ISSN (Print): 0974-6846
Indian Journal of Science and Technology, *Vol 11(35), DOI: 10.17485/ijst/2018/v11i35/122033, September 2018* ISSN (Online): 0974-5645

Analysis of Load Balancing for a New Approach to Support Traffic Engineering in IPv6 Networks

Line Y. Becerra[1*] and Jhon J. Padilla[2]

[1]Faculty of Basic Sciences and Engineering, Universidad Católica de Pereira, Risaralda, Colombia; line.becerra@ucp.edu.co
[2]Faculty of Electronic Engineering, Universidad Pontificia Bolivariana, Bucaramanga, Colombia; jhon.padilla@upb.edu.co

Abstract

Objective: To propose a solution to support traffic engineering in IPv6 networks; such proposal is based on IPv6 capabilities. **Methods/Analysis:** Our proposal uses the IPv6 flow label field for packet switching in IPv6 networks and it also uses extended traffic engineering protocols like RSVP-TE and OSPFv3-TE. An advantage of our approach is that an MPLS transport network is not required to support traffic engineering. Our solution makes use of the tunneling concept, which has a high potential to support traffic engineering because it allows separation of different traffic among service/users in different tunnels. In this paper, we describe the main characteristics of our proposal and also, we present the evaluation of load balancing, which is a typical situation in traffic engineering studies. We compare our approach with MPLS performance because it is a technology commonly used to support traffic engineering. **Findings/Results**: Results show that load balancing in our solution has similar performance than MPLS when the number of tunnels over links is optimized. **Improvements**: This evaluation proves that our layer-3 proposal has traffic engineering capabilities in IPv6 networks independently of lower layers.

Keywords: IPv6, IPv6 Flow Label, Load Balancing, Packet Switching, Traffic Engineering

1. Introduction

An important problem in the current Internet is the use of the shortest path routing algorithm, which leads to congestion of certain common paths to many communications. One solution to this problem is to use switching technologies that allow traffic engineering support. One of the most used is MPLS[1]; however, this requires to transport an additional header at 2.5 layer level.

On the other hand, IPv6[2] is the protocol of the next generation networks which offers significant advantages for the current trend technologies such as a great number of addresses and the provision of quality of service, mobility, among others. A complete study of IPv6 deployment is described in[3,4]. A new field in the IPv6 header is the called "flow label". In IETF, several debates were presented regarding the purpose of this field. In RFC6294[5] describes the questions made for the IETF designers, such questions were as follows: "Was it to be key in handling fast switching? Was it to be meaningful to applications and used to specify quality of service? Must it be set by the sending host? Could it be set by routers? Could it be modified in route? Must it be delivered with no change? Because of these uncertainties, as well as more urgent work in other areas, the IPv6 flow label was ignored by implementers and today it is set to zero in almost every IPv6 packet"[5]. Due to this reason, several proposals to use the IPv6 flow label field have been made for different purposes. A study of these proposals is presented in[5] and [6]. The flow label field initially was specified with 28-bit length in RFC1710[7]; then, it was reduced to 24 bits by RFC1883[8] and finally established as 20 bits by RFC2460[2]. Since the definition of this field, there have been several uncertainties about its use such that, in RFC2460 it was defined as experimental and subject to change[7].

*Author for correspondence

The IETF performed several preliminary works to its full specification, such as in[9–12], until a more detailed specification was published in RFC3697[13], which provided useful information for its use and was in force for approximately seven years. During this time, several solutions were published by researchers, proposing methodologies of the use of flow label field for different purposes, such as quality of service support, packet switching and packet filtering, among others, but these solutions violated, in one manner or another, the recommendations given by RFC3697[13].

Because of the above, the IETF performed a study of use cases for the "flow label," published in RFC6294[5]. Such a study was performed because they found that, the flow label field was not used in practice. Therefore, RFC6294[5] describes a study of various published proposals focused on the use of the flow label field for different purposes and highlight the inconsistencies found between such proposals and flow label field specification, RFC3697[13]. Then, IETF took into account the minimal practical use of the flow label field at that moment, and IETF was motivated to change that specification to clarify it and introduce some additional flexibility; as a result, RFC6436[14] was published. Then, IETF made a recommendation to update RFC3697[13] and as a consequence, the new specification of the flow label field was published in RFC6437[15], making RFC3697[13] obsolete.

Several proposals have been published by different researchers; these proposals are described as follows. In[16,17], the authors describe a proposal called "IPv6 Label Switching Architecture" (6LSA). In this architecture, the flow label field is used to packets switch; every packet identified with the same flow label value must receive the same treatment and should be sent to the same hop. 6LSA works similarly to MPLS since it considers that a label has significance only between components of its architecture and in the manner how it establishes the flow label in every router. Unlike the traditional routing techniques, but similar to MPLS, 6LSA packets are classified within an FEC and routers send packets over different paths depending on the FEC.

In[18], the authors propose a combination of the flow label and class of service fields as a switching tag and to support QoS, in a similar manner to how MPLS works. This proposal uses the DiffServ Code Point (DSCP) RFC2474[19] to indicate that the flow label is a switching tag. Another similar work based on QoS and packet switching is described in[20], which was designed as a hop-by-hop option. Alternately, [21]presents a new model of sending packets by flow label to improve IPv6 packet switching, which requires service differentiation and as the author says, it provides more effective functions than MPLS.

Other use cases of the flow label field as QoS support are described in[22–30]. Some uses for mobility are proposed in[31–35]. Additionally, the flow label was used to identify an IPv4-in-IPv6 tunnel in[36], also as a tool for load balancing by equal Cost Multi-Path Routing in[37,38]. Besides, flow label field was proposed as a mechanism of traffic filtering in[39,40] and for security purposes in[41–43].

On the other side, Internet traffic engineering is responsible for the optimization and evaluation of the performance of IP networks in operation. The aim is to improve the network performance by optimizing the use of resources and traffic by applying technologies and scientific principles to allow the measurement, characterization, modeling and control of Internet traffic. In the last fifteen years, different solutions have been proposed to support Internet traffic engineering. A previous study of traffic engineering proposals is described in[44]. We organized these proposals in five categories: TE based on IP by link weight optimization, TE based on MPLS, TE based on LISP, TE based on Segment Routing and TE based on IPv6 facilities. These categories will be explained as follows.

IP-based proposals are focused on the IGP link weight adjustment. The first IP-based TE solution was proposed by[45–47]. The main goals in their approach were to set the link weights of Interior Gateway Protocols (IGPs), such as OSPF and IS-IS, according to the given network topology and traffic demand to control intradomain traffic and meet TE objectives. In[48–52] algorithms and optimization problems to set link weights are proposed. Alternately, a generalized routing framework to realize the optimal TE, which can potentially be implemented via OSPF- or MPLS-based approaches, is presented in[53].

The concept of traffic engineering in MPLS-based environments was introduced in[54,55], by setting up dedicated switched paths (LSPs). The MPLS specifications are detailed in RFC3031[1]. TE based on MPLS can provide an efficient paradigm for traffic optimization. The most distinct advantage of MPLS-based TE is its capability of explicit routing and arbitrary splitting of traffic, which is highly flexible for both routing and forwarding optimization purposes. Many solutions have been presented in the literature using MPLS for traffic engineering and QoS purposes; many of them are directed to propose constraint-based routing algorithms, and studies over these proposals have been presented in[56,57].

