# Inbound Traffic Engineering for Multihomed ASs Using AS Path Prepending

**Rocky K. C. Chang, The Hong Kong Polytechnic University**
**Michael Lo, Open University of Hong Kong**

## Abstract

This article considers the AS path prepending approach to engineer inbound traffic for multihomed ASs. The AS path prepending approach artificially inflates the length of the AS path attribute on one of the links in hopes of diverting some of the traffic to other links. Unlike the current practice that determines the prepending length in a trial-and-error way, we propose `AutoPrepend`, a systematic and automated procedure for conducting prepending. The entire process consists of four components: passive measurement, active measurement, traffic change prediction, and AS path update. The main idea is to predict the traffic change due to a new prepending value before effecting the change. We have also employed a number of mechanisms to minimize intrusion into the normal operation of the Internet. We have deployed `AutoPrepend` on a noncommercial site and evaluated its effectiveness based on the measurements collected over six months.

The Internet routing architecture today consists of the *intradomain* and *interdomain* levels. The former refers to routing within a domain or an autonomous system (AS), while the latter refers to the routing between ASs. An AS is defined as "a connected group of one or more IP prefixes run by one or more network operators which has a single and clearly defined routing policy" [1], and each AS is uniquely identified by an AS number. An IP prefix (or just prefix) is the network part of an IP address that is examined by routers to make forwarding decisions. Figure 1 shows an example of interconnected ASs. Both AS1 and AS9 are *stub ASs*, while AS2–8 are *transit ASs*, which provide transit service for their customers and peers. Moreover, each stub AS is multihomed to two transit ASs; thus, they can receive and send packets via both links at the same time. The figure also shows the end-to-end routing path from a host in AS9 to another host in AS1. The entire routing path is therefore composed of intradomain routing paths and interdomain routing paths, alternating between them.

While there are a number of routing protocols available on the intradomain level, such as Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Intermediate System to Intermediate System (IS-IS), the Border Gateway Protocol (BGP), currently at version 4, is the only standard for exchanging reachability information on the interdomain level [2]. Besides supporting classless interdomain routing, an important function of BGP is to facilitate *policy routing*. That is, each AS exercises its own preference for which routes to accept and where to further advertise them. To support such autonomous route decisions, a prefix announced in a BGP route advertisement is usually attached with a number of *path attributes*. An important attribute is known as *AS path*, which records the forwarding path in terms of the AS numbers. For example, the AS path attached to a prefix announced by AS1 in Fig. 1 is {1} (assume that the AS number is the digit after AS). When AS2 continues to announce the prefix, the AS path becomes {1 2}. Eventually, the prefix announced to AS9 by a BGP router in AS7 is attached with an AS path of {1 2 5 6 7}.

A BGP router makes a route decision based on the values of the BGP path attributes, which will be elaborated further in the next section. Therefore, the final end-to-end forwarding path is essentially a result of the autonomous route decisions of the ASs between the two endpoints. Each link in Fig. 1 may represent a BGP connection between two BGP routers in the respective ASs. Assume that AS1 advertises a prefix to the two links, which is in turn advertised to all BGP connections in the figure. Based on the final forwarding path, AS5's preferred next hop for the prefix is AS2 (instead of AS3), whereas AS6's preferred next hop is AS5 (instead of AS4 or AS3), and so on.

Traffic engineering is another important problem to tackle at the routing layer. The general problem at hand is how to influence the traffic flowing into (inbound) and out of (outbound) an AS, such that a given set of performance objectives can be achieved. Traditionally, the traffic engineering problem does not concern stub ASs, because most of them are single-homed. However, as the number of multihomed stub ASs has been increasing rapidly for the last few years, the problem of engineering the inbound and outbound traffic becomes very important for a large number of ASs in the Internet. In this article we mainly consider coarse-grained traffic engineering issues over a longer timescale, such as load balancing the links on an hourly basis. With additional mechanisms at the transport and even application layers, a more fine-grained control could be exerted on the inbound traffic; however, these schemes are not the focus of this article.

The outbound case concerns the selection of the best egress point for traffic originated within the AS. Engineering the outbound traffic can be performed based on the traffic matrix


0890-8044/05/$20.00 © 2005 IEEE

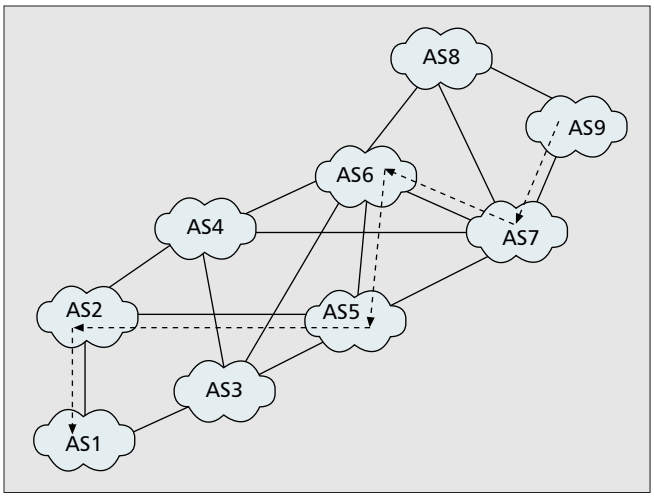



Figure 1. *Interconnected ASs and a forwarding path from stub AS9 to stub AS1.*

observed in the network. The inbound case, on the other hand, concerns the selection of the best ingress point for receiving traffic generated outside the AS. Based on the previous discussion on policy routing, it is clear that a multihomed stub AS cannot effectively dictate the inbound traffic distribution. In this article we propose a systematic and automated procedure called `AutoPrepend` to influence the inbound traffic for multihomed ASs. This procedure is based on a long practiced method called *AS path prepending*. However, this method is often performed in a trial-and-error way, and there is a lack of detailed measurements on its effectiveness. This article attempts to fill these gaps by proposing `AutoPrepend`, which has been deployed on a noncommercial site. We have also evaluated its effectiveness based on a six-month measurement study.

