prolonged Halftime to trade heavier storage overload for better cache efficiency. At last, PED was configured with a PEDI 35 seconds.

4.2 On BGP Churn Reduction

To reduce BGP churn is one of the most straighfoward ways of improving BGP scalability. **Fig. 5(a)** shows the cumulative distribution function (Cdf) of the percentage of reduced BGP updates across all the monitor ASes. Generally speaking, PEA performs the best, followed by RFD and then PED, while RFD-HT performs the worst. As for the percentage of reduced updates, the averages are 36.2%, 29.2%, 23.6% and 7.1%, respectively. More details of the reduciton are shown in **Table. 4**. Both PEA and RFD can significantly reduce the BGP churn, but PEA has a much stabler performance. For example, the best performance of PEA and RFD are comparable, i.e., 63.1% in PEA *vs* 62.5% in RFD, but the worst performance of PEA is 6.2%, 6.75 times higher than that of RFD, 0.8%.

To know how these mechanisms work on prefix level, we conducted a case study with the data observed from AS 3356, which was selected for Tier1 ASes are believed to have more severe scalability issues. **Fig. 5(b)** shows the ccdf of the number of updates relevant to each prefix before and after damping. For comparison purpose, prefixes were categorized into two groups according to their activities. For details, prefixes regarding which fewer than 100 BGP updates had been observed were classified as *low activity prefixes*, and others were *high activity prefixes*. We selected 100 updates as the division line since the prefixes producing more than 100 updates happened to be the most active 1% prefixes (0.976% exactly) in our study period.



(a) **(b)**

Fig. 5. (a) Percentage of reduced BGP updates across *rv*2 and *rrc*03 monitors; **(b)**Prefix-level update count before and after damping observed from AS3356.

As shown in **Fig. 5(b)**, PEA leads the competition in reducing BGP updates of most of the prefixes until it is outperformed by RFD when the number of updates per prefix reaches 1,007 or higher (accounting for 0.0087% of Internet prefixes). In fact, the advantage of PEA against RFD keeps decreasing as the increase of prefix activity considering the log scale of y-axis. The reason is that once triggered, RFD will arbitrarily suppress all the subsequent updates relevant to the trigger prefix, and is much more efficient in reducing updates than PEA.

PED does not perform as well as PEA and RFD in general, but its performance is very stable and even better on some low activity prefixes than RFD. PED works in a different way from RFD. Instead of operating on highly active prefixes only, PED is designed to reduce the transient updates during path exploration, which is inherent to the path vector nature of BGP and universal to all the Internet prefixes. Thus its impact on BGP churn reduction is evenly distributed across the whole set of prefixes.

RFD-HT impacts only extremely active prefixes. For example, the curves of RFD-HT and No Damping in **Fig. 5(b)** do not separate from each other until the update count relevant to a prefix reached 2,000 or higher. Both RFD and RFD-HT act as a low pass filter, however, by raising its cutoff threshold value from 2,000 in RFD to 12,000, RFD-HT allows most of the updates relevant to low activity prefixes to transparently go through.

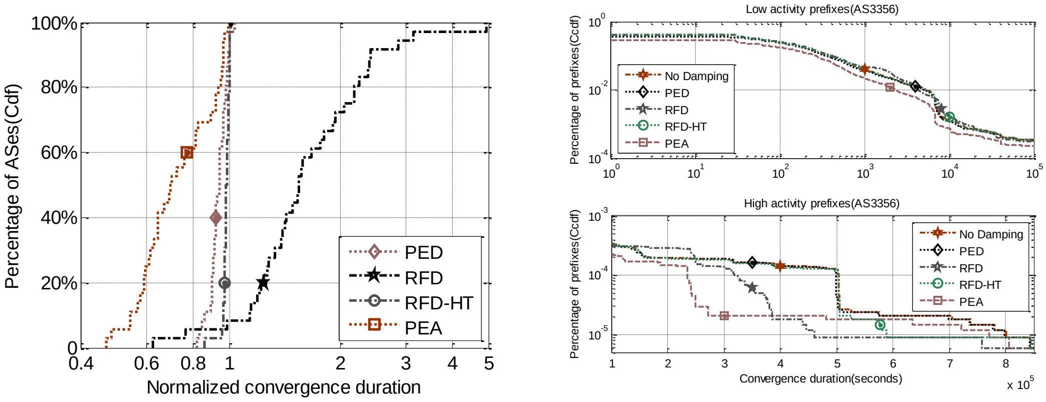
4.3 On Interference with BGP Convergence

