Guidelines for Interdomain Traffic Engineering

Nick Feamster
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139
feamster@lcs.mit.edu

Jay Borkenhagen
AT&T IP Services
AT&T Labs
Middletown, NJ 07748
jayb@att.com

Jennifer Rexford
Internet and Networking Systems
AT&T Labs – Research
Florham Park, NJ 07932
jrex@research.att.com

Abstract

Network operators must have control over the flow of traffic into, out of, and across their networks. However, the Border Gateway Protocol (BGP) does not facilitate common traffic engineering tasks, such as balancing load across multiple links to a neighboring AS or directing traffic to a different neighbor. Solving these problems is difficult because the number of possible changes to routing policies is too large to exhaustively test all possibilities, some changes in routing policy can have an unpredictable effect on the flow of traffic, and the BGP decision process implemented by router vendors limits an operator's control over path selection.

We propose *fundamental objectives* for interdomain traffic engineering and specific guidelines for achieving these objectives *within the context of BGP.* Using routing and traffic data from the AT&T backbone we show how certain BGP policy changes can move traffic in a predictable fashion, despite limited knowledge about the routing policies in neighboring AS's. Then, we show how operators can gain greater flexibility by relaxing some steps in the BGP decision process and ensuring that neighboring AS's send consistent advertisements at each peering location. Finally, we show that an operator can manipulate traffic efficiently by changing the routes for a small number of prefixes (or groups of related prefixes) that consistently receive a large amount of traffic.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols–*Routing Protocols*; C.2.3 [**Computer-Communication Networks**]: Network Operations–*Network management, Network monitoring, Public networks*; C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks–*Internet*

General Terms

Measurement, Performance

1. Introduction

Operating a large IP backbone requires continuous attention to the distribution of traffic over the network. Equipment failures and changes in routing policies in neighboring domains can trigger sudden shifts in the flow of traffic. Flash crowds caused by special events and new applications can also cause significant changes in the load on the network. Network failures and traffic fluctuations degrade user performance and lead to inefficient use of network resources. Network operators adapt to changes in the distribution of traffic by adjusting the configuration of the routing protocols running on their routers. Additionally, routing configuration changes are often necessary after deploying new routers and links. Developing effective techniques for adapting routes to the prevailing traffic and topology has been an active area of research and standards activity during recent years [1, 2, 3, 4]. Previous work on traffic engineering has focused predominantly on *Interior* Gateway Protocols (IGPs), such as OSPF, IS-IS, and MPLS, which control the flow of traffic within a single Autonomous System (AS).

In practice, though, most traffic in a large backbone network traverses multiple domains, making interdomain routing an important part of traffic engineering. We motivate the need for interdomain traffic engineering with three examples:

- *Congested edge link:* The links between domains are common points of congestion in the Internet. Upon detecting an overloaded edge link, an operator can change the interdomain paths to direct some of the traffic to a less congested link.
- *Upgraded link capacity:* Operators of large IP backbones frequently install new, higher-bandwidth links between domains. Exploiting the additional capacity may require routing changes that divert traffic traveling via other edge links to the new link.
- *Violation of peering agreement:* An AS pair may have a business arrangement that restricts the amount of traffic they exchange; for example, the outbound and inbound traffic may have to stay within a factor of 1.5. If this ratio is exceeded, an AS may need to direct some traffic to a different neighbor.

The state of the art for interdomain traffic engineering is extremely primitive. The IETF's Traffic Engineering Working Group, which has focused almost exclusively on intradomain traffic engineering, recently noted that interdomain traffic engineering "is usually applied in a trial-and-error fashion. A systematic approach for inter-domain traffic engineering is yet to be devised" [1]. Operators make manual changes in the routing policies without a good understanding of the effects on the flow of traffic or the impact on other domains.

Ultimately, this ad hoc approach to interdomain traffic engineering must evolve into mature, well-tested guidelines and mechanisms. This paper is a first step in that direction. Recent previous work has presented a high-level overview of interdomain traffic engineering [5] and described the traffic data that must be measured to perform interdomain traffic engineering [6]. In addition, several commercial products help large campus and corporate networks balance load over connections to multiple upstream providers [7]; however, these products do not address the challenges of traffic engineering for large ASes in the core of the Internet. Our work is the first to propose *fundamental objectives* for interdomain traffic engineering, as well as specific guidelines for service providers to achieve these objectives within the context of BGP. We argue that the guidelines we present are practical by characterizing traffic and routing data from a large, tier-1 IP backbone.

Neighboring ASes use the Border Gateway Protocol (BGP) to exchange routing information to provide end-to-end connectivity between hosts in different domains [8, 9, 10]. Each BGP advertisement announces reachability to a destination prefix that represents a block of IP addresses. Each advertisement includes a list of the ASes in the path, along with several other attributes. The routers in each AS apply local routing policies that manipulate these attributes to influence the selection of the best route for each destination prefix and to decide whether to propagate this route to neighboring ASes. Operators affect the flow of traffic by tuning the local routing policies that affect the selection of the best path for a destination prefix. Choosing the appropriate configuration is difficult since it depends on the network topology, the IGP parameters, the BGP advertisements from neighboring ASes, and the current traffic patterns. Our work focuses on the impact of BGP policies on the flow of traffic *leaving* an AS at the egress points that connect to neighboring domains. Some traffic engineering tasks necessitate changes to how traffic *enters* the network. However, controlling how traffic enters the network in a predictable way requires coordination with neighboring domains [1]. The results of our analysis of outbound traffic can be applied by the neighboring ASes to influence how traffic enters the network.

Interdomain traffic engineering is significantly more complicated than intradomain traffic engineering. While IGPs select paths based on link metrics, such as static weights or dynamic load information, BGP advertisements do not explicitly convey any information about the resources available on a path. BGP routing policies are complex and depend on a variety of factors, such as the commercial relationships with neighboring ASes [11]. The selection of the best path for each prefix depends not only on the routing policies but also on the advertisements sent by neighboring domains. Operators have, at best, indirect influence on BGP path selection. In fact, changing the BGP policy in one AS may alter the advertisements propagated to neighboring domains, which may inadvertently affect how traffic enters the AS. The constraints that BGP imposes on making "good" routing decisions makes moving to a radically different interdomain routing paradigm desirable, but extremely difficult in practice. Rather than proposing a new routing protocol, our analysis identifies ways to support traffic engineering within the existing BGP framework.

Router vendors support a wide variety of configuration commands that provide significant flexibility in specifying BGP policies. Selecting the right policy changes for a particular traffic-engineering task is challenging, especially for service providers that have many connections to neighboring domains. Our study focuses on developing traffic engineering techniques that achieve the following objectives:

- *Achieving predictable traffic flow changes:* Some routing changes have effects that are difficult to predict in advance, due to the routing policies in other domains. Our analysis identifies approaches for tuning policies in ways that have predictable outcomes and limit the changes seen by neighboring domains.

- *Limiting the influence of neighboring domains:* Certain practices, such as sending inconsistent advertisements at different peering locations, can have a significant impact on the path selection process. Our analysis shows how operators can check for these practices and use BGP policies that limit their effects.

- *Reducing the overhead of routing changes:* Changing the routing policy may trigger new advertisements that impose a load on the routers and a delay for converging to a new set of routes. Our analysis shows that operators can limit overhead by focusing on the small number of prefixes (or groups of prefixes) that consistently receive a large amount of traffic.

Although this paper primarily describes how to achieve these objectives within the context of BGP, these objectives are applicable to interdomain traffic engineering *in general*. We discuss our results for these three objectives after a brief background section on the BGP protocol and traffic engineering tools and an overview of our measurement data from AT&T's IP backbone.

2. BGP Traffic Engineering

This section presents an overview of BGP and the attributes associated with route advertisements. We briefly describe tools that could allow operators to adjust the routing configuration to the prevailing traffic.

2.1 Border Gateway Protocol

Internet routing operates at the level of address blocks, or prefixes. Each prefix consists of a 32-bit address and a mask length; for example, 192.0.2.0/24 represents the 256 addresses ranging from 192.0.2.0 to 192.0.2.255. An IP router constructs a forwarding table that is used to select the output interface for each incoming packet, based on the longest-matching prefix for that destination address. Routers in different ASes use BGP to exchange update messages about how to reach different destination prefixes. A router sends an *announcement* to notify its neighbor of a new route to the destination prefix and sends a *withdrawal* to revoke the route when it is no longer available. Each advertisement includes a number of attributes about the route, including the list of ASes along the path to the destination prefix. Before accepting an advertisement, the receiving router checks for the presence of its own AS number in the AS path to detect and remove routing loops.