In the next section we first detail the BGP route selection process and then discuss other approaches to the inbound traffic engineering problem. After that, we present the passive and active measurement components of `AutoPrepend`. We then continue with the traffic prediction component of `AutoPrepend` and the experimental results. We finally conclude this article by discussing some future work.

## BGP and Inbound Traffic Engineering

A BGP router in a transit AS generally receives several routes for a given prefix from its neighboring BGP routers, and each route is attached with various path attributes, such as LOCAL-PREF (local preference), AS path, and others, including proprietary attributes [3]. The BGP router determines which route to accept based on the AS's import routing policy and attribute values. The route selection can be based on a highest LOCAL-PREF value, a shortest AS path length, e-BGP routes over i-BGP routes, and so on [4]. The AS path length is equal to the count of AS numbers in the AS path attribute. After determining the best route to a prefix, the BGP router may further announce this route to a selected set of neighboring BGP routers but withhold it from another set, depending on the AS's export routing policy. As a result, different BGP routers end up having different views of the routes in the Internet. Moreover, without additional mechanisms, an AS cannot control the end-to-end forwarding path from an external source to a prefix inside the AS. Before introducing `AutoPrepend`, it is helpful to review other approaches to controlling the inbound traffic.

**Selective announcement:** This approach announces nonoverlapping prefixes to different links. For example, instead of announcing 158.32.0.0/16 to both upstream Internet service providers (ISPs) in Fig. 2a, this approach announces two nonoverlapping longer prefixes to two different links. As a result, the traffic destined to these two prefixes will reach the network via the two respective links. Although this approach is very easy to deploy, it reduces the network resilience as only a single ISP is used for each prefix. Moreover, the actual AS path could be lengthened.

**Prefix splitting:** Similar to the first approach, this approach splits a prefix into longer prefixes. The difference is that the original prefix is also announced. As shown in Fig. 2b, the two prefixes advertised are overlapped, and the more specific one is sent to ISP 2. Under the longest-prefix-matching packet forwarding algorithm, traffic destined to 158.32.128.0/17 is expected to reach the network via ISP 2 only. Therefore, ISP 1 essentially serves as a backup for 158.32.128.0/17. Clearly, this approach also suffers from the same problem of incurring a longer AS path. More important, the longer prefixes introduced by both approaches will cause BGP routing tables to grow very quickly. Because of that, BGP routers today are usually configured not to accept routes that exceed a certain prefix length (24 currently).

**NAT-based approaches:** Another approach is based on dynamic network address translation (NAT). The idea is to translate the source address in an outgoing packet, such as TCP SYN, to the external address of an multihomed NAT router, such that the returned traffic can be affixed to the corresponding link [5]. The main advantage of this approach is to offer a more fine-grained control. It is also useful for traffic engineering on a shorter time scale (minutes), whereas the first two approaches are on a longer timescale (hours). However, this approach requires dynamic domain name service (DNS) to remove the corresponding DNS record upon detecting link failures.

**Overlay on top of BGP:** An overlay policy control architecture (OPCA) has recently been proposed to address BGP's slow convergence problems [6]. The OPCA can be seen as an overlay on top of BGP. The key idea is to separate the policy

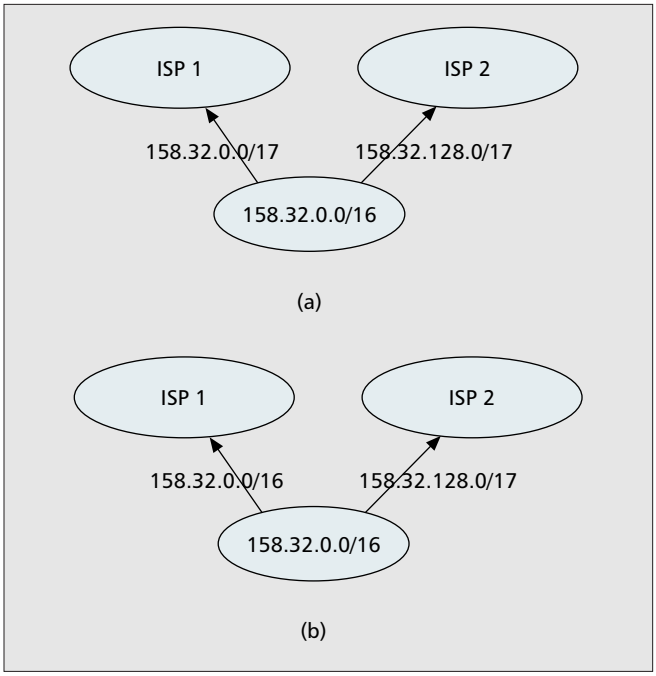



Figure 2. *Two static methods of controlling the inbound traffic: a) selective announcement; b) prefix splitting.*

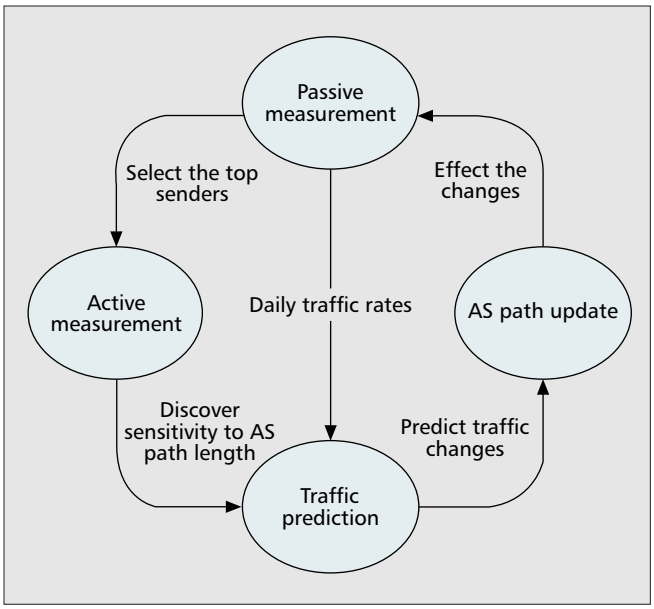



Figure 3. *The four components of* `AutoPrepend`.

from routing, both of which are supported by BGP, so that a faster channel (the overlay network) can be used to handle routing policy changes. Possible applications of the OPCA include improving route failover time and balancing the inbound traffic load for multihomed networks. The OPCA consists of a number of agents, databases, directories, and an overlay policy protocol for communications.

