MPLS Under the Microscope: Revealing Actual Transit Path Diversity

Yves Vanaubel
Université de Liège
Belgium
yves.vanaubel@ulg.ac.be

Pascal Mérindol
Université de Strasbourg
France
merindol@unistra.fr

Jean-Jacques Pansiot
Université de Strasbourg
France
pansiot@unistra.fr

Benoit Donnet
Université de Liège
Belgium
benoit.donnet@ulg.ac.be

ABSTRACT

Traffic Engineering (TE) is one of the keys for improving packet forwarding in the Internet. It allows IP network operators to finely tune their forwarding paths according to various customer needs. One of the most popular tool available today for optimizing the use of networking resources is MPLS. On the one hand, operators may use MPLS and label distribution mechanisms such as RSVP-TE in conjunction with BGP to define multiple transit paths (for a given edge pair) verifying different constraints on their network. On the other hand, when operators simply enable LDP for distributing MPLS labels in order to improve the scalability of their network, another kind of path diversity may appear thanks to the ECMP feature of IGP routing.

In this paper, using an MPLS labels analysis, we demonstrate that it is possible to better understand the transit path diversity deployed within a given ISP. More specifically, we introduce the Label Pattern Recognition (LPR) algorithm, a method for analyzing `traceroute` data including MPLS information. LPR reveals the actual usage of MPLS according to the inferred label distribution protocol and is able to make the distinction between ECMP and TE multi-path forwarding. Based on an extensive and longitudinal `traceroute` dataset obtained from CAIDA, we apply LPR and find that each ISP behavior is really specific in regard to its MPLS usage. In particular, we are able to observe independently for each ISP the MPLS path diversity and usage, and its evolution over time. Globally speaking, the main outcomes of our study are that (*i*) the usage of MPLS has been increasing over the the last five years with basic encapsulation being predominant, (*ii*) path diversity is mainly provided thanks to ECMP and LDP, and, (*iii*), TE using MPLS is as common as MPLS without path diversity.

IMC'15, October 28–30, 2015, Tokyo, Japan.

DOI: http://dx.doi.org/10.1145/2815675.2815687.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Network topology

General Terms

Measurements

Keywords

network discovery; MPLS; ECMP; multipath; LDP; RSVP-TE; traffic engineering; traceroute

1. INTRODUCTION

One of the cornerstones of the Internet is the way data is forwarded through routing paths. Typically, most of the IP flows are treated the same way whatever their specific Quality of Service (QoS) needs, their destination, or their origin. This absence of privileges and flow distinction is called *best effort routing* or *Internet neutrality*. Tools allowing operators to easily enable path diversity and, so, to perform *Traffic Engineering* (TE) are *Equal Cost MultiPath* (ECMP) load balancers [1] at the IP level and *Multiprotocol Label Switching* (MPLS [2]).

Historically, MPLS has been designed to reduce the time required to make forwarding decisions thanks to the insertion of *labels* before the IP header. Nowadays, it is commonly believed that MPLS is mainly used for providing additional virtual private networks (VPN) services [3] and TE capabilities [4, 5]. Recently, a few studies focused on MPLS, explaining essentially how to reveal its presence and its deployment level [6, 7, 8] or studying its impact on packet forwarding [9]. However, to the best of our knowledge, none of them evaluated how MPLS is actually used in today's Internet.

Two label distribution protocols are used to construct tunnels, according to the intended MPLS usage. On the one hand, the *Label Distribution Protocol* (LDP) [10] on top of IGP enables inter-domain routing stability and extensibility but also preserves ECMP features of the underlying IGP, if any. On the other hand, distributing labels with the *Resource Reservation Protocol TE* (RSVP-TE) [11] allows operators to enable service differentiation (i.e., TE) through the use of multiple *forwarding equivalent classes* (FEC – i.e., a set of packets a given hop forwards to the same next hop,

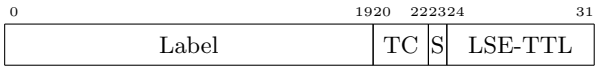


Figure 1: The MPLS label stack entry (LSE) format. An LSE is made of four fields, the *label*, the *traffic class*, the *bottom of the stack*, and the *time-to-live*.

via the same interface with the same treatment). In practice the same ISP may use concurrently both types of label distribution protocols depending on the tunnel usage. Usually, LDP seems to be deployed as a default protocol (to build a full-mesh between edge routers) on MPLS enabled networks. This observation is aligned with our findings presented in this paper as we observe more LDP than RSVP-TE.

Note that VPN based on MPLS may rely either on LDP or RSVP-TE depending on the QoS requirements. Since our study focuses on transit traffic in the public Internet, and since we did not observe many tunnels through VPNs, we will not consider VPNs in the following.

In this paper, in order to differentiate standard IP equal cost paths (that LDP allows) from tunnels built with RSVP for actual TE purposes, we present the *Label Pattern Recognition* (LPR) algorithm. LPR is a passive algorithm in the sense that it does not require any additional probing to standard `traceroute`. It must be applied once the data has been collected, as long as this data contains information related to MPLS tunnels [7, 8]. Briefly, LPR classifies each <Entry point; Exit point> pair of a tunnel into one particular class according to the recognition of the standard behaviors of RSVP-TE versus LDP in terms of label distribution.

We apply LPR on an extensive dataset obtained from CAIDA. Running LPR on five years of data, we find that the usage of MPLS has increased over this period, and that the use of the basic encapsulation method (i.e., LDP for enabling IGP/BGP routing scalability) seems predominant, with or without path diversity. Further, we are able to observe the evolution of each Autonomous System (AS) independently and understand whether it enables path diversity, how, and when it evolves (e.g., from almost no path diversity to a wide deployment of TE). Finally, when TE is deployed, in many cases, the different MPLS paths between endpoints often take the same IP path. We thus observe that TE using MPLS is as common as MPLS without path diversity. This seems to imply that bandwidth is often sufficiently abundant for allowing all tunnels to follow the same route in practice.

The remainder of this paper is organized as follows: Sec. 2 provides the required background for this paper. In particular, it describes MPLS, label distribution, and how MPLS tunnels can be revealed using `traceroute`; Sec. 3 presents the main contribution of this paper with the Label Pattern Recognition (LPR) algorithm; Sec. 4 applies LPR on an extensive and longitudinal CAIDA dataset; Sec. 5 discusses the limits of LPR and of the dataset we used as well as potential future research directions; Sec. 6 positions this paper regarding the state of the art; finally, Sec. 7 concludes this paper by summarizing its main achievements.

2. BACKGROUND

In this section, we provide the required background for the remainder of the paper. In Sec. 2.1, we discuss generalities about MPLS. In Sec. 2.2, we explain the mechanisms for distributing labels in an MPLS network. Finally, Sec. 2.3 explains how MPLS tunnels can be revealed through basic `traceroute` measurements.

2.1 MPLS Overview

The *Multiprotocol Label Switching* (MPLS) [2] was originally designed to speed up the forwarding process. In practice, this was done with one or more 32 bits *label stack entries* (LSE) inserted between the frame header (Data-link layer) and the IP packet (Network layer). A given packet can manage several LSEs at the same time. In this case, the packet is said having a *stack of labels*. Each LSE is made of four fields, as illustrated in Fig. 1: a 20-bit label value used for forwarding the packet to the next router, a 3-bit Traffic Class field for quality of service (QoS), priority, and Explicit Congestion Notification (ECN) [12], a 1-bit bottom of stack flag (when set the current label is the last in the stack [13]), and an 8-bit time-to-live (LSE-TTL) field having the same purpose as the IP-TTL field [14].

MPLS routers, called *Label Switching Routers* (LSRs), exchange labelled packets over *Label Switched Paths* (LSPs). The first MPLS router (*Ingress Label Edge Router*, or Ingress LER, i.e., the tunnel entry point) adds the label stack, while the last MPLS router (*Egress Label Edge Router*, or Egress LER, i.e., the tunnel exit point) removes the label stack. In some cases, for performance reasons, the LSE stack may be removed by the penultimate MPLS router (*penultimate hop popping*, PHP). The Egress LER then performs a classic IP lookup and forwards the traffic, reducing so the load on the Egress LER (specially if the Egress LER is shared among several LSPs). This means that, when using PHP, the tunnel exit is one hop before the Egress LER. Fig. 2 illustrates the main vocabulary associated to MPLS tunnels.

Historically, MPLS has been designed to reduce the time required to make forwarding decisions (an exact match in a LSE Label Information Base – LIB – is faster than a longest prefix match in a trie). Nowadays, MPLS has two main usages: (*i*) a basic encapsulation technique allowing to transparently transmit packets through an MPLS domain using best effort IP routes computed by an IGP, and (*ii*) a traffic engineering (TE) tool allowing to better control routing and resources used by some flows [4, 5]. These two different usages leverage two different signalling protocols to allocate and distribute labels, as we will see in more details in Sec. 2.2.

Moreover, note that there exist other kinds of usage such as MPLS fast reroute [15] to enable faster reaction to network failures with backup LSPs that are computed in a proactive way. Eventually, one really promising MPLS usage, known as *segment routing* [16], will enable in a near future the possibility to easily define routing paths that are orthogonal to those "forced" by the IGP. Indeed, one may be able to recompose IGP paths as a collection of routing segments to deal with numerous interesting use cases such as load balancing, routing policies, or fast-rerouting [17].