A router may receive routes for the same prefix from multiple neighboring ASes. The router applies *import policies* to filter unwanted routes and to manipulate the attributes of the remaining routes. Ultimately, the router invokes a *decision process* to select exactly one "best" route for each destination prefix among all the routes it hears. The router then applies *export policies* to manipulate attributes and decide whether to advertise the route to neighboring ASes. In addition to exchanging BGP messages with neighboring domains, an AS may use internal BGP (iBGP) to distribute routing information among its routers. Ultimately, every router must select a single best route for each prefix among the advertisements from the various external BGP (eBGP) and iBGP neighbors.

BGP advertisements can include numerous attributes [9], and the BGP decision process implemented by router vendors has several steps, which proceed in order and sequentially eliminate candidates for the best route [12, 13, 14]. To simplify the discussion, we focus on five main steps in the selection process:

1. *Highest local preference*: Prefer routes with the highest local preference, assigned by the import policy and conveyed to other routers via iBGP.

2. *Shortest AS path*: Prefer routes with the shortest AS path length, as conveyed in the BGP advertisement.

3. *eBGP over iBGP*: Prefer routes learned via eBGP over routes learned via iBGP, since leaving the AS directly is preferable to traveling through the AS.

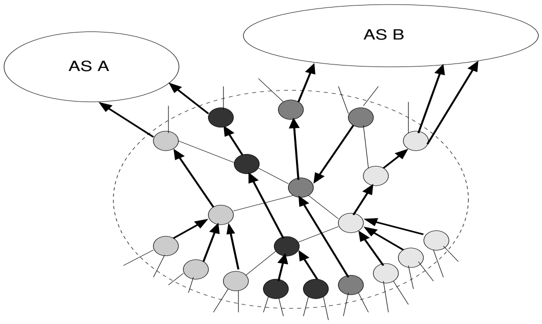

Figure 1: Flow of traffic from ingress routers to the egress links. Each node represents a router within the AS. Routers with the same shading have the same closest egress point.

4. *Lowest IGP metric*: Prefer routes with the smallest intradomain (IGP) metric to reach the next hop. This enables each router to select its "closest" exit point.
5. *Lowest router ID*: Prefer the route learned from a router with the lowest identifier, as conveyed during BGP session establishment. This step breaks ties between routes that are equally good after the previous steps have been applied.

We primarily focus on how operators can assign local preference to influence the first step of the BGP decision process; router configuration languages provide operators with flexibility in assigning local preference based on the destination prefix, the AS path, and other BGP attributes. Our observations also apply to other BGP attributes, such as the origin type and the multiple-exit discriminator (MED), as discussed in the Appendix.

2.2 Traffic Engineering Tools

The construction of the forwarding table at each router depends on the complex interaction of BGP routing policies, the distribution of update messages via iBGP, and the IGP parameters. Over time, each router receives eBGP messages from neighboring domains and iBGP messages that report the best routes seen at other routers in the AS. The routers also participate in an IGP that affects their selection of the best path, as well as the route through the domain to reach the BGP next hop. Figure 1 shows a collection of routers that select different routes toward a destination prefix reachable via ASes A and B. Each router selects a route with the "closest" egress point, based on the IGP weights. Modeling the impact of interdomain routing on the flow of traffic in the network requires a way to separate the roles of BGP policies and IGP parameters in the construction of the forwarding table. It also requires a way to capture how the asynchronous exchange of eBGP and iBGP messages affects the selection of the best path at each router.

Operators can use tools to predict the influence of changes to the BGP policies and IGP weights on the flow of traffic[1], as shown in Figure 2. The first module [17] captures the first three steps of the BGP decision process that do not depend on the IGP weights. For each prefix, this produces a set of egress points, where the final selection of the closest egress point may vary at different routers inside the AS, as shown in Figure 1. The second module [4] captures the selection of the closest egress point, based on the IGP cost and the router ID tie-break (steps 4-5) for each router in the domain; this module also identifies the IGP path(s) associated with the minimum cost. Together, the two tools predict the how traffic would flow through the AS for each ingress point and destination prefix. By combining this information with traffic measurements from the ingress points [18], the tools can predict how a change in routing configuration would influence the load on each link in the domain.

However, to use these tools effectively, operators must first be able to identify good candidate changes to the routing configuration. BGP is a policy-based routing protocol that provides substantial flexibility in matching and assigning the attributes in the advertisement messages. This is important for two main reasons. First, the search space of changes to BGP policies and IGP weights is extremely large—far too large to explore exhaustively. Second, BGP permits operators to make ineffectual or even harmful changes in an attempt to shift traffic from one path to another. Making these kinds of changes in the operational network can cause significant degradation in user performance, and trigger unnecessary routing updates throughout the Internet. Experiments with routing changes should be conducted outside of the network, using accurate tools to predict the effects. Still, it is important to avoid spending valuable time exploring innumerable changes to BGP policy in the tool. In this paper, we identify effective and efficient ways to tune the BGP import policies for traffic engineering.

3. Measurement Data

Effective traffic engineering requires an understanding of the network paths and traffic volume associated with each destination prefix. This section describes the collection of BGP routing tables and flow-level traffic measurements from the routers that connect the AT&T backbone to other large providers.

3.1 BGP Routing Tables

Ideally, the operator would have a complete, up-to-date snapshot of all of the BGP updates heard from eBGP neighbors, which would enable the operator to precisely determine how a change in import policies would affect the routing decision made by each router. However, acquiring a timely view of all of the BGP update messages in the network may be difficult. Ideally, IP routers would be able to provide a continuous feed of all of the routes (both best and alternate paths) as they arrive, but this feature is not universally available. An alternate approach is to extract the set of paths from the BGP routing table (the Routing Information Base) from each router at the edge of the network. A simple script can connect to each router and issue a command to dump the current routing table (e.g., "show ip bgp" in Cisco IOS). Figure 3 shows an example line in a BGP routing table. The entry lists a single route for prefix 38.138.55.0/24 that was learned via iBGP (the "i" before the prefix) and has a next-hop IP address of 192.168.0.10. The routing table entry includes other attributes such as the multiple-exit discriminator (MED) value (2130), local preference (100), AS path (1 701 17031), and the origin type ("i" for IGP). The ">" symbol indicates that this is the router's "best" route for this prefix.

Using routing tables to extract all paths to a prefix imposes two limitations on the quality of the data. The first limitation concerns the *consistency* of the data. Dumping the entire routing table imposes a load on the router, making it impractical to collect these tables very frequently. In fact, since routing table dumps do not occur instantaneously, the state of the table may change during the dump itself; most router implementations avoid this problem by defer-

[1] The use of these tools rests on the assumption that the inputs are relatively stable. The operator controls the import policies and the IGP weights, and topology changes occur only in response to unexpected failures and planned maintenance/upgrades. Although the BGP updates from other ASes change over time, the BGP routes for most prefixes stay the same for weeks at a time [15]; the BGP routes for the most popular prefixes are especially stable [16]. In addition, we envision that operators would not need to change BGP policies all that frequently—only in response to significant changes in the topology or traffic demands.

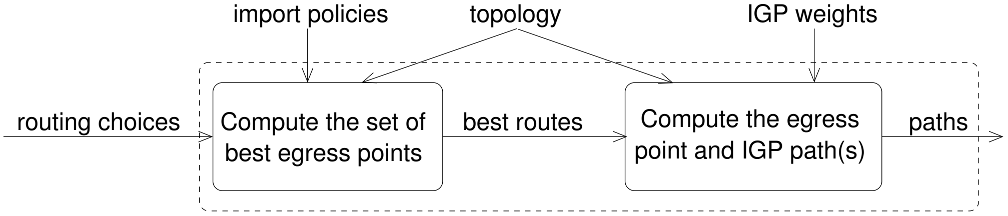
import policies
topology
IGP weights
routing choices
Compute the set of best egress points
best routes
Compute the egress point and IGP path(s)
paths

Figure 2: Predicting the impact of BGP policies and IGP weights on the flow of traffic.

```
   Network          Next Hop            Metric LocPrf Weight Path
*>i38.138.55.0/24   192.168.0.10          2130    100      0 1 701 17031 i
```

Figure 3: Example BGP routing table entry for prefix 38.138.55.0/24

ring changes in the routing table until the dump is complete. Table dumps may not occur at exactly the same time across all routers, thus causing occasional inconsistencies in the network-wide view of the routing choices. The significance of these issues depends on how often routing changes occur relative to the frequency of the routing table dumps. Given that many routes are stable for days or weeks at a time [15, 16], these types of inconsistencies are likely uncommon. However, a live feed of BGP updates from each router would provide precise information about the routes available to each router at a particular time and would eliminate this concern entirely.