## `Autoprepend`: An AS Path Prepending Approach

Besides selective announcement and prefix splitting, AS path prepending is another popular approach to controlling the inbound traffic at the IP layer. The prepending method artificially inflates the AS path by including multiples of its own AS number. For example, if AS1 prefers to receive AS9's traffic through AS3 in Fig. 1, it will advertise to AS2 with an AS path of {1 1 1 1}, whereas the one to AS3 is still {1}. The AS path in the former case is said to have a *prepending length* of 3. A sufficient increase in the AS path length could change the routing path to the extent that incoming traffic is diverted to the link connected to AS3. It has been reported that over 30 percent of observed routes have some amount of AS path prepending, and most of these prepended routes have prepending lengths of 1 and 2 [7].

The prepending approach offers a number of advantages over the selective announcement and prefix splitting methods. First of all, it does not increase the BGP table size or compromise on resilience. It has been widely deployed, and its effectiveness has already been demonstrated. Moreover, the prepending approach can be used together with the BGP community attribute to bring about an even better result [8]. However, the prepending approach suffers from two interrelated problems. First, in the lack of a systematic approach to determine the prepending length, the prepending method is often performed in an ad hoc manner, which may end up overly effective. That is, too much traffic is redistributed from one link to another, resulting in possible link congestion. Our proposed `AutoPrepend` is designed to address these shortcomings, so that the AS prepending approach can be conducted more effectively. Before delving into the details, we show in Fig. 3 the four components of `AutoPrepend`.

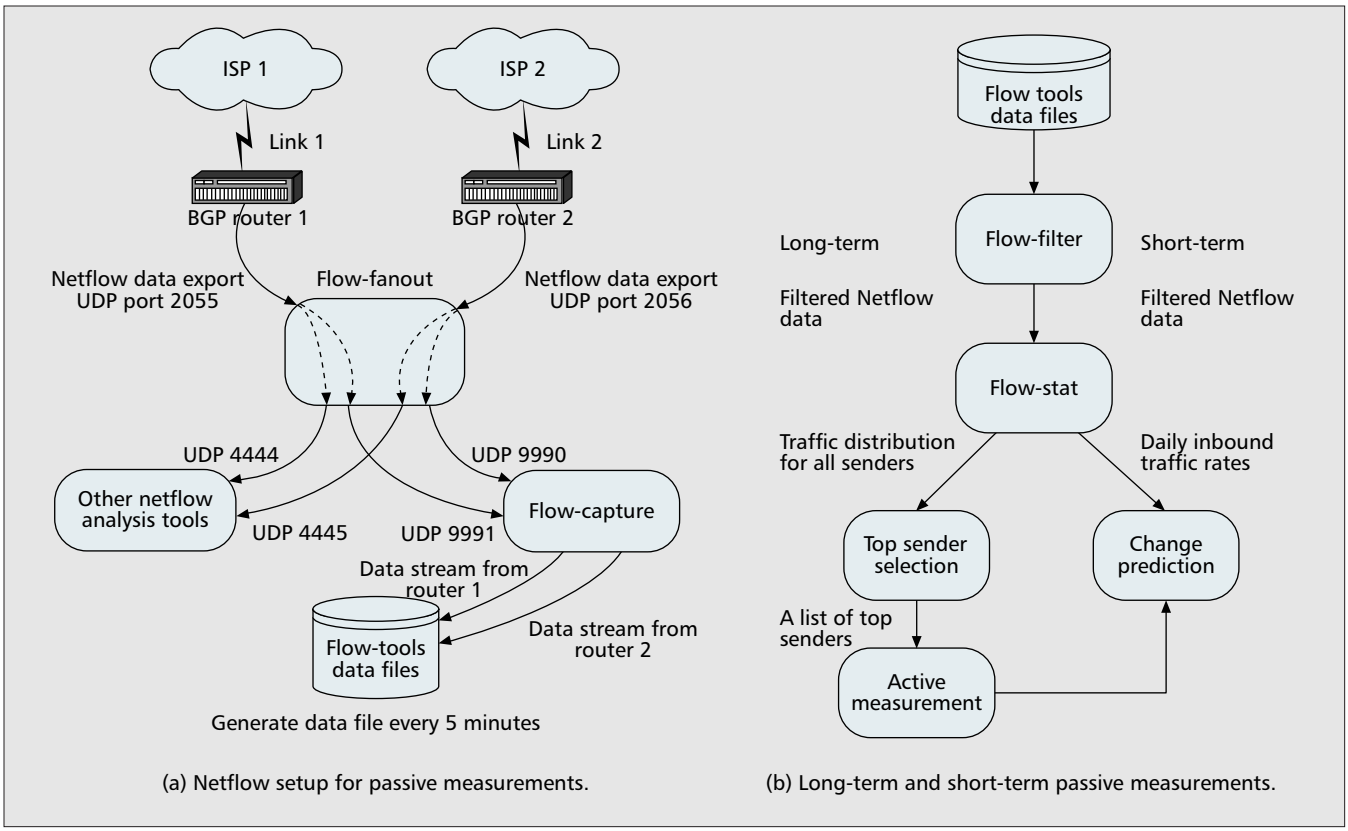
- The *passive measurement* component records the statistics of all inbound traffic flows. An important purpose is to identify the long-term top traffic senders that are responsible for most of the inbound traffic. These top senders are usually popular Web sites, proxy servers, firewalls, and NAT boxes. To minimize disruption to the Internet, only these top senders are considered in the active measurement component.
- The *active measurement* component discovers whether the traffic from the top senders would arrive at a different link when the advertised AS path length is artificially lengthened on one of the links. One way to discovering the change is to send an ICMP echo request to each top sender, and record the link that receives the ICMP echo reply.
- Based on the active measurement and short-term passive measurement results, the *traffic prediction* component predicts the changes in the traffic volume coming into the links when the prepending length changes.
- If the predicted outcomes satisfy the traffic engineering goal, the *AS path update* component will effect the change by advertising the prefixes with the new AS path.