Rather than the convergence property itself, we focus on the change to **convergence duration** and **delay** that an AS would experience when these four mechanisms are deployed. These two measures are similar to each other but still different. Given a routing event, convergence duration refers to the time distance between its first and last update, while convergence delay indicates the time period that it takes for BGP to converge to the last update after the root cause event happens. Intuitively, convergence duration and delay are positively correlated that short convergence duration means short convergence delay as well. Nevertheless, in practice a router is able to trade long convergence delay for short convergence duration, such as MRAI and PED. In the extreme case, for a routing event, a router can wait a long enough period for BGP to converge to the last update and propagate only that update (the convergence duration is zero in this case).

The interference with BGP convergence was evaluated on routing event basis, and we used a threshold T=5 min to split updates into events. Let $e_1e_2\ldots e_m$ (m≥1) be the sequence of original routing events regarding a particular prefix. Since every update within an event e_i is associated with a unique ID, we are able to track the transformed event of e_i, i.e., e'_i. Let $e_i.s$ and $e_i.t$ denote the start and the end of e_i, the event duration $l(e_i)$ can be that $l(e_i)= e_i.t- e_i.s$.

The definition of convergence delay is much more complicated. Given a routing event e_i, the caused convergence delay change by PEA and PED can be simply formalized as $h(e_i) = e'_i.t- e_i.t$. However, this definition only works for routing events that are converged. According to the parameter setting, two consecutive routing events regarding the same prefix are spaced by at least 5 minutes. Thus, *the precondition for BGP routing to convergence is that the introduced delay on the last update within each routing event should never exceed 5 minutes*.

While PEA and PED meet this requirement well, RFD and RFD-HT do not. For details, PEA does not delay or suppress any updates, and the last update within each event of PED would be delayed if any at most 35 seconds (a PED interval). RFD and RFD-HT cannot assure the routing convergence since some updates may be delayed as much as one hour [6]. In fact, according to *cisco* default parameter setting (shown in **Table. 3**), once a route is suppressed, it would take at least 21.25 minutes for this route to be reusable and released, only after which the routing converges [19]. In this case, a routing event does not converge until its last update is received, or the first update of subsequent routing event regarding the same prefix is received, where a new convergence process starts.



(a) **(b)**

Fig. 6. **(a)** Cdf of normalized convergence duration across *rv*2 and *rrc*03 monitors; **(b)** Prefix level convergence duration observed in AS3356 before and after damping