Relying on routing tables can adversely affect the *completeness* of the routing information. The routing table represents the collection of routes *after* the import policies have been applied. Hence, the table does not include any routes filtered by the import policy. Since we do not try to model changes in the filtering policy, this is not a significant limitation. Each routing table entry includes attributes such as local preference, origin type, and MED *after* manipulation by the existing import policy. This does not preclude experimenting with different import policies that change the assignment of local preference or origin type, or that reset the MED value. Finally, routing table entries such as the one shown in Figure 3 do not include the community values included in the BGP advertisement. As such, these BGP tables are not useful for experimenting with import policies based on communities. Despite these shortcomings, the routing table data is sufficient for evaluating policies that set local preference based on the prefix and AS path.

We collected BGP routing tables from routers that connect the AT&T backbone to other large providers and extracted the routes for each prefix. We focused on the routes learned via eBGP and ignored the routes that were propagated from other routers via iBGP. To focus on routes that traverse the peering links, we excluded prefixes that are reached directly by connections to customers of the AT&T backbone. Suppose a prefix has routes learned from both customers and peers. If the customer route has a high local preference, then we do not include any of the routes for this prefix in our analysis, since traffic to this prefix should travel via the customer link(s) rather than peering links. On the other hand, if the customer route has a low local preference (indicative of a backup route), then we include the routes learned from peers, since traffic to this prefix should travel via peering links rather than customer links.

3.2 Flow-Level Traffic Measurements

The influence of changes in BGP import policies depends on the amount of traffic that moves to new routes. Our analysis draws on daily summaries of the traffic leaving the AT&T backbone via peering links. The data were collected using Cisco's Netflow feature [19]. Netflow produces a single measurement record for each "flow"—a group of packets that match in key IP and TCP/UDP header fields and appear close together in time. Each Netflow record includes the start and finish time of the flow, the number of bytes and packets, the source and destination IP addresses, the mask length for the longest-matching prefix in the forwarding table, the TCP/UDP port numbers, and several other fields. The routers that connect AT&T to other large providers are configured to run Sampled Netflow [20], which performs one-out-of-N sampling of the packets before constructing the flows. This reduces the packet handling overhead and the number of flow records, at the expense of a reduction in accuracy.

The routers in each Point-of-Presence (PoP) were configured to send the measurement records to a dedicated collection machine. Each collection server was configured to aggregate the flow-level records to compute the volume of traffic for each destination prefix on an hourly time scale. Each flow-level record was associated with a destination prefix based on the destination IP address and the mask length. The collection server was configured to aggregate the measurement records separately for inbound and outbound traffic. Each Netflow record includes identifiers for the input and output links that carried the traffic for the packets in the flow. These links can be classified as edge and core links, based on a snapshot of the network topology. Outbound traffic travels from a core link to an edge link, whereas inbound traffic travels in the opposite direction.

The collection server corrected for the influence of one-out-of-N sampling at the router by multiplying the resulting traffic volumes by N. In addition, the collection server applied stratified sampling to reduce the processing overhead [21]. This sampling scheme focuses on a subset of the records based of the number of bytes associated with the flow. Records for large flows are always included in the aggregation. Smaller flows are included with a probability proportional to their size; the aggregation applies an appropriate correction factor to account for the effects of sampling. Together, the two forms of sampling make it possible to collect and analyze measurement data on a large number of high-speed links. Adjusting for the effects of sampling produces an unbiased estimator of the volume of traffic destined to each prefix. The estimates have very low variance, except for destination prefixes that receive an extremely low volume of traffic. In the next three sections, we analyze *daily* totals of outbound traffic volumes. We also avoid drawing conclusions about the amount of traffic associated with prefixes that have low (and, thus, potentially inaccurate) traffic volumes.

The Netflow measurements were collected throughout the day on March 1, 2002 and the BGP routing tables were dumped at approx-

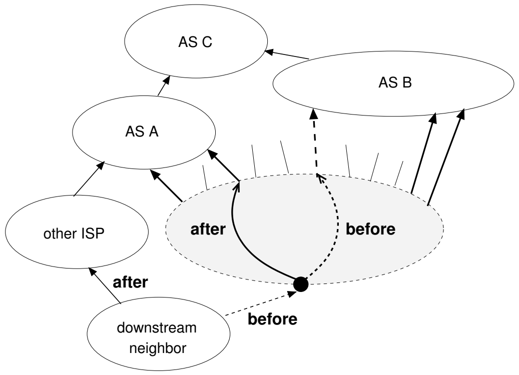

Figure 4: Neighbor's behavior upon receiving a new route.

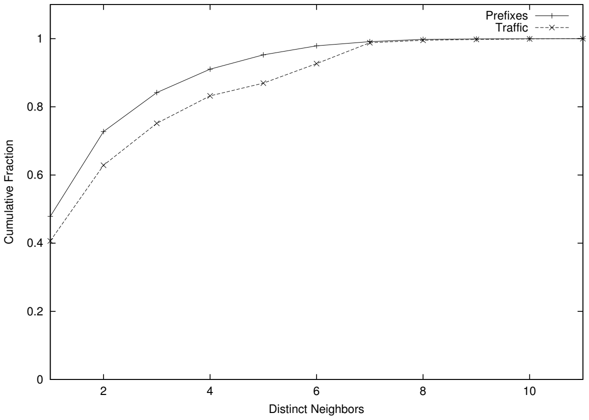

Figure 5: Cumulative distribution of the number of next-hop ASes among the shortest AS paths for a prefix. Prefixes that have advertisements from only one next-hop are ideal candidates for BGP traffic engineering, since policy changes can be made without changing inter-AS traffic flow.

imately 2 a.m. EST on the same day. Additionally, we collected the same data on April 1, 2002 to verify the results of the analysis. We parsed and preprocessed the data and stored the results in a MySQL database. One database table stores the set of eBGP-learned routes for each destination prefix. Each entry in this table includes the date of the BGP dump, the associated router, the advertised prefix, the AS path, and whether or not this route was a "best" route for that destination prefix. A second table stores daily summaries of the outbound traffic. Each entry in this table includes the date when the measurements were collected, the associated router, the destination prefix, and the number of bytes sent outbound to the destination prefix via that router.

4. Achieving Predictable Traffic Flow Changes

Effective traffic engineering relies on policy changes that have a *predictable* influence on the flow of traffic through the network, which is inherently difficult for two main reasons. First, modifying the import policies may cause the AS to change its choices for best routes, and thus send new routing advertisements *to* neighboring domains, which may in turn affect where and whether traffic enters the AS from these neighbors. Second, small changes in the advertisements sent *by* neighboring domains may cause unintended changes in the selection of the best routes for a destination prefix. In this section, we show that careful modification of import policies can control these effects and thus improve predictability of changes to the flow of traffic.

4.1 Avoid Globally-Visible Changes

When adjusting routing policies, operators should minimize the impact on the behavior of downstream neighbors. If a policy change causes neighboring domains to change their behavior (e.g., by selecting a different best route for a prefix), the amount of traffic entering the AS from these neighbors may be unpredictable. Suppose that a particular edge link is congested and the network operator assigns a lower local preference value to some of the routes traversing the congested link. The new import policy will remove these routes from the set of possible best routes for these prefixes, thus causing some routers to direct traffic for these destination prefixes to a different route via a different egress link. Moving the traffic reduces the load on the congested link. However, the affected routers might advertise a new route to their eBGP neighbors, such as downstream customers, potentially causing significant changes in the volume of inbound traffic.

In the example shown in Figure 4, ASes A and B both advertise paths to destinations in AS C. Initially, there are five "best" routes—two via AS A and three via AS B. Routers on the west coast route via AS A and routers on the east coast route via AS B. Suppose that the leftmost link to AS B is congested (as illustrated by the dashed line), and the import policy for this egress point is modified to assign a lower local preference to routes originating from AS C. After this change, some routers might switch from a route via the leftmost link to AS B to a route via the rightmost link to AS A. These routers would advertise the new best path to downstream neighbors. Depending on the neighbor's routing policies, the new advertisement might cause the neighbor to select a different next-hop AS (i.e., another ISP) for reaching this prefix. This could result in an unpredictable decrease in the volume of traffic entering the domain at this router. Similarly, the routing change could trigger an increase in traffic if other neighbors preferred the (A, C) route over the (B, C) route.