2.2 Label Distribution and Usage

In practice, it is worth to notice that, while label distribution protocols are standardized in several RFCs (e.g., [10, 11]), some of their characteristics are specific to router vendors, e.g., the labels range and the default configuration modes. Consequently, some of the following descriptions and statements come from manufacturers' documentation

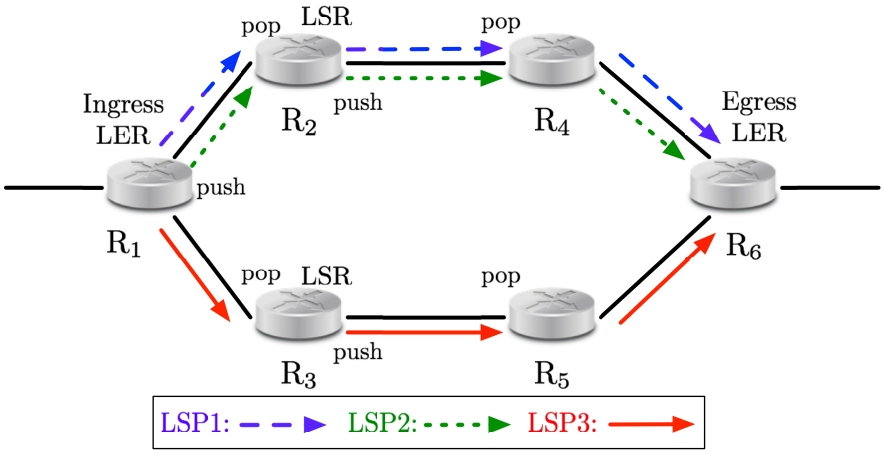


Figure 2: General overview of MPLS (with PHP enabled). Here, there are three different LSPs between the Ingress LER and the Egress LER. LSPs between the Ingress LER and the Egress LER might be physically different (i.e., different IP addresses, as LSP1 and LSP3) or logically different (i.e., same IP addresses but different labels, as LSP1 and LSP2).

(instead of RFCs). Those descriptions have been experimentally verified in our labs, with different configurations and pieces of equipment.

2.2.1 *Basic Encapsulation and LDP*

The basic encapsulation method cited in Sec. 2.1 is mainly used in two common scenarios. First, in the BGP transit scenario, a transit network using BGP as an inter-domain routing protocol and an IGP (e.g., IS-IS or OSPF) as an intra-domain routing protocol may use MPLS tunnels between its border routers to transparently carry packets between them. This way, intermediate routers do not need to know about external destinations[1], only the incoming border routers need to know the outgoing one (by the BGP decision process, it is the BGP next-hop) and the corresponding LSP. Such an MPLS usage may prevent routing loops and other kinds of anomalies [18] and, mostly, enables scalability. Another similar usage is for basic BGP MPLS VPN (Virtual Private Networks [19]). Again LSPs are constructed between the provider equipment (PE) of the VPN and, this way, packets belonging to a VPN may transparently cross the MPLS domain. In both cases, the objective pursued is to separate routing within the domain from routing outside (inter-domain routing or routing in the VPN), not to select routes in the network.

For this basic encapsulation method, labels are allocated through the *Label Distribution Protocol* (LDP) [10]. A router announces to its MPLS neighbors the association between a prefix in its routing table and a label it has chosen. Therefore, labels are allocated from downstream and, for a given prefix, a router advertises the same label to all its neighbors. Depending on the implementation, LDP may advertise a label for all prefixes in its IGP routing table (default case for Cisco routers) or only for loopback addresses (default case for Juniper routers). For transit traffic, LSPs are constructed by LDP towards loopback addresses of the exit border router. The IP route followed by the LSP is the best effort IP route(s) computed by the IGP. If there is no IGP load balancing in the network, there is only one route between the two endpoints (for instance, only LSP1 in Fig. 2 between the Ingress and the Egress LERs). On the other hand, if load balancing is used there may be several routes (usually with equal cost: ECMP Equal Cost Multipath – for instance LSP1 and LSP3 on Fig. 2 between the Ingress and the Egress LERs): the load balancing is generally performed using a hash function on particular IP and transport header fields. Note that LDP builds an LSP-tree towards the destination prefix. Note also that, while the prefix used to build this tree may be very specific (i.e., a single IP loopback address), the *Forwarding Equivalence Class* (FEC – i.e., a set of packets a single router forwards to the same next-hop, via the same interface with the same treatment) may be very large, e.g., all traffic exiting a Tier1 AS from the same border router. On the contrary, using access control list for instance, it is also possible that the FEC is defined at a finer grain (by considering other fields than the IP destination).

2.2.2 *Traffic Engineering and RSVP-TE*

Another quite different usage of MPLS is TE, where the goal is to tune the routes used by flows either to give them requested QoS, or to optimize the network usage. In this case, it is expected that different flows entering the MPLS domain at the same Ingress LER and leaving at the same Egress LER may use different routes (for instance, LSP2 and LSP3 on Fig. 2). Therefore, an IGP adapted for TE

[1]It has been shown [8] that, in practice, there exists a significant difference between the share of routers unable to reply to `ping` if they are involved in an LSP (without being LER). It means that many LSRs do not have a global IP routing plan (no BGP redistribution within the IGP). Instead, IGP routers may then use a default route via a route server for example.

(e.g., OSPF-TE [20] or ISIS-TE [21]) is in charge of computing routes satisfying the TE constraints, while the *Resource Reservation Protocol TE* (RSVP-TE) [11] is the signaling protocol in charge of reserving resources and allocating labels along the route. Note that it is expected that several LSPs can be built between the same pair of LERs. Their label sequences are completely different while their IP path may or may not be distinct (for instance, LSP1 and LSP2 on Fig. 2). One can also define source routed MPLS tunnels to tune its network "manually". In such a case, the label sequences among LSPs are likely to be also specific for each LSP.

Note that the RSVP-TE signalling protocol may be used conjointly with LDP or not. These two protocols are independent even if there is no reason to use only RSVP-TE for specific purposes without using LDP globally within the network.

2.3 Revealing MPLS Tunnels

MPLS routers may send ICMP `time-exceeded` messages when the LSE-TTL expires. In order to debug networks where MPLS is deployed, routers may also implement RFC 4950 [22], an extension to ICMP allowing a router to embed an MPLS LSE in an ICMP `time-exceeded` message. In that case, the router simply quotes the MPLS LSE (or the LSE stack) of the received packet in the ICMP `time-exceeded` message. RFC4950 is particularly useful for operators as it allows them to verify the correctness of their MPLS tunnels and TE policy. This extension mechanism has been implemented by router manufacturers since 1999 [23], and is displayed by modified versions of `traceroute` [24] that report the LSE returned by each hop in addition to RTT values currently displayed.

If the Ingress LER copies the IP-TTL value to the LSE-TTL field rather than setting the LSE-TTL to an arbitrary value such as 255, LSRs along the LSP will reveal themselves via ICMP messages even if they do not implement RFC4950. Operators can configure this action using the `ttl-propagate` option provided by the router manufacturer [14] (while, to the best of our knowledge, the RFC4950 is just a matter of implementation and cannot be deactivated on recent routers supporting it).

Donnet et al. [8] have discussed in detail the impact of those two features (i.e., RFC4950 and `ttl-propagate`) on MPLS tunnel discovery based on `traceroute`. In particular, they show that it is possible to reveal *implicit MPLS tunnels* (i.e., MPLS tunnels with `ttl-propagate` enabled but RFC450 disabled) because core LSRs do not reply to `ping` by themselves but rather forward the reply to their Egress LER (because they do not necessarily benefit from BGP route redistribution).

In this paper we focus on *explicit MPLS tunnels*, i.e., tunnels that can be fully revealed via `traceroute` as they implement both TTL propagation (they are seen in traces) and RFC4950 (they are seen as LSRs providing their LSE). Indeed, mechanisms developed in the following need to interpret the displayed labels to classify MPLS usages, so that we cannot consider implicit MPLS tunnels.

3. LABEL PATTERN RECOGNITION ALGORITHM

In this section, we discuss the *Label Pattern Recognition* (LPR) algorithm, our solution for classifying MPLS tunnels according to their usages and the path diversity they bring. LPR is a passive algorithm in the sense that it does not require any additional probing to standard `traceroute`. It must be applied once the data has been collected. Briefly, our algorithm classifies each <Ingress LER; Egress LER> pair into one of several classes according to the recognition of the standard behaviors of RSVP-TE versus LDP in terms of label distribution. Those behaviors have been experimentally tested and validated in our lab (using both routing emulators and real routers) with different configurations and pieced of equipment.

The whole classification process is illustrated at Fig. 3. Once the `traceroute` data has been collected, we retrieve from the raw dataset MPLS explicit tunnels[2]. Next, the our algorithm can be applied. It works in two fundamental steps. First, it filters the MPLS explicit tunnels in order to remove noise and ensure we only focus on transit tunnels diversity (see Sec. 3.1). Once the data has been sanitized, the classification itself is performed. At the end, each considered *In-Out Transit Pair* (IOTP) is assigned to one of the defined classes. In the remainder of this paper, when we talk about "IOTP", we refer to a <Ingress LER; Egress LER> pair, i.e., a set of explicit MPLS tunnels having the same IP entry and exit points. This means that this IOTP may have several branches, each one corresponding to a particular LSP (as illustrated on Fig. 2, where the IOTP <R_1; R_6> has three branches).