In recent years, an entirely different strategy has been proposed by the LISP protocol (Locator/ID Separation Protocol). The LISP Specifications are contemplated in RFC6830[58]. LISP is a network-layer-based protocol that enables separation of IP addresses into two new numbering spaces: Routing Locators (RLOCs) and Endpoint Identifiers (EIDs). RLOCs are topologically assigned to network attachment points; these locators are used for routing and forwarding of packets through the network[58]. EIDs are assigned independently from the network topology; these identifiers are used for numbering devices and are along administrative boundaries[58]. In[59], the authors describe how LISP re-encapsulating tunnels can be used for traffic engineering purposes. Thus, a packet can take an administratively specified path, a congestion avoidance path, a failure recovery path or multiple load-shared paths as it travels from the ITR (Ingress Tunnel Router) to the ETR (Egress Tunnel Router). By introducing an Explicit Locator Path (ELP), an ITR can encapsulate a packet to a Re-encapsulating Tunnel Router (RTR), which decapsulates the packet and then encapsulates it to the next locator in the ELP. Some documents are dealing with traffic engineering support by LISP include[60–65].

With respect to TE based on Segment Routing (SR), it has been recently proposed as an alternative traffic engineering technology. Enabling relevant simplifications in control plane operations takes advantages of the source routing paradigm. IETF is working on standardizing of the segment routing architecture[66]. In SR a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based and a segment can have a semantic local to an SR node or global within an SR domain[66]. Some works have been proposed to support traffic engineering by segment routing, such as in[67], which details the segment routing policy for traffic engineering. In[68], the authors consider the problem of determining the optimal parameters for segment routing in the offline and online cases. Alternately, in[69] a Label Encoding Algorithm for MPLS Segment Routing is proposed. In[70], the authors present an SR path assignment algorithm for the flow assignment problem. Finally, in[71], the authors purpose ILP models and heuristics that are successfully utilized to assess the TE performance of SR-based packet networks.

In TE based on IPv6 facilities, are proposals that use IPv6 issues such as the flow label and others. Here, there are allocated 6LSA[72] and IPngls[17]. The main advantage of this category of proposals is that there is no other necessary layer 2.5 technology to support TE. Additionally, they try to take advantage of benefits that are not being used for IPv6 yet, which is still in deployment and will be the main protocol for the Internet for a long time. Our proposal PSA-TE6 falls into this category. Because it has some similarities with the other mentioned proposals, we provide a summary of the similarities and differences according to important characteristics for supporting traffic engineering in Section 2.

In this paper, we describe the main characteristics of our proposal which we have called PSA-TE6 and also, we present the evaluation of load balancing, which is a typical situation in traffic engineering studies. We compare our approach with MPLS performance because it is a technology commonly used to support traffic engineering. Both solutions (PSA-TE6 and MPLS) use the tunneling concept (referred as Label Switching Path or LSP), which has a great traffic engineering potential that allows separation of diverse traffic for different service/users in different tunnels. However, in both cases (PSA-TE6 and MPLS) overload difficulties in routers can be presented if the number of tunnels is not limited and the load balancing is not optimized. In consequence, from a management point of view, it may also be desirable to limit the number of tunnels on a router or a link. For this reason, in this paper, we present the PSA-TE6 evaluation by MIP (Mixed-Integer Programming) formulation in order to balance load and to minimize the maximum number of tunnels over all links.

This article is organized as follows: Section 2 explains our PSA-TE6 proposal. Section 3 presents the evaluation of the load balancing in PSA-TE6. Section 4 shows the results. Finally, Section 5 describes conclusions and future works.

2. Packets Switching Architecture to Support Traffic Engineering in IPv6 Networks (PSA-TE6)

PSA-TE6 is a new solution proposed to support traffic engineering in IPv6 networks. The goal of this architecture is to use the IPv6 flow label field for packet switching in IPv6 networks in a manner similar to how MPLS[1] works but without the need of an MPLS architecture being installed. This proposal is created as a result of the study of the IPv6 flow label field from its creation to the latest recommendations of the IETF, plus

the analysis of various proposals for the use of the IPv6 flow label field[5,6] and by observing its structure, which shows a great similarity to the MPLS label regarding size (20 bits) and contents.

In the literature, two proposals for label switching have been presented; these are described in[17,72]. In[72], the authors propose the forwarding of IPv6 packets using label switching techniques, with similar advantages to the Multiprotocol Label Switching (MPLS) architecture. This forwarding process is performed by means of a mapping of all MPLS header fields within the IPv6 header. Alternately, [17]introduces an architectural framework to use the IPv6 packet header flow labels to set up labeled paths in a similar manner to how MPLS works. Although our proposal and the two mentioned above use the label switching concept via IPv6 flow label switching in a similar manner to how MPLS works, our proposal presents important differences that strengthen traffic engineering support in IPv6 networks. In Table 1, such differences are described.

Table 1. Comparison between proposals using IPv6 flow label to switch packets

Characteristic	6LSA	IPNGLS	PSA-TE6
It uses a Packet switching Mechanism via the IPv6 Flow Label	Yes, it is.	Yes, it is	Yes, it is.
It describes the routing table fields	Yes, it is.	No, it is not.	Yes, it is.
It uses the label switching paths like MPLS	Yes, it is.	Yes, it is.	Yes, it is.
It splits the label value in different fields	Yes. It divides the flow label field into three parts	No. It maps the label value directly from MPLS.	No. Uses the 20 bits without dividing it. The value is assigned in a similar manner as MPLS.
It uses label operations: push-swap-pop	Yes, it is.	It is not described. It is assumed like in MPLS.	Yes, it is.
It describes how the label is generated	Described three ways: 1. Locally based on a certain algorithm or policy. 2-In the entry packet like a flow label from the source node. 3-Distributed through a label distribution process.	No, it is not.	Yes, it is distributed through a label distribution process in a similar manner as MPLS.
It uses a label distribution protocol	It contemplates the option of using it for case 3 but does not assume one in particular	It contemplates the possibility of using it but does not assume one in particular.	It contemplates using RSVP-TE.
It defines the label-FEC relation in every router	Yes, it is.	No, it is not.	Yes, it is.
It defines the operation within a Domain and defines the elements that comprise it.	Yes, it is.	No, it is not.	Yes, it is.
It allows Label Stacking	No, it is not allowed.	Yes, it is allowed by an IPv6 option header.	Yes, it is allowed by Generic Packet Tunneling in IPv6 or using an IPv6 option header.
It uses extended routing protocols to support traffic engineering	No, it is not.	No, it is not.	Yes, it is.
It uses constraint-based routing algorithms	No, it is not.	No, it is not.	Yes, it is.
It defines a flow label restoration mechanism	No, it is not.	No, it is not.	Yes, it is.

2.1 Principles of Design of the PSA-TE6 Proposal

Alternately[15] and the use cases of the IPv6 flow label in[5]. Both documents describe three basic rules for the use of the IPv6 flow label field, which are as follows:

- IPv6 nodes MUST NOT assume any mathematical or other properties of the flow label values assigned by source nodes[5,15].
- Router performance SHOULD NOT be dependent on the distribution of the flow label values.Specifically, the flow label bits only make poor material for a hash key[5,15].

The flow label must not be changed in route but allow routers to set the label on behalf of hosts that do not do so[5,15].

According to the fundamental rules mentioned above, our proposal would violate the first part of the rule (iii), i.e., "not to be changed in route." However, the second part of the rule (iii) states: "but allow routers to set the label on behalf of hosts that do not do so," which is an open issue discussed in RFC6294[5]. Additionally, RFC6294 mentioned with respect to rule (iii) that it does not exclude the flow label from being used for switching or routing purposes.