4.3.1 On Convergence Duration

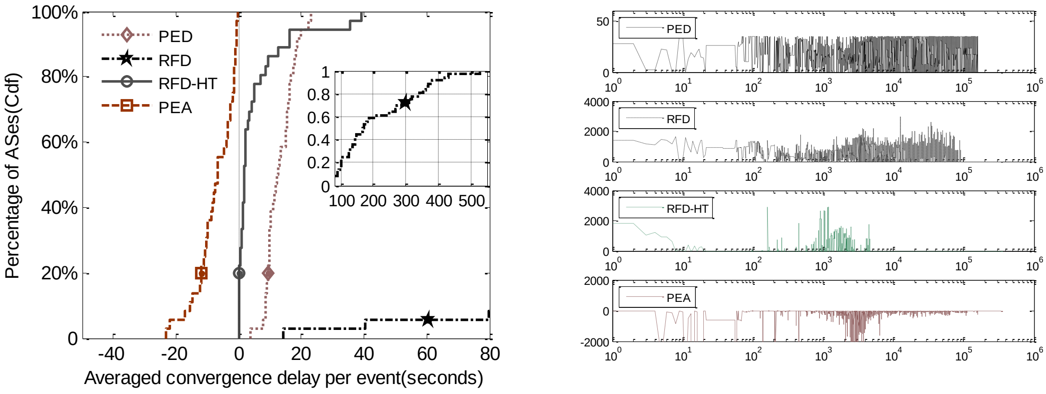
Since convergence durations vary across monitors, for comparison purpose, the convergence duration after applying any of these four mechanisms observed from a monitor is divided by the original convergence duration observed from the same monitor. **Fig. 6(a)** shows the cdf of normalized convergence durations across all *rv*2 and *rrc*03 monitors alive in our BGP traces. The line *x*=1 divides the graph into left and right part. The data points falling into the left part imply shorter durations than original ones, and others indicate longer durations accordingly.

Although all these four mechanisms are able to shorten BGP convergence duration, PEA is the only one that would monotonously shorten it. For instance, its entire curve is restricted to the left part since PEA does not delay updates under any circumstance. As for the best case, PEA can reduce as much as 53.3% of the total duration, followed by 37.7% in RFD, 19.2% in PED and 14% in RFD-HT. More evaluation details are shown in **Table. 4**, where we can see that on average RFD actually prolongs the convergence duration by 74.6%.

Still taking AS 3356 as example, **Fig. 6(b)** shows the distribution of convergence duration that each prefix had experienced before and after damping. Similarly, prefixes were classified into two classes with the same method aforementioned. This figure is very similar to **Fig. 5(b)**, for instance, PEA still leads the competition in reducing BGP convergence duration except for those extremely active prefixes (account for 0.0021% of all prefixes), where it is outperformed by RFD. However, the differences are still evident. At first, as opposed to the performance in reducing BGP churn, PED hardly shortens convergence duration that its curve always overlaps the No Damping curve. Secondly, despite the advantage of RFD against others on those highly active prefixes, it actually prolongs the BGP convergence process.

4.3.2 On Convergence Delay

For a routing event, to calculate its convergence delay requires not only the time when the last update is received, but also the moment that root cause event happens. Given the fact that root cause inference is hard if possible [12], we evaluated the change of convergence delay that an AS would experience relative to original routing when PEA is deployed.



(a) **(b)**

Fig. 7. (a) Averaged convergence delay change per routing event observed from different monitor ASes; **(b)** Prefix level convergence delay observed from AS 3356. For each prefix, its x-axis value is the rank of this prefix according to its activity, and the y-axis value is the averaged change of convergence delay that this prefix had experienced per routing event in our study period.

Fig.7(a) shows the Cdf of this change per routing event observed from each monitor AS. In general, PEA has the shortest delay, followed by RFD-HT and PED, and RFD has the longest delay. In fact, PEA can accelerate the convergence when the last update within a routing event is translated into the same format as previous update. Compared with original routing, the convergence delay change due to the deployment of PEA can be negative in some cases. To the opposite, other three mechanisms would more or less delay the convergence. For instance, the averages are -7.39 seconds in PEA, 5.22 seconds in RFD-HT, 13.15 seconds in PED and 212.69 seconds in RFD. Interested readers can refer to **Table. 4** for more results. What we want to emphasize is that *the convergence delay in RFD and RFD-HT usually means reachability loss in reality* since the suppressed routes would be excluded from route selection.

We then conducted a case study to examine the change of convergence delay relative to original routing that each prefix had experienced, as shown in **Fig. 7(b)**, where prefixes were sorted in descending order according to their activities, i.e., the number of updates. There is no doubt that PEA outperforms other mechanisms in accelerating convergence, since while other three mechanisms always cause positive delay, the delay due to PEA is negative.

