Implications of Interdomain Traffic Characteristics on Traffic Engineering

STEVE UHLIG AND OLIVIER BONAVENTURE
Infonet group, University of Namur, Belgium
{*suhlig,obonaventure*}*@info.fundp.ac.be*

Abstract. We study the interdomain traffic as seen by a non-transit ISP, based on a six days trace covering all the interdomain links of this ISP. Our analysis considers the relationships between the interdomain traffic and the interdomain topology. We first discuss the day-to-day stability of the interdomain traffic matrix to evaluate the feasibility of interdomain traffic engineering. Then, we study the variability of the interdomain flows for several aggregation levels (prefix, AS and sink tree) and with respect to the interdomain topology seen by BGP. We show that despite the important variability of interdomain flows, it would be useful for a non-transit ISP to traffic engineer its access traffic by relying on a sink tree aggregation level.

1 INTRODUCTION

Traffic engineering has received a lot of attention during the last few years [3, 24]. Initially, traffic engineering was considered as a solution to allow large tier-1 ISPs to optimize the utilization of their network. In these large networks, there are typically several possible paths to reach a given destination or border router. Ideally, to achieve a low network utilization, the traffic should be spread evenly among all the available links. Unfortunately, this does not correspond to the way traditional IP routing protocols behave. In most cases, the IP routing protocol is not aware of the load on the various parts of the network and selects for each destination the shortest path based on static metrics such as the hop count or the delay. This destination based routing creates an uneven distribution of the traffic that may lead to periods of congestion inside the ISP backbone. Several techniques have been proposed to better spread the load throughout the entire network [3]. A first solution is to select appropriate link metrics based on a known traffic matrix [12, 23]. This solution can provide some interesting results if the traffic matrix is known and stable. A second solution is to rely on a connection-oriented layer-2 technology [4] such as ATM, MPLS or one of the emerging optical technologies. In this case, layer-2 connections can be established statically [9] or dynamically between distant routers and the layout of these connections can be optimized to achieve an even distribution of the traffic inside the network [3]. It is also possible to dynamically create new layer-2 connections in order to quickly respond to link failures or changes in the traffic pattern [3].

Unlike large Tier-1 ISPs, small ISPs and multi-homed corporate networks have very different traffic engineering requirements. Their networks usually consist of a simple topology and are frequently over-provisioned. The traffic engineering solutions mentioned above are not really useful in such networks. For these networks, the costly resource that needs to be optimized with traffic engineering is their interdomain connectivity. Until now, few work has addressed the interdomain aspects of traffic engineering.

In this paper, we present a detailed analysis of the interdomain traffic from a medium ISP and evaluate the feasibility of interdomain traffic engineering. The paper is organized as follows. In Section 2, we briefly explain the existing mechanisms to control the flow of interdomain traffic. In Section 3, we describe the ISP where we gathered our measurements. In Section 4, we study the topological distribution of interdomain traffic. In Section 5, we evaluate in details the temporal variability of the traffic.

2 INTERDOMAIN TRAFFIC ENGINEERING

Medium ISPs are usually multi-homed and to optimize the utilization of their interdomain connections, the only tool that they can rely on are the establishment of new physical links and tweaking the configuration of their BGP routers. The first solution does not allow an ISP to react quickly since it usually requires a few months.

By tweaking the configuration of its BGP routers, the

ISP can in some ways control the utilization of its interdomain links. The BGP routing protocol [18] used to distribute the interdomain routes throughout the Internet allows each ISP to define its own policies. These policies specify how routes received from a BGP peer will be accepted, selected and redistributed towards other BGP peers. Different policies can be applied by each ISP to influence its incoming and its outgoing traffic.

When considering the outgoing traffic, a BGP router will often select as the best route towards an external destination the route received with the smallest AS path length. However, if required, the ISP can easily bypass this selection by configuring its border routers to insert the LOCAL-PREF attribute in the routes redistributed by iBGP[18]. For example, in figure 1, if AS20 wants to force the packets towards AS10 to be sent through AS21, it can configure its BGP router peering with AS21 to attach a large LOCAL-PREF value to the AS10 routes received on this link while other routers will insert a default LOCAL-PREF value.

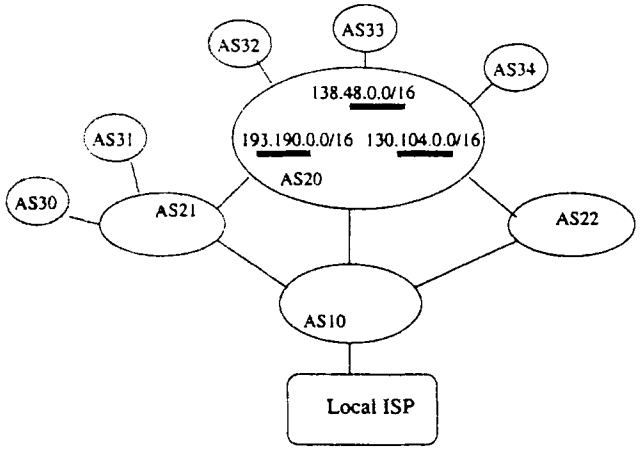

Figure 1: Simple interdomain topology.

To optimize its incoming traffic, the ISP will proceed differently. In this case, it needs to influence the redistribution and the selection of its own routes by remote ISPs. Since the default configuration of many BGP routers is to select the route with the smallest AS path length, a common technique is to artificially increase the length of the AS path for the routes with the lowest preference. For example, in Figure 1, if AS20 wanted to indicate that it prefers to receive its traffic towards subnet `138.48.0.0/16` through its link with AS22, then it would announce this prefix as usual on this link and announce it with a `AS20:AS20:AS20:AS20` path to AS21 and AS10. If AS10 and AS21 rely solely on the AS path length to select the best BGP route, they will prefer the shorter route received through AS22. This requires a manual configuration of the BGP routers, but the configuration burden can be reduced by using the BGP community attribute [18]. Recently, several large ISPs have gone one step further by defining BGP community values that allow their customers to influence the redistribution of their routes. For example, in Figure 1, AS20 could configure its BGP routers to always prepend four times AS20 when they announce to external peers a route received from its customers with a special community value. This implies that AS20 informs its customers of these communities.