Similarly, RFC6294[5] makes recommendations for designers who use the IPv6 flow label for packet switching. Such recommendations refer to overlooking the rules within a given domain. Within that domain, routers could establish and interpret the IPv6 flow label field as it was designed and then in the router of the last hop of the domain, the label should be set to zero; this rule should be enforced for packets arriving at the domain with the label set as zero.

For the case in which packets arrive at the domain with a label value other than zero, an alternative recommendation given for RFC6294 is to define a hop-by-hop option header to carry the original label through the domain so that it can be restored at the output of the domain. All those recommendations were taken into account in the design of the proposed PSA-TE6 solution. Alternately, OSPF and IS-IS protocols were extended in response to the requirements of RFC2702 and these extensions are also associated with support MPLS traffic engineering (OSPF-TE and IS-IS-TE). Therefore, in the design of PSA-TE6, such protocols also play an important role in the forwarding process since PSA-TE6 uses label switching and its goal is to provide traffic engineering. Thus, our PSA-TE6 proposal includes the use of the OSPFv3-TE protocol[73], which is the extended protocol for working on IPv6 networks and supporting traffic engineering. Additionally, the RSVP-TE signaling protocol[74], which was extended to the MPLS label distribution, has been taken and defined as a tool for the distribution of IPv6 flow labels in our proposal.

Finally, in the PSA-TE6 proposal, it is necessary to find appropriate constraint-based paths. This functionality is provided by CBR (Constraint-based Routing) algorithms, which select the best route that corresponds to the constraint set. Restrictions can be imposed by administrative policies, quality of service or traffic engineering requirements[75]. In the last fifteen years, many CBR algorithms have been proposed, and in[56,75,76], a study is presented. We have selected the CSPF (Constraints Shortest Path First) algorithm for use in our proposal because CSPF is one of the most common algorithms used to address this issue[77].

2.2 Architecture of the PSA-TE6 Proposal

The PSA-TE6 architecture is comprised of the following elements (see Figure 1): Ingress/egress 6DER (Ingress/Egress PSA-TE6 Domain Edge Router); 6DTR (PSA-TE6 Domain Transit Router); and 6DLSP (PSA-TE6 Domain Label Switching Paths). These elements must operate under a PSA-TE6 domain, which we have denominated 6D. The idea of having a domain with PSA-TE6 is referred at RFC6294[5] in Section 4, as a recommendation to proposals using the IPv6 flow labels to switch packets, as stated previously.

To explain the PSA-TE6 proposal, we assume a network with five nodes as in Figure 1. It is a network that works under the IPv6 protocol and it is composed of a 6D domain with routing and signaling protocols as OSPFv3-TE[73] and RSVP-TE[74] respectively. OSPFv3-TE is responsible of routing information flooding, regarding the network topology and traffic engineering information. Meanwhile, RSVP-TE is responsible for the establishment of the label switched paths and the distribution of IPv6 flow label values. It is important to emphasize that the RSVP-TE protocol has an object named "Label" which was created to distribute MPLS labels, however, in our solution we use "Label" object of RSVP-TE to distribute IPv6 flow labels. This can be done because the MPLS labels and IPv6 flow labels have the same length (20 bits). On the other hand, to establish IPv6 label switched paths, it is necessary to find an appropriate path, not necessarily the shortest, but one based on constraints. Several algorithms have been proposed about it, but in this proposal, we assume that PSA-TE6 works with CSPF algorithm (Constrained Shortest Path First), which is one of the

most used. CSPF is based on the Dijkstra algorithm with a modification: The addition of a bandwidth constraint; this algorithm is explained in[78].

In the 6D domain, Ingress/Egress-6DER routers must be able to read the IPv6 header; they also should set the 6DLSPs (IPv6 Flow Label Switching Paths) and perform label insertion and deletion operations (push and pop operations). Meanwhile, 6DTRs routers must be able to read the first 64 bits of the IPv6 header (see Figure 2) to switch and to route packets through exchanges of IPv6 flow labels and they also should be able to do the necessary operations for IPv6 label stacking. The fact that the 6DTRs routers have to read the first 64 bits of the IPv6 header is because the DS (Differentiated Services), IPv6 flow label and TTL (Time to Live) fields of the packet are located in this part of the IPv6 header and must be known to support quality of service and label packet switching (see Figure 2). It is possible to do this in one reading operation with the currently used 64-bit network processors.

When it is necessary to establish a new communication and the first packet reaches the Ingress-6DER, it reads the IPv6 header and captures the source and destination address. Then, by means of RSVP-TE, the establishment of the label switching path and label distribution process are accomplished using the standard procedures of such protocols. The label switching path is found by the constraint-based routing algorithm based on the OSPFv3-TE information. Then, the ingress-6DER puts the label value in the IPv6 flow label field in all the packets belonging to the correspondent flow and sends them to the next hop. This proposal initially assumes that packets come from a domain that does not use the IPv6 flow label so that this value will be zero according to RFC6437[15]. Then, the packet travels on that path and each 6DTR interior router will exchange the label (swap operation) in each packet and then send the packet to the appropriate output interface. When the packet arrives at the Egress-6DER, it removes the label (pop operation) and sends the packet to the destination. The router also returns the flow label field to its original value (or zero) according to RFC6294.

In the packet forwarding process, it is necessary to have information for the operations performed on the IPv6 flow label field, which must be analyzed before packet forwarding to the next hop. For this proposal, routers have an FIB (Forwarding Information Base) that is specific for each router, i.e., if it is 6DER or 6DTR. This FIB can be of two types: one that maps a FEC (Forwarding Equivalent Class) to N6FLD (Next Hop IPv6 Flow Label Forwarding Data), which we have called FTN6 (Forwarding Equivalence Class To Next Hop IPv6 Flow Label Forwarding data) and another FIB that maps an incoming IPv6 flow label to N6FLD, which we have called I6LTN (incoming IPv6 flow label to next Hop IPv6 flow label Forwarding data). These tables are similar in content and function to those used in MPLS[1]. These tables will be available in routers of IPv6 flow label switching according to their roles in the PSA-TE6 domain. In IP networks, a router considers that two packets belong to the same FEC if there is any address prefix X in the routing table such that X prefix is the longest match for each packet's destination address. In the PSA-TE6 architecture, the FEC will be determined in the ingress routers where the 6DLSPs are established.

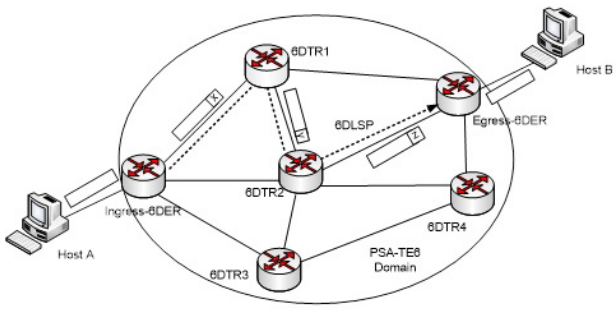


Figure 1. PSA-TE6 architecture.

