CONTRIBUCIÓN AL SOPORTE DE INGENIERÍA DE TRÁFICO EN REDES IPv6

Tesis Doctoral

MsC. Line Yasmin Becerra Sánchez

Director
Jhon Jairo Padilla Aguilar, PhD.

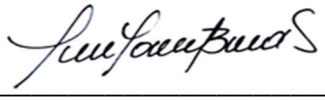
Universidad Pontificia Bolivariana
Escuela de Ingenierías
Doctorado en Ingeniería área Telecomunicaciones
Medellín
2019

Declaración de originalidad

Junio 5 de 2019

Line Yasmin Becerra Sánchez

"Declaro que este trabajo de grado no ha sido presentado con anterioridad para optar a un título, ya sea en igual forma o con variaciones, en ésta o en cualquiera otra universidad". Art. 92, parágrafo, Régimen Estudiantil de Formación Avanzada.

Firma 

RESUMEN

La ingeniería de tráfico de Internet se encarga del problema de evaluación y optimización del rendimiento de redes IP en operación. En la Internet actual, un problema importante por resolver es la congestión producida por el uso del enrutamiento del camino más corto, lo que hace que caminos comunes a muchas comunicaciones se saturen y otras rutas alternativas queden sub-utilizadas. Con la aparición de nuevas tecnologías tanto para redes fijas como móviles, la ingeniería de tráfico juega un rol importante y sigue siendo un tema de investigación abierto, permitiendo la creación de propuestas que analicen desde diferentes frentes los problemas existentes, con el fin de proporcionar soluciones y aportes que contribuyan a dar soporte de ingeniería de tráfico en Internet. Una tecnología muy aceptada hasta el momento para dar soporte de ingeniería de tráfico en Internet es MPLS, sin embargo, esta sugiere de una capa de transporte adicional, la capa 2.5.

El objetivo principal de esta tesis es proponer una solución para el soporte de ingeniería de tráfico en Internet para redes IPv6, que incorpore las bondades adquiridas con la migración del protocolo IPv4 a IPv6. Por tanto, se presenta una nueva propuesta para el uso de la Etiqueta de Flujo para conmutación de paquetes y el soporte de ingeniería de tráfico, la cual se ha denominado *Packet Switching Architecture to Support Traffic Engineering in IPv6 Networks*, (PSA-TE6). La meta de esta solución es utilizar el campo Etiqueta de Flujo IPv6 para conmutación de paquetes en redes IPv6 de una manera parecida a como trabaja MPLS, pero sin la necesidad de una arquitectura MPLS instalada.

Para lograr esto, se presenta el estudio de las diferentes especificaciones de la IETF en lo que respecta al campo Etiqueta de Flujo IPv6, ingeniería de tráfico, al protocolo IPv6, protocolos de enrutamiento y señalización que han sido extendidos para el soporte de ingeniería de tráfico. También se proporciona el estudio de propuestas de casos de uso de la Etiqueta de Flujo IPv6, del estado del arte de soluciones para soportar ingeniería de tráfico en Internet y de algoritmos de enrutamiento basados en restricciones. Finalmente, se presenta la evaluación de PSA-TE6, la cual se ha orientado a cinco aspectos principales: evaluación del espacio de etiquetas de flujo Ipv6 en PSA-TE6, evaluación del costo de apilamiento de etiquetas de flujo Ipv6, minimización del costo de apilamiento de etiquetas, evaluación del balanceo de carga y operación de PSA-TE6 en redes IPv6 móviles con IPv6 móvil jerárquico.

Los resultados muestran que PSA-TE6 permite el soporte de ingeniería de tráfico en redes IPv6 fijas y móviles. PSA-TE6 comparada con IP/MPLS presenta costos menores alrededor del 45% cuando no se usa apilamiento de etiquetas. Adicionalmente, PSA-TE6 y MPLS tienen costos similares en presencia del 40% al 45% de túneles de nivel superior a 1. Además, PSA-TE6 tiene el mismo comportamiento que MPLS en un escenario de balanceo de carga cuando el número de túneles es optimizado. Finalmente, la combinación de PSA-TE6 con IPv6 móvil jerárquico proporciona reducción en tiempos de conmutación, tiempos en cola y tiempos de transmisión de paquetes con respecto al protocolo IPv6 móvil jerárquico trabajando en su forma estándar.

Palabras Clave: Ingeniería de tráfico, Internet, IPv6, IPv6 Móvil, Etiqueta de flujo IPv6.

TABLA DE CONTENIDO

Pág.

LISTA DE FIGURAS

Pág.

LISTA DE TABLAS

Pág.

LISTA DE ACRÓNIMOS

6DER:	*PSA-TE6 Domain Edge Router*
6DLSP:	*PSA-TE6 Domain Label Switching Paths*
6DTR:	*PSA-TE6 Domain Transit Router*
I6LTN:	*Incoming Ipv6 Flow Label to Next Hop IPv6 Flow Label Forwarding Data*
CBR:	*Constraints Based Routing*
CN:	*Correspondent Node*
CoA:	*Care of Address*
CSPF:	*Constrained Shortest Path First*
DS:	*Differentiated Services*
EID:	*Endpoints Identifiers*
ETR:	*Egress Tunnel Router*
FEC:	*Forwarding Equivalence Class*
FIB:	*Forwarding Information Base*
HMIPv6:	*Hierarchical Mobile Internet Protocol version 6*
IETF:	*Internet Engineering Task Force*
ILP:	*Integer Linear Programming*
IPv6:	*Internet Protocol version 6*
IS-IS-TE:	*Intermediate System to Intermediate System- Traffic Engineering*
ITR:	*Ingress Tunnel Router*
FEC:	*Forwarding Equivalent Class*
FIB:	*Forwarding Information Base*
FTN6:	*Forwarding Equivalence Class to Next Hop IPv6 Flow Label Forwarding Data*
HA:	*Home Agent*
HMIPv6:	*Hierarchical Mobile IPv6*
LISP:	*Locator/ID Separation Protocol*
LSA:	*Link State Advertisement*
LSP:	*Label Switching Path*
MAP:	*Mobility Anchor Point*
MCP:	*Multi-Constrained Path*
MCOP:	*Multi-Constrained Optimal Path*
MIPv6:	*Mobile Internet Protocol version 6*
MN:	*Mobile Node*
MPLS:	*Multiprotocol Label Switching*
MPLS-TE:	*Multiprotocol Label Switching- Traffic Engineering*
N6FLD:	*Next Hop IPv6 Flow Label Data*
OSPF-TE:	*Open Shortest Path First-Traffic Engineering*
PSA-TE6:	*IPv6 Flow Label Switching Architecture*
QoS:	*Quality of Service*
RLOC:	*Routing Locator*
RSVP-TE:	*Resource Reservation Protocol-Traffic Engineering*
RFC:	*Request for Comments*
SR:	*Segment Routing*
TE:	*Traffic Engineering*
TLV:	*Type/Length/Value*
TTL:	*Time to Live*

LISTA DE PUBLICACIONES

Line Yasmin Becerra Sanchez, Jhon Jairo Padilla Aguilar, Josep Paradells, "Hmipv6-BI: A Proposal to Improve the Bandwidth of the Radio Channel in Hmipv6 Networks". *IEEE Latin America Transactions*, pp.603 - 609, 2011. DOI: 10.1109/TLA.2011.6030966.

Line Yasmin Becerra Sánchez, Jhon Jairo Padilla Aguilar, "Estudio de Propuestas para Soportar Ingeniería se Tráfico en Internet". *Entre Ciencia e Ingeniería*, Vol. 6, No. 11, pp.53 - 76, 2012.

Line Yasmin Becerra Sanchez, Jhon Jairo Padilla Aguilar, "Review of Approaches for the Use of the Flow Label of IPv6 Header". *IEEE Latin America Transactions, Vol. 12, No. 8, December 2014*.

Line Yasmin Becerra Sánchez, "Mininet: Una Herramienta Versátil para Emulación y Prototipado de Redes Definidas Por Software". *Entre Ciencia e Ingeniería,* vol.9, no.17, Jan/June 2015.

Line Yasmin Becerra Sánchez, Bryan Valencia, Santiago Santacruz, Jhon Jairo Padilla Aguilar. Using Mininet, OpenFlow 1.3 and RYU controller for an IPv6 Software Defined Network Emulation. *Revista Educación en Ingeniería*, 12(24), pp. 89-96, Julio, 2017. DOI:10.26507/rei.v12n24.794.

Line Yasmin Becerra Sánchez, Jorge Bañol, Jhon Jairo Padilla Aguilar, "Un estudio sobre algoritmos basados en restricciones: objetivos ingeniería de tráfico y calidad de servicio. *Entre Ciencia e Ingeniería* vol.11, no.21, Jan/June 2017.DOI: http:10.31908/19098367.3288.

Line Yasmin Becerra Sánchez, Jhon Jairo Padilla Aguilar, "Analysis of Load Balancing for a New Approach to Support Traffic Engineering in IPv6 Networks" *Indian Journal of Science and Technology*, Vol 11(35), DOI: 10.17485/ijst/2018/v11i35/122033, September 2018.

Line Yasmin Becerra Sánchez, Jhon Jairo Padilla Aguilar, "An Approach to support traffic engineering in IPv6 networks based on IPv6 facilities. *Telecommunication Systems Modelling, Analysis, Design and Management,* Springer Link, ISSN: 1018-4864 (Print) 1572-9451 (Online). DOI: 10.1007/s11235-018-00543-7.

.

1. INTRODUCCIÓN

La ingeniería de tráfico de Internet se encarga del problema de evaluación y optimización del rendimiento de redes IP en operación [1]. Un problema importante en la Internet actual es el uso del enrutamiento del camino más corto, lo que trae como consecuencia la congestión de ciertos caminos comunes a muchas comunicaciones y que otras rutas alternativas permanezcan sub-utilizadas. Con la aparición de las diferentes tecnologías de redes fijas y móviles, la ingeniería de tráfico ha jugado un rol importante y ha un sido tema de investigación abierto hasta el momento, dando como resultado un número importante de propuestas que analizan desde diferentes frentes los problemas existentes, con el fin de dar soluciones y aportes que contribuyan a dar soporte de ingeniería de tráfico de Internet. Una solución a este problema consiste en usar tecnologías de conmutación que permitan hacer ingeniería de tráfico. Una de las más usadas es MPLS, sin embargo, esta sugiere de una capa de transporte adicional, la capa 2.5.

Por otro lado, el protocolo IPv6 [2] ofrece grandes ventajas en comparación a IPv4 [3] con respecto al aumento de direcciones y a la provisión de calidad de servicio, movilidad, entre otras. Un campo nuevo en la cabecera del protocolo IPv6, es el campo de "Etiqueta de Flujo". En la IETF se presentaron varios debates acerca del propósito de este campo, desde su creación. Las preguntas que se hacían los diseñadores eran: ¿Iba a ser para manejar conmutación rápida, o iba a ser significativa para aplicaciones y usada para especificar calidad de servicio?, ¿debe ser esta enviada por el host emisor, o podría ser fijada por los routers?,¿Podría ser modificada en ruta o debería ser entregada sin cambio? A causa de estas incertidumbres, y debido a que había trabajo más urgente, la Etiqueta de Flujo IPv6 ha estado siendo ignorada, por lo que está siendo establecida a cero en casi cada paquete IPv6 [4]. Debido a esto, varias propuestas se han hecho para usar el campo de Etiqueta de Flujo IPv6 con diferentes propósitos como: para el soporte de calidad de servicio, movilidad, identificación de túnel IPv4-in-IPv6, balanceo de carga, seguridad y conmutación de paquetes [5], [6].

En esta tesis se presenta una solución para el soporte de ingeniería de tráfico en Internet incorporando las ventajas protocolo IPv6, lo que constituye una propuesta robusta que beneficia y mejora el rendimiento de las redes y servicios de nueva generación. Su enfoque principal está en el uso de la Etiqueta de Flujo para la conmutación de paquetes y el soporte de ingeniería de tráfico, la cual se ha denominado PSA-TE6 (*Packet Switching Architecture to Support Traffic Engineering in IPv6 Networks*).

En PSA-TE6 se aprovecha la ubicación del campo Etiqueta de Flujo en los primeros 64 bits de la cabecera IPv6, al igual que los campos DS (*Differenciated Services*) y Límite de Saltos

(*Hop Limit*), siendo los dos últimos mapeados por MPLS cuando trabaja sobre IP (IP/MPLS). Esta propuesta nace como consecuencia del estudio realizado al campo Etiqueta de Flujo desde su creación hasta las últimas recomendaciones de la IETF, además del análisis de las diferentes propuestas de uso del campo Etiqueta de Flujo, ya realizado y también de la observación de su estructura, la cual muestra una gran similitud con la etiqueta MPLS en lo que respecta al tamaño (20 bits) y su contenido. La meta con PSA-TE6 es facilitar la provisión de ingeniería de tráfico a nivel IP, en redes, tecnologías y servicios de nueva generación.

1.1 MOTIVACIÓN

La motivación principal para el desarrollo de esta tesis fue dar un aporte al problema de congestión en Internet. Aunque hasta el momento varias propuestas han sido publicadas desde varios enfoques, es un tema que sigue abierto y permite proponer nuevas soluciones. El propósito de esta tesis es proporcionar una solución para soportar ingeniería de tráfico en redes IPv6, teniendo en cuenta las ventajas que trae la migración del protocolo IPv4 a IPv6, las recomendaciones y especificaciones realizadas por la IETF en cuanto a protocolos de enrutamiento y señalización para el soporte de ingeniería de tráfico en Internet.

Igualmente, otra motivación importante fue evidenciar entre los cambios de la cabecera IPv6, el campo Etiqueta de Flujo y sus características similares a la etiqueta MPLS, lo cual hace que este campo sea un candidato potencial para desarrollar propuestas de conmutación por etiquetas y la provisión de ingeniería de tráfico a nivel IP, en redes IPv6. El estudio de este campo desde su creación hasta las especificaciones actuales junto con las recomendaciones de la IETF para el soporte de Ingeniería de tráfico permitió el paso al desarrollo de la propuesta presentada en esta tesis.

1.2 OBJETIVOS DE LA TESIS

1.2.1 Objetivo General

Diseñar una solución para el problema de soporte de Ingeniería de tráfico en redes fijas y móviles que trabajen bajo el protocolo IPv6.

1.2.2 Objetivos Específicos

– Evaluar las propuestas existentes que buscan dar soporte de ingeniería de tráfico en redes IP.

- Diseñar una propuesta que permita dar soporte de ingeniería de tráfico en redes IPv6 fija y móvil. Esta propuesta podría reutilizar soluciones existentes haciendo adaptaciones o también podría ser totalmente nueva.
- Evaluar la propuesta resultante, de tal manera que se puedan comparar los resultados con aproximaciones existentes de ingeniería de tráfico basada en MPLS e ingeniería de tráfico basada en IP convencional.

1.3 APORTES ORIGINALES DE LA TESIS DOCTORAL

En esta tesis se obtiene una solución para el soporte de Ingeniería de Tráfico en redes IPv6 fijas y móviles. La solución propuesta se denomina PSA-TE6 (*Packet Switching Architecture to Support Traffic Engineering in IPv6 Networks*), es una aproximación para la utilización del campo Etiqueta de Flujo IPv6, en la conmutación de paquetes en redes IPv6 tanto fijas como móviles. Los aportes específicos son:

- Esta propuesta aprovecha las ventajas del nuevo protocolo de Internet IPv6 y el campo Etiqueta de Flujo incluido en su cabecera, para control de conmutación, denominada PSA-TE6.
- Propone dos métodos para el apilamiento de etiquetas IPv6 con el fin resolver el problema del espacio reducido de etiquetas.
- Se obtiene un modelo para realizar balanceo de carga con PSA-TE6.
- Se obtiene un modelo que permite evaluar los costos de envío cuando el apilamiento de etiquetas está habilitado, y para cuando está funcionando sin apilamiento de etiquetas.
- Propone un método de operación de la propuesta PSA-TE6 con redes IPv6 móviles mediante el protocolo IPv6 móvil jerárquico.

1.4 METODOLOGÍA EMPLEADA

Para cumplir con los objetivos planteados se desarrollaron una serie de etapas que se describen a continuación.

1.4.1 Estudio del estado del arte

En esta etapa se realiza una revisión de los antecedentes desde varios enfoques, los cuales son pertinentes para un conocimiento general del entorno de la problemática a tratar. Un primer enfoque se concentró en el estudio de las especificaciones dadas por la IETF. Luego las diferentes soluciones propuestas por investigadores alrededor del tema. Con el estudio de las publicaciones de la IETF, se hace revisión de las RFCs referentes a especificaciones y recomendaciones para hacer Ingeniería de Tráfico en Internet, protocolos de enrutamiento

y de señalización extendidos para soportar ingeniería de tráfico y arquitecturas existentes como MPLS. Igualmente, las especificaciones para el protocolo IPv6 y las extensiones para movilidad como IPv6 móvil e IPv6 móvil jerárquico. Fue muy importante revisar las especificaciones de la Etiqueta de Flujo IPv6, su historia y evolución, ya que este campo de la cabecera IPv6 es fundamental en la solución propuesta en esta tesis.

También, se hace revisión de soluciones relacionadas con el soporte de ingeniería de tráfico en Internet, algoritmos de enrutamiento basados en restricciones, redes móviles IP y el estudio de casos de uso de la Etiqueta de Flujo IPv6.

1.4.2 Pasantía

Se realiza una pasantía en Estados Unidos en la Ciudad de Kansas City, en la Universidad de Missouri Kansas City con la supervisión del PhD. Deepankar Medhi, durante 5 meses. Durante el tiempo de la pasantía se tuvo la oportunidad de asistir el curso de postgrado en "Network Routing" orientado por el doctor Medhi. Otra actividad fue la revisión y el estudio de una propuesta de ingeniería de tráfico en LISP (*Locator/ID Separation Protocol*) con la asesoría del Dr. Medhi, este era un tema en el cual, él estaba trabajando y que está relacionado con la ingeniería de tráfico en Internet. También se estudió y practicó con la herramienta de optimización AMPL, con la realización de pruebas del modelo de programación lineal para balanceo de carga propuesto por el Dr Medhi, para ingeniería de tráfico con LISP.

1.4.3 Diseño de la propuesta

En base a las actividades anteriores y de acuerdo con la información recopilada y analizada, se diseña la solución para el soporte de ingeniería de tráfico en Internet para redes IPv6, se establecen los principios de diseño y su arquitectura general proyectándose el aporte para la ingeniería de tráfico en redes IPv6 fijas como móviles.

1.4.4 Evaluación de la Solución propuesta

En esta etapa, se determinaron los principales aspectos a evaluar, la metodología y herramientas para la evaluación. Para la evaluación de cada uno de estos aspectos, se analizó los procesos involucrados y se desarrollaron los modelos correspondientes. Estos modelos arrojaron resultados los cuales fueron analizados y permitieron obtener conclusiones importantes que evidenciaron el comportamiento de la solución propuesta. Los resultados de cada modelo evaluado, junto con el proceso y metodología de evaluación fueron el insumo para la elaboración de las diferentes publicaciones de resultados.

1.4.5 Publicaciones

Durante todo el proceso de tesis se elaboraron algunas publicaciones de tipo revisión en revistas nacionales e internacionales, cuyo insumo fue el estudio realizado del estado del arte. Igualmente, con los resultados obtenidos también se realizan publicaciones internacionales.

1.4.6 Desarrollo del informe final y presentación de la Tesis

En esta etapa se realiza el informe final de la tesis que recopila los aspectos más relevantes de la solución propuesta.

1.5 ESTRUCTURA DE LA TESIS DOCTORAL

La estructura de esta tesis doctoral está organizada como se describe a continuación. En el capítulo 2, se hace la descripción del estado del arte, con respecto a las especificaciones de la IETF relacionadas al protocolo IPv6, IPv6 móvil e IPv6 móvil Jerárquico y el campo Etiqueta de Flujo IPv6; también con respecto a ingeniería de tráfico, protocolos de enrutamiento y de señalización. Igualmente, soluciones propuestas por investigadores para el soporte de ingeniería de tráfico, algoritmos basados en restricciones y casos de uso del campo Etiqueta de Flujo para varios propósitos. En el capítulo 3, se describe el diseño de la solución propuesta PSA-TE6, una arquitectura para soportar ingeniería de tráfico en Internet mediante el uso de la Etiqueta de Flujo IPv6 para la conmutación de paquetes. Se presentan los antecedentes, los principios de diseño para redes fijas y móviles IP, y se describe su funcionamiento general. En el capítulo 4, se presenta la evaluación de PSA-TE6 con respecto evaluación del costo de apilamiento de etiquetas de flujo IPv6, minimización del costo de apilamiento de etiquetas, evaluación del balanceo de carga y operación de PSA-TE6 en redes IPv6 móviles con IPv6 móvil jerárquico. En el capítulo 5, se dan las conclusiones. Y en el capítulo 6, se dan algunas recomendaciones para trabajos futuros productos de esta tesis.

2. ESTADO DEL ARTE

En este capítulo se presenta la descripción algunos conceptos relacionados con el tema de la tesis que fueron necesarios e importantes para su desarrollo. También se proporciona un resumen de los trabajos relacionados con ingeniería de tráfico en Internet, algoritmos basados en restricciones, redes móviles IP y el campo Etiqueta de Flujo IPv6.

2.1 IPv6

En los años 90s la IETF (*Internet Engineering Task Force*), comenzó sus esfuerzos por desarrollar un sucesor del protocolo para IPv4 [3], con el fin de resolver la limitación prevista de espacio de direcciones y también para proporcionar funcionalidades adicionales. En 1994 el grupo encargado de la IETF, recomendó la creación de IPv6, cuya especificación fue descrita en la RFC1752 [7] denominada "*The Recommendation for the IP Next Generation Protocol*". Luego esta recomendación fue evolucionando hasta la publicación de la especificación completa en 1998 con la RFC2460 [8] (RFC8200 [2], actualmente). Varios cambios surgieron con el paso de IPv4 a IPv6, entre ellos se pueden destacar [9].

El tamaño de direcciones para IPv6 fue incrementado a 128 bits, esto resuelve el problema del espacio de direcciones limitado de IPv4 y ofrece una jerarquía de direccionamiento más profunda y de configuración más simple.

La cabecera IPv6 tiene a una longitud de 40 bytes. Algunos campos de la cabecera IPv4 fueron removidos o se dejaron opcionales. De esta forma, los paquetes pueden ser manejados rápidamente y con un costo de procesamiento más bajo.

Con IPv4 las opciones están integradas dentro de la cabecera básica de IPv4. Con IPv6 las opciones son manejadas como "cabeceras de extensión". Las cabeceras de extensión son opcionales y son insertadas entre la cabecera IPv6 y la carga útil del paquete. De esta forma el paquete IPv6 se puede construir de manera flexible y ágil.

Los paquetes pertenecientes al mismo flujo de tráfico que requieren un manejo especial, como por ejemplo un cierto nivel de calidad de servicio, pueden ser etiquetados por el emisor con un identificador especial.

2.1.1 Formato de la Cabecera IPv4 Vs IPv6

Como se puede apreciar en la figura 1 y 2, la mayoría de los campos en la cabecera IPv6 fueron cambiados o modificados, el único campo que permanece sin cambio es el campo llamado "*Version*", esto se hizo para permitir a IPv4 e IPv6 coexistir en el mismo enlace local.

Este campo "Version" es un valor de 4 bits, y para IPv4 debe ser igual a 4, mientras que para IPv6 debe tener el valor de 6.

El campo "*Header length*" de IPv4 es irrelevante para IPv6 porque todas las cabeceras IPv6 son de la misma longitud; IPv4 requiere este campo porque sus cabeceras pueden ser tan cortas como de 20 bytes y tan largas como de 60 bytes para acomodar las opciones IP.

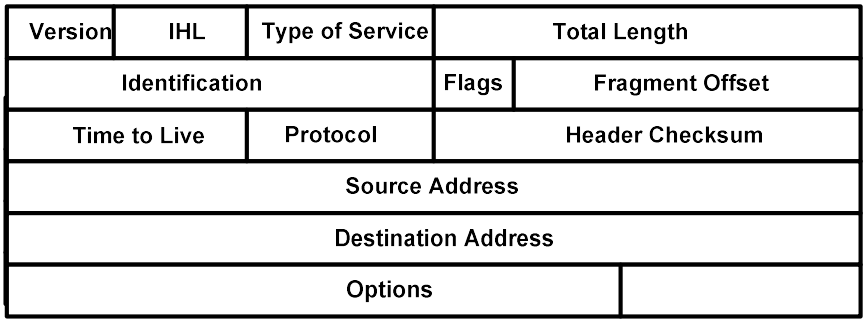


Figura 1. Cabecera IPv4 RFC791 [3].

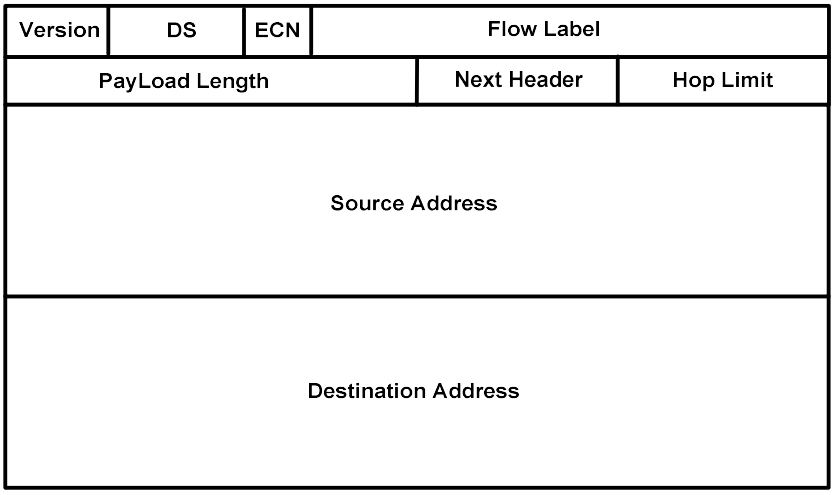


Figura 2. Cabecera IPv6 RFC8200 [2] y RFC2474 [10].

El campo "*DS*" en IPv6, consiste de 6 bits que son usados para especificar el tipo de tratamiento que un paquete debería recibir por los routers para proporcionar una calidad de servicio (QoS) apropiada. La RFC2474 [10], denominada "*Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*", hace la especificación de este campo para ambos protocolos IPv4 e IPv6. Esta especificación requiere la sustitución del campo "*ToS*" de IPv4 y el campo "*Traffic class*" de IPv6 por el campo "DS".

El campo "*payload length*" en IPv6, es el campo "*Total Length*" de IPv4. El campo "*Total lenght*" de IPv4 específica la longitud del datagrama entero, incluyendo las cabeceras IP; de esta manera los routers pueden calcular la longitud de carga útil del datagrama IPv4 mediante la substracción de la longitud de la cabecera de la longitud del datagrama. En IPv6 este cálculo es innecesario, porque la longitud de carga útil de IPv6 incluye las cabeceras de extensión.

El campo "*Hop Limit*" en IPv6 es el campo "*Time-to-Live (TTL)"* de IPv4. TTL fue originalmente concebido como límite superior del tiempo de vida (dado en segundos) de un paquete en Internet, sin embargo, se decidió interpretar como el número de saltos máximo que puede dar un paquete debido a que debe ser asignado un valor de tiempo de vida finito y entero que sea fácil de medir en diferentes routers.

El campo "*Protocol*" de IPv4, se involucró en IPv6 dentro del campo "*Next Header*", donde este especifica la próxima cabecera, ya sea una cabecera de extensión IPv6 u otra cabecera de protocolos pertenecientes a la capa superior.

Los campos "*Source/destination address*", pasa de 32 bits para paquetes IPv4 a 128 bits para paquetes IPv6.

Adicionalmente, se agregaron dos campos a IPv6, el campo "*ECN*" y el campo "*Flow Label*". El campo "*ECN*" es de dos bits y es usado como banderas de notificación de congestión explícita. Por su parte, el campo "*Flow Label*" es un valor usado para identificar paquetes pertenecientes al mismo flujo. La Etiqueta de Flujo y la dirección del nodo fuente podrían servir para identificar los flujos. Este campo fue originalmente establecido a 24 bits en la RFC1883 [11], pero cuando el campo "DS" fue incrementado a 8 bits, el campo "*Flow Label*" fue decrementado a 20 bits para compensar.

Por otro lado, Las opciones de IPv6 existen como cabeceras separadas después de la cabecera principal pero antes de la carga útil del paquete.

Algunos campos de la cabecera IPv4 desaparecieron, por ejemplo, como los protocolos de capa superior como TCP y UDP calculan sus propias sumas de comprobación sobre las cabeceras, la suma de comprobación de IPv4 fue considerada redundante y por tanto, desapareció en la cabecera IPv6. Otros campos que desaparecieron fueron los que tienen que ver con fragmentación. El campo "*Identification*" es usado para identificar un datagrama como parte de un paquete fuente fragmentado. Esto es útil solamente en instancias donde la fragmentación del paquete es permitida, como en IPv4. En IPv6 este campo no es necesario porque IPv6 no permite fragmentación de nodo intermedio. Los campos "*Flags*" y "*Fragment offset*" son también usados para permitir la fragmentación, por lo que tampoco se necesitan en IPv6.

2.2 INGENIERÍA DE TRÁFICO EN INTERNET

De acuerdo con la especificación de la IETF RFC3272 [1], la Ingeniería de Tráfico (*Traffic Engineering, TE*) de Internet se encarga del problema de optimización y evaluación del rendimiento de redes IP en operación, el objetivo es mejorar el rendimiento de la red, optimizando el uso de los recursos y el tráfico mediante la aplicación de tecnologías y principios científicos que permitan la medición, caracterización, modelado y control del tráfico de Internet.

La IETF ha publicado especificaciones de trabajos relacionados con el soporte de ingeniería de tráfico en redes IP, las cuales se describen en la sección 2.2.1. Por otro lado, también existen aproximaciones resultantes de trabajos de investigación en el tema, los cuales se categorizan y se presentan en secciones posteriores.

2.2.1 Especificaciones de la IETF Relacionados con Ingeniería de Tráfico de Internet

A continuación, se describen las especificaciones de la IETF más relevantes relacionadas con la ingeniería de tráfico de Internet y redes IP.

2.2.1.1 Multiprotocol Label Switching – MPLS

MPLS extiende el modelo de enrutamiento de Internet mejorando el envío de paquetes y control del camino. La especificación está descrita en la RFC3031 [12]. La arquitectura de MPLS está conformada por: LSRs (*Label Switching Routers*), son nodos de la red con capacidad para conmutar paquetes por medio de etiquetas. LERs (*Label Edge Routers*), son nodos o routers que se encuentran en la frontera de la red, sirven para controlar el ingreso y el egreso del tráfico de los usuarios. LSP (*Label Switching Path*), es el camino que siguen los paquetes, este camino se define en términos de la transición de etiquetas.

En el ingreso a un dominio MPLS, los routers de conmutación de etiquetas, *LERs,* clasifican paquetes IP en clases de equivalencia de envío FECs (*Forwarding Equivalence Classes*) basados en una variedad de factores, incluyendo una combinación de la información llevada en la cabecera IP de los paquetes y la información de enrutamiento local mantenida por los LSRs. Una etiqueta MPLS es luego antepuesta a cada paquete de acuerdo con su FEC. Un LSR examina la etiqueta y usa esta información para tomar decisiones de envío del paquete. Un LSR toma decisiones de envío mediante el uso de la etiqueta ubicada en los paquetes como el índice dentro de una entrada de la NHLFE (*Next Hop Label Forwarding Entry*). El paquete es luego procesado como está especificado en la NHLFE. La etiqueta entrante puede ser remplazada por una etiqueta saliente, y el paquete puede ser conmutado al próximo LSR.

Antes de que un paquete deje un dominio MPLS, su etiqueta MPLS debe ser removida. Un LSP es el camino entre un LSR de ingreso y uno de egreso, el cual es atravesado por paquetes etiquetados.

El camino de un LSP explícito está definido desde el nodo de ingreso origen del LSP. MPLS puede usar un protocolo de señalización tal como RSVP-TE [13] o LDP [14] para establecer LSPs. Los requerimientos para ingeniería de tráfico sobre MPLS están descritos en la RFC2702 [15]. También, extensiones para soportar instanciación de LSPS explícitos son discutidas en la RFC3209 [13]. Finalmente, extensiones para LDP, variante conocida como CR-LDP, para soportar LSPs explícitos son descritas en la RFC3212 [16].

2.2.1.2 RSVP-TE (Protocolo de reserva de recursos extendido para Ingeniería de Tráfico)

RSVP-TE es la extensión al protocolo RSVP [17] para el soporte de ingeniería de tráfico y está especificado mediante la RFC3209 [13]. Fue desarrollado para realizar el proceso de establecimiento de caminos conmutados por etiquetas, LSPs (*Label Switching Paths*), para el soporte de ingeniería de tráfico en combinación con MPLS. RSVP-TE hace la distribución de etiquetas de RSVP-TE basado en una aproximación downstream por demanda. Una diferencia clave entre RSVP-TE y RSVP es que RSVP-TE se utiliza para la señalización entre los enrutadores para establecer flujos de LSPs, a diferencia del RSVP original, que se usa entre los hosts para establecer microflujos. Por tanto, RSVP-TE no tiene el problema de escalabilidad que enfrenta RSVP, en cuanto a la gestión de microflujos. RSVP-TE se utiliza para establecer LSPs unicast direccionales, mientras que RSVP permite establecimiento de flujos multicast [18].

RSVP usa diferentes tipos de mensajes, estos mensajes están compuestos de diferentes entidades estandarizadas llamadas objetos. Cada objeto lleva información estandarizada específica, como la dirección IP del nodo destino llevado en el objeto *Sesion*. Dos mensajes fundamentales están definidos en RSVP, el mensaje *Resv* que una solicitud de reserva que viaja desde el receptor al emisor, y un mensaje *Path* que viaja downstream desde el host emisor a lo largo de rutas unicast/multicast hacia el receptor. El mensaje *Path* sigue la ruta provista por el protocolo de enrutamiento, usando el mismo camino como el flujo de datos. Cada mensaje *Path* almacena un estado de camino en todos los nodos intermediarios a lo largo del camino. El estado del camino incluye información recuperada del mensaje *Path*, o de otros procesos específicos de ese nodo. El mensaje *Resv* es enviado exactamente sobre el camino inverso usado previamente por el mensaje *Path*. El mensaje *Path* es enviado usando la misma dirección fuente y destino que los datos. En respuesta, el mensaje *Resv* es enviado salto por salto usando información de estado del camino almacenado en los nodos intermediarios. Los mensajes *Resv* y *Path* son usados para crear el estado del camino, pero

también para realizar refrescos del estado del camino y de reserva en los nodos. En esta forma, si una ruta cambia, debido a actualizaciones de enrutamiento o falla de nodo, el nuevo mensaje *Path* creará un nuevo estado de *Path* en los nodos de la nueva ruta y el mensaje *Resv* establecerá estado de reserva en esos nodos para la nueva ruta [19].

En total 7 tipos de mensajes son definidos en la RFC2205 [17]. Estos son: *Path, Resv, PathTear, ResvTear, PathErr, ResvErr y ResConf*. También 15 diferentes clases de objetos son introducidos en la misma RFC, cada objeto consiste en palabras múltiplos de 32 bits comenzando con la cabecera común, la cual especifica la longitud en bytes del objeto, un campo *Class-Num* y un campo *C-type*. El valor *Class-Num* es únicamente usado para identificar una clase de objetos. En total 15 valores de *Class-Num* son especificados en la RFC2205 [17]. Dentro de una clase el campo *C-type* identifica un tipo de objeto preciso. Estos tipos y sus valores son únicos dentro de un *Class-Num*.

Por su parte, RSVP-TE soporta la creación de LSPs enrutados explícitamente con o sin reserva de recursos. Un LSP creado por RSVP puede ser usado para llevar una agregación de flujos de tráfico de la misma clase. Debido a que los flujos de tráfico a lo largo de un LSP son identificados por las etiquetas MPLS asociadas con este, tal camino puede ser tratado como un túnel. Los LSPs que son usados en esta forma se conocen como túneles LSPs.

Para soportar la característica de establecimiento de túneles LSPs, se definieron los nuevos objetos *Rsvp Sesion, Sender Template y Filter Spec*. Además, cinco nuevos objetos fueron introducidos: *Label Request, Label, Explicit Route, Record Route y el Sesion Attribute*. El objeto *Label Request* indica que un enlace de etiqueta es solicitado, pero también provee información acerca del protocolo de capa de red usado sobre caminos específicos. El objeto *Label* transmite la etiqueta asociada con el tráfico saliente para un túnel LSP específico. El objeto *Explicit Route* lleva la ruta que ha de ser seguida por un LSP como una secuencia de nodos. El *Record Route* es usado para informar al nodo emisor acerca a de la ruta actual que un túnel LSP atraviesa. El objeto *Sesion Attribute* es usado para ayudar en la identificación de sesión y diagnóstico [13].

Para crear un túnel LSP, el nodo frontera en la red MPLS genera un mensaje *Path* de RSVP con el nuevo objeto *Sesion* definido y el objeto *Label Request* incluido. Si este nodo tiene información del camino acerca del cumplimiento de los requerimientos de QoS del túnel, o satisface algunos criterios de políticas, un objeto *Explicit Route* es introducido en el mensaje *Path* también. Este objeto puede ser cambiado si una ruta mejor es encontrada más tarde, creando en esta forma la posibilidad para reenrutar la sesión en una forma dinámica. También un *Record Route* y un objeto *Sesion Attribute* pueden ser opcionalmente insertados en el mensaje *Path*. El nodo destino responde al objeto *Label Request*, mediante la inclusión en el mensaje *Resv* de un objeto *Label*. Como en el caso de RSVP original, el mensaje *Resv* es

enviado de regreso *upstream* siguiendo el estado de camino creado por el mensaje *Path*. Cada nodo a lo largo del camino que recibe un mensaje *Resv* que contiene un objeto de *Label*, usará esa etiqueta para el tráfico de salida asociado con ese túnel LSP (ver figura 3). Si el nodo receptor del mensaje *Resv* no es el emisor, una nueva etiqueta es localizada y ubicada en el objeto *Label* del mensaje *Resv*, el cual es enviado upstream. Los nuevos objetos definidos son opcionales con respecto a RSVP. Sin embargo, los objetos *Label Request* y *Label* son obligatorios para extensión de ingeniería de tráfico RSVP-TE. EL objeto *Label* contiene solamente una única etiqueta codificada en 4 octetos. El nodo downstream es el responsable para seleccionar una etiqueta para el flujo. Si una etiqueta no está disponible entonces el nodo envía un mensaje *Path Err* indicando la falla en la ubicación de la etiqueta. El objeto *Label* es almacenado en el bloque de estado de reserva, y el nodo debería estar preparado para enviar paquetes llevando la etiqueta asignada [13].

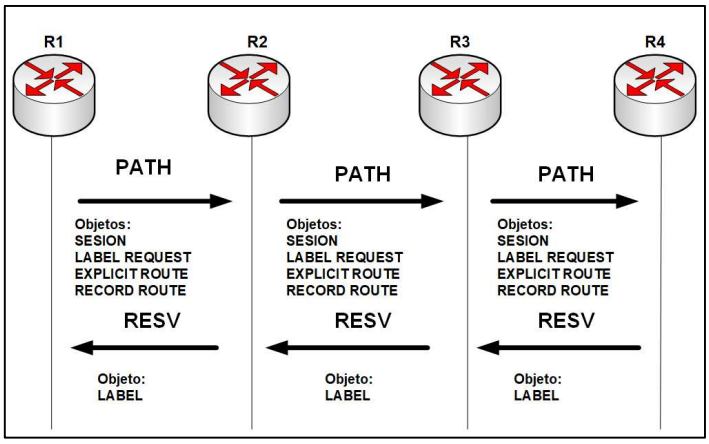


Figura 3. Creación de túneles LSPs con RSVP-TE [13].

