On the Prevalence and Characteristics of MPLS Deployments in the Open Internet

Joel Sommers
Colgate University
jsommers@colgate.edu

Brian Eriksson
Boston University
eriksson@cs.bu.edu

Paul Barford
University of Wisconsin-Madison and Qualys, Inc.
pb@cs.wisc.edu

ABSTRACT

Multi-Protocol Label Switching (MPLS) is a mechanism that enables service providers to specify virtual paths through IP networks. The use of MPLS in the open Internet (*i.e.*, public end-to-end paths) has important implications for users and network neutrality since MPLS is frequently used in traffic engineering applications today. In this paper we present a longitudinal study of the prevalence and characteristics of MPLS deployments in the open Internet. We use path measurement data collected over the past 3.5 years by the CAIDA Archipelago project (Ark), which consist of over 10 billion individual traceroutes between hosts throughout the Internet. We use two different techniques for identifying MPLS paths in Ark data: direct observation via ICMP extensions that include MPLS label information, and inference using a Bayesian data fusion methodology. Our direct observation method can only identify uniform-mode tunnels, which very likely underestimates MPLS deployments. Nonetheless, our results show that the total number of tunnels observed in a given measurement period has varied widely over time with the largest deployments in tier-1 providers. About 7% of all autonomous systems deploy MPLS and this level of deployment has been consistent over the past three years. The average length of an MPLS tunnel has decreased from 4 hops in 2008 to 3 hops in 2011, and the path length distribution is heavily skewed. About 25% of all paths in 2011 cross at least one MPLS tunnel, while 4% cross more than one. Finally, data observed in MPLS headers suggest that many ASes employ some types of traffic classification and engineering in their tunnels.

Categories and Subject Descriptors

C.2.3 [**Network Operations**]: Network management, Network monitoring; C.2.5 [**Local and Wide-Area Networks**]: Internet (*e.g.*, TCP/IP; C.4 [**Performance of Systems**]: Measurement Techniques

General Terms

Algorithms, Design, Experimentation, Measurement, Performance

IMC'11, November 2–4, 2011, Berlin, Germany.

Keywords

MPLS, `traceroute`, traffic engineering, tunnels

1. INTRODUCTION

In the late 1990's, the limitations and inflexibility in IP routing and forwarding coupled with the on-going quest to improve switching performance led to the development of Multiprotocol Label Switching (MPLS). Label switching[1] was envisioned and designed as a simple mechanism that would operate between layers 2 and 3 in the standard Internet protocol stack, and enable efficient lookups at each hop on a designated path. Standardization efforts in the IETF began in 1997 [14] and have resulted in an assortment of drafts and RFCs that define and specify the protocol. Today, MPLS is a standard feature in routers and is available on a wide variety of platforms from many different vendors.

Along with improved switching speeds, MPLS was quickly recognized as a highly useful protocol for traffic engineering [10]. MPLS affords the ability to configure multi-hop tunnels that supersede shortest path routes for definable traffic aggregates. It enables service providers to operate their infrastructures in a fashion that more directly addresses their business and operational requirements. Anecdotal reports suggest that MPLS is currently used by service providers in a variety of ways including facilitating traffic engineering, implementation of multi-service networks (including virtual private networks), and improving robustness [34]. Despite its appeal and implications of its broad use, to the best of our knowledge there are no studies to date that provide a systematic empirical assessment of MPLS deployments in the Internet.

In this paper, we present a longitudinal analysis of MPLS deployments in the open Internet (*i.e.*, paths that are not part of virtual private networks). The goal of our work is to establish a broad empirical baseline for MPLS since tunnels have a direct impact on traffic behavior and measurement, and have significant implications for network neutrality, which is an increasingly active topic of conversation. We seek to answer questions such as: How many networks use MPLS? Has the use of MPLS grown over time? What are the characteristics of individual MPLS tunnels? How likely is it that my traffic will encounter an MPLS tunnel? To address these questions, we require measurements of a large set of Internet paths conducted over a period of years.

Studies of Internet-wide phenomena related to end-to-end, router-level path properties typically rely on active probe-based measurement tools such as `traceroute` for gathering data. At first glance, MPLS appears to present an insurmountable challenge to standard TTL-limited active probing methods that rely on layer 3 messaging

[1]Originally it was called "tag switching" [30] and had roots in Ipsilon's flow management protocol [29].

since MPLS is not a layer 3 protocol. However, relatively recent extensions to the ICMP protocol [12] that enable it to include the entire MPLS label stack overcome this challenge (as long as the measurement tool and target routers implement the extensions and the router is not configured to hide the tunnel). Traceroute-style path measurements that include MPLS label stacks are the starting point for our work.

We use data collected by the Ark project [22], which, since 2008 includes MPLS label stacks wherever they are in use and visible on an end-to-end path. Network operators can configure MPLS tunnels in such a way as to hide them from `traceroute`. Thus, our work relies on tunnels that are configured in uniform (visible) mode, which we describe below. The Ark infrastructure and methodologies have been designed to efficiently measure all routable /24's in the Internet. Since its inception, Ark has conducted over 10B individual `traceroute` measurements, and as such offers are a compelling source of data for our longitudinal study.

Our analysis of MPLS labels in Ark path measurement data reveals a broad set of characteristics about the deployment and use of the protocol. In particular, we find that the total number of tunnels observed in any measurement period has varied widely over time from a low of about 200K in '09 to a recent high of around 350K. Curiously, we find that the variability of tunnel deployments correlates closely with key economic indicators. Roughly 7% of all autonomous systems use MPLS on some subset of their paths with the largest deployments in tier-1 providers. This level of deployment has been relatively stable over the past three years. We also find that the average length of an MPLS tunnel has decreased from 4 hops in 2008 to 3 hops in 2011. The tunnel length distribution is heavily skewed with over 90% of tunnels at 7 hops or less, however some tunnels extend beyond 15 hops. Approximately 25% of all paths in 2011 cross at least one MPLS tunnel, while 4% cross more than one, and data observed in MPLS headers suggest that many ASes employ some kinds of traffic classification and engineering in their tunnels.

While label-based analysis enables a compelling characterization of MPLS path properties, it precludes analysis of other path measurement data archives (*e.g.,* Skitter [13]) that do not include labels. Analysis of data sets that lack the ICMP extension information could expand our perspective and enable a more comprehensive longitudinal analysis. To address this problem we develop an MPLS tunnel identification method that is based on analysis of round trip time measurements instead of labels. The observation is that with some MPLS configurations, when a packet's TTL expires, the ICMP time exceeded message will be encapsulated in an MPLS header and forwarded *to the end of the tunnel* prior to being sent back to the source host. This observation is the basis for a Bayesian inference method that we show to be effective for identifying MPLS tunnels. Applying this method to additional data sets to gain a broader view of MPLS deployments over time is a goal for future work.

A summary of the key findings our work are as follows. First, the increasing trend in deployments over the past three years and the wide use by tier-1 providers means that it is increasingly likely that packets will encounter an MPLS tunnel. Second, tunnels are likely to span the entire edge-to-edge distance of a transit provider, with typical transit times on the order of 10s of milliseconds. Third, our examination of the use of traffic classifiers indicates that while multiple classes are not uncommon, the diversity of classes has not changed over the past three years.

The remainder of this paper is organized as follows. In Section 2, we provide an overview of MPLS, how MPLS interacts with `traceroute`, and our label-based method for identifying MPLS hops on an end-to-end path. In Section 3, we provide details on the Ark data set and report the details of our label-based MPLS deployment study. In Section 4, we describe our inference-based method for identifying MPLS paths and report our findings on applying this method to the Ark data. We summarize our work and describe next steps in our study in Section 6.

2. MPLS AND TRACEROUTE

MPLS is typically thought of as a protocol that exists between layers 2 and 3 of the Internet protocol stack. Fundamentally, it is path-based encapsulation and forwarding protocol that adds a 32-bit header to packets as they enter an MPLS tunnel, or label-switched path (LSP). We use the terms tunnel and LSP interchangeably in this paper. The MPLS header consists of a 20-bit label, 8-bit time-to-live (TTL) field, 3-bit traffic class field, and a 1-bit end of label stack field.

Specific labels are applied to packets based on forwarding equivalence class (FEC), which is a generalization of longest-prefix match. As packets traverse an MPLS tunnel, forwarding decisions are made based on *exact* matching of the MPLS label. Labels are only locally significant between a pair of routers, so as packets traverse a tunnel, labels are swapped (overwritten) prior to forwarding. Hierarchies of tunnels can also be created (*i.e.*, an already-tunneled packet can enter a new tunnel); stacks of MPLS headers can be used for this purpose.

MPLS tunnels must be configured (and labels distributed) on each label switch router (LSR) along a designated path (series of LSRs), and supersede layer 3 routes. The flexibility to define paths through a network not simply based on shortest paths makes MPLS highly attractive for traffic engineering tasks. Details on MPLS can be found in [31], and several other IETF RFCs.

Our work relies on recent modifications to routers and `traceroute` programs that implement extensions to ICMP specified in RFC 4950. These extensions permit the inclusion of the entire MPLS label stack (*i.e.*, *all* MPLS header information) in the ICMP message that is returned to a source host [12]. Thus, it is this information included in ICMP time exceeded messages that are generated as part of the `traceroute` process that allows us to positively identify a router as participating in an MPLS tunnel.

There are two basic ways in which the IP time-to-live (TTL) field is processed in the presence of an MPLS tunnel. These are referred to as *uniform mode* and *pipe mode* [9], and they determine whether an MPLS tunnel is visible to a public user of `traceroute` or not[2]. Figure 1 illustrates these two modes of operation.

In pipe mode, the MPLS tunnel is not exposed to a `traceroute` user. At LSR B in Figure 1, the IP TTL is decremented by one and inspected upon router ingress. After that, the packet is encapsulated in an MPLS header and forwarded to the next LSR along the path. The MPLS header that is constructed at the first LSR is initialized with a prespecified TTL value, typically 255. Upon egress from a pipe mode tunnel, the MPLS header is removed and the TTL value in the IP header is unchanged. Thus, the MPLS TTL field has no relationship to the IP TTL in a pipe-mode tunnel, and the IP TTL is only decremented by one regardless of the number of LSRs that switch the packet through the tunnel.

Note that with the pipe model, the first router of an MPLS tunnel is visible, but the ICMP time exceeded response generated from the

[2]Note that the uniform and pipe modes more commonly refer to tunneling models to support differentiated services in MPLS networks [19]. We use these terms similar to the way in which they are used to describe TTL processing in MPLS networks in RFC 3443 [9].