These BGP techniques are in use in today's Internet. [19] reports a percentage of the unique prepended AS Paths a little over 50 % of the total number of unique AS Paths (prepended and not prepended). Our BGP routing table confirms this trend. However, it should be noted that these techniques rely on a manual configuration of the border routers and are difficult to automate. Furthermore, to finely tune their interdomain traffic, some ISPs need to divide their IP address space into several distinct prefixes that are announced separately. This utilization of BGP for traffic engineering coupled with the growth of multi-homing tends to increase the size of the BGP routing table [13] which is not a desired feature. Furthermore, every time an ISP wants to change its interdomain traffic engineering policies, it must withdraw and readvertise routes. This increases the instability of the BGP routing table and is even less desirable.

We believe that specific interdomain traffic engineering solutions should be developed by taking into account the specific requirements of this type of traffic. Until know, few works have addressed this problem. In [1], a set of BGP attributes were defined to allow an ISP to indicate its traffic engineering preferences when announcing routes. This proposal is mainly a generalization of the existing BGP MED attribute whose scope is limited to direct peers [18].

A more interesting proposal is the BGRP protocol described in [16]. The objective of the BGRP protocol is to provide QoS reservations across the entire Internet in a scalable manner. For this, BGRP operates at the interdomain level and allows an AS to reserve resources along sink trees. For example, on Figure 1, the BGRP sink tree from AS20 would aggregate the reservations for the IP packets originated by AS20, AS32, AS33 and AS34.

3 MEASUREMENT ENVIRONMENT

To better understand the dynamics of interdomain traffic, we have gathered measurements from the Belgian research ISP, BELNET, during six consecutive days in December 1999. Additional information about this trace may be found in [21]. This ISP provides access to the Internet as well as to high-speed European research networks to universities, government and research institutions in Belgium. At that time, its national network was based on a 34 Mbps backbone linking the major Belgian universities and BELNET was the ISP with the highest capacity in Belgium. Its users are mainly researchers or students that are attached through their campus networks to the 34 Mbps backbone. In most cases, the campus networks have 100 Mbps Fast

Ethernet access to the BELNET backbone with at least 10 Mbps Ethernet LANs inside the campus, although some universities also provide access facilities, either through a pool of dialup modems or through cable modems. BELNET does not provide transit service, but is connected to about 40 external networks with high bandwidth links. It maintains high bandwidth peerings with two transit ISPs, has a direct connection with the Dutch SURFnet network, and was part of the TEN-155 European research network. In addition, BELNET had a router on the Belgian interconnection point (BNIX) and a router with a 34 Mbps link on the Dutch national interconnection point (AMS-IX).

Many researchers have analysed the behavior of Internet traffic based on measurements gathered in operational networks [14, 10, 8, 20]. Often, the analysis focuses on the packets captured on a single or a few different links [10, 20] for a relatively short period of time. These packet traces are very precise but require a large storage space. Other studies have relied on less precise information like SNMP statistics [8]. For this paper, we rely on a trace that differs by several aspects from the traffic traces usually analysed in the literature. First, the traffic trace we analyse covers six successive days of traffic and corresponds to the transmission of 2.1 Tbytes of IP packets. This duration is larger than many similar studies [10, 20]. Second, our trace covers all the interdomain links of the studied ISP. This contrasts with studies that often consider a single link or different links at different periods of time [10]. Third, we correlate the traffic information with the BGP routing topology. Fourth, our trace is not a packet trace but a microflow trace. It was gathered by using the `Netflow` [7] facility supported by the border routers of the studied ISP. When `Netflow` is activated, the border router exports to a monitoring station the starting and ending times, source and destination IP addresses, volume of traffic, IP protocol type (TCP/UDP) and TCP/UDP port numbers of each microflow that passed through the border router. This information allows the monitoring station to have detailed statistics of all the layer-4 flows that passed through the border router. Compared to packet traces, the `Netflow` trace is less precise but its main advantage is that it is possible to collect long `Netflow` traces. The trace was collected with `Cflowd` [6] and its true granularity is one minute. Our analysis only considers the incoming traffic of the studied ISP. It was three times larger than the outgoing traffic.

3.1 INTERDOMAIN TOPOLOGY

Before analyzing in details the collected traffic statistics, it is useful to have a first look at the BGP table of the studied ISP. We collected the BGP table of the studied ISP at the beginning of the measurements period, but did not record the changes to this table. For this paper, we assume that the BGP table of the ISP was stable during the six days period and perform all our analysis based on this BGP table. This is an approximation since we ignore the variations of the BGP routing table during the measurement period. However, since we rely on the BGP table of the studied ISP our analysis is more precise that other studies that relied on a BGP routing table collected at a different place and time than the studied packet traces [10, 16].

The BGP routing table of the studied ISP contained about 70000 active entries (prefixes), covering about 25 % of the total IPv4 address space. There were 6298 distinct AS in this BGP table. Among these autonomous systems, 4097 were only originating routes, 35 were only providing a transit service and 2145 were both originating routes and providing a transit service. The average size of the announced prefix was 22.19 bits.

An interesting point to consider in this BGP table is the length of the AS paths. These paths contain on average 4.5 hops, with a maximum length of 10. The average AS connectivity degree was about 3. Figure 2 presents the distribution of the reachable IP addresses in function of the length of the AS paths. This figure illustrates the concentration of ASs at a distance of $3-4$ AS hops. Only 4 % of the IP address space is reachable through the AS directly connected to the studied ISP. 90 % of the IP addresses of the BGP table are reachable through a path containing at most 4 AS hops. Clearly, the diameter of the Internet is relatively small. On this basis, the Internet viewed by our ISP does not significantly differ from the view of other ISPs [13].