To prevent the effects of routing changes from propagating to neighboring domains, a network operator should only adjust routing policies for prefixes for which every potential best route has the same BGP attributes (except for the next-hop IP address, of course). This approach still gives an operator significant flexibility, because the operator can route traffic for that destination via *any subset* of these advertised routes without affecting the BGP advertisements seen by neighboring ASes. Depending on the BGP implementation, downstream ASes may not even receive a new BGP advertisement, since none of the attributes conveyed to eBGP neighbors has changed (this feature is called *non-transitive attribute filtering*).

For our data, 47.8% of the prefixes have shortest AS paths with a single next-hop AS, as shown in Figure 5; these prefixes contribute over 40% of the outbound traffic. For these prefixes, reducing the local preference at one peering location would shift traffic to another egress link to the *same* peer. In some cases, an operator may need to move traffic from one next-hop AS to another. As shown in Figure 5, a reasonable fraction of prefixes and traffic have shortest paths with two next-hop ASes (e.g., if these two ASes share a common, multi-homed customer like AS C). This is useful for moving traffic between two neighboring ASes without having to select routes with different AS path lengths. (The network may have routes to two ASes via the *same egress router*. In this case, it is possible to move traffic between egress links without changing the traffic flow *within* the AS.) Although this type of routing change requires sending a new advertisement to some downstream ASes,

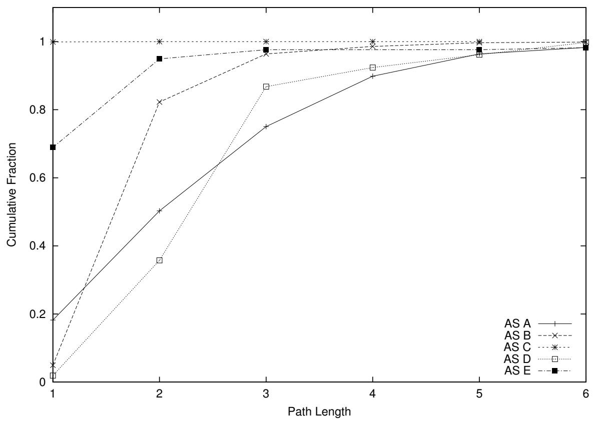

Figure 6: Cumulative fraction of outbound traffic vs. AS path length for various next-hop ASes. Assigning a lower local preference to all traffic destined for certain AS and path length can have vastly different effects depending on which ASes advertisements are affected.

advertising a route with the *same AS path length* reduces the likelihood that a downstream AS selects a best path through a different provider.

4.2 Limit Reaction to Minor Changes

Router configuration languages provide significant flexibility in assigning local preference values to routes. For example, these languages allow an operator to assign local preference based on the destination prefix or regular expressions on the AS path. However, the import policy has only an *indirect* affect on the path selection process. Changes in the advertisements sent by neighboring domains may cause the existing import policies to assign a different local preference value and shift traffic to or from a particular edge link. For example, suppose a neighboring domain D advertises a three-hop AS path "D B C" to reach a particular destination, and then later changes to the path "D A C"; this may occur for traffic engineering reasons, similar to the example shown in Figure 4. An import policy that sets local preference based a specific AS path "D B C" would assign a different value for a route with the path "D A C", which may cause an unintended shift in the traffic associated with the destination prefix.

Network operators can design import policies that are robust to small changes in BGP advertisements by avoiding policies that make such fine-grain distinctions between different AS paths. For example, suppose a network has several high-bandwidth links to AS D and one low-bandwidth link. Then, the import policy for the low-bandwidth link could be configured to assign a lower local preference value to certain routes based just on the origin AS (e.g., C) or even the AS path length. For example, the import policy could assign a small local preference to all destination prefixes with three-hop AS paths. This would divert traffic for destination prefixes with a three-hop (shortest) AS path to other egress points that have shortest paths with three AS hops. This approach is simple and does not depend on the exact sequence of ASes in the path. However, the specific effects of this technique depend on how traffic is distributed over different lengths of AS paths. This may vary across different next-hop ASes.

A network operator might expect to see differences in the distribution of traffic over AS path lengths and may have to consider per-AS traffic patterns when designing policies that are based on AS path length. Figure 6 shows the cumulative distribution of outbound traffic carried by best paths of different lengths. Each curve corresponds to the traffic traversing a different next-hop AS, identified by A, B, C, D, and E; for example, nearly all traffic to AS C follows a one-hop path, and nearly 70% of the outbound traffic to AS E travels over a one-hop AS path (where AS E is the next-hop AS). In contrast, the majority of traffic traveling via the other three ASes travels on AS paths of length two or three. These differences stem from the various roles ASes in the Internet can play, as well as historical and network-specific artifacts (e.g., a single ISP network might consist of multiple ASes). In some cases, an AS hosts a large number of services and directly-connected customers that do not have their own AS numbers. This type of network sends traffic over paths with a single AS hop, as shown in the plot for AS E. In other cases, an AS is a transit provider for a large number of tier-2 providers or multi-homed institutions. Outbound traffic to these types of networks is likely to traverse paths of different lengths, as shown in the plots for ASes A, B, and D.

5. Limiting the Influence of Neighboring ASes

The routing choices for each prefix depend on the routing advertisements heard from neighboring domains. The common practice of AS prepending (i.e., repeating an element in the AS path before readvertising to make the path appear longer) limits the ability to spread traffic over a large number of egress points in different parts of the network. In addition, although BGP import policies can reassign some attributes (such as origin type and MED), other attributes, such as the AS path, depend on the policies applied in other ASes. Inconsistencies in the routes advertised via different eBGP sessions with the same next-hop AS can reduce an operator's control over traffic flow. In this section, we quantify these effects and suggest techniques for increasing control over the flow of outbound traffic.

5.1 Limiting the Influence of AS Path Length

Even if advertisements are consistent across eBGP sessions to the same next-hop AS, path length has a significant influence on the comparison of routes via different ASes. AS prepending increases the length of the AS path by repeating an AS number multiple times to artificially make a path look longer. Consider an AS 100 that connects to AS 200 and AS 300, as in Figure 7. AS 100 may send a one-hop route to AS 200 and a two-hop route to AS 300 to encourage traffic destined to AS 100 to traverse a route via AS 200. An AS that connects to these two ASes would receive routes $(200, 100)$ and $(300, 100, 100)$, perhaps at different locations in the network. If both routes are assigned the same local preference, the AS would direct all of the traffic to the $(200, 100)$ paths. Alternatively, the operator could assign lower local preference to the $(200, 100)$ path, which would force all of the traffic to use the $(300, 100, 100)$ path. Using both paths (via different egress points in the network) is not possible in general.

We investigated the frequency of AS prepending in the BGP routing data from the AT&T network. Approximately 32% of the routes in the BGP tables had some amount of AS prepending. Figure 9 shows the distribution of the amount of prepending in these paths. The majority of the paths were extended by one or two hops; four paths were extended by as many as 16 hops. AS path prepending contributes to the diversity of AS path lengths, as shown by the cumulative distribution plot in Figure 10. The majority of prefixes have AS paths of a single length, and the majority of traffic is associated with these prefixes. However, about 40% of the prefixes have paths with different lengths. Most of these prefixes have paths with just two or three unique lengths. The different lengths stem

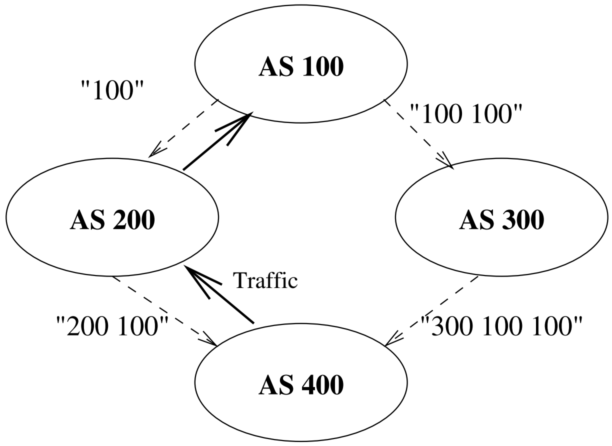

Figure 7: Example of AS path prepending. AS 100 can make a path look arbitrarily longer to downstream networks (e.g., AS 400) by prepending its AS to the path one or more times.