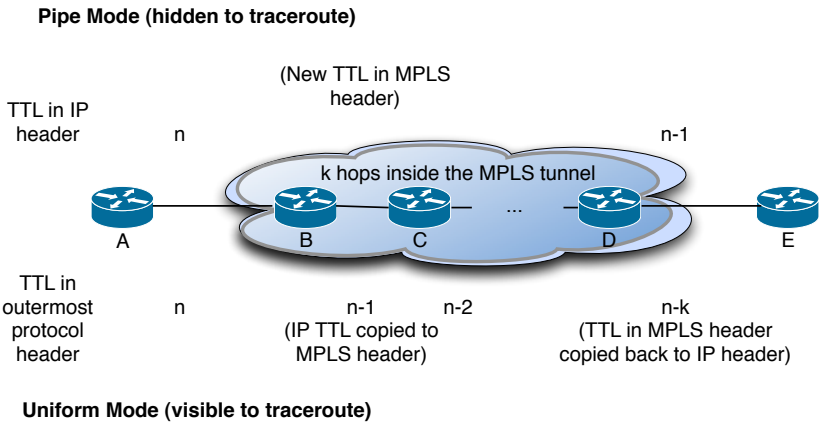

Figure 1: Two modes of IP header TTL processing in the presence of an MPLS tunnel: pipe and uniform.

router (if the IP TTL is decremented to zero) does not indicate that the packet expired at the edge of a tunnel, since it has not actually entered one yet. Thus, there is no indication to a `traceroute` user that an MPLS tunnel is traversed, and only the ingress router is visible at all. (Note that penultimate hop-popping (PHP) does not affect this picture from the standpoint of `traceroute` [9]. We discuss PHP further in § 3.3.) The upshot is that identifying pipe mode tunnels with end-to-end measurements remains a significant measurement challenge and one that we do not address in this paper.

In uniform mode, the LSRs along an MPLS tunnel are visible to a `traceroute` user. At the first LSR (B in the figure), the IP TTL is decremented by one and inspected upon router ingress, just as in the pipe model. Upon encapsulation in an MPLS header, however, the IP TTL is copied into the MPLS header. At each LSR along the tunnel, the TTL in the MPLS header is decremented by one. Upon egress from a uniform mode tunnel, the MPLS TTL field is copied back to the IP header. If the TTL falls to zero at any router along the tunnel, an ICMP time exceeded message will be generated and sent back to the source of the original packet. Thus, *all* routers in the tunnel are visible to a user of `traceroute`. Importantly, only if the router implements MPLS extensions for ICMP [12] will a `traceroute` user be able to clearly identify the fact that the packet's TTL expired while inside an MPLS tunnel.

Interestingly, uniform mode is the default mode of operation for Cisco and Juniper routers (among others) [4, 5]. This behavior is likely due to language in the MPLS architectural RFC (3031) that specifies that the total number of hops through a tunnel `SHOULD` be reflected in the IP TTL when the packet emerges from the tunnel. Also, it is important to note that pipe and uniform mode TTL processing can be configured on a per-LSP basis. Even nested tunnels can have different visibility characteristics (*i.e.*, while the innermost tunnel may be visible via `traceroute`, a nested tunnel may not be). From a practical perspective, this means that network operators must explicitly configure routers *not* to propagate the IP TTL to the MPLS header, thus hiding tunnels from the public's eye.

Another important impact that MPLS tunnels can have is on delay measurements of individual `traceroute` packets. When the TTL of a packet is decremented to zero inside an MPLS tunnel (in uniform mode, since that is the only way that an outsider can observe an MPLS tunnel), the ICMP time exceeded message may need to be re-encapsulated in an MPLS header and forwarded to the end of the tunnel [8]. If the LSR at which the ICMP packet is generated does not have sufficient information (*e.g.*, routes imported via BGP), the packet will be re-encapsulated and traverse the entire length of the tunnel. At the end of the tunnel, the ICMP message can be routed back to the source host. The result is that the `traceroute` output will show roughly equivalent delay for the series of hops along the MPLS tunnel. In that case, we would measure the latency across the tunnel to be approximately zero. This observation is the starting point for our tunnel inference method described in Section 4.

While we focus on measurement of MPLS tunnels from outside a service provider's network, internal network operators can use other mechanisms for measuring and troubleshooting MPLS LSPs. For example RFC 4379 describes MPLS-based versions of ping and traceroute for this purpose [24].

3. ARK DATA ANALYSIS

In this section we present our analysis of MPLS tunnels and their characteristics as observed in the CAIDA IPv4 Routed /24 AS Links Dataset [22]. Our focus in this section is on characteristics of MPLS tunnels that can be directly observed through ICMP extension information.

3.1 Data and Limitations

Since we do not have insight into the ways in which labels have been assigned, or generally how MPLS has been configured in a given provider network, we use a pragmatic definition of *tunnel* in our data analysis. Our goal in the present work is not to try to identify *how* different MPLS tunnels have been configured, but rather to report and analyze their observed characteristics. Our working definition of *tunnel* is any consecutive series of label-switched routers within the same autonomous system[3]. As described in Section 2, we can identify LSRs by the presence of MPLS extension information in an ICMP time exceeded message from a router. Note that with this definition, the length of a tunnel refers only to the innermost label switched path (*i.e.*, the length of the tunnel identified by the sequence of labels at the bottom of the label stack) and does not consider nested LSPs. (We comment on nested tunnels and nested tunnel lengths below.) Note also that within an AS, we may see some number of layer 3 (IP) hops before entering a tunnel. We

[3]In our analysis, we detected no instances of a tunnel that spanned multiple ASes.

may also see some number of layer 3 hops *after* exiting a tunnel, as we discuss below.

The data that we use are collected as part of the Archipelago (Ark) project, and include `traceroute` measurements to all routed /24 prefixes in the IPv4 Internet, initiated from a set of widely distributed probing hosts. The probing tool used in Ark is Scamper [26], and the specific `traceroute` method used is ICMP Paris [28]. The Ark project was initiated in 2007, and data are available starting from September of that year. However, the version of Scamper used at that time did not support the ICMP extensions for MPLS. Support for those extensions was introduced into Scamper in early 2008, and a version of Scamper with support for ICMP extensions for MPLS was rolled out to Ark in mid-May of 2008[4] [21, 27]. For that reason, we restrict our analysis in this section to the Ark data from June 2008 to August 2011. Lastly, rather than analyzing *all* the available Ark data from June 2008 onward, we selected the first full set of data available for each month resulting a corpus of over 250M individual `traceroute` measurements. Investigating MPLS deployment dynamics at finer levels of granularity is a topic for future study. In our analysis below, the scope of our characterizations is bounded by the Ark measurement data, and placement of probing nodes. However, the Ark project has gone to great lengths to be comprehensive and we believe it provide an important and representative view of Internet structure.

Along with the Ark data, we used the CAIDA IPv4 prefix to autonomous system (AS) mapping data, which is based on analysis of Routeviews data [1]. We also use UCLA Cyclops data to classify a given AS as a tier 1, large ISP, small ISP, or stub network [2]. We use these data to perform AS-specific analyses, and to identify the number and characteristics of ASes that appear to be employing MPLS for traffic management. Since the prefix-to-AS mapping data are based on a measured perspective of Internet routing activity, there are imperfections and complications. In our analyses, we omit any `traceroute` paths that have IP addresses for which we do not know the AS number, and any IP addresses that have multiple ASes identified in the prefix-to-AS mapping. We do this to simplify our analyses of specific ASes. We note that the overall amount of data discarded is small (under 1%), thus we do not believe this materially affects our overall results.

For many analyses in this section, we wanted to identify the set of *unique* tunnels within an AS. We found that either using the sequence of IP addresses of the LSRs, or using the sequence of MPLS labels at the bottom of the label stack gave virtually equivalent results (differences were under 0.01% of the total number of observed tunnels).

Finally, as we discuss above, our analysis relies on the presence of ICMP extension information to positively identify MPLS LSRs. Thus, our analysis is limited to MPLS tunnels that have been configured in uniform mode. While we are certain that there are tunnels configured in pipe mode that evade our analysis, we have no way to presently quantify or characterize these tunnels. Indeed, there are ISP-published maps of MPLS networks for which we do not see any (or exceedingly few) LSRs in our data (*e.g.*, see Sprint's [6] network map and other maps available at the Internet Topology Zoo project [7]). Moreover, we have no visibility into any MPLS virtual private networks since our measurements are taken from hosts attached to the public Internet.

[4]The roll-out of a new version of Scamper corresponded to the passing of the Internet Measurement Conference paper deadline that year.

3.2 MPLS Prevalence

We begin by examining the prevalence of MPLS tunnels, and how many ASes appear to use MPLS. We turn first to the total number of ASes that we observe to use MPLS. In the top plot of Figure 2, we see this number has remained fairly steady over the past 3 years, at around 7%.

The bottom plot of Figure 2 shows the fraction of ASes, classified by AS type, that are empirically observed to employ MPLS. We use the UCLA Cyclops data to perform the AS classification [2]. (Note that these classification data are only available starting in October 2008.) We observe that nearly all tier 1 providers use MPLS (in October 2008 there were 8 tier 1 providers, and in August 2011 there were 11). We observe a lesser percentage of large ISPs (50–55%), an even smaller percentage of small ISPs (25–30%), and few stub networks (around 5%) to use MPLS. Because there are so many stub networks (13817 in August 2011), the overall percentage of ASes using MPLS is rather low (again, around 7%).

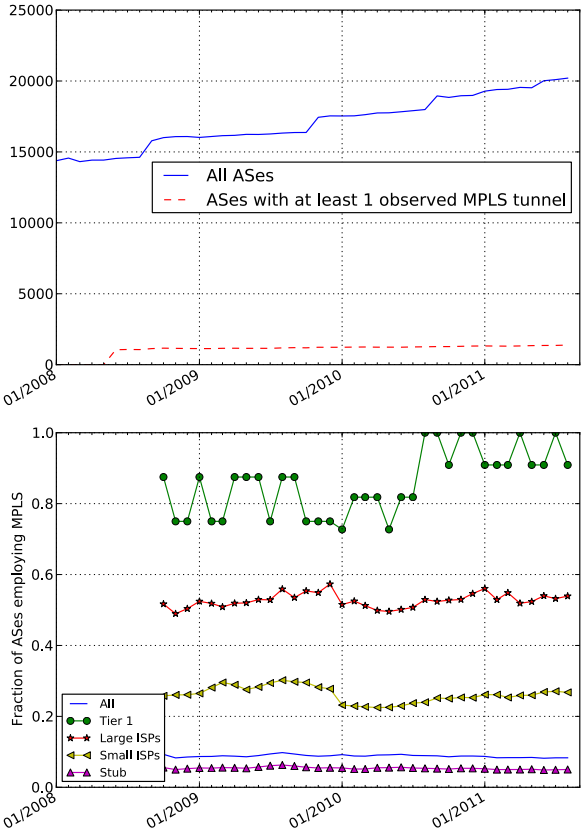

Figure 2: Comparison of the total number of ASes seen in our Ark dataset versus the number of ASes for which at least 1 MPLS tunnel is observed (top), and the fraction of ASes by AS type that employ MPLS (bottom).

Figure 3 plots the total number of unique tunnels observed in a given measurement period over the past three years. The figure shows a significant dip in the total number of tunnels beginning in '08, a rebound in mid '09 and a steady increase to the current peak of nearly 350K tunnels. For comparison, we also plot the Dow Jones Industrial Average in the figure[5]. Interestingly, the total number of MPLS tunnels over time appears to indirectly track this eco-

[5]Data obtained from `http://finance.yahoo.com/`.

nomic indicator (and other, similar indicators such as the S&P 500, not shown). So, while it is apparent from Figure 2 that the sheer number of ASes using MPLS is independent of economic activity, the deployment of tunnels *within* these ASes roughly correlates with economic conditions. Our conjecture is that this phenomenon reflects merger and acquisition activity within the telecom industry. We have engaged ISPs to better understand the underlying reasons behind our observations, and we hope to report on our findings in the future.