### The Passive Measurement Component

Figure 4a depicts a typical passive measurement setup in a dual-homed AS using Netflow [9]. Netflow provides per-flow traffic characteristics, such as the start and end times of the flow, and the total amount of data [9]. Compared with other packet-based traffic capturing tools, such as Simple Network Management Protocol (SNMP) and remote monitoring (RMON), the flow-based tool is sufficient for the purpose of traffic engineering. Moreover, the flow-based tool does not generate as much data and does not require specific hardware to run. The Netflow data captured by the two routers are analyzed using Flow-tools [10]. The right side path in Fig. 4a shows that the Flow-capture program collects the received data and stores them in compressed data files every five minutes, amounting to around 80–150 Mbytes data daily. Moreover, the set of top senders can be identified from an analysis of the Netflow data, as depicted on the lefthand path in Fig. 4b.

We deployed the passive measurement setup in a test site (a dual-homed AS as in Fig. 4a) to collect data between September 2002 and January 2003. In each month, the number of unique source IP addresses observed ranges between 500,000 and 650,000. We rank the addresses in a nonincreasing order of the traffic volume that they sent to the test site. That is, the source with the highest traffic volume is ranked first, and the second highest traffic volume is ranked second, and so forth. We call this order of source addresses as *sender order*. Figure 5a shows the results for the entire test site in which the total cumulative traffic volume is normalized to 100 percent. All five graphs are very similar to each other, in spite of some slight variations in the lower portions of the graphs. These results are consistent with the previous studies on the related issues, such as [11, 12], in that a relatively small number of senders are responsible for the majority of the traffic. In our case, the first 100 senders contributed to around 40–50 percent of the total traffic. Moreover, there were no more than 110,000 senders responsible for 99 percent of the inbound traffic volume, which comprised around 15 percent of the total number of senders. By relaxing the threshold to 95 percent, the number of top senders dropped to around 25,000.

In a later section we use an operational dialup network in the test site to study the effectiveness of `AutoPrepend`. The main reason for using the dialup network is that it involves a relatively small number of users, and yet the traffic destined to this network is realistic enough. Therefore, we also present the passive measurement results for the dialup network in Fig. 5b. The five graphs are even closer to each other than those in Fig. 5a. In each month, the total number of unique source




■ Figure 4. *The passive measurement component in* `AutoPrepend`.

IP addresses observed ranges between 180,000 and 200,000. Similar to the previous case, the first 20 senders contributed to 10 percent of the total traffic, and the first 100 senders contributed to around 30 percent of the traffic. Around 30,000 (17 percent) senders contributed to 99 percent of the total inbound traffic entering into the dialup network.

### The Active Measurement Component

To estimate the impact of the prepending method on the incoming traffic distribution, a *BGP-ASPP beacon* is set up as part of the active measurement component. The BGP-ASPP beacon is simply a BGP router that announces a *beacon prefix* inside the AS according to a pre-determined time and prepending schedule. In our setup, the beacon prefix is a /24 subnet that contains only a single host for conducting passive measurements. Before conducting experiments for a new prepending length, we use ICMP echo requests to record the incoming links where packets from the top senders are received. After advertising the beacon prefix with a new prepending length, the incoming links for the set of top senders are examined again. By comparing the two sets of incoming links, we can therefore identify the addresses for which the incoming link is changed for reaching the beacon prefix. This result also applies to other prefixes inside the AS if they are subject to the same routing policies in the upstream ISPs as for the beacon prefix.

Therefore, the active measurement component does not affect the BGP routes for the prefixes inside the AS, except for the beacon prefix, which normally does not expect any incoming traffic from the Internet. As a result, the active measurement incurs a minimal cost on BGP routers, and the routes for the beacon prefix will eventually time out. It is also worthwhile to compare the BGP-ASPP beacon and other BGP beacons used for BGP route convergence and other BGP studies [13, 14]. Our BGP-ASPP beacon, first of all, sends only route updates, but not route withdrawals. Moreover, the BGP-ASPP beacon updates routes only for a limited number of times, not periodically as other BGP beacons do.

We performed active measurement experiments in the test site (Fig. 4a) based on the passive measurement results obtained in August 2002. In all experiments, we performed AS prepending only on link 1 with different prepending lengths: 0 to 5. The case of 0 refers to no prepending. A maximum prepending length of 5 is sufficient, because over 90 percent of active ASs are located less than six AS hops away [7, 12]. In each prepending case, the link receiving the ICMP reply was recorded. Before starting another experiment with a new prepending value, it is important to wait for a sufficiently long period of time for the Internet to include the new AS path attribute. In our case, we waited for at least one day between experiments. The entire procedure is sketched in Fig. 6.

The number of 99 percent top senders involved in the active measurement was 7196. However, not all the senders were responsive to ICMP echo requests, and the total number of responsive top senders was 4770. However, pinging these top senders may still be too intrusive to the normal Internet operation. Thus, we have further reduced the set of *target addresses* by selecting only a single address based on a prefix length of 24. That is, if there are multiple top senders that share the same /24 prefix, we will ping only one of them. The resulting number of target addresses was further reduced to 2746. Clearly, the underlying assumption is that the packets sent from these addresses with the /24 prefix will subject to the same routing policies. If that assumption is not correct, the prediction results to be presented in the next section will be affected.