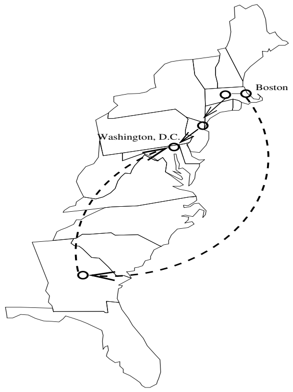

Figure 8: Shortest AS path length does not always reflect shortest network distance. A shorter path from Boston to Washington, D.C. that traverses two intermediate ASes on the way may be shorter than a path with one intermediate AS that that does not have a geographically proximal exchange point.

from a mixture of AS prepending and routes with a different number of unique ASes in the path. In either case, the different lengths limit flexibility in selecting a set of best routes, since the second step in the BGP decision process forces all best paths to have the same length.

While small differences in AS path length restrict routing choices significantly, they are also not often indicative of the best route to a particular prefix. As shown in Figure 8, a path from Boston to Washington, D.C. that crosses two intermediate networks with conveniently-located exchange points is preferable to a path that has fewer AS hops, but requires the packets to travel to a distant exchange point[2]. Similarly, a path with fewer AS hops may traverse a network that is experiencing high latency or loss or contains many intra-AS router hops. Forcing all best paths to have the same AS path length may be unnecessarily restrictive. Figure 6 shows that, for many ASes, the majority of traffic travels over shortest AS paths

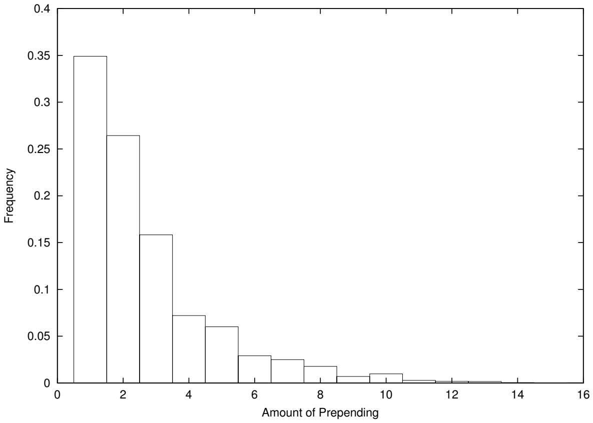

Figure 9: Frequency of AS prepending of different lengths for the 32% of all advertised routes that include some amount of prepending. Twelve advertised paths were extended by at least 14 hops.

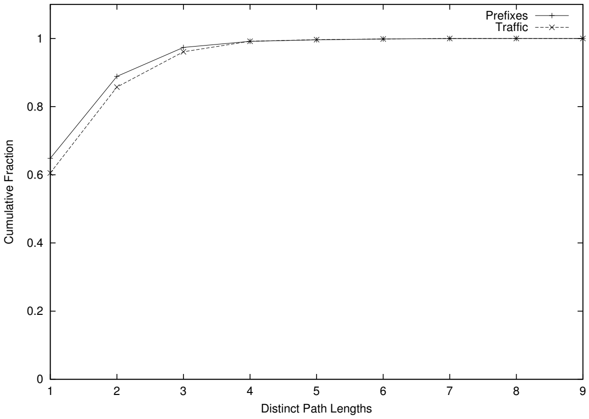

Figure 10: Cumulative distribution of the number of unique AS path lengths. Prefixes that have multiple AS path lengths limit flexibility in selecting a set of best routes.

of length 2 or 3. Furthermore, almost no traffic traverses AS paths of length 4 or longer.

Consequently, it may be effective to allow the set of best paths to include AS paths with small differences in length (e.g., having one 2-hop path and one 3-hop path to a prefix, rather than allowing only the 2-hop path). Coarse-grained AS path length categorization can be achieved by *disabling* step 2 of the BGP decision process and instead assigning local preference ranges based in part on AS path length. For example, a network operator could assign a range of local preference values to one-hop paths, another range to paths of length 2 or 3, and so on. This ensures that AS path length has an influence on the decision process without imposing the strict requirement that all best paths for a prefix must have the same length.

5.2 Consistent Advertisements from Neighbors

BGP update messages from neighboring ASes have a significant impact on the flow of traffic through a network. A neighbor AS can exert influence on how traffic leaves a network by sending inconsistent routing advertisements over different eBGP sessions. For example, suppose that a network connects to AS A at locations on the east and west coast. If AS A advertises a prefix only on the east

[2] Network operators in Europe face these challenges continually. These operators typically tag transatlantic routes with a particular community value and assign a different local preference value accordingly.

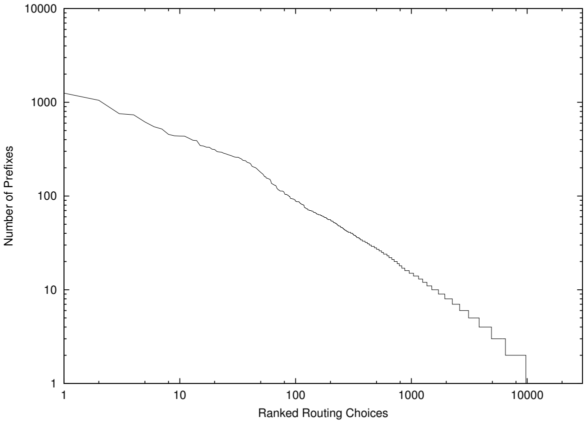

Figure 11: Distribution of the number of prefixes with the same routing choices (log-log scale).

coast, then this would force the other network to carry all of the outbound traffic for this prefix to the east coast. Alternatively, AS A might advertise the path with a different AS path length or origin type at different locations. Advertising inconsistent routes can have a significant and unpredictable influence on the flow of traffic by limiting the number of possible egress points; in addition, this practice is often a violation of peering agreements.

We analyzed the routes in the BGP tables to identify paths of different AS path lengths from the same next-hop AS for the same destination prefix. All but two peers, which local routing policy indicated were special cases, advertised consistent AS path lengths for more than 99% of advertised prefixes. However, our preliminary evaluation shows many instances where a peer advertises a prefix at some peering points but not others. Some inconsistencies can likely be explained by the asynchrony in downloading the BGP tables from the routers; in our ongoing work, we are trying to better understand the nature of these inconsistencies.

6. Reducing the Overhead of Routing Changes

Traffic engineering involves moving a portion of the traffic in the network from one link to another. The BGP import policies can select this traffic based on the prefixes and the attributes in the route advertisements. An operator could conceivably configure import policies to manage traffic on a per-prefix basis. In this section, we first argue that simpler import policies that focus on *groups* of related prefixes, such as prefixes with the same routing choices or the same origin AS, can achieve traffic engineering goals with a relatively small number of policy changes. Then, we argue that import policies should focus on the routes to *popular* destinations to move large amounts of traffic with a small number of routing changes. Finally, we discuss how operators can focus on prefixes (or groups of prefixes) with *stable* traffic volumes over time.

These three techniques reduce the overhead of routing changes in several ways. Moving groups of prefixes that carry more traffic eases management overhead by reducing the number of changes that an operator must make to achieve a task, and setting policies based on groups of prefixes with stable traffic volumes reduces the likelihood that an operator will have to be constantly adjusting routing policies to achieve a certain traffic engineering task.

6.1 Group Related Prefixes

Because a typical default-free BGP routing table contains routes for more than 100,000 prefixes, exploring all possible combinations of import policies is computationally intractable. Furthermore, import policies that are tailored to every prefix at every router would be extremely complicated to configure and expensive for the router to apply. Such fine-tuned policies might not remain appropriate following a shift in traffic or a change in the neighbors' routing advertisements.

Many prefixes have the same attributes across all eBGP advertisements from neighboring domains (the *routing choices* in Figure 2). For example, a single institution, such as a company or university campus, may announce a dozen different destination prefixes from a single location. These prefixes tend to have identical routes in a BGP table at an arbitrary point in the Internet[3]. Because many prefix advertisements have the same characteristics, a network operator can effect policy changes for a significant number of prefixes simply by changing policies based on characteristics in the routing advertisements (e.g., AS path properties), rather than on the specific prefix.

To identify groups of related prefixes, we propose a canonical representation of the *routing choices* announced by neighboring domains. Most of the steps in the BGP decision process depend on the import policy or IGP weights, except for the step based on AS path length. We classify each prefix based on the routers where the routes for that prefix were learned, as well as the AS paths that were learned for each prefix. If several prefixes are advertised to the same set of routers and, at each router, the routes for those prefixes have the same AS paths, we say that those prefixes have the same *routing choices*. This concept facilitates comparisons between different destination prefixes and can be useful for predicting the impact of changes in import policy, since many computations can be performed once per group of prefixes.