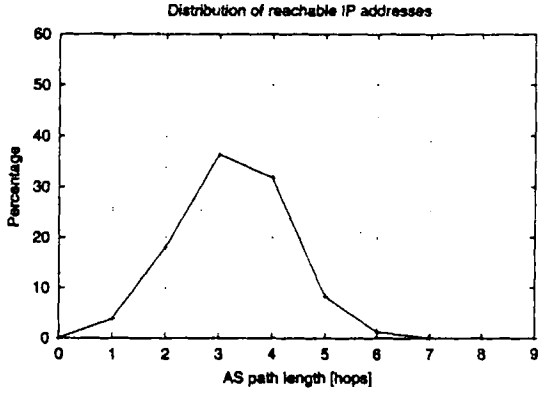

Figure 2: Distribution of reachable IP addresses.

3.2 RELATED WORK

Before looking at the traffic trace considered in this paper in details, it is useful to briefly compare it with existing studies of operational IP networks. While many papers have studied the packet size distribution or the types of applications in use [8, 20, 17], few papers have studied the variability of Internet traffic at the interdomain level.

[8] describes a detailed analysis of the traffic inside the T1 NSFNet backbone. This study relies on the traces and SNMP data gathered during two years. This paper also studied the packet size distribution, link loads, application

distribution and also the "topological" distribution of the traffic from a few NSFNet sites. This study revealed that some networks were responsible for a very large fraction of the backbone traffic. It should be noted that the fact that a small number of sources were responsible for a large portion of the traffic was already noticed on the ARPANET more than 25 years ago [14].

In [10] several one hour traffic traces from various US universities attached to the high-speed vBNS and from a commercial ISP are studied. This paper is one of the few that studies Internet traffic at the AS level. They show that a few AS are responsible for a large proportion of the traffic. More precisely, when considering flows for AS pairs, they show that 9 % of these AS pairs are responsible for 86.7 % of the total traffic of the studied traces.

Other researchers have studied the dynamics of the BGP routing protocol [15] and of the BGP routing tables [13]. These studies reveal important performance issues with the BGP protocol. However, they do not correlate the findings about the routing protocol and tables with the actual interdomain traffic.

Finally, several authors have proposed tools and methods [5, 11, 2] to accurately collect statistics in operational networks in order to perform traffic engineering. In these papers, the focus is mainly on the tools and unfortunately the characteristics of the traffic are not discussed in details except for illustration purposes.

4 TOPOLOGICAL TRAFFIC DISTRIBUTION

To understand the topological variability of interdomain traffic and the possible level of aggregation, we consider in this paper three different types of flows. Generally, a flow is defined as a set of IP packets that share a common characteristic. For example, a micro-flow is usually defined as the set of IP packets that belong to the same TCP connection, i.e. the IP packets that share the same source address, destination address, IP protocol field, source and destination ports. In this paper, we consider three different types of network-layer flows. A *prefix flow* is the set of IP packets whose source addresses belong to a given network prefix as seen from the BGP table of our ISP. An *AS flow* is defined as the set of IP packets whose source addresses belong to a given AS as seen from the BGP table of our ISP. Finally, a *level-n sink tree flow* is defined as the set of IP packets whose source addresses belong to the set of network prefixes that appear in the BGP routing table of the studied ISP with the same first n AS hops. This definition is similar to the sink trees of [16].

For example, in Figure 1, a prefix flow could be the set of IP packets whose source address belongs to the 138.48.0.0/16 subnet. In the same figure, the AS flow of AS20 would aggregate the IP packets originating from three different subnets (138.48.0.0/16, 193.190.0.0/16 and 130.104.0.0/16). The sink trees are defined on the basis of the BGP table maintained by the local ISP. Assuming that this ISP always selects as the best path the shortest path measured in AS hops, three level-2 sink trees would be defined in the simple topology shown in Figure 1. The AS21 sink tree would comprise the IP packets originating from AS21, AS30 and AS31. The AS20 sink tree would aggregate the IP packets originated by AS20, AS32, AS33 and AS34.

4.1 TRAFFIC DISTRIBUTION

The distribution of the interdomain traffic sources having the largest amount of traffic volume over the whole measurements should give a broad idea of the number of sources that need to be taken into account to engineer a given percentage of the total traffic. This number of traffic sources shall obviously depend on the granularity of the sources we consider.

Figure 3 presents the cumulative percentage of the total traffic over the measurements contributed by the interdomain sources having sent the largest amount of traffic volume, for network prefixes and ASs. In this figure, we have ordered the traffic sources by decreasing amount of the bytes sent over the whole measurements and computed the cumulative percentage of traffic contributed by the top sending traffic sources.

This allows to determine the minimum number of traffic sources required to capture a given percentage of the total traffic over the measurements. Note that we use throughout this paper the term *order statistics* to denote the traffic sources ordered by decreasing amount of the bytes computed over the entire measurements period. AS and prefixes have a similar distribution, the difference being due to the aggregation level. The top 100 ASs account for a little less than 60 % of the total traffic, while the top 100 prefixes for a little more than 40 %. This indicates that the traffic engineering task is possible at the interdomain level since a limited number of traffic sources already capture an important fraction of the total traffic. However, it should be noticed that more and more traffic sources are required to capture a higher percentage of the total traffic, which is graphically lessened by the use of a logarithmic x-axis scale. Our results confirm the findings of [10] where it was shown that 9 % of the flows between ASs were responsible for 90.7 % of the transmitted bytes, or [14] where they report that 90 % of the traffic is between 192 (12.6 %) of the site pairs. Our study shows that 9.8 % of the source ASs capture 90 % of total traffic while only 4.5 % of the source prefixes are able to capture 90 % of the total traffic.