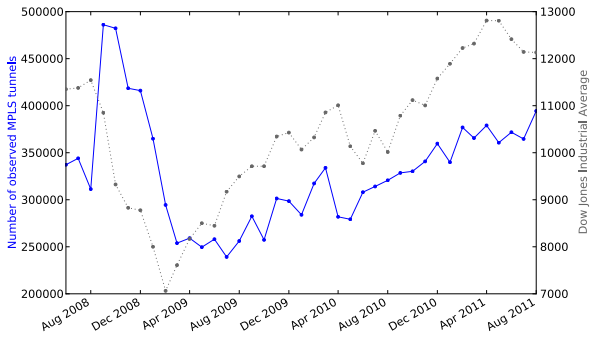

Figure 3: The total number of visible MPLS tunnels over time (solid line). The Dow Jones Industrial Average is plotted (dotted line) for reference over the same time period.

Figure 4 shows the number of unique tunnels deployed by a small set of ASes over the past three years. First, we see a sharp drop in the number of tunnels deployed by AS7018. This drop mirrors the drop in the total number of tunnels seen in Figure 3. We see from the figure that other ASes show some variability over time in the number of deployed tunnels. For example, AS1273 emerges in early 2010 from very few (nearly zero) observed tunnels. On the other hand, the number of tunnels observed from AS3320 drops to zero in mid-2010. We conjecture that in addition to economic factors, there are likely AS policy decisions (*e.g.*, to change tunnels to "invisible" pipe mode, or to reveal previously hidden tunnels) that play a role in the observed variability.

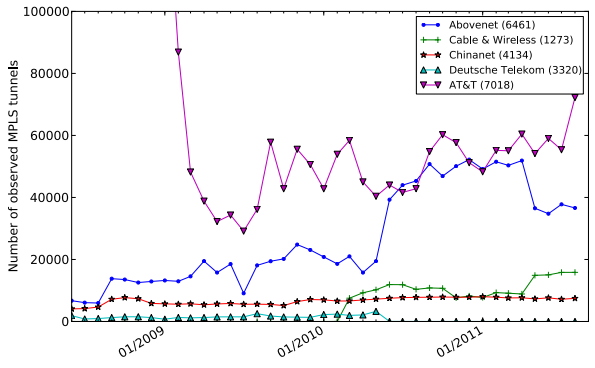

Figure 4: Number of observed MPLS tunnels over time across selected ASes.

Figure 5 provides another perspective on the number of deployed tunnels per AS. The figure shows empirical cumulative distribution functions of the number of observed tunnels per AS for six-month time intervals over the measurement period. We see that 20% of all ASes have fewer than 10 tunnels. We also see that 80% of ASes are observed to have under 200 tunnels, and that about 10% of ASes have at least 1000 tunnels.

Table 1 identifies the numbers and names of the top 10 ASes in terms of the number of observed MPLS tunnels. The table shows the top 10 ASes for six month periods starting from mid-2008. These results support the intuition and evidence in the bottom plot of Figure 2 that larger and more complex infrastructures are more likely to use MPLS tunnels.

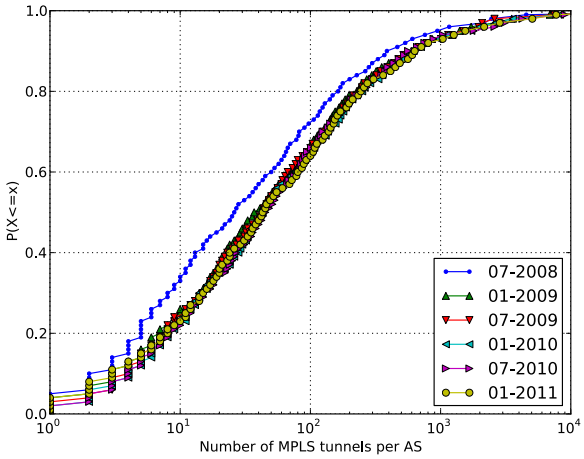

Figure 5: Empirical cumulative distribution functions of the number of visible MPLS tunnels per AS. CDFs are plotted for six month periods starting 6/2008. (Note that the x-axis is on log scale.)

Lastly, we examine the fraction of paths probed by Ark that include one or more MPLS tunnels. Figure 6 shows the fraction of probing paths that cross *at least* 1 MPLS tunnel, and the fraction of paths that cross exactly 1, 2, or 3 tunnels. Interestingly, although only about 7% of all ASes are observed to use MPLS, about 25% of all probing paths crossed at least 1 tunnel in the most recent measurement data. Approximately 4% of all paths cross more than one MPLS tunnel; this observation holds across the three years. We can see that the increase in likelihood of crossing at least 1 MPLS tunnel from about 20% in June 2008 to about 25% in August 2011 is primarily due to higher occurrences of encountering a single tunnel on a path. This result is consistent with the fact that larger networks are more likely to appear on end-to-end paths and are more likely to deploy MPLS.

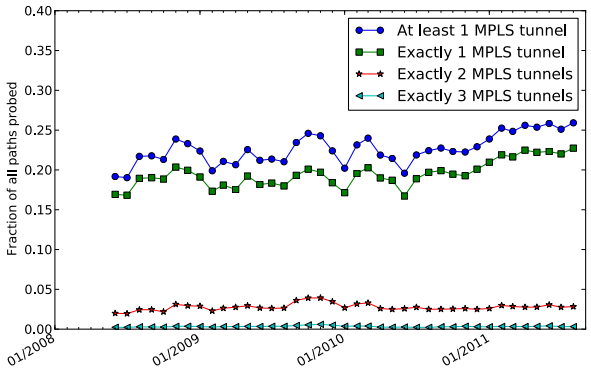

Figure 6: Fraction of paths with some MPLS.

Table 1: MPLS heavyweights: the top 10 ASes in terms of number of observed MPLS tunnels.

Rank	7–12/2008	1–6/2009	7–12/2009	1–6/2010	7–12/2010	1–6/2011
1	7018 AT&T	7018 AT&T	7018 AT&T	7018 AT&T	7018 AT&T	7018 AT&T
2	6389 Bellsouth	6389 Bellsouth	6461 Abovenet	6461 Abovenet	6461 Abovenet	6461 Abovenet
3	6461 Abovenet	6461 Abovenet	6389 Bellsouth	19262 Verizon	4837 China-169	6830 UPC
4	6453 Tata	3269 Telecom Italia	3269 Telecom Italia	6389 Bellsouth	6453 Tata	6453 Tata
5	3292 TDC	6453 Tata	19262 Verizon	6453 Tata	1273 CW	174 Cogent
6	4134 Chinanet	19262 Verizon	6453 Tata	4837 China-169	6830 UPC	1273 CW
7	4230 Embratel	4230 Embratel	4230 Embratel	4230 Embratel	4134 Chinanet	4837 China-169
8	19262 Verizon	4134 Chinanet	4837 China-169	4134 Chinanet	19262 Verizon	4134 Chinanet
9	5462 Virgin	4837 China-169	4134 Chinanet	6774 Belgacom	6774 Belgacom	4230 Embratel
10	4837 China-169	2914 NTT	2914 NTT	1273 CW	4230 Embratel	10318 Cablevision SA

Surprisingly, there were some probing paths that crossed 7 MPLS tunnels, which was the maximum we observed. Also surprisingly, within a single AS we observed multiple, separate tunnels in a single probing path. That is, from the `traceroute` output, we observed a series of MPLS LSR hops within an AS, followed by one or more "normal" IP hops, followed by another series of MPLS LSR hops. The frequency of occurrence of either many tunnels (> 4) on a single path, or multiple tunnels within a single AS was quite rare (about 0.3% of all paths in May 2011).

3.3 MPLS Tunnel Characteristics

We now examine characteristics of observed MPLS tunnels, both globally and within different ASes.

We first examine first-order path length statistics in ASes. We consider three segments of a path through an AS: IP hops *before* a tunnel is entered, hops *within* a tunnel, and IP hops *after* a tunnel, prior to exiting the AS. For this analysis, we omitted any paths through an AS that contained multiple separate tunnels. Figure 7 shows the average number of pre-tunnel, post-tunnel, and in-tunnel hops over the past three years. We see that the average length of an MPLS tunnel has decreased over the measurement period from just over 4 hops, to around 3 hops. This change likely is due to changes in operational policy or underlying infrastructure. We also see that the average number of pre- and post-tunnel hops has remained roughly constant over the past three years, at around 1 hop and 1.8 hops, respectively.

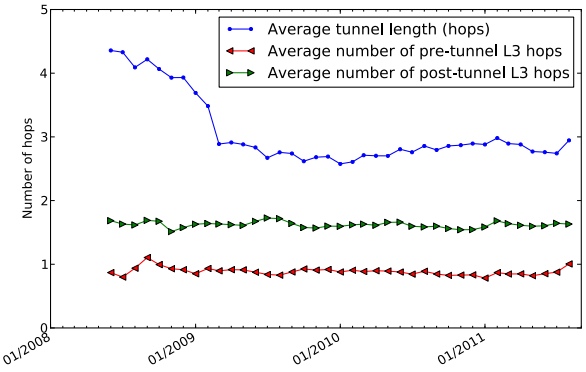

Figure 7: Average number of hops before, inside, and after MPLS tunnels for all ASes.

Figure 8 plots average tunnel lengths over time for 4 different ASes. We observe a variety of behaviors. While the average length of a tunnel in AS7018 has decreased by around 1 hop over the past three years, the average length of a tunnel in AS6461 has increased by around 2 hops. The average length of a tunnel in AS4230 has increased, but is quite short, and the plot for AS3320 shows that tunnels in that AS are currently an average of about 1 hop long.

An interesting phenomenon suggested by Figure 8 is that of very short 1-hop MPLS tunnels. In Figure 9, we show the fraction of tunnels over time that consist of one hop, for a set of ASes and globally across all tunnels. We again see a variety of behaviors. While most ASes have very few tunnels of just 1 hop, the majority of tunnels in AS4134 consist of only one hop, and tunnel lengths in AS3320 have decreased to close to a single hop.

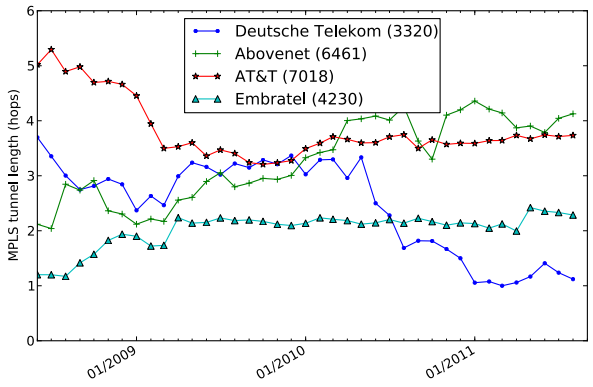

Figure 8: Average MPLS tunnel lengths for selected ASes over time.

While there are certainly operational policy decisions and other configuration factors which may lead to the observation of 1-hop tunnels (*e.g.*, there may be nested tunnels that are configured in pipe mode, making some segment of the tunnel invisible to `traceroute`), one simple reason we might see such short tunnels is the following. Consider the network in Figure 1, and assume that there are only three LSRs in the tunnel (B, C, and D). When a packet enters the tunnel at LSR B, it will be encapsulated in an MPLS header and forwarded to C. C might then do a label switch and forward it to D. At that point, the label is popped and the IP packet emerges from the MPLS tunnel.

In `traceroute` output, router B would appear to be a "normal" layer 3 hop since the packet has not yet been encapsulated, and routers C and D would appear to be MPLS routers. Thus, we would observe a tunnel of length 2.