3.1 Filtering

The first step consists in the filtering and sanitizing of the set of LSPs and/or IOTPs. This step is done through four different filters sequentially applied.

First, based on our observations, using inter-domain MPLS tunnels for transit traffic is negligible[3]. We therefore chose to not consider them in our study. As a consequence, IP addresses involved in a given LSP must belong to the same AS, otherwise it is rejected. This first filtering step is done by the `IntraAS` filter (see Fig. 3).

The objective of the `TargetAS` filter is to ensure that the `traceroute` destination is in a different AS than the tunnel itself. Indeed, imagine a situation in which the tunnel and the `traceroute` destination belong to the same domain. In that case, we are not in a scenario in which the tunnel is used for carrying transit traffic and, so, unlikely to be used for TE purpose.

[2]Any `traceroute` dataset can serve as input to LPR. The only condition is to be able to retrieve MPLS explicit tunnels from the `traceroute`, as explained in Sec. 2.3.

[3]Note that the observed rise in remote peering [25] does not contradict this measurement statement for several reasons. First, remote peering are made of invisible MPLS tunnels (i.e., no RFC4950 and no `ttl-propagate`– see Sec. 2.3) such that our measurement methodology is unable to retrieve them. Second, the purpose of remote peering is to avoid transit through another network while the scope of our study is the transit traffic. Eventually, remote peering consists in MPLS tunnels that belong to the remote peering provider, not tunnels going through several subsequent ASes

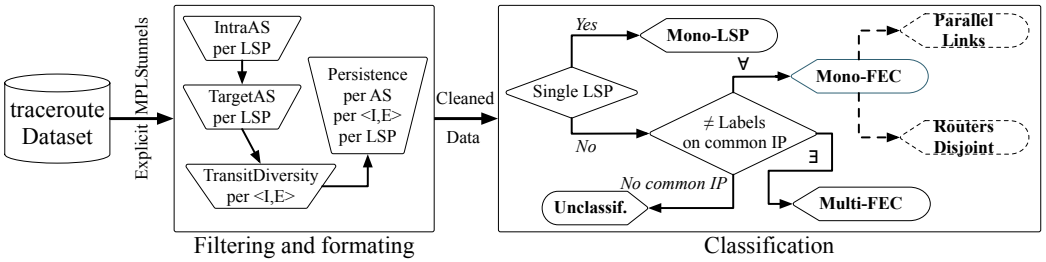


Figure 3: Overview of the LPR algorithm. LPR is divided in two important steps: (*i*) data filtering and (*ii*) MPLS tunnels classification.

To have a better view of the routing diversity, we next want to keep only IOTPs that are used to reach at least two destinations belonging to different ASes (`TransitDiversity` filter). The idea here is to capture multiple FECs (*multi-FEC* – remind that a FEC refers to a set of packets a single router forwards to the same next hop, via the same interface with the same treatment) scenarios based on IP destination prefixes that, by definition of IP routing, represent the more classical practice of TE. It is worth to notice that, even with that filter, we may underestimate the transit tunnel diversity (and, so, TE usage).

Finally, we verify the persistence in time of the LSPs to remove noise due to routing changes (`Persistence` filter). We keep LSPs encountered in measurement cycle X only if they are also seen in measurement cycle $X+1$, $X+2$, ..., or $X+j$ (the impact of the number of additional cycles j is evaluated in Sec. 4.2). In this case, the j cycles are consecutive and taken in the same month as cycle X. Note that we keep track of dynamics as it can represent in itself a TE usage rather than routing changes. In practice, if the vast majority of LSPs disappear for a given AS, we reinject the whole set of its LSPs to perform a standard classification on a given snapshot[4]. That is, we do not remove such an AS and continue to process it as the others but adding a dynamic tag to it.

The resulting set of IOTPs can then be classified with LPR whose pseudo-code is given in Algorithm 1. Those subsets of LSPs are robust, i.e., they are persistent in time, possibly diverse in targets, and focus on transit traffic through a single ISP. In the following, for each LSP falling within the same AS and the same couple of border edge routers <Ingress LER, Egress LER>, we compare their content both in terms of IP addresses and labels in order to determine the actual usage of the tunnel.

Algorithm 1 Classification step of the LPR algorithm.

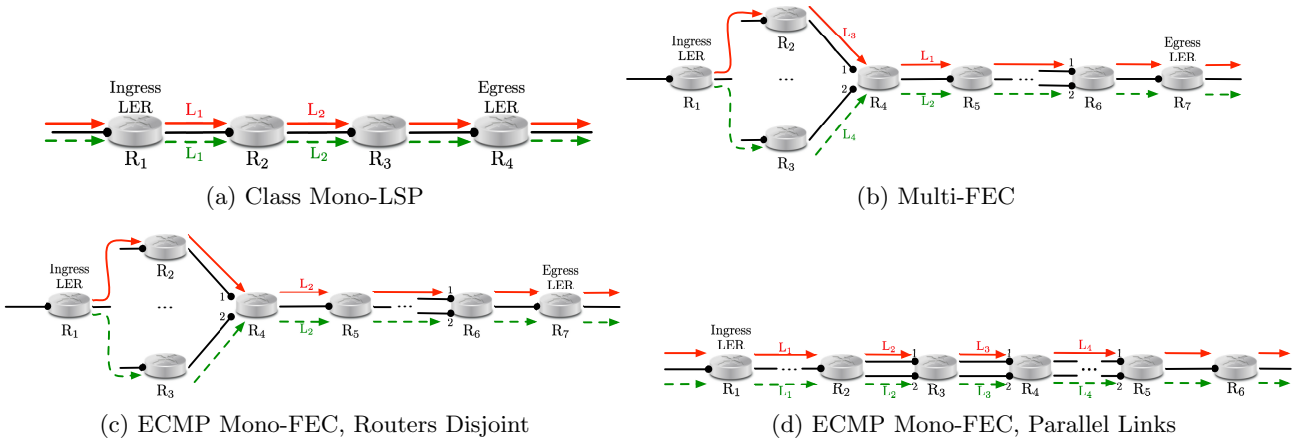
```
Require: 𝕋, the set of IOTPs after filtering
 1: function LPR_CLASSIFICATION(𝕋)
 2:     /* creating resulting classes */
 3:     MonoLSP ← ∅
 4:     MultiFEC ← ∅
 5:     MonoFEC ← ∅
 6:     Unclassified ← ∅
 7:     for IOTP_i in 𝕋 do
 8:         /* obtaining all LSPs of a given tunnel */
 9:         LSPs ← IOTP_i.getLSPs()
10:         if |LSPs| == 1 then
11:             /* class 1 (Mono-LSP) */
12:             MonoLSP ← MonoLSP ⋃ IOTP_i
13:             continue
14:         /* checking for common IP addresses */
15:         CIP_i ← IOTP_i.getCommonIP()
16:         if |CommonIP| == 0 then
17:             /* class 4 (Unclassified) */
18:             Unclassified ← Unclassified ⋃ IOTP_i
19:             continue
20:         for IP_j in CIP_i do
21:             if |IP_j.getLabels()| > 1 then
22:                 /* class 2 (Multi-FEC) */
23:                 MultiFEC ← MultiFEC ⋃ IOTP_i
24:                 IOTP_i_ismFEC = TRUE
25:                 continue
26:         if ¬IOTP_i_ismFEC then
27:             /* class 3 (ECMP) */
28:             MonoFEC ← MonoFEC ⋃ IOTP_i
29:     /* returning all tunnel classes */
30:     return MonoLSP, MultiFEC, MonoFEC, Unclassified
```

3.2 Classification

The first class, illustrated in Fig. 4(a), is called *Mono-LSP*. That is, for a given IOTP, there exists only a single LSP (i.e., same IP addresses and same labels) for different destination ASes. This means we do not observe transit tunnel diversity, the same LSP being always used whatever the destination. As a consequence, for this tunnel, we are not able to reveal ECMP load balancing (by definition of the class) or the deployment of several FECs used to reach different ASes with different routing constraints (although we consider at least two destination ASes as stated in the filtering subsection). The condition to reveal such a class is described at line 10 of Algorithm 1.

The second class, illustrated in Fig. 4(b), is called *Multi-FEC*. That is, for a given IOTP, there exists at least one convergence point, i.e., a *common IP* interface belonging to an LSR where at least two LSPs converge. This convergence occurs on a given IP address of the LSR but we observe that the different LSPs use different MPLS labels at that convergence point. For instance, on Fig. 4(b), the first LSP (plain line) considers labels (L_3, L_1), while the second (dashed line) considers labels (L_4, L_2). Labels L_1 and L_2 being used on

[4]In this paper, the reinjection is realized only if the whole set of LSPs is deleted by the filter.