In our data, we find a total of $20,086$ unique representations of routing choices. Figure 11 shows the distribution of the number of prefixes associated with each set of routing choices, starting with the set with the largest number of prefixes. A set of routing choices is associated with five destination prefixes on average. However, in some cases, many more prefixes are associated with a particular routing choice. $2,142$ destination prefixes had exactly the same set of routing choices, and 88 sets of routing choices were associated with 100 or more prefixes. Because many prefixes have the same routing choices, a network operator can affect the routes for a large group of prefixes by selecting import policies based on the attributes in the routing advertisements, rather than on each specific prefix. For example, a network operator can manipulate the traffic for a group of prefixes by assigning local preference to these advertisements based on their common attributes, such as AS path characteristics.

6.2 Focus on Popular Destinations

Defining independent import policies even for $20,000$ unique routing choices is still an unreasonable requirement. Fortunately, the bulk of the traffic is concentrated in a small fraction of routing choices. The bottom curve in Figure 12 shows the cumulative distribution of the proportion of traffic destined to most popular prefixes. For example, traffic destined for the top 1% of the prefixes is responsible for about 20% of the outbound traffic volume. The top 10% of prefixes accounts for approximately 70% of the traffic. These results are consistent with the trends seen in earlier traffic measurement studies [18, 24, 25]. The results are more dramatic

[3] Previous work has made similar observations [22, 23]. However, this work did not consider the *volume* of traffic associated with these groups of prefixes, and focused on grouping the routes from a *single* BGP routing table, rather than constructing an *AS-wide* view of routing choices across multiple edge routers.

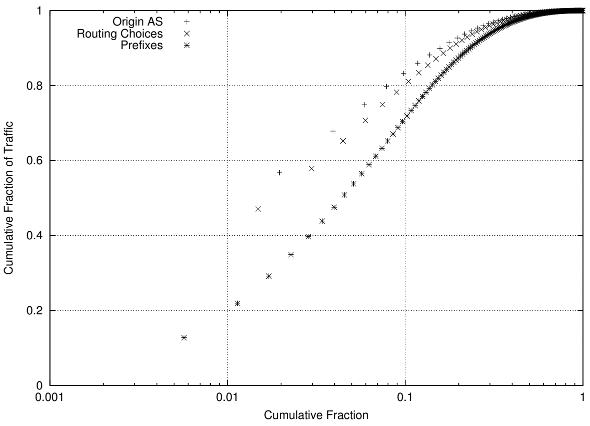

Figure 12: Cumulative distribution of traffic for by individual prefixes, prefixes grouped by common origin AS, and prefixes grouped by common routing attributes.

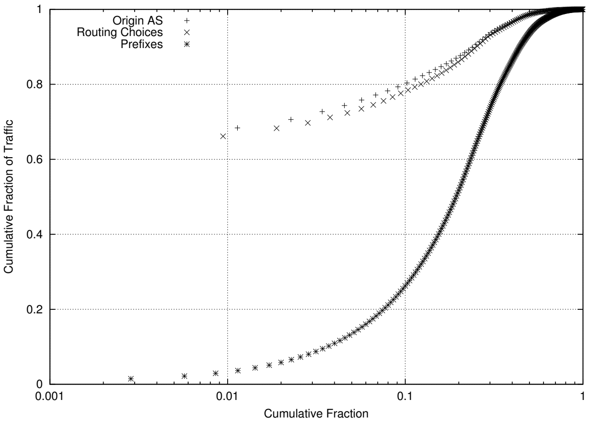

Figure 13: Cumulative distribution of traffic for one next-hop AS at one router. A small number of groups of prefixes with common routing advertisements are responsible for the majority of outbound traffic.

when we group prefixes with the same routing choices, as shown by the middle curve in Figure 12. For example, 10% of the sets of routing choices contribute more than 80% of the traffic. Grouping traffic by origin AS—the AS that originates the BGP announcement and receives the traffic—produces similar results, as shown by the top curve. The top 10% of origin ASes are responsible for approximately 82% of the outbound traffic.

By focusing on the small fraction of prefixes that carry the majority of the traffic, an operator can manipulate a large volume of traffic with a small number of routing changes. For example, an operator who wishes to reduce the load on an outgoing link might assign a smaller local preference value to the route advertisements associated with one or more popular prefixes at that router, thus shifting traffic destined for these prefixes to a different egress point. That is, each ingress point that is sending traffic to these destinations prefixes via this outgoing link would start sending the traffic via the next closest egress point with a "best" route. Rather than moving traffic for individual prefixes, the import policy modifications based on route advertisement attributes can move the traffic associated with popular *groups* of related prefixes. Figure 13 shows the distribution of traffic for a single egress point (a particular next-hop AS at one router). Compared to Figure 12, the bottom curve in Figure 13 shows a more even distribution across the destination prefixes, since each egress point carries traffic destined to some subset of prefixes. Nevertheless, the top curves show that a few *groups* of prefixes carry most of the traffic.

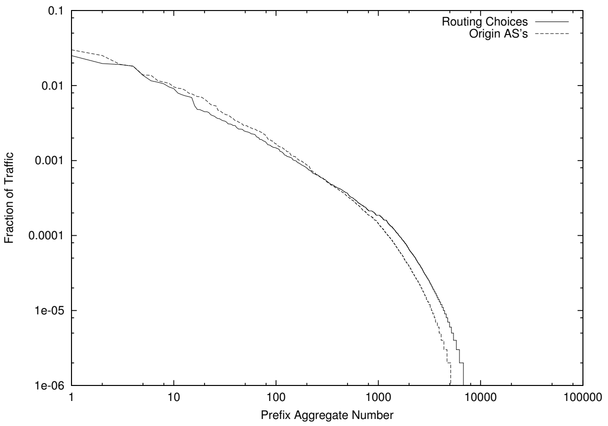

Figure 14: Proportion of traffic for each group of prefixes for one next-hop AS at one router. Each group of prefixes carries a different proportion of total outbound traffic at that router, providing network operators flexibility in shifting arbitrary amounts of traffic.

A relatively simple change in import policy can move a significant amount of traffic to or from a particular egress link. However, the appropriate amount of traffic to move may depend on the current link loads. Typically, an operator selects a set of prefixes to shift based on the current traffic distribution. Figure 14 shows the proportion of traffic traversing a particular egress point associated with each origin AS and each set of unique routing choices. Knowledge about traffic distributions for each origin AS and each set of prefixes with common routing attributes allows the operator to identify groups of prefixes associated with a certain proportion of the traffic and devise changes to import policy that manipulate an appropriate traffic volume.

6.3 Move Stable Traffic Volumes

Section 6.2 describes how to shift traffic by making import policy changes that affect the routes taken to groups of destination prefixes. The effects of these types of changes depend on the volume of traffic traveling to the destination associated with these routes. Even if the aggregate utilization of the link is relatively stable, the contribution of individual prefixes or origin ASes can be highly variable. Figure 15 shows the cumulative distribution of origin ASes experiencing a particular change in traffic between April 1, 2002 and April 8, 2002. The bottom curve shows the results for all origin ASes. For example, the point $(1, 0.7)$ on the lower line indicates that 70% of all origin ASes experience less than a 100% fluctuation in traffic from week-to-week; the remaining 30% of the origin ASes experience *more* fluctuation. This amount of variation would make it difficult to use traffic measurements from one day to drive traffic engineering decisions on another day. In particular, the prediction tools described in Section 2.2 would not make accurate estimations about how much traffic would be affected by changes in import policies.

