Empirical analysis of core-edge separation by decomposing Internet topology graph

Yangyang Wang, Jun Bi, and Jianping Wu
Network Research Center
Department of Computer Science & Technology,
Tsinghua University, Beijing, China, 100084

***Abstract*— Border Gateway Protocol (BGP) is the de facto standard protocol for the inter-domain routing. Due to multi-homing and traffic engineering, the BGP routing table size of default free zone (DFZ) is growing rapidly. Inter-domain routing is facing the scaling challenge. Many solutions have been proposed. Among them, the core-edge separation scheme gets more attentions than others due to its practical advantages. It separates specific prefixes of edge networks from entering into transit core, and reduces the DFZ BGP routing table size. However, there has less evaluation on how much scalability can be improved from core-edge separation. In this paper, we take the further step to quantifying the impact of the core-edge separation on Internet inter-domain routing. We find that separation at stub-transit can reduce 43% routing table size and prevent more than half of BGP updates. We decompose the topology graph by k-core and customer-provider based decomposition methods, and analyze the impact of deploying separation at different level of topological hierarchy. We believe that complicated separation deployment strategies (not the simple stub-transit split) are feasible to approaching an optimal effect.**

***Index Terms*— BGP routing table, routing scalability, inter-domain routing, Internet routing**

I. Introduction

BGP Protocol is the de facto standard protocol used in the Internet inter-domain routing to connect all of the autonomous systems (ASes). In recent years, as the growth of Internet, the fast increase of BGP routing table size in the default free zone (DFZ) indicates a potential scaling challenge in the case of migrating today's Internet to IPv6 huge address space. The routing scaling issue has got more attentions from research community and the Internet Engineering Task Force (IETF).

A workshop report [1] of the Internet Architecture Board (IAB) summarized the Internet routing scaling challenge. Due to multi-homing and traffic engineering at the Internet edge, more specific prefixes are announced into transit core networks. And, Edge networks prefer Provider-Independent (PI) address space to Provider-Allocated (PA) address space for avoiding renumbering on changing upstream Internet Service Providers (ISPs). PI address cannot be aggregated in upstream providers, and then increase the number of prefixes of global routing table.

A number of solutions for routing scaling issue have been proposed. Some of them are clean-slate designs that need to change the addressing format, routing scheme and the whole routing architecture. However, most of the solutions focus on how to practically improve the Internet routing scalability in a short term future. Some of them, called core-edge separation scheme, get more attentions. The typical solutions of core-edge separation scheme include LISP (Locator Identifier Separation Protocol) [2], eFIT[3], Ivip[4], etc. Ballani et al. proposed a method that can be deployed in individual ISP network independently, and reduce the forwarding table size of the whole ISP network by prefixes Virtual Aggregation (VA)[5]. VA is considered as a promising and practical solution for an individual ISP, but not for the whole Internet. B. Zhang et al. [6] presented an evolutionary approach from the intra-domain VA applied to the individual ISP network to the emergence of inter-domain VA by merging the existing VA isolands, which will lead to the core-edge separation situation of the whole Internet finally. This approach indicates the practice of core-edge separation scheme.

The core-edge separation scheme separates the customer networks at the Internet edge from the provider networks at the transit core. Therefore, the non-aggregateable specific prefixes announced by the edge networks cannot enter into the transit core networks, and the global routing table size in the transit core will be reduced. However, in this situation, only the address space of transit core is globally routable, while the edge network address space cannot be seen in the global routing system and is not globally routable. When a host in an edge network communicates with a remote host in another edge network across the transit core, the source and destination IP addresses of edge networks need to be mapped to the corresponding transit IP addresses. As shown in figure 1, R1~R4 routers are termed as ingress/egress tunneling routers (ITR/ETRs) in some proposals[2], which separate edge networks ASX and ASY from transit core. When a packet that host A sends to host B reaches R1, R1 will look up mapping system, and encapsulate the packet in a IP-in-IP tunnel using

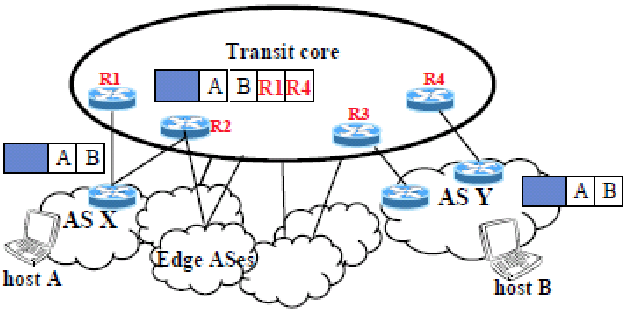

Fig. 1. Core-edge separation scheme

This work was supported in part by Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant 200800030034, and in part by the National Basic Research Program ("973" Program) of China under Grant 2009CB320501.