Consider, however, if the ISP has configured routers to do penultimate hop-popping (PHP), which is basically an optimization to avoid encapsulating a packet that has only one more hop in a tunnel [31], and to reduce label stack popping load on a tunnel egress router. In this case, we would only observe router C to employ MPLS; D would appear as a "normal" layer 3 hop. Given our definition of MPLS tunnel, we consider this to be a one-hop tunnel,

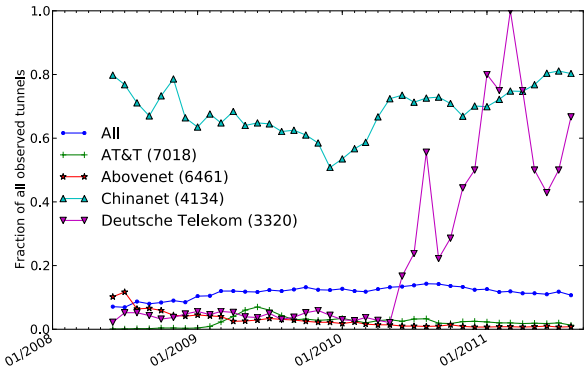

Figure 9: Fraction of tunnels that consist of one hop, over time.

though it is arguable whether this is, in fact, a two-hop tunnel. Given the measurements at hand, we cannot say for certain that an ISP is doing PHP, so the best we can say is that the tunnel is of length 1, and there was one more IP hop in the AS prior to egress.

Next, we examine the distribution of tunnel lengths, both in terms of number of hops, and in latency (milliseconds). Figure 10 shows complementary CDFs for all tunnels, and for three ASes. Note that the y-axis is log scale. Lines in each plot correspond to six month periods, starting in January 2009. We choose this form of plot to emphasize the tail of the distribution. We see that the distribution is skewed: for all tunnels, 90% are 7 hops or fewer, but there are instances of long tunnels, even beyond 15 hops. Plots for individual ASes are somewhat similar to the plot for all ASes. Interestingly, we observe some inflation in path lengths over time for AS6461.

Figure 11 shows similar complementary cumulative distribution function plots of tunnel lengths for all tunnels and the same three ASes, but now considering length in terms of milliseconds. Again, note that the y-axis is log scale. Lines in each plot correspond to six month periods, starting in January 2009. We observe that half of all tunnel lengths are approximately *zero* milliseconds in length, 90% are 150 milliseconds or less. (Refer to Section 2 for an explanation of why we may see close to zero latency across a tunnel.) We also see that some tunnels exceed 300 milliseconds in length. Tunnel latency distributions for the three ASes differ more obviously than the tunnel hop-length distributions. While the majority of tunnels in ASes 7018 and 6461 are relatively short, 10% of the tunnels in the most recent six months for AS6453 exceeded 340 milliseconds.

The above analysis considers only the length of inner-most tunnels, *i.e.*, not nested tunnels. In Figure 12 we plot the distribution of lengths of nested tunnels for all ASes for six time periods (*cf.* the upper left plot in Figure 10). Interestingly, while there is no clear distributional trend in the similar plot in Figure 10, it appears from Figure 12 that nested tunnels appear to be growing longer.

Lastly, we examine the use of multiple depths of label stacks, and the use of different traffic class identifiers. The main uses of label stacking are for supporting VPN services, for more sophisticated traffic engineering, and for LSP protection ("fast reroute") in the event of link or router failure [34]. Since it is highly unlikely that the label stacks we observe are due to VPN services since the Ark nodes are connected to the public Internet, we hypothesize that observed uses of label stacking are due to traffic and network resilience engineering. Figure 13 shows the fraction of tunnels using up to three MPLS labels. Six separate months are shown for all tunnels and for three selected ASes. We see that typically just over 80% of all tunnels have used a stack of only one label, and virtually no ASes use stacks of three labels. (We observed only a single instance of a label stack of four labels, which was AS3549 in April 2011.) The three ASes shown exhibit much different behavior. While AS7018 used only single-level labels in the earliest measurement period, the majority of its tunnels now employ stacks of two labels. AS2119 was one of the few ASes we observed to employ label stacks of depth 3 (we never observed more than 10 ASes to use 3-deep stacks). We also see that it has very few tunnels that have just a single label on the stack. Lastly, we see that AS19262 has made roughly equal use of label stacks with 1 or 2 labels. In future work we hope to gain insight into the operational practices that lead to these observed behaviors.

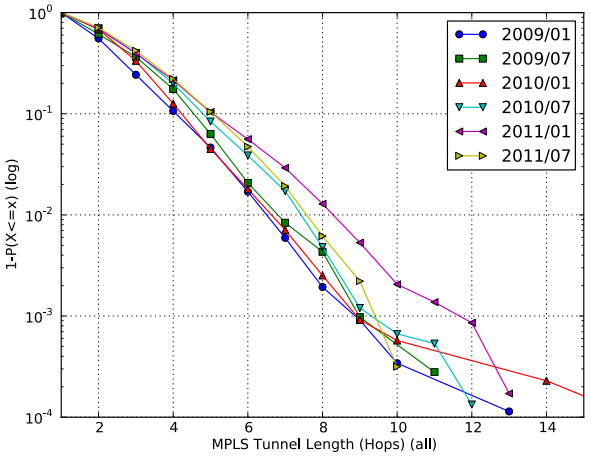

Figure 12: Complementary CDF of the length of all *nested* tunnels (measured in hops). Note that the y-axis is log scale.

The MPLS header contains a 3-bit traffic class field, which can be used by ISPs for implementing different quality of service policies and for prioritizing LSPs. Table 2 shows the fraction of ASes employing different unique traffic class identifiers, over the first month of the past three years. We see that in January 2009, 68% of all ASes used a single traffic class, 22% used two classes, and about 10% used more than two classes. We also see that traffic class identifier usage was roughly the same in January 2010 and 2011.

Table 2: Fraction of ASes employing a given number of traffic class identifiers.

Traffic class labels in use	01/2009	01/2010	01/2011
1	0.682	0.649	0.651
2	0.223	0.224	0.283
3	0.063	0.091	0.033
4	0.010	0.010	0.007
5	0.006	0.006	0.004
6	0.004	0.003	0.003
7	0.005	0.007	0.003
8	0.006	0.006	0.010

Figure 14 shows cumulative probabilities of using one of the 8 traffic class identifiers across all ASes (left) and in selected ASes (right). We see, similar to Table 2, that the majority of ASes used a single identifier of 0. However, a number of ASes used a variety of traffic class identifiers. Indeed, we see that AS6799 and AS6834 used the traffic class identifiers with what appears to be almost equal probability. This is surprising, given the fact that the

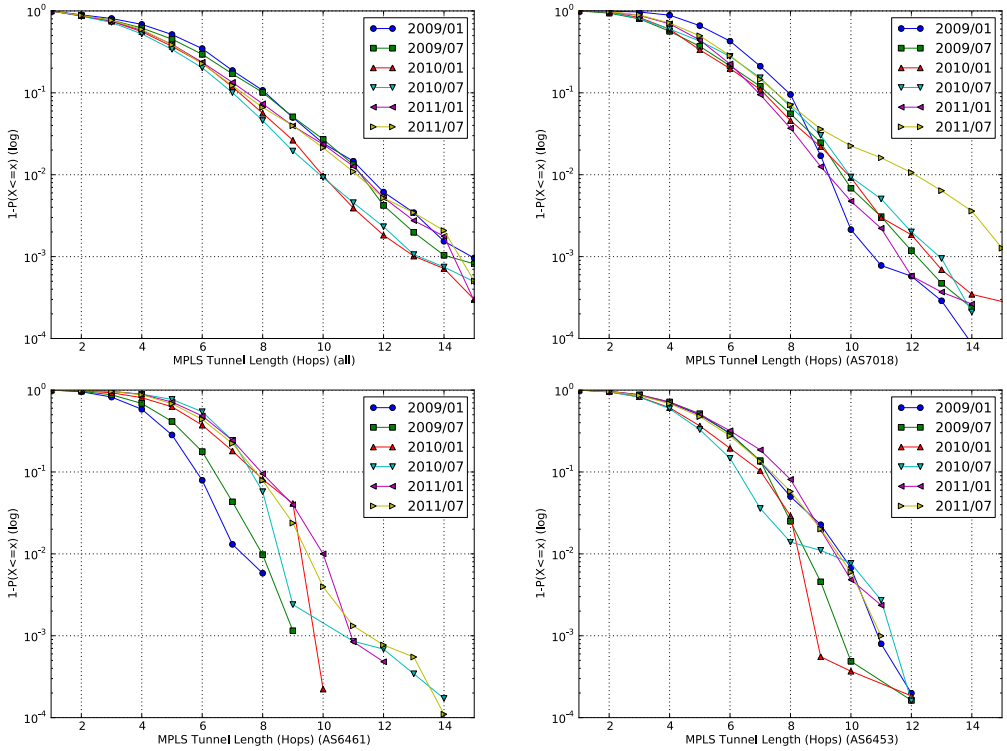

Figure 10: Complementary CDF of length of MPLS tunnels in terms of hops. Note that the y-axis is log scale. Plots shown for all tunnels (upper left) and three selected ASes (7018, 6461, and 6453).

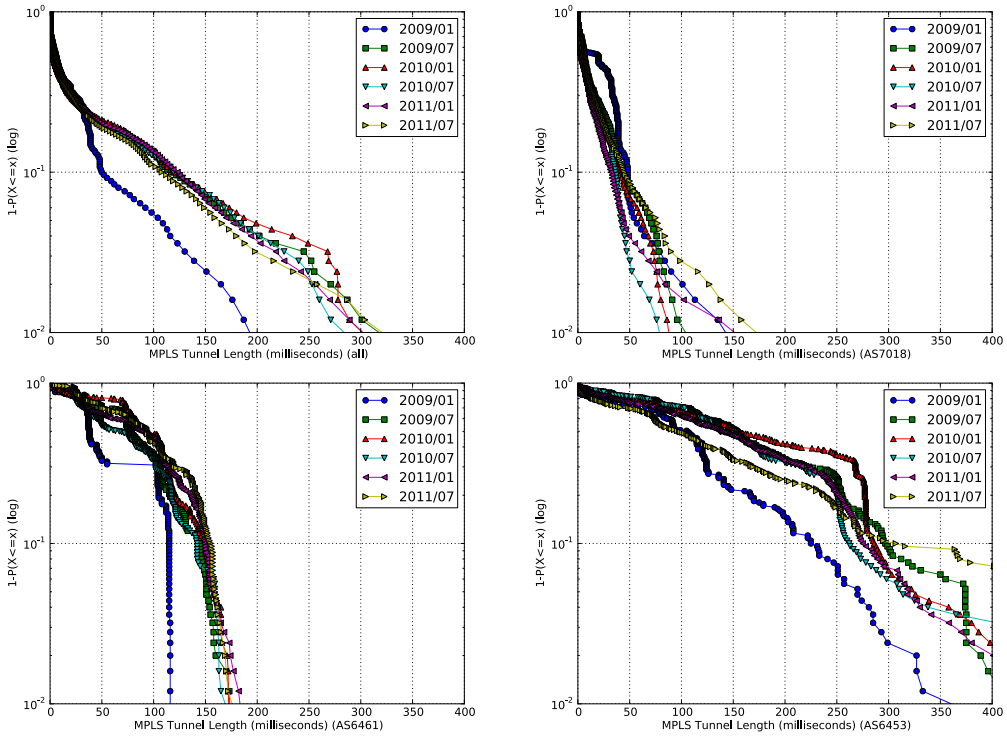

Figure 11: Complementary CDF of length of MPLS tunnels in terms of latency (milliseconds). Note that the y-axis is log scale. Plots shown for all tunnels (upper left) and three selected ASes (7018, 6461, and 6453).