The active measurement results are presented in Fig. 7a. Without prepending, almost 90 percent of the replies were received through link 1. The results thus reflect that almost all the upstream ISPs preferred paths to link 1 for this set of target addresses. Prepending the path by one or two AS numbers, as the figure shows, did not affect the results

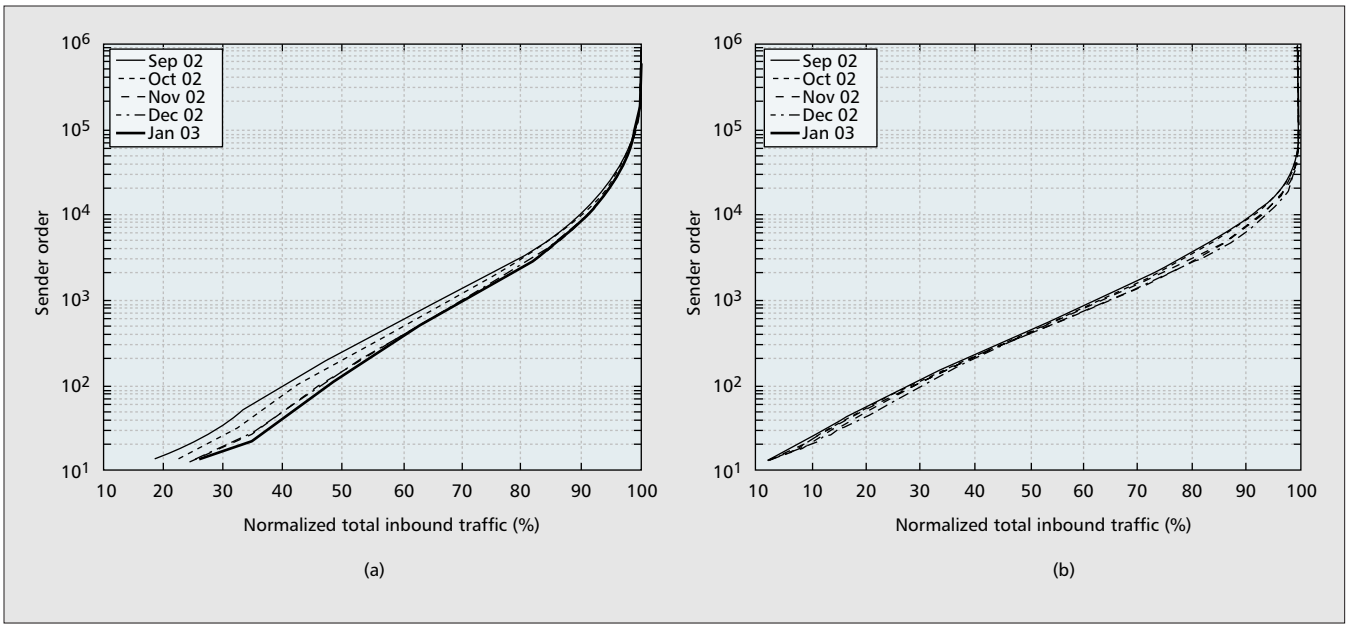



Figure 5. *Passive measurement results: total inbound traffic vs. sender order: a) traffic destined to the test site; b) traffic destined to the dialup network inside the test site.*

significantly—-link 1 still received at least 80 percent of the received replies — in spite of a noticeable downward trend. With an additional AS prepending, i.e., a prepending length of 3, the situation completely reversed. Now link 1 received only 23 percent of the replies. Further prependings did not seem to change the results.

Figure 7b provides more detailed information about the changes in the routes effected by AS path prepending. The bar chart for each case indicates the distribution of changes (no change, from link 1 to link 2, and from link 2 to link 1) when one more AS number was prepended on link 1. Therefore, the graph charts the changes in the two adjacent cases, in contrast to the isolated values as presented in Fig. 7a. The percentages of route changes for 0-to-1 and 1-to-2 prependings were 2.4 and 8.4 percent, respectively. Almost all the changes in both cases were due to the route changes from link 1 to link 2. In the most dramatic case, which was 2-to-3 prepending, nearly 60 percent of the routes were affected and almost all of them were due to the changes from link 1 to link 2. This dramatic change perhaps is not too surprising, because the average AS hop distance of the Internet traffic is around 3 or less [12]. Furthermore, it has been suggested that the manner of multihoming may have an impact on the effect of AS path prepending [8].

## Change Prediction and Experimental Results

With the active measurement results, it is now possible to predict the impact of a change in the AS path length before effecting it. To this end, we propose in this section a simple algorithm to predict the amount of shifted traffic as a result of a new prepending length. It is important to first point out that traffic changes due to AS path prepending can be predicted accurately only when several assumptions hold. The first one is that the set of top senders is quite stable, which has been confirmed from our passive measurements and other measurement studies (e.g., [11]). Second, the routing paths for the flows generated from the top senders are relatively stable. That is, the upstream ISPs' routing policies affecting these traffic flows do not change often, at least on the daily or even weekly basis. Third, the daily traffic rates entering into the network are quite uniform without significant variations (there are exceptions though, such as during denial-of-service attacks and flash crowds).

Same as before, we consider a dual-homed AS and we apply the prepending method only to link 1. Furthermore, it is useful to classify the set of senders that have sent packets to the test site according to Fig. 8a. To aid the discussion, we introduce the following notations; these terms are referenced to a certain period of time, such as a particular month:

- *S*: The set of all senders that have sent packets to the test site

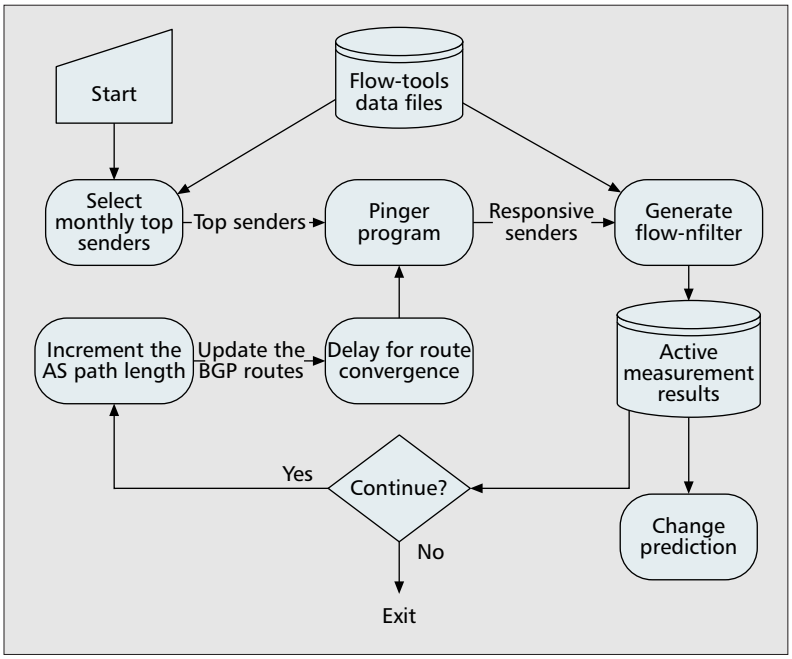



Figure 6. *The active measurement component in* `AutoPrepend`.