More interestingly, **Fig. 7(b)** looks like a frequency graph, and shows clearly the part of prefixes that are affected by each of those four mechanisms, assuming that affected prefixes would experience a non-zero change of delay. Now we can see how these four mechanisms are different. PED affects the most number of prefixes (159,990), followed by PEA (158,500) and RFD (91,000), and RFD-HT affects the least (4,452). This observation can be explained that PED and PEA are triggered by path exploration, which are universal to all the prefixes, while RFD and RFD-HT by highly or extremely active prefixes. Notice that the amount of prefixes affected by RFD almost catches up with that by PED and PEA, its false positives should be treated seriously for the caused potential reachability losses.

4.4 On PEA Memory Cost

In addtion to the penalty value that RFD and RFD-HT maintain for each prefix, PEA incurs higher memory cost for the buffered history as-paths used to assist aggregation decision.The evalatuion of memory cost relevant to as-path buffering considered two situations, path sharing disabled and path sharing enabled, respectively. We used the as-paths buffered in PEA's prefix specific auxiliary data structure to approximate the memory cost when path sharing was disabled. These paths were further compared with local RIB (Route Information Base) [5] and only the unique ones were considered as the memory cost when path sharing was enabled.

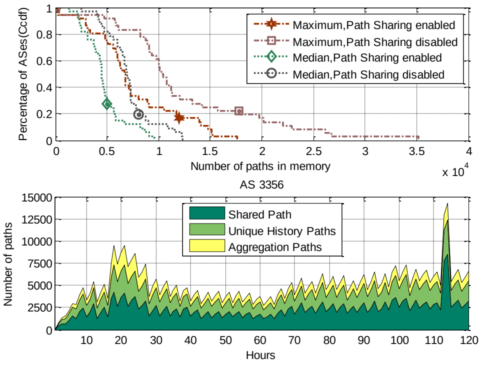


Fig. 8. PEA memory cost observed from all *rv*2/*rrc*03 monitors and AS 3356

In **Fig. 8**, we first show in the top the maximum and median of the number of as-paths that each monitor AS would have buffered when PEA is deployed. At first, we can find obvious burst nature of PEA memory cost by comparing the maximum with the median. For instance, when path sharing is disabled, the maximum can reach as high as 35,160 in the extreme case, 2.85 times higher than the median, 12,320, and this fraction is 1.83 (17,640/9,637) when path sharing is enabled. Secondly, path sharing can significantly reduce the storage overload due to the deployment of PEA. For details, path sharing can reduce the average of maximum from 12,406 to 7,554, accounting for only 16.5% of averaged RIB size (45,368 as-paths). Notice that, this is the upper bound of memory cost of PEA, which can be further reduced if path sharing is further extended to the as-paths stored in Adj-Rib-Ins [5]. During the experiment, the memory cost relevant to PEA auxiliary data structure is bounded by 30MB.

We selected AS3356 again to observe the evolution of PEA memory cost over time, as shown in the bottom of **Fig. 8**. We further divided the buffered as-paths by PEA into three parts, *shared paths with RIB*, *unique paths to PEA* and *aggregation paths*, which are resulted from as-path aggregation. On the average, the fractions of shared paths, unique paths and aggregation paths are 46.5%, 35.5% and 18.2%, respectively, highlighting the important role of path sharing. In addition, since PEA releasing phase is scheduled every four hours, these curves present an indented distribution where the cycle between two consecutive peaks is right four hours.