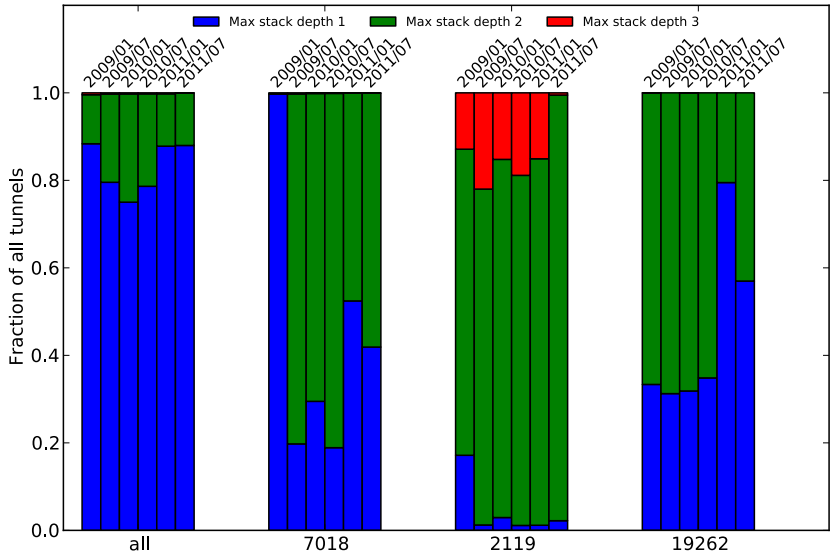

Figure 13: Fraction of tunnels employing different label stack depths, for all tunnels and three selected ASes. Bars for each group are produced from data from one month, as labeled in the plot.

`traceroute` measurements we analyze are produced using the ICMP-paris method over the entire data collection period, *i.e.*, a single type of source traffic provokes different traffic classifications within the same AS. Taken together, the use of label stacks greater than 1 label and more than 1 traffic class identifier by many ASes suggests that they employ some types of traffic classification and traffic engineering methods. While the details of these methods are not clear, in the future we hope to examine these issues more closely by emitting hop-limited packets using different transport protocols and application payloads along paths that contain MPLS tunnels.

4. TUNNEL INFERENCE

Although ICMP extensions enable direct identification of MPLS tunnels in a network, sole reliance on these labels limits the scope of our study. In order to examine the prevalence of MPLS in a wide range of network topologies lacking these MPLS annotations (*e.g.,* the Skitter [13] dataset), this requires the construction of an MPLS inference methodology. We demonstrate how to exploit observable characteristics of MPLS tunnels across various measurement features in order to accurately estimate which paths in a network go through an MPLS tunnel. Note that in this analysis we can only hope to identify tunnels that have been configured in uniform mode, but for which we do not have ICMP extension information to clearly identify the router as being part of an MPLS tunnel.

4.1 MPLS Tunnel Features

Consider observable characteristics of a network path which may indicate a particular router interface is in an MPLS tunnel. Assuming `traceroute`-like probes, we focus on three path properties: latency, hop count, and the IP subnet.

First, consider observations of round-trip time along a `traceroute` path. As previously described, uniform mode MPLS deployments are configured such that the expiration of a packet within a tunnel causes the packet to be forwarded to the end of the tunnel prior to being routed back to the source host. Intuitively, this results in latency observations for interfaces in the same tunnel that are all roughly equal. From mining the Ark dataset, we observe the difference in latency between the interfaces under consideration and the next consecutive interfaces in an observed path. Using kernel density estimators [20], we approximate the probability of observed latency given an interface in an MPLS tunnel (*i.e.,* $\widehat{P}_{lat}^{(k)}(\ell \mid \text{MPLS})$, where ℓ is the observed latency between the interface under consideration and the k-hops away interface in the observed path) and the probability of observed latency given that an interface is *not* in an MPLS tunnel (*i.e.,* $\widehat{P}_{lat}^{(k)}\left(\ell \mid \text{MPLS}^C\right)$). In Figure 15, we clearly demonstrate for the April 2011 MPLS-labeled Ark data that MPLS annotated interfaces have different latency characteristics than non-MPLS interfaces. Specifically, for all three figures we find that MPLS interfaces have pairwise latency closer to zero compared with non-MPLS interfaces, which matches our intuition.

We further assume that interfaces in the same MPLS tunnel will be allocated with similar IP addresses, since all the tunnels we observed reside in the same AS. Therefore, two interfaces consecutively found in a path are more likely to be in an MPLS tunnel if they are closer in IP space (*e.g.,* two interfaces in the same $/24$ are more likely to be in an MPLS than two interfaces in the same $/4$). In Figure 16, we plot the MPLS-labeled April 2011 Ark mined kernel density estimates with respect to the observed IP subnet with the k-th consecutive interface in observed paths (*i.e.,* $\widehat{P}_{IP}^{(k)}(s \mid \text{MPLS})$, where s is the observed IP subnet between the interface under consideration and the k-hops away interface in the observed path) and the probability the interface is *not* in an MPLS tunnel (*i.e.,* $\widehat{P}_{IP}^{(k)}\left(s \mid \text{MPLS}^C\right)$) Again, we find a difference in characteristics between MPLS interfaces and non-MPLS interfaces in this Ark dataset, with MPLS interface having higher IP address subnet compared with non-MPLS interfaces.

Notice that while both of these features show different characteristics for MPLS and non-MPLS interfaces, neither are powerful enough to accurately classify using a single component of information alone. This motivates fusing multiple observed characteristics from the observed path measurement. Specifically, we exploit

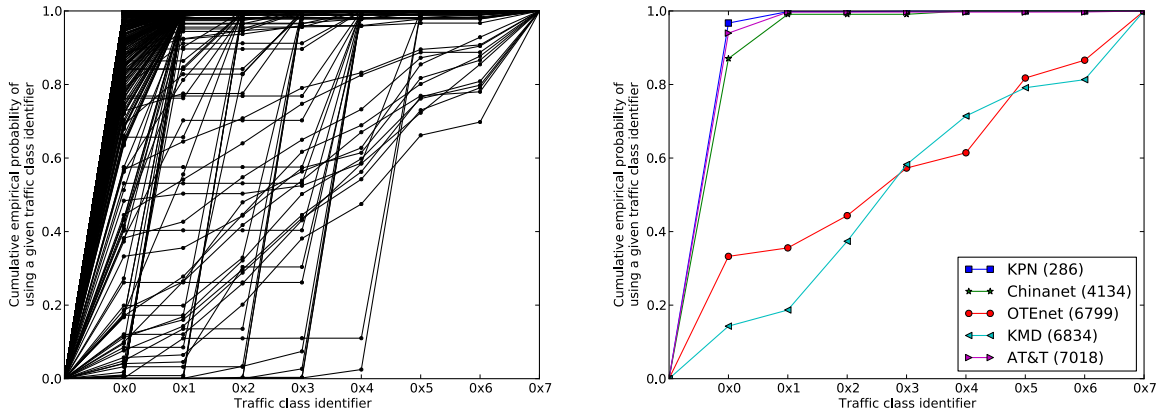

Figure 14: Cumulative empirical probabilities of using different traffic class identifiers for all ASes (left) (each curve corresponds to one AS) and for tunnels in selected ASes (right).

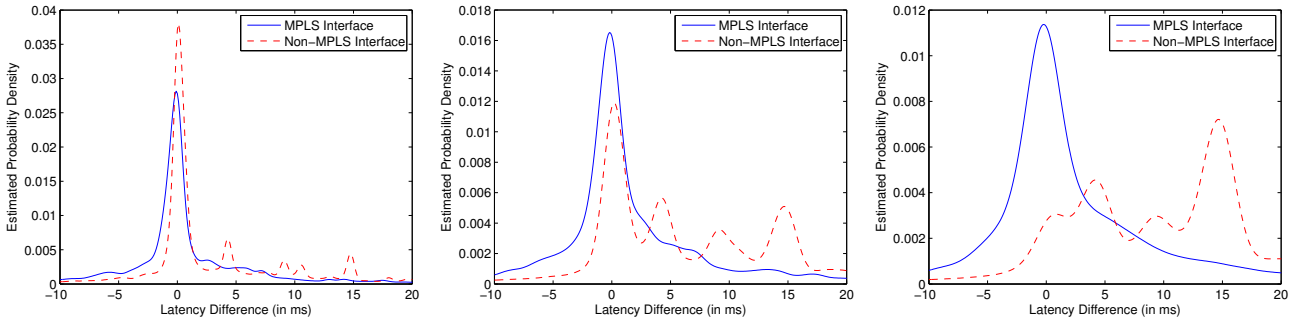

Figure 15: Estimated latency-based distributions between non-MPLS and MPLS interfaces, $\widehat{P}_{lat}^{(k)}(\ell \,|\, \text{MPLS})$ and $\widehat{P}_{lat}^{(k)}(\ell \,|\, \text{MPLS}^C)$, respectively. Where ℓ is the latency between interfaces separated by k hops in observed `traceroute` paths. (Left) interfaces separated by one hop, $k=1$, (Center) interfaces separated by two hops, $k=2$, and (Right) interfaces separated by three hops, $k=3$.

the continuous property of MPLS tunnels in terms of observed hop count. For example, for a path with three interfaces in the same MPLS tunnel, all three of these interfaces must be observed consecutively. This property both reinforces our classification and helps us distinguish between MPLS tunnels and simply an observation of co-located routers in the same PoP along a `traceroute` path.

4.1.1 *Bayesian Data Fusion Methodology*

We state the probability of an interface i being in an MPLS tunnel given our observed measurements ($\mathscr{M}_i$, the collection of latencies and IP subnets) for this interface, $P(\text{MPLS} \,|\, \mathscr{M}_i)$, as

$$\begin{aligned} P(\text{MPLS}_i) &= P(\text{MPLS} \,|\, \mathscr{M}_i) \\ P(\text{MPLS}_i) &\propto P(\mathscr{M}_i \,|\, \text{MPLS})\, P(\text{MPLS}) \end{aligned}$$

Using Bayes Rule (*i.e.,* $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$) and where $\mathscr{M}_i$ is the set of observed features from the K interfaces found before and after the interface i in the observed path.

We do not argue that these are the only features that can distinguish MPLS paths; there may indeed be others. This issue is a topic for future investigation.

Unfortunately, as more information is brought to bear on this problem (*i.e.,* the set of features, $\mathscr{M}_i$, grows), the higher the dimension of the probability distribution $P(\mathscr{M}_i \,|\, \text{MPLS})$ becomes, which results in a computationally intractable problem [20]. In addition, to exploit multiple characteristics from interfaces along the path, it is necessary to fuse together disparate data types (*i.e.,* latency, IP subnet, etc.), for which many off-the-shelf density estimation procedures were not designed. To avoid these limitations, we look to a *Naive Bayesian* data fusion approach. The Naive Bayesian approach converts the problem from estimating one M-dimensional density (which may be computationally intractable) to estimating M one-dimensional densities, such as those estimated in Figure 15 and Figure 16.