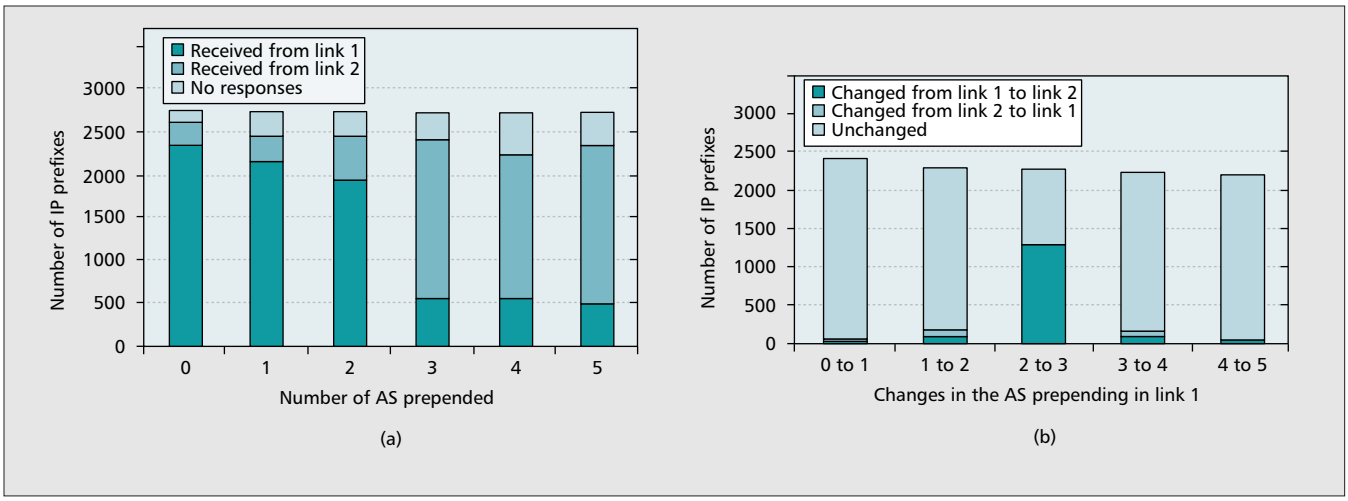



Figure 7. *Active measurement results: a) distribution of the ICMP echo replies when the prepending method is applied to link 1; and b) changes in the receiving links when the prepending method is applied to link 1.*

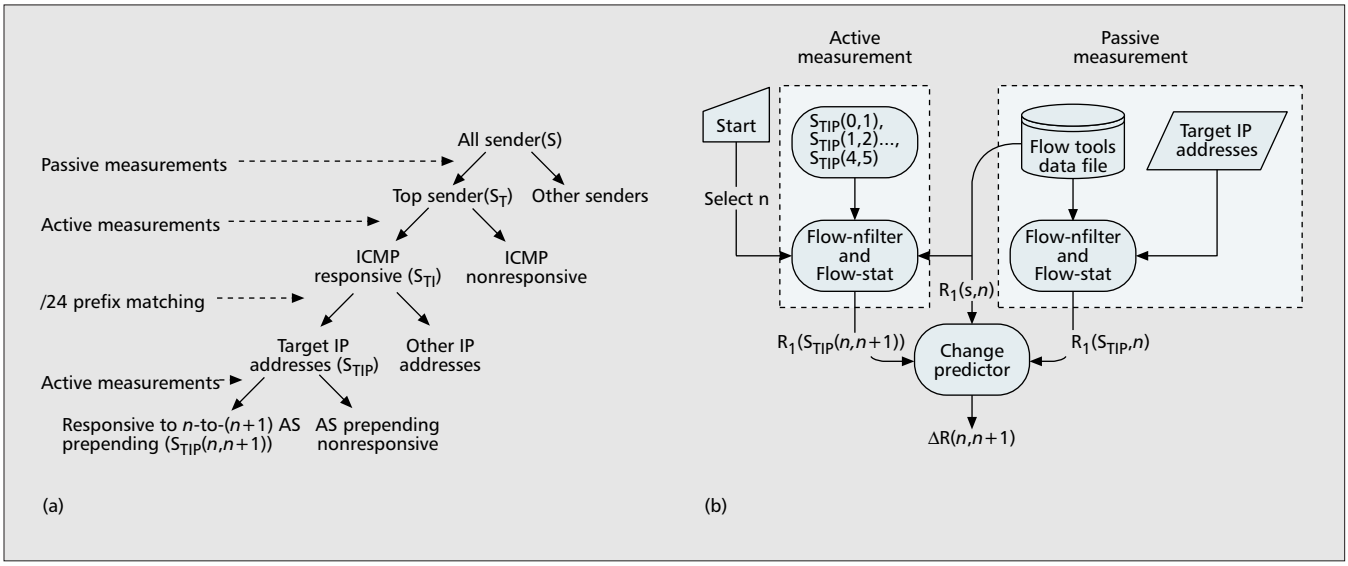



Figure 8. *The traffic prediction component in* `AutoPrepend`*: a) a classification of the senders for the purpose of traffic prediction; b) the traffic prediction system.*

- $S_T$: The set of top senders that have contributed to a certain percentage of the total inbound traffic
- $S_{TI}$: A subset of $S_T$ that responds to ICMP echo requests
- $S_{TIP}$: A set of target addresses, which is a subset of $S_{TI}$ that comprises distinct IP addresses based on a certain prefix length, such as /24
- $S_{TIP}(n, n+1)$, $n \geq 0$: A subset of $S_{TIP}$ for which their traffic will be switched from link 1 to link 2 when the prepending length for link 1 is increased from $n$ to $n+1$

We also use $R_i(S,n)$, $i = 1, 2$, to denote the daily average rate of traffic coming into link $i$ that is generated from senders in set $S$ and when the prepending length is $n$ ($n$ is omitted if it can be uniquely deduced from $S$). $S$ can be $S$, $S_{TIP}$, or $S_{TIP}(n, n+1)$. Moreover, we let $\Delta R(n, n+1)$ be the amount of daily traffic rate shifted from link 1 to link 2 when the prepending length for link 1 is increased from $n$ to $n+1$.