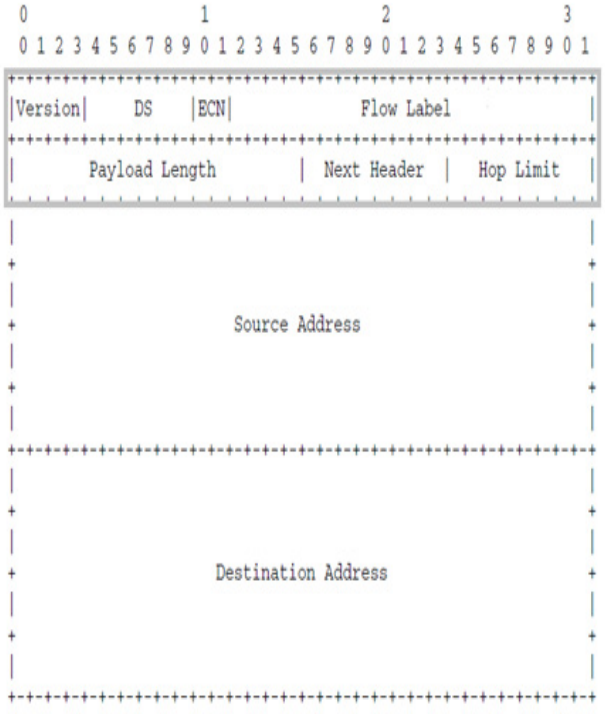


Figure 2. IPv6 header RFC2460[2], RFC8200[78] and RFC2474[19].

3. Evaluation of the Load Balancing in PSA-TE6

In the packet forwarding process, PSA-TE6 establishes label-switching paths and performs packets switching by label swapping in a similar manner as MPLS works. However, in this process, there is a difference with regard to the packet length since MPLS insert a header of 32 bits which contains the label value. On the other hand, PSA-TE6 does not require an additional bit since PSA-TE6 uses the flow label field and this field is contained in the IPv6 header (see Figure 3).

It is important to determine whether traffic distribution is affected by such difference in packet length. Therefore, this article has oriented the evaluation of the PSA-TE6 solution towards the analysis of traffic distribution under load balancing, which is a typical situation for traffic engineering. PSA-TE6 uses the tunneling concept (also referred as 6DLSPs) which has a great traffic engineering potential that allows separation of diverse traffic of different service/users in different tunnels in a similar manner as MPLS works. In both technologies (PSA-TE6 and MPLS), overload difficulties in routers can be presented if the number of tunnels is not limited and the load is not balanced. Therefore, from a management point of view, it may also be desirable to limit the number of tunnels on a router or a link. For this reason, we present an MIP formulation whose objective function problem is to carry different traffic classes in a network through the creation of tunnels in such a way that the number of tunnels on each router/link is minimized and load balanced. Network topologies, link capacities, traffic demands and candidate paths are previously specified for the optimization problem. A similar MIP formulation is described in[79], but we do a modification to that model to compare load balancing in PSA-TE6 versus MPLS. To do this, we introduce a parameter λ, which represents the effect of differences in the traffic demands due to packet lengths for each technology. Such difference is referred to the addition of the MPLS header in IP/MPLS networks and the not existence of the MPLS header when PSA-TE6 is used.

Now we introduce a mathematical model of the problem using the notation as in[79]. We use the identifier $d = 1,2,\ldots,D$ to denote a demand associated with a node pair (source and destination nodes) that require bandwidth h_d to be routed in the network. The volume h_d of demand d can be carried over multiple tunnels (paths) from ingress to egress of the tunnel. We use index $p = 1,2,\ldots,P_d$ to denote candidate paths for demand d. The fraction of the demand volume for demand d to be carried on tunnel p is denoted as X_{dp}. Note that X_{dp} is a continuous decision variable. We have the demand constraint, which guarantees that the sum of all fractional flows X_{dp} over all candidate paths $p = 1,2,\ldots,P_d$ must add up to the whole demand volume h_d in (Equation 1).

$$\sum_{p} X_{dp} = 1 \tag{1}$$

Since a flow could be a very small fraction of traffic demand, we establish a lower bound on the fraction of a flow on a path. We use a positive quantity ε to be the lower bound on such fraction of flow on a tunnel (path) and we use the binary variable $U_{dp} = 1$ to denote the selection of a tunnel if the lower bound is satisfied, and 0, otherwise. We add a parameter λ, which represents a bandwidth percentage of each demand that differentiates one technology from the other according to the length of the packet for each one (see Figure 3). We have the following two constraints:

$$\varepsilon U_{dp} \leq \lambda h_d X_{dp} \tag{2}$$

$$X_{dp} \leq U_{dp} \tag{3}$$

Constraint (Equation 2) assures that if a tunnel is selected, then the tunnel must have at least the fraction of allocated flow which is set to ε. Constraint (3) guarantees that if a tunnel is not selected, then the flow fraction associated with this tunnel should be forced to be equal to 0.

Since the network topology is given and link capacity is known, we must assure that physical link capacity C_e of link e is not exceeded. On the other hand, we use the binary variable δ_{edp}, which is 1 if link e belongs to path p realizing demand d, and 0 otherwise. Thus, the capacity feasibility constraint is the following:

$$\sum_{d} h_d \lambda \sum_{p} \delta_{edp} X_{dp} \leq C_e \tag{4}$$

The left-hand side of (Equation 4) is the flow on link e, which is calculated taking into account all demands $d = 1,2,\ldots,D$, the candidate paths $p = 1,2,\ldots,P_d$, whether the given demand d uses path p ($\delta_{edp} = 1$) and the flow fraction X_{dp}. Here, we add λ parameter that represents an additional bandwidth percentage of each demand that differentiates PSA-TE6 of MPLS. The number of tunnels on link e is given by (Equation 5):

$$\sum_{d} \sum_{p} \delta_{edp} U_{dp} \tag{5}$$

The complete formulation is described in the Table 2 and the network topologies used in this evaluation are shown in Figure 4. Since the goal of optimization is to

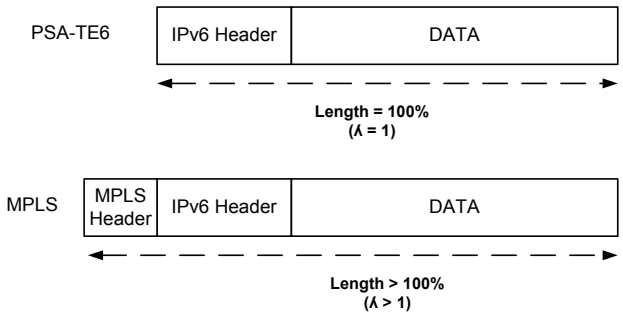
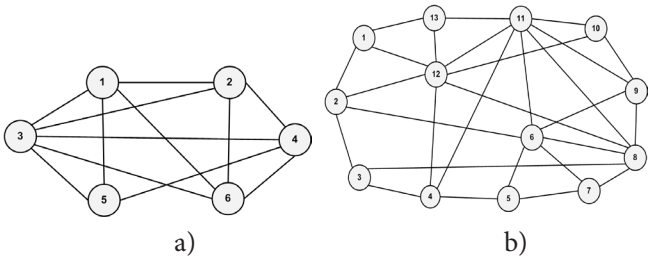


Figure 3. Comparison of the packet length of PSA-TE6 and MPLS.

minimize the total number of tunnels, the objective function minimizes a number (r) that represents the maximum number of tunnels over all links (Equation 6). Constraint in (Equation 7) selects the fraction of the demand volume to be carried on a tunnel. Constraint in (Equation 8) is the capacity feasibility constraint. Constraint in (Equation 9), restricts the fraction of demand to a minimum designated as ε. Constraint in (Equation 10) forces the flow fraction to be zero if a tunnel is not selected. Finally, the constraint in equation (Equation 11) computes the number of tunnels over each link.



a) b)

Figure 4. Test Networks.