El nodo *upstream* usa la etiqueta desde el objeto *Label* como la etiqueta asociada con la interfaz de salida para ese emisor. También, una nueva etiqueta es localizada en el mismo nodo y es enlazada a la interfaz entrante para esta sesión/emisor. La interfaz es la misma como la usada para enviar mensajes *Resv* a previos saltos. El objeto *Label Request* fue definido en el contexto de tres posibles *C-Types*. El primer tipo es llamado *Label Request* sin *Label Range* y es empleado para indicar el protocolo de capa 3 que usa el camino. Los otros dos tipos son específicos para las tecnologías ATM y Frame Relay. El objeto *Label Request* es almacenado en cada nodo a lo largo del camino en el *Path State Block*. Cada nodo que acepta un objeto *Label Request* debe incluir un *Label* en el mensaje *Resv* que responda a ese mensaje *Path*. Cada nodo que envía un objeto *Label Request* debe estar listo a recibir y manejar el objeto *Label* en el mensaje *Resv* resultante.

2.2.1.3 OSPF-TE (Protocolo OSPF extendido para Ingeniería de tráfico)

La versión de OSPF para IPv4 es conocida como OSPFv2, está definida en la RFC2328 [20] y la versión para IPv6 está especificada en RFC5340 [21]. OSPF es una instancia de un protocolo de estado de enlace que se basa en la comunicación de información de enrutamiento salto por salto. Está diseñado para proveer enrutamiento intradominio en una red IP.

OSPF tiene la funcionalidad para dividir una red intradominio (un sistema autónomo) en subdominios, comúnmente referidos como "*áreas*". Cada red intradominio debe tener un área núcleo, referido como un área "*backbone*" la cual es identificada con ID=0. OSPF permite una instalación jerárquica con el área *backbone* como el nivel superior mientras que otras áreas, conectadas al área del *backbone* son referidas como áreas de bajo nivel, lo que significa que el área *backbone* es la encargada de resumir la topología de una área a otra área y viceversa (ver Figura 4) [18].

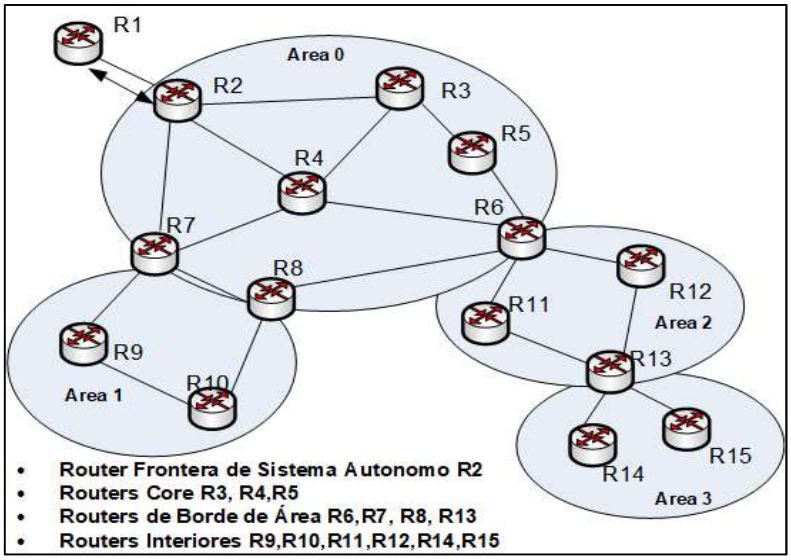


Figura 4. Áreas OSPF [18].

De acuerdo con la funcionalidad de OSPF de dividir una red en áreas, los routers son clasificados en cuatro tipos diferentes:

Routers frontera. Son los que están ubicados entre el *área backbone* y las áreas de nivel bajo, tal como se muestra en la Figura 4.

*Routers Internos. S*on los routers en cada área de nivel bajo que tienen interfases solamente hacia otros *routers internos* en la misma área.

Routers backbone. Están localizados en el *área 0*, con al menos una interfaz a otros *routers backbone*, (ver figura 4). Los routers frontera también pueden ser considerados como *routers backbone*.

Routers de frontera de Sistema autónomo. Están ubicados en el área 0 con conectividad a otro sistema autónomo; estos deben ser capaces de manejar más de un protocolo de enrutamiento. OSPF está diseñado para direccionar cinco tipos diferentes de redes: Redes punto a punto, redes broadcast, redes multiacceso no-broadcast, redes punto a multipunto y enlaces virtuales.

OSPF usa funcionalidades in-Network (información llevada en la misma red como tráfico de usuario) para inundar información de enrutamiento tal como mensajes de anuncio de estado de enlace LSA (*Link State Advertisement*). Múltiples LSAs pueden ser combinados en un paquete de actualización de estado de enlace OSPF. La Inundación es la técnica utilizada para propagar paquetes de actualización de estado de enlace, como también para paquetes LSA [18].

OSPF-TE es la extensión de OSPF para Ingeniería de tráfico, la recomendación para trabajar con el protocolo IPv4 se encuentra en la RFC3630 [22] (OSPFv2-TE, para IPv4). Estas extensiones proporcionan una descripción de la topología de ingeniería de tráfico (incluyendo ancho de banda y restricciones administrativas) y la distribución de esta información dentro de un área OSPF. Por su parte, la RFC5329 [23] describe las recomendaciones para OSPFv3 para soportar ingeniería de tráfico intra-área en IPv6, con base en definiciones de nuevos TLVs (*Type/Length/Value*) y sub-TLVs para extender las capacidades en redes IPv6. Muchas de las extensiones especificadas en los anteriores documentos son respuesta a los requerimientos enunciados en la RFC2702 [15], y así están referidos como "extensiones de ingeniería de tráfico" y son también comúnmente asociadas con ingeniería de tráfico MPLS.

En OSPF, un LSA contiene un TLV de nivel superior. Hay dos TLVs de nivel superior definidos: de dirección de router y de enlace. El TLV de dirección de router especifica una dirección IP estable del anuncio de router que está siempre alcanzable, es típicamente implementada como una "*dirección loopback*". El TLV de enlace, describe un solo enlace y es construido de un conjunto de sub-TLVs. No hay requerimientos de ordenamiento para los sub-TLVs. Solamente un TLV de enlace será llevado en cada LSA. El TLV de enlace es tipo 2, y la longitud es variable. Los siguientes sub-TLVs del TLV de enlace son definidos y están especificados y descritos en más detalle en la RFC3630 [22].

1 – Tipo de enlace (1 octeto).

2 - ID de enlace (4 octetos).

3 – Dirección IP de interfaz Local (4N octetos).

4 – Dirección IP de Interfaz Remota (4 octetos).

5 - Métrica de Ingeniería de Tráfico (4 octetos).

6 – Ancho de banda máximo (4 octetos).

7 – Máximo Ancho de banda reservable (4 octetos).

8 – Ancho de banda no reservado (32 octetos).

9 – Grupo Administrativo (4 octetos).

2.3 TRABAJOS RELACIONADOS CON INGENIERÍA DE TRÁFICO EN INTERNET

En las dos últimas décadas se han realizado varias propuestas producto de investigación alrededor de la ingeniería de tráfico cuyo objetivo ha sido dar aportes que contribuyan a mejorar el rendimiento en las redes IP en operación. Con el estudio de estas propuestas se pudo detectar que se pueden categorizar en cinco líneas de trabajo, que son las más relevantes para esta tesis, las cuales son: las soluciones basadas en IP, las que están basadas en MPLS, las que hacen una combinación o híbrido entre MPLS e IP, las basadas en arquitecturas nuevas como LISP y enrutamiento por segmentos (ver figura 5). A continuación, se presenta el conjunto de propuestas que se han realizado en cada una de estas categorías mostrando los aportes proporcionados en cada una de ellas.

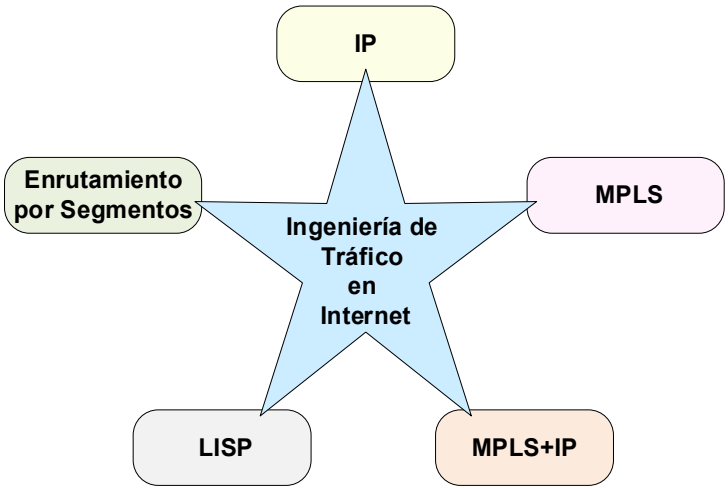


Figura 5. Categorías de propuestas de trabajo en Ingeniería de Tráfico

2.3.1 Propuestas de Ingeniería de tráfico basadas en IP

Una de las primeras propuestas para ingeniería de tráfico intradominio basada en IP fue expuesta por Fortz en [24] y [25]. En estas propuestas los autores proponen un modelo de enrutamiento para hacer ingeniería de tráfico mediante los protocolos IGPs como OSPF e IS-IS y hacen comparaciones con el modelo de enrutamiento óptimo. En [24], se propone la optimización del establecimiento de pesos mediante la solución de un problema de programación lineal y proponen un método heurístico denominado HeurOSPF, del cual concluyen que los pesos obtenidos pueden ser utilizados "universalmente", ya que estos no son ajustados a una función de costo particular y demuestran que los resultados obtenidos con HeurOSPF están muy cercanos a los resultados de la optimización óptima. También exponen que un algoritmo de ajuste de peso inteligente para el enrutamiento OSPF es una poderosa herramienta para aumentar la capacidad de una red para satisfacer el aumento de demandas de tráfico y para hacer ingeniería de tráfico. En [25], los mismos autores describen una aproximación para ingeniería de tráfico intradominio en redes IP mediante la monitorización del tráfico y la topología, y realizan la optimización de pesos de los enlaces; los autores concluyen que los protocolos tradicionales de enrutamiento son efectivos para ingeniería de tráfico en redes grandes.

Se han propuesto algunas herramientas que facilitan la ejecución de procesos de ingeniería de tráfico, tales como las descritas en [26], [27] y [28]. En [26], los autores hacen la descripción de una herramienta denominada NetScope de Laboratorios AT&T, la cual comprende un conjunto de herramientas de software unificadas para gestionar el rendimiento en redes backbone IP. Con esta herramienta se puede generar una visión global de la red mediante la configuración y uso de datos asociados a los elementos individuales de red, para luego poder inferir y visualizar las implicaciones de cambios locales de red amplia en tráfico, configuración y control, la cual es útil para los proveedores de red para experimentar con cambios en la configuración de red en un ambiente simulado, en lugar de hacerlo en una red en operación. Los autores demostraron las capacidades de la herramienta mediante un ejemplo de ingeniería de tráfico con un enlace fuertemente cargado, identificando las demandas de flujo de tráfico y los cambios de configuración de enrutamiento intradominio que reducen la congestión.

En [27], los autores presentan un estudio de interconexión IP desde el punto de vista de un proveedor de servicio IP (ISP) y describen las clases de errores que pueden aparecer dentro y a través de archivos de configuración de routers. Luego presentan un caso de estudio de una herramienta prototipo que desarrollaron en colaboración con AT&T para chequear la configuración del backbone IP de AT&T y proveer entradas a otros sistemas

como NetScope para visualización de la red y hacer ingeniería de tráfico, también mostró cómo verificar la configuración real de la red con respecto a su consistencia.

En [28], los autores presentan el diseño e implementación de una herramienta de software denominada YATES (*Yet Another Traffic Engineering*) para experimentar con diferentes aproximaciones de ingeniería de tráfico. YATES permite hacer prototipado y evaluación de sistemas de ingeniería de tráfico, incluye herramientas para modelar topologías, esquemas de enrutamiento, demandas y algoritmos para predicción de fallas. YATES incluye un simulador de red que calcula la congestión, el rendimiento, pérdidas, la latencia, etc., y una implementación basada en SDN que se puede usar para validar los resultados obtenidos mediante la simulación. Los autores también exponen las limitaciones que tiene YATES, entre las cuales mencionan que la herramienta no está diseñada para razonar sobre el comportamiento de paquetes específicos en la red. Por tanto, YATES no es ideal para realizar micro-evaluaciones comparativas como evaluar algoritmos de control de congestión de TCP o estudiar los efectos de la ordenación de paquetes o retrasos en la cola.

Para el establecimiento de pesos de enlace se han propuesto algunos algoritmos en [29]–[32]. En [29], los autores presentan un algoritmo genético denominado GAOSPF, para resolver los problemas de establecimientos de pesos de OSPF, que buscan encontrar un conjunto de pesos que optimice el rendimiento de la red. El algoritmo fue probado sobre varias redes y se comparan los resultados obtenidos con aproximaciones heurísticas y también con el límite más bajo obtenido mediante programación lineal. En [30], se planteó un algoritmo para ingeniería de tráfico dinámica dentro de sistemas autónomos denominado AMP (*Adaptative Multi-Path*). El algoritmo AMP fue simulado en NS-2 y los autores lo compararon con estrategias de enrutamiento estándar en un escenario de simulación real. En [31], se hizo una propuesta denominada Método de Ingeniería de Tráfico de K-conjuntos, para desarrollar ingeniería de tráfico en redes OSPF mediante la partición de tráfico desigual en k-conjuntos, el cual se hace solo en redes de borde. Los autores proponen también un algoritmo heurístico y lo evalúan usando simulación. En [32], los autores presentan un método de búsqueda heurística para encontrar el establecimiento de *pesos balanceados-L;* este método le permite a los operadores aplicar reglas para controlar la máxima utilización de enlace y controlar la cantidad de capacidad disponible necesaria para manejar variaciones de tráfico repentinas.

Por otro lado, en [33], los autores propusieron una aproximación para desarrollar ingeniería de tráfico en *backbones* (núcleos de red) ejecutando el enrutamiento nativo IP sobre una topología física, en lugar de una topología virtual, controlando la distribución de flujos de tráfico a través del establecimiento de pesos de enlace apropiados para enrutamiento del camino más corto. La solución presentada es denominada “propuesta

integrada", la cual permite ingeniería de tráfico sin la superposición de malla completa, en contraposición a la aproximación de superposición (Overlay).

También se han presentado estudios que pretenden organizar conceptos teóricos respecto a los problemas de ingeniería de tráfico en redes IP, tal como lo descrito en [34], donde los autores formulan los requerimientos básicos de una arquitectura de ingeniería de tráfico práctica construida sobre OSPF y presentan un marco teórico reuniendo estos requerimientos prácticos. Por otra parte, en [35], se establece la teoría fundamental y algoritmos de representatividad de caminos más cortos, y se concluye que en general es mucho más difícil la tarea de calcular caminos representables de caminos más cortos que calcular los pesos de los enlaces para tales caminos. En [36], se presenta el estudio del problema de cálculo de la fiabilidad de una red en operación, usando el protocolo OSPF donde los enlaces fallan con probabilidades independientes dadas. El autor desarrolló métodos de aproximación basados en trabajos relacionados para redes conmutadas por circuitos, mostrando resultados que concluyen la robustez de los pesos OSPF optimizados en comparación con el establecimiento de pesos usado tradicionalmente.

De otra parte, en [37] se presenta un estudio de la estabilidad de enrutamiento en Internet como un todo, usando los cambios observados en el número de rutas sobre cada enlace de AS-AS (Sistema Autónomo-Sistema Autónomo) como una métrica y la medida de tales cambios desde múltiples monitores sobre un periodo de un año. Los resultados muestran que muy pocos eventos de enrutamiento afectan a la totalidad Internet, y que esos eventos se debieron al anuncio de nuevos prefijos ya sea en forma de fugas de ruta o espacio de direcciones de desagregación. Además, el análisis de las rutas de AS muestra que pocos enlaces tienen un gran peso y esos enlaces son generalmente más bajos en la jerarquía del nivel de la topología (por ejemplo, un gran ISP regional que se conecta a los ISP globales). Los resultados de medición durante un período de un año muestran que, si bien los cambios de enrutamiento ocurren con frecuencia, su alcance suele ser limitado a una pequeña porción de la Internet global, lo que puede atribuirse a que la naturaleza de la topología de los ASs es cada vez más flexible.

En [38], los autores estudian una metodología para ingeniería de tráfico basado en tráfico multidimensional origen-destino (OD), utilizando el análisis de componentes principales como una técnica para la reducción de la dimensionalidad, y un modelo lineal local como una técnica para el análisis de tendencias. Los resultados validados con datos sobre una red real presentan un margen satisfactorio de error por su adopción en situaciones prácticas.

En [39], se muestran los resultados de la exploración al problema de alcanzar la ingeniería de tráfico óptima en redes IP intradominio, modelando la ingeniería de tráfico óptima como la maximización de la utilidad de flujos multi-commodity. Teóricamente se muestra que unos

conjuntos dados de rutas óptimas corresponden a una función objetivo particular que puede ser convertida a caminos más cortos con respecto a un conjunto de enlaces positivos, los cuales pueden ser directamente configurados sobre protocolos basados en OSPF.

Finalmente, en [40], el autor hace contribuciones en lo que respecta a ingeniería de tráfico proactivo en redes IP. Estas contribuciones están referidas a problemas relacionados con ingeniería de tráfico proactivo en redes IP backbone grandes. También se hacen contribuciones de investigación sobre técnicas de estimación de matrices de tráfico, desarrollo de técnicas de optimización de enrutamiento, implementación de técnicas de optimización en redes MPLS y estudio de procedimientos de establecimiento de pesos para enrutamiento de estado de enlace. Todas estas contribuciones se encuentran resumidas en [41], [42], [43].

Todas las propuestas de ingeniería de tráfico basada en IP enunciadas anteriormente corresponden a soluciones intradominio; en lo que se refiere a ingeniería de tráfico interdominio las aproximaciones planteadas se basan en el protocolo BGP (*Border Gateway Protocol*) [44]. La TE (*Traffic Engineering*, ingeniería de tráfico) interdominio es desarrollada tomando en cuenta la información de enrutamiento anunciada por dominios adyacentes. El cambio de configuración de TE en un dominio podría afectar las decisiones de enrutamiento de otros ASs (*Autonomous Systems*, Sistemas Autónomos) cercanos y esto se puede propagarse en cascada. Esto por lo general introduce problemas de inestabilidad de ruta a través de Internet donde un solo camino interdominio puede tomar mucho tiempo para converger [45], [46]. Por tanto, la TE interdominio debe tomar en consideración cómo preservar su predictibilidad y la estabilidad para asegurar una distribución estable al igual que la convergencia de enrutamiento [47]. De acuerdo a lo anterior, algunas investigaciones han propuesto directrices para realizar ingeniería de tráfico interdominio como en [46], [48].

Existen varias propuestas para realizar ingeniería de tráfico interdominio, a continuación, se mencionan tres propuestas las cuales presentan metodologías y técnicas para realizar TE globalmente. En [49], los autores describen varias técnicas que son usadas para controlar flujo de paquetes en la Internet global, también explican cómo es configurado BGP para de ingeniería de tráfico interdominio. Por otra parte, en [50], propusieron una herramienta denominada " Tweak-it" de fuente abierta para ingeniería de tráfico interdominio en redes ISP (*Internet Service Provider*) grandes, la cual consiste de un heurístico de múltiples objetivos acoplado con un simulador BGP. Y en [51], los autores presentan el protocolo ITER (Interdomain Traffic Engineering Routing), como primer paso hacia un diseño sistemático de un protocolo escalable y eficiente de enrutamiento para ingeniería de tráfico interdominio. En la tabla 1 se proporciona un resumen de las propuestas basadas en IP.

Tabla 1. Propuestas para el soporte de ingeniería de tráfico basadas en IP

Referencia	Propuesta	Intra/Inter dominio
[24],[25]	Optimización del establecimiento de pesos- Algoritmo HeurOspf	Intradominio
[26]	Herramienta que facilita la ejecución de procesos de ingeniería de tráfico denominada NestScope	Intradominio
[27]	Herramienta prototipo para chequear la configuración del backbone IP de AT&T para hacer ingeniería de tráfico	Intradominio
[28]	YATES: Una herramienta de software para prototipado de aproximaciones de ingeniería de tráfico.	Intradominio
[29]	Algoritmo Genético GAOSPF para establecimiento de pesos	Intradominio
[30]	Algoritmo para ingeniería de tráfico dinámica AMP (Adaptative Multi-Path)	Intradominio
[31]	Método de Ingeniería de Tráfico de K-conjuntos	Intradominio
[32]	Heurística para encontrar el establecimiento de pesos balanceados-L	Intradominio
[33]	Propuesta Integrada sin superposición de malla	Intradominio
[34][35][36]	Conceptos teóricos respecto a problemas de ingeniería de tráfico IP	Intradominio
[37]	Estudio de estabilidad del enrutamiento en Internet	Intra e Inter dominio
[38]	Metodología para ingeniería de tráfico basados en tráfico multidimensional origen-destino (OD)	Intradominio
[39]	Modelado de ingeniería de tráfico óptima como la maximización de la utilización de flujos multi-commodity.	Intradominio
[40][41][42][43]	Contribuciones de investigación sobre técnicas de estimación de matrices de tráfico, técnicas de optimización de enrutamiento, estudio de procedimientos de establecimiento de pesos.	Interdomain
[46][48]	Directrices para realizar ingeniería de tráfico interdominio	Interdomain
[49]	Descripción de técnicas usadas para controlar flujo de paquetes en la Internet global.	Interdomain
[50]	Herramienta "Tweak-it" de fuente abierta para ingeniería de tráfico interdominio en redes ISP grandes.	Interdomain
[51]	Protocolo ITER (Interdomain Traffic Engineering Routing)	Interdomain

Existen varias propuestas para realizar ingeniería de tráfico interdominio, a continuación, se mencionan tres propuestas las cuales presentan metodologías y técnicas para realizar TE globalmente. En [49], los autores describen varias técnicas que son usadas para controlar flujo de paquetes en la Internet global, también explican cómo es configurado BGP para de ingeniería de tráfico interdominio. Por otra parte, en [50], propusieron una herramienta denominada " Tweak-it" de fuente abierta para ingeniería de tráfico interdominio en redes ISP (*Internet Service Provider*) grandes, la cual consiste de un heurístico de múltiples objetivos acoplado con un simulador BGP. Y en [51], los autores presentan el protocolo ITER (Interdomain Traffic Engineering Routing), como primer paso hacia un diseño sistemático de un protocolo escalable y eficiente de enrutamiento para ingeniería de tráfico interdominio. En la tabla 1 se proporciona un resumen de las propuestas basadas en IP.

2.3.2 Propuestas de Ingeniería de tráfico basadas en MPLS.

Una de las primeras propuestas de ingeniería de tráfico basada en MPLS fue expuesta en [52]. En el artículo el autor presenta la aplicación de MPLS para hacer ingeniería de tráfico en redes IP enfocándose especialmente en redes de proveedor de servicios. Primero hace una conceptualización sobre ingeniería de tráfico, los desafíos para redes IP, el modelo del proceso de ingeniería de tráfico y los objetivos. También describe el proceso de ingeniería de tráfico con el modelo Overlay clásico y finalmente explica el modelo de ingeniería de tráfico de MPLS.

El modelo expuesto en [52], consiste de cuatro componentes funcionales: gestión del camino, asignación de tráfico, diseminación de información de estado de red y gestión de red. La gestión del camino comprende todos los aspectos relacionados a la selección de rutas explícitas, la instanciación y establecimiento de túneles LSP. Este componente consta de tres funciones principales que son: Selección del camino, ubicación del camino y mantenimiento del camino. La primera función, la selección del camino, especifica la ruta explícita para un túnel LSP en el nodo origen del túnel. El enrutamiento basado en restricciones [53] es usado para calcular rutas que satisfacen un conjunto de requerimientos, un subconjunto de restricciones impuestas por la red y políticas administrativas. La segunda función, la ubicación del camino, la cual es usada para instanciar túneles LSP (*Label Switching Path*) usando un protocolo de señalización como RSVP [17] y un protocolo de distribución de etiquetas LDP (*Label Distribution Protocol*) [54]. La tercera función es el mantenimiento del camino, el cual mantiene y termina túneles LSPs ya establecidos. La asignación de tráfico se refiere a todos los aspectos relacionados con la ubicación de tráfico para establecer túneles LSP. En la diseminación de información de estado de red, conciernen a la distribución de información relevante de estado de la topología a través del dominio MPLS, esto es llevado a cabo mediante las extensiones de IGPs. Finalmente, la gestión de red incluye un conjunto de funciones para la gestión de configuración, rendimiento, cálculos y fallas.

En otro estudio en [55], se describen los problemas generales de diseño de un sistema MPLS para ingeniería de tráfico, y los inconvenientes que se presentan con los protocolos IGPS. Luego presentan el diseño de un sistema MPLS para una red nacional y basado sobre las experiencias proponen un procedimiento genérico, para el despliegue de un sistema MPLS. Los autores argumentan que es factible la implementación de un sistema de ingeniería de tráfico MPLS para una red ISP grande, además sugieren que en MPLS, el enrutamiento basado en restricciones, CBR y algunas extensiones de IGPs tienen que ser desarrollados en todos los routers que participan del sistema MPLS. Los autores concluyen que un sistema MPLS es muy útil para hacer ingeniería de tráfico, porque se alcanza el comportamiento

deseado del tráfico y evita significativamente el trabajo de administradores de red de la manipulación de métricas de enrutamiento.

Con el desarrollo de servicios diferenciados (Diffserv) [56], se presentaron algunas propuestas de ingeniería de tráfico basadas en MPLS-Diffserv como aparece en [57] y [58]. En [57], los autores propusieron un marco general para aprovisionamiento de QoS (Quality of Service) intradominio a través de TE basada en MPLS en redes Diffserv, y en [58] se propuso una solución de TE en la cual utilizaron las técnicas de optimización con *Simulated Annealing* y método de proyección gradiente [59]. En [60], los autores hacen una discusión de la integración Diffserv-MPLS para alcanzar QoS (*Quality of Service*) e ingeniería de tráfico en redes MPLS con el fin de mejorar la eficiencia del backbone. Además, realizan una simulación con el fin de evaluar que tanto la ingeniería de tráfico y la calidad de servicio con MPLS pueden mejorar el desempeño de las redes actuales.

Un gestor de ingeniería de tráfico automatizado (TEAM, *Traffic Engineering Automated Manager*) para redes MPLS/Diffserv es propuesto en [61]. Este es un gestor adaptativo que provee la calidad de servicio requerida para los usuarios y reduce la congestión en la red. TEAM está compuesta de una herramienta de ingeniería de tráfico TET (*Traffic Engineering Tool*), la cual maneja las rutas y el ancho de banda adaptativamente en la red, una herramienta de medición y evaluación de desempeño MPET (*Measurement/Performance Evaluation Tool*), la cual mide parámetros importantes en la red y una herramienta de simulación ST (*Simulation Tool*) que puede ser usada por (TET) para consolidar sus decisiones. Los autores concluyen que TEAM es un eficiente gestor de red en condiciones diferentes y no predecibles de tráfico a expensas de un incremento limitado en complejidad computacional y costo.

Otra propuesta basada en Diffserv, donde se propone una nueva política de preferencia de LSPs está descrita en [62], conocida como V-Prept. La propuesta intenta evitar el reenrutamiento de LSPs. De forma similar al esquema TEAM, la optimización para preferencia de LSPs considera múltiples criterios, incluyendo prioridad de LSPs, el número de LSPs, y el ancho de banda preferido.

Teniendo en cuenta que una de las desventajas de MPLS es la baja escalabilidad, en algunos artículos han propuesto modelos alternativos para aliviar los problemas de escalabilidad en construcción y mantenimiento de LSPs, tales como el modelo embudo (multipunto a punto, MP2P) en [63]–[65] y el modelo manguera punto a multipunto P2MP en [66].

Con el objetivo de reducir el número total de LSPs y de etiquetas necesarias, en [63] se propone un esquema de TE usando múltiples LSPs Multipunto a punto MP2P; esta aproximación consiste de dos procedimientos distintos: Construcción de LSP MP2P y

asignación de flujo. El problema de asignación de flujo es formulado como un problema de programación entera mezclada, en el que la carga de enlace máxima es minimizada. Los autores hacen comparaciones numéricas con la aproximación convencional de LSP punto a punto y muestran que la aproximación MP2P puede reducir el número de etiquetas y LSPs requeridos. De igual manera hacen comparaciones numéricas entre el esquema de asignación de flujo el esquema convencional de SPF (*Shortest Path Fast*) basado en asignación de flujo y concluyen que el esquema de asignación de flujo propuesto reduce la carga de enlace máxima.

En [64], la solución MP2P es usada para TE con garantías de QoS extremo a extremo determinística, además se incluyen dos algoritmos de control de admisión a nivel de paquete. Por otra parte en [65], la ingeniería de tráfico de MP2P fue también estudiada con el fin de dar solución a los problemas de escalabilidad.

De acuerdo a [45], se clasifica la ingeniería de tráfico basada en MPLS en línea en dos categorías: en ajuste dinámico de la tasa de división de tráfico entre LSPs estáticos previamente construidos y en el cálculo dinámico sobre la marcha de cada nueva demanda de troncales de tráfico.

En la primera categoría está la propuesta de [67], en la cual, los autores describen un mecanismo de ingeniería de tráfico adaptativo multi-camino para redes MPLS, llamado MATE (*MPLS Adaptive Traffic Engineering*). El objetivo de MATE es evitar la congestión de red balanceando adaptativamente la carga entre múltiples caminos, basados en la medición y análisis de congestión de camino. La aproximación presenta un modelo analítico de balanceo de carga, genera una clase de algoritmos MATE asincrónicos y prueba su estabilidad y optimalidad mediante simulación. Los resultados indican que MATE elimina los desbalances de tráfico que pueden ocurrir entre múltiples LSPs. Los autores concluyen que las altas tasas de pérdidas de paquetes pueden ser reducidas significativamente mediante el desplazamiento apropiado de porciones de tráfico a LSPs menos cargados.

En la segunda categoría están [68]–[73]. En [68], se propuso un software denominado servidor de ingeniería de tráfico y enrutamiento RATES (*Routing and Traffic Engineering Server*), desarrollado para ingeniería de tráfico MPLS. En [69], los autores proponen un algoritmo de enrutamiento online dinámico, para construcción de caminos de ancho de banda garantizado en redes MPLS, denominado DORA (*Dynamic Online Routing Algorithm*). El objetivo es ubicar caminos con ancho de banda reservado uniformemente a través de la red para permitir que más caminos futuros sean aceptados dentro de la red y para balancear carga. La clave de operación de DORA es primero evitar enrutamiento sobre enlaces que tengan una alta probabilidad a ser parte de otro camino y segundo tener bajo ancho de banda residual disponible. La evaluación del algoritmo DORA la realizan los autores mediante

NS-2 ("Network Simulator NS-2,"), haciendo comparación con otras propuestas y dando como resultado que, durante fallas de enlace, DORA requiere que menos caminos sean reenrutados y sea menos costoso computacionalmente.

Tabla 2. Propuestas para el soporte de ingeniería de tráfico basadas en MPLS

Referencia	Propuesta
[52]	Describe el proceso de ingeniería de tráfico con el modelo Overlay clásico y finalmente explica el modelo de ingeniería de tráfico de MPLS.
[55]	Descripción de los problemas generales de diseño de ingeniería de tráfico MPLS y los inconvenientes que se presentan con los protocolos IGPS.
[57][58][60]	Ingeniería de tráfico basadas en MPLS-Diffserv.
[61]	Gestor de ingeniería de tráfico automatizado (TEAM, Traffic Engineering Automated Manager) para redes MPLS/Diffserv.
[62]	Ingeniería de tráfico basada en MPLS/Diffserv, donde se propone una nueva política de preferencia de LSPs, denominada V-Prept.
[63][64][65]	Modelos para aliviar problemas de escalabilidad en MPLS-modelo Multipunto a punto MP2P.
[66]	Modelos para aliviar problemas de escalabilidad en MPLS-modelo punto a Multipunto P2MP.
[67]	Descripción de un mecanismo de ingeniería de tráfico adaptativo multi-camino para redes MPLS, llamado MATE (MPLS Adaptive Traffic Engineering).
[68]	Software denominado servidor de ingeniería de tráfico y enrutamiento RATES (Routing and Traffic Engineering Server) para ingeniería de tráfico MPLS.
[69]	Algoritmo de enrutamiento online dinámico, para construcción de caminos de ancho de banda garantizado en redes MPLS, denominado DORA (Dynamic Online Routing Algorithm).
[70][71]	Problema de interferencia de LSPs.
[73]	Enrutamiento en línea para ingeniería de tráfico MPLS.
[74]	Solución óptima para balanceo de carga dinámico (ODLB).
[75]	Aproximación llamada MPLS-PHS que adapta el concepto de la técnica de supresión de cabecera de carga útil (PHS, Supression Header Payload).

El problema de interferencia de LSPs son enfocados en [70] y [71]. La interferencia se presenta por la competencia de los LSPs sobre los enlaces críticos que no tienen suficiente ancho de banda disponible para soportar todas las demandas de LSP. Los autores expresan que debido al establecimiento directo de LSPs (usando CSPF, *Constrained Shortest Path First*) sin consideración de la ubicación de los nodos de ingreso/egreso para troncos de tráfico entrantes, se presenta congestión en algunos enlaces críticos que usan LSPs múltiples. En [73], se considera un esquema de supervivencia en redes MPLS con ingeniería de tráfico en línea, el cual construye LSPs dinámicamente mediante la aplicación del algoritmo de camino más corto a la métrica de peso de enlace, que refleja el requerimiento de TE específico.

Por otro lado, en [74], los autores proponen una solución óptima para balanceo de carga dinámico (ODLB), donde los resultados son comparados con otros algoritmos y se evidencia un mejoramiento en cuanto a uso de ancho de banda y reenrutamiento de tráfico saliente. ODLB es garantizado para proveer la solución de enrutamiento óptimo a las redes con rutas paralelas. Y por último en [75], se propone una aproximación llamada MPLS-PHS que adapta el concepto de la técnica de supresión de cabecera de carga útil (PHS, *Supression Header Payload*) para ser aplicable en LSPs de un dominio MPLS, en el cual se encuentra conexiones

multipunto a multipunto, los resultados descritos por los autores muestran un incremento en el rendimiento de datos para flujos IPv6 de tiempo real. En la tabla 2 se da un resumen de las propuestas basadas en MPLS.

2.3.3 Propuestas de Ingeniería de tráfico basadas en metodologías híbridas MPLS+IP.

A continuación, se describen las propuestas encontradas que utilizan metodologías híbridas de ingeniería de tráfico basadas en MPLS+IP. Por ejemplo, en [76] se presenta un método de enrutamiento híbrido (HMR, Hybrid Routing Method) para ingeniería de tráfico de una red IP, el cual enruta un número limitado de flujos de tráfico usando MPLS y enruta el tráfico restante usando protocolos de enrutamiento IGPs (tales como OSPF e IS-IS). El método HMR fue utilizado para proponer una herramienta de ingeniería de tráfico que permite a los operadores visualizar y manejar tráfico para evitar congestión, también como decidir dónde ubicar routers y túneles MPLS.

Por otro lado, en [77] se expone un modelo de ingeniería de tráfico en línea, el cual usa una aproximación de enrutamiento híbrida IGP+MPLS que combina optimización de la ruta y selección del camino con el fin de alcanzar enrutamiento eficiente de flujos en redes IP. El modelo de optimización de la ruta combina confiabilidad y optimalidad para enrutar menos flujos bajo falla de enlace y rechazar menos flujos bajo condiciones de carga pesada. El enrutamiento IGP es alcanzado mediante establecimiento de la métrica de enlace OSPF inversamente proporcional a la capacidad del enlace.

Una técnica heurística escalable para ingeniería de tráfico IGP+MPLS se presenta en [78], para optimizar las métricas de enlace. El método está basado sobre el metaheurístico de templanza simulada (*Simulated Annealing*), y puede ser usado para seleccionar LSPs que optimicen cualquier objetivo operacional dado. Por último, en [79], los autores hacen una formulación del problema de ingeniería de tráfico como un problema de flujo multi-commodity de programación lineal con enrutamiento MPLS/OSPF(MCFTE). El resultado es que la mayoría de tráfico es enrutado por OSPF, mientras solo un pequeño número de túneles MPLS son necesarios para afinar la distribución de tráfico. En la tabla 3 se da un resumen de las propuestas para el soporte de ingeniería de tráfico basadas en la combinación IP y MPLS.

Tabla 3. Propuestas para el soporte de ingeniería de tráfico basadas en IP+MPLS

Referencia	Propuesta
[76]	Método de enrutamiento híbrido (HMR, Hybrid Routing Method).
[77]	Modelo de ingeniería de tráfico en línea para IGP+MPLS.
[78]	Heurística escalable para ingeniería de tráfico IGP+MPLS.
[79]	Formulación del problema de ingeniería de tráfico como un problema de flujo multi-commodity de programación lineal con enrutamiento MPLS/OSPF(MCFTE).

2.3.4 Propuestas de Ingeniería de tráfico basadas en el protocolo LISP

En los últimos años, una estrategia completamente diferente para el enrutamiento en Internet ha sido propuesta mediante el protocolo LISP (*Locator ID Separation Protocol*). Las especificaciones de LISP se contemplan en RFC6830 [80]. LISP es un protocolo basado en la capa de red que permite la separación de direcciones IP en dos nuevos espacios de numeración: localizadores de enrutamiento (RLOC, *Router Locators*) e identificadores de punto final (EID, *Endpoints Identifiers).* Los RLOC se asignan topológicamente a puntos de conexión de red; estos localizadores se utilizan para enrutar y reenviar paquetes a través de la red. Los EID se asignan independientemente de la topología de la red; estos identificadores se utilizan para numerar dispositivos y están a lo largo de límites administrativos [80].

En [81], los autores describen cómo los túneles de reencapsulación de LISP se pueden usar para propósitos de ingeniería de tráfico. Por lo tanto, un paquete puede tomar una ruta especificada administrativamente, una ruta para evitar la congestión, una ruta de recuperación de fallas o múltiples rutas de carga compartida a medida que viaja desde el ITR (*Ingress Tunnel Router*) al ETR (*Egress Tunnel Router*). Al introducir una ruta de localización explícita (ELP, *Explicit Locator Path*), una ITR puede encapsular un paquete a un enrutador de túnel reencapsulado (RTR, *Reencapsulating Tunnel Router*), que desencapsula el paquete y luego lo encapsula en el siguiente localizador en el ELP. Algunos documentos relacionados con el soporte de ingeniería de tráfico por LISP incluyen [82]–[86].

2.3.5 Propuestas de Ingeniería de tráfico basadas en Enrutamiento por Segmentos.

SR (*Segment Routing*, Enrutamiento por segmentos) toma ventajas del paradigma de enrutamiento por el origen. La especificación de SR está en la RFC8402 [87]. En SR, un nodo dirige un paquete a través de una lista ordenada de instrucciones, llamadas segmentos. Un segmento puede representar cualquier instrucción, topológica o basada en el servicio, y un segmento puede tener una semántica local a un nodo SR o global dentro de un dominio SR. SR permite la ejecución de un flujo a través de cualquier ruta topológica y cadena de servicio, al tiempo que mantiene el estado de flujo solo en los nodos de ingreso al dominio SR.

Enrutamiento por Segmento y MPLS. SR se puede aplicar directamente a la arquitectura MPLS [12] sin cambios en el plano de reenvío. Para esto, un segmento se codifica como una etiqueta MPLS, y una lista ordenada de segmentos se codifica como una pila de etiquetas. El segmento a procesar está en la parte superior de la pila. Al completar un segmento, la etiqueta relacionada se extrae de la pila [87].