978-1-4244-5638-3/10/$26.00 ©2010 IEEE

the R1 and R4 globally routable IP addresses as the tunneling source and destination addresses. After receiving the packet, R4 will de-encapsulate the tunnel and forward the original packet to host B. During this process, mapping and tunneling are two important steps. Therefore, core-edge separation is also called Map-Encap scheme.

The gains of core-edge separation include reducing routing table size, and no need to change host protocol stacks and applications. But, it also introduces many costs. It relies on an extra mapping system and bilateral deployment of ITR/ETR devices. It also causes the map-encap delay during packets delivery. More discussions about the advantages and challenges of core-edge separation are described in [7].

Although core-edge separation doesn't need host change, its wide deployment will bring a large change to the Internet routing architecture at the side of transit ISPs. Before that, understanding and evaluating the possible effect of these core-edge separation solutions on actual Internet routing is important for their improvement, deployment and other new mechanism and architecture design.

In the past years, there have been some study on the evaluation of the LISP solution, including the performance and cost of ID/Locator mapping [8] [9], and the basic impact on current routing of Internet [10]. Different from related work, we will analysis the core-edge separation scheme, not only on the LISP solution. This paper takes the further step to quantifying the impact of the core-edge separation scheme on Internet inter-domain routing.

Our contributions include the following folds:

1) We give a primary analysis for the impact of separating stub ASes from transit ASes on routing system by examining the growth of transit networks and reduction in routing tables and BGP churn.

2) We use three methods to decompose the AS-level topological hierarchies of Internet, and estimate the impact of other possible deployment position on the effect of core-edge separation. We show that prefixes mainly gather at lower tiers and top tiers, there is a wide range of middle belt with less prefixes distributed, where core-edge separation deployment will improve the reduction of routing table size to more than 80%. In this belt, moving separation line towards top tier doesn't obtain visible benefits until into the top tier.

3) We estimate the possible mapping table size and deployment areas. Deploying separation at stub-transit level will involve a wider area of transit ASes. And, the mapping size will grow faster than routing tables.

II. Methodology

This section discusses the key points to be analyzed, and the dataset used in analysis experiments.

A. Key points for analysis

The key points that are relevant to the impact of core-edge separation is focused on in our analysis. The basic point to be examined is routing table size reduction. It indicates how many prefixes can be prevented from entering into the global routing table. The network reachability information for blocked prefixes is stored in the mapping tables of mapping system. In other words, the reduction of global routing table growth is transforred to the mapping table growth. Therefore, the growth of mapping information in mapping table is also an important metrics to be considered. The core-edge separation also prevents the routing dynamics of the edge prefixes from propagating into the transit core, thus we will estimate this benefit via statistics on update messages.

The placement of separation points (i.e., ITR/ETRs) impacts the effectiveness of core-edge separation. There is no specific borderline for the core-edge separation in the Internet. Deploying ITR/ETRs at the place close to tier-1 networks could improve the reduction of the area of transit core and its routing table size. In our primary evaluation, we choose the place between stub ASes and transit ASes. Moreover, we examine the impact of other deployment position by making fine-granularity decomposition on Internet topological hierarchy.

B. Dataset

We collect BGP routing tables and updates data from Routeviews [11] and RIPE NCC [12].Four dataset are used in our analysis. One is the routing tables collected from the collector route-views2 of Routeviews from Jan. 2004 to Dec. 2009. We named this dataset as DHIS. To get a more comprehensive view of Internet topology, we collect a total of 19 routing tables of the same day 2009/12/15 from six collectors of Routeviews and 13 collector of RIPE NCC. This dataset is denoted as DDEC. The third dataset is eight months of routing updates from the "route-views2.oregon-ix.net" collector of Routeviews from June to December in 2009. It is used to analyze the impact on routing dynamics. We also collect Internet topology data with annotated AS relationships of the day 2009-12-15 from UCLA [13], which we named DUCLA.