(a) Class Mono-LSP

(b) Multi-FEC

(c) ECMP Mono-FEC, Routers Disjoint

(d) ECMP Mono-FEC, Parallel Links

Figure 4: Typical MPLS Label Based Patterns. Note that big dots refer to a router IP interface.

the same IP address, thus on the same router, this case suggests the use of multiple FECs for that tunnel. On the contrary to the use of standard LDP where, by default, labels have a router scope (i.e., each LSR proposes the same label for a given destination to all its upstream routers[5]), distinct labels proposed by the same LSR for a given Egress LER, indicate distinct FECs. This use of multi-FEC suggests TE practice for that particular tunnel. In order to provide an upper bound of TE usage, we classify an IOTP as multi-FEC as soon as distinct labels appear on a given common IP address as described at lines 20–25 of Algorithm 1.

Note that the concept of common IP address is fundamental for the classification as we cannot conclude anything if there does not exist any of them for a given IOTP. Fortunately, most LSPs of a given IOTP converge at some point[6], such that we can distinguish the label distribution protocol in use. In practice, we introduce the notion of common IP address sets computed independently for each IOTP (line 15 of Algorithm 1). It simply consists of all IP addresses belonging to LSRs that are traversed by at least two distinct LSPs of a given IOTP. The two LSPs may differ at a given hop either in terms of IP addresses (this hop is then not included in the common IP set by definition) or in terms of labels (at any hop, including a common IP one).

The third class, shown in Fig. 4(c) and Fig. 4(d), refers to IOTPs that perform load balancing between their LSPs. More specifically, this class implies that, for all LSPs of a given IOTP traversing a common LSR, labels are identical (on the contrary to the Multi-FEC class where there exists at least one difference). Therefore, this class refers to the use of a single FEC between the Ingress and the Egress LERs. It typically corresponds to the use of MPLS load balancing on the top of IP thanks to ECMP. Note that an IOTP falls in this class if and only if all its common IP addresses verify this behavior as stated at lines 26–28 of Algorithm 1. This class, that we call *ECMP Mono-FEC* (or *Mono-FEC* on Fig. 3 for readability reasons because it consists in Multi-LSP Mono-FEC), may be divided into two subclasses. First, if labels are the same all along the LSPs but IP addresses differ, we believe IP addresses at a given hop are aliases. Indeed, recalling that the scope of LDP labels is local to the LSR, it is unlikely that two distinct LSRs will propose the same label. We thus conclude, in this case, that the IOTP only uses parallel links to perform ECMP load balancing (Fig. 4(d)). On the contrary, if LSPs differ both in labels and IP addresses on, at least, a given hop, the IOTP seems to perform ECMP load balancing through disjoint routers (Fig. 4(c)). This distinction is of the highest interest: we do not require the use of an active resolution probing to show that a large portion of ECMP load balanced paths relies actually only on parallel links.

Finally, when PHP is used, the common IP set for a given IOTP may be empty if the LSPs converge only at the Egress LER. In this situation, we arbitrarily tag this IOTP as *Unclassified*, as indicated at lines 16–18 of Algorithm 1. Sec. 5 proposes an alternative method to avoid this limitation, but in practice, it rarely occurs as demonstrated in Sec. 4.

4. EVALUATION

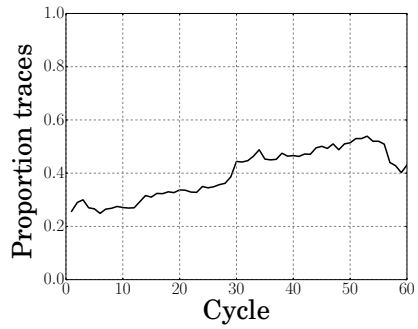
In this section, we use our LPR algorithm in order to evaluate and understand the usage of MPLS. We first describe the considered dataset (Sec. 4.1). Next, we evaluate the impact of the filtering steps on the dataset and the classification (Sec. 4.2). We also focus on several IOTPs properties in Sec. 4.3. Then we focus on a subset of five large ASes (mainly Tier-1) whose MPLS deployment is characteristic and discuss how it is used and deployed over the five years of data (Sec. 4.4). Finally, we discuss some properties of MPLS label dynamics on a specific AS (Sec. 4.5).

4.1 Dataset

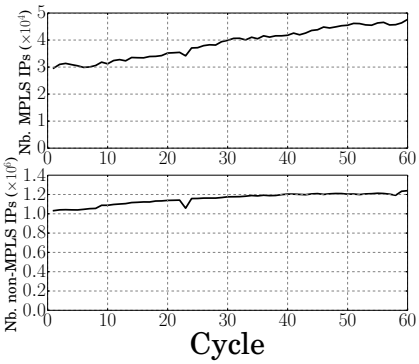
In order to evaluate how MPLS is used and how its usage has evolved over time, we apply LPR on an extensive dataset obtained from the CAIDA Archipelago infrastructure [26], a distributed infrastructure in which teams of monitors perform Paris traceroute [27] to all routed /24 prefixes in a short amount of time. Currently, the Archipelago infrastructure is made of more than 100 monitors scattered around the world.

[5]In practice, such destinations are generally loopback addresses of BGP edges routers. It is used here for scalability reasons and refers to the use of the BGP next-hop feature: it does not distinguish FECs for a given edge router.

[6]It is not necessarily at the Egress LER since the use of PHP does not exhibit labels at this last hop in practice.



(a) Proportion of traceroutes traversing at least one MPLS explicit tunnel



(b) MPLS usage in IP addresses

Figure 5: Global deployment of MPLS in our dataset.

For our needs, we download data collected between January 2010 and December 2014. For each month in that period, we consider the first run of each team and call such a data a *cycle*. We therefore work on 60 cycles. For each cycle, we perform IP2AS mapping using Routeviews data [28] collected the same day as the considered cycle. Next, for each cycle, we extract all explicit tunnels, i.e., MPLS tunnels that exhibit the `ttl-propagate` option and ICMP extension (see Sec. 2.3).

Fig. 5(a) shows, for each cycle, the proportion of `traceroute` that traverse at least one explicit tunnel (before any kind of filtering performed by our cleaning stage). During the five years of data, we observe a significant increase in the deployment of MPLS. Around the 30^{th} cycle, we see an increase of roughly 10% in the traces showing at least one MPLS tunnel. Similarly, at the end, we observe a decrease. This phenomenon is due to the rise and fall of MPLS usage inside AS3356 (see Sec. 4.4, and Fig. 16 in particular, for details about AS3356 – a Tier-1 whose name is Level3). Fig. 5(b) shows the raw number of unique IP addresses used in MPLS (upper graph) and those not used in MPLS (lower graph) for each cycle. If we see a slight increase in the number of IP addresses observed in the data collected (21% over the five years), we observe a much larger increase in the number of IP addresses used for MPLS (60%). Further, two drops are observed in Fig. 5(b): at cycle 23 and 58. Those drops are caused by measurements issues in the Archipelago infrastructure.

Filter	Average
Incomplete LSPs	0.853 ±0.01
`IntraAS`	0.844 ±0.01
`TargetAS`	0.717 ±0.009
`TransitDiversity`	0.644 ±0.009
`Persistence`	0.534 ±0.007

Table 1: Cumulative average (and confidence interval), over the 60 cycles, of the proportion of tunnels remaining after applying each filter. On average, a cycle contains 14×10^6 LSPs before filtering.

Globally speaking, and as already stated previously [7, 8], this evolution shows operators seem to deploy more and more MPLS tunnels during the last decade. This increase confirms the interest of many operators for MPLS and opens the question of its actual usage.

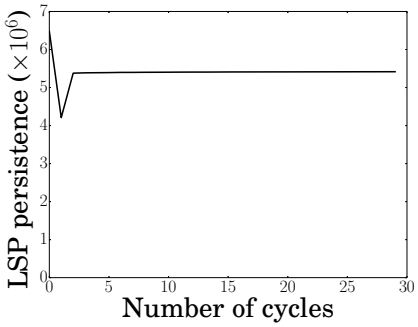
4.2 Filtering Impact

In this section, we evaluate the impact of the first step of LPR, i.e., the filtering process (see Sec. 3.1). Table 1 gives a first overview of filtering effects on LSPs. It provides the cumulative average (with confidence intervals) over the 60 cycles of the LSPs proportion remaining after applying each filter. The "Incomplete LSPs" line refers to LSPs that are incomplete in the `traceroute` sense, i.e., at least one of the LSRs composing the LSP did not reply to probes (i.e., anonymous router). In that case, the whole LSP is removed from the dataset. We see that the strongest filter (i.e., the one that removes most of tunnels) is the Incomplete LSPs as it removes, on average, 14.7% of tunnels. There is nothing surprising here as it is very common to have incomplete `traceroute` due to anonymous routers [29]. This filter impact is also considerable relatively to others because this is the first one being applied.

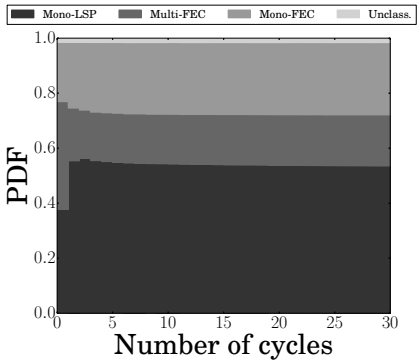
In Sec. 3.1, we claimed that using inter-domain MPLS tunnels is negligible. This is confirmed by statistics associated to the `IntraAS` filter as only a low 0.9% of LSPs concerns inter-domain tunnels. Further, on average, 12.7% of the tunnels concern a destination located in the same AS as the tunnel itself (`TargetAS`) while 7.3% were used for reaching a single destination (`TransitDiversity`). Finally, 10% of remaining tunnels are removed due to routing noise (`Persistence`). This filter keeps LSPs encountered in measurement cycle X only if they are also seen at least once in measurement cycle $X+1$, $X+2$, ..., or $X+j$. Here, the j cycles are taken consecutively in the same month as cycle X. For Table 1, we fixed $j = 2$.