Enrutamiento por Segmento e IPv6. SR puede aplicarse a la arquitectura IPv6 con un nuevo tipo de encabezado de enrutamiento. Un segmento está codificado como una dirección IPv6.

Una lista ordenada de segmentos se codifica como una lista ordenada de direcciones IPv6 en el encabezado de enrutamiento. El segmento activo se indica mediante la dirección de destino del paquete. El siguiente segmento activo se indica mediante un puntero en el nuevo encabezado de enrutamiento [87].

E*nrutamiento por Segmento para ingeniería de tráfico*. SR se ha propuesto recientemente como una tecnología alternativa de ingeniería de tráfico que permite simplificaciones relevantes en las operaciones del plano de control. Se han propuesto algunos trabajos para respaldar la ingeniería de tráfico mediante enrutamiento por segmento, como en [88], que detalla la política de enrutamiento de segmento para ingeniería de tráfico. En [89], los autores consideran el problema de determinar los parámetros óptimos para el enrutamiento de segmentos en los casos fuera de línea y en línea. En [90], se propone un algoritmo de codificación de etiquetas para el enrutamiento de segmentos MPLS. En [91], los autores presentan un algoritmo de asignación de ruta SR para el problema de asignación de flujo. En [92], los autores proponen modelos de ILP y heurísticas que se utilizan con éxito para evaluar el rendimiento de TE de las redes de paquetes basadas en SR. Finalmente, en [93] los autores analizan datos de tráfico de una red troncal europea en un período de 2011 a 2015, donde el nivel de tráfico total aumenta significativamente. Analizan las diferencias geográficas para seleccionar las horas pico representativas de tráfico como escenarios de referencia para una evaluación de TE usando SR. Los resultados encontrados demuestran que SR ofrece resultados casi óptimos.

En la tabla 4 se da un resumen de las propuestas para el soporte de ingeniería de tráfico basadas en LISP y en enrutamiento por segmentos.

Tabla 4. Propuestas para el soporte de ingeniería de tráfico basadas en LISP y en enrutamiento por segmento.

Referencia	Propuesta	LISP	SR
[80]	Especificaciones de LISP.	X	
[81]	Casos de uso para Ingeniería de tráfico.	X	
[82][83][84][86]	Ingeniería de tráfico interdominio.	X	
[85]	Sistema de Gestión Centralizado e Ingeniería de tráfico.	X	
[88]	Políticas para ingeniería de tráfico de enrutamiento por segmentos.		X
[89]	Optimización de enrutamiento por segmentos en línea y fuera de línea.		X
[90]	Algoritmo de codificación de etiquetas.		X
[91]	Algoritmo de asignación de ruta SR.		X
[92]	Modelos ILP y heurísticas para ingeniería de tráfico.		X
[93]	Evaluación de Ingeniería de tráfico usando SR con datos reales de una ISP europea.		X

2.4 ALGORITMOS DE ENRUTAMIENTO BASADOS EN RESTRICCIONES

El enrutamiento basado en restricciones ha sido un tema de gran interés y abierto en la comunidad de Internet. La búsqueda de soluciones que permitan resolver el problema de congestión y carencia de calidad de servicio en Internet ha permitido que se propongan gran cantidad de aproximaciones cuyos objetivos están enfocados a resolver estos problemas. En esta sección se presenta una revisión de propuestas de algoritmos de enrutamiento encontrados en la literatura. Los algoritmos se han organizado en tres categorías de acuerdo con sus funcionalidades: La primera categoría son los algoritmos de enrutamiento basados en restricciones para soportar ingeniería de tráfico, la segunda son los que contribuyen al soporte de calidad de servicio y la tercera incluye los híbridos, es decir, combina objetivos hacia el soporte tanto de ingeniería de tráfico como calidad de servicio. En la literatura se ha demostrado que los algoritmos cuyos objetivos apuntan al soporte de ingeniería de tráfico son NP-hard [70], [94], y los que se enfocan en soporte de calidad de servicio son NP-complete [53], [95], esto conduce a que sea un tema abierto y permite que investigadores propongan nuevas soluciones heurísticas para resolver las problemáticas mencionadas.

2.4.1 Algoritmos de Enrutamiento Basados en Restricciones para el soporte de Ingeniería de tráfico

La Ingeniería de Tráfico de Internet se encarga del problema de optimización y evaluación del rendimiento de redes IP en operación. El objetivo es mejorar el rendimiento de la red optimizando el uso de los recursos y del tráfico mediante la aplicación de tecnologías y principios científicos que permitan la medición, caracterización, modelado y control del tráfico de Internet [1]. El problema del enrutamiento de Internet es que este se hace seleccionando la ruta más corta o de mínimo coste. Los algoritmos más utilizados son Dijkstra y Bellman Ford [18]. Este tipo de enrutamiento trae como consecuencia que algunos caminos se congestionen porque son sobre-utilizados y otros sean sub-utilizados, lo que conlleva a la congestión de la red. Para evitar estos problemas de congestión surgen numerosas propuestas y mecanismos para el soporte de Ingeniería de Tráfico [96]. Dentro de estas propuestas surge MPLS [12], como tecnología que proporciona ingeniería de tráfico y calidad de servicio. Para el enrutamiento y el soporte de ingeniería de tráfico, MPLS se apoya en un algoritmo basado en restricciones; muchos algoritmos han sido propuestos para este fin, pero el más sencillo y más utilizado es el denominado CSPF *(Constraint Shortest Path First*) [18], el cual es una extensión del algoritmo Dijkstra, ya que implementa una modificación, que se refiere a la adicción de restricciones de ancho de banda.

Las restricciones, cuyo objetivo es el de contribuir al soporte de ingeniería de tráfico se enfocan en requerimientos de ancho de banda, balanceo de carga, utilización eficiente de

los recursos de red, enrutamiento multicamino y minimización del costo de enrutamiento. La gran mayoría de soluciones que se han propuesto han sido pensadas para trabajar con MPLS [97]. Las primeras propuestas para CBR (*Constraint-Based Routing*) incluyen WSP [97] (*Widest Shortest Path* – Ruta más Corta más Amplia) y SWP [98] (*Shortest-Widest Path*, Ruta más Amplia más Corta). WSP y SWP utilizan un algoritmo del camino más corto, Dijkstra o Bellman-Ford, para los cálculos de la ruta. WSP selecciona un camino con máxima capacidad de ancho de banda entre los que tienen una longitud de saltos dada. SWP optimiza primero en conteo de saltos y cuando existen múltiples caminos con la misma cantidad de saltos, selecciona entre ellos uno con máximo ancho de banda [94].

El algoritmo MIRA *(Minimum Interference Routing Algorithm*) [71], es un algoritmo de enrutamiento de interferencia mínima que fue propuesto para balanceo de carga. Opera en los router de ingreso y establece una sola ruta para una petición. Durante la construcción de un camino, el algoritmo MIRA busca la minimización de interferencia con los caminos potenciales entre todos los otros pares de ingreso y egreso. Para hacer esto, el algoritmo intenta minimizar las cargas de los enlaces críticos en la red. Un enlace crítico es aquel que se encuentra en un cruce potencial de tráfico del dominio [94]. MIRA tiene dos versiones, el S-MIRA (involucra un criterio de suma ponderada) y el L_MIRA (involucra un criterio lexicográfico). En ambas versiones se intenta minimizar la interferencia que se está formando con el resto de las rutas respetando la importancia administrativa en S-MIRA y la capacidad de ruta residual de L_MIRA.

FRA (*Fuzzy Routing Algorithm*) [99], opera en lógica difusa. FRA busca tres objetivos: El primero es maximizar el Maxflow, es decir, la capacidad del enlace cuello de botella en el camino. FRA primero optimiza para seleccionar la ruta con recurso residual máximo sobre su enlace cuello de botella. El segundo objetivo es maximizar el ancho de banda residual en los enlaces diferentes al del enlace cuello de botella. Y el tercero es minimizar la longitud de la ruta por su número de saltos. El algoritmo aplica una función a un conjunto de miembros difusos combinando estos tres objetivos dentro un criterio para elegir incrementalmente entre los enlaces y establecer la ruta comenzando a partir del router de Ingreso [94].

PBR (*Profile Based Routing*) [100]. Es un algoritmo de enrutamiento restringido orientado a control de admisión, está basado en el pre-conocimiento de clases de flujos nombrados como perfiles que esperan solicitar entrada al dominio. La aproximación es la división de recursos de dominio dentro de los perfiles entre pares de ingreso y egreso conocidos a priori, lo cual se hace en la primera fase del algoritmo. En la segunda fase, los flujos son admitidos uno a la vez con base en las solicitudes de recursos y la capacidad restante en la clase de tráfico que se asignó. El objetivo principal del PBR es maximizar el flujo de admisión. Los resultados muestran que PBR supera a MIRA, WSP y el enrutamiento del camino más corto

en varias métricas, incluida la fracción de solicitudes enrutadas y la fracción del ancho de banda solicitado enrutado. [94].

PPF (*Primary Path First*) y M-AIMD (*Multipath-Additive Increase Multiplicative Decrease*), se proponen para el control de admisión y ajuste de tráfico en rutas preestablecidas [101]. Ambas versiones tienen como objetivo el consumo primario de los recursos de la ruta y distribuir el tráfico global de manera equitativa con base en la demanda declarada y la capacidad primaria de las fuentes [94]. En PPF, esta equidad se logra mediante el uso de un parámetro umbral fijado en la distribución de la carga actual. En M-AIMD, los parámetros externos se utilizan para el control de admisión. El rendimiento de M-AIMD/PPF es insignificante comparado con M-AIMD.

V-PREPT en [62], tiene como objetivo la minimización de re-enrutamiento causado por preferencia. El algoritmo formula este objetivo como tres factores combinados aditivamente, cada uno tiene coeficientes de afinación que son entradas al algoritmo. Estos factores son: i) Minimizar el número de LSPs anticipados: la correlación directa con el objetivo, ii) Minimizar la cantidad de ancho de banda anticipado: destinado a reducir el desperdicio de ancho de banda por la anticipación del LSP con la menor cantidad de ancho de banda que satisfaga la solicitud, iii) Minimizar sobre la prioridad del LSP: de manera que, entre los LSPs cuyas prioridades lo permitan, sus prioridades relativas sean consideradas. En [62], también se describe Adapt-V-PREPT, una versión de V-PREPT que también considera las preferencias para las tasas de ancho de banda. Los algoritmos son probados en distintas solicitudes de ancho de banda y su funcionamiento es bastante cerca de lo óptimo [94].

Entre otras propuestas, en [102], los autores proponen dos algoritmos de enrutamiento basados en restricciones de ancho de banda multicamino para TE usando MPLS. Estos algoritmos dividen las restricciones de ancho de banda en múltiples sub-restricciones y encuentran un camino restringido para cada sub-restricción. Este enfoque se da como solución a que en el algoritmo CSPF hay una alta probabilidad de no encontrar el camino factible para restricciones de un ancho de banda grande, la cual es una de las restricciones más importantes para ingeniería de tráfico.

En [103], los autores proponen dos algoritmos que contribuyen al soporte de ingeniería de tráfico en redes MPLS en dos áreas: enrutamiento basado en restricciones y balanceo de carga. El primer algoritmo denominado "*Parallel-Path-Based Bandwidth Scheme*" (PPBS), esquema de ancho de banda basado en camino paralelo, hace uso de LSPs paralelos en la escogencia de caminos con restricciones de ancho de banda. El segundo, es un algoritmo de balanceo de carga basado en retroalimentación (FBLB, *Feedback-based Load Balancing Algorithm*) para distribuir tráfico sobre los LSPs paralelos determinados por PPBS. Los resultados demuestran un mejoramiento en la probabilidad de bloqueo de los flujos usando

PPBS con respecto a la carga promedio de los enlaces, los saltos a lo largo del camino y el número posible de LSPs paralelos. También demuestran la efectividad y estabilidad del algoritmo FBLB para la distribución de tráfico y balanceo de carga.

Un algoritmo heurístico basado en NSGA-II [104], es propuesto para el problema de ingeniería de tráfico multi-objetivo en las redes backbone de Internet que utilicen MPLS. El objetivo es distribuir las demandas de tráfico sobre la red de tal forma que simultáneamente minimice el costo de enrutamiento y el balanceo de carga. El rendimiento de la propuesta es comparado con la solución exacta generada usando el método Chevyshev lexicográfico.

Finalmente, en [105], proponen un algoritmo de optimización de enrutamiento, para alcanzar equilibrio de tráfico, mientras reduce la penalidad que deben pagar a los usuarios los operadores de red debido a la interrupción de los servicios. Los experimentos realizados muestran que el algoritmo alcanza el equilibrio de tráfico y reducen efectivamente la penalidad potencial de los operadores de red.

En la tabla 5, se da un resumen de los algoritmos cuyo objetivo es el soporte Ingeniería de tráfico, se proveen las restricciones y los objetivos propuestos.

Tabla 5. Especificación de los algoritmos basados en restricciones que contribuyen al soporte de ingeniería de tráfico.

Aproximación	*Restricciones*	*Objetivo*
MIRA [71]	*Bw*	*Balanceo de Carga.*
FRA [99]	*Bw, Número de saltos*	*Utilización eficiente de los recursos y Minimizar la longitud de la ruta.*
PBR [100]	*Bw*	*Maximizar el flujo de admisión.*
PPF & M-IMD [101]	*Bw*	*Utilización eficiente de los recursos y Balanceo de Carga.*
V-PREPT [62]	*Bw*	*Utilización eficiente de recursos y Minimización de LSPs anticipados.* *Minimizar el ancho de banda por anticipación deLSPs.* *Minimizar sobre la prioridad de LSPs.*
[102]	*Bw*	*Balanceo de carga.*
PPBS-FBLB [103]	*Bw*	*Caminos Paralelos y Balanceo de Carga.*
[104]	*Costo, Bw*	*Minimizar Costo de enrutamiento y Balanceo de Carga.*
[105]	*Costo, Bw*	*Utilización eficiente de los recursos y Equilibrio de Tráfico-Balanceo de carga.*

2.4.2 Algoritmos de Enrutamiento basados en Restricciones para el soporte de Calidad de Servicio

Esta categoría incluye aquellos algoritmos cuyo objetivo es seleccionar el camino que satisfaga un conjunto de restricciones que apuntan al soporte de calidad de servicio. Este es un problema NP-complete [95], [106] , [107] y esto ha conducido a que se realicen muchas propuestas de algoritmos heurísticos [95]. Los objetivos de estos algoritmos están enfocados en restricciones de ancho de banda, delay, jitter y pérdidas de paquetes. En la tabla 6, se

resumen los algoritmos basados en restricciones de QoS y una breve descripción de cada propuesta fue publicada en [108].

En [109], se presenta un estudio de los algoritmos para enrutamiento de calidad de servicio basado sobre métricas, las cuales son restringidas u optimizadas durante el proceso de selección del camino. El estudio presentado está enfocado a los problemas de enrutamiento MCP (*Multi-Constrained Path Problem*) y MCOP (*Multi-Constrained Optimal Path Problem)*. El problema MCOP es una abstracción y extensión del enrutamiento de QoS. El problema MCOP intenta encontrar el camino de costo mínimo que satisface las restricciones y es NP-complete aún para una sola restricción. Por su parte, MCP es el problema MCOP sin optimización del camino y también es NP-complete para más de una restricción.

En [110], Jaffe propone dos algoritmos para el problema MCP bajo dos restricciones (uno polinomial y otro pseudo-polinomial). En esta aproximación las dos restricciones son combinadas dentro de una sola métrica usando una función lineal. Usa el algoritmo del camino más corto Dijkstra y encuentra un camino factible.

El algoritmo Iwata en [111], es propuesto como solución al problema MCP. El algoritmo primero calcula un camino más corto basado en una medida de QoS y luego revisa si otras restricciones son encontradas. Si existe una ruta más corta, que de acuerdo con cierta medida de QoS cumple con todas las restricciones, el algoritmo se detiene, de lo contrario el algoritmo se repite con otra medida hasta que un camino factible sea encontrado. La complejidad del peor caso de este algoritmo es "m" veces la del algoritmo Dijkstra. Un inconveniente con esta aproximación es que no hay garantía que la optimización de la selección de la ruta con respecto a alguna medida única conduzca a un camino factible [109], [95].

TAMCRA (*Tuneable Accuracy Multi-Constrained Routing Algorithm*, Algoritmo de Enrutamiento Multi-Restringido de Precisión Sintonizable) [112] y SAMCRA (*Self-Adaptive Multi-Constrained Routing Algorithm*, Algoritmo de enrutamiento Multi-Restringido Auto-Adaptativo) en [113], están basados en tres conceptos fundamentales: i) Una medida no lineal de la longitud de la ruta, ii) Un enfoque de ruta k-shortest, iii) Principio de rutas no-dominadas [109]. SAMCRA es el sucesor de TAMCRA, incluye los tres conceptos fundamentales de TAMCRA y agrega otro más "Mirar hacia adelante". La ventaja de SAMCRA sobre TAMCRA es que el espacio de búsqueda se reduce por el pre-cálculo del camino más corto hacia el destino[109].

Chen and Nahrstedt [114], proponen dos algoritmos basados en programación dinámica el EDSP (Extended Dijkstra's Shortest Path) y el EBF (Extended Bellman-Ford). Cuando el grafo es escaso y el número de nodos relativamente grandes, se espera que EBF pueda dar un mejor rendimiento que EDSP en términos de tiempo de ejecución. Sin embargo, para lograr

un buen desempeño se necesitan grandes restricciones, lo cual hace que este enfoque tenga alta complejidad computacional para fines prácticos [95]. En estos algoritmos el problema MCP es simplificado reduciendo el tamaño del peso de los enlaces [109].

El algoritmo Aleatorio descrito en [115], es propuesto para el problema MCP. Este algoritmo se compone de dos partes, la fase de iniciación y la búsqueda aleatoria. El algoritmo inicia desde la fuente y explora el grafo usando un algoritmo aleatorio BFS (Breadth-First Search). Mediante el uso de la información obtenida en la fase de inicialización, el algoritmo aleatorio BFS puede comprobar si existe una posibilidad de éxito antes de descubrir el nodo. Si no hay ninguna posibilidad, el algoritmo prevé la trampa y no vuelve considerar tales nodos. En el peor de los casos la complejidad del algoritmo aleatorio es mayor m+1 veces que el Dijkstra [95], donde m es el número de medidas de QoS. La ventaja principal de este algoritmo es que trabaja bajo cualquier número de restricciones.

En el algoritmo H_MCOP, la búsqueda del camino viable se hace mediante la aproximación de la función no lineal, igual a como se hace en TAMCRA [116] . Para cumplir estos objetivos, H_MCOP (*Multi-Constrained Optimal Path*) ejecuta dos versiones modificadas del algoritmo Dijkstra en direcciones atrás y adelante. Dado que el algoritmo considera rutas completas antes de llegar al destino, se pueden prever algunos caminos viables durante la búsqueda. Si se utiliza este algoritmo sólo para el problema MCP, la ejecución termina cuando encuentra o prevé un camino factible, lo que reduce el tiempo de ejecución del algoritmo.

El algoritmo A*Prune [117] considera el hecho de encontrar no sólo uno, sino múltiples caminos más cortos que están dentro de las restricciones. La función de longitud lineal usada en este algoritmo es la misma utilizada en el algoritmo Jaffe's. Si no hay caminos factibles presentes, el algoritmo retornará aquellos que están dentro de las restricciones. Para cada medida de QoS, el algoritmo calcula los caminos más cortos. Los pesos de estos caminos serán utilizados para evaluar si una sub-ruta puede convertirse en un camino viable, algo similar a como lo realiza el algoritmo H_MCOP. El nodo con el peso más corto seleccionado de extremo a extremo es extraído de la pila y luego todos sus vecinos son analizados. Los vecinos que causan un bucle o una violación de las restricciones son dados de baja [95]. En el peor de los casos la complejidad del algoritmo A*Prune crece exponencialmente con el tamaño de la red [109], es posible implementar un algoritmo A*Prune limitado, el cual corre en tiempo polinomial con el riesgo de perder exactitud.

En [118], el algoritmo de enrutamiento NLR_MCP (*Non linear Lagrange*) para resolver el problema MCP ejecuta dos veces el algoritmo Dijkstra, una en dirección de reversa y otra en dirección hacia adelante . El algoritmo Dijkstra en dirección inversa trata de buscar el camino desde cualquier otro nodo que minimice la función de costo. Si el algoritmo Dijkstra en dirección de reversa puede encontrar el camino deseado, el algoritmo se detiene y retorna

esta ruta. Si no, continúa ejecutando el algoritmo Dijkstra en dirección hacia adelante. El resultado de las simulaciones muestra que el costo de NLR_MCP comparado con el de H_MCP puede ser considerablemente mejor sin sacrificar tiempo de complejidad [109].

Un algoritmo exacto llamado A*_MCSP (*Multi Constraint Shortest Path*) es propuesto en [119], el cual introduce la noción de estado y la relación de dominio entre los estados. A*_MCSP es un algoritmo de búsqueda primero en amplitud que utiliza una estrategia de búsqueda muy conocida en la inteligencia artificial. A*_MCSP elimina el costo de mantener los caminos parciales similar a como lo hace el algoritmo A*Prune [109].

Por otro lado, DCLC (*Delay Constrained Least Cost*) tiene como fin encontrar un camino *p* desde la fuente hasta el destino, de tal manera que el mínimo costo sea alcanzado y que satisfaga las restricciones de retraso [114] [120].

El algoritmo SF-DCLC (*Selection Function -Delay Constraint Least Cost*) [121], es un algoritmo basado sobre la función de selección para los problemas de costo mínimo de restricciones de retardo, requiere información de red limitada en cada nodo y es capaz de encontrar un camino que satisface el retraso dado, si existe tal camino. El mayor problema que tiene este algoritmo es que encuentra una ruta de menor costo con respecto al retraso que no es más que el límite autorizado [109].

En [122], se describe LCLD (*Least Cost Least Delay*), que es una versión modificada de SF-DCLC. Este algoritmo utiliza una función de peso, que siempre es capaz de encontrar un camino basado en menor costo y retraso satisfaciendo el retraso dado si existe tal camino. El objetivo de este algoritmo es satisfacer los requerimientos de QoS para cada conexión admitida y para lograr una mejor eficiencia en la utilización de recursos [109].

El algoritmo FPSA en [123] propuesto para el problema MCP tiene como objetivo encontrar una ruta viable si hay más de una ruta viable dentro de la red. La ruta viable es seleccionada de tal manera que debería consumir menos recursos de red entre los múltiples caminos viables disponibles. La optimalidad puede ser considerada por la selección del número de saltos contados, menor retraso, rendimiento y ancho de banda. El algoritmo FPSA encuentra la ruta viable mediante la adopción de dos restricciones, ancho de banda y retardo. Los resultados de la simulación presentados muestran que ofrece una tasa de éxito mayor en comparación a H_MCOP [109].

Algoritmo híbrido (DCCR+SSR) es propuesto en [124], y combina los algoritmos de enrutamiento de restricción Costo-Retardo DCCR (*Delay-Cost-Constraint Routing*) y Reducción de espacio de búsqueda SSR (*Search Space Reduction*) [125]. DCCR es una variante del enrutamiento del camino más corto K-ésimo. Es un problema DCLC convertido a DCC (*Delay Cost-Constraint)*. La diferencia entre DCCR y DCLC es que tanto el costo como también

el retardo es limitado a dos restricciones diferentes. Como un resultado de la búsqueda el espacio es reducido, tanto como los caminos no satisfagan ambas restricciones se eliminan.

Por su parte, el algoritmo heurístico de ruta limitada, presenta dos heurísticas para el problema MCP: la heurística "granularidad limitada" y la LPH (*Limited Path Heuristic*, Heurística Ruta Limitada) [126]. LPH tiene altas probabilidades de encontrar una ruta viable, siempre y cuando la ruta exista. LPH está basado en el algoritmo Bellman-Ford y usa dos de los conceptos fundamentales de TAMCRA, que son el no-dominio y el almacenamiento en la mayoría de las *k* rutas por cada nodo [95]. Sin embargo, mientras TAMCRA usa una k-shortest como enfoque, LPH guarda la primera ruta k, que no es necesariamente las más corta. LPH no comprueba si una ruta obedece a unas restricciones dadas, lo hace cuando encuentra el destino.

El algoritmo HAMCRA, es un algoritmo híbrido que combina los conceptos de los algoritmos SAMCRA y TAMCRA. HAMCRA usa la misma función no lineal y los mismos conceptos agregando uno nuevo LB (*Lower Bound*, Límite Inferior), que permite la reducción del espacio de búsqueda. El concepto de LB se incluye para para calcular y comprobar si la ruta de extremo a extremo cumple con las restricciones [127]. HAMCRA encuentra caminos factibles muy rápido, pero la complejidad aumenta cuando las restricciones están estrictamente relacionadas con el peso de las rutas multidimensionales de caminos más cortos y los pesos de los enlaces están negativamente relacionados [109].

En [128], presentan un algoritmo de enrutamiento para encontrar caminos factibles que minimicen el costo incurrido por una red MPLS para soportar las solicitudes de ancho de banda del usuario. El costo es atribuido al transporte de ancho de banda, esfuerzos de señalización y conmutación para la conexión solicitada. El algoritmo de enrutamiento es escalable y opera bajo información de red inexacta. El rendimiento del algoritmo es superior comparado con el algoritmo de enrutamiento del camino más corto, pero con incremento en la complejidad computacional.

Un algoritmo heurístico es propuesto en [129], TS_MCOP, que opera mediante la aplicación de búsqueda tabú al algoritmo Dijkstra. Este heurístico primero traslada múltiples restricciones de QoS en una única métrica y luego encuentra un camino factible mediante búsqueda tabú. Los resultados presentados por los autores muestran que el algoritmo tiene características de buen rendimiento, relación de éxito y costo.

Un algoritmo de enrutamiento eficiente para mejorar la calidad de servicio en Internet es propuesto en [130], es una clase de algoritmo de enrutamiento multi-restringido. Para evitar el problema *NP-complete* y el incremento de la eficiencia computacional, adicionaron algunas mejoras como la definición del camino no lineal donde los sub-caminos pueden no

ser los caminos más cortos. Los autores concluyen que, aunque es una metodología efectiva se encontraron algunas pérdidas de exactitud en el cálculo del sub-camino.

En [131], los autores muestran una solución para enrutamiento de QoS jerárquico mediante la introducción del protocolo llamado Macro-enrutamiento multi-restringido. Este protocolo usa el protocolo de Macro-enrutamiento con la técnica de agregación de malla completa extendida para determinar múltiples rutas de QoS jerárquicas. Las pruebas presentadas por los autores muestran que el protocolo de multi-enrutamiento multi-restringido supera cualquier otro protocolo de enrutamiento jerárquico que use la agregación de malla completa, encontrando mejores y más caminos de QoS.

Un método para resolver el enrutamiento de múltiples restricciones es propuesto en [132]. Primero establece un modelo de enrutamiento de QoS multi-restringido y construye una función de valor de conveniencia mediante la transformación de restricciones de calidad de servicio con una función de penalidad. Luego, varias fórmulas iterativas del original PSO (*Particles Swarm Optimization*) son mejoradas para adaptar el espacio de búsqueda no continuo del problema de enrutamiento. Las mutaciones de ideas y selección natural de los algoritmos genéticos GA (*Genetic Algorithm*) son aplicadas al PSO para mejorar el desempeño convergente. Los autores mencionan que combinan PSO y GA, teniendo en cuenta que PSO es un algoritmo de optimización que ha sido aplicado para encontrar caminos más cortos en la red, sin embargo, este puede caer en la solución óptima local y no ser capaz de resolver el enrutamiento basado en múltiples restricciones.

En [133], proponen un algoritmo de enrutamiento multi-restringido bidireccional BMCOP, el cual emplea los conceptos de no dominancia, búsqueda bidireccional, enrutamiento independiente, manejo de diferentes métricas, los cuales se ha demostrado que son útiles para abordar los problemas MCOP. Los resultados de simulación de BMCOP y los detalles teóricos, demostraron que puede alcanzar rendimiento cerca al óptimo para ambos algoritmos MCP y MCOP.

Un algoritmo de enrutamiento heurístico con restricciones de ancho de banda y retardo, denominado HRABDC (*Heuristic Routing Algorithm with Bandwidth Delay Constraints*), es propuesto en [134]. La meta del algoritmo es aceptar tantas solicitudes de enrutamiento como sea posible. El algoritmo calcula relativamente los pesos de enlace basado en los anchos de banda de los enlaces. Luego la heurística es aplicada al algoritmo Dijkstra para encontrar un camino que satisfaga restricción de retardo y con los pesos más bajos posibles. Los experimentos sobre diferentes topologías tienen mejor rendimiento que soluciones existentes con respecto a la relación de admisión y bajo tiempo de cómputo. En la tabla 6, se resume los algoritmos basados en restricciones de QoS.

Tabla 6. Especificación de los Algoritmos que contribuyen a la Calidad de Servicio

Aproximación	*Restricciones*	*MCP*	*MCOP*
Jaffe's [110]	*Pesos, Longitud.*	*x*	
Iwata's [111]	*Costos, delay.*	*x*	
TAMCRA [112]	*Longitud, Costos.*	*x*	
SAMCRA [113]	*Longitud, Costos.*	*x*	
EDSP [114]	*Costos.*	*x*	
Algoritmo Aleatorio [115]	*Amplitud, Peso, Costo, Retardo.*	*x*	
H_MCOP [116]	*Pesos.*		*x*
Algoritmo A*Prune [117]	*Pesos.*	*x*	
NLR_MCP [118]	*Costos.*	*x*	
A*MCSP [119]	*Costos.*	*x*	
DCLC [120]	*Costo, Retardo.*		*x*
SF-DCLC [121]	*Costo, Retardo.*		*x*
LCLD [122]	*Costo, Retardo.*		*x*
FPSA [123]	*Retardo, Bw.*		*x*
DCCR+SSR [124]	*Costo, Retardo.*		*x*
LPH [126]	*Delay, Jitter, Bw.*	*x*	
HAMCRA [127]	*Delay, jitter.*	*x*	
[128]	*Bw, Retardo, Distancia.*		
TS_MCOP [129]	*Costo.*		*x*
[130]	*Retardo, jitter, Bw.*	*x*	
[131]	*Costo, Conteo de Saltos y Retardo.*	*x*	
GA-PSO [132]	*Bw, retardo,perdida de paquetes, jitter.*		*x*
BMCOP [133]	*Costo.*	*x*	*x*
HRABDC [134]	*Bw, Retardo.*	*x*	

2.4.3 Algoritmos de Enrutamiento basados en Restricciones para el soporte de TE y QoS

En esta sección se describen algunos algoritmos encontrados en la literatura para soportar tanto ingeniería de tráfico como QoS. Es importante resaltar aquí que, aunque en los algoritmos presentados en la anterior categoría, la literatura mencione que sus metas están solamente enfocadas al soporte de QoS, podrían estar en esta categoría si algunas de sus restricciones están enfocadas al ancho de banda y eso conduzca a balanceo de carga, ya que eso permitiría una mejor utilización de los recursos y contribuiría a reducir la congestión, que son los intereses principales de la ingeniería de tráfico. En la tabla 7, se resumen los algoritmos presentados en esta categoría, las restricciones utilizadas y los objetivos involucrados. Una breve descripción se encuentra publicada en [108].

Un algoritmo que optimiza la utilización de la red, mientras ofrece garantías de calidad de servicio, denominado Q-BATE es propuesto en [135]. Presenta una nueva función de longitud del camino para perseguir tales objetivos. El algoritmo se basa en una fase de pre-análisis y estrategia de primero-profundidad para encontrar eficientemente el camino factible con longitud más pequeña. Q-BATE supera en cuanto a relación de admisión y rendimiento a otros con los cuales fue comparado.

BGMRA en [136], es un algoritmo de enrutamiento de LSP de ancho de banda garantizado para redes MPLS. Entre los objetivos del algoritmo están: balancear carga de tráfico usando caminos subutilizados, optimizar la utilización de recursos de la red usando el algoritmo Dijkstra y minimizar niveles de interferencia entre origen-destino para reservar más recursos para futuras demandas de ancho de banda. En los experimentos realizados se observa que BGMRA tiene mejor rendimiento en redes complejas en cuanto a la razón de bloqueo.

Proponen un algoritmo híbrido denominado HMTA (*Hybrid MPLS tunneling Algorithm*) en [137], el cual comprende de técnicas de diferenciación de ancho de banda y retardo como también con base en bits de prioridad contenidos en cada paquete. La diferenciación de enlaces está basada en umbrales de ancho de banda y retardo, lo cual es llamado como información de estado de enlace. La diferenciación es hecha para reducir la probabilidad de bloqueo y el número de intercambios de estados de calidad de servicio que se llevan a cabo resultando en incremento de sobrecarga de QoS.

Tabla 7. Especificación de los Algoritmos Basados en Restricciones que Contribuyen al soporte de Ingeniería de Tráfico y Calidad de servicio.

Aproximación	*Restricciones*	*Objetivo*
Q-BATE [135]	*Longitud del camino*	*Utilización eficiente de los recursos QoS*
BGMRA [136]	*Bw*	*Balanceo de Carga* *Utilización eficiente de los recursos* *Minimizar niveles de interferencia*
HMTA [137]	*Bw, Retardo*	*Probabilidad de Bloqueo* *Utilización Eficiente de los recursos QoS*

2.5 IPV6 MÓVIL

El protocolo IPv6 Móvil está especificado en la RFC6275 [138], y fue creado para permitir que los nodos sean alcanzables mientras se mueven en la Internet IPv6. IPv6 móvil permite que un nodo móvil se mueva de un enlace a otro sin cambiar la dirección local del nodo móvil. Los paquetes pueden ser encaminados al nodo móvil utilizando esta dirección, independientemente del punto actual de vinculación a Internet del nodo móvil.

Cada nodo móvil (MN, *Mobile Node*) se identifica por su dirección local "*Home Address*" sin importar el punto de ubicación en Internet. Cuando el nodo móvil se encuentra en una red diferente a la red local, le es asignada una dirección temporal denominada "*care-of-address*", la cual provee información de su ubicación actual. Entonces todos los paquetes IPv6 direccionados a la dirección local del nodo móvil son enrutados transparentemente a su *care-of-adress.* El protocolo habilita a los nodos IPv6 para almacenar en caché el enlace de una dirección local del MN con su "care-of-address", por tanto, cualquier paquete destinado

al MN es enviado directamente a éste, a la *care-of-address* desde cualquier nodo móvil o estacionario.

Por tanto, un MN es siempre localizable mediante su dirección local, ya sea que este esté conectado a su enlace local o esté fuera de él. La dirección local "Home Address" es una dirección IP asignada al nodo móvil dentro del prefijo de la subred local sobre su enlace local. Mientras un MN está en su subred local, los paquetes direccionados a su dirección local son enrutados al enlace local del nodo móvil, usando mecanismos de enrutamiento de Internet convencional.

Cuando un nodo móvil está conectado a algún enlace foráneo lejos de su enlace local, este es también alcanzable por su dirección "*care-of-address*". La "*care-of-address*" es una dirección IP asociada con un nodo móvil que tiene el prefijo de red de un particular enlace foráneo. El MN puede adquirir su "*care-of-address*" a través de mecanismos IPv6 convencionales, tales como auto-configuración *stateless* o *stateful*. Tanto como el MN permanezca en esta ubicación, los paquetes direccionados a su "*care-of-address*" serán enrutados al MN. En la Figura 6, se puede observar la arquitectura de una red IPv6 móvil.

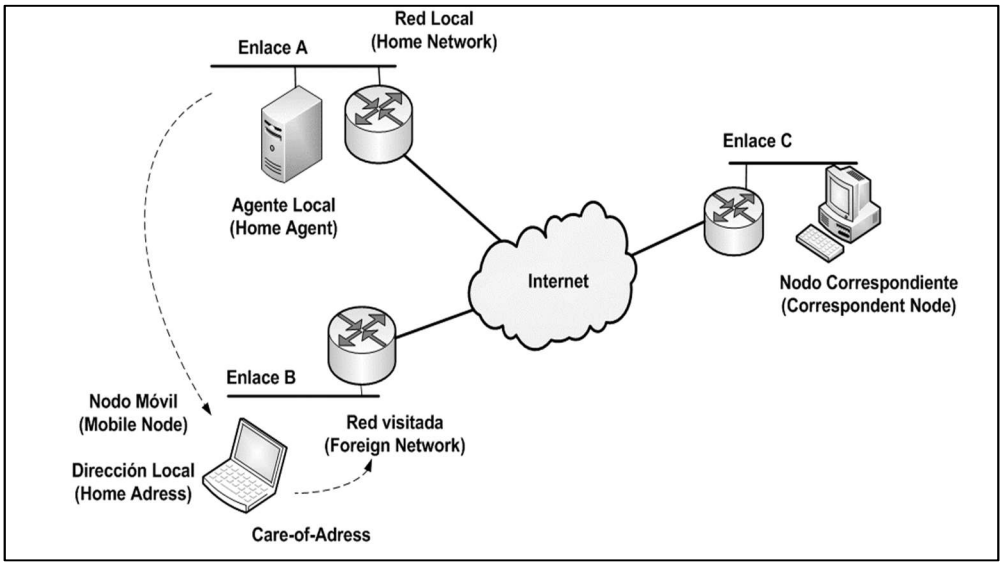


Figura 6. Arquitectura de Red IPv6 Móvil [138].

La asociación entre una dirección local del MN y la "*care-of-address*" es conocida como "*binding*" para el nodo móvil. Cuando el MN esta fuera de su subred local, el MN registra su "*care-of address*" primaria con un router en su red local, el cual funciona como el agente local "*Home Agent*"(HA) para el nodo móvil. El MN hace su registro enviando mensajes de

actualización "*Binding Update*"(BU) al agente local. El agente local responde al MN mediante mensajes de reconocimiento "*Binding Acknowledgement*"(BAck).

Por otro lado, cualquier nodo que se comunique con un MN es referido como un nodo correspondiente "*Correspondent node*"(CN) del nodo móvil, y puede este ser móvil o estacionario. Hay dos modos posibles para la comunicación entre el nodo móvil y un nodo correspondiente. El primer modo, es "tunelización bidireccional", que no requiere soporte de IPv6 Móvil desde el nodo correspondiente y está disponible aún si el nodo móvil no tiene registrado su enlace actual con el nodo correspondiente. Los paquetes desde el nodo correspondiente son enrutados al agente local y luego tunelizados al nodo móvil. Los paquetes dirigidos al nodo correspondiente son tunelizados desde el nodo móvil al agente local (revertir el túnel) y luego enrutados normalmente desde la red local al nodo correspondiente. En este modo, el agente local usa "*Proxy Neighbor Discovery*" para interceptar cualquier paquete IPv6 direccionado a la dirección local del nodo móvil sobre el enlace local. Cada paquete interceptado es tunelizado a la *care-of-address* primaria del nodo móvil. La tunelización es desarrollada usando encapsulación IPv6. El Segundo modo, es optimización de la ruta, requiere que el nodo móvil registre su binding actual al nodo correspondiente. Los paquetes desde el nodo correspondiente pueden ser enrutados directamente a la dirección *care-of address* del MN. El enrutamiento directo a la *care-of-address* del MN permite usar caminos de comunicación más cortos. Esto también elimina la congestión al enlace local y agente local del MN [138].