Looking at traffic sources without concern about the topological locality of the top traffic sources is not sufficient in the context of the Internet. Indeed, one must also take into account the distance between the traffic source and its destination, because interdomain traffic engineering will probably be more difficult if the source is several AS hops away from the local network. Figure 4 presents the

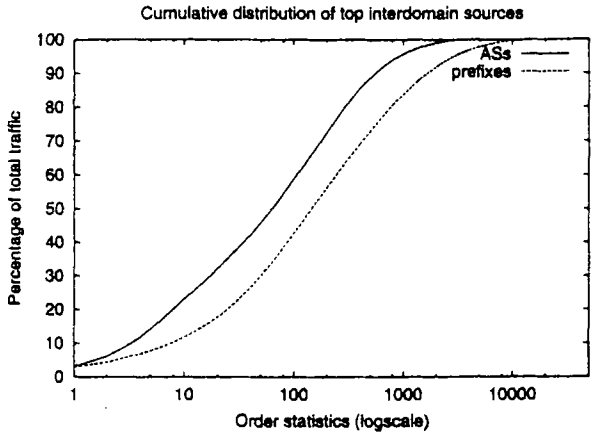

Figure 3: Cumulative traffic distribution for traffic sources.

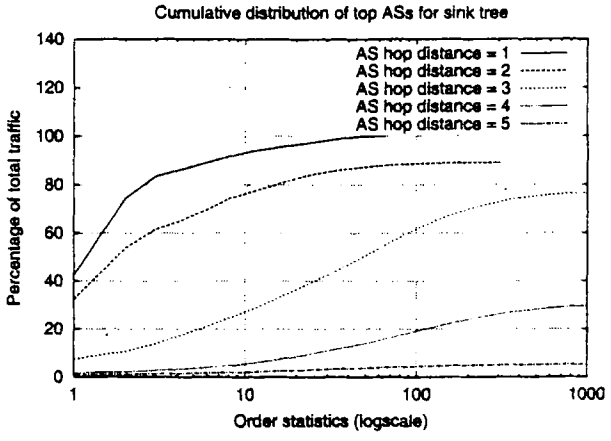

Figure 4: Cumulative traffic distribution for sink tree.

cumulative traffic distribution for the top ASs for each level of the sink tree. Because all levels of the sink tree do not contribute equally to the total traffic, we have plotted the cumulative traffic percentage for every level with respect to the total traffic seen during the measurements, thus independent of the locality of the traffic. This figure shows the uneven distribution of the total traffic among the different levels of the sink tree. All the traffic is received through the direct (level 1) peers, with more than 74 % of all traffic received from the two most important BGP peers. The second level is also dominated by a small number of ASs, with the top 10 accounting for more than 77 % of the total traffic. However, the traffic produced by ASs at 2 or more hops corresponds to only 89 % of the total traffic, meaning that about 11 % of the traffic comes directly from the ASs located at an AS hop distance of 1. Subsequent levels of the sink tree require an increasingly important number of ASs to capture a large fraction of the traffic. 76 % (resp. 30 %) of the total traffic is produced by ASs located at 3 (resp. 4) or more AS hops. The most important ASs for each of the four successive levels of the sink tree produce respectively 42, 32, 3.9 and 0.9 percent of the total traffic. The contribution to the total traffic of each level of the tree follows the distribution of reachable IP addresses shown on Figure 2. The first level generates 11 % of the total traffic for 3.9 % of the reachable address space, level 2 has 12.4 % for 18.1 % of the reachable addresses, level 3 46.6 % for 36.4 % of the reachable addresses, level 4 24.9 % for 31.8 % of the reachable addresses and level 5 4.3 % for 8.3 % of the reachable addresses. Subsequent levels of the sink tree represent less than 0.9 % of the total traffic.

Based on these measurements, interdomain traffic engineering will have to rely on very few levels of the sink tree, probably just the first two in our case. Our study also shows that in order to traffic engineer an important part of the total traffic requires the ability to influence the traffic up to several AS hops, not just one.

4.2 STABILITY OF THE TRAFFIC SOURCES

Predictability of the interdomain traffic should ideally rely on the stability of the highest sending traffic sources from one day to another. For the possibility of using the information of the *order statistics* for choosing a subset of the sources that constitutes a good prediction for the next day's *order statistics*, it is required that the top n sources for some given day be a good indicator for the top n sources for the next day (stability in presence). In addition, the top n traffic sources should also cover a similar percentage of the total traffic over the day, which should also be stable (stability in traffic volume).

Figure 5 presents the cumulative distribution of the traffic for each day of the measurements, for source ASs. All days have a similar distribution, with the top 100 ASs representing about 60 % of the total traffic over the day. Hence, stability in traffic volume can be assumed. For what concerns the stability in presence for *order statistics*, table 1 gives the number of top ASs that are the same for two consecutive days of the measurements. It shows that most ASs present in the top n *order statistics* for one day are still in the same range of *order statistics* during the next day. The low value for the similarity between the top 10 between day 1 and 2 could be explained by the fact that the first day happens to be a Sunday, while the other rows correspond to weekdays. It could also be a statistical outlier or even a perfectly common event.

Table 1: Day-to-day similarity of order statistics.