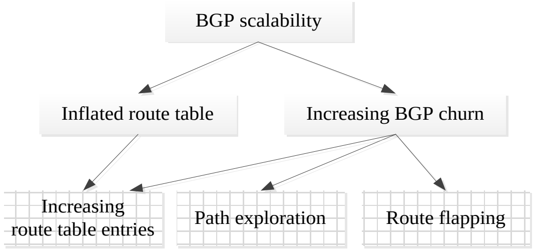


Fig. 9. A simple taxonomy of researches dedicated to improving BGP scalability

5. Related Work

Inter-domain routing scalability problem has drawn broad attentions from both industrial and academic community in recent years. As concluded at a workshop organized by Internet Architecture Board(IAB) that ,"routing scalability is the most important problem facing the Internet today" [20]. Routing scalability is a problem in two different aspects: inflated route table and increasing BGP churn. **Fig. 9** presents a rough taxonomy of the current researches dedicated to improving BGP scalability, and they are categorized into three different threads, targeting at increasing route table entries, BGP exploration and route flapping, respectively.

The first thread scales BGP by shrinking the size of routing information base (RIB) or forwarding information base (FIB). Typical proposals include LISP [21], which separates the address spaces of end systems from those used by routing system, and Virtual Aggregation [22], which organizes the regular IP space into virtual prefixes (VP), and uses tunnels to aggregate the sub-prefixes within each VP.

Both accelerating and rate limiting path exploration can reduce the exchanged updates during path exploration. The acceleration solutions include network based ones such as EPIC [23], and router-level ones such as Ghost Flush [24]. Network based solutions ask BGP routers to piggyback the root cause event onto relevant routing changes such that receiver routers can avoid the use of a route that would be affected by the same event. However, their deployments require remarkable upgrade of BGP routers, thus are hard to deploy. On the contrary, router level solutions are easy to deploy but the effects are still in debate. In addition to PED [11], MRAI [5] is another solution that reduces BGP churn by rate-limiting path exploration. MRAI allows a router to explore its alternatives for best route without exposing the intermediate step to its neighbors, thus reduces the number of updates during path exploration, at the cost of delaying convergence a few seconds, which may be unaffordable since restoration time after a failure below 50 milliseconds is a common requirement [25].

The only countermeasure working on the third thread is RFD, which was once considered as an important contributor to the overall routing stability [26]. Unfortunately, RFD was lately found to interact with path exploration and may cause severe reachability loss once triggered [6]. To this end, several modifications have been proposed. The latest proposals are two variations of RFD, RFD-HT [10] and RFD-RG [27]. RFD-HT trades damping efficiency for safety that it lowers the false positive ratio of labeling a well-behaved prefix as unstable one by raising the cutoff threshold value, but it would cause reachability loss as well once triggered. RFD-RG does not suppress a prefix unless its reachability can be assured via a different next-hop or a less specific prefix. However, a less specific prefix in current inter-domain routing does not necessarily mean reachability.

PEA is a solution simultaneously targeting at path exploration and route flap. In essence, PEA is a prediction based solution that it learns the pattern of as-path changes during path exploration, and then uses it to predict the upcoming as-path changes. Similar as RFD, PEA performs better on prefixes with higher activities since the continuous instabilities of them can provide PEA more samples to learn the pattern of as-path changes.

6. Conclusion

In this paper, we propose a router level mechanism, PEA, to scale current inter-domain routing system by reducing the BGP churn caused by route flap and amplified by path exploration. The most significant point that makes PEA different from other solutions is that PEA would neither cause prefix unreachable nor slow routing convergence. This characteristic further allows PEA to adopt more aggressive parameter setting, e.g., lower cutoff threshold, longer Half time, thus better performance in reducing BGP churn can be expected. PEA incurs higher memory cost for the buffered as-paths that used to assist the aggregation decision, but the cost is controllable, especially after the use of efficient path sharing. Experimental results show that PEA outperforms RFD, RFD-HT and PED in reducing BGP churn, shortening path exploration duration and accelerating BGP convergence.

References

[1] A. Elmokashfi, A. Kvalbein and C. Dovrolis, “On the scalability of BGP: the roles of topology growth and update rate-limiting,” in *Proc. of ACM CoNEXT'08*, pp. 1-12, December 10-12, 2008. Article(CrossRef Link).