III. Decomposing Internet Topology

In this section, we decompose the Internet AS-level topology in three different ways to analyze the impact of different core-edge separation position. In each decomposition way, we analyze ASes distribution among these hierarchies, which give some reference for the adoption of core-edge separation position.

A. Basic decomposition: Stub ASes and Transit ASes

The first and straight hierarchical layers for the domains of Internet are transit and stub. A stub AS only carries traffic to or from its networks. A transit AS usually conveys traffic between other ASes. Therefore, stub ASes only appear at the end of AS paths of BGP routing table or update messages, while transit ASes appear between two ASes in an AS path. The idea of core-edge separation basically arises from the fact that there are numerous "edge" stub ASes around a small "core" of transit ASes. The position between stub ASes and their transit ASes is the primary separation line. Some proposals, such as LISP, are mainly deployed at this position.

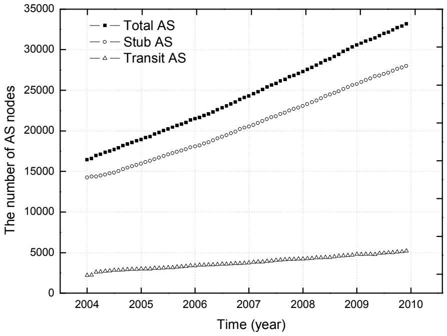

Fig. 2. The growth of AS nodes

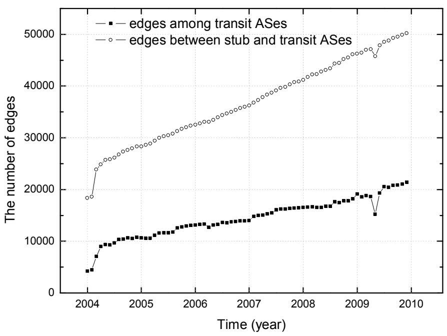

Fig. 3. Growth of edges of S(stub) and S(transit)

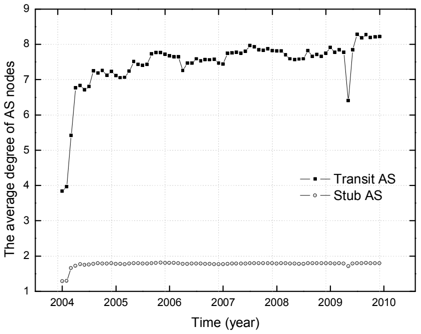

Fig.4.Growth of connectivity of S(stub) and S(transit).

Although there is a large number of stub ASes, and the physical links connected to transit ASes are changed frequently, we should not ignore the growth happened in the transit ASes that seem to like a "stable" transit core of the Internet. We first examine the growth of the number of ASes. As shown in Figure 2, the total number of ASes is more than 33,000 in Dec. 2009, among which more than 84% are stub ASes (shown in Figure 8). The total number of ASes and transit ASes are doubled over the past six years from 2004 to 2009. The transit ASes grow slowly, and the most contribution of growth of total ASes is from stub ASes.

Then, we examine the edges growth of transit ASes and stub ASes. Here, we cut the Internet topology graph into two parts by isolating the stub ASes from transit ASes. The part of stub ASes is a set including stub ASes nodes and the edges connecting stub ASes with transit ASes. We denote this part as S(stub). Another part has *only* transit AS nodes and the edges among them, denoted by S(transit). Figure 3 shows that the edges of both S(transit) and S(stub) are also doubled over the past six years. Now, we measure the edge density by average degree to reflect the density of connectivity of each part.