Such as was mentioned above, we add a parameter λ in constraint 8 and 9 (Table 2), which represents a bandwidth percentage of each demand that differentiates one technology from the other according to the length of the packet for each one. As we said before, such difference is caused in MPLS because it adds a header to establish a label in the packet forwarding process, which is not necessary for PSA-TE6. We assume that applications keep the same packet periodicity and in consequence, a reduction in the packet length results in a reduction in traffic demand for such flow. As we can see in Figure 3, PSA-TE6 packets are shorter than MPLS packets due to additional MPLS header in IP/MPLS networks. Thus, we take as a reference to PSA-TE6 as λ=1 (see Figure 3) and λ>1 for MPLS case.

Table 2. MIP formulation

Parameters:
d = traffic demand associated with a node pair and a traffic class.
h_d *= bandwidth required for each demand.*
P_d *= Number of different possible tunnels for each demand d.*
λ = bandwidth percentage of each demand that differentiates one technology from the other according to *the length of the packet.*
ε *= lower bound on the fraction of a flow on a path.*
C_e *= link Capacity*
δ_{edp} *= Indicator link-path.*
Variables:
r = Maximum number of tunnels over all links. It is Integer
X_{dp} *= Fraction of demand volume d to be carried on tunnel p.*
It is a continuous and non-negative variable.
U_{dp} *= Select a tunnel (=1) if the lower bound is satisfied (and 0, otherwise).It is a binary variable.*
Objective:

$$Mininize_{x,u,r} F = r \quad (6)$$

Constraints.

$$\sum_p X_{dp} = 1 \quad (7)$$

$$\sum_d h_d \lambda \sum_p \delta_{edp} X_{dp} \le C_e \quad (8)$$

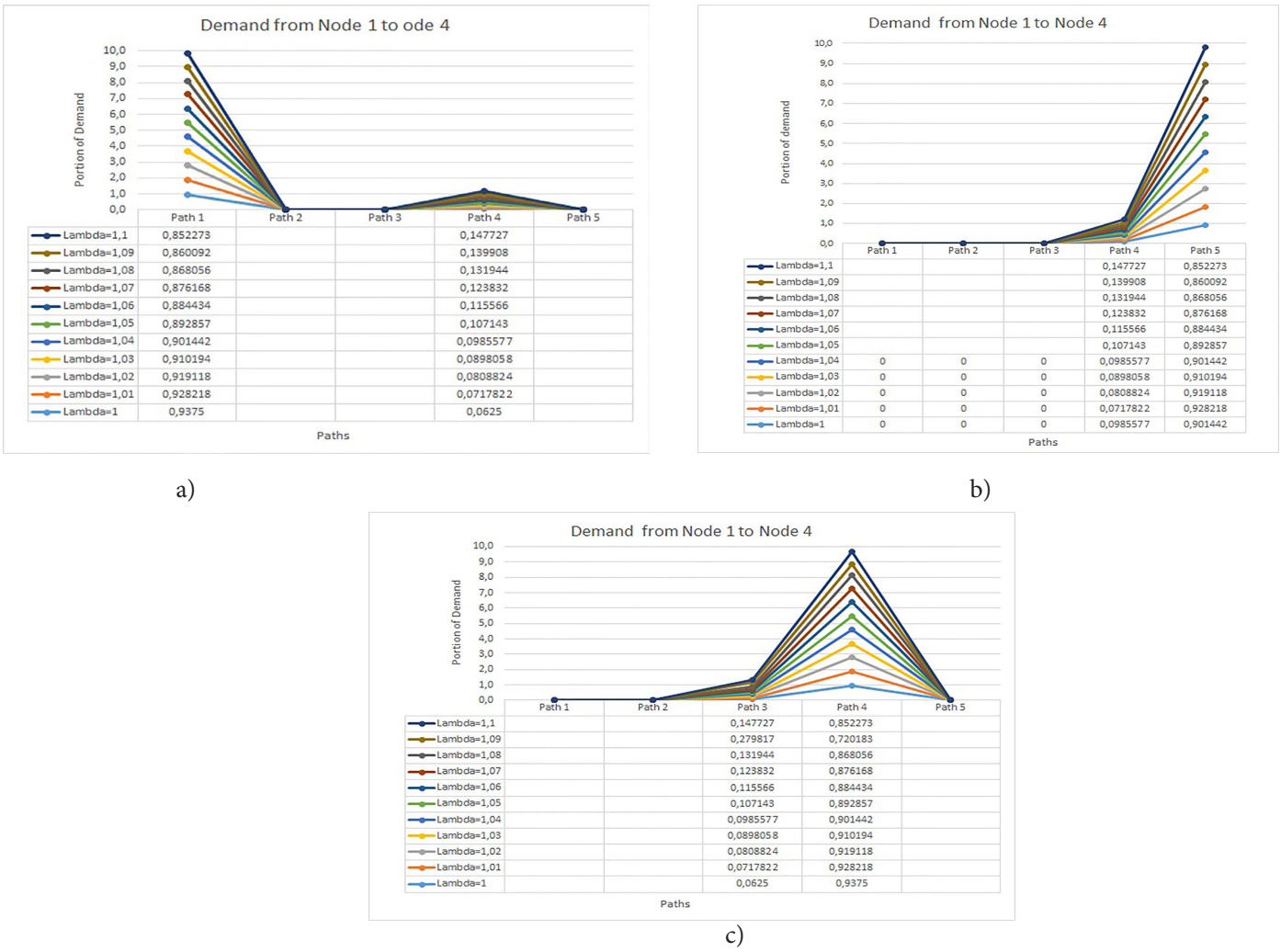
$$\varepsilon U_{dp} \le \lambda h_d X_{dp} \quad (9)$$

$$X_{dp} \le U_{dp} \quad (10)$$

$$\sum_d \sum_p \delta_{edp} U_{dp} \le r \quad (11)$$

4. Results

As we said before, in order to evaluate PSA-TE6 and compare it with MPLS performance, we assume that bandwidth percentage of each demand that differentiates one technology from the other according to the length of the packetis represented as parameter λ; we take as reference to PSA-TE6 as λ=1 (see Figure 3), and λ>1 for MPLS case. As we can see in Figure 3, PSA-TE6 packets are shorter than MPLS packets due to additional MPLS header in IP/MPLS networks. For evaluation purposes, we give several values to λ to analyze the effect of packet length in the traffic distribution over the established paths to compare both technologies.



a)

	Path 1	Path 2	Path 3	Path 4	Path 5
Lambda=1,1	0,852273			0,147727	
Lambda=1,09	0,860092			0,139908	
Lambda=1,08	0,868056			0,131944	
Lambda=1,07	0,876168			0,123832	
Lambda=1,06	0,884434			0,115566	
Lambda=1,05	0,892857			0,107143	
Lambda=1,04	0,901442			0,0985577	
Lambda=1,03	0,910194			0,0898058	
Lambda=1,02	0,919118			0,0808824	
Lambda=1,01	0,928218			0,0717822	
Lambda=1	0,9375			0,0625	

b)

	Path 1	Path 2	Path 3	Path 4	Path 5
Lambda=1,1				0,147727	0,852273
Lambda=1,09				0,139908	0,860092
Lambda=1,08				0,131944	0,868056
Lambda=1,07				0,123832	0,876168
Lambda=1,06				0,115566	0,884434
Lambda=1,05				0,107143	0,892857
Lambda=1,04	0	0	0	0,0985577	0,901442
Lambda=1,03	0	0	0	0,0898058	0,910194
Lambda=1,02	0	0	0	0,0808824	0,919118
Lambda=1,01	0	0	0	0,0717822	0,928218
Lambda=1	0	0	0	0,0985577	0,901442

c)

	Path 1	Path 2	Path 3	Path 4	Path 5
Lambda=1,1			0,147727	0,852273	
Lambda=1,09			0,279817	0,720183	
Lambda=1,08			0,131944	0,868056	
Lambda=1,07			0,123832	0,876168	
Lambda=1,06			0,115566	0,884434	
Lambda=1,05			0,107143	0,892857	
Lambda=1,04			0,0985577	0,901442	
Lambda=1,03			0,0898058	0,910194	
Lambda=1,02			0,0808824	0,919118	
Lambda=1,01			0,0717822	0,928218	
Lambda=1			0,0625	0,9375	

Figure 5. Traffic distribution over several paths in a network of 6 nodes for 1, 3 and 5 demands.