To evaluate the impact of the `Persistence` filter and, in particular, the impact of the number of subsequent snapshots considered for applying the filter, we focus on the 29 snapshots composing the whole December 2014 Archipelago dataset. We vary the parameter j between 0 (no `Persistence` filter) and 29 (the whole month is used for the persistence analysis but a few measurement campaigns last a bit more than one day). Results are shown in Fig. 6.

Fig. 6(a) shows the amount of tunnels kept after `Persistence` filtering. We observed that increasing the parameter j does not affect that much the number of tunnels. When $j \geq 2$, the amount of tunnels remains mostly stable. A drop is observed compared to when $j = 0$. This is expected as, by definition, when $j = 0$, no `Persistence` filter is applied.



(a) Number of tunnel kept



(b) Impact on classification

Figure 6: Impact of `Persistence` filter on the December 2014 dataset.

In the same fashion, the large drop at $j = 1$ is due to the necessity to retrieve the LSP in two subsequent snapshots while the condition is, by definition, more relaxed for $j \geq 2$. Fig. 6(b) shows the impact of the `Persistence` filter on the classification at a global scale. It would be a matter of concern if the persistence had a strong impact on the classification. We observe that when $j \geq 2$, the classification remains stable. The impact, for $j \leq 1$ is to trade Mono-LSP tunnels with Multi-FEC tunnels and is partially explained by the use of dynamic Multi-FEC LSPs (see Sec. 4.5).

In the remainder of this paper, we consider the value $j = 2$ for the `Persistence` filtering, meaning that an LSP observed in cycle X will be considered for classification if it is also encountered in measurement snapshot $X + 1$ or $X + 2$ of the same month.

4.3 Tunnel Length, Width, and Symmetry

In order to describe observed IOTPs at a high level, we adapt metrics proposed by Augustin et al. [1] for load balanced paths. Recall that an IOTP is defined as an <Ingress LER; Egress LER> pair corresponding to a set of explicit MPLS tunnels having the same entry and exit points.

We first define an IOTP *length* as the number of LSRs in the longest LSP of that IOTP without counting the Ingress and Egress LERs, i.e. the number of intermediate LSRs in the longest LSP. Fig. 7 provides the length distribution of IOTPs for the last cycle of our data set (i.e., December 2014). We observe that, in most cases (i.e., $> 65\%$), tunnels

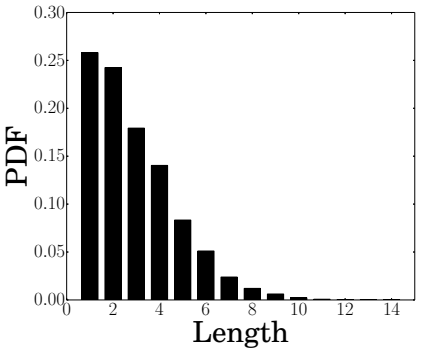
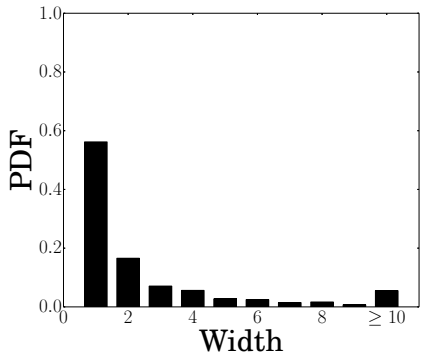
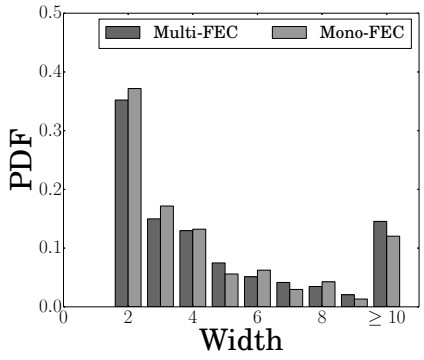


Figure 7: IOTP length distribution (Cycle 60).



(a) All classes



(b) Mono-FEC and Multi-FEC

Figure 8: IOTP width distribution (Cycle 60).

are rather short, i.e., ≤ 3 LSRs[7], although a very low proportion of them can be quite long. This property is related to the short diameter of many ASes [30].

Fig. 8(a) provides IOTPs' *width* distribution, i.e., the number of "branches" between the Ingress LER and the Egress LER. Branches may differ physically (i.e., IP addresses) or logically (i.e., MPLS labels). It is worth to notice that, by definition, only tunnels falling in the Mono-LSP class will have a width of 1. Although a very low proportion of the IOTPs are very large (up to 236 branches in our dataset),

[7]To which one has to add the LERs to obtain the complete LSP.

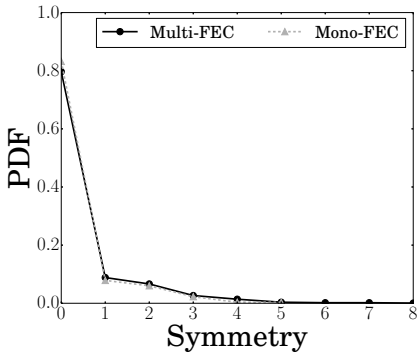


Figure 9: IOTP symmetry distribution (Cycle 60) for Multi-FEC and Mono-FEC classes.

most of them (56%) are narrow, i.e., a width of 1. This reveals that, in our dataset, the observed IOTPs are mainly Mono-LSP. It is worth to notice that this result is consistent with Fig. 6(b) in which the majority of IOTPs are indeed Mono-LSP. Additionally, Fig. 6(b) also reveals how tunnels are used in general. The usage of RSVP-TE (and thus TE) is limited (roughly 20% of the IOTPs), leading thus to the conclusions that tunnels are essentially built based on LDP.

Fig. 8(b) shows the width distribution for each class separately, except the Mono-LSP one[8]. The key point here is that Mono-FEC and Multi-FEC tunnels have nearly the same distribution. Although the tail of the distribution is slightly dominated by Multi-FEC, this result is quite surprising as it tends to indicate that TE-based LSPs do not use much more path diversity than basic ECMP (with a different way to perform load balancing though). Since such a property is biased by the measure itself (it is a lower bound as it does not provide a complete exploration, see Sec. 5), the following property may be useful to better understand structural differences between this two kinds of path diversity.

Finally, an IOTP *symmetry* is defined as the difference between that IOTP length and the number of LSRs in the smallest LSP of that IOTP. If the symmetry equals 0, we say that the IOTP is *balanced* (or *symmetrical*). Otherwise, it is said to be *unbalanced* (or *asymmetrical*) and the value obtained gives an idea of the imbalance. Obviously, by definition, an IOTP that would fall in the Mono-LSP class is balanced. Therefore, the distribution given in Fig. 9 does not show the Mono-LSP case[9]. The first observation is that balanced paths represent 80% of the distribution (for both classes). This property is derived from two facts: ECMP forwarding paths are often similar in terms of hop count (80% of Mono-FEC IOTPs are balanced) and many multi-FEC LSPs actually go through the same path, they only differ in terms of labels – note that it is the main cause of the 80% share of the balanced Multi-FEC IOTPs. The second one is, again, that there are no significant differences between the two classes. This similarity is really surprising as one may speculate that TE paths may completely

[8]Unclassified tunnels (not shown on Fig. 8(b) as they are marginal – see Fig. 6(b)) are mainly narrow.

[9]Unclassified IOTPs are marginal and, thus, not shown on Fig. 9.

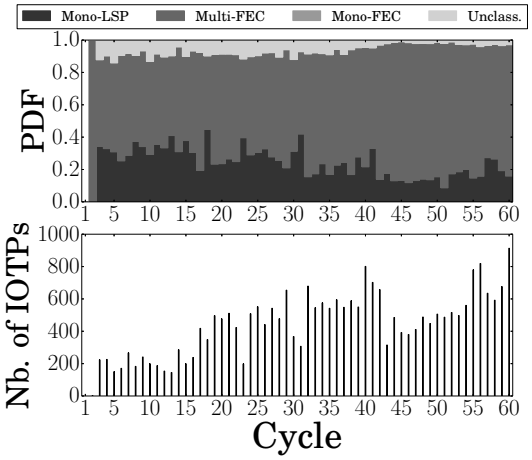


Figure 10: Tunnels classification for AS1273 (Vodafone). We observe mainly a Multi-FEC usage of MPLS.

differ among themselves according to each FEC constraint. In practice, constraints seem to be satisfied by a unique IP path in most cases, meaning that reserving bandwidth in an over-provisioned network is equivalent to using resource pooling mechanisms such as ECMP.