[2] C. Villamizar, R. Chandra and R. Govindan,“BGP Route Flap Damping,” RFC 2439, November, 1998. http://www.ietf.org/rfc/rfc2439.txt

[3] J. Rexford, J. Wang, Z. Xiao and Y. Zhang, “BGP routing stability of popular destinations,” in *Proc. of 2nd ACM SIGCOMM Workshop on Internet Measurment*, pp.197-202, November 6-8, 2002. Article(CrossRef Link).

[4] R.V. Oliveira, R. Izhak-Ratzin, Z. Beichuan and Z. Lixia, “Measurement of highly active prefixes in BGP,” in *Proc. of IEEE Global Telecommunications Conference 2005*, pp. 894-898, November 28-December 2, 2005. Article(CrossRef Link).

[5] Y. Rekhter, T. Li and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” RFC 4271, January, 2006. http://www.ietf.org/rfc/rfc4271.txt

[6] Z.M. Mao, R. Govindan, G. Varghese and R.H. Katz, “Route flap damping exacerbates internet routing convergence,” in *Proc. of ACM SIGCOMM'02*, pp. 221-233, August 19-23, 2002. Article(CrossRef Link).

[7] A. Fabrikant, U. Syed and J. Rexford, “There's something about MRAI: Timing diversity can exponentially worsen BGP convergence,” in *Proc. of 30th IEEE INFOCOM*, pp. 2975-2983, April 10-15, 2011. Article(CrossRef Link).

[8] P. Smith and C. Panigl, “RIPE Routing Working Group Recommendations on Route-flap Damping,” RIPE Document, May 11, 2006. http://www.ripe.net/ripe/docs/ripe-378

[9] X. Wang, O. Bonaventure and P. Zhu, “Stabilizing BGP routing without harming convergence,” in *Proc. of 14th IEEE Global Internet Symposium*, pp. 840-845, April 10-15, 2011. Article(CrossRef Link).

[10] C. Pelsser, O. Maennel, P. Mohapatra, R. Bush and K. Patel, “Route Flap Damping Made Usable,” in *Proc. of Passive and Active Measurement*, pp. 143-152, March 20-22, 2011. Article(CrossRef Link).

[11] G. Huston, M. Rossi and G. Armitage, “A technique for reducing BGP update announcements through path exploration damping,” *IEEE Journal on Selected Areas in Communications*, vol 28, no 8, pp. 1271-1286, 2010. Article(CrossRef Link).

[12] A. Feldmann, O. Maennel, Z.M. Mao, A. Berger and B. Maggs, “Locating internet routing instabilities,” in *Proc. of ACM SIGCOMM'04*, pp. 205-218, Aug 30–Sept 3, 2004. Article(CrossRef Link).

[13] R. Oliveira, B. Zhang, D. Pei, R. Izhak-Ratzin and L. Zhang, “Quantifying path exploration in the internet,” in *Proc. of 6th ACM SIGCOMM Conf. on Internet Measurement*, pp. 269-282, October 25–27, 2006. Article(CrossRef Link).

[14] University of Oregon Route Views Project. http://www.routeviews.org/

[15] P. Cheng, X. Zhao, B. Zhang and L. Zhang, “Longitudinal study of BGP monitor session failures,” *Computer Communication Review*, vol 40, no 2, pp. 34-42, April, 2010. Article(CrossRef Link).

[16] J.H. Park, D. Jen, M. Lad, S. Amante, D. McPherson and L. Zhang, “Investigating occurrence of duplicate updates in BGP announcements,” in *Proc. of 11th Int. Conf. on Passive and Active Measurement*, pp.11-20, April 7-9, 2010. Article(CrossRef Link).

[17] W. Muhlbauer, A. Feldmann, O. Maennel, M. Roughan and S. Uhlig, “Building an AS-topology model that captures route diversity,” in *Proc. of ACM SIGCOMM'06*, pp. 195-206, September 11–15, 2006. Article(CrossRef Link).