Using Naive Bayes, the resulting log-likelihood estimated probability for an interface i being in an MPLS tunnel is formulated as:

$$\begin{aligned} \log \widehat{P}(\text{MPLS}_i) = &\sum_{k=-K}^{K} \log \widehat{P}_{lat}^{(k)}\left(\ell_{i,i+k} \,|\, \text{MPLS}\right) \\ &+ \sum_{k=-K}^{K} \log \widehat{P}_{IP}^{(k)}\left(s_{i,i+k} \,|\, \text{MPLS}\right) \end{aligned} \tag{1}$$

While the log-likelihood estimated probability for an interface i *not being* in an MPLS tunnel is formulated as :

$$\begin{aligned} \log \widehat{P}(\text{MPLS}_i) = &\sum_{k=-K}^{K} \log \widehat{P}_{lat}^{(k)}\left(\ell_{i,i+k} \,|\, \text{MPLS}^C\right) \\ &+ \sum_{k=-K}^{K} \log \widehat{P}_{IP}^{(k)}\left(s_{i,i+k} \,|\, \text{MPLS}^C\right) \end{aligned} \tag{2}$$

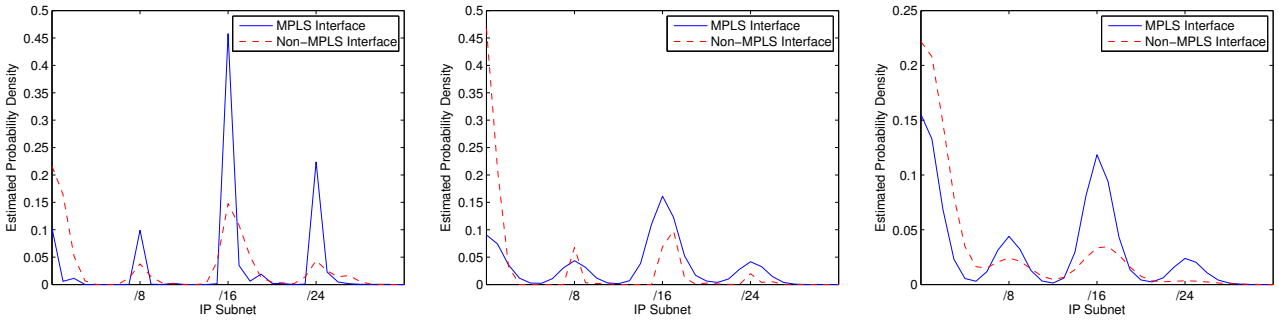

Figure 16: Estimated IP subnet-based distributions between non-MPLS and MPLS interfaces, $\widehat{P}_{IP}^{(k)}(s \mid \text{MPLS})$ and $\widehat{P}_{IP}^{(k)}(s \mid \text{MPLS}^C)$, respectively. Where s is the IP subnet between interfaces separated by k hops in observed `traceroute` paths. (Left) interfaces separated by one hop, $k=1$, (Center) interfaces separated by two hops, $k=2$, and (Right) interfaces separated by three hops, $k=3$.

Where K is the number of hops in the observed `traceroute` path that we examine with respect to the interface i.

From this formulation, we see that the computational complexity of this methodology for each interface is only $O(M)$, where M is the number of path features under consideration. On a reasonably equipped host, we find that to resolve the estimated MPLS probability for every interface along a path takes on average 6 milliseconds.

Prior work on using Naive Bayes methodologies on Internet measurements have been explored in the context of IP geolocation in [18]. Our approach differs here through the use of path-based measurements (as opposed to only end-to-end measurements in the prior work) and application (MPLS identification vs. geolocation). We direct the readers to this prior work for a detailed introduction to the Naive Bayes approach.

4.2 Inference Experimental Results

We now consider the ability to estimate MPLS tunnels on an Ark dataset from April 2011 containing over 9 million `traceroute` paths. Due to the use of our learning-based Naive Bayesian inference approach, hold out cross validation [36] is performed to avoid potential bias in our results. This is performed by holding out 5% of the observed paths (randomly selected) as training data, learning the MPLS characteristics using kernel density estimators, then testing our Bayesian inference method on the remaining 95% of the observed paths. The detection accuracy results presented are with respect to this held-out test set.

4.2.1 MPLS Interface Detection

First, we consider the performance of our methodology with respect to detecting if an interface is in an MPLS tunnel given an observed path containing that interface. Using our Naive Bayesian approach, we fuse together latency, hop count, and IP subnet information to estimate the log-likelihood probability an interface is in an MPLS tunnel. In order to classify our test set of interfaces, we consider a simple thresholding approach between the estimated probability the interface is in an MPLS, against the estimated probability it is not in an MPLS tunnel. The intuition is that the more likely an interface is to be in an MPLS tunnel, the larger this margin will be. Given a set threshold λ, an interface i is assigned,

$$i \in \begin{cases} \text{MPLS} & : \log \widehat{P}(\text{MPLS}_i) - \log \widehat{P}\left(\text{MPLS}_i^C\right) \geq \lambda \\ \text{MPLS}^C & : \log \widehat{P}(\text{MPLS}_i) - \log \widehat{P}\left(\text{MPLS}_i^C\right) < \lambda \end{cases}$$

Using hold-out cross validation on our set of over 9 million paths, the false alarm/detection characteristics for detecting if interfaces in our test set are in MPLS tunnels can be seen in Figure 17. The figure shows the region operating characteristics (ROC) curve of our classifiers across all feasible values of the threshold λ (where each point in the figure represents a different value of λ and the associated false alarm and detection rate). These results include our full Naive Bayes data fusion technique (using both latency and IP subnet information), Naive Bayes using only IP subnet information, and Naive Bayes using only latency information. We find that our full technique can detect over 55% of the MPLS interfaces with only declaring 10% of the non-MPLS interfaces incorrectly, a significant improvement over using a single characteristic.

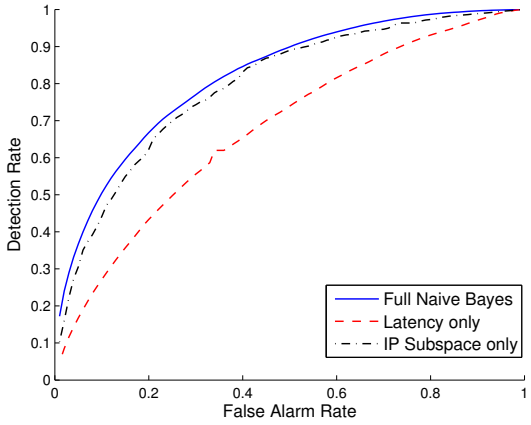

Figure 17: Comparison of the full Naive Bayesian methodology with using only latency or IP subnet to detect MPLS interfaces using `traceroute` paths from the April 2011 Ark Dataset.

4.2.2 MPLS Path Detection

More valuable to a network operator may be knowledge of whether or not a given `traceroute` path has traversed an MPLS tunnel. In order to classify each observed path as an *MPLS path* (*i.e.*, a path that crossed an MPLS tunnel) or a *non-MPLS path*, we consider using the characteristics inferred on each of the interfaces along the observed path. Specifically, we consider aggregating the interface information in the form of the maximum interface log-likelihood probability. Such that, for an observed path

$\mathbf{p}_j = \{p_1, p_2, ..., p_M\}$ we find,

$$\log \widehat{P}\left(\text{MPLS}_{\mathbf{p}_j}\right) = \max_{i=\{1,2,...,M\}} \left(\log \widehat{P}\left(\text{MPLS}_{p_i}\right) - \log \widehat{P}\left(\text{MPLS}^{C}_{p_i}\right)\right)$$

Again we threshold these margin values against a parameter λ to classify each path as MPLS or non-MPLS. As expected, the length of the MPLS tunnel is directly related with the ability to detect if the path does or does not go through an MPLS tunnel. Separating out with respect to MPLS tunnel length, we find the detection characteristics in Figure 18 across the specified April 2011 Ark dataset and all feasible values of λ.

For MPLS tunnels of length greater than or equal to 4, we can detect roughly 80% of the MPLS paths with only a 10% false alarm rate. This is in contrast to considering all MPLS tunnels, where we only detect 35% of the tunnels for the same false alarm rate.

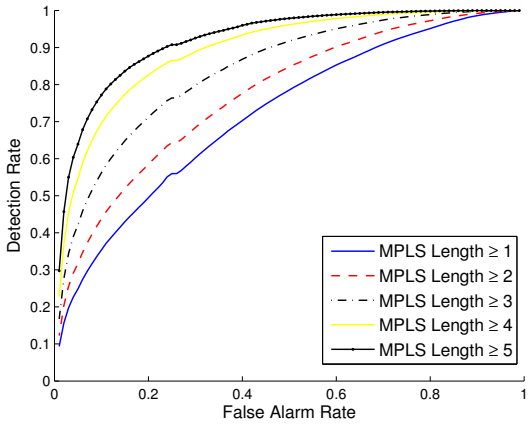

Figure 18: Comparison of the MPLS path detection methodology with respect to the length of the MPLS tunnel using the April 2011 Ark Dataset.

4.3 Unlabeled MPLS Path Estimation

Finally, we examine the performance on prior Ark data that does not have MPLS labels. We train our classifier using a subset of labeled June 2008 Ark data, and then test on the prior months (March, April, and May 2008) for which we have no ground truth. We present our results with respect to the four most frequently encountered ASes in our training set. In Table 3, we find the results of our MPLS path detection in terms of the percentage of observed paths for a given autonomous system that we estimate to include an MPLS tunnel. The table shows the percentage of inferred MPLS tunnels as gross characteristics consistent with the labeled June 2008 data. The level of variability in our inference is consistent with the variability that we see in the labeled MPLS data.

While our pre-June 2008 data does not contain uniform-mode MPLS labels, we do have the ability to examine paths that are found in *both* unlabeled pre-June 2008 data and labeled June 2008 data. Using the June 2008 MPLS labels as ground truth, we can examine performance of our inference methodology on an intersection of these two sets. Figure 19 shows the performance of our inference technique on unlabeled Ark data from March, April, and May 2008. As expected, the classifier performance is more accurate for time frames closer to the labeled set (*i.e.,* May dataset), when compared with the more temporally distant unlabeled set (*i.e.,* March dataset).

Table 3: Comparison of estimated percent of MPLS paths for unlabeled Ark data.

	Estimated Paths with Tunnels				Obs. Data
AS	03/08	04/08	05/08	06/08	06/08
7018	60.7%	59.2%	76.7%	62.5%	69.2%
6389	54.2%	67.5%	66.4%	69.5%	71.0%
4134	7.1%	8.1%	8.5%	3.1%	0.4%
4230	35.5%	27.0%	29.8%	29.4%	25.8%

5. RELATED WORK

The literature on MPLS largely falls into three categories. The first are studies that describe methods for expanding and enhancing MPLS beyond the original RFCs describing the protocol. Examples include methods for improving reliability and fault tolerance (*e.g.,* [15, 25]) and extension for a wide range of label switched paths including photonic networks (*i.e.,* GMPLS [11]) and wireless networks (*e.g.,* [23]). The second category are studies that describe methods for employing MPLS to meet various operational objectives within a given network infrastructure. The most prominent among these are studies that describe a wide variety of traffic engineering methods based on MPLS (*e.g.,* [17, 37]). In the third category are studies describing new routing algorithms that can be used in conjunction with MPLS to establish paths with target characteristics (*e.g.,* [35]).