4.4 IOTPs Classification

Fig. 10, 11, 12, 14 and 15 present the IOTPs classification for several ASes. We selected this set of five ASes as they are representative of typical MPLS behaviors across the whole dataset. It is worth to notice that they all are Tier-1 ASes except Vodafone (AS2173), a Transit network. Each of those figures is made of two parts:

- The lower part gives, for each cycle (X-Axis), the number of IOTPs that were considered for classification.
- The upper part provides, for each cycle (X-Axis), the proportion of tunnels in each of the four classes.

Empty zones in both parts of the graph refer to cycles where no MPLS tunnel was encountered. Finally, note that those figures must be read conjointly with Table 2 that provides the name and statistics about all analyzed network infrastructures (i.e., minimum/maximum/average number of IP addresses, tagged as MPLS or not, observed over a year).

Fig. 10 gives the classification for AS1273 (Vodafone, large Transit ISP). AS1273 is interesting in several points of view. First, as we notice in Table 2 and in Fig. 10 (below part), the usage of MPLS (for transit traffic) within this AS has increased over time. Second, AS1273 seems to deploy MPLS mostly for TE reasons (the Multi-FEC class usage also grows with time to the detriment of Mono-LSP usage). Moreover Mono-FEC (ECMP) is almost invisible.

Fig. 11 presents tunnels classification for AS7018 (AT&T, Tier-1 ISP). The usage of MPLS relatively decreases over time while the Multi-FEC class is more and more used in place of Mono-FEC tunnels. There is a drop in the number of IOTPs around cycle 22, that seems to correspond to a transition in MPLS usage.

Fig. 12 presents tunnels classification for AS6453 (Tata Communications, Tier-1 ISP). This AS has almost no Multi-FEC and a strong (although declining) usage of Mono-FEC

		2010			2011			2012			2013			2014		
		min	max	avg	min	max	avg	min	max	avg	min	max	avg	min	max	avg
AS1273 (Vodafone)	non MPLS	796	1097	887	828	914	871	901	993	950	918	1,014	971	928	1,088	990
	MPLS	0	160	115	108	192	147	149	230	199	134	234	178	147	222	171
AS7018 (AT&T)	non MPLS	36,614	37,656	36,984	34,537	39,988	37,874	40,022	48,839	45,764	48,946	57,995	50.597	50,518	57,403	52,489
	MPLS	379	588	493	555	775	663	608	701	655	554	648	605	486	554	508
AS6453 (Tata)	non MPLS	2,099	2,269	2,181	1,896	2,209	2,043	1,883	2,118	2,005	2,017	2,165	2,084	2,054	2,162	2,099
	MPLS	435	513	463	326	454	380	291	334	308	260	336	298	268	314	293
AS2914 (NTT)	non MPLS	2,760	3,061	2,884	3,138	3,368	3,243	3,293	3,429	3,349	3,349	3,641	3,490	3,652	4,116	3,885
	MPLS	191	247	216	226	270	248	243	293	262	283	314	300	308	328	316
AS3356 (Level3)	non MPLS	9,121	9,417	9,253	9,245	9,775	9,503	9,464	10,033	9,700	9,434	9,997	9,708	9,342	11,108	10,162
	MPLS	0	14	1	0	0	0	0	646	369	570	726	676	3	776	518

Table 2: Statistics about IP addresses for some ASes of interest (after filtering).

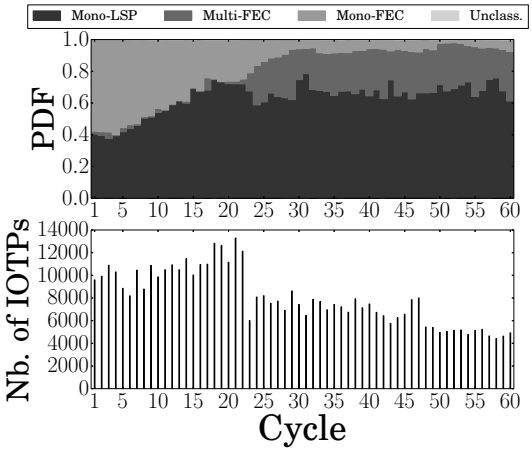


Figure 11: Tunnels classification for AS7018 (AT&T).

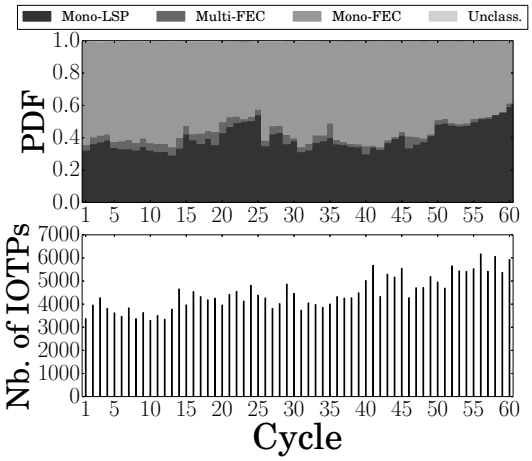


Figure 12: Tunnels classification for AS6453 (Tata Communications). We observe mainly a Mono-FEC usage of MPLS.

that exhibit topology logical properties enabling a large use of ECMP.

Fig. 13 deepens the Mono-FEC class for AS6453 (Tata Communications). Indeed, this class can be split into "Routers Disjoint" (i.e., LSPs in a given IOTP differ both in labels and

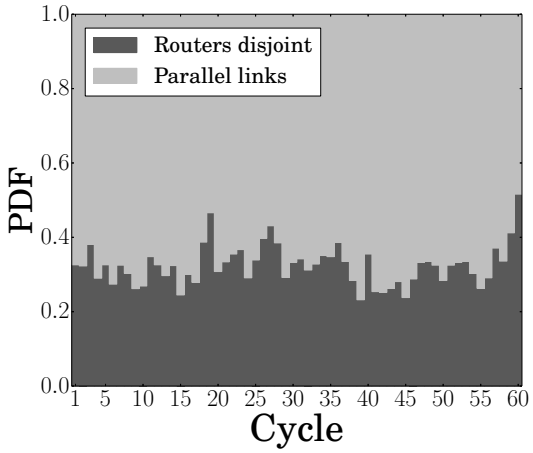


Figure 13: Split between "Routers Disjoint" and "Parallel Links" for Mono-FEC class of AS6453 (Tata Communications).

IP addresses on at least a given hop) and "Parallel Links" (i.e., LSPs in a given IOTP in which labels are the same all along the LSPs while the corresponding IP addresses are different). Over time, AS6453 has deployed Mono-FEC tunnels mostly based on parallel links (between 60 and 70% of tunnels belong to the Parallel Links subclass).

Fig. 14 presents the tunnel classification for AS2914 (NTT, Tier-1 ISP. This AS shows an increase in MPLS usage. The number of IOTPs observed during the period of interest has been multiplied by three, which is consistent with the fact that the number of IP addresses used in MPLS has also increased (see Table 2), while the usage of MPLS has remained mostly stable over time: Mono-LSP. We, however, see that, with time, the usage of Mono-LSP slightly and relatively decreases in favor of Mono-FEC.

Fig. 15 presents tunnels classification for AS3356 (Level3, Tier-1 ISP), which has a quite curious shape: MPLS appears during the 29th cycle, observes a period of stability, and, finally, presents a sharp decrease starting at cycle 55.

We investigated the data before cycle 29, in order to determine whether the rise of MPLS during that cycle matches with new hardware (or IP addresses) deployment that is dedicated to MPLS or a dramatic change in routing. Looking at the Archipelago files, we find that, in an infrastructural point of view, nothing has changed between Cycle 28 and Cycle 29, i.e., we observe mostly the same set of IP addresses. This

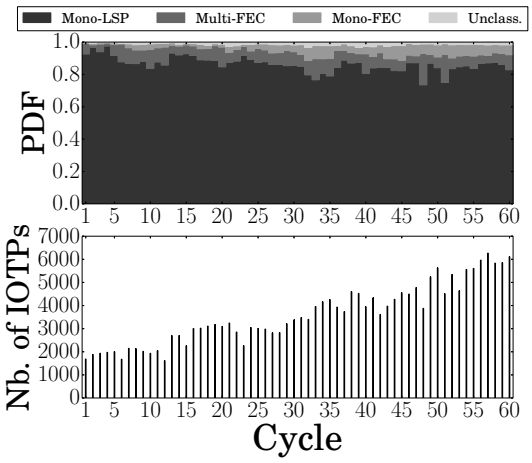


Figure 14: Tunnels classification for AS2914 (NTT). We observe mainly a Mono-LSP usage of MPLS.

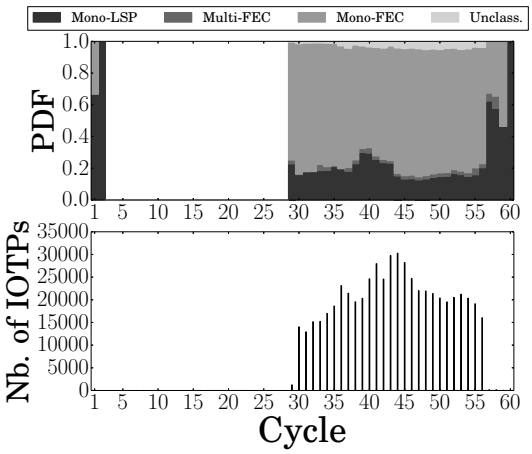


Figure 15: Tunnels classification for AS3356 (Level3). We mainly observe a Mono-FEC usage of MPLS.

means that only the usage of the existing infrastructure has been modified between cycle 28 and cycle 29.