Day	Top 10	Top 100	Top 1000
$1 \rightarrow 2$	5	70	773
$2 \rightarrow 3$	9	79	819
$3 \rightarrow 4$	8	64	723
$4 \rightarrow 5$	7	58	713
$5 \rightarrow 6$	9	74	821

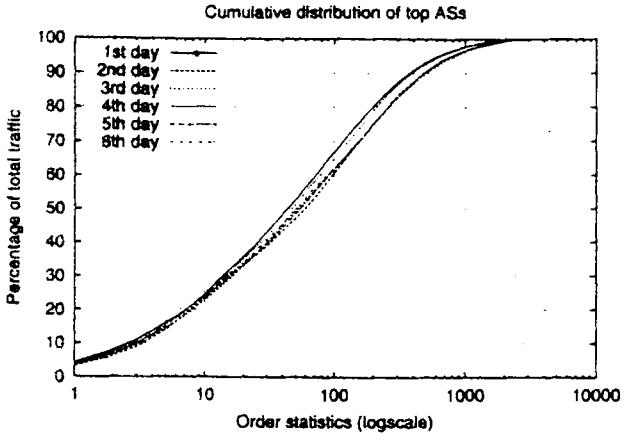

Figure 5: Cumulative traffic distribution for source ASs.

5 TRAFFIC VARIABILITY

To obtain an idea of the actual complexity of the inter-domain traffic engineering task, one should care about the number of intermediate ASs our ISP would have to influence in order to engineer its traffic pattern. Assuming that it might be able to predict its forthcoming traffic demand, how many traffic sources should it need to influence in order to control its access point load? Section 4.1 has dealt with this problem.

This section on the other hand studies the variability of the interdomain traffic sources, i.e. the number of sources sending traffic over some time interval as well as their stability expressed in terms of their "activity". We call in this section a traffic source (or intermediate AS) *active* whenever there is at least one byte of traffic that originates from it (or flows through it) during given time interval. It is important to remember in the case of prefixes and source ASs, we only consider the actual traffic sources, from which traffic originates. For the sink tree, we consider all ASs that send traffic as well as ASs that are crossed by the traffic to attain the local ISP, as seen by the BGP routing table.

5.1 ACTIVE SOURCES

Figures 6, 7 and 8 show the average number of *active* sources for each AS hop count, for timescales equal to 15 minutes, 1 hour, 4 hours and 12 hours, for prefixes, ASs and the sink tree respectively. Comparing Figures 6 and 7 illustrates the important reduction in terms of traffic sources when considering ASs instead of prefixes, with a reduction that depends on the timescale used, longer timescales providing a higher reduction. On the other hand, the number of ASs that are *active* on the sink tree (Figure 8) does not differ much from the one of AS sources (Figure 7), meaning that at a topological point of view, the location of the AS sources corresponds more or less to the sink tree, at least as seen by our BGP routing table. All results from this section show that there is a strong relationship between the topological distribution of the traffic sources and the distribution of the reachable IP addresses shown on Figure 2, with most traffic sources located at an AS hop distance between 3 and 4 AS hops. An important issue concerns the total number of *active* sources ranging on average from about 7000 to 24000 for prefixes, 3000 to 6000 for source ASs and also from 3000 to 6000 for the sink tree.

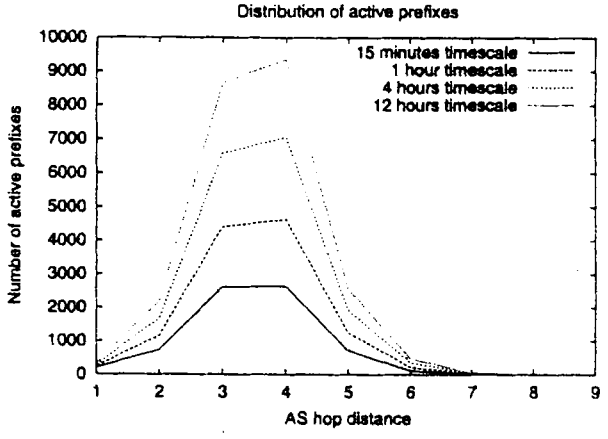

Figure 6: Number of active prefixes.

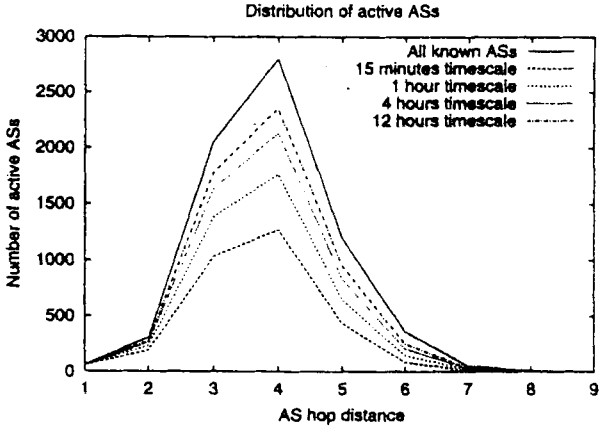

Figure 7: Number of active ASs.

Table 2 compares the average gain of using ASs instead of prefixes and the sink tree instead of the ASs. The gain is defined by the ratio of the average number of *active* sources (or transit ASs in the case of the sink tree) for the initial aggregation level divided by the average number of *active* sources for the target aggregation level for four timescales, over the whole measurements. While using ASs instead of prefixes allows for an important reduction going from a little more than 2 to more than 4, the sink tree on the other hand makes the average number of ASs that need to be taken into account increase ($gain < 1$). This happens because the two aggregation levels are conceptually different. The sink tree takes into account every source as well as every intermediate AS that need to be crossed. This

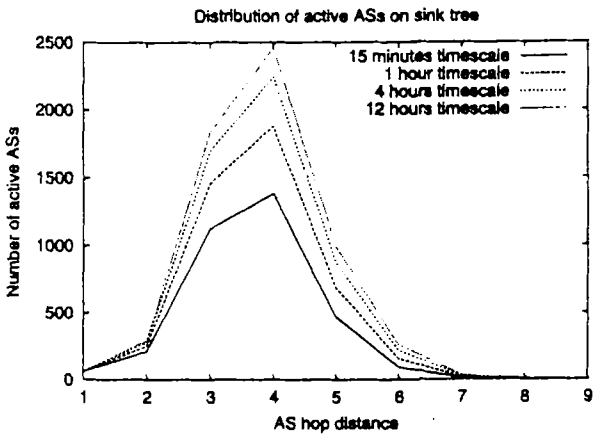

Figure 8: Number of active ASs on the sink tree.