Details of the MPLS protocol are described in various RFCs which are all linked from the IETF's MPLS working group homepage [3]. The main MPLS architectural reference is RFC 3031 [31], and the most relevant standards document to our study is RFC 4950, which defines the ICMP extensions that enable label stacks to be returned to clients [12]. Beyond RFC documents, Davie and Rekhter wrote a comprehensive textbook reference on MPLS that broadly treats the protocol [16]. There are also numerous online references and notes on practical aspects of MPLS configuration and management (*e.g.,* [34])

We are aware of no prior studies on Internet-wide MPLS deployment characteristics. Perhaps the most relevant empirical studies were by Sherwood *et al.* in [32, 33]. The former study describes a method for measuring router-level topologies that includes the capability to discover MPLS nodes using ICMP extensions. That study provides a small set of measurements on routers that respond with MPLS labels. The latter study also discusses certain aspects of MPLS, again in the context of router-level topology discovery. Our study differs from theirs in its specific focus on MPLS and the longitudinal characterization of its deployment.

6. SUMMARY AND CONCLUSIONS

MPLS offers compelling capabilities for traffic engineering, multi-services networks and improved network robustness. In this paper, we describe a longitudinal study of the prevalence and characteristics of MPLS deployments in the open Internet. The goal of our work is to establish a comprehensive baseline for the evolution and current status of MPLS deployments since they have important implications for issues such as quality of service and network neutrality. We use the large `traceroute` archive from the Ark project as the basis for our work. Over the past 3 years, this data set has included MPLS label stacks, which enable direct evaluation of MPLS tunnels configured in uniform mode. We note again that our direct observations likely underestimate MPLS deployments due to the inability to empirically identify pipe-mode tunnels.

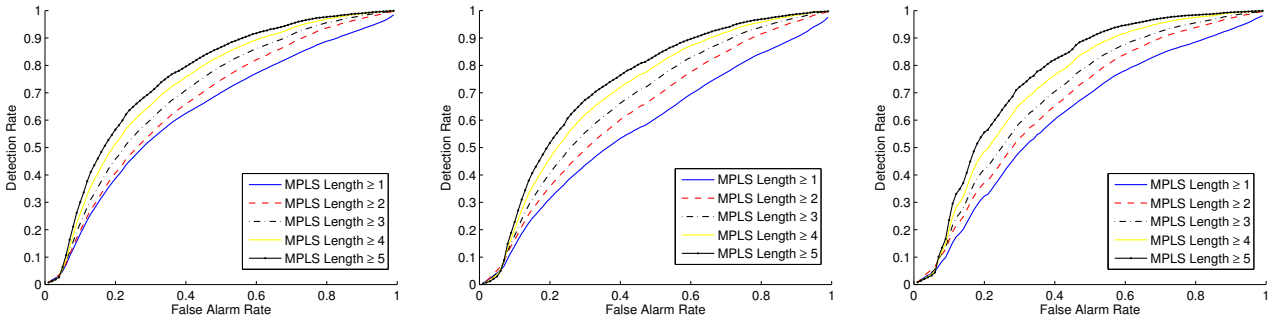

Figure 19: Comparison of the MPLS path detection of unlabeled data using June 2008 labels. (Left) - March 2008 Ark Dataset, (Center) - April 2008 Ark Dataset, (Right) - May 2008 Ark Dataset

The summary findings of our analysis show an increasing trend in MPLS deployments over the past three years and the wide use by tier-1 providers, implying that it is increasingly likely that packets will encounter an MPLS tunnel on end-to-end paths. We also find that tunnels are likely to span the entire edge-to-edge distance of a transit provider, with typical transit times on the order of tens of milliseconds. Lastly, our examination of the use of traffic classifiers indicates that while multiple classes are not uncommon, the diversity of classes has not changed over the past three years.

We develop an MPLS tunnel inference method that is based on the observation that for certain configurations, RTTs for internal hops will all be similar. Our inference method uses Bayesian data fusion to efficiently identify MPLS paths based on RTTs. Using the labeled data, we demonstrate that this method can indeed identify MPLS paths with high accuracy.

In future work, we plan to investigate MPLS deployments in additional `traceroute` data archives using our Bayesian inference method. A short term target is the Skitter data set, which would give us the opportunity to investigate MPLS deployments over a longer time period. We also plan to expand our survey by conducting more targeted investigations using distributed infrastructures such as Planetlab.

7. ACKNOWLEDGMENTS

This work was supported in part by NSF grants CNS-0716460, CNS-0831427 and CNS-0905186, and NSF CAREER award NSF-1054985. Any opinions, findings, conclusions or other recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the NSF.

Support for the IPv4 Routed /24 AS Links Dataset is provided by the NSF, the US Department of Homeland Security, the WIDE Project, Cisco Systems, and CAIDA Members. We sincerely thank CAIDA for making these high quality data available to the community.

8. REFERENCES

[1] Routeviews Prefix to AS mappings Dataset (pfx2as). `http://www.caida.org/data/routing/routeviews-prefix2as.xml`.

[2] Cyclops. `http://irl.cs.ucla.edu/topology/data/`, Accessed August 2011.

[3] IETF Multiprotocol Label Switching (mpls) Working Group. `http://datatracker.ietf.org/wg/mpls/charter/`, Accessed August 2011.

[4] Junos OS MPLS Applications Configuration Guide. `http://www.juniper.net/techpubs/en_US/junos11.1/information-products/topic-collections/config-guide-mpls-applications/config-guide-mpls-applications.pdf`, Accessed August 2011.

[5] Multiprotocol Label Switching on Cisco Routers. `http://www.cisco.com/en/US/docs/ios/12_1t/12_1t3/feature/guide/rtr_13t.pdf`, Accessed August 2011.

[6] Sprint IP/MPLS Network Maps. `https://www.sprint.net/network_maps.php`, Accessed August 2011.

[7] The Internet Topology Zoo. `http://www.topology-zoo.org/`, Accessed August 2011.

[8] The Traceroute Command in MPLS. `http://www.cisco.com/en/US/tech/tk436/tk428/technologies_tech_note09186a008020a42a.shtml`, Accessed August 2011.

[9] P. Agarwal and B. Akyol. RFC 3443: Time to Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks. `http://www.ietf.org/rfc/rfc3443.txt`, January 2003.

[10] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. RFC 2702: Requirements for Traffic Engineering Over MPLS. `http://www.ietf.org/rfc/rfc2702.txt`, September 1999.

[11] L. Berger. RFC 3473: Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffiic Engineering (RSVP-TE) Extensions. `http://www.ietf.org/rfc/rfc3473.txt`, January 2003.

[12] R. Bonica, D. Gan, D. Tappan, and C. Pignataro. RFC 4950: ICMP Extensions for Multiprotocol Label Switching. `http://www.ietf.org/rfc/rfc4950.txt`, August 2007.

[13] CAIDA. The Skitter Project. http://www.caida.org/tools/measurement/skitter/, Accessed August 2011.

[14] R. Callon, P. Doolan, N. Feldman, A. Fredette, and G. Swallow. Draft: A Framework for Multiprotocol Label Switching. http://tools.ietf.org/html/draft-ietf-mpls-framework-00, May 1997.

[15] H. Chengcheng, V. Sharma, and K. Owens. Building Reliable MPLS Networks Using a Path Protection Mechanism. *IEEE Communications Magazine*, 40(3), March 2002.

[16] B. Davie and Y. Rekhter. *MPLS: Technology and Applications*. Morgan Kaufmann, 2000.

[17] A. Elwalid, C. Jin, S. Low, and I. Widjaja. MATE: MPLS Adaptive Traffic Engineering. In *Proceedings of IEEE INFOCOM '01*, April 2001.

[18] B. Eriksson, P. Barford, J. Sommers, and R. Nowak. A Learning-based Approach for IP Geolocation. In *Proceedings of Passive and Active Measurements Conference*, April 2010.

[19] F. Le Faucheur *et al.* RFC 3270: Multi-Protocol Label Switching (MPLS) Support of Differentiated Services. http://www.ietf.org/rfc/rfc3270.txt, May 2002.

[20] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[21] Y. Hyun. Personal communication, May 2011.

[22] Y. Hyun, B. Huffaker, D. Andersen, E. Aben, M. Luckie, kc claffy, and C. Shannon. The IPv4 Routed /24 AS Links Dataset: January 2008–August 2011. http://www.caida.org/data/active/ipv4_routed_topology_aslinks_dataset.xml.

[23] H. Kim, K. Wong, W. Chen, and C. Lau. Mobility-aware MPLS in IP-based Wireless Access Networks. In *Proceedings of IEEE Globecom '01*, November 2001.

[24] K. Kompella and G. Swallow. RFC 4379: Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures. http://www.ietf.org/rfc/rfc4379.txt, February 2006.

[25] S. Lee and M. Gerla. Fault Tolerance and Load Balancing in QoS Provisioning with Multiple MPLS Paths. In *Proceedings of the 9th International Workshop on Quality of Service*, June 2001.

[26] M. Luckie. Scamper: a Scalable and Extensible Packet Prober for Active Measurement of the Internet. In *Proceedings of ACM SIGCOMM Internet Measurement Conference*, November 2010.

[27] M. Luckie. Personal communication, May 2011.

[28] M. Luckie, Y. Hyun, and B. Huffaker. Traceroute probe method and forward IP path inference. In *Proceedings of ACM SIGCOMM Internet Measurement Conference*, pages 311–324, October 2008.

[29] P. Newman, W. Edwards, R. Hinden, E. Hoffman, C. Liaw, T. Lyon, and G. Minshall. RFC 1953: Ipsilon Flow Management Protocol Specification for IP. http://www.ietf.org/rfc/rfc1953.txt, May 1996.

[30] Y. Rekhter, B. Davie, D. Katz, E. Rosen, and G. Swallow. RFC 2105: Cisco's Tag Switching Architecture Overview. http://www.ietf.org/rfc/rfc2105.txt, February 1997.

[31] E. Rosen, A. Viswanathan, and R. Callon. RFC 3031: Multiprotocol Label Switching Architecture. http://www.ietf.org/rfc/rfc3031.txt, January 2001.

[32] R. Sherwood, A. Bender, and N. Spring. Discarte: a disjunctive internet cartographer. In *Proceedings of ACM SIGCOMM '08*, pages 303–314, August 2008.

[33] R. Sherwood and N. Spring. Touring the Internet in a TCP Sidecar. In *Proceedings of ACM SIGCOMM Internet Measurement Conference*, pages 339–344, October 2006.

[34] R. Steenbergen. Nanog Tutorial: MPLS for Dummies. http://www.nanog.org/meetings/nanog49/presentations/Sunday/mpls-nanog49.pdf, June 2010.

[35] B. Wang, S. Xu, and C. Chen. A New Bandwidth Guaranteed Routing Algorithm for MPLS Traffic Engineering. In *Proceedings of ICC '02*, April 2002.

[36] L. Wasserman. All of nonparametric statistics (springer texts in statistics). Springer, May 2007.

[37] X. Xipeng, A. Hannan, B. Bailey, and L. Ni. Traffic Engineering with MPLS in the Internet. *IEEE Network*, 14(2), March/April 2000.

Summary Review Documentation for

"On the Prevalence and Characteristics of MPLS Deployments in the Open Internet"

Authors: J. Sommers, B. Eriksson, P. Barford

Reviewer #1

Strengths: First large scale Internet wide study of the deployment of MPLS.
Proposal of a reasonable approach to detect some of the hidden tunnels.

Weaknesses: Some of the MPLS deployments can be completely transparent to traceroutes even with the proposed methodology to detect some cases of hidden deployments. Comparison of routing data with stock market data...