For the S(stub), the average degree is the following:

$$\frac{\text{The number of edges in S(stub)}}{\text{The number of AS nodes in S(stub)}}$$

It is right the average degree of stub ASes connected to the upstream transit ASes. For the S(transit), the average degree is computed as the following:

$$\frac{\text{The number of edges in S(transit)} \ * 2}{\text{The number of AS nodes in S(transit)}}$$

It is different from the actual average degree of transit ASes in the complete Internet topology, because the S(transit) does not include the stub-transit edges, it only has the transit-transit edges. From Figure 4, we can see that the average degree of S(transit) grows faster than that of S(stub). The average degree of S(transit) is about 7~8 with a continual growth trend, while the stub ASes keep a flat value of less 2. This means that these transit ASes tend to be a flat world with more and more dense connectivity, however, multihoming connections of stub ASes don't increase observably against the number of stub ASes.

There is no specific borderline for the core-edge separation in the Internet. Separation points are not only be deployed between stub and transit Ases. Moving separation points upwards to be closer to tier-2 or tier-1 networks can lead to a smaller transit "core" and a wider range of "edge", and improve the reduction of the area of the transit core and its routing table size. To examine the effect of further separation close to Internet core, we analyze the topological hierarchy by other decomposing ways in the following paragraphs.

B. K-core decomposition

K-core decomposition, or named with K-shell decomposition [14], is one way used to rank the node and study the hierarchies of Internet graphs. For a graph G, after removing all nodes with degree less than k repeatedly, the remained unique subgraph is defined as a k-core. Each of the nodes in k-core has a degree large than value k. And, the node that belongs to k-core, but be removed in (k+1)-core are assigned a shell index (or said coreness) of value k. All of the nodes of shell index k form the k-shell. We use this way to decompose the Internet AS graph into k-shells. The distribution of AS nodes cross these shells is shown in figure 5.

C. Customer-provider relationship based decomposition

There are business relationships between neighboring AS in the Internet [15]. These relationships mainly include customer-provider (c2p), peer-peer (p2p), and sibling-sibling (s2s) relationships. They are agreements about traffic flow and reflected by BGP policies. Since s2s relationship is rare and its inference is incomplete and incorrect, they are ignored in our data source and analysis. In the customer-provider relationship, provider networks sell Internet connectivity to their customers, and customers get to the rest of Internet through the transit service provided by provider networks. Thus, the Internet hierarchy is mostly determined by the customer-provider relationships. Provider ASes will advertise the routes learned from customers to other neighbor ASes, as well as advertise the customers of the routes and prefixes from the outside Internet. In this way, the prefixes in the stub ASes can be propagated into tier-1 ISP networks along the customer-provider hierarchy.