means that some source ASs require intermediate ASs on their paths that are not themselves traffic sources. But the number of intermediate ASs that need to be added is limited since the difference between the source ASs and the sink tree is less than 10 %. Using a sink tree is hence due to provide an important advantage in comparison to source ASs (or prefixes) because most source ASs (or prefixes) are located at an AS hop distance of 3 or 4. The actual number of ASs that will need to be aware of this traffic should be multiplied by some factor to take into account the intermediate ASs on the path for each traffic source. The true number of *active* ASs to be influenced when considering source ASs aggregation should hence be in-between the one of prefixes and the one of the sink tree.

The sink tree aggregation provides an interesting gain in comparison to source ASs (or prefixes), in terms of the number of ASs that will have to be aware of the traffic.

Table 2: Gain from source aggregation for number of active sources.

Timescale	Prefixes → ASs	ASs → sink tree
15 minutes	2.27	0.92
1 hour	2.81	0.94
4 hours	3.48	0.95
12 hours	4.14	0.96

5.2 IN+OUT OPERATIONS

For assessing the dynamics of the interdomain traffic, this section relies on the number of traffic sources that become *active* or stop sending traffic from a given time interval to the next time interval, which we call *in+out* operations. This provides an idea of the stability (through its presence or absence) of the traffic sources. In the case traffic variability is low, then the number of traffic sources that change their state between *active* and *inactive* between two consecutive time intervals should also be low.

Figures 9, 10 and 11 present the average number of *in+out* operations for traffic sources and several time interval lengths, for prefixes, ASs and the sink tree respectively. Once more, the largest reduction arises when aggregating prefixes into ASs. Variability on the sink tree (Figure 11) is similar to the one of the ASs (Figure 10), thus using the complete sink tree provides an important advantage in comparison with AS sources, at least for *in+out* operations. The average number of *in+out* operations for the sink tree represents about 36 % of the number of *active* ASs for a timescale of 15 minutes, but decreases to about 21 % for 1 hour, 14 % for 4 hours and 10 % for 12 hours timescales. This is a direct consequence of the fact that larger timescales allow for more traffic sources to be *active* as well as less *in+out* operations to occur. In addition, variability is mainly located at levels 3 and 4 of the tree, the first two levels being quite stable. The total number of *in+out* operations ranges on average from about 4500 to 9000 for prefixes and from 650 to 1200 for both source ASs and the sink tree. The disadvantage of the prefix aggregation level is illustrated on Figure 9 where we can see that increasing the timescale increases the number of *in+out* operations while other aggregation levels exhibit the opposite behavior. This means that prefixes have very short periods during which they are *active*, with at the same time few prefixes that are active during two consecutive time intervals on average. Source ASs and the sink tree hence provide a better aggregation level for what concerns the variability of the traffic sources.

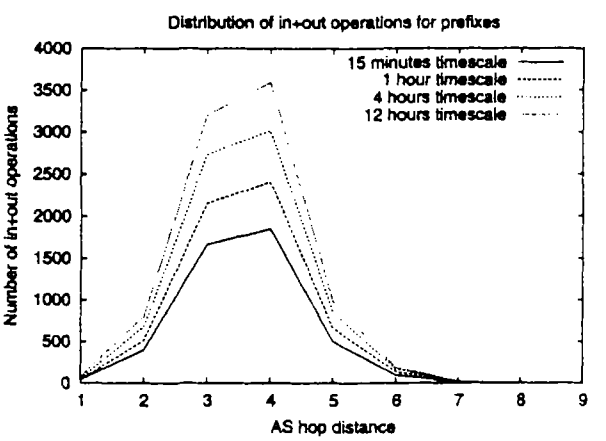

Figure 9: Number of in+out operations for prefixes.

The average gain from using coarser traffic sources (and intermediate ASs for the sink tree) is shown on Table 3. The gain is defined as the ratio of the average number of *in+out* operations for the initial aggregation level divided by the average number of *in+out* operations for the target aggregation level, for four timescales and the whole measurements period. The number of *In+out* operations is greatly reduced when going from prefixes to ASs. The re-

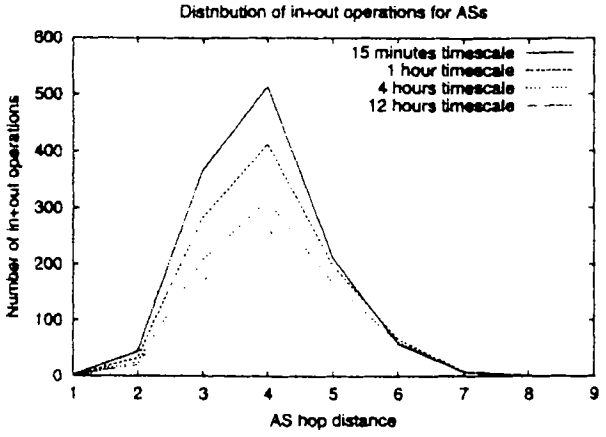

Figure 10: Number of in+out operations for ASs.