Fortunately, by focusing on the groups of prefixes that carry significant portions of traffic, a network operator can make the effects of BGP policy changes more predictable. This is illustrated in the

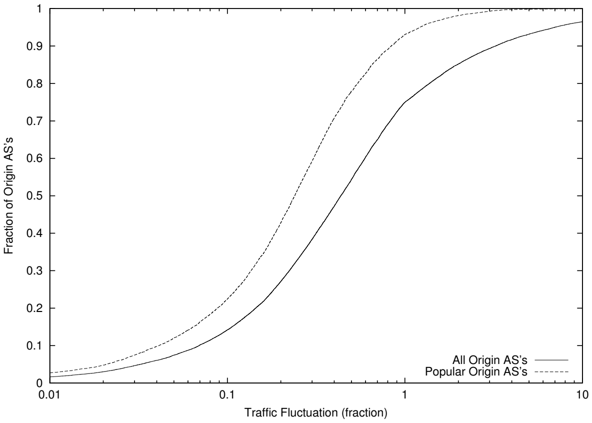

Figure 15: Cumulative distribution of origin ASes experiencing a particular fraction change in traffic from week-to-week. The graph shows this characteristic both for all origin ASes and for "popular" origin ASes—those that receive at least 0.01% of the total outbound traffic. Popular origin ASes tend to be more stable: only 20% of all popular origin ASes experienced more than a 50% traffic fluctuation from week to week, even though 45% of all origin ASes experienced such a fluctuation.

top curve in Figure 15, which focuses on the origin ASes that receive at least 0.01% of the total traffic ("popular" origin ASes). The graph shows that popular origin ASes tend to have more stable traffic volumes: only 20% of all popular origin ASes experienced more than a 50% traffic fluctuation from week to week, even though 45% of all origin ASes experienced such a fluctuation. Graphs for other pairs of days one week apart show similar trends, which are consistent with earlier studies that show that aggregation results in more stable traffic loads over time [25]. Thus, network operators should focus their attention on changing routes for prefixes and groups of prefixes that are responsible for larger fractions of the traffic. Fine tuning by moving small amounts of traffic may prove rather difficult in practice.

Nevertheless, the degree of stability varies across the popular destinations—certain destinations have remarkably stable traffic volumes, as shown by the left portion of the two curves in Figure 15. Just over 7% of all origin ASes have a traffic volume that fluctuates less than 5% between the two dates. This amount of fluctuation is arguably small enough to enable the traffic engineering tools to make accurate predictions of the volume of traffic that would move from one route to another. Tracking measurement data over time would allow an operator to identify the specific prefixes (and groups of prefixes) with relatively stable traffic volumes. The operator can focus on routes for these destinations when trying to move traffic from one link to another.

7. Conclusion and Future Work

BGP is a flexible interdomain routing protocol that scales to the large number of ASes in today's Internet. However, BGP was not designed with traffic engineering in mind. The attributes available in BGP advertisements, the restrictions in the BGP decision process, and the constraints imposed by configuration languages all limit an operator's ability to tune routing policies to the prevailing traffic patterns. A network operator can achieve certain traffic engineering goals by making changes to the BGP import policies running on its routers. Using BGP policies to shift traffic requires extreme care: changes should result in predictable and stable changes in traffic flow and minimize the possibility of affecting inbound traffic volumes. In particular, our analysis suggests that operators should make policy changes based on large groups of prefixes (e.g., groups of prefixes that have a common origin AS, or other common attributes), limit policy sensitivity to AS path changes by assigning policies based on AS path regular expression matches, and assign local preference based on ranges of AS path lengths, rather than using AS path length as an absolute metric.

The techniques we suggest can be used together to solve real traffic engineering problems. For example, suppose an operator realizes (say, via SNMP data) that a particular edge link is congested. First, fine-grain measurement data (such as Netflow) can be used to identify the destination prefixes responsible for the bulk of the traffic traversing this link; historical measurement data could be used to determine which of these prefixes have stable traffic volumes. Next, the operator could analyze the routing data to focus on the popular, stable prefixes that have a single "best" AS path across all of the egress points. Then, the operator could consider modifying the import policy at the congested router to assign a lower local preference to some of these destination prefixes to divert this traffic to the other egress links. Rather than assigning local preference directly to each prefix, the operator could inspect the routing data to select a suitable regular expression on the AS path attribute. Finally, the operator could test this policy using the prediction tool in Figure 2 to check how the proposed change would affect the flow of traffic in the network. In fact, ultimately, the traffic engineering tools could evolve to automate many of these steps by identifying specific destination prefixes and import policy changes for the operator.

Interdomain traffic engineering using BGP policies presents many interesting avenues for future work:

- *Traffic stability:* The amount of traffic traveling to each destination prefix varies over time. Effective traffic engineering relies on understanding how traffic stability varies with the level of aggregation and over time. Section 6.3 makes a few initial observations about the stability of traffic volumes for prefixes and groups of prefixes, but a better understanding about traffic stability could enable operators to make traffic engineering changes with higher confidence. The notions of "operational constancy" and "predictive constancy" [26] may be helpful in identifying which kinds of fluctuations in traffic volume might affect traffic engineering decisions.

- *Inbound traffic:* In this paper, we have focused on the influence of BGP import policies on *outbound* traffic; however, a complete solution should consider inbound traffic as well. Since an operator has limited control over how traffic enters the network (using crude techniques such as AS prepending), we believe that neighboring ASes should coordinate to gain a greater level of predictability with respect to how traffic enters each network. We are considering ways for neighboring ASes to cooperate without revealing their network topologies and routing policies [27].

- *Performance objective:* Traffic engineering involves tuning routing policies based on a target performance objective. The commercial relationships between ASes impose constraints and costs based on the volume of traffic exchanged with neighboring domains. In addition, the distribution of traffic after network failures may also play a role in evaluating possible changes to the routing configuration. Drawing on earlier work on IGP optimization, our ongoing work considers new objective functions that capture the constraints of both intradomain and interdomain routing, including the influence of peering agreements.

- *End-to-end performance:* Changes in BGP policies affect the *end-to-end* path from a source to a destination which, in turn, influences performance. We are investigating ways to collect information about the performance properties of the rest of the path to help weigh the benefits of different changes in BGP policies and IGP weights. For example, active measurements that identify congestion problems in other ASes would lend insight into which policy changes would improve end-to-end performance.

These ongoing research efforts can draw on and extend the insights from the analysis of routing and traffic data we have presented.

Appendix

In this appendix, we discuss lower-level details related to the configuration of import policies on BGP-speaking routers. First, we describe configuration options that operators should enable to make the BGP decision process deterministic and reduce the overhead of making changes in import policies. Then, we discuss the influence of other BGP attributes (besides local preference and AS path) on the decision process.

A. Router Configuration Options

The traffic engineering framework in Figure 2 of Section 2.2 depends on the ability to predict how BGP import policies affect the selection of the best route. We discuss configuration options that an operator should enable to ensure that the BGP decision process has a deterministic outcome. Then, we describe other configuration options that enable network operators to modify an import policy without resetting the BGP session.

A.1 Deterministic BGP Decision Process

Some router vendors have an additional step in the BGP decision process that occurs between the "lowest IGP metric" and the final "lowest router ID" steps. This additional step prefers the "oldest" route—the route that was received the earliest among the ones still in consideration. Including this step has the desirable effect of favoring the more stable routes over the routes that change frequently. However, this makes the outcome of the BGP decision process dependent on the *order* the router receives the advertisements, making it impossible to predict the selection of the best route from a static snapshot of the routing choices. Disabling age-based tie-breaking forces a deterministic selection based on the smallest router ID. Other BGP features, such as route flap damping [28], can help reduce the likelihood of selecting unstable routes.

The MED attribute is another potential source of non-determinism. As discussed above, the comparison of MED values applies only to routes learned from the same next-hop AS. As a result, the comparison between routes is not necessarily transitive—route r_1 being "better" than route r_2 and route r_2 being "better" than r_3 does *not* necessarily imply that route r_1 is "better" than r_3. This can make the selection of the best path dependent on the *order* of the comparison between paths, as illustrated by a detailed example in [29]. Router vendors recommend enabling the "bgp deterministic-med" option for deterministic path selection in the presence of MEDs.

A.2 Avoiding BGP Session Resets

A router applies the import policy to filter and manipulate BGP advertisements as they arrive from a BGP neighbor, as part of constructing the Routing Information Base (RIB). After a change in the import policy, the router needs to apply the new import policy to the existing routes learned from the BGP neighbor. However, the RIB only stores the routes as they appear *after* import processing under the *old* policy, and the old import policy may have filtered some routes and manipulated the attributes of others. Applying a new import policy could conceivably require the router to *reset* the session with the BGP neighbor in order to receive a fresh copy of each advertisement. This introduces substantial overhead on both routers and causes temporary routing instability that could spread to other parts of the Internet.