To investigate further, Fig. 16 presents the state of MPLS deployment the month prior to the 29$^{\text{th}}$ cycle (April 2012). To do so, we downloaded all daily Archipelago data dumps for April 2012, without respecting the probing cycle (as described in Sec. 4.1). This means that the number of considered Archipelago vantage points differs from one day to another. Fig. 16 provides two pieces of information: (*i*) the number of encountered IOTPs each day before and after the filtering steps (upper part of Fig. 16) and (*ii*) the number of LSPs identified each day, again before and after the filtering steps (lower part of Fig. 16). Peaks and drops observed in the number of IOTPs identified after April 25$^{\text{th}}$ are due to the variations in the number of considered Archipelago vantage points.

We observe that the deployment of MPLS has started around April 15$^{\text{th}}$, and it took a half month to fully deploy MPLS in AS3356. It exhibits an incremental MPLS

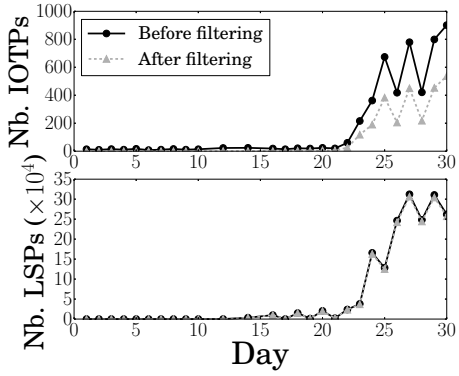


Figure 16: The rise of MPLS deployment in AS3356 (Level3), focus on April 2012.

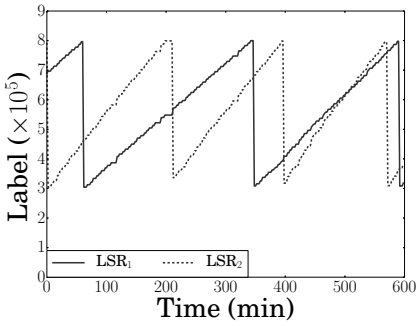


Figure 17: Label range evolution in case of Multi-FEC tunnels, belonging to AS1273 (Vodafone), as seen from a vantage point located in Strasbourg.

deployment rather than an abrupt transition. Further, the number of LSPs does not differ before and after filtering (the `Persistence` filter is not used here), while it does for IOTPs. It is due to the fact that most LSPs are "shared" by several IOTPs, i.e., only one of the two LSP extremities (the LER) differ among a subset of them in many cases (see Sec. 5 for discussions about LPR limitations and alias resolution in particular).

Generally speaking, we observe in this section that using LDP is the rule (with – Mono-FEC – or without ECMP – Mono-LSP) while RSVP-TE seems to be more marginal both in terms of deployment and, more surprisingly, in terms of the structural path diversity it enables compared to ECMP.

4.5 Multi-FEC and Label Dynamics

The `Persistence` filter (see Sec. 4.2) was introduced to remove side effects of routing dynamics. For example, we could infer two LSPs in a given IOTP (hence a Multi-LSPs IOTP in theory) while these two LSPs were not simultaneously used in practice. It may be due to a routing change during the measurement. However we found that in some (rare) cases this filter could hide an interesting phenomenon, namely LSPs that change their labels very frequently. This type of LSPs is filtered out by the `Persistence` filter so that we study those ASes in a different way (in practice, they are

easily retrievable since all or almost all LSPs disappear with the filter)[10].

To investigate this a little bit further, we selected a small number of ASes exhibiting this behavior (this was the case for AS1273 we discussed in Sec. 4.4), and launched an additional `traceroute` campaign from a unique vantage point located in Strasbourg, the purpose being to monitor the label changes frequency. So for each destination, the route was traced at a very high sampling rate, i.e., a `traceroute` was sent to each destination every two minutes. The results for a single LSP made of two LSRs is given in Fig. 17. The Y axis represents the labels (labels range from 300,000 to 800,000) and each line represents the label evolution. When a label reaches its maximum, it starts again from the minimum).

Fig. 17 shows that the LSP is reconstructed at a high frequency and almost periodically. Our interpretation is that this LSP is built by RSVP-TE and that the Ingress LER is configured to "re-optimize" frequently the LSP (although in practice it follows the same IP route). Moreover, we point out three interesting findings. First, this systematic temporal-based behavior seems to be mainly related to Juniper hardware [31]. Second, we also observe on Fig. 17 that label changes seem to be also performed on a factual basis (on each curve, some step durations differ at some points). Eventually, the shape of the curve resulting from LSR_2 evolves more quickly than the one of LSR_1: it suggests that LSR_2 is more solicited than LSR_1 in terms of number of LSPs going through it.

5. DISCUSSION

In this section, we discuss possible extensions of our works. While the core mechanism of LPR, i.e., the inference technique that allows for distinguishing among LDP and RSVP-TE, has been experimentally tested and validated, our work can be improved in two directions to extend its scope and to increase its accuracy. We first discuss potential measurement improvements (the active campaign) and then consider possible LPR extensions (the passive analysis).

The first hypothesis we need to verify and explore is about the way MPLS TE is deployed and used by today's operators. In particular, we considered that the traffic differentiation is performed according to destination prefixes (as in standard IP forwarding), i.e., distinct FECs are set up on this basis. Some operators may consider other ways for distinguishing types of services: the source IP prefix, the incoming AS, or even other fields from both the IP and the transport headers for setting FECs at the flow grain. Instead of simply optimizing their own resources according to the entry point in their networks, some operators may enable multipath forwarding for more specific objectives (using complex access control lists – ACLs – rather than standard commands). In this paper, as a first global study on the MPLS usage, we mainly rely on an extensive and longitudinal existing dataset that provides historical traces. The main purpose of this dataset being to provide topology discovery data for helping researchers to understand IP network properties, it does not bring enough information for such a specific study. However, based on a dedicated and finely calibrated probing campaign, we should better understand whether such practices are common or not.

Second, as an ongoing work for providing ground truth results, we currently check if our distinction between IP and MPLS multipath routing is valid. One way to do that would be to launch an extensive Paris traceroute [27] campaign to understand if the LSPs we tag as Mono-FEC ECMP (and so using LDP) are actually also visible with such a tool. Beside, we also plan to check whether Multi-FEC LSPs are, indeed, not visible through Paris traceroute. If those two properties are correct, we argue that it would be a ground proof of our label-based analysis done through LPR. We believe that such a validation campaign will also provide us the ability to state if IP-only (i.e., not using TCP or UDP ports) load balancers actually exist for standard traffic. Indeed, the study of Augustin et al. [1] mentions this possibility while we suspect it is likely to be rather due to multi-LSP deployed on a destination basis.

Third, in this paper, we decided not to rely on any alias resolution mechanisms [32] to avoid well known biases that they may induce. However, it can be interesting to define an IOTP at the router level rather than at the IP level. Such a consideration should provide more refined data for distinguishing our MPLS classes. In particular, it will reduce the number of IOTPs and so provide more consistent results that may be closer to the actual MPLS usage.

On the LPR algorithm itself, we may modify two main aspects. As a future work, we envision to extend our analysis in order to take into account *LSP-trees* (i.e., the capability of forwarding packets belonging to the same FEC but issued from several Ingress LER, and so arriving from distinct IP addresses – but with the same LSE label considering LDP – using the same outgoing label [2]). It may allow for a better understanding of the underlying use of LDP and so improve the applicability of our classification. Indeed, using such an extension, more LSPs will be classified with LPR because they will be indexed only through the Egress LER. Note that, in practice, we will have to consider DAGs rather than trees due to the use of ECMP. This study is in the same vein as the one that envisions to use preliminary alias resolution techniques as it can provide more consistent data.

Finally, if PHP is used (i.e., the LSE is removed by the penultimate LSR – see Sec. 2.1), the Egress LER generally does not exhibit labels. In such a situation, LSPs within a given tunnel may never reach a common IP address providing labels[11]. To avoid such a limitation and so achieve a complete classification, we can rely on a simple heuristic for alias resolution mechanism. In a usual situation with a series of point-to-point links inferred thanks to a `traceroute` path, all previous IP level interfaces of a given common IP address should belong to the same router. Indeed, if we assume that a router answers to `traceroute` probes using the incoming interface of the probe and there is no layer-two device connected to this interface, then such IP addresses are aliases of the same router (since to enter a router through the same IP interface, it is necessary to use the same point-to-point link on the same upstream router). In the case of Mono-FEC, we should thus observe the same label on previous IP addresses while we should observe distinct ones in case of Multi-FEC case. This simple mechanism reen-

[10]Remind that if the vast majority of LSPs disappear for a given AS (due to the `Persistence` filter, we reinject the whole set of its LSPs in order to perform the classification. The difference with the standard classification is that those particular ASes are tagged as dynamic.