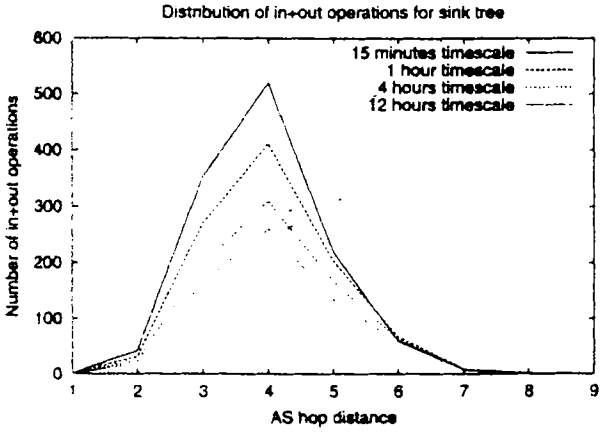

Figure 11: Number of in+out operations for ASs on sink tree.

duction goes from 3.77 for a 15 minutes timescale to more than 13 for the 12 hours timescale. Using traffic source aggregation is hence extremely effective for reducing the number of *in+out* operations. The column related to the change in the aggregation level from ASs to the sink tree confirms numerically the similarity of Figures 10 and 11, by showing a gain very close to 1. This means that the number of *in+out* operations is almost identical for source ASs and the sink tree.

As in the case of *active* sources, the number of *in+out* operations for source ASs do not take into account the fact that in practice, this number must be multiplied by a factor of about 3 or 4 because an *in+out* operation for a given source AS will correspond to modifying the behavior of all intermediate ASs on the AS path, not just the source AS. Once more, the sink tree provides an important gain in terms of the number of ASs to be influenced for some traffic engineering task, because the numbers for the sink tree take into account all ASs, traffic sources as well as intermediate hops.

Table 3: Gain from source aggregation for number of in+out operations.

Timescale	Prefixes → ASs	ASs → sink tree
15 minutes	3.77	1
1 hour	5.89	1
4 hours	9.56	1.01
12 hours	13.46	1.01

5.3 Traffic volume-aware variability

Sections 5.1 and 5.2 have studied the variability of the interdomain traffic without considering the variability in volume. Traffic engineering on the other hand is due to mainly affect high bandwidth flows. To evaluate the impact of high bandwidth flows, we study in this section the activity and the *in+out* operations for the high-bandwidth flows. For this, we consider only the ASs on the sink tree whose average bandwidth over a given time interval is at least equal to *trigger* bytes per minute.

Table 4 compares the average number of active ASs (without *trigger*) with the average number of ASs sending at least 100 Kbytes or 1 Mbytes per minute on the sink tree. As expected, the number of high bandwidth ASs is much lower than the number of active ASs (without *trigger*). When considering 15 minutes time intervals, there are on average more than 3 times less ASs sending at least at 100 Kbytes per minute than the total number of *active* ASs on the sink tree. During a 12 hours period, there are on average 2.5 times less ASs sending on average at 1 Mbytes per minute than the total number of active ASs, on the sink tree.

Table 4: Number of active ASs on sink tree.

Timescale	All ASs	100 Kbytes p. min. ASs	1 Mbytes p. min. ASs
15 minutes	3346	924	396
1 hour	4512	1615	801
4 hours	5403	2602	1497
12 hours	5496	3631	2347

Figure 12 (to be compared with Figure 11) presents the effect of *triggers* on the average number of *in+out* operations that arise when considering a given time interval length (timescale) for the sink tree. The impact of a *trigger* is higher for an AS hop distance of 3 and 4, because there are much more AS sources for these levels of the sink tree. Considering the 1 Mbytes trigger shows the small effect of the timescale on the reduction of the average number of *in+out* operations. There are on average about 48 *in+out* operations on the sink tree for the 15 minutes timescale while about 60 for the 12 hours timescale. Using a trigger directly on AS sources (not shown here) provides similar

results to the ones of the sink tree, except that all AS hop distances exhibit an important reduction in the number of *in+out* operations. The advantage of the sink tree transpires because of heavy traffic multiplexing, even if the gain is restricted to the two first two levels of the tree. Although the drastic reduction in the number of *in+out* operations could imply that the *trigger* has suppressed almost all ASs, the total number of ASs that are active after having applied the *trigger* is still important, with about 396 ASs on the sink tree for a timescale of 15 minutes and 2347 for a timescale of 12 hours, for the 1 Mbytes *trigger*.

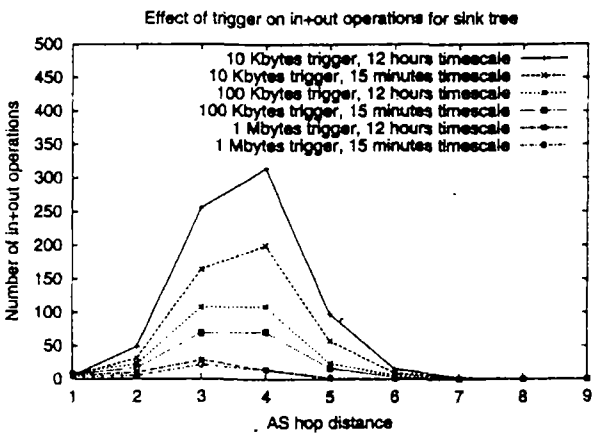

Figure 12: Effect of a trigger on in+out operations for sink tree.

Considering only high volume flows allows to reduce the number of *active* ASs on the sink tree, but this number is still large. Table 5 presents the average number of *in+out* operations on the sink tree for all ASs and for the high bandwidth ASs. This table should be compared with Table 4. We can see from Table 5, that during 15 minutes periods, about 15 % of the 100 Kbytes ASs will be affected by an *in+out* operation.

Table 5: Average number of in+out operations for sink tree.

Timescale	All ASs	100 Kbytes p. min. ASs	1 Mbytes p. min. ASs
15 minutes	1206	185	48
1 hour	997	185	41
4 hours	773	238	57
12 hours	646	280	60

Reducing interdomain traffic variability while capturing an important fraction of the total traffic is possible by taking into account the traffic volume information of the interdomain flows. Note however that [22] has shown that interdomain traffic was self-similar over large time-scales. This means that even if we found some stability on the topological aspects of the traffic dynamics, the burstiness of the interdomain flows is a real concern.