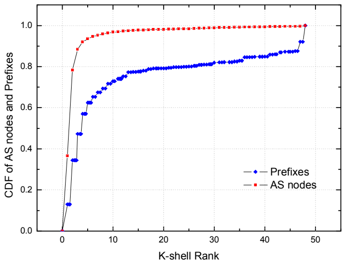

Fig. 5. CDF of ASes and prefixes K-core decomposition

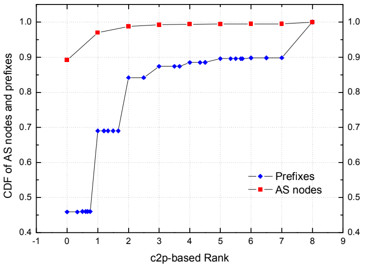

Fig. 6. CDF of ASes and prefixes in c2p-based decomposition

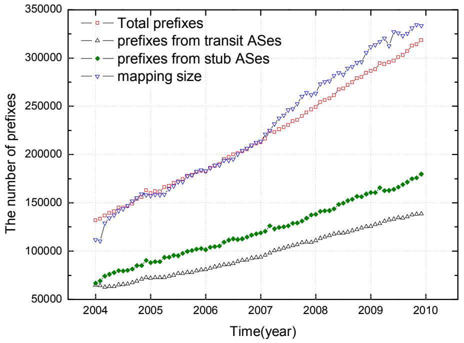

Fig. 7. The growth of prefixes and mapping size

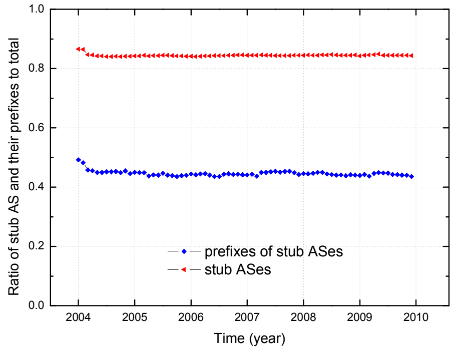

Fig. 8. Proportion of stub ASes and their prefixes in the all Internet

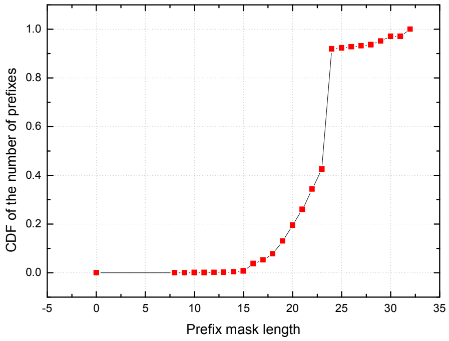

Fig. 9. CDF of prefixes on different mask length

We can decompose Internet AS-level topology graph in terms of the customer-provider hierarchical structure. We use G_{c2p} to denote the graph that is composed of all the edges of c2p relationship, and the associated nodes of each edge. It is a subgraph of the complete Internet topology. The nodes that only have peer-peer edges with neighbors don't appear in this subgraph. Each edge in graph G_{c2p} is a directed edge from a customer node to a provider node. Then, we remove the leave nodes that have empty in-degree, and assign them level-0. Repeat this process, we will get level-1, level-2, …, until there is no nodes with empty in-degree. The remaining nodes and edges form the top level. In this way, we will obtain a sequence of level from 0 to the top. The levels with larger value will be closer to the tier-1 AS nodes. We use the dataset DUCLA to do c2p-based decomposition. The result and distribution of AS nodes cross these levels is shown in figure 6.

IV. Prefix Distribution

In this section, we will discuss the prefixes distribution according to the three decomposition schemes above. We first observe the prefix distribution in terms of prefix mask length using dataset DDEC. Figure 9 shows that more than 96% prefixes have a mask length larger than 16. And prefixes of length 24 contribute 50% to all prefixes. Less 10% prefixes have a length more than 24. Subsection B and C will analyze their distribution on the hierarchies of K-core and c2p-based decomposition respectively.

A. Stub ASes and Transit ASes

This is the primary decomposition, and some solutions (e.g., LISP) are mainly deployed in terms of this separation. We make a statistics for the prefixes originated from stub ASes and find that stub ASes (about 84% of total ASes) have announced about 43% prefixes, as shown in Figure 8. This proportion has little fluctuation over six years from 2004 to 2009. It shows that blocking the prefixes from the large amount of stub ASes does not imply a significant reduction in routing table size.

B. K-core and c2p-based decomposition

Figure 5 and 6 are the cumulative distribution function (CDF) of prefixes and ASes in different decomposition scheme. In Figure 6, more than 90% ASes are at the levels less rank 5, and originate about 60% prefixes. In Figure 7, more than 95% ASes are at the level-0 and level-1, and originate 70% prefixes.

From figure 5 and 6, we also can see that, to prevent 90% of all prefixes, we need to deploy separation point at levels larger than 45 in k-core ranking and levels larger than 7 in c2p-based ranking. It seems to have *a middle belt* connecting a small "core" with "outer" areas. This belt is from level-10 to level-45 in k-core, and from level-3 to level-7 in c2p-based ranking, where less prefixes are distributed.

To investigate the usage conventions of prefixes, we divide all the prefixes into two parts: one includes the prefixes with length larger than and equal to 16, and the rest is another part. We denote the former as PFXL16, and the latter as PFXS16. Figure 10 and 11 shows that more than 60% prefixes in both of the two parts are distributed at the levels closer to the bottom or the top, although there are more prefixes (about 30%) gathering in the area of top tier in both decomposition schemes for PFXS16. To verify whether prefixes of shorter length tend to be used in higher and top tier ASes or not, we further compute the distribution of prefixes of length less than 9. The result shows that more than 50% prefixes distribute in lower tiers (e.g. lower than 3 in c2p-based rank and less than 20 in k-shell rank). It means that most prefixes can be blocked at the lower tiers even though they are shorter prefixes.