To avoid this problem, operators can configure their routers to store a local copy of each received advertisement. Enabling the "soft-reconfiguration" feature on inbound routes allows the router to apply the new import policy without disrupting the BGP session with the neighbor [30]. Enabling soft reconfiguration has the additional advantage of allowing operators to inspect or dump a copy of the received routes (e.g., using the "show ip bgp received-routes" command on a Cisco router). Dumping the received routes is useful for diagnosing routing problems and provides a more complete view of the routing choices learned from neighboring domains than the RIB does. However, the soft-reconfiguration feature has the disadvantage of consuming additional memory on the router. The relatively new "route refresh" option [31] in BGP is a viable alternative, if the neighbor's router supports it. This feature allows a router to signal a BGP neighbor to send a fresh copy of each advertisement without resetting the BGP session.

B. BGP Attributes and the Decision Process

To simplify the discussion, Section 2.1 presented a view of the BGP decision process that omitted the influence of two BGP attributes, origin type and multi-exit discriminator (MED). The origin type identifies how the origin AS learned about the route—within the AS (e.g., static configuration), EGP (a now-defunct distance-vector protocol), or injection from another routing protocol. These origin types are known as IGP, EGP, and INCOMPLETE. After considering AS path length, the BGP decision process prefers IGP routes over EGP routes, and EGP routes over INCOMPLETE. The MED attribute is an integer value set by an eBGP neighbor to encourage the recipient to pick a particular egress point for traffic. After considering the origin type and before considering "eBGP vs. iBGP," the decision process selects routes with the lowest MED value. The default behavior in most routers is to compare MED values only across routes with the same next-hop AS.

We have focused on how local preference assignment influences the first stage in the BGP decision process. After local preference, AS path length influences the selection of the "best" routes. Because origin type and MED affect the next two stages of the decision process, these attributes may *remove* some of the routes from consideration, reducing the set of "best" routes. In some cases, an eBGP neighbor may require the AS to accept these attributes as they appear in the advertisement messages. For example, a neighbor may use the MED attribute to override "hot potato" routing, where an AS can select the "closest" egress point based on the IGP path costs. Two AS's may have an agreement in advance to send and accept the MEDs. Alternatively, an operator may choose to ignore these two attributes by *resetting* their values in the import policy. Operators sometimes choose to reset the origin type and MED values to prevent an eBGP neighbor from using these attributes to affect the outcome of the decision process.

Alternatively, operators can *reassign* origin type or MED values in the import policy to influence route selection. This complements the influence of local preference on the decision process by giving an operator control *after* the selection of the routes with the shortest

AS path. For example, if an AS has multiple BGP sessions with a neighboring domain and wants to shed some traffic from an egress point, an operator can assign a higher MED value (or less preferable origin type) to some prefixes at this egress point. An operator can also override the default behavior of limiting MED comparison to routes with the same next-hop AS. For example, Cisco IOS has a "always-compare-MED" command that causes the BGP decision process to compare MED values across all routes, irrespective of the next-hop AS. For all of these techniques for configuring import policies, network operators can draw on the insights in the main body of our paper to decide *which* traffic they should move between egress points.

Network operators can employ a variety of other techniques to influence the decision process. Section 5.1 described how a neighboring domain might employ AS prepending in the export policy to inflate the AS path length. An operator can use this technique in the *import* policy to influence the selection of best routes, effectively eliminating some routes in the "AS path length" step. Operators might also use the BGP community attribute to "program" a wide variety of policies for path selection. A community is an opaque string that is assigned to a route by an import or export policy. The import policy could tag a route with a community string to label whether the route was learned from a peer or a customer, or based on the geographic location. A network operator can use these tags to affect the assignment of other BGP attributes or the decision of whether to export the route to certain neighboring AS's. For example, an operator could use the tags to instruct routers in Europe to assign lower local preference values to routes learned in the United States in order to minimize the use of slow (and often expensive) transatlantic links.

Acknowledgments

We thank Tim Griffin for many very helpful discussions and Carsten Lund for providing the aggregated Netflow data. Thanks also to Dave Andersen, Hari Balakrishnan, Randy Bush, Steve Garland, Joel Gottlieb, Jaeyeon Jung, Carsten Lund, Aman Shaikh, Alex Snoeren, Iljitsch van Beijnum, Jia Wang, and the anonymous reviewers for very helpful feedback.

C. References

[1] D. O. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of Internet traffic engineering." Request for Comments 3272, May 2002.

[2] D. O. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for traffic engineering over MPLS." Request for Comments 2702, September 1999.

[3] D. O. Awduche, "MPLS and traffic engineering in IP networks," *IEEE Communication Magazine*, pp. 42–47, December 1999.

[4] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford, "NetScope: Traffic engineering for IP networks," *IEEE Network Magazine*, pp. 11–19, March 2000.

[5] B. Quoitin, S. Uhlig, C. Pelsser, L. Swinnen, and O. Bonaventure, "Interdomain traffic engineering with BGP," *IEEE Communications Magazine*, pp. 122–128, May 2003.

[6] J. Wepman and J. Abley, "Inter-domain Traffic Engineering: Principles, Applications, and Case Studies." `http://www.nanog.org/mtg-0202/te.html`, February 2002. Tutorial at NANOG 24. Miami, FL.

[7] Z. Kerravala, N. Maynard, and A. Phull, "Intelligent routing: The high IQ Internet," July 2002. The Yankee Group. `http://www.sockeye.com/pdf/Yankee_IntellRouting_July02.pdf`.

[8] Y. Rekhter and T. Li, "A Border Gateway Protocol." Request for Comments 1771, March 1995.

[9] S. Halabi and D. McPherson, *Internet Routing Architectures*. Cisco Press, second ed., 2001.

[10] J. W. Stewart, *BGP4: Inter-Domain Routing in the Internet*. Addison-Wesley, 1998.

[11] G. Huston, "Interconnection, peering, and settlements," in *Proc. INET*, June 1999.

[12] "BGP Best Path Selection Algorithm." `http://www.cisco.com/warp/public/459/25.shtml`.

[13] "How the Active Route Is Determined." `http://arachne3.juniper.net/techpubs/software/junos42/swconfig-routing42/html/protocols-overview4.html#1045417`.

[14] "Foundry Switch and Router Installation and Configuration Guide, Chapter 19, Configuring BGP4." `http://www.foundrynet.com/services/documentation/SRguide/FoundryManual_BGP4.html`.

[15] C. Labovitz, A. Ahuja, and F. Jahanian, "Experimental study of Internet stability and wide-area network failures," in *Proc. International Symposium on Fault-Tolerant Computing*, June 1999.

[16] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang, "BGP routing stability of popular destinations," in *Proc. Internet Measurement Workshop*, November 2002.

[17] N. Feamster and J. Rexford, "Network-wide BGP route prediction for traffic engineering," in *Proc. Workshop on Scalability and Traffic Control in IP Networks, SPIE ITCOM Conference*, August 2002.

[18] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True, "Deriving traffic demands for operational IP networks: Methodology and experience," *IEEE/ACM Trans. Networking*, vol. 9, June 2001.

[19] "Cisco Netflow." `http://www.cisco.com/warp/public/732/netflow/index.html`.

[20] "Sampled Netflow." `http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120limit/120s/120s11/12s_sanf.htm`.

[21] N. Duffield, C. Lund, and M. Thorup, "Charging from sampled network usage," in *Proc. Internet Measurement Workshop*, November 2001.

[22] A. Broido and K. Claffy, "Analysis of RouteViews BGP data: Policy atoms," in *Workshop on Network-Related Data Management*, May 2001.

[23] T. Bu, L. Gao, and D. Towsley, "On Characterizing BGP Routing Table Growth," in *Proc. IEEE Global Internet*, (Taipei, Taiwan), November 2002.

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

[25] N. Taft, S. Bhattacharyya, J. Jetcheva, and C. Diot, "Understanding traffic dynamics at a backbone POP," in *Proc. Workshop on Scalability and Traffic Control in IP Networks, SPIE ITCOM Conference*, August 2001.

[26] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the constancy of Internet path properties," in *Proc. Internet Measurement Workshop*, November 2001.

[27] J. Winick, S. Jamin, and J. Rexford, "Traffic engineering between neighboring domains." `http://www.research.att.com/~jrex/papers/interAS.pdf`, July 2002.

[28] C. Villamizar, R. Chandra, and R. Govindan, "BGP Route Flap Damping." Request for Comments 2439, November 1998.

[29] "How BGP Routers Use the Multi-Exit Discriminator for Best Path Selection." `http://www.cisco.com/warp/public/459/37.html`.

[30] "BGP Soft Reset Enhancement." `http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t7/sftrst.htm`.

[31] E. Chen, "Route refresh capability for BGP-4." Request for Comments 2918, September 2000.