The topologies used in this evaluation are shown in Figure 4. Over both topologies, we evaluate traffic distribution by varying the number of demands from 1 to 5, which are sent from an ingress node to egress node for different values of λ. We simulate the path distribution for λ = 1 until λ = 1.1 with steps of 0.01 in each running. In every experiment, each demand can be distributed in five possible candidate paths.

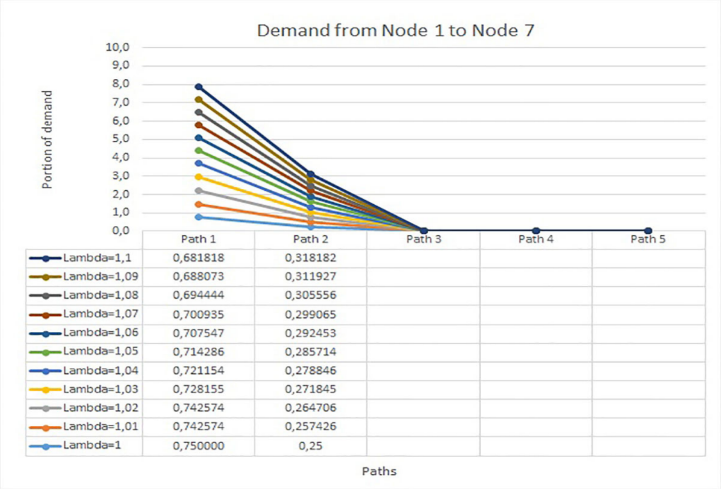
Results for a network of 6 nodes and 1, 3 and 5 demands, are shown in Figure 5. Also, results for a network of 13 nodes for 1, 3 and 5 demands are shown in Figure 6.

As we can observe, in both network topologies the traffic distribution is similar to the same number of demands when we vary lambda values from 1 to 1.1. For example in Figure 5 (6 nodes network), for three demands, the traffic

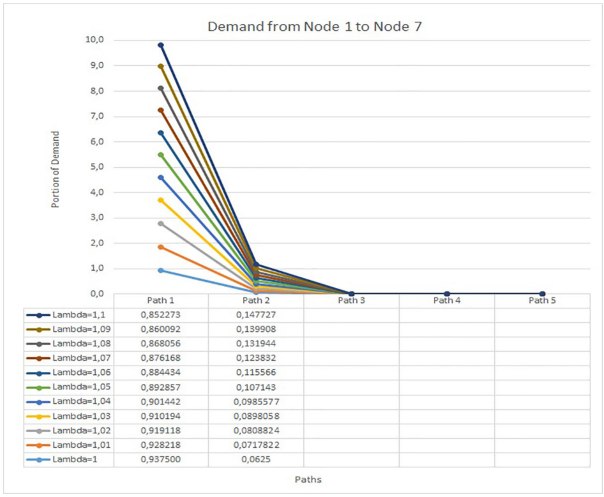
distribution in demand from N1 to N4, is similar for several values of lambda. Similar behavior can be observed in Figure 6 for three demands in a 13 nodes network for demand from N1 to N7. Although the selected paths are different for each network topology, we did not find differences in the proportion of traffic distribution in both topologies for several numbers of demands while the value of lambda is changing. Thus, such traffic distribution variations remain in the same proportion of lambda variation (1% to 10% of the occupation of a link or path).

To support our range of lambda ($\lambda = 1$ to $\lambda = 1.1$), we take the VoIP service as an example for the worst case in packet length difference between MPLS and PSA-TE6. We calculate a typical packet length for this service as follows. For typical G.729 codec, the packet length is 108 bytes, we should add headers to this payload as: 40 bytes of IPv6 Header, plus 8 bytes of UDP header and 12 bytes of RTP header; if we add the MPLS header (4 bytes), we will take a total length of 172 bytes for the case $\lambda>1$. Thus, in PSA-TE6, we will take only 168 bytes (without MPLS header), i.e. $\lambda = 1$, which is 2.3% less than MPLS case; this is $\lambda = 1.023$ for MPLS.

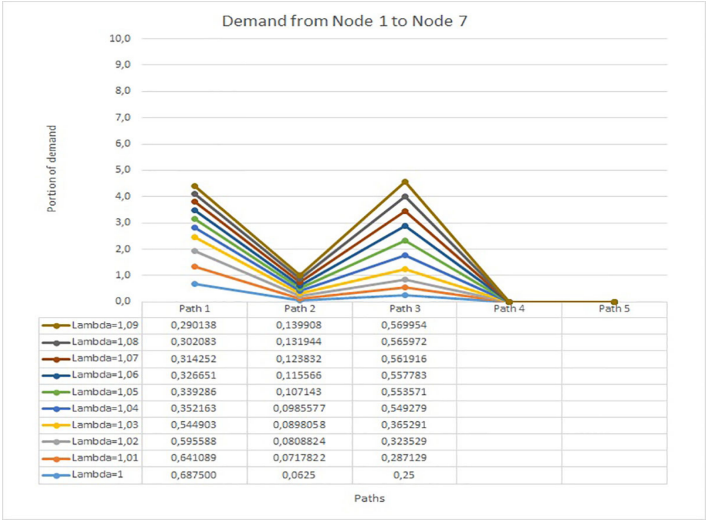
As a conclusion, we have demonstrated that PSA-TE6 has the same behavior than MPLS in load balancing scenario when the number of tunnels over links is optimized. Also, with PSA-TE6, IPv6 networks have the advantage that they do not require another technology to support traffic engineering. Besides, IPv6 networks would occupy less bandwidth than MPLS.



a)



b)



c)

Figure 6. Traffic distribution over several paths in a network of 13 nodes for 1, 3 and 5 demands.

5. Conclusions and Future Works

This study presented a new proposal to support traffic engineering based on the use of the IPv6 flow label field for packet switching, which has been called PSA-TE6. For the evaluation of our proposal, we performed an analysis of load balancing and the minimization of the number of tunnels on each link in order to evaluate the behavior of PSA-TE6with respect to MPLS. Results show that small differences in packet length for both technologies (2.3% for the worst case) are not meaningful and the traffic distribution behavior over the links for both technologies is the same when the number of tunnels over links is optimized. Then, our proposal PSA-TE6 is a useful tool to make traffic engineering in IPv6 networks and if PSA-TE6 is implemented in IPv6 networks, MPLS could be not necessary to offer traffic engineering. Besides, IPv6 networks would occupy less bandwidth than MPLS. It is important to highlight that PSA-TE6 solution satisfies the recommendations of the IETF for using the IPv6 flow label field for switching control and such recommendations support that IPv6 flow label can be used for packet switching in combination with routing and signaling protocols (OSPFv3-TE and RSVP-TE respectively) that support traffic engineering. The above allows establishing paths using constraints based routing algorithms such as CSPF without the need to use IPv6 over MPLS. Future works may be related to the behavior of PSA-TE6 in mobile environment related to the handover time and traffic engineering support. Other future works could be focused on the evaluation of our PSA-TE6 proposal with other strategies or in combination with them like Segment Routing or LISP.