2.5.1 Pv6 móvil Jerárquico

El protocolo IPv6 móvil jerárquico, HMIPv6 (*Hierarchical Mobile IPv6*) [139], es una extensión del protocolo IPv6 móvil (MIPv6, *Mobile IPv6*) [138], con el objetivo de solucionar los problemas de micromovilidad, reduciendo la carga de señalización y por tanto, la latencia del handover y pérdida de paquetes. En una red HMIPv6 intervienen los mismos elementos de red de MIPv6 como son el agente local (*Home Agent, HA*), el nodo correspondiente (*Correspondent Node, CN*), routers de acceso a la red (*Access Router, AR*). Adicionalmente, HMIPv6 introduce un nuevo nodo denominado MAP (*Mobility Anchor Point*). El MAP puede localizarse en cualquier nivel de una red jerárquica de routers, incluyendo el router de acceso (AR). El MAP es el encargado de gestionar la movilidad del MN. La función principal del MAP es disminuir la cantidad de señalización en comparación con MIPv6 cuando el nodo móvil se encuentra fuera de un dominio local. La reducción de la cantidad de señalización se debe a que el nodo móvil envía mensajes de actualización de su ubicación (*Bindings Update*, BU) al MAP local, en vez de a su agente de movilidad local (*home agent,* HA) y al nodo correspondiente (*Correspondent Node*, CN), como en MIPv6. Debido a

que estos últimos por lo general están más lejos del nodo móvil[139].También a que se requiere transmitir solo un mensaje BU por el MN antes de que el tráfico enviado desde el HA y todos los CNs sean re-enrutados a su nueva ubicación. Esto es independiente del número de nodos correspondientes con los que el nodo móvil se esté comunicando.

Un MAP es esencialmente un HA local. El propósito de introducir el modelo de gestión de movilidad jerárquico en MIPv6, es para mejorar su rendimiento, reduciendo el impacto sobre MIPv6 u otros protocolos IPv6.

HMIPv6 también soporta traspasos rápidos mediante el protocolo *Fast Handover* para IPv6 móvil, FMIPv6 [140]. Al igual que MIPv6, esta solución es independiente de las tecnologías de las de acceso, permitiendo movilidad dentro o entre diferentes tipos de redes de acceso.

Un nodo móvil que entra a un dominio MAP recibe anuncios de router (*Router Advertisement*, RA) que contiene información sobre uno o más MAPs locales. El MN puede enlazar su ubicación actual LCoA (*Local Care-of address*), con una dirección configurada con el prefijo de su dominio MAP, llamada RCoA (*Regional Care-of address*). El MAP actúa como un HA, recibe todos paquetes enviado desde el CN hacia el MN, los tuneliza y envía hacia la nueva ubicación del MN identificada con la nLCoA (*New Local Care-of address*). De igual forma, si el MN envía información hacia un CN, la información enviada es tunelizada por el MN hacia el MAP. El MAP se encarga de retirar la cabecera del túnel y reenviar la información original hacia el CN. El mecanismo de tunelización utilizado tanto en MIPv6 como en HMIPv6, está descrito en la especificación RFC2473 [141].

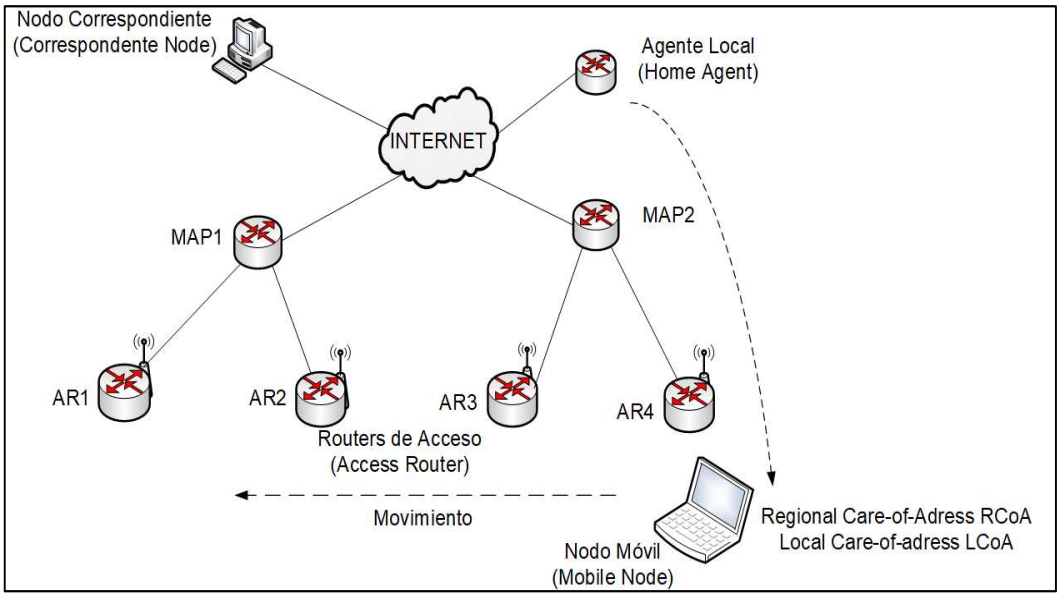


Figura 7. Arquitectura de una red HMIPv6 [139].

Si el nodo móvil cambia su dirección actual dentro de un dominio MAP (LCoA), éste solamente necesita registrar la nueva dirección con el MAP. La dirección CoA regional (RCoA), se registra con los nodos correspondientes y el agente local, HA. La RCoA cambia solo cuando el nodo móvil cambia de dominio MAP. En la figura 7, se puede apreciar la arquitectura de una red IPv6 móvil jerárquica.

La tunelización IPv6 es una técnica para establecer un enlace entre dos nodos IPv6 por medio de encapsulamiento IPv6 [141]. Los dos nodos juegan un papel específico, un nodo tuneliza los paquetes recibidos (encapsulamiento IPv6) y el otro nodo desencapsula los paquetes y los envía a su destino. El nodo encapsulador es llamado "punto de entrada al túnel", y es la fuente de los paquetes tunelizados. El nodo desencapsulador es llamado "punto de salida del túnel", y este es el destino de los paquetes tunelizados.

2.6. ETIQUETA DE FLUJO IPV6

2.6.1 Historia del Campo Etiqueta de Flujo IPV6

El campo "Etiqueta de Flujo" es un campo nuevo que fue adicionado a la cabecera IPv6 y es el foco de la propuesta descrita en esta tesis. El campo "Etiqueta de Flujo" no ha sido muy utilizado y ha sido tema de investigación desde su creación, buscando soluciones para darle un uso apropiado. Desde la creación de este campo, la IETF (*Intenet Engineering Task Force*) ha publicado varias recomendaciones y especificaciones que dan luces sobre la finalidad con la que fue creado, su actualización y evolución, las cuales se describen a continuación.

De acuerdo a [5], la primera apreciación sobre la necesidad de un campo en la cabecera IP para indicar cuando un paquete pertenece a un flujo y que además permitiese el envío de paquetes salto por salto, se dió en la RFC1707 [142], CATNIP (*Common Architecture for Next Generation Internet Protocol*). A este concepto se le conoce como el precursor de la etiqueta MPLS RFC3031 [12]. Luego la IETF desarrolló una propuesta conocida como SIPP (Simple Internet Protocol Plus) RFC1710 [143], la cual incluyó etiquetamiento de paquetes pertenecientes a flujos de un tráfico particular. Finalmente en la RFC1752 [7], quedó la elección final y especificación inicial del campo Etiqueta de Flujo dentro del diseño de IPv6.

El campo "Etiqueta de Flujo " inicialmente fue especificado con una longitud de 28 bits, luego mediante la RFC1883[11], se redujo a 24 bits y finalmente fue reducida a 20 bits con la RFC2460 [8]. Desde la definición de este campo se presentaron varias incertidumbres acerca de su uso, tanto que, en la RFC2460 [8] es definido como experimental y sujeto a cambios. La IETF realizó varios trabajos preliminares a su especificación completa tales como en [144], [145], [146], [147], hasta que se publicó una especificación más detallada en la

RFC3697 [148], la cual, proporcionaba información útil para su uso y estuvo vigente por siete años aproximadamente.

Durante este tiempo varias soluciones fueron publicadas por investigadores proponiendo metodologías de su uso para diferentes propósitos tales como: soporte de calidad de servicio, conmutación de paquetes, filtrado de paquetes entre otros, pero estas soluciones violaban de alguna manera las recomendaciones de dadas por la RFC3697 [148].

Debido a lo anterior, el grupo de trabajo de la IETF hace un estudio de casos de uso del campo "Etiqueta de Flujo " publicado en la RFC6294 [5] y el análisis de las incompatibilidades de varias propuestas para el uso de la etiqueta con lo especificado en la RFC3697 [148]. Dicho análisis fue publicado en la RFC6436 [4], y en esta última, se recomienda la actualización de la RFC3697 [148]. Como consecuencia de esto, es publicada la nueva especificación de la Etiqueta de Flujo mediante la RFC6437[149], quedando como obsoleta la RFC3697. En la tabla 8, se muestra el resumen de las especificaciones de la Etiqueta de Flujo.

Tabla 8. Evolución de las especificaciones del campo Etiqueta de Flujo IPv6.

RFC	Año	Descripción
RFC1707 [142]	*1994*	*Propuso utilizar un campo de cabecera compartida por todos los paquetes pertenecientes a un flujo.*
RFC1710 [143]	*1994*	*Valor de la Etiqueta de Flujo de 28 bits.*
RFC1752 [7]	*1995*	*Etiquetamiento de paquetes pertenecientes a flujos de tráfico particular para cual el emisor solicita un manejo especial.*
RFC1883 [11]	*1995*	*Reducción de la Etiqueta de Flujo a 24 bits.*
RFC2460 [8]	*1998*	*Estandarización de la Etiqueta de Flujo a 20bits.*
RFC3697 [148]	*2005*	*Especificación del campo Etiqueta de Flujo.*
RFC6294 [5]	*2011*	*Estudio de casos de uso del campo Etiqueta de Flujo.*
RFC6436 [4]	*2011*	*Racionalización para actualizar la especificación del campo Etiqueta de Flujo.*
RFC6437 [149]	*2011*	*Especificación actual de la Etiqueta de Flujo.*

2.6.2 Especificación del Campo Etiqueta de Flujo IPV6 RFC6437

El campo de Etiqueta de Flujo de la cabecera IPv6, con una longitud de 20 bits, fue definido para etiquetar paquetes pertenecientes a un mismo flujo. De acuerdo a [148], [149], desde el punto de vista de capa de red, un flujo se define como una secuencia de paquetes desde una fuente a un destino (*Unicast, anycast, multicast*). Desde el punto de vista de capa superior, un flujo puede consistir en todos los paquetes en una conexión de transporte específica (también conocida en inglés como *media stream*).

Para la clasificación de los paquetes, la especificación en RFC6437 [149], recomienda el uso de la tri-tupla (Direcciones fuente, destino y la Etiqueta de Flujo); esto es porque

tradicionalmente la clasificación se hacía sobre la quíntupla (Direcciones fuente y destino, puertos fuente, destino y tipo de protocolo), pero algunos de estos datos pueden no estar disponibles debido a fragmentación o cifrado o la localización de ellos puede ser ineficiente debido al posible paso por varias cabeceras de opción IPv6.

La RFC6437 [149] se centra en el uso de la Etiqueta de Flujo en un escenario *stateless* (sin estado), aunque enuncia que también puede ser usada en un escenario *stateful* (con estado), a diferencia de la RFC3697 [148], que sugería su uso solamente en el escenario *stateful*. Un escenario *Stateless* se refiere a que cualquier nodo que procesa la Etiqueta de Flujo en cualquier forma no necesita almacenar alguna información acerca de un flujo antes o después de que el paquete haya sido procesado [149]. En un escenario *stateful*, un nodo que procesa la información del valor de la Etiqueta de Flujo necesita almacenar información acerca del flujo, lo cual puede incluir el valor de la Etiqueta de Flujo. Un escenario *stateful* podría también requerir un mecanismo de señalización para informar a los nodos *downstream* (nodos destino) que la Etiqueta de Flujo está siendo usada en una cierta forma y para establecer estado del flujo en la red. Algunos protocolos de señalización usados para este propósito son RSVP RFC2205 [17] y GIST (*General Internet Signaling Transport*) RFC5971 [150].

La RFC6437 [149] aconseja que los valores del campo Etiqueta de Flujo sean escogidos de manera que sus bits exhiban un grado de variabilidad, y que además, sean adecuados para usar como parte de una entrada a una función hash usada en un esquema de distribución de carga, e igualmente que provea seguridad, de forma que sea poco probable que terceras personas adivinen el próximo valor que una fuente de etiquetas de flujo podría elegir. De acuerdo con lo anterior, se recomienda que se use una distribución uniforme discreta.

La inmutabilidad de las etiquetas es también una de las restricciones dadas por la RFC6437 [149], igual que en la especificación anterior la RFC3697 [148]. Por tanto, una Etiqueta de Flujo con valor cero indica que el paquete no pertenece a ningún flujo. La inmutabilidad implica que la Etiqueta de Flujo debe ser entregada sin cambios al destino. Se permiten cambios por parte del nodo emisor solo por razones de seguridad operacional.

La RFC6437 [149] también establece que los nodos de envío tales como enrutadores y distribuidores de carga (*Load Balancers*) no deben depender solamente de los valores de Etiqueta de Flujo distribuidos uniformemente. En algunos casos de uso, tales como un clave hash para distribución de carga, los bits de la Etiqueta de Flujo deben ser combinados con bits desde otras fuentes dentro del paquete, para generar un valor hash constante para cada flujo y una distribución adecuada de valores hash a través de los flujos. Los otros campos usados podrían ser algunos o todos los componentes de la usual 5-tupla.

2.6.3 Propuestas relacionadas al uso de La Etiqueta de Flujo

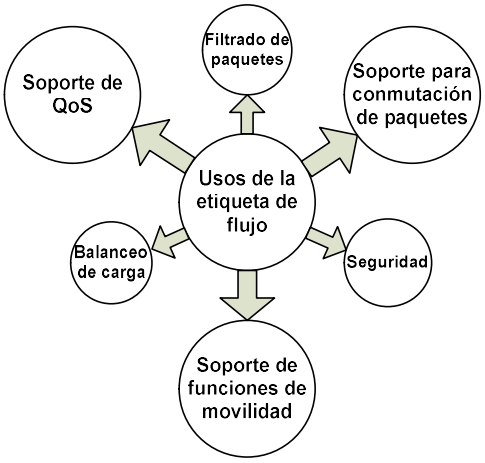


Figura 8. Casos de uso de la Etiqueta de Flujo IPv6.

En esta sección se da una breve descripción de las soluciones propuestas para el uso de la Etiqueta de Flujo para diferentes propósitos. La mayoría de las propuestas encontradas se inclinan a usar la Etiqueta de Flujo para soportar QoS, algunas se enfocan en usarla para soportar conmutación de paquetes, otras usan la etiqueta para contribuir en funciones de movilidad, mientras que otros usos se dan para balanceo de carga, filtrado de paquetes y seguridad (Ver figura 8).

2.6.3.1 Propuestas de uso de la Etiqueta de Flujo para el soporte de Calidad de Servicio (QoS).

En lo que respecta al soporte de calidad de servicio, en [151] se propone un mecanismo para aprovisionamiento de QoS end-to-end mediante la utilización de la 3-tupla en vez de la 5-tupla en la cabecera IPv6, usando el campo de la Etiqueta de Flujo y el de Clase de tráfico para reservar recursos y alcanzar provisión de QoS personalizada. Se especifica un valor pseudo-aleatorio de 17 bits para identificar únicamente un flujo dado. La propuesta es simulada en simulador NS-2, mostrando que el mecanismo se mantiene operando eficientemente durante la congestión de red. Los autores exponen la estructura de la Etiqueta de Flujo dividida en tres partes que son: LF que es el bit de Flag de Etiqueta, LT que significa el Tipo de Etiqueta y LN el número de etiqueta generado aleatoriamente por el nodo fuente.

En [152] se propone una aproximación para utilizar eficientemente el campo de Etiqueta de Flujo para indicar requerimientos de QoS en comunicaciones de voz, datos y multimedia. Esta propuesta usa los primeros bits del campo de Etiqueta de Flujo como códigos para soportar diferentes aproximaciones. Permite una opción pseudo-aleatoria, pero también adiciona opciones para una solicitud directa de QoS y para Diffserv. En la opción pseudo-

aleatoria los dos primeros bits son 01 y los 18 bits restantes son ocupados por un número pseudo-aleatorio entre 0 y 3FFFF. Para la especificación de los parámetros directos de QoS, se usan 18 bits para codificar requerimientos de retardo de una vía, variación de retardo IP, ancho de banda, y pérdidas de paquetes en una vía. Y para la opción de soporte a *Diffserv* se usan los primeros cuatro bits en 1100 y los 16 restantes para indicar el cógido *Diffserv* PHB-ID (*Differentiated Services Per-Hop-Behavior Identification Code*).

Por otro lado, en [153] divide el campo de Etiqueta de Flujo en cinco partes, con los primeros 3 bits usados para diferenciar el tipo de uso (Número aleatorio ó híbrido). Si los primeros 3 bits son 001, la Etiqueta de Flujo será usada como un identificador aleatorio del flujo, pero si es igual a 101, es de tipo híbrido, los bits que quedan en la Etiqueta de Flujo incluirán un requerimiento de QoS híbrido para ese paquete, el cual es subdividido en: tipo de tráfico (1 riguroso, o 0 *best effort*), ancho de banda, uso del buffer, y requerimientos de retardo. Esta propuesta usa la Etiqueta de Flujo para indicar requerimientos de QoS de Intserv y para clasificar tráfico dentro de caminos conmutados de etiquetas con *Diffserv*. Los paquetes con una Etiqueta de Flujo aleatoria son mapeados dentro de un camino virtual genérico (*best effort*).

La propuesta expuesta en [154], usa la Etiqueta de Flujo como un remplazo para el campo Clase de tráfico; esta propuesta sugiere que la Etiqueta de Flujo entrante pueda indicar los requerimientos de calidad de servicio mediante *matching* (un tipo de búsqueda en las tablas de enrutamiento) a un clasificador *Diffserv*. Los autores usaron los primeros tres bits para indicar esto, y los siguientes 16 bits para indicar un código de identificación de comportamiento por salto de servicios diferenciados (Diffserv PHB-ID) RFC3140 [155]; mientras que el último bit está reservado para un futuro uso.

Los autores en [156], [157], proponen un método para escoger un camino más óptimo que garantice requerimientos de calidad de servicio en aplicaciones multimedia, usando la Etiqueta de Flujo basado en redes overlay.

Por otro lado, en [158] los autores proponen una nueva aproximación para soportar calidad de servicio sobre Internet para redes fijas e inalámbricas. Esta aproximación es denominada Intserv6, está basada sobre la arquitectura de servicios integrados, ISA [159] y usa el campo de Etiqueta de Flujo para mejorar un conjunto de propiedades del estándar ISA, tales como reserva de recursos dentro de túneles, agregación de flujos e interconexión con redes de transporte con MPLS.

2.6.3.2 Propuestas de uso de la Etiqueta de Flujo IPv6 para el soporte de conmutación de paquetes

En lo que respecta a conmutación de paquetes, en [160] y [161] se describe una propuesta llamada "*IPv6 Label switching Arquitecture*" (6LSA). En 6LSA, los componentes de red

identifican un flujo revisando el campo de la Etiqueta de Flujo en la cabecera del paquete IPv6; todos los paquetes con la misma Etiqueta de Flujo deben recibir el mismo tratamiento y ser enviados al mismo salto. Sin embargo, 6LSA se asemeja a MPLS al considerar que una etiqueta solamente tiene significado entre routers 6LSA y establece la Etiqueta de Flujo en cada salto. A diferencia de las técnicas de enrutamiento tradicionales, pero de manera similar a MPLS, los paquetes 6LSA son clasificados dentro de un FEC, y los routers envían paquetes sobre diferentes caminos mediante la revisión del FEC.

En [162], se propone mezclar la Etiqueta de Flujo como un tag de conmutación como en MPLS con Diffserv. La propuesta usa un bit en el punto de código de Diffserv RFC2474 [10] para indicar que la Etiqueta de Flujo es un tag de conmutación.

Otra propuesta similar para conmutación basada en calidad de servicio de paquetes IPv6 [163] es diseñada para usar una opción salto por salto, la cual no fue aprobada por la IETF.

En [164], se propone un nuevo modelo de envío por etiqueta con el fin de mejorar la conmutación rápida de paquetes IPv6 que requieren diferenciación de servicios y de proporcionar funciones más eficaces que MPLS (*Multiprotocol Label Switching*) [12].

2.6.3.3 Propuestas de uso de la Etiqueta de Flujo IPv6 para soportar funciones de movilidad

En cuanto a movilidad, en [165] se presenta un esquema de enrutamiento basado en flujo en redes móviles IPv6 para soportar tráfico en tiempo real usando la Etiqueta de Flujo , y en [166] los autores proponen un esquema de QoS basado en la Etiqueta de Flujo para redes móviles. El esquema propuesto utiliza la Etiqueta de Flujo para identificar flujos y servicios. Por otro lado, en [167] se propone un protocolo de reserva de recursos para proveer servicios en tiempo real para usuarios móviles, en esta propuesta la Etiqueta de Flujo es usada para llevar la etiqueta MPLS, pero su valor original es restaurado en el nodo de egreso con el fin de guardar un mecanismo transparente desde el punto de vista de usuario.

En [168], se propone usar la Etiqueta de Flujo para remplazar el ID de flujo del enrutamiento fuente (DSR, *Dynamic Source Routing*) para redes inalámbricas Ad hoc, incluyendo provisión de QoS en dicho mecanismo de enrutamiento.

2.6.3.4 Propuestas de uso de la Etiqueta de Flujo IPv6: Identificación de túnel IPv4-in-IPv6, Balanceo de carga, Filtrado de paquetes y seguridad.

Entre otros usos, [169] propone un caso de uso mediante el cual, ciertos flujos encapsulados en un tipo particular de túnel IPv4-in-IPv6, serían distinguidos en el terminal remoto del túnel mediante un valor de Etiqueta de Flujo específico. Esto permite a un proveedor de servicio entregar una calidad de servicio a la medida.

En [170], se describe cómo las restricciones sobre el uso de la Etiqueta de Flujo se aplican cuando esta se utiliza para balanceo de carga mediante enrutamiento multicamino de igual costo (ECMP-*Equal Cost Multi Path Routing* [171]) y para agregación de enlace, particularmente para tráfico tunelizado IP-in-IPv6.

Finalmente, en [172] se propone un metodología de uso de la etiqueta para prevenir ataques de *spoofing*, estableciendo un valor aleatorio en el campo Etiqueta de Flujo para hacer la cabecera del paquete más compleja y menos fácil de descifrar por un atacante. En la tabla 9, se da un resumen de las diferentes propuestas por categorías según el propósito de cada una.

Tabla 9. Resumen de Propuestas para el uso del campo Etiqueta de Flujo IPv6.

Categoría (Propósito)	Propuestas
Soporte de Calidad de servicio.	[151-158]
Conmutación de paquetes.	[160-164]
Movilidad.	[165-168]
Identificación de túnel IPv4-in-IPv6.	[169]
Balanceo de Carga.	[170]
Seguridad.	[172]

2.7 PUBLICACIONES REALIZADAS DE ESTE CAPÍTULO

- Becerra L.; Padilla J.; Paradells J., "HMIPv6-BI: A Proposal to Improve the Bandwidth of the Radio Channel in HMIPv6 Networks," IEEE Latin America Transactions IEEE, pp. 603 – 609.

- L. Y. Becerra and J. J. Padilla, "Estudio de propuestas para soportar Ingeniería de tráfico en Internet," Entre Ciencia e Ingeniería, vol. 6, no. 11, pp. 53–76, 2012.

- L. Y. Becerra, J. L. Bañol, and J. J. Padilla, "Un estudio sobre algoritmos basados en restricciones: objetivos ingeniería de tráfico y calidad de servicio," Entre Ciencia e Ingeniería, vol. 11, no. 21, pp. 103–111, 2017.

- L. Y. Becerra S. and J. J. Padilla A., "Review of Approaches for the use of the Label Flow of IPv6 Header," IEEE Latin America Transactions, vol. 12, no. 8, pp. 1602–1607, 2014.

2.8 CONCLUSIONES

La amplia exploración del estado del arte presentada permitió el diseño y evaluación de la propuesta descrita en esta tesis. En primera instancia el estudio de las diferentes recomendaciones y especificaciones de la IETF concernientes al protocolo IPv6 y al soporte de ingeniería de tráfico en Internet, permite un mejor entendimiento del problema de congestión en Internet y evidencia que dicho problema no ha sido resuelto, que es un tema abierto que permite hacer nuevas propuestas. A su vez, este marco de referencia proporcionó conocimiento clave para la selección de protocolos de enrutamiento y señalización. De acuerdo a tales especificaciones, se seleccionan para el diseño de la propuesta los protocolos de enrutamiento y de señalización a OSPFv3-TE[23] y RSVP-TE[13] respectivamente.

Con la exploración de propuestas publicadas para el soporte de ingeniería de tráfico en Internet, se pudieron establecer cinco categorías, las cuales son foco de investigación. Estas son: las propuestas basadas en IP, basadas en MPLS, basadas en la combinación IP+MPLS, y en nuevas arquitecturas como LISP y enrutamiento por segmentos. La solución descrita en esta tesis tiene similitudes con el funcionamiento de MPLS, por esta razón en la evaluación se hacen comparaciones con MPLS.

Teniendo en cuenta las recomendaciones de la IETF en [1] y [15] de la necesidad de algoritmos basados en restricciones (CBR, *Contraints based Algorithms*) para el soporte de ingeniería de tráfico, se realiza una exploración general de diferentes propuestas alrededor del tema entontrando que estos se pueden categorizar en tres líneas de enfoque, para TE, QoS y TE+QoS. La finalidad de esta exploración fue tener una visión general de los de algoritmos basados en restricciones existentes para TE. Para el diseño de la propuesta se asume el uso de un algoritmo basado en restricciones denominado CSPF (Constrained Shortest Path First)[15][59], por ser este un algoritmo basado en Dijkstra, además de ser uno de los algoritmos pioneros y el que es mencionado en la literatura como el algoritmo CBR comúnmente utilizado con MPLS, tal como se menciona en [45] . Esta tesis no incluye pruebas con algoritmos CBR, ni la creación de estos, ya que es un trabajo que se contempla para el futuro.

La exploración realizada tanto en las epecificaciones de la IETF para el campo de etiqueta de flujo, como de las propuestas de uso de este campo para diferentes propósitos, brindó conocimiento y razones suficientes para apoyar el uso del campo etiqueta de flujo IPv6 para la conmutación de paquetes y el soporte de ingeniería de tráfico en Internet propuesto en esta tesis.

3. DISEÑO DE UNA ARQUITECTURA DE CONMUTACIÓN DE PAQUETES PARA SOPORTAR INGENIERÍA DE TRAFICO EN REDES IPV6 (PSA-TE6)

PSA-TE6 (*Packet Switching Architecture To Support Traffic Engineering In IPV6 Networks*) es una contribución de esta tesis, con el fin de soportar ingeniería de tráfico en redes IPv6, mediante una arquitectura de conmutación de paquetes por etiquetas utilizando el campo de "Etiqueta de Flujo IPv6". La meta de esta arquitectura es utilizar el campo Etiqueta de Flujo IPv6 para conmutación de paquetes en redes IPv6 de una manera parecida a como trabaja MPLS [12], pero sin la necesidad de una arquitectura MPLS instalada. Esta propuesta nace como consecuencia del estudio realizado al campo Etiqueta de Flujo desde su creación hasta las últimas recomendaciones de la IETF, además del análisis de las diferentes propuestas de uso del campo Etiqueta de Flujo [6] y también de la observación de su estructura, la cual muestra una gran similitud con la etiqueta MPLS en lo que respecta al tamaño (20 bits) y su contenido.

3.1 PROPUESTAS RELACIONADAS

En esta sección se describirán 2 propuestas relevantes antecesoras a esta propuesta. Una primera propuesta publicada en el año 2002, denominada "*IP Next Generation Label Switching*" (*IPNGLS*) en [173], propone el envío de paquetes IPv6 mediante la técnica de conmutación por etiquetas con las mismas ventajas que la arquitectura MPLS. Los autores proponen mapear todos los campos de MPLS dentro de la cabecera IPv6 y mencionan que la integración de MPLS e IPv6 decrementa la complejidad ya que se elimina una cabecera extra sobre el sistema, reduce la sobrecarga de cabecera debido a que hay un ahorro de 4 octetos en los paquetes IPv6 cuando no hay apilamiento de etiquetas en comparación con MPLS.

La segunda propuesta ya explicada en la sección 2.6.3.2, es "*IPv6 Label Switching Arquitecture*" (*6LSA*) descrita en [161] y [160], propone el uso de la etiqueta de flujo para conmutación de paquetes de una forma similar a MPLS. Una crítica importante mencionada en la RFC6294 [5] de 6LSA, es que si la fuente original establece una Etiqueta de Flujo diferente de cero no hay mecanismo para restaurarla, lo cual incumplía la especificación actual en su momento que era la RFC3697 [148]. También se menciona en la RFC6294 [5] que los autores hicieron una etapa de discusión del uso de la opción salto por salto para corregir este problema pero no fue documentado. Otra crítica fue el hecho que esta propuesta divide el campo de Etiqueta de Flujo en tres partes. Los primeros tres bits identifican el FEC, el cual ayudará al router o a nodos 6LSA a agrupar los paquetes IP que reciben el mismo tratamiento de envío y los envía sobre el mismo camino virtual. Los siguientes 4 bits describen el tipo de aplicación, y los 13 bits finales (definido por cada nodo

o un grupo de nodos) definen la etiqueta específica del salto. Esta división del campo de Etiqueta de Flujo también viola la segunda regla de la especificación de la Etiqueta de Flujo, que menciona que el rendimiento del router no debe depender de la distribución de los valores de la Etiqueta de Flujo.

En la tabla 10, se proporciona un resumen de las similitudes y diferencias de las dos aproximaciones antecesoras a la solución propuesta en esta tesis (PSA-TE6), de acuerdo con características importantes para el soporte de ingeniería de tráfico.

Tabla 10. Comparación entre propuestas que usan la Etiqueta de Flujo IPv6 para conmutar paquetes.

Característica	6LSA	IPNGLS	PSA-TE6
Usa el mecanismo de conmutación de paquetes mediante la etiqueta de flujo IPv6	SI	SI	SI
Describe los campos de la tabla de enrutamiento	SI	NO	SI
Usa caminos conmutados por etiquetas como MPLS	SI	SI	SI
Divide el valor de la etiqueta de flujo IPv6 en diferentes campos	Si. Divide el campo etiqueta de flujo en tres partes.	No. Mapea el valor de etiqueta directamente desde MPLS.	No. Usa los 20 bits sin dividirlo. El valor es asignado de una manera similar como en MPLS.
Usa operaciones de etiqueta como: push-swap-pop	SI	Esto no está descrito. Es asumido como en MPLS.	SI
Describe como es generada la etiqueta	Describe tres formas: 1-Basada localmente sobre un cierto algoritmo o política. 2-En el paquete de entrada como una etiqueta de flujo desde el nodo fuente. 3-Distribuida a través de un proceso de distribución de etiqueta.	NO	SI. Esta es distribuida a través de un proceso de distribución de etiqueta en una manera similar como MPLS.
Usa protocolo de distribución de etiquetas	Este contempla la opción de usarlo para el caso 3, pero no asume uno en particular.	Este contempla la opción de usarlo, pero no asume uno en particular.	SI. Describe el uso de RSVP-TE.
Define la relación Etiqueta-FEC en cada router	SI	NO	SI
Define su operación dentro de un dominio y define los elementos que comprende su arquitectura	SI	NO	SI
Permite apilamiento de etiquetas	No, no es permitido.	SI, es permitido mediante una cabecera de opción.	SI, es permitido mediante tunelización de paquete genérica en IPv6 o usando una cabecera de opción.
Usa protocolos extendidos para soportar ingeniería de tráfico	NO	NO	SI
Usa algoritmos basados en restricciones	NO	NO	SI
Define mecanismos de restauración de valor de etiqueta	NO	NO	SI

3.2 PRINCIPIOS DE DISEÑO DE LA PROPUESTA

El diseño de PSA-TE6 comenzó con el estudio de las especificaciones de IETF del campo Etiqueta de Flujo IPv6 [149] y de los casos de uso para diferentes propósitos en [5]. Ambos documentos describen tres reglas básicas para el uso del campo de Etiqueta de Flujo, las cuales se enuncian a continuación.

i. *Los nodos IPv6, no deberán suponer ninguna propiedad matemática u otras propiedades de los valores de la Etiqueta de Flujo asignadas por los nodos fuente* [4], [149].

ii. *El rendimiento del router no debe depender de la distribución de los valores de la Etiqueta de Flujo. Especialmente, los bits de Etiqueta de Flujo no contienen suficiente información para generar una clave hash* [4], [149].

iii. *Etiqueta de Flujo no debe ser cambiada en ruta, pero permite que los routers fijen la etiqueta en nombre de hosts que no lo hacen* [4], [149].

De acuerdo con las anteriores reglas, la propuesta PSA-TE6, violaría la primera parte de la regla (*iii*), es decir, "no debe ser cambiada en ruta". Sin embargo, la segunda parte de la regla (*iii*) enuncia "pero permite a routers establecer la etiqueta en lugar de host que no lo hacen". Esto es un tema abierto que ha sido presentado y discutido en la RFC6294 [5]. Adicionalmente, la RFC6294 [5] mencionó con respecto a la regla (*iii*) que ésta no excluye que la Etiqueta de Flujo sea usada para propósitos de enrutamiento o conmutación de paquetes.

De igual forma, la RFC6294 [5], también hace recomendaciones para el uso de la Etiqueta de Flujo para conmutación de paquetes. Tales recomendaciones se refieren a pasar por alto las reglas dentro de un dominio dado, en el cual los routers podrían establecer e interpretar el campo de Etiqueta de Flujo IPv6 para el propósito que haya sido diseñada y luego en el router del último salto, la etiqueta podría ser establecida a cero. Esta regla debería ser aplicada para los paquetes que llegan al dominio con la etiqueta establecida a cero.

Para el caso en que los paquetes llegan al domino con una etiqueta de valor diferente de cero, una recomendación alternativa suministrada por la RFC6294 [5], es definir una cabecera de opción salto por salto para llevar la etiqueta original a través del dominio, así que la etiqueta pueda ser restaurada al final del dominio. Todas estas recomendaciones fueron tomadas en cuenta en el diseño de la solución propuesta con PSA-TE6.

Por otro lado, los protocolos OSPF y IS-IS fueron extendidos por la IETF en respuesta a los requerimientos de la RFC2702 [15], y estas extensiones también están asociadas con el soporte de ingeniería de tráfico de MPLS (OSPF-TE and IS-IS-TE). Por tanto, en el diseño de PSA-TE6, tales protocolos también juegan un rol importante en el proceso de envío, ya que

PSA-TE6 usa conmutación de etiquetas y su meta es proporcionar ingeniería de tráfico. Así que la propuesta PSA-TE6 incluye el uso del protocolo OSPFv3-TE[23], el cual es el protocolo extendido para trabajar sobre redes IPv6 y para el soporte de ingeniería de tráfico. Adicionalmente, el protocolo de señalización RSVP-TE [13], el cual fue extendido para la distribución de etiquetas de MPLS, ha sido tomado y definido como protocolo de distribución de etiquetas de flujo IPv6 en PSA-TE6.

Finalmente, en PSA-TE6 es necesario encontrar caminos basados en restricciones. Esta funcionalidad es proporcionada por algoritmos CBR (*Contraint-Based Routing*), los cuales seleccionan la mejor ruta que corresponde a un conjunto de restricciones. Las restricciones pueden ser impuestas por políticas administrativas, requerimientos de calidad de servicio o de ingeniería de tráfico [18].

En las últimas décadas, muchos algoritmos han sido propuestos y en [94], [53], and [108], se presenta un estudio de esas propuestas. Para PSA-TE6 se ha seleccionado el algoritmo CSPF (*Constraints Shortest Path First*) debido a que es uno de los algoritmos más comúnmente usados [18]

3.3 ARQUITECTURA DE PSA-TE6

La arquitectura PSA-TE6 está compuesta de los siguientes elementos (ver figura 9):

- *Routers de ingreso y egreso* al dominio PSA-TE6, los cuales se han denominado 6DER (*Ingress/Egress PSA-TE6 Domain Edge Router*).
- *Routers de tránsito*, denominados 6DTR (*PSA-TE6 Domain Transit Router*).
- *Caminos conmutados por etiquetas* denominados 6DLSP (*PSA-TE6 Domain Label Switching Paths*).

Los elementos de la arquitectura operan bajo un dominio PSA-TE6 (*PSA-TE6 Domain*), el cual ha sido denominado 6D. La idea de tener un dominio con PSA-TE6 es recomendada en la RFC6294 [5], en la sección 4. Tal como se mencionó en la sección 3.2 de esta tesis, la recomendación dada por la RFC6294 es para propuestas que usan la Etiqueta de Flujo IPv6 para conmutación de paquetes.

Para explicar la propuesta PSA-TE6, se asume una red de 5 nodos como en la figura 9. La red trabaja bajo el protocolo IPv6, y está compuesta de un dominio 6D con protocolos de enrutamiento y señalización OSPFv3 [23] and RSVP-TE [13], respectivamente. OSPFv3-TE [23] es el responsable de inundar la información de enrutamiento en correspondencia a la topología de red y a la información de ingeniería de tráfico. Mientras que RSPV-TE es

responsable del establecimiento de caminos conmutados por etiquetas, 6DLSPs, y de la distribución de valores de etiquetas de flujo IPv6.

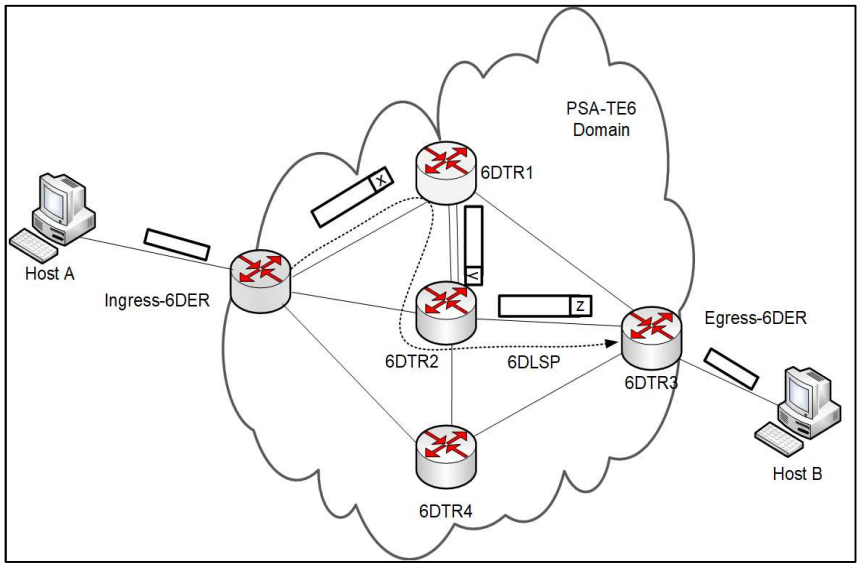


Figura 9. Arquitectura PSA-TE6.

Es importante enfatizar que el protocolo RSVP-TE tiene un objeto denominado "Label", el cual fue creado para distribuir etiquetas MPLS. Sin embargo, en la solución PSA-TE6, se usa el objeto "Label" de RSVP-TE para distribuir etiquetas de flujo IPv6. Esta designación puede ser ejecutada porque las etiquetas MPLS y el campo Etiqueta de Flujo IPv6 tienen la misma longitud (20 bits). Además, para establecer caminos conmutados por etiquetas IPv6, es necesario encontrar el camino más apropiado, no necesariamente el más corto, sino uno basado en restricciones. Varios algoritmos han sido propuestos para este propósito, pero en esta propuesta, se asume que PSA-TE6 trabaja con el algoritmo CSPF (*Constrained Shortest Path First*), el cual es uno de los más usados. CSPF está basado en el algoritmo Dijkstra con la adición de una restricción de ancho de banda, este algoritmo es explicado en [18].