Comments to Authors: I am glad to see that finally someone spends time on this topic; it is an important topic that has been neglected for a long time.

Figure 2 and general comments about the deployment of MPLS: you should really break down the statistics by AS type. For instance, you could use the "Cyclops" classification and provide stats for Tier 1 ISPs, Large ISPs or stub ASs. I think this would provide a better view about the extent of the deployment of MPLS. My guess is that your 7% would significantly increase if you were to look at Tier 1 ISP or large ISPs. You might want to summarize it as at least 7% of all ASs and X% of large ISP or Tier 1 ISPs.

While you do a good job in the paper of explaining the caveats (there are cases where you can simply not detect MPLS), I think you need to highlight this also better in the abstract, intro and conclusion. This is all the more important as you propose a heuristic to detect MPLS deployments, so some readers might think that your heuristic covers all the cases and your 7% is a good approximation.

In the intro, you should also mention that multi service networks go beyond MPLS VPNs, MPLS also enables IPv6 and VPLS.

PLEASE remove Fig 3. Don't compare routing data to the stock market (or the weather!). Trying to correlate the two doesn't make any sense to me. Furthermore, Fig 4 already shows that AT&T experiences a significant drop in early 2009 (how significant?). How would Fig 3 look like without AT&T? It seems that you simply detected a major architecture change within AT&T around that period.
Did you contact Deutsche Telekom to explain what you are observing on Fig 4?

You mention the issue of PHP and how it impacts the length of the tunnel. That's a great observation. But it seems to me that you could go further and try to detect the use of PHP by looking at traces within an ISP in 2 opposite directions (the path will be asymmetrical so it will complicate things but I'm sure you can work something out).

On traffic classes: you should highlight again that this is for public Internet traffic. It is possible that ISPs use one class for their Internet traffic but use another 3 other classes for otehr services like MPLS VPN that you cannot detect.

Regarding AS6799 and AS6834: you should check if the traffic class for these ISPs depend on the the previous AS hop: maybe the traffic class is X if the traffic enters from AS Y and traffic class Z if the traffic enters from AS A because one is a peer and one is a customer. Just an idea.

You claim that you can detect 80% of the MPLS paths that are longer than 4. Is Fig 4 not showing that most of the MPLS paths are shorter than 4 hops? Could you quantify it here?

Reviewer #2

Strengths: Internet topology studies have ignored MPLS until now. Only anecdotal evidence about MPLS deployment was discussed in the community. This study is among the first that try to measure how many ASes use MPLS and also how it is used. The paper give a detailed introduction into the methodology and the measurement pitfalls.

This paper is critical to every researcher working on any traceroute-based topology measurements.

Weaknesses: The study is a bit premature. While it nicely discusses methodology and in how many ASes MPLS has been detected. it relies on CAIDA's ARK data, which collects traceroutes from a set of specific locations. This results in measurements, which are heavily biased towards those observation points. The authors do not provide any insights into how suitable this dataset is for studying MPLS deployment and what the limitations are. The paper does not make any effort into addressing this issue at all (e.g., not even as an explanatory warning to the reader). Many observations in this paper are described in relation to "number of observed MPLS tunnels". This, of course heavily relies on the used data and is of little insight to the reader in what it really means.

Comments to Authors: This is a very well written paper, and relevant for everyone who is trying to understand the Internet topology. While the paper could do a better job in explaining what the authors know about the limitations and the measurement bias, this is overall a well done study. It includes an excellent introduction into what is relevant to know for a measurement

researcher if he/she wants to include MPLS measurements into their work.

As mentioned above, it is difficult to get a good feeling for the data. Talking about 350K MPLS tunnels or hearing that 20% of all ASes have fewer than 10 tunnels, is relatively meaningless overall. For a different set of observation points those numbers could look different. The reader lacks a bit perspective of how relevant or irrelevant this is. The paper provides little topological information, and it remains unclear what the quality or suitability of using CAIDA's probing is for the purpose of this study.
Admitted, the authors are not able to do things differently, as this would easily lead into areas of pure speculations and could introduce a strong bias in the data. However, the authors could have added a section that gives a bit more intuition about the CAIDA ARK data. For example, AT&T relies heavily on MPLS (as the paper shows), but could there be a region in Asia which uses as much MPLS at AT&T, but due to the lack of measurement points this might not be detectable?

It remained a mystery to me why the authors develop the MPLS detection method in "regular" traces, validate it, and then study 3 additional month in 2008. This data does not add any new insights! What would have been nice is a long-term analysis of, for example skitter data; and trying to detect properties of the early adaptors.

Reviewer #3

Strengths: Longitudinal, high-quality data of something the IMC community should know more about, but does not. The high deployment rate of MPLS underscores the importance of this type of work.

Weaknesses: None? The text failed to properly emphasize that all data here is inherently a lower bound on MPLS deployments due to the measurement technique.

Comments to Authors: Great paper - MPLS is a big blind spot for the academic community and this paper does a lot to try to address it. First and foremost, this paper quantifies how prevalent MPLS deployments are. I would have loved to have seen the deployment numbers broken down by tier-1 (perhaps defined by http://en.wikipedia.org/wiki/Tier_1_network#List_of_tier_1_networks, for a common definition), tier-2, etc.

I suspect that the numbers presented here (i.e, 25% of all hops go through an MPLS tunnel) is in fact a significant underestimate due to pipe-mode tunnels, so I suggest the authors next try to work on this for their future work. As a basis, one could consider extending the Bayesian classifier to look at more characteristics of MPLS vs. non-MPLS links, as suggested in the text. Also, note that L1 often imposes physical limitations on real link lengths that would appear violated by an MPLS-tunnel, so that might be another direction to look.

Regarding MPLS-usage tracking the DOW, my guess is that if you were to do the homework, you would find that your data better correlated with ISP mergers or out-right bankruptcies, which transitively tracks the DOW, rather than some more direct interaction. You should be able to correlate this in your data by looking at unique ASes in the same period. Interesting in any case.

The break down of tunnels by class is a very interesting addition, and should be listed as a first class contribution. The authors might consider correlating it with existing traffic studies or talking to the operators directly to discover what the traffic classes are used for, e.g., VoIP, IPTV, best-effort, etc. I suspect that if you do so, you might be able to dig more information from the data, e.g., quantify the extent of IP-TV deployments or some such.

From the text, it was not clear if the techniques in S4 were being applied to both pipe-mode and uniform-mode tunnels. If both, then can you quantify the fraction of pipe vs. uniform tunnels? That is, the fraction of paths that according to the classifier appear to be going through an MPLS tunnel, but don't have the ICMP MPLS information? This might help shed some light on the pipe vs. uniform deployments.

Reviewer #4

Strengths: Despite its importance as one of the most important new lower level technologies of the evolved Internet, I am not aware of another large scale measurement study of MPLS on the open Internet.
Clear quantifications: 7% ASes employing MPLS, average length of tunnels 3-4 hops, 25% of paths crossing at least one tunnel, ~30% of tunnels employing more than 1 traffic class.
The tunnel inference method based on roundtrip times and TTLs is smart but it is not clear how many tunnels fall in this category (uniform mode but without MPLS extensions enabled.)

Weaknesses: Although the reports are clear and the inference technique interesting the paper is rather flat in terms of conclusions. The connection to net neutrality is rather superficial and overall the authors fails to connect the reported numbers with any bigger questions about the state of the current Internet and the dilemmas regarding its evolution.

Comments to Authors: Good solid paper with interesting numbers and a bit of methodological contribution on inference based on deep understanding of the workings of the protocol. Two issues mainly for me: I wish the paper connected better its findings with some more general questions about the Internet (e.g., to what extent MPLS, covers design goals claimed as motivation for clean slate re designs). Also, I find the paper a bit sparse in terms of results. Maybe it would have been more interesting as a short paper. Also I would like to have seen some more discussion regarding the limitations of the measurement (e.g., an estimation of the MPLS tunnels in pipe mode that go unnoticed).

Reviewer #5

Strengths: - Well written and first of the kind study of MPLS.
- The authors observe interesting technical challenges in detecting these MPLS tunnels and propose new solutions to overcome such problems.

Weaknesses: - Some of the problems involving visibility are fundamental and the heuristics proposed work only for a limited set of scenarios.
- Due to lack of ground truth, some of the reasons mentioned are largely speculative and there is no way to verify.

Comments to Authors: There are a lot of reasons to accept this paper. (1) The paper focuses measurement-based characterization

of MPLS prevalence, which has not received too much attention in the past. (2) The paper does a good job exposing and outlining what information can be observed about MPLS tunnels using traceroute data already collected by the Ark project. (3) Some of the findings of this study are interesting, although the paper does not confidently specify the reasons, but that's okay.

One constructive criticism: This study would tremendously benefit from engaging some large ISP into the mix. The authors have lot of contacts from their prior research with research labs that may provide this necessary angle. For example, the authors look at the MPLS tunnels in the AT&T network; it might be interesting to get some operational insight into what the operational reasons for the increase/decrease/variation of the number of tunnels and other characteristics.

I like Section 2 that provides a good overview of the MPLS state-of-the-art and stuff. The section however focused on measurement techniques from outside the AS. I believe there are more direct tools at the MPLS layer for measurement within the AS (for example, for ISPs to troubleshoot) such as the MPLS trace route. It would be good to mention some of those as well in this section, although I understand we, as researchers, cannot obtain that data unless we collaborate with the ISP.

There was some mention of the ISP MPLS maps, but comparisons are largely missing. It would be helpful to see if the findings in this paper are consistent with those MPLS maps.

In Sec. 3.2, the statement "While the rate of increase in ASes that use MPLS is greater than the rate of increase of all ASes, the fraction of ASes employing MPLS has remained roughly constant at 7%" is absurd. If the numerator changes at a faster rate than the denominator, how can the ratio remain fixed? The correlation with Dow Jones is quite interesting! The statement "So, while it is apparent from Figure 2... roughly correlate with economic conditions" is quite confusing and appears quite absurd. Please elaborate on this.

Figure 8: It would be nice if you discussed with one of the ISPs to see what the operational reasons might be.

I found the description of the nested tunnels devoid of any intuition. It would be good to give some reasons or intuition why somebody would opt for nested tunnels of depth 2-3 (or even 4). Also, are the outer labels reused across two LSPs?

Response from the Authors

We thank the reviewers for their comments and suggestions.

In response to the reviews, we have (1) emphasized in additional places in the paper that our results are limited to uniform-mode tunnels and that it is almost certain (though impossible to quantify) that we underestimate MPLS deployments, (2) added analysis of MPLS deployments by AS type (i.e., tier 1, large ISP, small ISP, and stub networks), (3) expanded our inference analysis to include more unlabeled Ark data and (4) fixed a number of minor textual issues raised. Moreover, we have extended the time period over which we analyzed Ark data to August 2011 in order to provide the most up-to-date view of MPLS activity possible within the time constraints.

We stand by our conjecture that the number of MPLS tunnels deployed over time is related to merger and acquisition activity, and indirectly tracks economic indicators. We believe that Figure 3 makes a compelling case and that this observation could lead to further work on relationships between economics and network configurations. We are in discussions with ISPs to better understand and validate our observations. We emphasize that we merely hypothesize a relationship between MPLS deployments and economic conditions; it is not our intent (as is implied by the word conjecture!) to make a solid claim at this time. Indeed, there may be several reasons behind the observed number of MPLS tunnels over time and it is our hope to better understand these reasons in future work.