V. Deployment Positions and Mapping entries

In this section, we will examine how many places are involved to deploy the separation points (i.e., tunnel routers, TRs). This represents a deployment cost. We could not get the concrete number of tunnel routers, but we try to estimate the amount of ASes that will need to deploy separation point. We make the estimation based on the dataset DDEC, and apply to the stub-transit separation scheme. If deploy TRs only in stub ASes, then more than 84% ASes need to deploy new routers from previous statistics about stub ASes. If deploy TRs at the transit ASes side that connect with stub ASes directly, it doesn't imply that only a small number of ASes at lower tiers are involved. We find that about 4804 ASes (near to 90%) are involved among these 5363 transit ASes. And these ASes are distributed over all of hierarchies in k-core and c2p-based decomposition. But more than 80%~90% in the lower tiers, as shown in figure 12 and 13.

We also roughly estimate the growth of information of mapping table. Here, we assume that each entry is an ordered tuple from one prefixes to one separation point (e.g., an interface IP address of one ITR/ETR) in the mapping table. Mapping size is measured by the number of mapping entries, which grows with the number of both prefixes and separation point. In the stub-transit separation case, we assume that separation is placed at transit ASes. For one stub AS, its different upstream transit AS node represents a different

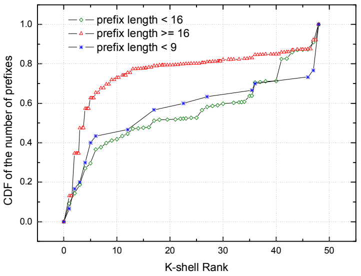

Fig. 10. CDF of prefixes of different lengths across k-shell ranks

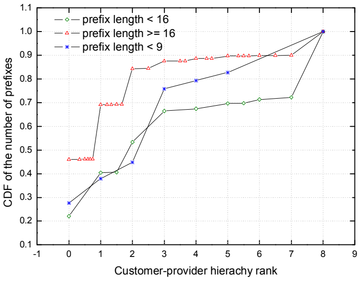

Fig. 11. CDF of prefixes of different lengths across c2p-based ranks

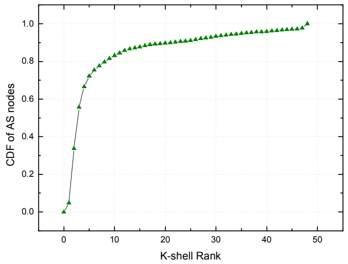

Fig. 12. CDF of Transit ASes that deploy separation points. (k-shell rank)

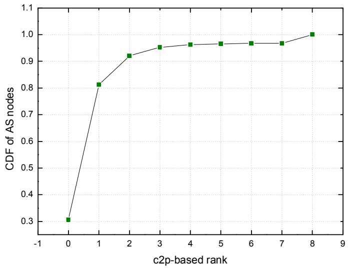

Fig. 13. CDF of Transit ASes that deploy separation points. (c2p-based rank)

separation point. Thus, the mapping size can be estimated by the formula:

$$\sum_{i\in S_{stub}} p(i) * od(i)$$

Where S_{stub} represents the set of all stub ASes, and *i* is each one of them. *P(i)* represents the number of prefixes originated from stub AS *i* whose out-dgree is *od(i)*. This formula will underestimate the cases that one stub AS has multiple different separation point at the same transit AS. The result in Figure 7 shows that mapping size is nearly the same as the global routing table size. After the year 2007, the mapping size grows faster than the routing table size. This is mainly caused by the increasing multi-homing of stub ASes.

VI. Analysis for Routing Dynamics

In this section, we analyze the impact of stub-transit separation on routing dynamics. Figure 4 shows the BGP churn for six months from Jun. to Dec. 2009. The blue triangles represent the BGP churns of stub ASes. It can be seen that much more than half of BGP churns arise from stub ASes and can be blocked entering into transit core. The total announcement bursts present a correlation with the bursts from stub ASes. It has more obvious correlation in level shifts between total withdraws and stub AS withdraws from 2009-8 to 2009-10. These correlations indicate that it is highly possible that the causes leading to large BGP churn bursts are mainly from stub ASes.