6. References

1. Rosen E, Viswanathan A, Callon R. Multiprotocol label switching architecture. IETF RFC3031; 2001. p. 1–61.
2. Deering S, Hinden R. Internet protocol. Version 6 (IPv6) Specification. IETF RFC2460; 1998. p. 1–39.
3. Dhamdhere A, Luckie M, Huffaker B. Measuring the deployment of IPv6: Topology, Routing and Performance. Internet Measurement Conference (IMC); 2012. p. 537–50.
4. Czyz J, Allman M, Zhang J, Iekel-Johnson S, Osterweil E, Bailey M. Measuring IPv6 adoption. Proceedings of the 2014 ACM Conference on SIGCOMM. 2014; 44(4):87–98.
5. Hu Q, Carpenter B. Survey of proposed use cases for the IPv6 flow label. IETF RFC6294; 2011. p. 1–18.
6. Becerra LY, Padilla JJ. Review of approaches for the use of the label flow of IPv6 header. IEEE Latin America Transactions. 2014; 12(8):1602–7. https://doi.org/10.1109/TLA.2014.7014534.
7. Hinden R. Simple internet protocol plus white paper. IETF RFC1710; 1994.
8. Deering S, Hinden R. Internet protocol, Version 6 (IPv6) Specification. IETF RFC1883; 1995.
9. Metzler J, Hauth S. An end-to-end usage of the IPv6 flow label. Work Program; 2000. p. 1–5. PMid: 17016272.
10. Conta A, Carpenter B. A proposal for the IPv6 flow label specification. IETF Internet-Draft; 2001.
11. Conta A, Rajahalme J. A model for diffserv use of the IPv6 flow label specification. IETF Internet-Draft; 2001.
12. Hagino J. Socket API for IPv6 flow label field. IETF Internet-Draft; 2001.
13. Rajahalme J, Conta A, Carpenter B, Deering S. IPv6 flow label specification. IETF RFC3697; 2004.
14. Amante S, Carpenter B, Jiang S. Rationale for update to the IPv6 flow label specification. IETF RFC6436; 2011.
15. Amante S, Carpenter B, Jiang S, Rajahalme J. Ipv6 flow label specification. IETF RFC6437; 2011.
16. Chakravorty S. Challenges of IPv6 flow label implementation. Proceedings IEEE MILCOM2008; 2008.
17. Chakravorty S, Bush J, Bound J. IPv6 label switching architecture. Work Program; 2008.
18. Beckman M. IPv6 dynamic Flow Label Switching (FLS). IETF Internet-Draft; 2007.
19. Nichols K, Blake S, Baker F, Black D. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. IETF RFC2474; 1998.
20. Roberts L, Harford J. In-band QoS signaling for IPv6. Work Program; 2005.
21. Chin-Ling C. A study of IPv6 labeling forwarding model supporting diffserv. Procedia Engineering. Elsevier. 2011; 15:5590–4. https://doi.org/10.1016/j.proeng.2011.08.1037
22. Banerjee R, Malhotra S, MM. A modified specification for the use of the IPv6 flow label for providing an efficient Quality of Service using a hybrid approach. Working Group; 2002. p. 1–25.
23. Lin C, Tseng P, Hwang W. End-to-end QoS provisioning by flow label in IPv6. ICIS; 2006.
24. Lee I, Kim S. A QoS improvement scheme for real-time traffic using IPv6 flow labels. International Conference on Computational Science and its Applications; 2004. p. 278–85.
25. Prakash B. Using the 20 bit flow label field in the IPv6 header to indicate desirable Quality of Service on the internet. University of Colorado in partial fulfillment of the requirement for the degree of Master of Science; 2004.

26. Aazam M, Syed AM, Eui-Nam H. Redefining flow label in IPv6 and MPLS headers for End to End QoS in virtual networking for thin client. 3 19th Asia-Pacific Conference on Communications (APCC); 2013. p. 585–90.
27. Hassan R, Jabbar R. End-to-End (e2e) Quality of Service (QoS) for IPv6 video streaming. 19th International Conference on Advanced Communication Technology (ICACT) IEEE; 2017. p. 1–4.
28. Glennan T, Leckie C, Erfani SM. Improved classification of known and unknown network traffic flows using semi-supervised machine learning. Australasian Conference on Information Security and Privacy; 2016. p. 493–501. https://doi.org/10.1007/978-3-319-40367-0_33.
29. Yin A, Zhang S. Design and implementation of trusted routing strategy based on IPv6 flow identification. 10th International Conference on Communications and Networking in China (ChinaCom); 2015. p. 887–92. PMid: 25466433.
30. Padilla J, Paradells J. Intserv6: An approach to support QoS over IPv6 wired and wireless networks. European Transactions on Telecommunications. 2007; 19(6):1–19.
31. Padilla JJ, Paradells J, Rodriguez A. Supporting QoS over IPv6 wireless networks with IntServ6. IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications; 2006. p. 1–6.
32. Doan H. Flow-based forwarding scheme and performance analysis in mobile IPv6 networks. 8th International Conference Advanced Communication Technology. 2006; 3:1490–6.
33. Zheng T, Wang L, Daqing G. A flow label based QoS scheme for End-to-End mobile services. ICNS 2012: The Eighth International Conference on Networking and Services; 2012. p. 169–74.
34. Stephane O, Samuel P. HPMRSVP-TE: A hierarchical proxy mobile resource reservation protocol for traffic engineering. IEEE Vehicular Technology Conference; 2006. p. 1–5.
35. Tai WY, Eng Tan Ch, Ping Lau S. Towards utilizing flow label IPv6 in implicit source routing for Dynamic Source Routing (DSR) in Wireless Ad Hoc Network. IEEE Symposium on Computers and Informatics (ISCI); 2012. p. 101–6. PMCid: PMC3363133.
36. Donley C, Erichsen K. Using the flow label with Dual-Stack Lite. Work Group; 2011.
37. Carpenter B, Amante S. Using the IPv6 flow label for equal cost multipath routing and link aggregation in tunnels. IETF Internet-Draft, Draft; 2011.
38. Hartmond F, Rouhi M, Scholz D. Detecting load balancers in the Internet. Proceedings of the Seminars Future Internet (FI) and Innovative Internet Technologies and Mobile Communication (IITM) Focal Topic: Advanced Persistent Threats; Munich, Germany. 2017. p. 17–23.
39. Braden R, Clark D, Shenke S. Integrated services in the internet architecture: An overview. IETF RFC1633; 1994.
40. Melnikov DA, Lavrukhin YN, Durakovsky AP, Gorbatov VS, Petrov VR. Access control mechanism based on entity authentication with IPv6 Header "flow label" field. 3rd International Conference on Future Internet of Things and Cloud; 2015. p. 158–64.
41. Blake S. Use of the IPv6 flow label as a transport-layer nonce to defend against off-path spoofing attacks. Work Group; 2009.
42. Hendriks L, Velan P, Schmidt R, de O., Boer P-T de, Pras A. flow-based detection of IPv6-specific network layer attacks. IFIP International Conference on Autonomous Infrastructure, Management and Security; 2017. p. 137–42.
43. Bobade S, Goudar R. Secure data communication using protocol steganography in IPv6. International Conference on Computing Communication Control and Automation. 2015. p. 275–9. https://doi.org/10.1109/ICCUBEA.2015.59
44. Becerra LY, Padilla JJ. Study of proposals for supporting internet traffic engineering. Entre Ciencia e Ingeniería. 2012; 6(11):53–76.
45. Fortz B, Thorup M. Internet traffic engineering by optimizing OSPF weights. Internet Traffic Engineering by Optimizing OSPF Weights; 2000. p. 519–28.
46. Fortz B, Rexford J, Thorup M. Traffic engineering with traditional IP routing protocols. IEEE Communications Magazine. 2002; 40(10):118–24. https://doi.org/10.1109/MCOM.2002.1039866.
47. Thorup M, Fortz B. Optimizing OSPF/IS-IS weights in a changing world. IEEE Journal on Selected Areas in Communications. 2002; 20(4):756–67. https://doi.org/10.1109/JSAC.2002.1003042.
48. Ericsson M, Resende M, Pardalos P. A genetic algorithm for the weight setting problem in OSPF routing. Journal of Combinatorial Optimization. 2002; 6(3):299–333. https://doi.org/10.1023/A:1014852026591.
49. Gojmerac I, Ziegler T, Ricciato F, Reichl P. Adaptive multipath routing for dynamic traffic engineering. IEEE GLOBECOM. 2003; 6:3058–62.
50. Wang J. Edge-based traffic engineering for OSPF networks. Computer Network. 2005; 48(4):605–25. https://doi.org/10.1016/j.comnet.2004.11.008.
51. Abrahamsson H, Bjorkman M. Robust traffic engineering using L-balanced weight-settings in OSPF/IS-IS. Sixth International Conference on Broadband Communications, Networks and Systems; 2009. p. 1–8. PMid: 19228145. https://doi.org/10.4108/ICST.BROADNETS2009.7184.
52. Xu K, Liu H, Liu J, Shen M. One more wight is enough: Toward the optimal traffic engineering with OSPF. 31st International Conference on Distributed Computing Systems; 2011. p. 836–46.