## Change Prediction Computation

We first examine what we have (or have not) known about the possible traffic shift with one more AS prepending based on the active and passive measurement results. First of all, we have identified from the active measurement results $S_{TIP}$ and $S_{TIP}(n, n+1)$. Therefore, based on the daily traffic rates from the passive measurement results, the fraction of the traffic rate from $S_{TIP}$ that is predicted to switch to link 2 is given by

$$\frac{R_1(S_{TIP}(n,n+1))}{R_1(S_{TIP},n)}. \quad (1)$$

However, besides the target addresses, there are other senders contributing to the overall traffic rate flowing into link 1. To cater for them, we trace the tree in Fig. 8a, starting from the left leaf node and going toward the root. If the "other IP addresses" share similar routing policies as the corresponding target address, we can apply the same ratio in Eq. 1 to these senders. Therefore, after taking these senders into account, we have

$$\Delta R(n,n+1) = \frac{R_1(S_{TIP}(n,n+1))}{R_1(S_{TIP},n)} * R_1(S_{TI},n).$$

If we go one more level up, we need to include the top senders that do not respond to ICMP echo requests. In our measurements, these senders comprised almost 40 percent of

all top senders. Thus, any inaccurate prediction for this group would seriously affect the results. Unfortunately, our active measurement system cannot assess their sensitivity to the changes in the AS path length without making the change first. There are two possible ways to solve this problem. The first one is to compare the passive measurement data before and after the additional AS prepending for this set. Since this is an after-fact assessment, this approach can be used only to augment the prediction results. Another approach we have adopted is to assume that the fraction of traffic rate from this set that would be affected by the additional prepending is the same as that for $S_{TIP}(n, n + 1)$. We understand that this is a rather unjustified assumption; however, this is a reasonable assumption to start with. Thus,

$$\Delta R(n,n+1) = \frac{R_1(S_{TIP}(n,n+1))}{R_1(S_{TIP},n)} * R_1(S_T,n).$$

Finally, the contribution to the overall traffic rate from the nontop senders is negligible. Therefore, the predicted traffic change in terms of the daily rate is given by

$$\Delta R(n,n+1) = \frac{R_1(S_{TIP}(n,n+1))}{R_1(S_{TIP},n)} * R_1(S,n). \quad (2)$$

That is, the traffic rate of link 1 is predicted to be decreased by $\Delta R(n, n + 1)$, while link 2's is predicted to be increased by the same amount. The computations for the traffic prediction is illustrated in Fig. 8b.

### Change Prediction Experiments and Results

We have conducted two independent experiments on traffic change prediction in a dialup network of the test site with parameters given in Table 1. As before, we have performed AS path prepending only on link 1. The two experiments were performed in January and March 2003, respectively. For each case, the passive measurement results were conducted on the month prior to the month of making predictions and the actual changes in the AS path length. Both the 99 percent top-sender selection rule and a higher number of senders give a much higher number of top senders for experiment 2. The percentages for top senders responding to ICMP, on the other hand, are both slightly over 60 percent. Another main difference is that experiment 1 adopts a further /24 prefix matching to select target addresses, but experiment 2 does not. There are some obvious trade-offs between the two. The /24 prefix matching will definitely reduce the size of the target addresses, thus minimizing the impact on the Internet's normal operation. However, the side effect is that traffic volume belonging to other senders may also be included in the computation, which could inflate the actual value of $R_1(S_{TIP}(n, n + 1))$. Furthermore, we consider only the case of changing the prepending length from 2 to 3, because this case has been shown to incur the most significant traffic shift from link 1 to link 2.

| | Experiment 1 | Experiment 2 |
|---|---|---|
| Number of senders | 146,347 | 208,535 |
| Selection of top senders | 95% | 99% |
| Number of top senders | 11,224 | 15,467 |
| Number of ICMP responsive top senders | 7055 | 9487 |
| Prefix length for target address selection | 24 | 32 |
| Number of target addresses | 3805 | 9487 |

Table 1. *The parameters for the two traffic change prediction experiments.*

| Incoming traffic (kb/s) | Measured for $n = 2$ | Predicted for $n = 2$ to 3 | Measured for $n = 3$ |
|---|---|---|---|
| Into link 1 | 346.8 | 140.8 | 100.2 |
| Into link 2 | 34.3 | 240.3 | 256.2 |
| Into link 1 and from $S_{TIP}(2,3)$ | 159.2 | – | – |
| Into link 1 and from $S_{TIP}$ | 268.1 | – | – |

Table 2. *The first set of change prediction results for 2-to-3 AS prepending.*

| Incoming traffic (kb/s) | Measured for $n = 2$ | Predicted for $n = 2$ to 3 | Measured for $n = 3$ |
|---|---|---|---|
| Into link 1 | 358.3 | 121.9 | 112.8 |
| Into link 2 | 40.7 | 277.1 | 307.1 |
| Into link 1 and from $S_{TIP}(2,3)$ | 120.8 | – | – |
| Into link 1 and from $S_{TIP}$ | 183.0 | – | – |

Table 3. *The second set of change prediction results for 2-to-3 AS prepending.*

The prediction results for the two experiments are shown in Tables 2 and 3. The predicted traffic rates into links 1 and 2 are computed according to Eq. 2. The measured values for $n = 2$ in the third and fourth rows are for computing the predicted changes. After changing the prepending length to 3, we measured the actual traffic rates and compared the difference with the predicted values before the change. Since one of the main concerns is whether the change will congest the link receiving the shifted traffic, it is more important to evaluate the prediction results in link 2. The prediction error rates are computed by the difference between the two divided by the actual value, and the error rates are 6.2 and 9.8 percent for experiments 1 and 2, respectively. After taking into account the errors arising from normal traffic variation, the predicted values can be considered quite accurate. Nevertheless, it is yet to see whether the same accuracy can be obtained in other sites. Finally, based on the two experimental results, it is still inconclusive as whether the differences in the top-sender selection rule and prefix length have any impact on the prediction results.