[11]By definition, the Egress LER is a convergence point while intermediate LSRs are not necessarily traversed by multiple LSPs.

forces the analysis by ensuring that any set of common IP addresses cannot be empty. Indeed, even in a worst case situation, at least the penultimate hop of an IOTP (the upstream of the Egress LER) can serve as a common IP. We already observed that results seem to be robust in this regard since they do not significantly change the classification except by removing the *Unclassified* class. Again, we preferred to work without relying on other hypothesis than the ones about MPLS label distribution and so did not present those results in this paper.

6. RELATED WORK

During the last years, MPLS has been more and more investigated by the research community. However, all the work performed up to now focuses on MPLS tunnels identification. For instance, Sherwood et al. investigated the presence of anonymous and hidden routers as part of DisCarte [6] using the IP Record Route option. However, they note that routers involved in an MPLS LSP do not record any IP address in the provided IP option space, and so, the record route option is not able to identify hidden routers. More recently, Sommers et al. [7] examined the characteristics of MPLS deployments that are explicitly identified using RFC4950 extensions, as observed in CAIDA's topology data. They also developed a methodology to infer MPLS tunnels in archived data where ICMP extensions are not recorded. Donnet et al. [8] provided algorithms for detecting MPLS tunnels depending on the way LSRs react to the `ttl-propagate` and RFC4950 options.

It has also been demonstrated that MPLS tunnels may have an impact on Internet topology discovery tools. For instance, the presence of MPLS tunnels may interfere with load balancing detection [1] or violate the destination-based forwarding [9].

On the contrary to this paper, none of these studies discussed the actual usage of MPLS tunnels by operators, neither investigated the dynamic of labels.

7. CONCLUSION

In this paper we presented a classification algorithm, LPR, that takes as input data obtained from `traceroute` campaigns, and produces a classification of MPLS paths used for transit traffic. This gives some insights on the different usages of MPLS, either as a basic encapsulation technique allowing, through LDP, scalability and independence between routing/forwarding in the ISP network and outside, or, as a more refined technique, allowing, through RSVP-TE and Traffic Engineering (TE), extended routing protocols to differentiate traffic and finely tune network usage.

We gave results based on the classification by LPR over five years of data collected by Archipelago. As a result, we found that the usage of MPLS is increasing over this period, and that the use of the basic encapsulation method seems predominant, with or without path diversity. Although the use of RSVP-TE is less common, traffic engineering is also well represented in some Autonomous Systems (ASes). Another (not so surprising) lesson, is that the use of MPLS varies greatly depending on the considered AS, from almost all mono-path (no diversity) to a wide deployment of traffic engineering. For a given AS, class distribution may also vary greatly over time.

Since most major vendor routers implement MPLS capabilities, deploying (or removing) MPLS is mainly a system configuration process. Similarly deploying LDP, RSVP-TE, or both is a matter of configuration. A last lesson is that when TE is deployed, in many cases, different LSPs between the same endpoints take the same IP path (TE using MPLS is almost as common as MPLS without path diversity). This seems to imply that bandwidth is sufficiently abundant for allowing all LSPs to share the same physical route. We aim at investigating the traffic distribution among MPLS tunnels in future works.

Acknowledgments

This work is partially funded by the European Commission funded mPlane ICT-318627 project.

8. REFERENCES

[1] B. Augustin, R. Teixeira, and T. Friedman, "Measuring load-balanced paths in the Internet," in *Proc. ACM Internet Measurement Conference (IMC)*, November 2007.

[2] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," Internet Engineering Task Force, RFC 3031, January 2001.

[3] K. Muthukrishnan and A. Malis, "A core MPLS IP VPN architecture," Internet Engineering Task Force, RFC 2917, September 2000.

[4] C. Srinivasan, L. P. Bloomberg, A. Viswanathan, and T. Nadeau, "Multiprotocol label switching (MPLS) traffic engineering (TE) management information base (MIB)," Internet Engineering Task Force, RFC 3812, June 2004.

[5] X. Xiao, A. Hannan, and B. Bailey, "Traffic engineering with MPLS in the Internet," *IEEE Network Magazine*, vol. 14, no. 2, April 2000.

[6] R. Sherwood, A. Bender, and N. Spring, "Discarte: a disjunctive Internet cartographer," in *Proc. ACM SIGCOMM*, August 2008.

[7] J. Sommers, B. Eriksson, and P. Barford, "On the prevalence and characteristics of MPLS deployments in the open Internet," in *Proc. ACM Internet Measurement Conference (IMC)*, November 2011.

[8] B. Donnet, M. Luckie, P. Mérindol, and J.-J. Pansiot, "Revealing MPLS tunnels obscured from traceroute," *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 2, pp. 87–93, April 2012.

[9] T. Flach, E. Katz-Bassett, and R. Govindan, "Quantifying violations of destination-based forwarding on the Internet," in *Proc. ACM Internet Measurement Conference (IMC)*, November 2012.

[10] L. Andersson, I. Minei, and T. Thomas, "LDP specification," Internet Engineering Task Force, RFC 5036, October 2007.

[11] D. Awduche, L. Berger, D. Gan, T. Li, and G. Srinivasan, V. ans Swallow, "RSVP-TE: Extensions to RSVP for LSP tunnels," Internet Engineering Task Force, RFC 3209, December 2001.

[12] L. Andersson and R. Asati, "Multiprotocol label switching (MPLS) label stack entry: EXP field renamed to traffic class field," Internet Engineering Task Force, RFC 5462, February 2009.

[13] E. Rosen, D. Tappan, G. Fedorkow, Y. Rekhter, D. Farinacci, T. Li, and A. Conta, "MPLS label stack encoding," Internet Engineering Task Force, RFC 3032, January 2001.

[14] P. Agarwal and B. Akyol, "Time-to-live (TTL) processing in multiprotocol label switching (MPLS) networks," Internet Engineering Task Force, RFC 3443, January 2003.

[15] P. Pan, G. Swallow, and A. Atlas, "Fast reroute extensions to RSVP-TE for LSP tunnels," Internet Engineering Task Force, RFC 4090, May 2005.

[16] C. Filsfils, S. Previdi, A. Bashandy, B. Decraene, S. Litkowski, M. Horneffer, R. Shakir, J. Tantsura, and E. Crabbe, "Segment routing with MPLS data plane," Internet Engineering Task Force, Internet Draft (Work in Progress) draft-ietf-spring-segment-routing-mpls-00, November 2014.

[17] C. Filsfils, P. Francois, S. Previdi, B. Decraene, S. Litkowski, M. Horneffer, I. Milojevic, R. Shakir, S. Ytti, W. Henderickx, J. Tantsura, S. Kini, and E. Crabbe, "Segment routing use cases," Internet Engineering Task Force, Internet Draft (Work in Progress) draft-filsfils-spring-segment-routing-use-cases-01, October 2014.

[18] S. Vissicchio, L. Vanbever, C. Pelsser, L. Cittadini, P. Francois, and O. Bonaventure, "Improving network agility with seamless BGP reconfigurations," *IEEE/ACM Transactions on Networking*, vol. 21, no. 3, pp. 990–1002, June 2013.

[19] E. Rosen and Y. Rekhter, "BGP/MPLS IP virtual private networks (VPNs)," Internet Engineering Task Force, RFC 4364, February 2006.

[20] D. Katz, K. Kompella, and D. Yeung, "Traffic engineering (TE) extensions to OSPF version 2," Internet Engineering Task Force, RFC 3630, September 2003.

[21] T. Li and H. Smit, "IS-IS extensions for traffic engineering," Internet Engineering Task Force, RFC 5305, October 2008.

[22] R. Bonica, D. Gan, D. Tappan, and C. Pignataro, "ICMP extensions for multiprotocol label switching," Internet Engineering Task Force, RFC 4950, August 2007.

[23] ——, "Extended ICMP to support multi-part messages," Internet Engineering Task Force, RFC 4884, April 2007.

[24] "NANOG traceroute," ftp://ftp.netbsd.org/pub/NetBSD/packages/distfiles/traceroute-nanog/traceroute.c.

[25] I. Castro, J. C. Cardona, S. Gorinsky, and P. Francois, "Remote peering: More peering without internet flattening," in *Proc. ACM CoNEXT 2014*, December 2014.

[26] k. claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet mapping: from art to science," in *Proc. IEEE Cybersecurity Applications and Technologies Conference for Homeland Security (CATCH)*, March 2009.

[27] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, "Avoiding traceroute anomalies with Paris traceroute," in *Proc. ACM Internet Measurement Conference (IMC)*, October 2006.

[28] University of Oregon, "Route views, University of Oregon Route Views project," see http://www.routeviews.org/.

[29] M. H. Gunes and K. Sarac, "Resolving anonymous routers in the Internet topology measurement studies," in *Proc. IEEE INFOCOM*, April 2008.

[30] D. Magoni and J.-J. Pansiot, "Analysis of the autonomous system network topology," *ACM SIGCOMM Computer communication Review*, vol. 31, no. 3, pp. 26–37, July 2001.

[31] Y. Vanaubel, J.-J. Pansiot, P. Mérindol, and B. Donnet, "Network fingerprinting: TTL-based router signature," in *Proc. ACM Internet Measurement Conference (IMC)*, October 2013.

[32] K. Keys, "Internet-scale IP alias resolution techniques," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 50–55, January 2010.