En el dominio 6D los routers de ingreso y/o egreso deben ser capaces de leer la cabecera IPv6, establecer los caminos conmutados por etiquetas, 6DLSPs mediante RSVP-TE [13] y desarrollar la inserción y borrado de etiquetas (operaciones push y pop).

Los routers tránsito 6DTRs deben ser capaces de leer los primeros 64 bits de la cabecera IPv6 (ver figura 10) para conmutar y enrutar los paquetes a través de intercambios de etiquetas de flujo IPv6, y ellos deben ser capaces de realizar las operaciones necesarias para el apilamiento de etiquetas IPv6. El hecho de que los routers 6DTRs solo tengan que leer los

primeros 64 bits es porque los campos DS (Servicios Diferenciados), Etiqueta de Flujo IPv6, y límite de saltos de la cabecera del paquete están localizados en esos primeros 64 bits y deben ser conocidos para el soporte de calidad de servicio y conmutación de paquetes por etiquetas. Para esto, los routers solo deben hacer una operación de lectura teniendo en cuenta el uso de procesadores actuales de 64bits.

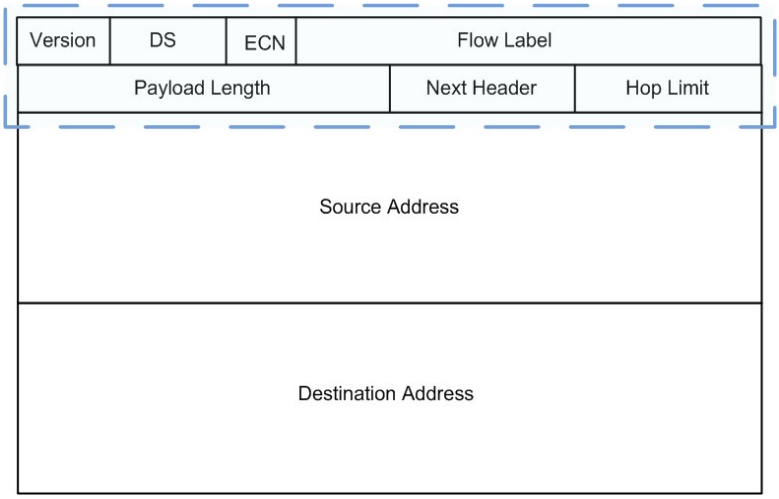


Figura 10. Campos de la Cabecera IPv6 para la conmutación mediante Etiquetas de Flujo IPv6 [2].

3.4 PROCESO PARA EL ESTABLECIMIENTO DE CAMINOS CONMUTADOS POR ETIQUETAS DE FLUJO IPV6

Cuando se establece una nueva comunicación y el primer paquete alcanza un 6DER de ingreso, este lee la cabecera IPv6 y captura la dirección fuente y destino. Luego, por medio de RSVP-TE, se realiza los procesos de distribución de valores de Etiqueta de Flujo IPv6 y el establecimiento del camino conmutado por etiquetas, este proceso es llevado a cabo de acuerdo con los procedimientos estándar del protocolo. El camino conmutado por etiquetas es encontrado por el algoritmo de enrutamiento basado en restricciones apoyado en la información de OSPFv3-TE. Luego, el 6DER de ingreso pone el valor de etiqueta en el campo de Etiqueta de Flujo en todos los paquetes pertenecientes al correspondiente flujo y los envía al próximo salto.

Esta propuesta inicialmente asume que los paquetes vienen de un dominio que no usa la Etiqueta de Flujo IPv6, así que este valor sería cero de acuerdo a la RFC6437 [149]. Luego, el paquete viaja sobre ese camino, y cada router interior, 6DTR, intercambia la etiqueta en cada paquete y luego envía el paquete a la interfaz de salida apropiada. Cuando el paquete llega al router de egreso, 6DER, este remueve la etiqueta y envía el paquete al destino. El router también retorna la Etiqueta de Flujo a su valor original (o cero) de acuerdo a RFC6294 [5].

3.5 BASES DE INFORMACIÓN REQUERIDAS EN PSA-TE6

En el proceso de envío, es necesario tener información de las operaciones desarrolladas sobre el campo de etiqueta flujo IPv6, la cual debe ser analizada antes del envío de paquetes al próximo salto. Para este propósito, los routers tienen un FIB (*Forwarding Information Base*) que es específica para cada router dependiendo de si es un 6DER o un 6DTR. Esta FIB puede ser de dos tipos: una que mapea una clase equivalente de envío o FEC (*Forwarding Equivalent Class*) a los datos de envío de Etiqueta de Flujo IPv6 de próximo salto o N6FLD (*Next Hop IPv6 Flow Label Forwarding Data*), la cual es una tabla denominada FTN6 (*Forwarding Equivalence Class To Next Hop IPv6 Flow Label Forwarding Data*). La otra FIB mencionada mapea una Etiqueta de Flujo IPv6 entrante al N6FLD y se ha denominado I6LTN (*Incoming IPv6 flow label to next Hop IPv6 Flow Label Forwarding Data*). Estas tablas son similares en contenido y función a las usadas por MPLS [12]. Estas tablas estarán disponibles en routers de conmutación de etiquetas de flujo IPv6 de acuerdo con sus roles en el dominio PSA-TE6 (ver tablas 11 y 12)

Tabla 11. Información de la tabla FTN6

Interfaz Entrante	*Dirección IPv6 Destino*	*Interfaz Saliente*	*Etiqueta Saliente*	*Dirección IPv6 de próximo salto*
#3	*2001:DB8:0::800:200C:417A*	*#5*	*10*	*2001:1:1::/64*

Interfaz Entrante	*Etiqueta Entrante*	*Dirección IPv6 destino*	*Interfaz Saliente*	*Etiqueta Saliente*	*Operación*
#5	*10*	*2001:DB8:0::800:200C:417A*	*#8*	*30*	*swap*

Tabla 12. Información de la tabla I6LTN

En redes IP, un router considera que dos paquetes pertenecen al mismo FEC si hay un prefijo de dirección X en la tabla de enrutamiento, tal que el prefijo X sea la coincidencia más larga (longest match) para cada dirección destino del paquete. En la arquitectura PSA-TE6, el FEC será determinado en los routers de ingreso donde los 6DLSPs son establecidos.

3.6 PROPUESTAS PARA EL APILAMIENTO DE ETIQUETAS DE FLUJO IPV6

En PSA-TE6, es necesario explorar mecanismos para apilar etiquetas de flujo IPv6 como una técnica para mejorar la gestión del espacio de etiqueta limitado, considerando que el

tamaño de la Etiqueta de Flujo IPv6 es solamente de 20 bits. En MPLS, se presenta el mismo problema, y la solución ha sido estudiada y evaluada mediante la creación de túneles [174].

Por tanto, en PSA-TE6, se propone soluciones para la creación de túneles para adicionar tráfico de diferentes 6DLSPs en segmentos comunes de la red con dos o más routers internos. Para hacer el proceso de apilamiento de etiquetas, se proponen dos mecanismos: el primero es el de tunelización IPv6 mediante RFC2473 [141] (*Generic Packet Tunneling in IPv6*), (ver figura 11) y el segundo método es mediante el uso de una cabecera de opción similar a lo descrito en [173].

3.6.1 Apilamiento de etiquetas mediante tunelización IPv6 usando GPT

Esta es una técnica para establecer un camino virtual entre dos nodos IPv6 por medio de encapsulamiento IPv6 y es muy utilizado en diferentes escenarios como por ejemplo en redes móviles IP. En este método, para poder manejar el apilamiento de etiqueta, y que los routers entiendan que hay apilamiento de etiquetas, se usará el campo *Next Header* de IPv6, donde se indica que la próxima cabecera es otra cabecera IPv6 (*Next Header* = 41, para tunelización IPv6 in IPv6) [2]. Con este mecanismo de tunelización, el proceso de conmutación de paquetes mediante la Etiqueta de Flujo puede continuar en la cabecera más externa del túnel IPv6, preservando la velocidad de conmutación y el mismo proceso de envío [141]. La encapsulación IPv6 consiste en anteponer al paquete original una cabecera IPv6 y opcionalmente, un conjunto de cabeceras de extensión, las cuales son colectivamente llamadas cabeceras IPv6 de túnel. La encapsulación ocurre en un punto de entrada del túnel IPv6 como resultado de enviar un paquete original sobre un enlace virtual representado por el túnel (ver figura 11).

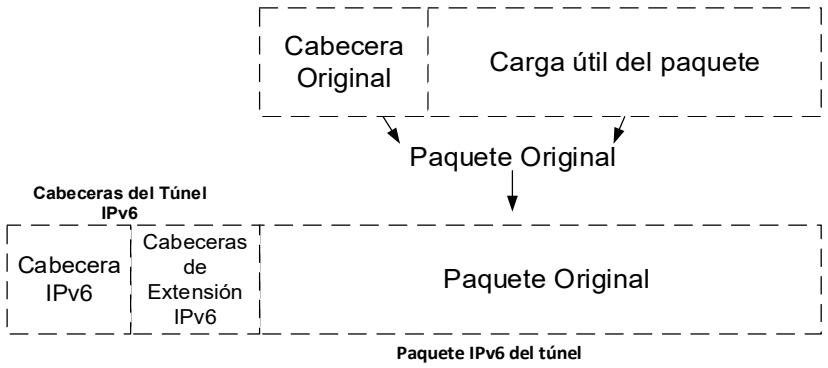


Figura 11. Tunelización IPv6 mediante la RFC2473 [141].

3.6.2 Apilamiento de etiquetas mediante el uso de una cabecera de opción

Este método consiste en hacer uso del soporte de cabeceras de opción de IPv6 [2]. La cabecera de opción que se ajusta para este propósito es la cabecera de opción salto por salto (hop-by-hop, Next header=00) [2]. En IPv6 la información opcional de la capa de Internet es codificada en cabeceras separadas que deben ser ubicadas entre la cabecera IPv6 y la cabecera de capa superior en un paquete. El paquete IPv6 puede llevar ninguna, una o varias cabeceras de extensión, cada una identificada en el campo de Next header de la cabecera precedente. Tal como se especifica en la RFC8200 [2], las cabeceras de extensión no son examinadas por ningún nodo a lo largo de la ruta, a excepción de la cabecera salto por salto que sí puede ser procesada y analizada por cada nodo a lo largo del camino del paquete. La cabecera de opción hop-by-hop cuando está presente debe seguir inmediatamente la cabecera IPv6. Cada cabecera de extensión es un múltiplo entero de 8-octetos.

El proceso de utilizar la cabecera de opción hop-by-hop para crear túneles es el siguiente: cuando un paquete llega al router de ingreso de un túnel, este router crea una cabecera de extensión hop-by-hop, en la cual va a guardar el valor de etiqueta con la cual viene el paquete y pone el nuevo de valor de etiqueta del túnel, en el campo de Etiqueta de Flujo de la cabecera IPv6, que sería la etiqueta más externa y con la cual se debe hacer la conmutación hacía el siguiente router basado en la información de la FIB. Esto permite seguir haciendo conmutación por etiquetas sobre estos primeros 64 bits de la cabecera IPv6 de la arquitectura PSA-TE6, manteniendo la misma velocidad de envío, como cuando no hay apilamiento a lo largo del camino. Una propuesta similar fue inicialmente descrita en [173]. La figura 12, muestra el formato de la cabecera de extensión hop-by-hop usado para el apilamiento de etiquetas de la arquitectura PSA-TE6.

Next Header	Hdr Ext Len=1	Option Type=X	Opt Data Len=4
Etiqueta de Flujo Original			Uso Futuro
Etiqueta de Flujo para L=2			Uso Futuro
Etiqueta de Flujo para L=3			Uso Futuro
Etiqueta de Flujo para L=4			Uso Futuro

Figura 12. Cabecera de extensión hop-by-hop con etiquetas de flujo apiladas para la arquitectura PSA-TE6.

En la cabecera de extensión hop-by-hop original se encuentran los siguientes campos: El campo Next Header es de 8 bits e identifica el tipo de próxima cabecera, la cual puede ser otra cabecera de extensión o una cabecera de capa superior. El campo *Hdr Ext Len*, es de 8 bits e identifica la longitud de la cabecera de extensión hop by hop en palabras de 8 octetos, excluyendo los primeros 64 bits. El campo de opciones es de longitud variable, contiene uno

o más TLVs (*type, length* and *value* de las opciones) codificados, puede ser llenado hasta de 2040 octetos de datos de opción, es decir la longitud máxima Hdr Ext Len igual a 255 en unidades de 8 octetos [2].

3.7 PROPUESTA DE INTEGRACIÓN DE IPV6 MÓVIL JERÁRQUICO CON PSA-TE6

IPv6 móvil Jerárquico, (HMIPv6, *Hierarchical Mobile IPv6*) es una extensión del protocolo IPv6 móvil, (MIPv6, Mobile IPv6) [138], cuyo objetivo es resolver los problemas de micromovilidad reduciendo la carga de señalización y por tanto, la latencia del handover y la pérdida de paquetes. La IETF especifica este protocolo para IPv6 mediante la RFC5380 [139]. Como se explicó en la sección 2.5.1, HMIPv6 permite una estructura de red jerárquica que facilita la gestión de movilidad a través de la introducción de un nuevo nodo llamado MAP (*Mobility Anchor Point*). El MAP es responsable de la gestión de movilidad del nodo móvil (MN*, Mobile Node*). Por tanto, cada paquete enviado desde el nodo correspondiente (CN, *Correspondent Node*) al MN es interceptado por el MAP y tunelizado a la nueva ubicación del MN, identificado con la nLCoA (*New Local Care-of Address*) del MN. De igual forma, si el MN envía información hacia el CN, los paquetes son tunelizados desde el MN hacia el MAP. El MAP es responsable de remover la cabecera del túnel y enviar la información original hacia el CN [139] (ver Figura 13).

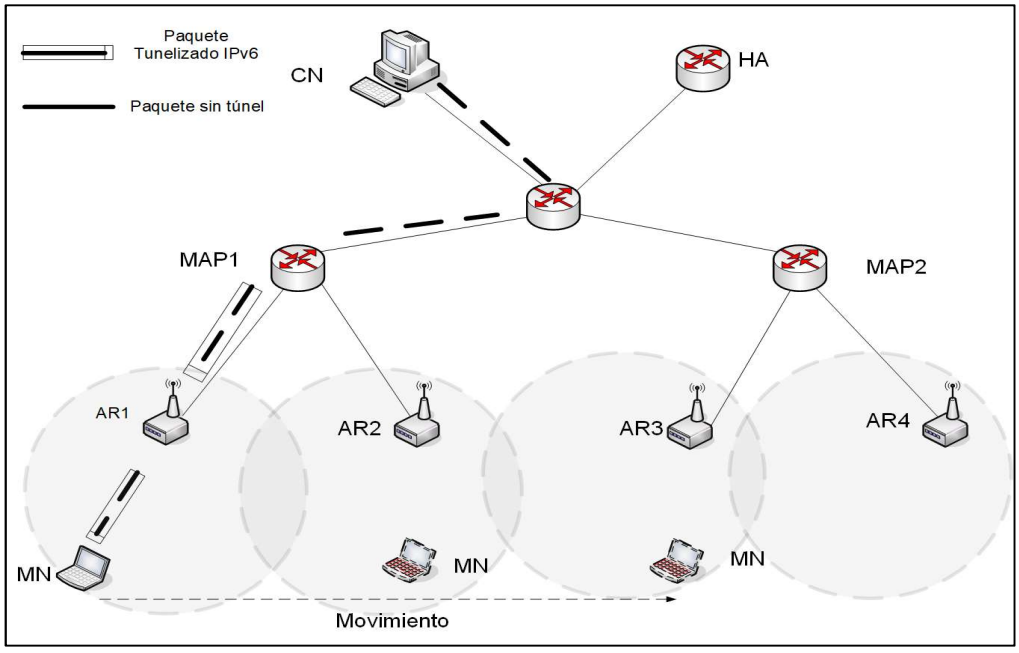


Figura 13. IPv6 móvil Jerárquico [139]

3.7.1 Principios de diseño de la integración de HMIPv6 con PSA-TE6

En la propuesta de integración del protocolo HMIPv6 con PSA-TE6 (HMIPv6+PSA-TE6) se considera que la región MAP es un dominio PSA-TE6 (6D). La arquitectura mostrada en la figura 14, es un ejemplo de esta propuesta de integración. El MAP y los routers de acceso (ARs), son routers de ingreso y egreso (*Ingress/Egress-6DERs, Ingress/Egress PSA-TE6 Damain Edge Routers*) entre el MAP y los routers de acceso (ARs) puede haber varios routers tránsito (6DTR, *PSA-TE6 Domain Transit Routers*). Por otro lado, el MAP conecta con redes externas y se comunica con agentes locales y nodos correspondientes de una manera normal como está establecido en el protocolo HMIPv6.

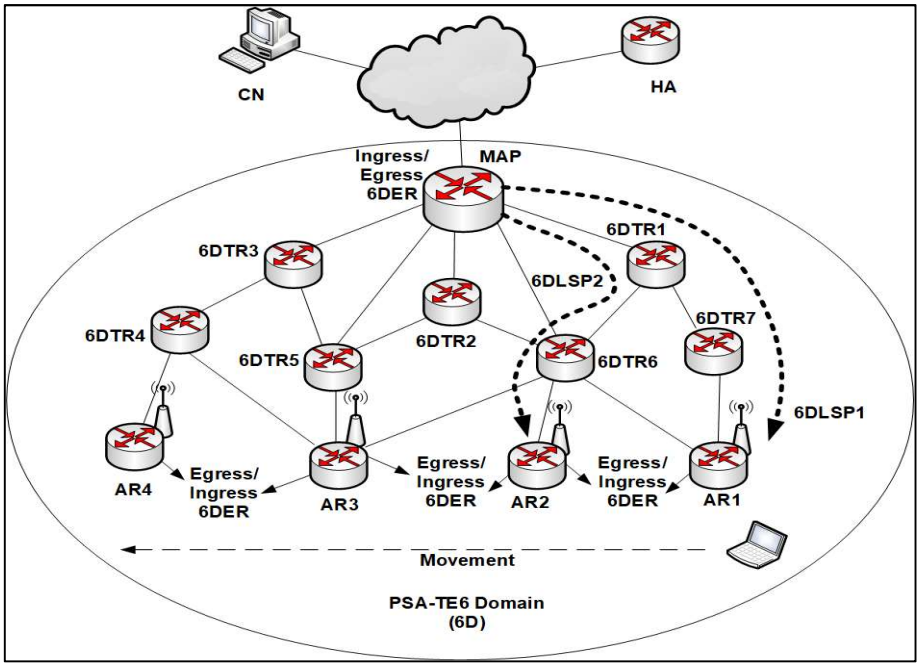


Figura 14. IPv6 móvil Jerárquico en combinación con PSA-TE6.

En redes móviles IPv6 que trabajan con HMIPv6, cada paquete enviado desde el nodo correspondiente (CN) es interceptado por el MAP y es tunelizado a la nueva LCoA (*Local Care of Address*) del nodo móvil (MN). Tal tunelización es desarrollada usando encapsulación IPv6 mediante las especificaciones establecidas en la RFC2473 [141]. Esta tunelización realizada entre el MAP y el MN agrega una cabecera IPv6 de 40 bytes. La solución de integración de HMIPv6 con PSA-TE6 propone establecer caminos conmutados por etiquetas en el dominio PSA-TE6 (6DLSP, *PSA-TE6 Domain Label Switching Paths*), entre el MAP y los ARs en lugar de encapsulación IPv6. Así que cada paquete enviado por el CN hacia el MN es interceptado por el MAP, para ser enviado a ARs mediante 6DLSPs. El MAP desarrolla operaciones de

asignación de etiquetas (push operations), los routers tránsito (6DTRs) intercambian etiquetas (swap operations), y los routers de acceso (ARs) desarrollan operaciones de borrado de etiquetas (pop operations) y envían los paquetes hacia al MN.

Como se mencionó anteriormente, en PSA-TE6, el enrutamiento de información, la selección de caminos y el establecimiento de los caminos conmutados por etiquetas son desarrollados por medio de OSPFv3-TE[23], CSPF [18] y RSVP-TE [13] respectivamente. En la figura 14, se muestra un ejemplo de la arquitectura de la integración HMIPv6+PSA-TE6.

4. EVALUACIÓN DE LA PROPUESTA PSA-TE6

4.1 EVALUACIÓN DEL ESPACIO DE ETIQUETAS DE FLUJO IPV6 EN PSA-TE6

En esta sección se analiza el comportamiento de PSA-TE6 cuando usa apilamiento de etiquetas y cuando no lo usa. Este análisis es necesario e importante porque el campo de Etiqueta de Flujo tiene solamente 20 bits. Por tanto, es requerido ahorrar el espacio de etiquetas. El proceso de apilamiento de etiquetas mediante tunelización es una solución útil para reducir el espacio de etiqueta cuando una red tiene segmentos que son comunes para varias comunicaciones, permitiendo el uso de menos etiquetas en el proceso de envío. En esta evaluación se presentan dos ejemplos, el primero con el fin de mostrar el proceso de intercambio de etiquetas de la propuesta PSA-TE6, para los casos en que no hay apilamiento y cuando el apilamiento es habilitado. El segundo, muestra la abstracción de una porción de red por la cual pasan tres flujos y se describe la conformación de los túneles en los segmentos comunes de los caminos conmutados por etiquetas de flujo IPv6 para los tres flujos y la cantidad de etiquetas que intervienen en el proceso.

Ejemplo 1. En las figuras 15 y 16, se muestra el enrutamiento de caminos conmutados por etiquetas en una red con arquitectura PSA-TE6. En la figura 15, se muestran dos 6DLSP establecidos, uno desde el Ingress-6DER1 al Egress-6DER3 (se observa en línea continua en color azul) y el otro desde el Ingress-6DER2 al Ingress-6DER4 (se observa en línea punteada en color negro), donde ambos 6DLSPs tienen un camino en común formado por los 6DTR1, 6DTR2, 6DTR3 y 6DTR4, en este caso no se usa tunelización. En la figura 16, está descrita una situación similar, usando tunelización.

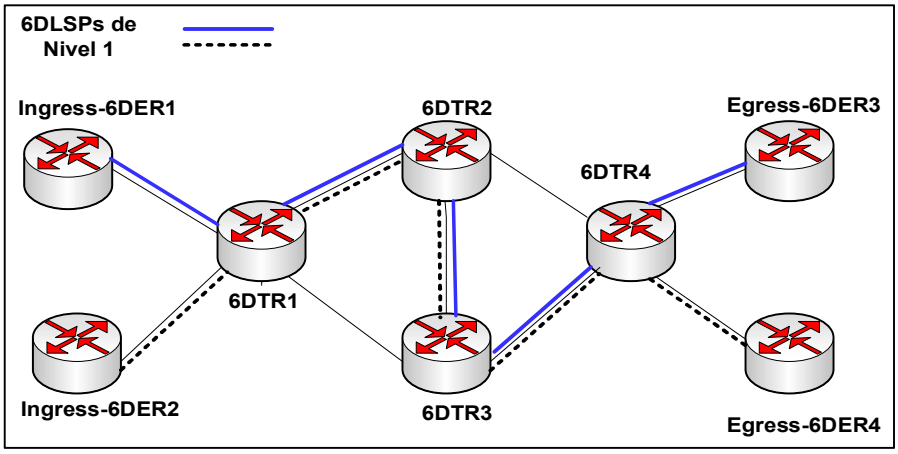


Figura 15. Caminos Conmutados por Etiquetas de Flujo IPv6.

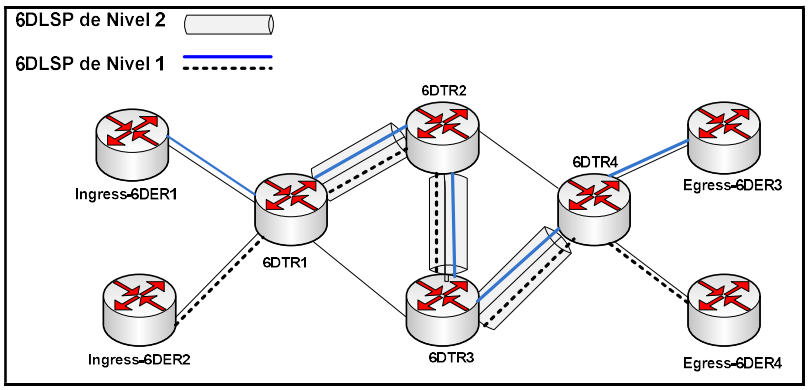


Figura 16. Apilamiento de Etiquetas en PSA-TE6.

En las tablas 13 y 14, se describe el resumen del proceso de enrutamiento mediante conmutación por etiquetas de flujo IPv6 sin usar apilamiento de etiquetas (figura 15, tabla 13), y el proceso usando apilamiento de etiquetas (figura 16, tabla 14), estas tablas se utilizan solo para describir el proceso. Para el caso que no usa apilamiento de etiquetas (figura 15, tabla 13), cuando un flujo llega al Ingress-6DER1, se le debe asignar una etiqueta (se asume que los paquetes vienen no etiquetados o con valor de etiqueta 0). En el caso del ejemplo de la figura 15, hay dos flujos, uno que llega al ingress-6DER1 y el otro al ingress-6DER2, por tanto, se asignan a estos paquetes las etiquetas A1 y B1 respectivamente (filas 1 y 2 de la tabla 13), siendo estas etiquetas aquellas con las que salen los paquetes hacia el próximo salto 6DTR1. Cada vez que los paquetes llegan a un router de tránsito se hace intercambio de etiquetas (operación swap), como en el caso de las líneas de la 3 a las 10 de la tabla 13, donde el flujo 1 utilizó las etiquetas A1, A2, A3, A4, A5 y el flujo 2, utilizó las etiquetas B1, B2, B3, B4, B5 (filas 3-10 de la tabla 13). Una vez los paquetes llegan a los respectivos routers de egreso 6DER3 y 6DER4, las etiquetas son borradas y se envían a su destino final (filas 11 y 12).

En el caso de apilamiento de etiquetas, como en la figura 16 y tabla 14, dos flujos llegan a los routers de ingreso, uno al router ingress-6DER1 y otro al 6DER2, (filas 1 y 2 de la tabla 14), a los cuales se asignan las etiquetas A1 y B1 respectivamente. En los dos casos se asume que los caminos fueron seleccionados previamente por el algoritmo de enrutamiento para cada flujo, y como se puede apreciar tienen en común los routers de tránsito 6DTR1-6DTR2-6DTR3-6DTR4. Cuando los dos flujos llegan al 6DTR1, que es el nodo punto de entrada del túnel, este debe hacer el proceso de apilamiento de etiquetas (por ejemplo, mediante encapsulación IPv6 mediante con GPT) y asigna la etiqueta C1 al túnel formado (ver fila 3 y 4 de la tabla 14). Luego en los siguientes routers de tránsito comunes (6DTR2 y 6DTR3) se hace la conmutación de los paquetes mediante intercambio de las etiquetas más externas, en donde ambos flujos utilizaron las etiquetas (C2-C3) en las filas 5 a 8 de la tabla 14 respectivamente. El router final de la tunelización es el 6DTR4; en este se hace el borrado de

la etiqueta más externa del túnel, quedando la etiqueta original de los 6LSPs individuales, y luego el router hace el último intercambio de etiquetas para este caso y luego envía los paquetes a los routers de egreso 6DER3 y 6DER4 con las etiquetas C4 y C5 respectivamente (fila 9-12 de la tabla 14). El proceso de intercambio de etiquetas sigue siempre y cuando los paquetes no lleguen al router de egreso. Finalmente, cuando los paquetes llegan a los routers de egreso (egress-6DER3 y egress 6DER4) la etiqueta es borrada y los paquetes son enviados al destino final (filas 13 y 14 de la tabla 14).

Como se puede apreciar al comparar los dos casos, el apilamiento de etiquetas permite un ahorro significativo en el espacio de etiquetas. En el caso sin apilamiento se usaron 10, en el enrutamiento de los dos flujos y en el caso con apilamiento se usaron 7 etiquetas, lo que signfica que para este ejemplo se obtiene un ahorro del 30% de espacio de etiquetas.

Tabla 13. Conmutación de Paquetes por Etiquetas de Flujo sin Tunelización.

Router& Interfaz Entrante	*Etiqueta In*	*Acción*	*Etiqueta Out*	*Próximo Salto*	*fila*
Ingress-6DER1	*0*	*Asignar Etiqueta A1*	*A1*	*6DTR1*	*1*
Ingress-6DER2	*0*	*Asignar Etiqueta B1*	*B1*	*6DTR1*	*2*
6DTR1 Desde Ingress-6DER1	*A1*	*Intercambio Etiqueta A2*	*A2*	*6DTR2*	*3*
6DTR1 Desde Ingress-6DER2	*B1*	*Intercambio Etiqueta B2*	*B2*	*6DTR2*	*4*
6DTR2 Desde6DTR1	*A2*	*Intercambio Etiqueta A3*	*A3*	*6DTR3*	*5*
6DTR2 Desde 6DTR1	*B2*	*Intercambio Etiqueta B3*	*B3*	*6DTR3*	*6*
6DTR3Desde 6DTR2	*A3*	*Intercambio Etiqueta A4*	*A4*	*6DTR4*	*7*
6DTR3 Desde 6DTR2	*B3*	*Intercambio EtiquetaB4*	*B4*	*6DTR4*	*8*
6DTR4 Desde 6DTR3	*A4*	*Intercambio EtiquetaA5*	*A5*	*Egress-6DER3*	*9*
6DTR4 Desde 6DTR3	*B4*	*Intercambio Etiqueta B5*	*B5*	*Egress-6DER4*	*10*
Egress- 6DER3	*A5*	*Borrar Etiqueta*	*0*	*N/A*	*11*
Egress- 6DER3	*B5*	*Borrar Etiqueta*	*0*	*N/A*	*12*

Tabla 14. Conmutación de Paquetes por Etiquetas de Flujo con Tunelización.

Router & Interfaz entrante	*Etiqueta Entrante*	*Acción*	*Etiqueta Saliente*	*Próximo Salto*	*Fila*
Ingress-6DER1	*0*	*Asignar Etiqueta A1*	*A1*	*6DTR1*	*1*
Ingress-6DER2	*0*	*Asignar Etiqueta B1*	*B1*	*6DTR1*	*2*
6DTR1 Desde Ingress-6DER1	*A1*	*Asignación Nueva Etiqueta Asigna C1*	*A1, C1*	*6DTR2*	*3*
6DTR1 Desde Ingress-6DER2	*B1*	*Asignación Nueva Etiqueta Asigna C1*	*B1, C1*	*6DTR2*	*4*
6DTR2 Desde 6DTR1	*C1*	*Intercambia A C2*	*A1, C2*	*6DTR3*	*5*
6DTR2Desde 6DTR1	*C1*	*Intercambia a C2*	*B1, C2*	*6DTR3*	*6*
6DTR3 Desde 6DTR2	*C2*	*Intercambia a C3*	*A1, C3*	*6DTR4*	*7*
6DTR3 Desde 6DTR2	*C2*	*Intercambia a C3*	*B1, C3*	*6DTR4*	*8*
6DTR4 Desde 6DTR3	*C3*	*Borrar Etiqueta externa*	*A1*	*6DTR4*	*9*
6DTR4 Desde 6DTR3	*C3*	*Borrar Etiqueta externa*	*B1*	*6DTR4*	*10*
6DTR4 Desde 6DTR4	*A1*	*Intercambia etiqueta a C4*	*C4*	*Egress-6DER3*	*11*
6DTR4 Desde 6DTR4	*B1*	*Intercambia etiqueta a C5*	*C5*	*Egress-6DER4*	*12*
Egress-6DER3 Desde 6DTR4	*C4*	*Borrar Etiqueta*	*0*	*N/A*	*13*
Egress- 6DER4 Desde 6DTR4	*C5*	*Borrar Etiqueta*	*0*	*N/A*	*14*

Ejemplo 2. En la figura 17, se muestra un ejemplo sencillo de tres comunicaciones que se están enviando por los nodos I, A, M, hacia los nodos J, G, N, los cuales serían nodos de

ingreso y egreso de la arquitectura PSA-TE6 y que para esta propuesta se denominan 6DLSPs. En el caso de no utilizar tunelización solo se formarían estos caminos conmutados por etiquetas, los cuales se establecen como de nivel 1, L=1 (Se utilizará L para asignar los diferentes niveles de apilamiento). Si el apilamiento de etiquetas es habilitado, en este ejemplo se podrían formar máximo dos túneles de nivel 2, L=2, que serían: B-F que tuneliza a A-G y I-J, C-E que formaría un túnel de nivel 2 con respecto a M-N y un túnel de nivel 3, que estaría también en el segmento C-E con respecto a B-F y a su vez A-G y I-J.

Como se puede apreciar, en este ejemplo y en los datos de la tabla 15, para las tres comunicaciones, existen segmentos de la red que son comunes y es donde se forman los túneles de nivel superior a 1, (L>1). En la tabla 15, se muestra el número de etiquetas que se utilizan para los casos, cuando no hay tunelización y cuando sí la hay, mostrando para este ejemplo (Figura 17) que el proceso de apilamiento permite un ahorro considerable en el espacio de etiquetas que alcanza un 37.5%.

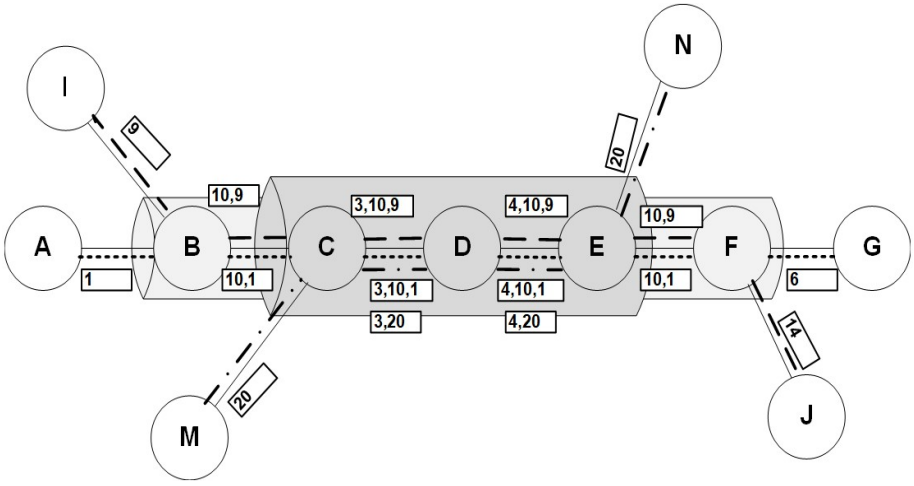


Figura 17. Posibles túneles formados para tres comunicaciones.

Tabla 15. Número Total de Etiquetas Utilizadas con y sin Apilamiento de Etiquetas.

Estado	*Túneles*	*Caminos*	*#Total Etiquetas*
Sin tunelización	*L=1*	*A-G*	*16*
		I-J	
		M-N	
Con Tunelización-Apilamiento de etiquetas	*L=1,2*	*A-G/B-F/C-E*	*10*
		I-J/B-F/C-E	
		M-N/C-E	

4.2 EVALUACIÓN DEL COSTO DE APILAMIENTO DE ETIQUETAS DE FLUJO IPV6

Con el fin de evaluar el costo de la creación de los túneles para diferentes niveles y poder tener una aproximación del funcionamiento de la arquitectura PSA-TE6 en este aspecto, se ha realizado un análisis con respecto a los costos de operaciones en cada router de acuerdo con su rol dentro de la arquitectura. Este mismo análisis se hace con IP/MPLS con el fin de comparar los resultados. Para ambas arquitecturas tenemos routers que cumplen el mismo rol en el proceso del establecimiento de caminos conmutados por etiquetas y creación de túneles y con la ayuda de los mismos protocolos de señalización, algoritmos de enrutamiento basados en restricciones y protocolos de enrutamiento extendidos para ingeniería de tráfico tal como se describió en el capítulo 3.

Para determinar los costos de operaciones en cada router se han hecho las siguientes suposiciones: a cada entrada en una tabla de envío del respectivo router se ha asignado un valor de 1, suponiendo que las arquitecturas evaluadas trabajan con el mismo método de búsqueda. Para el caso de inserción o borrado de cabeceras (lectura o escritura), se ha asignado un valor de 1 para cada 64 bits, suponiendo routers que trabajan a 64 bits. Entonces, los costos a tener en cuenta de acuerdo a las funciones que realizan los routers en el proceso de envío de información son: costos de operaciones en nodos de ingreso, C_{in}, costo de operaciones en nodos de egreso, C_{eg}, Costo de operaciones en nodos tránsito C_{tr}, Costo de operaciones en nodos punto de entrada del túnel, operaciones para poner una nueva etiqueta cuando hay apilamiento de etiquetas, C_{push}, Costos de operaciones en nodo punto de salida del túnel, de borrado de la etiqueta superior de la pila de etiquetas, C_{pop}. En la figura 18, se muestran los costos y su ubicación de acuerdo con los roles que cumplen los routers en el proceso de envío.

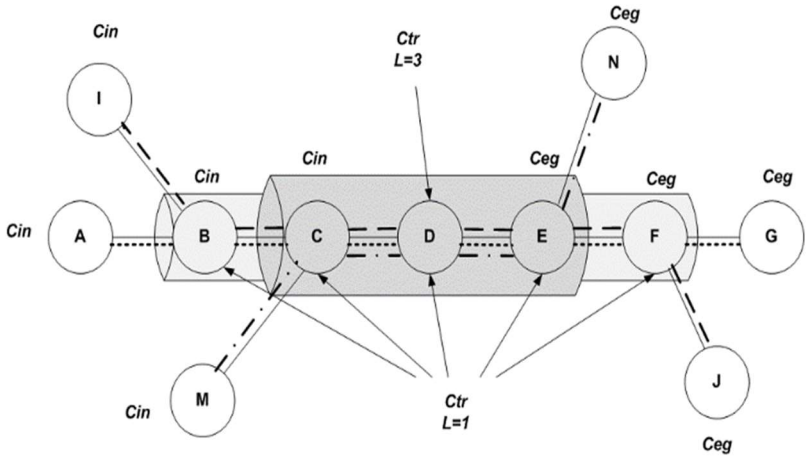


Figura 18. Ubicación de los costos de acuerdo con el rol de los routers.

En las tablas 16, 17 y 18, se pueden apreciar los valores de los costos para cada caso de acuerdo con las operaciones realizadas en cada router. En el caso de la arquitectura PSA-TE6 se tienen las tablas 16 y 17, teniendo en cuenta los dos mecanismos para la realización del apilamiento de etiquetas explicado en la sección anterior. En la tabla 18, se describen los costos de operaciones en la arquitectura IP/MPLS. Los costos se calculan con base en el número de lecturas/escrituras de memoria requeridas, las cuales se especifican como el número de operaciones entre paréntesis.

Tabla 16. Costos de operaciones en la arquitectura PSA-TE6 cuando se usa GPT.

PSA-TE6 con GPT		
Costo	**Operaciones**	**Valor**
C_{in}	- *Búsqueda en FIB tipo FTN6 FEC-to-N6HLFE (1 op)* - *Etiquetado del paquete (poner el valor de etiqueta en el campo de Etiqueta de Flujo IPv6)(1 op.)*	2
C_{eg}	- *Búsqueda de la etiqueta en la tabla FIB tipo I6LTN (1 op.)* - *Borrado de etiqueta (operación pop, poner el campo de Etiqueta de Flujo IPv6 en cero) (1 op.)*	2
C_{tr}	- *Búsqueda de la etiqueta en la tabla I6LTN (1 op.)* - *Intercambio de etiqueta (Operación swap, poner una nueva, modificación del campo Etiqueta de Flujo IPv6 de la cabecera superior) (1 op.)*	2
C_{push}	- *Búsqueda de la etiqueta en la tabla I6LTN (1 op.)* - *Encapsulación IPv6,agregar una cabecera IPv6 (operación push) (5 op.)*	6
C_{pop}	- *Búsqueda de la etiqueta en la tabla I6LTN. (1 op.)* - *Desencapsulación de la cabecera IPv6 (operación pop) (5 op.)* - *Buscar la nueva etiqueta superior de la pila en I6LTN consultando NHFLE (1 op), poner etiqueta saliente (1 op.)*	8

Tabla 17. Costos de operaciones en la arquitectura PSA-TE6 cuando se usa cabecera de opcion hop-by-hop.