[18] RIPE Routing Information Service(RIPE-RIS). http://www.ripe.net/data-tools/stats/ris/routing-information-service

[19] B. Zhang, D. Pei, D. Massey and L. Zhang, “Timer Interaction in Route Flap Damping,” in *Proc. of 25th IEEE Int. Conf. on Distributed Computing Systems*, pp. 393-403, June 06-10, 2005. Article(CrossRef Link).

[20] D. Meyer, L. Zhang and K. Fall, “Report from the IAB workshop on routing and addressing,” Internet Draft, April, 2007. http://tools.ietf.org/id/draft-iab-raws-report-02.txt

[21] D. Farinacci, V. Fuller, D. Meyer and D. Lewis, “Locator/ID Separation Protocol (LISP),” Internet Draft, May 4, 2012. http://tools.ietf.org/pdf/draft-ietf-lisp-23.txt

[22] P. Francis, X. Xu, H. Ballani, D. Jen, E. R. Raszuk and L. Zhang, “FIB Suppression with Virtual Aggregation,” Internet draft, December 30, 2011. http://tools.ietf.org/pdf/draft-ietf-grow-va-06.txt

[23] J. Chandrashekar, Z. Duan, Z.L. Zhang and J. Krasky, “Limiting path exploration in BGP,” in *Proc. of 24th IEEE INFOCOM*, pp.2337-2348, March 13-17, 2005. Article(CrossRef Link)

[24] Y. Afek, A. Bremler-Barr and S. Schwarz, “Improved BGP convergence via ghost flushing,”*IEEE Journal on Selected Areas in Communications*, vol 22, no 10, pp. 1933-1948, December, 2004. Article(CrossRef Link).

[25] O. Bonaventure, C. Filsfils and P. Francois, "Achieving sub-50 milliseconds recovery upon BGP peering link failures," *IEEE/ACM Transaction on Networking*, vol 15, no 5, pp. 1123-1135, October, 2007. Article(CrossRef Link).

[26] G. Huston, "Analyzing the Internet BGP Routing Table," *The Internet Protocol Journal*, vol 4, no 1, March, 2001.

[27] P. Cheng, J.H. Park, K. Patel and L. Zhang, "Route flap damping with assured reachability," in *Proc. of The 6th Asian Internet Engineering Conference*, pp. 24-31, November 15-17, 2010. Article(CrossRef Link).

Xiaoqiang Wang is currently a Phd student in School of Computer, National University of Defense Technology, China, from where he received his B.S. degree in Engineering of Computer Networks in 2006. He was a student visiting scholar at IP Networking Lab of University catholique de Louvain, Belgium, in 2010. His research interests include future Internet routing and routing security. He is a student member of IEEE.

Peidong Zhu is currently a professor in School of Computer, National University of Defense Technology(NUDT), China. He received his PhD degree in computer science from NUDT in 1999. During December 2008 and December 2009, he was the visiting professor at St Francis Xavier University, Canada. His research interests include network routing, network security and architecture design of the Internet and various wireless networks. He is a senior member of the IEEE.

Xicheng Lu received his B.S. degree in computer science from Harbin Engineering Institute, Harbin, China, in 1970. He was a visiting scholar at the University of Massachusetts from 1982 to 1984. He is currently a professor with School of Computer, National University of Defense Technology, China. His research interests include distributed computing, computer networks, and parallel computing. He is an academician of the Chinese Academy of Engineering and a member of the IEEE.

Kan Chen is now a Phd student in School of Computer, National University of Defense Technology, China, from where he received his B.S. and M.S. degree in computer science in 2007 and 2010 respectively. His research interests include social networking and security.

Huayang Cao is now a Phd student in School of Computer, National University of Defense Technology, China, from where he received his B.S. and M.S. degree in computer science in 2007 and 2010 respectively. His research interests include infrastructure networking and security.