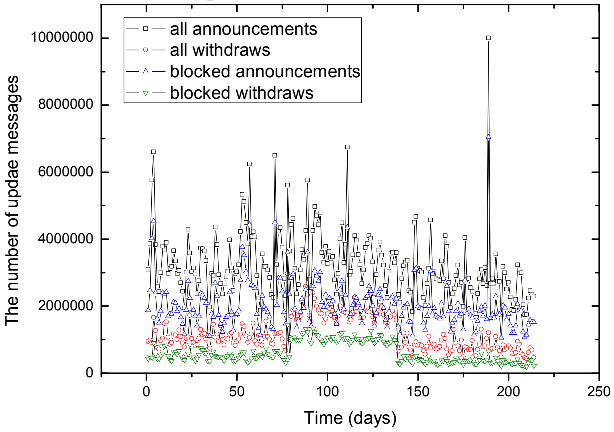

Fig. 14. The impact of stub-transit separation on BGP churn

VII. Conclusion

We use real Internet routing data to analyze the impact of core-edge separation on improving the Internet routing scalability. Firstly, we examine the possible effect of stub-transit separation, and the results show that separating all of stub ASes (84%) can block 43% prefixes into transit ASes. The reduction in routing table size is not significant, and it has a high deployment cost that near to 80% transit ASes are involved to deploy separation devices (i.e., tunnel routers). But it can lead to much more than half of reduction of BGP churns in transit ASes. We make a further analysis to Internet hierarchies by k-core and customer-provider-based topological structure decomposition, and examine the distribution of ASes and prefixes. We find that there is a middle belt in topological structure which connects lower tier ASes and top tier ASes. Moving core-edge separation points upwards to the middle belt will get a great effect, and more than 80-90% prefixes can be blocked at the lower tiers. It can also reduce the area of deployment. However, moving separation points upwards will lead to increase the number of blocked prefixes and separation points which one prefix can be mapped to, thus increase the size of mapping between blocked prefixes and separation points. Therefore, complicated separation strategies (not the simple stub-transit split) are needed to approach an optimal effect.

References

[1] Report from the IAB Workshop on Routing and Addressing",RFC4984

[2] D. Farinacci, V. Fuller, D. Meyer, and D. Levis. Locator/ID Separation Protocol (LISP). draft-farinacci-lisp-12, Sep. 2009

[3] The eFIT project, http://netlab.cs.memphis.edu/efit/index.php

[4] Ivip, http://www.firstpr.com.au/ip/ivip/

[5] Hitesh Ballani, Paul Francis, Tuan Cao and Jia Wang. "Making Routers Last Longer with ViAggre".USENIX NSDI 2009, Boston, MA, April 2009.

[6] B. Zhang, L. Zhang, Evolution Towards Global Routing Scalability, http://tools.ietf.org/html/draft-zhang-evolution-00

[7] D. Jen, M. Meisel, H. Yan, D. Massey, L. Wang, B. Zhang, L. Zhang, Towards A New Internet Routing Architecture: Arguments for Separating Edges from Transit Core, in: Proc. HotNets-VII, 2008.

[8] Luigi Iannone and Olivier Bonaventure, "On the cost of caching locator/ID mappings", ACM CoNEXT 2007

[9] Hong Zhang, Maoke Chen, Yuncheng Zhu: Evaluating the Performance on ID/Loc Mapping. GLOBECOM 2008: 2276-2280

[10] Ping Dong, Hongchao Wang, Yajuan Qin, Hongke Zhang, Sy-Yen Kuo, "Evaluation of Scalable Routing Architecture Based on Locator/Identifier Separation", GLOBECOM Workshops, 2009:1-6

[11] Routeviews project: http://www.routeviews.org/

[12] RIPE NCC: http://www.ripe.net/ris/

[13] UCLA topology collection project: http://irl.cs.ucla.edu/topology/

[14] Alvarez-Hamelin JI, Dall'Asta L, Barrat A, Vespignani A. K-core decomposition of Internet graphs. (2007) e-Print archive, http://arxiv.org/abs/cs.NI/0511007.

[15] Lixin Gao: On inferring autonomous system relationships in the internet . IEEE/ACM Trans. Netw. (TON) 9(6):733-745 (2001)

978-1-4244-5638-3/10/$26.00 ©2010 IEEE