PSA-TE6 con cabecera de opción hop-by-hop		
Costo	**Operaciones**	**Valor**
C_{in}	- *Búsqueda en FIB tipo FTN6 FEC-to-N6HLFE (1 op.)* - *Etiquetado del paquete (poner el valor de etiqueta en el campo de Etiqueta de Flujo IPv6) (1 op.)*	2
C_{eg}	- *Búsqueda de la etiqueta en la tabla FIB tipo I6LTN (1 op.)* - *Borrado de etiqueta (operación pop, poner el campo de Etiqueta de Flujo IPv6 en cero) (1 op.)*	2
C_{tr}	- *Búsqueda de la etiqueta en la tabla I6LTN (1 op.)* - *Intercambio de etiqueta (Operación swap, poner una nueva, modificación del campo Etiqueta de Flujo IPv6 de la cabecera superior) (1 op.)*	2
C_{push}	- *Búsqueda de la etiqueta en la tabla I6LTN (1 op.)* - *Operación push, copiar el valor de etiqueta original del paquete en la cabecera de opción hop-by-hop (2 op.), y poner la nueva etiqueta en cabecera IPv6 (1 op.)*	4
C_{pop}	- *Búsqueda de la etiqueta en la tabla I6LTN (1 op.).* - *Operación pop (Lectura del ultima etiqueta de pila en cabecera de opción (1 op.), escritura del nuevo valor de longitud de la cabecera de opción (1 op.), escritura de la etiqueta leída en el campo Etiqueta de Flujo de la cabecera IPv6 (1 op),* - *Buscar la nueva etiqueta superior de la pila en I6LTN consultando NHFLE (1 op.). Poner etiqueta saliente (1 op.)*	6

Tabla 18. Costos de operaciones en la arquitectura IP/MPLS.

IP/MPLS		
Costo	Operaciones	Valor
C_{in}	- *Búsqueda en FIB tipo FTN FEC-to-NHFLE NHLFE (1 op)* - *Mapeo de campos TTL y DS desde cabecera IPv6 a cabecera MPLS (lectura y Escritura) (2 op.)* - *Etiquetado del paquete (Inserción de la cabecera MPLS) (1 op.)*	4
C_{eg}	- *Búsqueda de la etiqueta en la tabla ILM. (1 op.)* - *Borrado de etiqueta (operación pop, extracción de cabecera MPLS) (1 op.)* - *Mapeo del campo TTL desde cabecera MPLS a cabecera IPv6 (2 op.)*	4
C_{tr}	- *Búsqueda de la etiqueta en la tabla ILM. (1 op.)* - *Intercambio de etiqueta (Operación swap: extracción de la cabecera MPLS (1 op.), inserción de cabecera MPLS con la etiqueta saliente (1 op).*	3
C_{push}	- *Búsqueda de la etiqueta en la tabla ILM. (1 op.)* - *Operación push: poner un nuevo valor de etiqueta- inserción de cabecera MPLS. (1 op.)*	2
C_{pop}	- *Búsqueda de la etiqueta en la tabla ILM. (1 op.)* - *Operación pop: borrar la etiqueta superior de la pila de etiqueta-extracción de la cabecera MPLS (1 op.).* - *Buscar la nueva etiqueta superior de la pila en ILM consultando NHFLE (1 op.). Poner etiqueta saliente (1 op.)*	4

4.2.1 Análisis de Costos Comunes en el Proceso de Envío

Los costos de ingreso y egreso (C_{in} y C_{eg}) están presentes tanto en los túneles de nivel 1, en routers de borde de la arquitectura PSA-TE6, como los costos de ingreso y egreso presentes en los puntos finales de los túneles de nivel superior a 1, L>1. Los costos de intercambio de etiquetas (C_{tr}) en nodos tránsito están presentes también en todos los niveles de los túneles. Para hacer el cálculo del costo total de nodos tránsito se debe conocer el número de nodos tránsito del camino respectivo, el cual está representado en la formula por $Ntr_{L=1,2,3}$. Las ecuaciones (1), (2) y (3), tienen en cuenta los costos que son comunes para todos los túneles en el proceso de envío, para las soluciones a comparar PSA-TE6 e IP/MPLS. Para la arquitectura PSA-TE6 con los mecanismos de apilamiento explicados en la sección anterior, con tunelización IPv6 mediante GPT y el otro por medio de la cabecera de opción IPv6, se han denominado PSA-TE6_GPT y PSA-TE6_HBH respectivamente, para los que se definen las ecuaciones (1) y (2). Para IP/MPLS, se obtuvo la ecuación (3). En estas tres ecuaciones se ha tenido en cuenta los valores determinados en las tablas 16, 17 y 18 como coeficientes y se ha establecido $C_{in}=C_{tr}=C_{eg}=1$; *NT* como número de túneles de cada nivel que se estén analizando, con el fin de obtener costo total de costo comunes en el proceso de envío.

$$Cost_{\substack{PSA-T \\ L=1,2,3\ldots}}{}_{GPT} = \left(2Cin + 2Ctr * \left(Ntr_{L=1,2,3}\right) + 2Ceg\right) * NT_{L=1,2,3..} \quad \text{EC. 1}$$

$$Cost_{\substack{PSA-TE6_{HBH} \\ L=1,2,3\ldots}} = \left(2Cin + 2Ctr * \left(Ntr_{L=1,2,3}\right) + 2Ceg\right) * NT_{L=1,2,3..} \quad \text{EC. 2}$$

$$\underset{L=1,2,3\ldots}{Cost_{IP/MPLS}} = \left(4Cin + 3Ctr * \left(Ntr_{L=1,2,3}\right) + 4Ceg\right) * NT_{L=1,2,3..} \quad \text{EC. 3}$$

4.2.2 Análisis de Costos de Apilamiento de Etiquetas de Flujo IPv6

Los costos presentes en los puntos finales del túnel, de poner y borrar etiqueta, es decir costos push y pop, C_{push}, C_{pop}, solo hacen parte del proceso de apilamiento de etiquetas y de túneles de nivel superior a 1, (L>1). Por otro lado, cuando se hace apilamiento de etiquetas, el costo asociado a los nodos tránsito, es tenido en cuenta solo para los nodos tránsito del túnel superior. Por tanto, se debe hacer un ajuste en el cálculo de número de nodos tránsito, de acuerdo con los túneles existentes de nivel superior a 1 con respecto a los túneles de nivel inferior. Para hacer el ajuste es necesario conocer el número de nodos del túnel superior que son nodos tránsito en los túneles de nivel inferior y restar el costo asociado a ese número de nodos. Esto se denota como | NL / Ntr_{L-1} |. Es decir, el número de nodos del túnel de nivel L=2, en el túnel B-F, en el ejemplo de Fig. 17, que eran tránsito en el túnel de nivel 1 (túnel A-G) es igual a 5. Estos 5 por el costo asociado, se deben restar en la ecuación 2, ya que se tuvieron en cuenta en la ecuación 1 y ahora son tunelizados. El cálculo del costo push, pop y el ajuste de costo de tránsito se definen en las ecuaciones (4), (5) y (6), para la propuesta PSA-TE6 y para IP/MPLS que es la arquitectura comparada. Al igual que en las ecuaciones anteriores hay que multiplicar por el número de túneles de cada nivel con las mismas características presentes, este número es denotado por $NT_{L>1}$.

$$\underset{L>1}{Cost_{PSA-TE6_{GPT}}} = (6Cpush - 2Ctr * |N_L, Ntr_{L-1}| + 8Cpop) * NT_{L>1} \quad \text{EC. 4}$$

$$\underset{L>1}{Cost_{PSA-TE6_{HBH}}} = (4Cpush - 2Ctr * |N_L, Ntr_{L-1}| + 6Cpop) * NT_{L>1} \quad \text{EC. 5}$$

$$\underset{L>1}{Cost_{IP/MPLS}} = (2Cpush - 3Ctr * |N_L, Ntr_{L-1}| + 4Cpop) * NT_{L>1} \quad \text{EC. 6}$$

4.2.3 Cálculo del costo Total donde están presentes túneles en diferentes niveles

Para calcular los costos totales finales se suman las dos partes para cada caso. Estos costos totales definen en las ecuaciones (7), (8) y (9).

$$TotalCost_{PSA-TE6_{GPT}} = \underset{L=1,2,3..}{Cost_{PSA-T\_GPT}} + \underset{L>1}{Cost_{PSA-T\_GPT}} \quad \text{EC. 7}$$

$$TotalCost_{PSA-TE6_{HBH}} = \underset{L=1,2,3..}{Cost_{PSA-TE6\_HBH}} + \underset{L>1}{Cost_{PSA-T\quad\_HBH}} \qquad \text{EC. 8}$$

$$TotalCost_{IP/MPLS} = \underset{L=1,2,3..}{Cost_{IP/MPLS}} + \underset{L>1}{Cost_{IP/MPLS}} \qquad \text{EC. 9}$$

4.2.4 Resultados del Análisis de Costos en el proceso de envío con y sin tunelización

En las Fig. 19-22, se muestran los resultados de los costos obtenidos para la arquitectura PSA-TE6, utilizando dos mecanismos para el apilamiento de etiquetas, PSA-TE6_GPT y PSA-TE6_HBH. Estas propuestas se han comparado con la arquitectura existente MPLS cuando trabaja a nivel IP, IP/MPLS. Estos datos fueron obtenidos utilizando las ecuaciones (1) a (9), para 500, 1000,10000 y 20000 comunicaciones, variando los porcentajes de túneles de nivel superior a 1, (L>1), presentes en el proceso de envío, desde el 5% hasta el 50%. Para este análisis, los túneles de L>1, tienen las mismas características en cuanto al nivel y al número de nodos tránsito del ejemplo de la figura 18. Los resultados de las figuras 19-22, muestran que nuestra propuesta PSA-TE6 utilizando los dos mecanismos de tunelización supera a IP/MPLS hasta cuando hay un 40% de presencia de túneles de nivel superior a 1, L>1. Lo que también indica que cuando no es necesario el apilamiento de etiquetas, los costos de PSA-TE6 están por debajo de la solución IP/MPLS, alrededor de un 45% más bajos (ver figura 22). A medida que se incrementa el número de túneles de nivel de L>1, en los porcentajes entre 40% y 45%, las dos soluciones PSA-TE6 y IP/MPLS tienen costos similares, como se observa en las figuras 19-21.

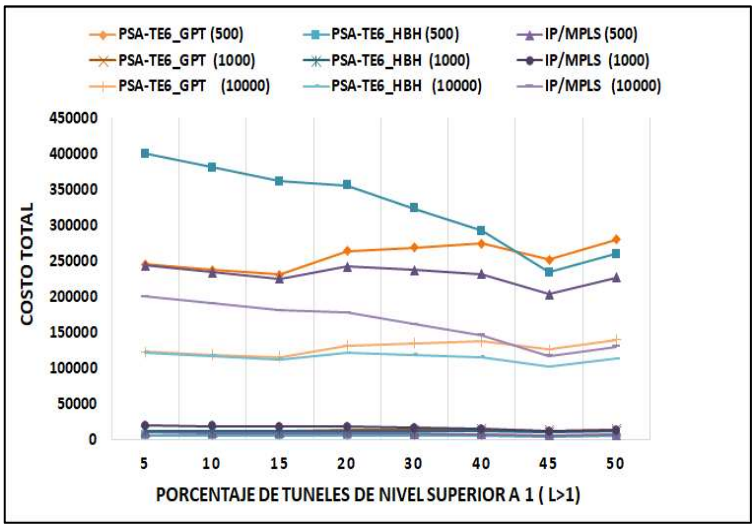


Figura 19. Costos totales para diferente número de comunicaciones y porcentajes de túneles de niveles superiores a 1, L>1.

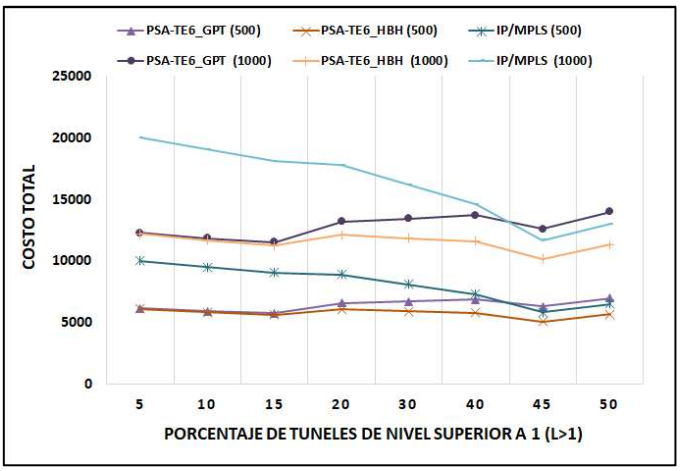


Figura 20. Costos totales para número de comunicaciones de 500 y 1000 y porcentajes de túneles de niveles superiores a 1, L>1.

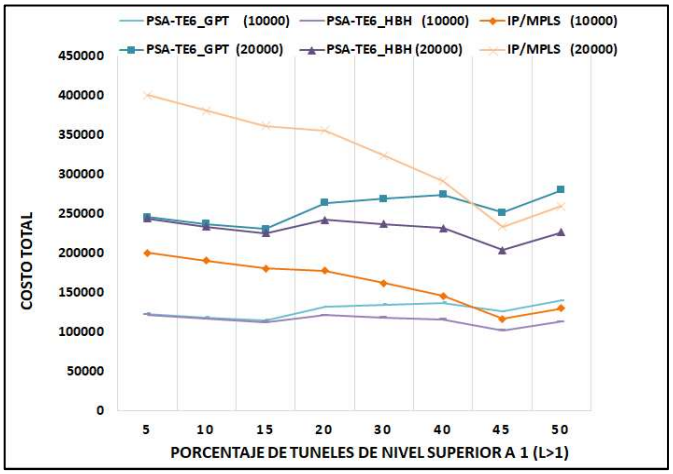


Figura 21. Costos totales para número de comunicaciones de 10000 y 20000 y porcentajes de túneles de niveles superiores a 1, L>1.

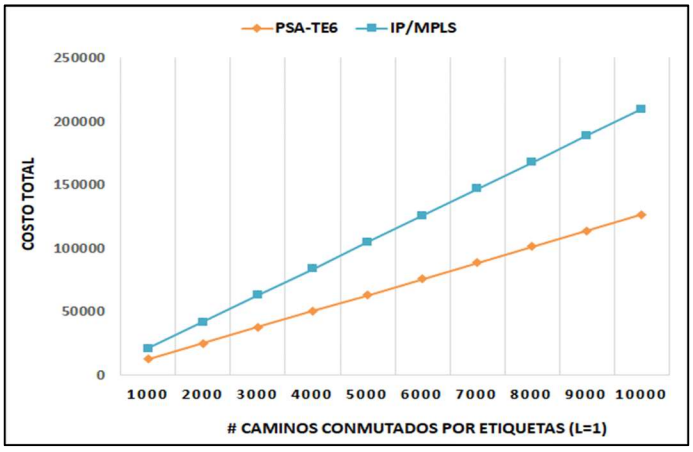


Figura 22. Costo Total PSA-TE6 Vs IP/MPLS para túneles L=1.

4.3 MINIMIZACIÓN DEL COSTO DE APILAMIENTO DE ETIQUETAS

Con el fin de optimizar el costo de apilamiento de etiquetas para la arquitectura PSA-TE6 propuesta en esta tesis, se presenta una formulación de programación lineal entera, ILP, similar a la usada en [175]. Al analizar los costos presentes en proceso de envío para la solución propuesta PSA-TE6 y para la existente IP/MPLS, se observa que PSA-TE6 tiene costos menores en los costos de ingreso, egreso y tránsito. La solución IP/MPLS tiene costos de push y pop menores que la propuesta PSA-TE6, es por esto que se presenta la formulación ILP con el fin de evaluar la influencia en la minimización de los costos cuando los costos de ingreso y egreso cambian, teniendo en cuenta que estos costos están presentes tanto cuando hay túneles de nivel superior a 1 como cuando la propuesta trabaja en su forma base, es decir sin apilamiento de etiquetas. También se evalúa cuando en presencia de tunelización hay cambios en los costos push. En esta sección se presenta la formulación ILP y las definiciones correspondientes en la sección siguiente se presentan los resultados.

Para la formulación ILP, se asume que el enrutamiento de los 6DLSPs que soportan de flujos de tráfico de extremo a extremo es dado y seleccionado a priori, de acuerdo con algún algoritmo basado en restricciones. A continuación, se describen algunas definiciones, los parámetros y variables de la formulación en la tabla 19 y la formulación en la tabla 20.

Tabla 19. Lista de Notaciones para el problema ILP

SD_PAIRS:	*Conjunto de pares de nodos fuente y destino de los flujos de tráfico o FECs.*
P1:	*Conjunto de caminos 6DLSPs, p_i representa el camino entre el par fuente-destino que soporta flujos de tráfico de extremo a extremo, donde i ϵ SD_PAIRS.*
P2:	*Es el conjunto de caminos de LSP candidatos de nivel superior a 1. (P2 = {Ø} cuando L=1, es decir en ausencia de apilamiento de etiqueta).*
P:	*P1 ∪ P2*
i, j:	*índices para 6DLSPs.*
l:	*Nivel de la pila.*
|pi,pj|:	*función que toma los nodos en el camino pi y retorna el número de aquellos que son nodos intermediarios o de tránsito en el camino pj. Así que |pi,pi| es el número de nodos tránsitos del camino pi.*
L:	*Profundidad de pila de etiqueta máximo.*
δ^l_i**:**	*Variable binaria, 1 si el 6DLSP candidato (donde pi ϵ P) es seleccionado para el nivel l, de lo contrario es 0.*
$\gamma^l_{i,j}$**:**	*Variable binaria, 1 si el 6DLSP candidato j del nivel l-1 (pj ϵ P) es tunelizado a través del 6FLSP candidato i del nivel l (pi ϵ P, pi ≠ pj).*
C_i**:**	*Costo de operaciones nodo de ingreso.*
C_{eg}**:**	*Costo de operaciones nodo de egreso.*
C_{tr}**:**	*Costo de operaciones nodos tránsito.*
C_{push}**:**	*Costo de operación de poner nueva etiqueta.*
C_{pop}**:**	*Costo de operación de borrar etiqueta.*

Tabla 20. Formulación del problema ILP

Objetivo:	
$Min: \sum_{l=1}^{L} \sum_{p_i \in P} (C_{tr} * \lvert p_i, p_i \rvert + C_{in} + C_{eg}) * \delta_i^l + \sum_{p_i \in P} \sum_{r=1}^{2} \sum_{\substack{p_j \in P_r: \\ p_j \supset p_i}} \sum_{l=r+1} (C_{push} + C_{pop} - C_{tr} * \lvert p_i, p_j \rvert) * \gamma_j^l$	EC. 10
Restricciones:	
$\sum_{l=1}^{L} \boldsymbol{\delta_i^l = 1} \; \forall_i: \boldsymbol{p_i \in P_1}$	EC.11
$\sum_{l=2}^{L} \boldsymbol{\delta_i^l \leq 1} \quad \forall_i: \boldsymbol{p_i \in P_2}$	EC.12
$\boldsymbol{2\gamma_{i,j}^l \leq \delta_i^l + \delta_j^{l-1}} \quad \forall \mathbf{p_i} \in \mathbf{P}, \forall \mathbf{p_j} \in \mathbf{P_r}: \mathbf{p_j} \supset \mathbf{p_i}$ $\forall l = r+1, \ldots., L, \forall r = 1,2$	EC.13
$\sum_{r=1}^{2} \sum_{l=r+1}^{L} \sum_{\substack{\boldsymbol{p_i \in P_r} \\ \boldsymbol{p_i \subset p_j}}} \boldsymbol{\gamma_{i,j}^l \leq \sum_{l=s}^{L} \delta_j^l} \quad \forall \boldsymbol{p_j \in P_s}, \forall \boldsymbol{s = 1,2}$	EC.14
$\boldsymbol{\delta_i^l \in \{0,1\}} \quad \forall \boldsymbol{p_i \in P_r}, \forall \boldsymbol{l = r, \ldots., L}, \; \forall \boldsymbol{r = 1,2}$	EC.15
$\boldsymbol{\gamma_{i,j}^l \in \{0,1\}} \quad \forall \boldsymbol{p_i \in P}, \forall \boldsymbol{p_j \in P_r}: \boldsymbol{p_j \supset p_i}$ $\forall l = r+1, \ldots, L, \forall r = 1,2$	EC.16

Las operaciones en cada router de acuerdo con su rol en la arquitectura PSA-TE6 están definidas así: en los nodos de ingreso, Ingress-6DER, hace operaciones de mapeo de un paquete no etiquetado a un FEC (FTN6), consultando los datos N6FLD y etiquetando el paquete, (C_{in}). Los routers de egreso Egress-6DER, hacen operaciones de búsqueda de la etiqueta en la tabla I6LTN, consultando los datos N6FLD, borrando la etiqueta y pasando el paquete a las capas superiores (C_{eg}). Los nodos tránsito, 6DTRs hacen operaciones de buscar la etiqueta en la tabla I6LTN, consultando los datos N6FLD, e intercambiando la etiqueta superior, es decir, borrando la etiqueta superior y poniendo una nueva, (C_{tr}). Las operaciones push incluyen búsqueda de la etiqueta en la I6LTN, consulta de los datos del N6FLD, poner un nuevo valor de etiqueta sobre la parte superior de la pila de etiqueta (C_{push}). Para borrar las etiquetas se hacen operaciones de búsqueda de la etiqueta en la I6LTN, consultando los datos N6FLD, borrando de la etiqueta superior desde la pila de etiquetas, búsqueda de la nueva etiqueta superior en la tabla I6LTN, y consultar de nuevo los datos N6FLD (C_{pop}).

El problema consiste en seleccionar el conjunto de 6DLSPs para soportar el conjunto dado de flujos de tráfico extremo a extremo y su nivel jerárquico. La función objetivo (Ecuación 10) es la minimización el costo total de las operaciones de los nodos para diferentes niveles de tunelización. La restricción (Ecuación 11) obliga la selección del nivel de cada 6DLSP en P1. La restricción (Ecuación 12) impone que solo un LSP candidato puede ser seleccionado en cualquier nivel de pila, para un par fuente-destino dado. La restricción (Ecuación 13) impone que un 6DLSP candidato puede crear un túnel a través de un 6DLSP candidato de nivel más alto solamente si ambos 6DLSPs han sido seleccionados. La restricción (Ecuación 14) impone que cualquier 6DLSP seleccionado puede ser asignado a solamente un 6DLSP de

nivel más alto. Las variables deben ser booleanas, ya que solamente un 6DLSP puede existir para un camino dado para un par fuente-destino (Ecuación 15), y cada LSP puede crear un túnel a lo sumo a través de un 6DLSP de nivel más alto, sobre un enlace dado (Ecuación 16).

La topología de prueba para la formulación ILP es la que se muestra en la figura 23. Los nodos de ingreso son los nodos 1,2, 3 y 13 y los nodos de egreso son los nodos 7,8 y 9. En la siguiente sección se muestran los resultados.

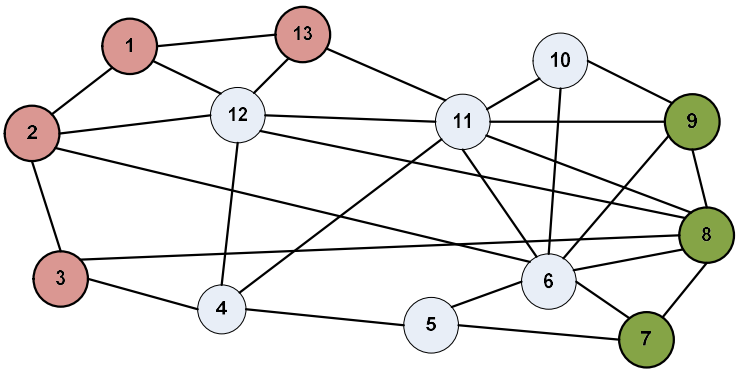


Figura 23. Red de prueba

4.3.1 Resultados

Se usaron las herramientas AMPL [87] y CPLEX solver [88] para ejecutar la formulación ILP usando la red de prueba de 13 nodos mostrada en la Figura 23. En la figura 24, el costo mínimo versus niveles de apilamiento es mostrado para diferentes valores de costos de ingreso y egreso, manteniendo los demás costos normalizados a 1 ($C_{tr} = C_{push} = C_{pop} = 1$). Esta prueba fue desarrollada para evaluar el efecto de incrementar los costos de ingreso y egreso en una red de conmutación de paquetes por etiquetas como la propuesta PSA-TE6 y también IP/MPLS. Los resultados muestran que, mediante el incremento de los costos de ingreso y egreso, hay un incremento en el costo mínimo total para ambos casos: cuando ambas soluciones trabajan en su forma base, es decir cuando hay solamente caminos conmutados por etiquetas (L=1) y cuando hay presencia de túneles de diferentes niveles (L>1), ver figura 24. Como se mencionó antes, los costos de ingreso y egreso son costos comunes del proceso de envío, y estos están presentes en ambos métodos, cuando el proceso de apilamiento de etiquetas no está habilitado y cuando si lo está.

En la figura 25, se muestran los resultados obtenidos con la formulación ILP, pero en este caso, los valores de costo push son cambiados desde 0.5 a 2 en pasos de 0.5. Los costos push están solamente presentes en el proceso de apilamiento. Para este escenario, se puede observar que cuando todos los costos son iguales (igual a 1) y el apilamiento está habilitado, es decir (L > 1), el costo mínimo es más bajo comparado a cuando no hay apilamiento es decir (L=1), lo cual es debido al cambio en el número de nodos tránsito, porque está presente

solamente el costo de nodos transito del túnel superior. Esta formulación encuentra un mínimo cuando el costo push está +/- 50% (50% por encima o por debajo) del valor promedio de los otros costos ($0.5 < C_{push} < 1.5$). En consecuencia, tomando en cuenta que el costo push está solamente presente en el proceso de apilamiento, en este modelo es deseable que el número de niveles de tunelización sea de 2 o 3 para reducir el costo total. Un valor de L=3 es un número típico de nivel de túnel en las redes actuales como es mencionado en [176]. Esta conclusión es válida para ambas tecnologías (PSA-TE6 y MPLS).

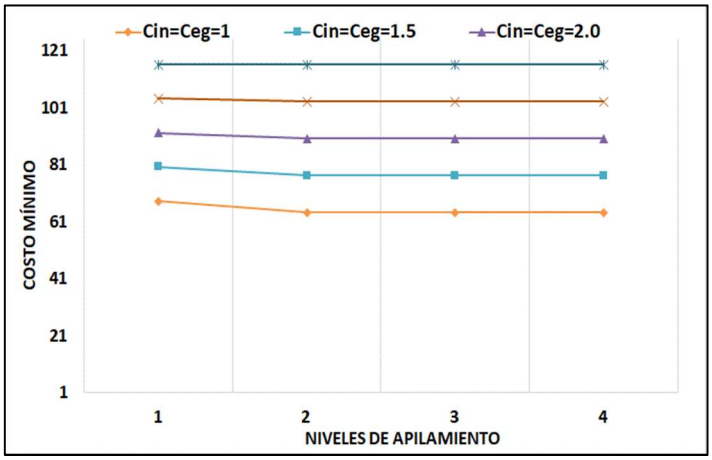


Figura 24. Minimización del costo en el proceso de envío para varios niveles de apilamiento con variación en los costos de ingreso y egreso.

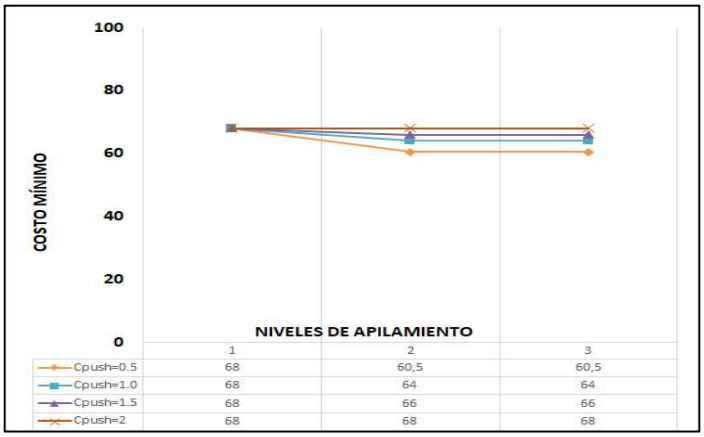


Figura 25. Minimización del costo de envío para cambios de C_{push} para tres niveles de apilamiento.

4.4 EVALUACIÓN DEL BALANCEO DE CARGA EN PSA-TE6

En esta sección se presenta la evaluación de balanceo de carga, lo cual es una situación típica en estudios de ingeniería de tráfico. En esta evaluación se compara la propuesta PSA-

TE6 con MPLS, ya que esta es una tecnología comúnmente usada para soportar ingeniería de tráfico.

En el proceso de envío de paquetes, PSA-TE6 establece caminos conmutados por etiquetas y desarrolla conmutación de paquetes mediante intercambio de etiquetas en una forma similar como trabaja MPLS. Sin embargo, en este proceso hay una diferencia con respecto a la longitud del paquete ya que MPLS inserta una cabecera de 32 bits, la cual contiene el valor de etiqueta. Por otro lado, PSA-TE6 no requiere bits adicionales ya que PSA-TE6 usa el campo de Etiqueta de Flujo y este campo está contenido en la cabecera IPv6 (ver figura 26).

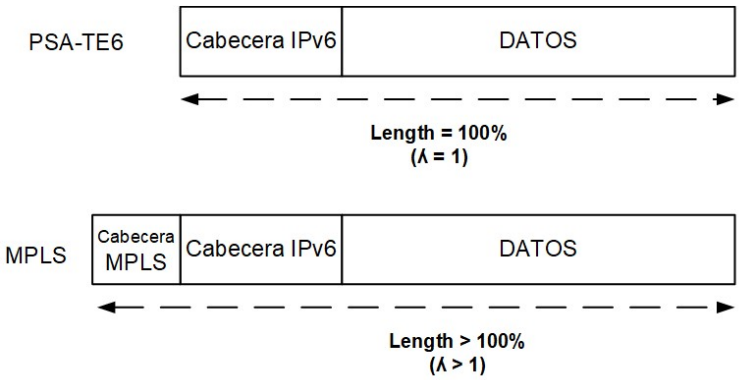


Figura 26. Comparación de la longitude del paquete de PSA-TE6 y MPLS.

De acuerdo con lo anterior, es importante determinar si la distribución de tráfico es afectada por tal diferencia en la longitud del paquete. PSA-TE6 usa el concepto de tunelización (denotados en esta evaluación como 6DLSPs), el cual tiene un gran potencial que permite la separación de tráfico diverso de diferentes usuarios o servicios en diferentes túneles de una manera similar como trabaja MPLS. En ambas tecnologías (PSA-TE6 y MPLS), se pueden presentar dificultades de sobrecarga en routers si el número de túneles no es limitado y la carga no es balanceada. Por tanto, desde un punto de vista de gestión, puede ser deseable limitar el número de túneles sobre un router o enlace. Por esta razón, se presenta una formulación MIP [59], donde la función objetivo del problema es llevar diferentes clases de tráfico en una red a través de la creación de túneles en tal forma que el número de túneles sobre cada router o enlace sea minimizado y la carga balanceada.

En el problema MIP se específica previamente la topología de red, las capacidades de los enlaces, las demandas de tráfico y los caminos candidatos. La formulación está basada en [59], a la cual se le ha realizado una modificación al modelo para comparar el balanceo de carga en PSA-TE6 versus MPLS. Se ha introducido un parámetro λ, el cual representa el efecto de las diferencias en las demandas de tráfico debido a la longitud de los paquetes en cada

tecnología. Tal diferencia es referente a la adición de la cabecera MPLS en redes IP/MPLS y la no existencia de la cabecera MPLS cuando PSA-TE6 es usada.

4.4.1 Optimización del número de túneles

En esta sección se describe el modelo matemático del problema usando la notación como en [59]. Se usa el identificador *d = 1, 2, ..., D* para denotar una demanda asociada con un par de nodos (nodos de origen y destino) que requieren que el ancho de banda h_d sea enrutado en la red. El volumen h_d de la demanda *d* se puede transportar a través de múltiples túneles (rutas) desde la entrada hasta la salida del túnel. Se utiliza el índice *p = 1, 2,..., Pd* para denotar caminos candidatos para la demanda *d*. La fracción del volumen de demanda para la demanda *d* que se transporta en el túnel *p* se denota como X_{dp}, la cual es una variable de decisión continua. La restricción de la demanda, que garantiza que la suma de todos los flujos fraccionarios X_{dp} sobre todos los caminos candidatos *p = 1,2, ..., Pd* debe sumar todo el volumen de la demanda h_d, está expresada en la ecuación 17.

$$\sum_p X_{dp} = 1 \qquad \text{EC.17}$$

Debido a que un flujo podría ser una fracción de demanda de tráfico muy pequeña, se establece el límite más bajo sobre la fracción de un flujo sobre el camino. Se usa una cantidad ε, la cual es el límite más bajo de una fracción de flujo sobre un túnel (camino), y se usa una variable binaria $U_{dp} = 1$ para denotar la selección de un túnel si el límite más bajo es satisfecho, y 0, de lo contrario.

Se adiciona un parámetro λ, el cual representa un porcentaje de ancho de banda de cada demanda de tráfico, que diferencia una tecnología de la otra, de acuerdo con la longitud del paquete (ver figura 26). Por tanto, se tienen las dos siguientes restricciones:

$$\varepsilon U_{dp} \leq \lambda\, h_d X_{dp} \qquad \text{EC. 18}$$

$$X_{dp} \leq U_{dp} \qquad \text{EC.19}$$

La restricción de la ecuación 18 asegura que, si un túnel es seleccionado, entonces el túnel debe tener al menos la fracción de flujo asignado, el cual es establecido a ε. La restricción de la ecuación 19 garantiza que, si un túnel no es seleccionado, entonces la fracción de flujo asociada con este túnel debería ser forzada a ser igual a cero.

Debido a que la topología es dada y la capacidad del enlace es conocida, se debe asegurar que la capacidad C_e del enlace *e* no es excedida. Por otro lado, se usa la variable binaria δ_{edp},

la cual es 1, si el enlace *e* pertenece a el camino *p* de la demanda *d*, y de lo contrario es cero. Así que, la restricción de factibilidad de la capacidad está expresada en la ecuación 20.

$$\sum_d h_d \lambda \sum_p \delta_{edp} X_{dp} \leq C_e \qquad \text{EC.20}$$

El lado izquierdo de la ecuación 20 es el flujo sobre el enlace *e*, el cual es calculado tomando en cuenta todas las demandas *d = 1, 2,…,D*, los caminos candidatos *p = 1,2,…,Pd*, si la demanda dada *d* usa el camino *p* (δ_{edp} = 1), y la fracción de flujo X_{dp}. En esta ecuación se adicionó el parámetro λ que representa un porcentaje de ancho de banda adicional de cada demanda, el cual diferencia a PSA-TE6 de MPLS. El número de túneles sobre el enlace *e* está dado por la ecuación 21.

$$\sum_d \sum_p \delta_{edp} U_{dp} \qquad \text{EC.21}$$

La lista de notaciones y formulación completa están descritas en las tablas 22 y 23. Y las topologías de prueba usadas en esta evaluación son mostradas en la figura 27. Teniendo en cuenta que la meta de optimización es minimizar el número total de túneles, la función objetivo *minimiza (r)* que representa el máximo número de túneles sobre todos los enlaces (ecuación 22). La restricción en la ecuación 23 selecciona la fracción del volumen de demanda a ser llevado sobre un túnel. La restricción en la ecuación 24 es la restricción de factibilidad de capacidad. La restricción de la ecuación 25 restringe la fracción de demanda a un mínimo designado como ε. La restricción de la ecuación 26 obliga a que la fracción de flujo sea cero si un túnel no es seleccionado. Finalmente, la restricción en la ecuación 27 calcula el número de túneles sobre cada enlace.

Tal como se mencionó anteriormente, se adicionó un parámetro λ en las restricciones 24 y 25 de la tabla 22, que representa un porcentaje de ancho de banda de cada demanda, el cual diferencia una tecnología de la otra de acuerdo con la longitud del paquete de cada una. Tal diferencia es causada en MPLS debido a que adiciona una cabecera para establecer una etiqueta en el proceso de envío del paquete, lo cual no es necesario en PSA-TE6. Se asume que las aplicaciones mantienen la misma periodicidad y, en consecuencia, una reducción en la longitud del paquete resulta en una reducción en demanda de tráfico para cada flujo. Como se puede apreciar en la figura 26, los paquetes PSA-TE6 son más cortos que los paquetes MPLS debido a la cabecera adicional en redes IP/MPLS. Así que se toma como referencia a PSA-TE6 como (λ = 1) (ver figura 26), y (λ > 1) para el caso MPLS.

Tabla 21. Lista de Notaciones del problema MIP.

Parámetros:
d *= demanda de tráfico asociada con un par de nodos y clase de tráfico.*
h_d *= ancho de banda requerido para cada demanda.*
P_d *= número de diferentes túneles posibles para cada demanda d.*
λ *= porcentaje de ancho de banda de cada demanda que diferencia una tecnología de la otra de acuerdo con la longitud del paquete.*
ε*= límite más bajo sobre la fracción de un flujo sobre un camino.*
C_e *= capacidad del enlace.*
δ_{edp}*= indicador camino-enlace.*
Variables:
r *= máximo número de túneles sobre todos los enlaces. Este es un entero.*
X_{dp} *= Fracción de volumen de demanda d a ser llevada sobre un túnel p. Esta es una variable continua no negativa.*
U_{dp} *= Selecciona un túnel (=1) si el límite más bajo es satisfecho y es cero de lo contrario. Esta es una variable binaria.*

Tabla 22. Formulación MIP

Objective:

$$Mininize_{x,u,r} \quad F = r \qquad \text{EC.22}$$

Restricciones:

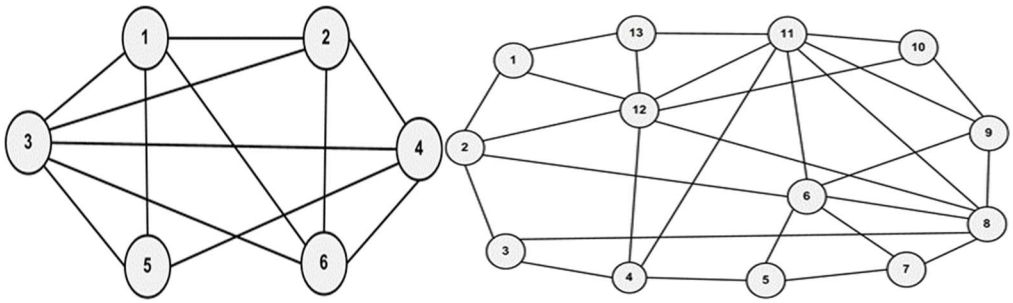
$$\sum_p X_{dp} = 1 \qquad \text{EC.23}$$

$$\sum_d h_d \lambda \sum_p \delta_{edp} X_{dp} \leq C_e \qquad \text{EC.24}$$

$$\varepsilon U_{dp} \leq \lambda\, h_d X_{dp} \qquad \text{EC.25}$$

$$X_{dp} \leq U_{dp} \qquad \text{EC.26}$$

$$\sum_d \sum_p \delta_{edp}\, U_{dp} \leq r \qquad \text{EC.27}$$



a) b)

Figura 27. Topologías de prueba

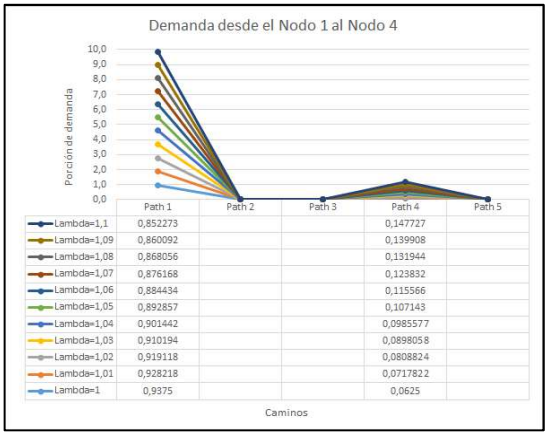
4.4.2 Resultados

Como se mencionó anteriormente, con el fin de evaluar PSA-TE6 y compararlo con el rendimiento de MPLS, se asume que el porcentaje de ancho de banda de cada demanda que diferencia una tecnología de la otra, con respecto a la longitud del paquete, está representada como λ. Se asigna ($\lambda = 1$) para PSA-TE6 y ($\lambda > 1$) para MPLS. Como se puede apreciar en la figura 26, los paquetes de PSA-TE6 son más cortos que los paquetes MPLS

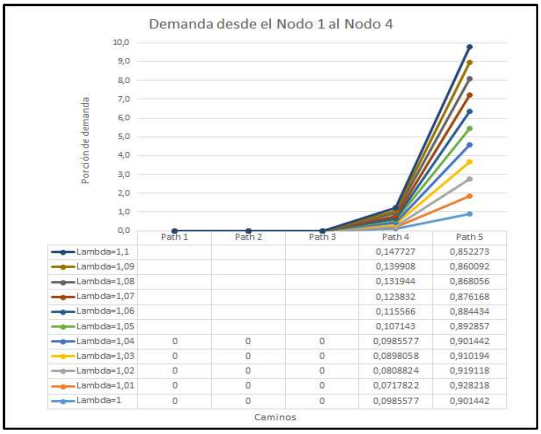
debido a la cabecera adicional en redes IP/MPLS. Para propósitos de evaluación, se asignan varios valores a λ, para analizar el efecto de la longitud del paquete en la distribución de tráfico sobre los caminos establecidos para comparar ambas tecnologías.