53. Xu K, Shen M, Liu H, Liu J, Li F, Li T. Achieving optimal traffic engineering using a generalized routing framework. IEEE Transactions on Parallel and Distributed Systems. 2016; 27(1):51–65. https://doi.org/10.1109/TPDS.2015.2392760.
54. Awduche D. Requirements for traffic engineering over MPLs. IETF RFC2702; 1999.
55. Daniel AO. MPLS and traffic engineering in IP Networks. IEEE Communications Magazine. 1999; 37(12):42–7. https://doi.org/10.1109/35.809383.
56. Karaman A. Constraint-based routing in traffic engineering. IEEE Communications Magazine; 2006. p. 49–54.
57. Nayak P, Murty GR. Survey on constrained based path selection QoS routing algorithms: MCP and MCOP problems. Journal of Information Systems and Communication. 2013; 4(1):1–6.
58. Farinacci D, Fuller V, Meyer D, Lewis D. The locator/ID Separation Protocol (LISP). IETF RFC6830; 2013.
59. Farinacci D, Kowal M, Lahiri P. LISP traffic engineering use-cases. Draft-Farinacci-lisp-te-11; 2016.
60. Saucez D, Donnet B, Iannone L, Bonaventure O. Interdomain traffic engineering in a locator/identifier separation context. IEEE Internet Network Management Workshop (INM); 2008. p. 1–6. https://doi.org/10.1109/INETMW.2008.4660330
61. Li K, Wang S, Wang X. Edge router selection and traffic engineering in LISP-capable networks. Journal of Communications and Networks. 2011; 13(6):612–20. https://doi.org/10.1109/JCN.2011.6157477.
62. Li K, Wang S, Xu S, Wang X. ERMAO: An enhanced intradomain traffic engineering approach in LISP-capable Networks. IEEE Global Telecommunications Conference - GLOBECOM 2011; 2011. p. 1–5.
63. Herrmann D, Turba M, Kuijper A, Schweizer I. Inbound interdomain traffic engineering with LISP. 39th Annual IEEE Conference on Local Computer Networks; 2014. p. 458–61. https://doi.org/10.1109/LCN.2014.6925816.
64. Jeong T, Liy J, Hyun J, Yoo J-H, Hong JW-K. Experience on the development of LISP-enabled services: An ISP perspective. Proceedings of the 1st IEEE Conference on Network Softwarization (NetSoft); 2015. p. 1–9. https://doi.org/10.1109/NETSOFT.2015.7116159.
65. Nguyen HDD, Secci S. LISP-EC: Enhancing LISP with Egress Control. IEEE Conference on Standards for Communications and Networking (CSCN); 2016. https://doi.org/10.1109/CSCN.2016.7785189.
66. Filsfils C, Previdi S, Decraene B, Litkowski S, Shakir R. Segment routing architecture. Draft-IETF-spring-segment-routing-11; 2017.
67. Filsfils C, Sivabalan S, Nanduri M, Lin S, Bogdanov A, Horneffer M. Segment routing policy for traffic engineering. Raft-filsfils-spring-segment-routing-policy-00; 2017.
68. Bhatia R, Hao F, Kodialam M, Lakshman TV. Optimized network traffic engineering using segment routing. IEEE Conference on Computer Communications (INFOCOM); 2015. p. 657–65. PMid: 25799108 PMCid: PMC4385177.
69. Rabah G, Olivier D, Samer L, Texier G. Label encoding algorithm for MPLS segment routing. IEEE 15th International Symposium on Network Computing and Applications (NCA); 2016. p. 113–7.
70. Salsano S, Siracusano G, Luca V, Luca D, Pier L. PSMR-Poor Man's Segment Routing, a minimalistic approach to segment routing and a traffic engineering use case. IEEE/IFIP Network Operations and Management Symposium; 2016. p. 598–604.
71. Moreno E, Beghelli A, Cugini F. Traffic engineering in segment routing networks. Computer Network. 2017; 114:23–31. https://doi.org/10.1016/j.comnet.2017.01.006.
72. Balbinot L, Andrade M, Tarouco L, Roesler V. IP next generation label switching. IEEE Workshop on IP Operations and Management; 2002. p. 21–5.
73. Ishiguro K, Manral V, Davey A, Lindem A. Traffic engineering extensions to OSPF Version 3. IETF RFC5329; 2008.
74. Awduche D. RSVP-TE: Extensions to RSVP for LSP Tunnels. IETF RFC3209; 2001.
75. Younis O, Fahmy S. Constraint-based routing in the Internet: Basic principles and recent research. IEEE Communications Surveys and Tutorials. 2003; 5(1):2–13. https://doi.org/10.1109/COMST.2003.5342226.
76. Becerra LY, Padilla JJ, Ba-ol JL. A survey on constraints-based routing algorithms: Objectives traffic engineering and Quality of Service. Entre Ciencia e Ingeniería. 2017; (21):112–22.
77. Medhi D, Ramasay K. Network routing: Algorithms, protocols and architectures. Morgan Kaufmann; 2007. p. 166–91. https://doi.org/10.1016/B978-012088588-6/50010-3.
78. Deering S, Hinden R. Internet protocol, Version 6 (IPv6) Specification. IETF RFC8200; 2017.
79. Pioro M, Medhi D. Routing, flow and capacity design in communication and computer networks. 1st Ed. Morgan Kaufmann; 2004. p. 1–800. https://doi.org/10.1016/B978-012557189-0/50002-0.