Although the experimental results for the dialup network show that the amount of shifted traffic due to the prepending on link 1 is rather significant, there are several ways to exert finer-grained traffic control. The control space depends mainly on the degree of multihoming (i.e., the

number of links) and the number of prefix advertisements. The traffic control granularity clearly increases with a higher degree of multihoming, because prepending can be performed on more than one link, and the prepending lengths can be different. Although we have considered only dual-homing in this article, `AutoPrepend` can be extended easily beyond the dual-homing scenario. Similarly, the traffic control granularity also increases with the number of prefix advertisement, because each prefix can be associated with a different prepending policy. However, it is important to note that it is not necessary to require both a high degree of multihoming and a large number of prefixes in order to have reasonably fine granularity of traffic control. A sufficient condition is to have a large product of degree of multihoming and the number of prefixes. For example, with dual-homing and 20 prefixes, the product is 40, whereas the product is still 40 for a degree of multihoming equal to 4 and 5 prefixes.

## Conclusions

In this article we have proposed AutoPrepend, a complete and automated process based on AS path prepending to engineer traffic coming into an multihomed AS. The entire process consists of four main components: passive measurement, active measurement, traffic change prediction, and AS path update. AutoPrepend offers several important advantages over the current ad hoc tuning of the prepending length. First, it can be readily deployed in other multihomed ASs, because it does not require special hardware and software, and the resource requirement is relatively low. Second, the process has been carefully engineered to minimize unnecessary disruption to the Internet's normal operation. Although the active measurement part is intrusive, the impact has been significantly reduced by using a beacon prefix, identifying the top senders, and a further prefix-based target address selection. Third, it provides a systematic procedure to determine how much prepending is needed and to predict the amount of traffic shift. As a result, the process can avoid possible link congestion and foresee performance impact.

There are several avenues of extending this work. One of them is to combine the AS path prepending with the BGP community attribute. If the upstream ISP supports the community attribute, the AS path prepending can be performed in the upstream ISP instead. Since the upstream ISP is one AS hop closer to the sender, a more fine-grained inbound traffic engineering is possible. Another interesting area is to study the convergence and performance issues when this kind of automated procedure of tuning the prepending length is widely deployed. For example, will the path prepending performed by multiple AS's cause the routes to oscillate among them? Can path prepending be used to distribute the Internet traffic more evenly on the links?

## Acknowledgment

The work described in this article was partially supported by a grant from The Hong Kong Polytechnic University (Project no. G-U055). We also thank the Editor-in-Chief and the anonymous reviewers for their careful reading of the manuscript and comments, which have helped improve the readability of this article.

## References

[1] J. Hawkinson and T. Bates, "Guidelines for Creation, Selection, and Registration of an Autonomous System," RFC 1930, 1996.
[2] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," RFC 1771, Mar. 1995.
[3] J. Stewart, *BGP4: Inter-Domain Routing in the Internet*, Addison-Wesley, 1998.
[4] Route preferences, http://arachne3.juniper.net/techpubs/software/junos42/swconfig-routing42/html/protocols-overview4.html#1045417.
[5] S. Kalyanaraman, "Load Balancing in BGP Environments using Online Simulation and Dynamic NAT," Presented at the Internet Statistic and Metrics Analysis Workshops, http://www.caida.org/outreach/isma/0112/talks/shiv/, 2001.
[6] S. Agarwal, C. Chuah, and R. Katz, "OPCA: Robust Interdomain Policy Routing and Traffic Control," *Proc. IEEE OPENARCH*, Apr. 2003.
[7] N. Feamster, J. Borkenhagen, and J. Rexford, "Controlling the Impact of BGP Policy Changes on IP traffic," *AT&T Res.*, Tech. rep. 011106-02, Nov. 2001.
[8] I. Beijnum, *BGP*, O'Reilly, 2002.
[9] Cisco ISO Netflow, http://www.cisco.com/warp/public/732/Tech/nmp/netflow/index.shtml
[10] Flow-tools, http://www.splintered.net/sw/flow-tools/docs/flow-tools.html
[11] J. Rexford *et al.*, "BGP Routing Stability of Popular Destinations," *Proc. ACM SIGCOMM Internet Measurement Wksp.*, Nov. 2002.
[12] B. Quoitin *et al.*, "Interdomain Traffic Engineering with BGP," *IEEE Commun. Mag.*, vol. 9, no. 3, May 2003.
[13] Z. Mao *et al.*, "BGP Beacons," *Proc. ACM SIGCOMM Internet Measurement Wksp.*, Oct. 2003.
[14] RIPE BGP beacons, http://www.ripe.net/ris/beacon.html

## Biographies

ROCKY K. C. CHANG (csrchang@comp.polyu.edu.hk) received his Ph.D. in computer engineering from Rensselaer Polytechnic Institute in 1990. Immediately after that, he joined the IBM Thomas J. Watson Research Center working on performance analysis and simulation tools till July 1993. He then joined the Department of Computing at The Hong Kong Polytechnic University in 1993, where he is now an associate professor. Most recently, he is leading an Internet Infrastructure and Security research group, working on relative stability of multiple queues, a new generation of denial-of-service attacks and defense, BGP measurement studies, energy saving algorithms, multicast routing protocols, and reliable forwarding mechanisms in P2P networks.

MICHAEL LO (mlo@ouhk.edu.hk) received his B.Eng. in electronic engineering from Hong Kong University of Science and Technology in 1994 and his M.Sc. degree in information technology from The Hong Kong Polytechnic University in 2003. Since 2000 he is a network and system administrator at the Open University of Hong Kong. His main research interests include interdomain routing, traffic engineering, network monitoring, and network security.