Las topologías usadas en esta evaluación son mostradas en la figura 27. Sobre ambas topologías se evalúa la distribución de tráfico mediante la variación del número de demandas de 1 a 5, las cuales son enviadas desde un nodo de ingreso a un nodo de egreso para diferentes valores de λ. Se simula la distribución del camino para $\lambda=1$ hasta $\lambda=1.1$ con pasos de 0.01 en cada ejecución. En cada experimento, cada demanda puede ser distribuida en 5 posibles caminos candidatos.

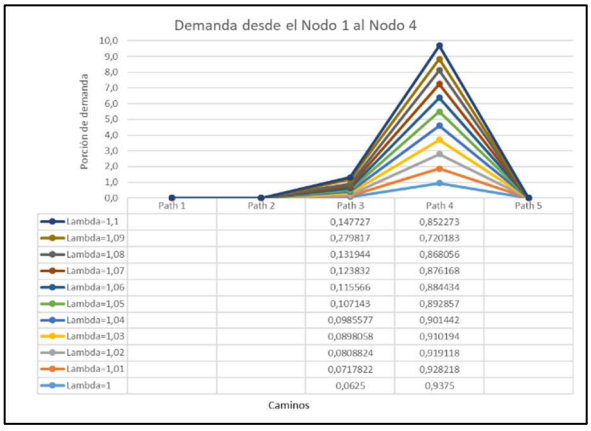
Los resultados para las redes de 6 y 13 nodos, y para valores de demandas 1,3 y 5, son mostrados en las figuras 28 y 29 respectivamente. Se puede observar que en ambas tecnologías la distribución de tráfico es similar para el mismo número de demandas cuando se varía los valores de λ desde 1 a 1.1.



a)



b)



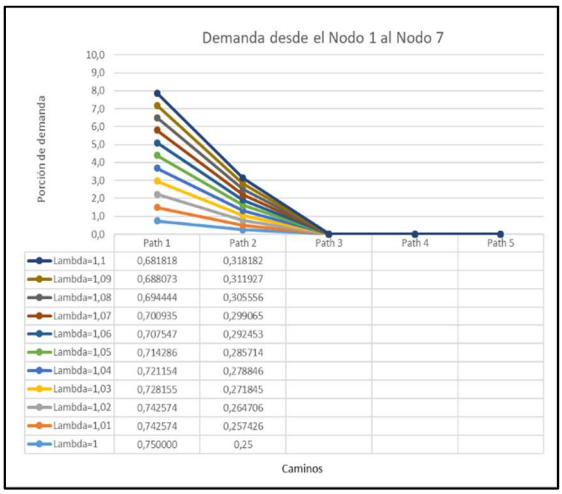
c)

Figura 28. Distribución de tráfico sobre varios caminos en una red de 6 nodos para 1, 3 y 5 demandas de tráfico.

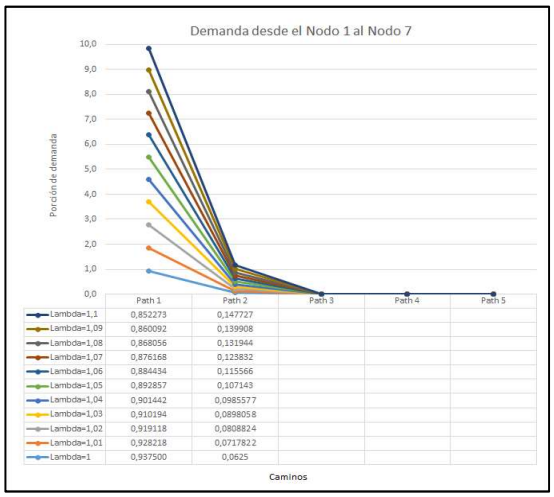
Por ejemplo, en la figura 28, en la red de 6 nodos, para tres demandas, la distribución de tráfico en la demanda de N1(nodo 1) a N4(nodo 4), es similar para varios valores de λ. Similar comportamiento puede ser observado en la figura 29, para tres demandas en la red de 13 nodos, para la demanda desde N1 a N7. Aunque los caminos seleccionados son diferentes para cada topología de red, no se encuentran diferencias en la proporción de distribución de tráfico en ambas topologías para varios números de demandas, mientras el valor de demanda va cambiando. Es así que tales distribuciones de tráfico permanecen en la misma proporción de variación de λ (1% a 10% de la ocupación de un enlace o camino).

Para soportar el rango de λ ($\lambda=1$ *to* $\lambda=1.1$), se toma el servicio de VoIP como un ejemplo para el peor caso en la diferencia de longitud del paquete entre MPLS y PSA-TE6. Para calcular la longitud de un paquete típico para este servicio se suman: para un codec típico G.729, la longitud del paquete es 108 bytes, se adicionan cabeceras, 40 bytes de la cabecera IPv6, más 8 bytes de la cabecera UDP y 12 bytes de la cabecera RTP, si se adiciona la cabecera MPLS de 4 bytes se tiene un total de 172 bytes para el caso de $\lambda > 1$. Para PSA-TE6, no se tiene en cuenta la cabecera MPLS, entonces serían 168bytes ($\lambda = 1$), lo cual es el 2.3%, menos que el caso MPLS. Esto es $\lambda=1.023$ para MPLS.

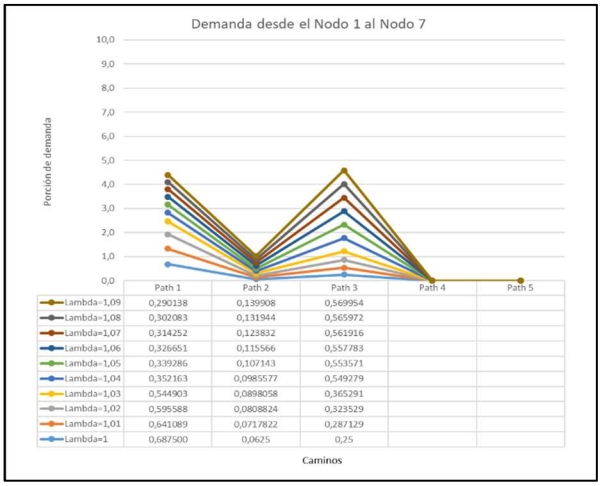
Como conclusión, se ha demostrado que PSA-TE6 tiene el mismo comportamiento que MPLS en un escenario de balanceo de carga cuando el número de túneles sobre los enlaces es optimizado. Por otro lado, con PSA-TE6, las redes IPv6 no requieren otra tecnología para soportar ingeniería de tráfico. Además, las redes IPv6 ocuparían menos ancho de banda que MPLS.



a)



b)



	Path 1	Path 2	Path 3	Path 4	Path 5
Lambda=1,09	0,290138	0,139908	0,569954		
Lambda=1,08	0,302083	0,131944	0,565972		
Lambda=1,07	0,314252	0,123832	0,561916		
Lambda=1,06	0,326651	0,115566	0,557783		
Lambda=1,05	0,339286	0,107143	0,553571		
Lambda=1,04	0,352163	0,0985577	0,549279		
Lambda=1,03	0,544903	0,0898058	0,365291		
Lambda=1,02	0,595588	0,0808824	0,323529		
Lambda=1,01	0,641089	0,0717822	0,287129		
Lambda=1	0,687500	0,0625	0,25		

c)

Figura 29. Distribución de tráfico sobre varios caminos en una red de 13 nodos para 1, 3 y 5 demandas de tráfico.

4.5 EVALUACIÓN DE PSA-TE6 EN IPv6 MÓVIL JERÁRQUICO

En esta sección, se presenta la evaluación de la propuesta de integración de PSA-TE6 con IPv6 móvil jerárquico, HMIPv6. Los principios de diseño de (HMIPv6+PSA-TE6) fueron explicados en la sección 3.7. En primera instancia, se presenta el análisis del retardo de paquetes para un solo nodo o router y luego se orienta la evaluación al análisis de tiempos con diagramas de mensajes para el protocolo HMIPv6 operando en su forma estándar como en combinación con PSA-TE6.

4.5.1 Evaluación del retardo de paquetes

La evaluación del retardo de paquetes se presenta para dos escenarios, uno con solo HMIPv6 y el otro con la integración de HMIPv6+PSA-TE6. Para ambos casos, el retardo de paquetes se ha definido como la suma de los tiempos de conmutación del paquete T_{ps}, el tiempo en cola T_q y el tiempo de transmisión del paquete T_{pt}, como se expresa en la ecuación 28.

$$T_{total} = T_{ps} + T_q + T_{ptx} \qquad \text{EC. 28}$$

El tiempo de conmutación se ha definido como la suma de los tiempos de lectura de la cabecera en routers T_{hr} y el tiempo de búsqueda en tabla de envío T_l para luego enviar el paquete por la interfaz de salida (ecuación 29).

$$T_{ps} = T_{hr} + T_l \quad \text{EC. 29}$$

Los tiempos de lectura de la cabecera se definen como el número de accesos necesarios para leer los datos de la cabecera multiplicado por el tiempo de latencia para operaciones de lectura y escritura de la memoria SRAM. Para el caso del protocolo HMIPv6 en su operación estándar, el número de accesos necesarios $N_{hr\text{-}HMIPv6}$, se determina teniendo en cuenta que el enrutamiento se hace como en IPv6, es decir mediante la dirección destino, por tanto, es necesario leer toda la cabecera de 40 bytes. En el caso de la integración de HMIPv6+PSA-TE6, el número de accesos necesarios $N_{hr\text{-}HMIPv6+PSA\text{-}TE6}$, es menor debido a que solo es necesario leer los 64 bits iniciales de la cabecera, los cuales contienen los campos de Etiqueta de Flujo , que es el campo importante para la conmutación de paquetes mediante etiquetas de flujo IPv6, y los demás campos como DS (*Differencitated Services,* Servicios Diferenciados), *Next Header* (próxima cabecera) y *hop limit* (Límite de saltos), como es mostrado en la sección 3.3 figura 10. El tiempo de lectura de la cabecera, se determina mediante las ecuaciones 30 y 31 para HMIPv6 y HMIPv6+PSA-TE6 respectivamente.

$$T_{hr-HMIPv6} = N_{hr-HMIPv6} * L_{SRAM} \quad \text{EC.30}$$

$$T_{hr-HMIPv6+PSA-T} = N_{hr-HMIPv6+PSA-} * L_{SRAM} \quad \text{EC. 31}$$

Por otro lado, para el tiempo de búsqueda en la tabla se ha tenido en cuenta que, en HMIPv6 se hace mediante *matching* documentada en [177], [178]. Esto implica que la probabilidad de éxito tiene una distribución geométrica. Por tanto, el tiempo medio de búsqueda viene determinado por el número medio de lecturas de la tabla (lecturas en SRAM). El número medio de intentos para una distribución geométrica se determina como ($1/P_{éxito}$) y la probabilidad de éxito se obtiene como $P_{éxito}=1/N_e$, donde N_e es el número de entradas en la tabla. De acuerdo con lo anterior, el tiempo de búsqueda en la tabla de envío para HMIPv6 está expresado en la ecuación 32.

$$T_{l-HMIPv6} = Ne * L_{SRAM} \quad \text{EC.32}$$

Para HMIPv6+PSA-TE6, la búsqueda se hace mediante *tagging*, el cual es usado para conmutación por etiquetas como está descrito en [177] [178], teniendo en cuenta que el enrutamiento en PSA-TE6 se hace mediante intercambio de etiquetas IPv6. En *tagging*, la búsqueda se hace mediante un campo explícito, un campo que contiene el índice que apunta a la tabla, en este caso el número de accesos de memoria N_A, es de una lectura y una escritura, debido al intercambio de valores de etiquetas, es decir, $NA = 2$ accesos de memoria. El tiempo de búsqueda en la tabla para HMIPv6+PSA-TE6, está expresado en la ecuación 33.

$$T_{l-HMIPV6+PSA} = N_A * L_{SRAM} \quad \text{EC.33}$$

El tiempo en cola depende de la propiedad de llegada estocástica de los paquetes y de la tasa de servicio de los paquetes[18]. Se asume para este análisis, que el tiempo en cola se comporta como una cola M/M/1, este supuesto para routers de Internet ya ha sido justificado en [179] y [18]. De acuerdo con lo anterior, es posible asumir que la llegada de paquetes a un enlace sigue un proceso de *Poisson*, con una tasa de llegada (λ) y una tasa de servicio de paquetes (μ) con un tiempo entre paquetes distribuido exponencialmente. Si se conoce la velocidad del enlace *B* en bits/segundos, y *L* como longitud del paquete promedio (bits), entonces $\mu=B/L$ *paquetes/segundos* [18]. Por tanto, el tiempo de la cola M/M/1 promedio está dada por la ecuación 34.

$$T_q = \frac{1}{\mu-\lambda} = \frac{1}{\frac{B}{L}-\lambda} \quad \text{EC.34}$$

Para varias comunicaciones, λ_i es la tasa de servicio por cada comunicación y *n,* el número de comunicaciones, la expresión queda como en la ecuación 35.

$$T_q = \frac{1}{\frac{B}{L}-(n*\lambda_i)} \quad \text{EC. 35}$$

Finalmente, el tiempo de transmisión se refiere al tiempo para enviar bits por el enlace, está determinado mediante la velocidad del enlace *B* en bits por segundo y la longitud del paquete *L* en bits [18]. El tiempo de transmisión está expresado en la ecuación 36.

$$T_{ptx} = \frac{L}{B} \quad \text{EC. 36}$$

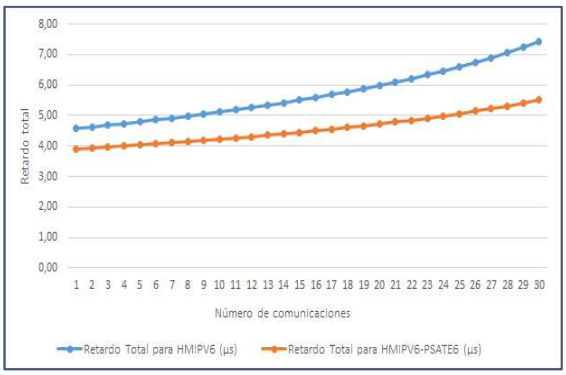
4.5.1.1 Resultados

En la tabla 19, se describen los valores de los datos utilizados en esta evaluación, para cada escenario HMIPv6 y HMIPv6+PSA-TE6. Para el tiempo lectura/escritura de memoria SRAM, se asume que los routers trabajan con procesadores de red IXP2800 (*Intel® IXP2800 Network Processor*) [180].

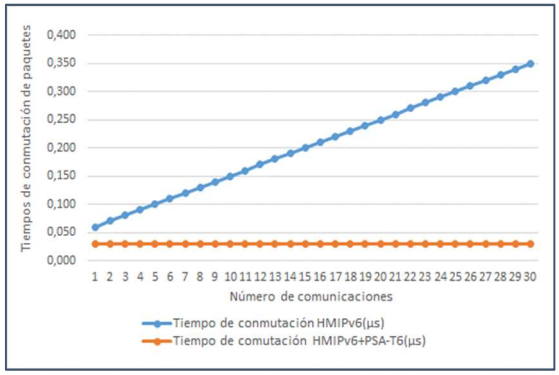
Tabla 23. Parámetros

Parámetro	*Definición*	*valor*
$N_{hr\text{-}HMIPv6}$	*Número de lecturas de 64 bits de los datos de la cabecera IPv6 en HMIPv6 (40 Bytes)*	*5*
$N_{hr\text{-}HMIPv6+PSA\text{-}TE6}$	*Número de lecturas de 64 bits de los datos de la cabecera IPv6 en HMIPv6+PSA-TE6 (64 bits)*	*1*
N_e	*Número de entradas en la tabla para HMIPv6 (Matching)*	*1,2,3….30*
N_A	*Número de accesos (tagging)*	*2*
L_{SRAM}	*Latencia de memoria RAM para lecturas en procesadores IXP2800*	*10ns*
n	*Número de comunicaciones*	*1,2,3….30*
L_{HMIPv6}	*Longitud del paquete (200Bytes de carga útil + cabecera) para el caso del protocolo HMIPv6, teniendo paquetes que van desde el MAP al MN*	*280Bytes*
$L_{HMIPv6+PSA\text{-}TE6}$	*Longitud del paquete (200Bytes de carga útil + cabecera) para el caso del protocolo HMIPv6+PSA-TE6, teniendo paquetes que van desde el MAP al MN*	*240Bytes*
B	*Ancho de banda del enlace alambrado*	*1Gbps*
λ_i	*Tasa de llegadas de paquetes*	*8000 paquetes/seg*

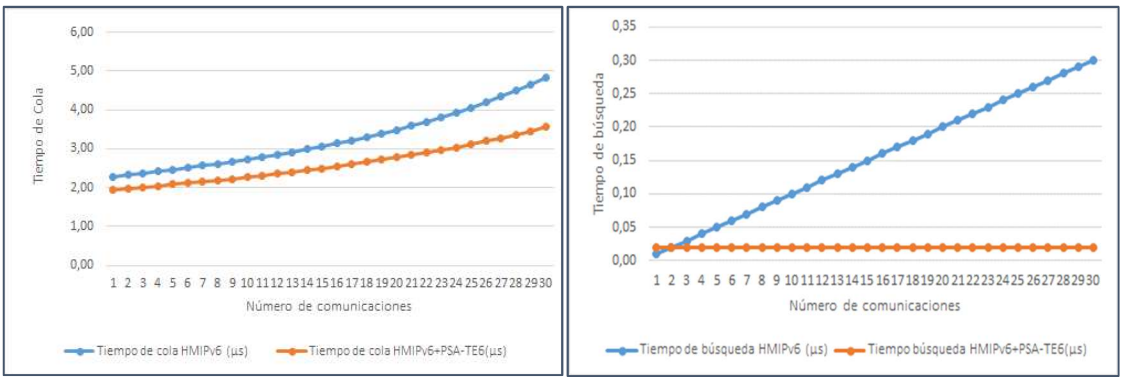
En la figura 30, se muestran los resultados del tiempo total del retardo de los paquetes, de acuerdo con las ecuaciones 28-36. Para ambos escenarios, HMIPv6 y HMIPv6+PSA-TE6, se ha calculado retardo variando el número de comunicaciones desde 1 hasta 30. En la figura 30.a, se presenta el retardo total de acuerdo con la ecuación 28. El resultado demuestra que la propuesta HMIPv6+PSA-TE6, tiene un retardo total menor tanto para una comunicación como a medida que aumenta el número de comunicaciones, en comparación con HMIPv6. Esto mismo se puede apreciar al comparar el tiempo de conmutación de paquetes (figura 30.b), el tiempo en cola (figura 30.c) y el tiempo de búsqueda en la tabla (figura 30.d). Los resultados permiten constatar que la propuesta PSA-TE6, proporciona una mejora en cuanto al retardo de los paquetes con respecto al protocolo en su forma estándar. Además, este análisis del retardo no solo aplica para redes móviles como fue presentado en este documento, sino también en redes fijas. El modelo presentado puede ser utilizado tanto para redes fijas como móviles, para lo cual PSA-TE6 presenta una mejora considerable.



(a)



(b)



(c) (d)

Figura 30. Resultados de tiempos de retardo de paquetes. (a) Tiempo de retardo total, (b) Tiempo de conmutación de paquetes (c) Tiempo en cola (d) Tiempo de búsqueda en la tabla.

4.5.2 Análisis de Tiempos con Diagramas de Mensajes Para HMIPv6 y HMIPv6+PSA-TE6

En esta sección, se presenta el análisis de la operación del protocolo de movilidad HMIPv6 y la integración de HMIPv6 con PSA-TE6. Este análisis se realiza mediante diagramas de mensajes y calculando los tiempos de cada mensaje de acuerdo con las ecuaciones 28-36 descritas en la sección anterior. Para ambos escenarios se describe su operación teniendo en cuenta los mensajes de señalización y el envío de datos. Para cada mensaje de señalización y envío de datos se calcula el tiempo teniendo cuenta arquitectura de la figura 14, la cual se toma como ejemplo. También se presenta el análisis, cuando ocurre handover utilizando el protocolo Fast Handover para ambos casos con el protocolo HMIPv6 estándar y con la integración HMIPv6+PSA-TE6.

4.5.2.1 Análisis de tiempos con diagramas de mensajes en HMIPv6

En la operación de HMIPv6, cuando un nodo móvil (MN) entra a un dominio MAP recibirá un anuncio de router (*RAds, Router Advertisement*), conteniendo información acerca de MAPs locales. El MN necesita configurar dos direcciones CoAs (*Care of Address*): una RCoA (*Regional CoA*) sobre la región del MAP y una sobre el enlace LCoA (*Local CoA*). El MN enlaza su ubicación actual (LCoA) con la dirección sobre la subred del MAP (RCoA) [139].

Luego, el MN envía una actualización de enlace local (*Local Binding Update, LBU*) al MAP. La actualización de enlace local se realiza tal como está definido en [138], e incluye la RCoA del nodo móvil en la opción de dirección local. La LCoA es usada como la fuente del BU. Este BU enlazará la RCoA del nodo móvil (similar a una *home address*) a su LCoA. El MAP actuando como un agente local, (*Home Agent, HA*) luego contestará con un reconocimiento de la

actualización (*Binding Acknowledgement, BAck*) al MN. Este reconocimiento identifica la actualización ya sea como exitosa o contiene el apropiado código de falla [138].

Luego del registro con el MAP, el nodo móvil debe registrar su nueva RCoA con su HA, mediante el envío de una actualización global (*Global Binding Update*, *GBU*) que especifica el enlace (RCoA-*Home Address*), como en IPv6 móvil [138]. El nodo móvil puede también enviar un GBU similar a su nodo correspondiente actual [139].

Finalmente, todos los paquetes direccionados a la RCoA del nodo móvil son interceptados por el MAP y tunelizados hacia la LCoA del nodo móvil. El nodo móvil desencapsulará los paquetes y los procesará de una manera normal. De igual forma, todos los paquetes enviados por el nodo móvil son tunelizados hacia el MAP. El MAP desencapsulará los paquetes y los enviará al CN [139].

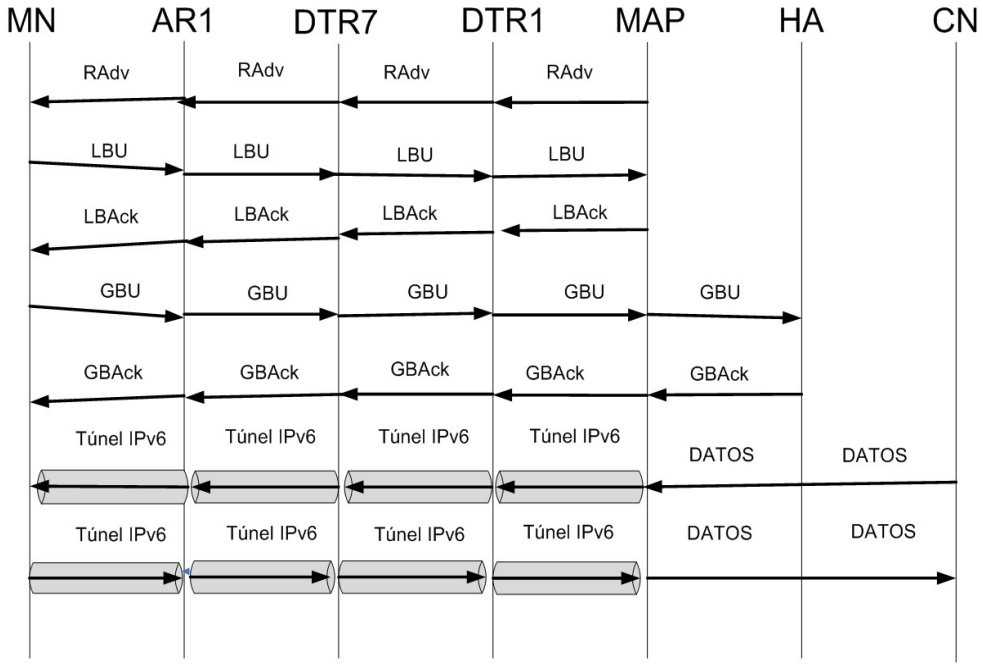


Figura 31. Diagrama de mensajes de HMIPv6.

En la figura 31, se muestra el diagrama de mensajes de la operación de HMIPv6. Para el análisis de operación del protocolo en su forma estándar. En el diagrama se puede apreciar el intercambio de mensajes de señalización y de envío de datos. Es importante notar aquí que, para el protocolo estándar, los paquetes enviados desde el nodo correspondiente son interceptados por el MAP y tunelizados mediante encapsulación IPv6, lo cual añade una cabecera de 40 bytes a los paquetes.

Tiempo de operación de HMIPv6. Es determinado por el tiempo de señalización $T_{HMIPv6\text{-}S}$ más el tiempo de envío del paquete T_{PF} como es expresado en la ecuación 37.

$$T_{HMIPv6-Op} = T_{HMIPv6-S} + T_{PF} \quad \text{EC. 37}$$

Análisis del tiempo de señalización de HMIPv6. Es calculado teniendo en cuenta el tiempo de conmutación, el tiempo en cola y el tiempo de transmisión de los mensajes de señalización los cuales son: Anuncios de Router (RAdvs), enviados por el MAPs hacia el MN; los mensajes de actualización global y local (GBU y LBU), enviados por el MN hacia el MAP y HA para actualizar su ubicación; mensajes de reconocimiento de actualización de enlace local y global desde el MAP hacia el MN en respuesta a los LBU y GBU (ver figura 31). El tiempo de mensajes de señalización total está expresado en la ecuación 38. La longitud de cada mensaje y los valores de ancho de banda tanto de los enlaces alambrados como inalámbricos se encuentran en la tabla 24.

$$T_{HMIPv6-S} = T_{RAdvs} + T_{LBU} + T_{LBACK} + T_{GBU} + T_{GBACK} \quad \text{EC.38}$$

Análisis del tiempo de envío de paquetes en HMIPv6. Es determinado por el tiempo de envío de paquetes desde el CN al MAP y luego del MAP al MN, el cual está expresado en la ecuación 39. Estos tiempos se hacen por separado teniéndose cuenta que en el trayecto CN-MAP el enrutamiento funciona de una manera normal, mientras que desde MAP al MN se hace tunelización IPv6 lo cual agrega una cabecera de 40 bytes al paquete.

$$T_{PF} = T_{CN-MAP} + T_{MAP} \quad \text{EC.39}$$

Tabla 24. Mensajes usados en HMIPv6 y FHMIPv6 [181]

Parámetro	*Valor*	*Tamaño Total incluyendo la cabecera IPv6*	*Significado*
RAdv	*64B*	*104B*	*Router Advertisement*
LBU	*90B*	*136a*	*Local Binding Update*
LBAck	*82B*	*128[a]*	*Local Binding Ack.*
GBU	*90B*	*136a*	*Global Binding Update*
GBAck	*82a*	*128a*	*Global Binding Ack.*
FBU	*72B*	*118B*	*Fast Binding Update*
HI	*72B*	*118B*	*Handover Initiate*
HAck	*32B*	*78B*	*Handover Ack.*
FBAck	*32B*	*78B*	*Fast Binding Ack.*
RtSolPr	*24B*	*64B*	*Router solicitation for Proxy Advertisement*
PrRtAdv	*104B*	*144B*	*Proxy Router Advertisement*
Bi	*600Mbps*		*Ancho de banda del enlace alambrado*
Ba	*1Gbps*		*Ancho de banda del enlace inalámbrico*
L	*200B*		*Longitud de los paquetes de datos.*

4.5.2.2 Análisis de tiempos con diagramas de mensajes para HMIPv6 + PSA-TE6

En la figura 32, se muestra el diagrama de mensajes de la propuesta de integración de HMIPv6 y PSA-TE6. En HMIPv6+PSA-TE6, todos los mensajes de señalización de HMIPv6 están presentes. La diferencia está en la metodología usada para el intercambio de mensajes de señalización y envío de datos entre el MAP y ARs.

Como se mencionó anteriormente la propuesta PSA-TE6 usa caminos conmutados por etiquetas de flujo IPv6 (6DLSPs), entre el MAP y los routers de acceso (ARs). En la integración HMIPv6 + PSA-TE6, tanto el intercambio de mensajes de señalización necesarios para soportar movilidad como para el envío de datos entre el MAP y MN, se hace mediante el establecimiento de 6DLSPs entre el MAP y el AR (conectado con el MN), en lugar de utilizar tunelización IPv6 como en HMIPv6 estándar. Por tanto, todos los paquetes enviados por CNs son interceptados por el MAP y enviados a los ARs mediante los 6DLSPs.

El MAP y los ARs son routers frontera de ingreso y egreso (*Ingress/Egress PSA-TE6 Domain Edge Routers*, 6DERs) y los routers tránsito entre el MAP y ARs son routers que conmutan los paquetes mediante intercambio de etiquetas de flujo IPv6 (*PSA-TE6 Domain Transit Router*, 6DTRs), (ver figura 32). Los caminos conmutados por etiquetas, 6DLSPs, son establecidos por RSVP-TE como fue explicado en el capítulo 4. Por tanto, los mensajes PATH and RESV de RSVP-TE están presentes, los cuales son usados para distribuir etiquetas IPv6 y establecer 6DLSPs antes del envío de paquetes. Se asume que los 6DLSPs para el intercambio de señalización HMIPv6 están previamente establecidos.

Para intercambiar etiquetas IPv6 en HMIPv6+PSA-TE6, los 6DTRs solamente necesitan leer los primeros 64 bits de la cabecera IPv6, donde están ubicados los campos necesarios para enrutar los paquetes de acuerdo con la propuesta PSA-TE6. En la figura 32, se muestra el diagrama de mensajes de la operación HMIPv6+PSA-TE6.

Análisis del tiempo de operación de HMIPv6+ PSA-TE6. Está determinado por el tiempo de señalización, más el tiempo de establecimiento de los 6DLSPs mediante RSVP-TE, más el tiempo de envío de los paquetes, como está expresado en la ecuación 40.

$$T_{HMIPv6+PSA-TE-} = T_{HMIPv6+PSA-TE6-S} + T_{RSVP-TE-} + T_{PF_{HMIPv6+PSA-TE6}} \quad \text{EC.40}$$

Análisis del tiempo de señalización en HMIPv6+PSA-TE6. Esta expresión es muy similar a la obtenida para el tiempo de señalización de HMIPv6, con la diferencia que para HMIPv6+PSATE6 los mensajes intercambiados entre el MAP y los ARs son transmitidos mediante caminos conmutados por etiquetas, y por tanto, con PSA-TE6 no es necesario leer toda la cabecera para el enrutamiento debido a que solo son necesarios los primeros 64 bits

de donde se extrae el valor del campo Etiqueta de Flujo. El tiempo de señalización para HMIPv6+PSA-TE6 está expresado en la ecuación 41.

$$T_{HMIPv6+PSA-TE} = T_{Advs} + T_{LBU} + T_{LBACK} + T_{GBU} + T_{GBACK} \quad \text{EC. 41}$$

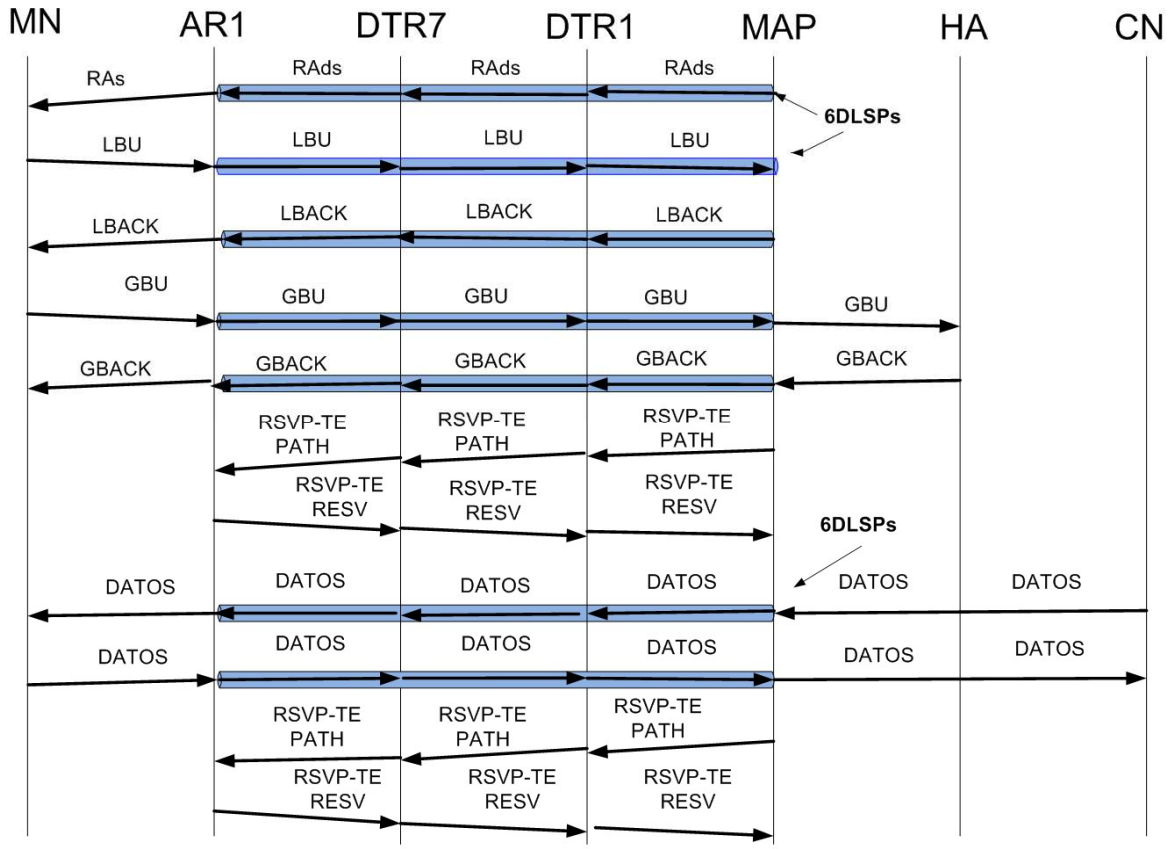


Figura 32. Diagrama de mensajes de HMIPv6 + PSA-TE6.

Análisis del tiempo de establecimiento de 6DLSPs. PSA-TE6 usa RSVP-TE, por tanto, los mensajes PATH y RESV son usados para distribuir etiquetas IPv6 y establecer 6DLSPs. El tiempo de establecimiento de 6DLSPs, está expresado en la ecuación 42.

$$T_{RSVP-TE-S} = T_{PATH} + T_{RESV} \quad \text{EC.42}$$

Análisis del tiempo de envío de paquetes con HMIPv6+PSA-TE6. Está determinado por el envío de paquetes desde CN hasta el MAP y luego desde el MAP hasta el MN mediante caminos conmutados por etiquetas (ecuación 43).

$$T_{\underset{HMIPv6+PSA-TE6}{PF}} = T_{CN-MA} + T_{MAP-MN} \quad \text{EC.43}$$

4.5.2.3 Análisis de tiempos con diagramas de mensajes en HMIPv6 con Fast Handover

La combinación del protocolo *Fast Handover* [140] y HMIPv6 [139], abreviado FHMIPv6, permite la anticipación del traspaso o handoff en capa 3, en tal forma que, el MAP podría redireccionar el tráfico a la nueva dirección del nodo móvil antes de que el nodo móvil cambie de ubicación. El intercambio de mensajes necesarios cuando un handover local (intra MAP) es desarrollado se describe a continuación.

Inicialmente, cuando el MN está en la cobertura del AR previo (AR1, en la figura 33), el MAP envía los paquetes a el nodo móvil a través de un túnel IPv6. Cuando el MN detecta un movimiento dentro del mismo dominio MAP, inicia el proceso de señalización del protocolo Fast Handover.

Luego, el MN envía un mensaje llamado RtSolPr (*Router Solicitation for Proxy Advertisement*) al previo AR para solicitar información de un handover potencial. El previo AR (AR1) responde mediante un mensaje PrRtAdv (*Proxy Router Advertisement*) informando al nodo móvil acerca de los enlaces vecinos. Luego el nodo móvil envía un mensaje FBU (*Fast Binding Update*) al MAP para darle instrucción de direccionar el tráfico al nuevo AR (AR2).

El MAP debe confirmar con el nuevo AR (AR2), si la CoA solicitada por el MN es válida, esto es realizado a través del mensaje HI (*Handover Initiate*). Si la dirección es válida, el MAP envía la confirmación por medio de un mensaje FBAck (*Fast Binding Acknowledgment*) indicando que el FBU fue recibido y se comenzarán a transmitir los paquetes al AR2.

Finalmente, cualquier paquete enviado por el nodo correspondiente al nodo móvil es interceptado por el MAP y tunelizado hacia el nodo móvil a través de AR2 (el nuevo AR). En la figura 33, se muestra el diagrama de mensajes.