6 Conclusion

We have studied in this paper the implications of interdomain traffic on traffic engineering by correlating the temporal traffic dynamics and the topological view provided by the BGP routing protocol. We have shown that the traffic was unevenly distributed among the different levels of the BGP sink tree in terms of AS hops. We have seen that most traffic originates from levels 3 and 4, consistently with the distribution of the reachable address space provided by the BGP routing table. We have studied the traffic distribution for the traffic sources and found that a limited percentage of them captured an important fraction of the total traffic, 9.8 % of the source ASs sending about 90 % of the total traffic. Then, we have looked at the day-to-day stability of the highest sending traffic source ASs and shown that they were stable both in terms of their contribution to the total traffic and in terms of their presence among the top n traffic sources.

We have then studied the variability of the traffic sources based on two measures. First, we have studied the average number of sources that were sending traffic during a given time interval. This number has shown that using traffic aggregation is useful for reducing the number of traffic sources, the sink tree providing an important gain in comparison to source ASs. Second, we have computed the average number of traffic sources that become active or inactive during a given time interval, which we called *in+out* operations. We have also shown that the sink tree aggregation was advantageous for limiting the number of such operations. However, the variability is still important for levels of the sink tree beyond 2 AS hops. Finally, we have shown that that the sources sending a large amount of data had a lower variability than all the sources.

Acknowledgements

This work was supported by the European Commission within the IST ATRIUM project. This paper would not have been written without the traffic trace provided by Marc Roger from the Belgian research network BELNET.

Manuscript received on December 13, 2001.

References

[1] B. Abarbanel and S. Venkatachalam. BGP-4 support for Traffic Engineering. Internet draft, draft-abarbanel-idr-bgp4-te-00.txt, work in progress, May 2000.

[2] P. Aukia, M. Kodialam, P. Koppol, T. Lakshman, H. Sarin, and B. Suter. RATES: A server for MPLS traffic engineer-

ing. *IEEE Network Magazine*, pages 34–41, March/April 2000.

[3] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao. Overview and Principles of Internet Traffic Engineering. Internet draft, draft-ietf-tewg-principles-02.txt, work in progress, December 2001.

[4] D. Awduche, J. Malcom, B. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. *Internet RFC 2702*, September 1999.

[5] R. Caceres, N. Duffield, A. Feldmann, J. Friedmann, A. Greenberg, R. Greer, T. Johnson, C. Kalmanek, B. Krishnamurthy, D. Lavelle, P. Mishra, K. Ramakrishnan, J. Rexford, F. True, and J. van der Merwe. Measurement and analysis of IP network usage and behavior. *IEEE Communications Magazine*, May 2000.

[6] CAIDA. cflowd: Traffic flow analysis tool. Available from http://www.caida.org/tools/measurement/cflowd/, 1998.

[7] Cisco. NetFlow services and applications. *White paper*, available from http://www.cisco.com/warp/public/732/netflow, 1999.

[8] K. Claffy, H. Braun, and G. Polyzos. Traffic characteristics of the T1 NSFNET backbone. *INFOCOM93*, 1993.

[9] S. Van den Bosch, F. Poppe, and G. Petit. Single-Path Traffic Engineering with Explicit Routes in a Differentiated-Services Network. *IEEE International Conference on Communications*, Helsinki, Finland, June 2001.

[10] W. Fang and L. Peterson. Inter-AS traffic patterns and their implications. *IEEE Global Internet Symposium*, December 1999.

[11] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving traffic demands for operational IP networks: methodology and experience. In *Proc. ACM SIGCOMM2000*, September 2000.

[12] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. *INFOCOM2000*, March 2000.

[13] G. Huston. Analyzing the Internet's BGP routing table. *Internet Protocol Journal*, 4(1), 2001.

[14] L. Kleinrock and W. Naylor. On measured behavior of the ARPA network. In *AFIS Proceedings, 1974 National Computer Conference*, Vol. 43, pages 767–780. John Wiley & Sons, 1974.

[15] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. An experimental study of Internet routing convergence. *SIGCOMM 2000*, August 2000.

[16] P. Pan, E. Hahne, and H. Schulzrinne. BGRP: A Tree-Based Aggregation Protocol for Inter-domain Reservations. *Journal of Communications and Networks*, 2(2), June 2000.

[17] V. Paxson and S. Floyd. Wide-Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Trans. on Networking*, Vol. 3 (1995), 226-244, 1995.

[18] J. Stewart. *BGP4 : interdomain routing in the Internet*. Addison Wesley, 1999.

[19] Telstra. BGP table report. Available from http://www.telstra.net/ops/bgp/, 2001.

[20] K. Thompson, G. Miller, and R. Wilder. Wide-Area internet traffic patterns and characteristics. *IEEE Network magazine*, 11(6), November/December 1997.

[21] S. Uhlig and O. Bonaventure. On the cost of using MPLS for interdomain traffic. In *Proc. first COST263 workshop on Quality of future internet services*, J. Crowcroft, J. Roberts, and M. Smirnov, editors, pages 141–152. Springer Verlag, LNCS1922, September 2000.

[22] S. Uhlig and O. Bonaventure. Understanding the Long-term Self-similarity of Interdomain Traffic. In *Proc. second COST263 workshop on Quality of future internet services*, M. Smirnov, J. Crowcroft, J. Roberts, and F. Boavida, editors, pages 286–298. Springer Verlag, LNCS2156, September 2001.

[23] Y. Wang, Z. Wang, and L. Zhang. Internet traffic engineering without full mesh overlaying. *INFOCOM2001*, April 2001.

[24] Z. Wang. Internet traffic engineering. *Special section of IEEE Network magazine*, March-April 2000.