Análisis del tiempo de operación de FHMIPv6. Es determinado por el tiempo de señalización más el tiempo de envío de paquetes durante el handover, como es mostrado en la ecuación 44.

$$T_{FHMIPv6-HO} = T_{FHMIPv6-HO-S} + T_{\underset{FHMIPv6-H}{PF}} \quad \text{EC. 44}$$

Análisis del tiempo de señalización de FHMIPv6. Está determinado por los tiempos de los mensajes que participan en la señalización de FHMIPv6, el cual está expresado en la ecuación 45.

$$T_{\substack{HMIPv \\ HO-S}} = T_{RtSolPr} + T_{PrRtAdv} + T_{FBU} + T_{HI} + T_{HAck} + T_{FBAck} \quad \text{EC. 45}$$

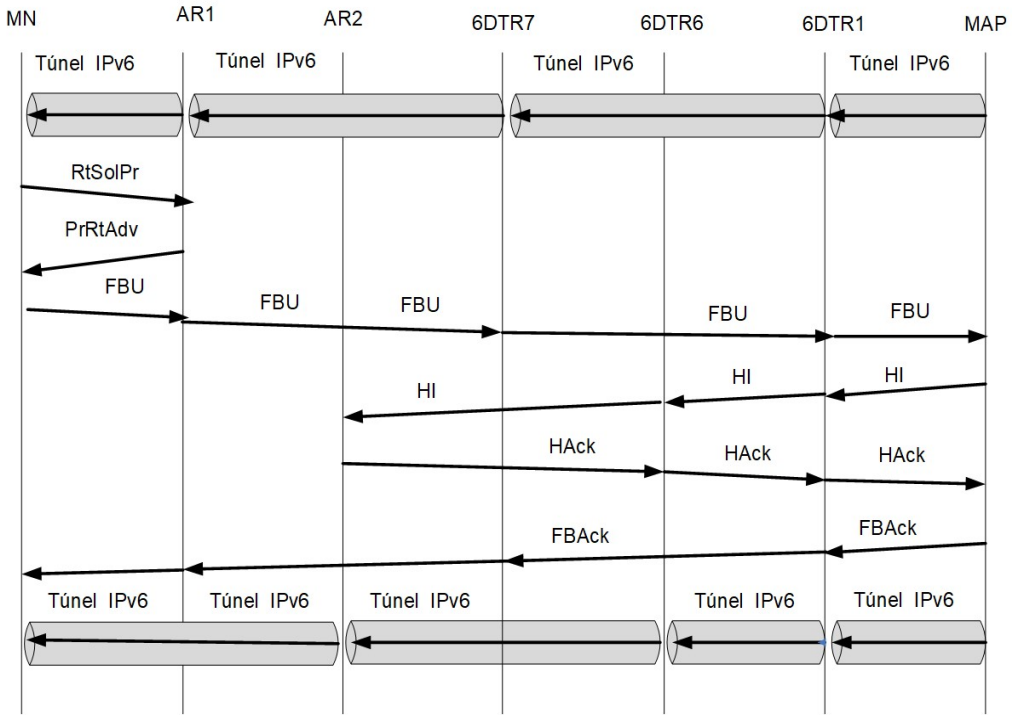


Figura 33. Diagrama de mensajes de FHMIPv6.

Análisis del envío de paquetes en FHMIPv6. En este análisis, se toma en cuenta dos momentos, cuando se envía paquetes al previo AR y cuando se hace el envío al nuevo AR. Este tiempo es expresado en la ecuación 46.

$$T_{PF-FHMIPv6} = T_{MAP-AR1-MN} + T_{MAP-AR2-MN} \quad \text{EC. 46}$$

4.5.2.4 Análisis de tiempos con diagramas de mensajes en HMIPv6 + PSA-TE6 con Fast Handover

El diagrama de mensajes de HMIPv6+PSA-TE6 con fast handover o FHMIPv6+PSA-TE6, es mostrado en la figura 34. Como se puede apreciar, cuando se establece la confirmación para el envío de paquetes mediante el nuevo AR, AR2, se realiza el establecimiento del camino conmutado por etiquetas IPv6, mediante RSVP-TE usando los mensajes PATH y RESV. Para este caso tanto los mensajes de señalización de FHMIPv6 como el envío de datos se realiza mediante 6DLSPs entre el MAP y el AR (al cual está conectado el nodo móvil).

Análisis del tiempo de operación de FHMIPv6 + PSA-TE6. Es determinado por el tiempo de señalización de FHMIPv6 más el tiempo de establecimiento de 6DLSPs (señalización de RSVP-TE) y el tiempo de envío de paquetes, como se expresa en la ecuación 47.

$$T_{FHMIPv6+PSA-TE6-O} = T_{\substack{FHMIPv6 \\ +PSA-TE6-S}} + T_{RSVP-TE-S} + T_{\substack{PF \\ FHMIPv6+PSA-}} \quad \text{EC.47}$$

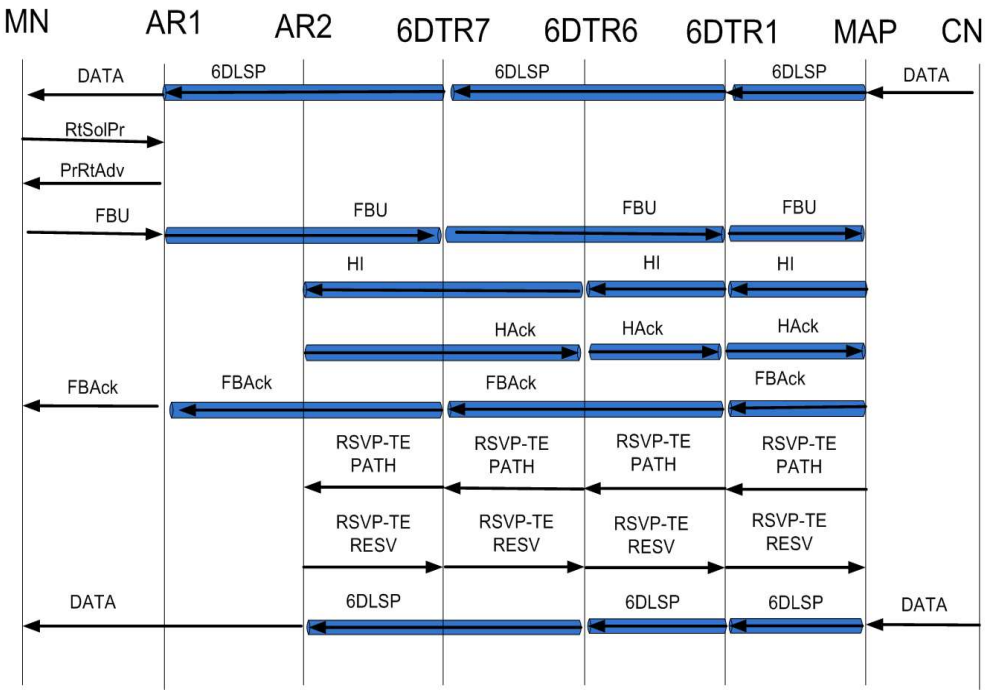


Figura 34. Diagrama de mensajes de FHMIPv6+PSA-TE6.

Análisis del tiempo de señalización de FHMIPv6+PSA-TE6. Está conformado por los tiempos de señalización de los mensajes de FHMIPv6, explicados en la sección anterior. Para FHMIPv6 + PSA-TE6, los mensajes de señalización son enviados por medio de 6DLSPs, los cuales se asumen como previamente establecidos. El tiempo de señalización de FHMIPv6+PSA-TE6, está expresado en la ecuación 48.

$$T_{\substack{FHMIPv6 \\ +PSA-TE6-}} = T_{RtSolPr} + T_{PrRtAdv} + T_{\substack{FBU \\ FHMIPv6 \\ +PSA-TE6}} + T_{\substack{HI \\ FHMIP \\ +PSA-TE6}} + T_{\substack{HAck \\ FHMIPv6 \\ +PSA-TE6}} + T_{\substack{FBAck \\ FHMIPv6 \\ +PSA-TE6}} \quad \text{EC.48}$$

Análisis del establecimiento de 6DLSPs. Para el envío de paquetes en FHMIPv6+PSA-TE6. Compuesto por los tiempos de los mensajes PATH y RESV de RSVP-TE, para la distribución de valores de etiquetas IPv6 y establecimiento de 6DLSPs (ecuación 49).

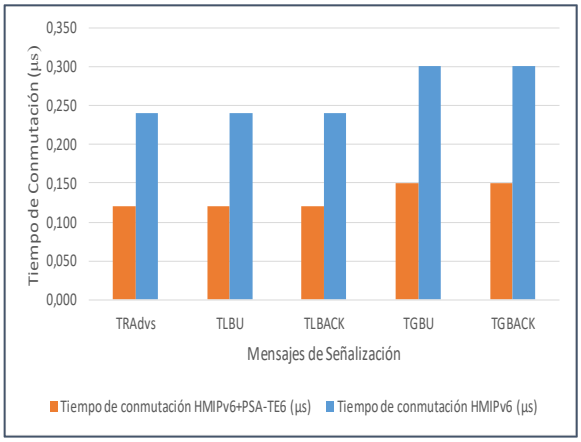
$$T_{\substack{RSVP-TE \\ Signaling}} = T_{PATH} + T_{RESV} \quad \text{EC. 49}$$

Análisis del envío de paquetes en FHMIPv6+HMIPv6. En este análisis, se toma en cuenta dos momentos, cuando se envía paquetes al previo AR y cuando se hace el envío al nuevo AR. Este tiempo es expresado en la ecuación 50.

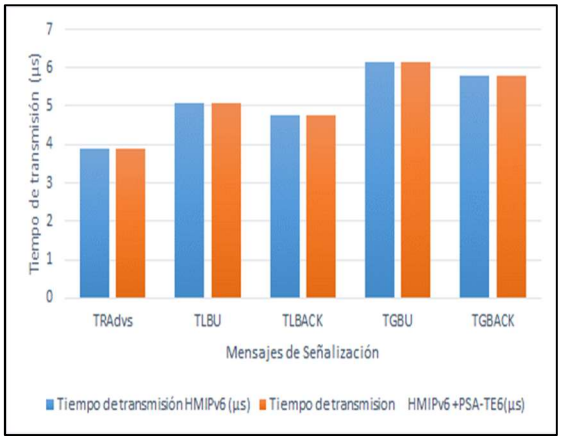
$$T_{PF-FHMIPv6+SA-TE6} = T_{MAP-AR1-MN} + T + T_{MAP-AR2-MN} \quad \text{EC. 50}$$

4.5.2.5 Resultados

En la figura 35, se muestran los resultados de los tiempos obtenidos a partir del análisis de los diagramas de mensajes. Los resultados muestran que para la propuesta PSA-TE6 integrada con HMIPv6, tiene una reducción en los tiempos de conmutación, esto es debido a que en el tiempo de conmutación interviene la búsqueda de la tabla y el tiempo de lectura de la cabecera y para cada escenario se tiene una metodología diferente de búsqueda en la tabla de acuerdo a su forma de enrutamiento, en HMIPv6 se hace mediante la dirección destino y en HMIPv6+PSA-TE6 mediante conmutación por etiquetas de flujo IPv6, como se explicó en la sección 4.5.1. En este análisis se tuvo en cuenta solo una comunicación y se nota en la figura 35a, que el tiempo de conmutación de la propuesta HMIPv6+PSA-TE6 es alrededor del 50% menor que en HMIPv6 estándar. Sin embargo, los tiempos de transmisión son prácticamente iguales, debido a que, en la transmisión de los mensajes de señalización, no hay cambios significativos, porque los mensajes no se encapsulan entre el MAP y ARs, en el protocolo estándar (ver figura 35b y 35c). Por tanto, los mismos valores de tamaño de mensajes y de velocidad de los enlaces se utilizan para ambos escenarios.



(a)



(b)

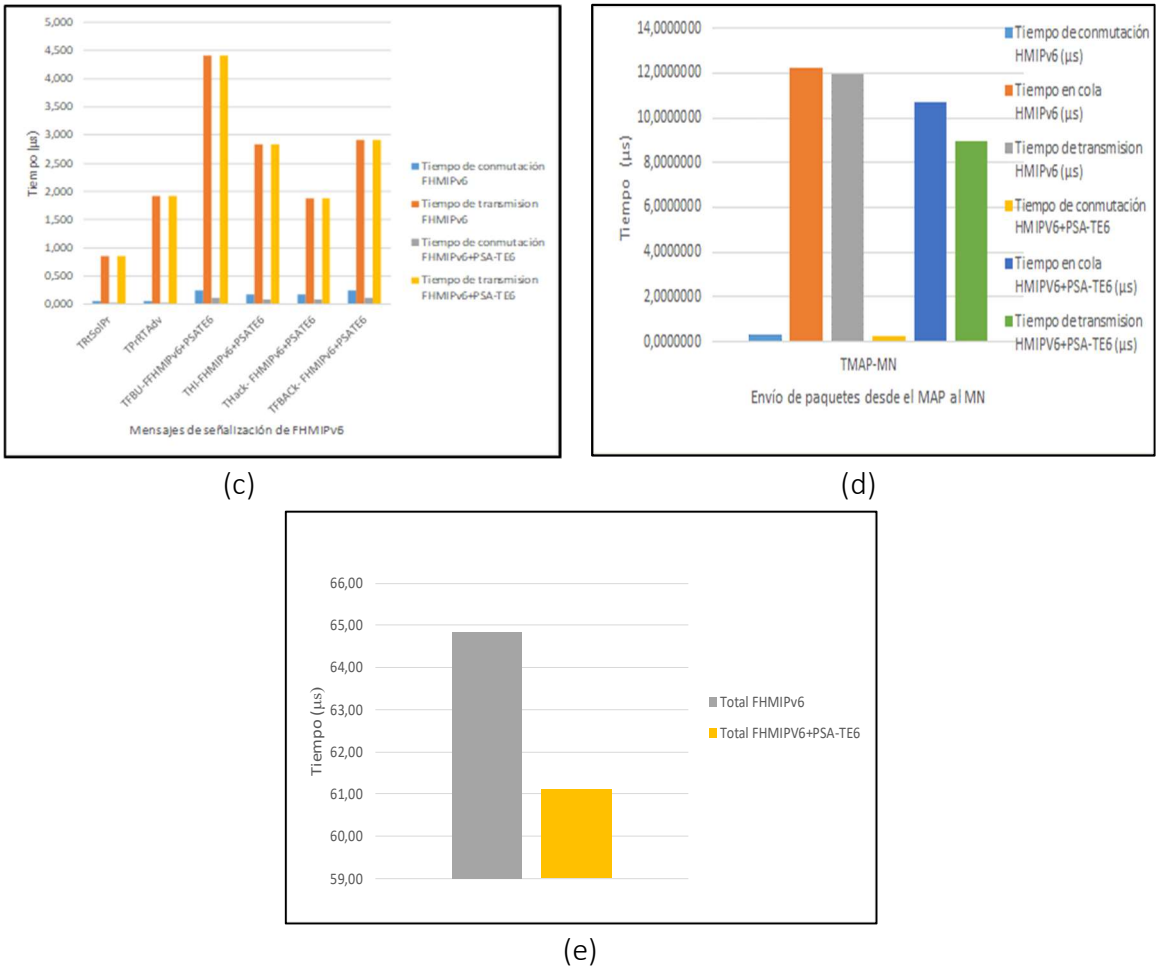


Figura 35. Resultados del análisis de los diagramas de tiempo.

En cuanto al envío de paquetes entre el MAP y el nodo móvil (MN) se observa que la propuesta HMIPv6+PSA-TE6 presenta tiempos menores en los procesos de conmutación, encolamiento y transmisión con respecto a HMIPv6, esto se puede apreciar en la figura 35d. Lo que permite constatar el beneficio de integrar HMIPv6 con la propuesta PSA-TE6.

Por otro lado, en cuanto al handover, también se aprecia una mejora en el tiempo total para la propuesta FHMIPv6 + PSA-TE6 con respecto a FHMIPv6, como se puede apreciar en la figura 35e.

4.6 PUBLICACIONES REALIZADAS DEL CAPÍTULO

- Line Yasmin Becerra Sánchez, Jhon Jairo Padilla Aguilar, "Analysis of Load Balancing for a New Approach to Support Traffic Engineering in IPv6 Networks" *Indian Journal of Science and Technology*, Vol 11(35), DOI: 10.17485/ijst/2018/v11i35/122033, September 2018.

– Line Yasmin Becerra Sánchez, Jhon Jairo Padilla Aguilar, "An Approach to support traffic engineering in IPv6 networks based on IPv6 facilities. Telecommunication Systems Modelling, Analysis, Design and Management, Springer Link, ISSN: 1018-4864 (Print) 1572-9451 (Online), DOI: 10.1007/s11235-018-00543-7.

5. CONCLUSIONES

Este documento de tesis presentó una nueva propuesta para soportar ingeniería de tráfico basada en el uso de la Etiqueta de Flujo IPv6 para la conmutación de paquetes en redes IPv6. Para la evaluación de la propuesta se realizó el análisis del espacio reducido de etiquetas, de los beneficios del apilamiento de etiquetas como solución para ahorrar espacio de etiquetas. También se analizaron los costos de procesamiento en routers, con base en el número de lecturas/escrituras y se presentó una formulación ILP en la cual se analizó la influencia de los costos de ingreso, egreso y push. Se presentó un modelo para el balanceo de carga, optimizando el máximo número de túneles sobre el enlace. Finalmente, se presentó el análisis del retardo de paquetes y el análisis de operación para el protocolo de movilidad IP, HMIPv6 y para el protocolo HMIPv6 integrado con PSA-TE6. Este análisis se hizo mediante diagramas de mensajes donde se evidenció el intercambio de mensajes de señalización y de datos para el protocolo estándar y el integrado con PSA-TE6.

Los resultados de la evaluación realizada con respecto a los costos de procesamiento muestran que PSA-TE6 tiene costos más bajos hasta un 45% menores con respecto a IP/MPLS cuando funciona en su forma base, es decir sin apilamiento de etiquetas. Para los dos mecanismos de apilamiento de etiquetas, PSA-TE6 supera a IP/MPLS cuando el apilamiento está habilitado hasta en un 40% de presencia de túneles de niveles superiores a 1 con respecto al número de comunicaciones establecidas, manteniendo un buen rendimiento en el costo de la conmutación por etiquetas.

Mediante la formulación ILP se pudo apreciar la influencia de los costos de ingreso y egreso, C_{in}, C_{eg}, en el costo total, al igual que el costo push. Se puede inferir a partir de los resultados que, en la obtención del costo mínimo, es más perjudicial tener costos de ingreso, egreso y tránsito altos porque estos costos están presentes cuando no hay apilamiento y para todos los niveles de apilamiento. Tener costos altos de push no influye tanto, debido a que el costo push solo está presente en el caso de apilamiento, el porcentaje de apilamiento puede mantenerse al mínimo, solo en algunos casos cuando sea estrictamente necesario.

También se realizó el análisis para el balanceo de carga y la minimización del número de túneles sobre cada enlace con el fin de evaluar el comportamiento de PSA-TE6 con respecto a MPLS. Los resultados mostraron que pequeñas diferencias en la longitud del paquete para ambas tecnologías (2.3% para el peor caso) no son significantes y la distribución de tráfico sobre los enlaces para ambas tecnologías es el mismo cuando el número de túneles sobre enlaces es optimizado. Por tanto, se concluye que PSA-TE6 es útil para para soportar ingeniería de tráfico y para ser implementada en redes IPv6, por tanto, al implementar PSA-TE6, no sería necesario una arquitectura MPLS. Además, las redes IPv6 ocuparían menos ancho de banda en comparación con MPLS. Es Importante resaltar que la solución PSA-TE6

propuesta satisface las recomendaciones de la IET con respecto al uso del campo la etiqueta IPv6 para control de conmutación en combinación con protocolos de enrutamiento y señalización (OSPFv3-TE and RSVP-TE respectivamente) que soporten ingeniería de tráfico.

La evaluación PSA-TE6 en combinación con el protocolo de movilidad HMIPv6 proporciona una reducción en tiempos de conmutación, tiempos en cola y tiempos de transmisión de paquetes con respecto al protocolo HMIPv6 trabando en su forma estándar. Por tanto, presenta una reducción en el retardo de los paquetes en el dominio MAP (con PSA-TE6). Las ventajas de PSA-TE6+HMIPv6 es que se elimina la tunelización IPv6 entre el MAP y el MN, y los paquetes son enviados mediante caminos conmutados por etiquetas, lo cual agiliza el proceso de conmutación y disminuye el tamaño del paquete y por tanto el ancho de banda necesario.

Es importante resaltar que las evaluaciones presentadas son fundamentales para determinar el rendimiento de los routers durante el enrutamiento de los paquetes. Igualmente se resalta que la solución PSA-TE6 propuesta cumple con las recomendaciones de la IETF para el uso de la Etiqueta de Flujo para control de conmutación. También soporta claramente que se puede usar la Etiqueta de Flujo IPv6 para conmutación de paquetes en combinación con protocolos de enrutamiento y de señalización que soportan ingeniería de tráfico para el establecimiento de caminos mediante algoritmos basados en restricciones sin la necesidad de IP/MPLS.

6. TRABAJOS FUTUROS

El diseño de PSA-TE6 abre puertas para la investigación y evaluación tanto de particularidades propias de su arquitectura como en combinación con otras arquitecturas o tecnologías. Entre las particularidades propias podría estar relacionadas con la evaluación o creación de algoritmos de enrutamiento basados en restricciones que funcionen apropiadamente con esta nueva arquitectura y propuestas relacionadas con el enrutamiento multicamino. Un asunto que puede ser interesantes es evaluar el comportamiento de PSA-TE6 bajo ráfagas de paquetes de diferentes tamaños desde la fuente al destino. Otro sería la implementación de PSA-TE6 en routers programables o NETFPGAs, donde se pueda emular su comportamiento. Algunos estudios podrían esta enfocados en evaluar como PSA-TE6 afecta los parámetros de calidad de servicio, tales como retardo de paquetes, ancho de banda, pérdida de paquetes por medios estadísticas y modelos de optimización.

Por otro lado, se puede combinar PSA-TE6 con otras tecnologías o arquitecturas nuevas, como el protocolo LISP o con el protocolo de Enrutamiento por Segmentos. Por ejemplo, con el protocolo LISP, en la propuesta que hicieron los creadores de LISP para soportar ingeniería de tráfico tanto para IPv4 como para IPv6, se puede estudiar la combinación con PSA-TE6 para agilizar la conmutación de paquetes mediante etiquetas, en el túnel LISP que se establece entre un ITR y ETR cuando se trabaja con redes IPv6. La conmutación rápida allí podría ser muy útil y contrarrestar los retardos producidos por la tunelización de LISP.

Trabajos futuros pueden estar relacionados al comportamiento de PSA-TE6 en ambientes móviles relacionados al tiempo de handover y el soporte de ingeniería de tráfico en protocolos de movilidad IP tales como Proxy Mobile IPv6. De igual manera, evaluaciones del comportamiento de PSA-TE6 en combinación con redes definidas por software tanto para redes fijas como móviles.

Finalmente, la creación de modelos de simulación y simuladores que permitan hacer evaluaciones en conjunto de las características de PSA-TE6, podría ser un trabajo futuro muy interesante.

7. REFERENCIAS

[1] Awduche D.; Chiu A.; Elwalid A.; Widjaja I.; and Xiao X., "Overview and Principles of Internet Traffic Engineering," *IETF RFC3272*, 2002.

[2] Deering S.; and Hinden R., "Internet Protocol, Version 6 (IPv6) Specification," *IETF RFC8200*, p. 40p, 2017.

[3] Postel J., "Internet Protocol," *IETF RFC791*, 1981.

[4] Amante S.; Carpenter B. and Jiang S., "Rationale for Update to the IPv6 Flow Label Specification," *IETF RFC6436*, 2011.

[5] Hu Q. and Carpenter B., "Survey of Proposed Use Cases for the IPv6 Flow Label," *IETF RFC6294*, 2011.

[6] Becerra L. Y. and Padilla J. J., "Review of Approaches for the use of the Label Flow of IPv6 Header," *IEEE Trans. Lat. Am.*, vol. 12, no. 8, pp. 1602–1607, 2014.

[7] Bradner S. and Mankin A., "The Recommendation for the IP Next Generation Protocol," *IETF RFC1752*, 1995.

[8] Deering S. and Hinden R., "Internet Protocol, Version 6 (IPv6) Specification," *IETF RFC2460*, 1998.

[9] Loshin P., "IPv6 -Theory, Protocol, and Practice," *ELSEIVER Morgan Kaufmann Publ.*, pp. 123–140, 2004.

[10] Nichols K.; Blake S.; Baker F. and Black D., "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," *IETF RFC2474*, 1998.

[11] Deering S. and Hinden R., "Internet Protocol, Version 6 (IPv6) Specification," *IETF RFC1883*, 1995.

[12] Rosen E.; Viswanathan A. and Callon R., "Multiprotocol Label Switching Architecture," *IETF RFC3031*, 2001.

[13] Awduche D. et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels," *IETF RFC3209*, 2001.

[14] Andersson L; Minei I. and Thomas B., "LDP Specification," *IETF RFC5036*, 2007.

[15] Awduche D. et. al., "Requirements for Traffic Engineering Over MPLs," *IETF RFC2702*, 1999.

[16] Jamoussi B.; Andersson L.; Collon R. and Dantu R., "Constraint-Based LSP Setup using LDP," *IETF RFC3212*, 2002.

[17] Braden R. et al., "Resource ReSerVation Protocol (RSVP) Version 1 Functional Specification," *IETF RFC2205*, 1997.

[18] Medhi D. and Ramasay K., "Network Routing-Algoritmos, Protocolos and Architectures," *Ed. Morgan Kaufmann*, pp. 166–191, 2007.

[19] Pana F; and Put F., "A Survey on the Evolution of RSVP," *IEEE Commun. Surv. Tutorials*, pp. 1859–1887, 2013.

[20] Moy J., "OSPF Version 2," *IETF RFC2328*, 1998.

[21] Coltun R.; Ferguson D.; Moy J. and Lindem A., "OSPF for IPv6," *IETF RFC5340*, 2008.

[22] Katz D.; Kompella K. and Yeung D., "Traffic Engineering (TE) Extensions to OSPF versión 2," *IETF RFC3630*, 2003.

[23] Ishiguro K.; Manral V.; Davey A. and Lindem A., "Traffic Engineering Extensions to OSPF Version 3," *IETF RFC5329*, 2008.

[24] Fortz B. and Thorup M., "Internet Traffic Engineering by Optimizing OSPF Weights," *Proc. IEEE INFOCOM*, pp. 519–528, 2000.

[25] Fortz B.; Rexford J. and Thorup M., "Traffic Engineering with Traditional IP Routing Protocols," *IEEE Commun. Mag.*, vol. 40, no. 10, pp. 118–124, 2002.

[26] Feldmann A. et al., "NetScope: Traffic Engineering for IP Networks," *IEEE Netw.*, vol. 14, no. 2, pp. 11–19, 2000.

[27] Feldmann A. and Rexford J., "IP Network Configuration for Intradomain Traffic Engineering," *IEEE Netw.*, vol. 15, no. 5, pp. 46–57, 2001.

[28] Kumar P.; Yu C.; Yuan Y.; Foster N.; Kleinberg R. and Soulé R., "YATES: Rapid Prototyping for Traffic Engineering Systems," in *Proceedings of the Symposium on SDN Research SOSR '18*, 2018.

[29] Ericsson M.; Resende M. and Pardalos P., "A Genetic Algorithm for the Weight Setting Problem in OSPF Routing," *J. Comb. Optim.*, vol. 6, no. 3, pp. 299–333, 2001.

[30] Gojmerac I.; Ziegler T.; Ricciato F. and Reichl P., "Adaptive Multipath Routing for Dynamic Traffic Engineering," *Proc. IEEE GLOBECOM*, pp. 3058–3062, 2003.

[31] Wang J. et al., "Edge Based Traffic Engineering for OSPF Networks," *Comp. Networks*, vol. 48, no. 4, pp. 605–625, 2004.

[32] Abrahamsson H. and Bjorkman M., "Robust Traffic Engineering using L-balanced Weight-Settings in OSPF/IS-IS," *Broadband Commun. Networks, Syst. 2009. BROADNETS 2009. Sixth Int. Conf.* , pp. 1–8, 2009.

[33] Wang Y.; Wang Z. and Zhang L., "Internet Traffic Engineering without Full Mesh Overlaying," *INFOCOM2001*, pp. 565–571, 2001.

[34] Rétvári G. and Cinkler T., "Practical OSPF Traffic Engineering," *IEEE Commun. Lett.*, vol. 8, no. 11, pp. 689–691, 2004.

[35] Rétvári G.; Szabó R.; and Bíró J., "On the Representability of Arbitrary Path Sets as Shortest Paths: Theory, Algorithms and Complexity," *Proc. IFIP Netw.*, pp. 1180–1191, 2004.

[36] Fortz B., "On the evaluation of the reliability of OSPF routingh in IP networks," 2002.

[37] Lad M.; Han Park J.; Refice T and Zhang L., "A Study of Internet Routing Stability Using Link Weight," *Tech. Rep. Comput. Sci. Dep. Univ. Calif.*, 2008.

[38] Bessa Maia J.; Da Silva A.; Silva J. and Cunha P., "A Methodology of Traffic Engineering to IP Backbone," *Comput. Sci.*, 2009.

[39] Xu K.; Liu H.; Liu J. and Shen M., "One More Wight is Enough: Toward the Optimal Traffic engineering with OSPF," *IEEE Comput. Soc. 31st Int. Conf. Distrib. Comput. Syst.*, pp. 836–846, 2011.

[40] Gunnar A., "Aspects of Proactive Traffic Engineering in IP Networks," *PhD Thesis, Stock. Sweden*, 2011.

[41] Gunnar A. and Johansson M., "Cautious Weight Tuning for Link State Routing Protocols," *Int. Conf. Netw. Serv. Manag.*, 2010.

[42] Gunnar A.; Abrahamsson H. and Soderqvist M., "Performance of Traffic Engineering in Operational IP-Networks - an experimental Study," *Proc. 5th IEEE Int. Work. IP Oper. Manag. IPOM 2005, Barcelona Spain*, 2005.

[43] Gunnar A.; and Johansson M., "Robust load balancing under traffic uncertainty-tractable models and efficient algorithms," *Telecommun. Syst. Journal, Press.*, 2010.

[44] Rekhter Y.; Li T. and Hares S., "A Border Gateway Protocol 4 (BGP-4)," *IETF RFC4271*, no. S, 2006.

[45] Wang N.; Ho K.; Pavlou G. and Howarth M., "An overview of routing optimization por internet traffic enginering," *IEEE Commun. 1st Quart.*, vol. 10 No 1, pp. 36–56, 2008.

[46] Gao L. and Rexford J., "Stable Internet Routing without Global Coordination," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 681–692, 2001.

[47] Yang et al., "On Route Selection for Interdomain Traffic Engineering," *IEEE Netw.*, vol. 19, no. 6, pp. 20–27, 2005.

[48] Feamster N. et al., "Guidelines for Interdomain Traffic Engineering," *ACM SIGCOMM*, vol. 33, no. 5, pp. 19–30, 2003.

[49] Quoitin B.; Pelsser C. and Swinnen L., "Interdomain Traffic Engineering with BGP," *IEEE Commun. Mag.*, vol. 41, no. 5, pp. 122–128, 2003.

[50] Uhlig S. and Quoitin B., "Tweak-it: BGP-based Interdomain Traffic Engineering for transit ASs," *IEEE*, pp. 75–82, 2005.

[51] Guo H.; Gao S. and Zhang H., "Inter-Domain Routing With AS Number :A Traffic Engineering Perspective ," *IEEE, Comput. Netw. Multimed. Technol.*, pp. 1–4, 2009.

[52] Awduche Daniel O., "MPLS and Traffic Engineering in IP Networks," *IEEE Commun. Mag.*, vol. 37, no. 12, pp. 42–47, 1999.

[53] Younis O.; and Fahmy S., "Constraint-Based Routing in the Internet: Basic Principles and Recent Research," *IEEE Commun. Surv. Tutorials*, 2003.

[54] Andersson L.; Doolan P.; Feldman N.; Fredette A.; and Thomas B.;, "LDP Specification," *IETF RFC3036*, 2001.

[55] Xiao X.; Hannan A.; Bailey B. and Ni L., "Traffic Engineering with MPLS in the Internet," *IEEE Netw.*, vol. 14, no. 12, pp. 28–33, 2000.

[56] Blak S. et al., "An Architecture for Differentiated Services," *IETF RFC2475*, 1998.

[57] Trimintzios P. et al., "Quality of service provisioning trough traffic engineering with aplicability to IP-based production networks," *Comput. Commun.*, vol. 26, no. 8, pp. 845–860, 2003.

[58] Tabatabaee V. et al., "Differenciated Traffic Engineering for QoS Provisioning," *Proc. IEEE INFOCOM*, pp. 2349–2359, 2005.

[59] Pioro M. and Medhi D., "Routing, Flow, and Capacity Design in Communication and Computer networks," vol. Morgan Kau, p. 170, 2004.

[60] Barakovic J.; Bajric H. and Husic A., "QoS Design Issues and Traffic Engineering in Next Generation IP/MPLS Network.," *9th Int. Conf. Telecomunications*, pp. 203–209, 2007.

[61] Scoglio C. et al., "TEAM: A Traffic Engineering Automated Manager for DiffServ Based MPLS Networks," *IEEE Commun. Mag.*, vol. 42, no. 10, pp. 134–145, 2004.

[62] Oliveira J.; Scoglio C.; Akyildiz F.; and Uhl G., "New Preemption Policies for Diffserv-Aware Traffic Engineering to Minimize Rerouting in MPLS Networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 4, pp. 733–745, 2004.

[63] Saito H.; Miyao Y.; and Yoshida M., "Traffic Engineering using Multiple Multipoint-to-Point LSPs," *Proc. IEEE INFOCOM*, pp. 894–901, 2000.

[64] Urvoy-Keller G.; Hébuterne G.; and Dallery Y., "Traffic Engineering in a Multipoint-to-Point Network," *IEEE JSAC*, vol. 20, no. 4, pp. 834–849, 2002.

[65] Bhatnagar S.; Ganguly S.; and Nath B., "Creating Multipoint-to-Point LSPs for Traffic Engineering," *IEEE Commun. Mag.*, vol. 43, no. 1, pp. 95–100, 2005.

[66] Trimintzios P. et al., "Engineering the Multi-Service Internet: MPLS and IP-based Techniques," *Proc. IEEE Int. Conf. Telecommun. (ICT'2001), Bucharest, Rom.*, vol. 3, pp. 129–134, 2001.

[67] Elwalid A.; Jin C.; Low S.; and Widjaja I., "MATE: MPLS Adaptive Traffic Engineering," *Proc. IEEE INFOCOM*, pp. 1300–1309, 2001.

[68] Aukia P. et al., "RATES: A Server for MPLS Traffic Engineering," *IEEE Netw.*, vol. 14, no. 2, pp. 34–41, 2000.

[69] Boutaba R.; Szeto W.; and Iraqi Y., "DORA: Efficient Routing for MPLS Traffic Engineering," *J. Netw. Sys. Mgmt*, vol.

10, no. 3, pp. 309–325, 2002.
[70] Kar K.; Kodialam M.; and Lakshman V., "Minimum Interference Routing of Bandwidth Guaranteed Tunnels with MPLS Traffic Engineering Applications," *IEEE JSAC*, vol. 18, no. 12, pp. 2566–2579, 2000.
[71] Kodialam M.;an Lakshman V., "Minimum Interference Routing with Applications to MPLS Traffic Engineering," pp. 884–893, 2000.
[72] Oliveira J.; Martinelli F.; and Scoglio C., "SPeCRA: A Stochastic Performance Comparison Routing Algorithm for LSP Setup in MPLS Networks," *Proc. IEEE GLOBECOM*, pp. 2190–2194, 2002.
[73] Walkowiak K., "Survivable Online Routing for MPLS Traffic Engineering," *Proc. QoFIS*, pp. 288–297, 2004.
[74] Li Ch.; Li P. and Mohammed T., "An Optimal MPLS-TE Solution to Route Selection and Redistribution on Congested Networks," *IEEE Comput. Soc. Int. Conf. Networking, Archit. Storage (NAS 2007)*, pp. 69–76, 2007.
[75] Mohamad I.; Wan T.; Alzyoud F. and Sumari P., "Optimizing the MPLS Support for Real Time IPv6-Flows using MPLS-PHS Approach," *TENCON 2009 - 2009 IEEE Reg. 10 Conf.*, pp. 1–6, 2009.
[76] Pham H. and Lavery B., "Hybrid Routing for Scalable IP/MPLS Traffic Engineering," *IEEE Int. Conf. Commun. 2003. ICC '03.*, vol. 1, pp. 332–337, 2003.
[77] Bagula A., "Hybrid IGP+MPLS Routing in Next Generation IP Networks: An Online Traffic Engineering Model," *Springer-Verlag Berlin Heidelb. 2005*, pp. 325–338, 2005.
[78] Skivée F.; Balon S. and Leduc G., "A Scalable Heuristic for Hybrid IGP/MPLS Traffic Engineering- Case Study on an Operational Network," *Networks, 2006. ICON '06. 14th IEEE Int. Conf. Telecommun.*, vol. 2, pp. 1–6, 2006.
[79] Zhang M.; Liu B. and Zhang B., "Multi-Commodity Flow Traffic Engineering with Hybrid MPLS/OSPF Routing," *Glob. Telecommun. Conf. 2009. GLOBECOM 2009. IEEE*, pp. 1–6, 2010.
[80] Farinacci D.; Meyer D.; and Lewis D.;, "The Locator/ID Separation Protocol (LISP)," *IETF RFC6830*, p. 70, 2013.
[81] Farinacci D.; Kowal M. and Lahiri P., "LISP Traffic Engineering Use-Cases.," *draft-ietf-lisp-te-04*, 2019.
[82] Saucez D.; Donnet B.; Iannone L. and Bonaventure O., "Interdomain Traffic Engineering in a Locator/Identifier Separation Context.," *IEEE Internet Netw. Manag. Work.*, 2008.
[83] Li K.; Wang S. and Wang X., "Edge Router Selection and Traffic Engineering in LISP-Capable Networks," *IEEE J. Commun. Networks*, vol. 13, no. 6, p. 612–620., 2011.
[84] Herrmann D.; Turba M.; Kuijper A. and Schweizer I., "Inbound Interdomain Traffic Engineering with LISP.," *39th Annu. IEEE Conf. Local Comput. Networks*, pp. 458–461, 2014.
[85] Jeong T.; Liy J.; Hyun J.; Yoo J. H. and Hong J. W. K., "Experience on the development of LISP-enabled services: An ISP perspective," *Proc. 2015 1st IEEE Conf. Netw. Softwarization*, p. 1–9., 2015.
[86] Nguyen H. D. D. and Secci S., "LISP-EC: Enhancing LISP with Egress Control.," *IEEE Conf. Stand. Commun. Netw.*, 2016.
[87] Filsfils C.; Previdi S.; Ginsberg L.; Decraene B.; Litkowski S. and Shakir R., "Segment Routing Architecture," *IETF RFC8402*, p. 32, 2018.
[88] Filsfils C. et al., "Segment Routing Policy for Traffic Engineering.," *Internet Draf Draft.*, 2017.
[89] Bhatia R.; Hao F.; Kodialam M. and Lakshman T. V., "Optimized Network Traffic Engineering using Segment Routing.," in *IEEE Conf. Comput. Commun.*, 2015.
[90] Rabah G.; Olivier D.; Samer L. and Texier G., "Label Encoding Algorithm for MPLS Segment Routing," *IEEE 15th Int. Symp. Netw. Comput. Appl*, pp. 113–117, 2016.
[91] Salsano S.; Siracusano G.; Luca V.; Luca D. and Pier L., "PSMR-Poor Man's Segment Routing, a minimalistic approach to segment routing and a traffic engineering use case.," *NOMS IEEE/IFIP Netw. Oper. Manag. Symp.*, pp. 598–604, 2016.
[92] F. Moreno, E.; Beghelli, A. and Cugini, "Traffic engineering in segment routing networks.," *Comput. Networks*, vol. 114, pp. 23–31, 2017.
[93] Schüller T.; Aschenbruck N.;Chimani M.; Horneffer M. and Schnitter S., "Traffic Engineering Using Segment Routing and Considering Requirements of a Carrier IP Network," *IEEE/ACM Trans. Netw.*, vol. 26, pp. 1851–1864, 2018.
[94] Karaman A., "Constraint-Based Routing in Traffic Engineering," *Comput. Networks*, pp. 49–54, 2006.
[95] Kuipers F.; Van Mieghem P.; Korkmaz T. and Krunz, "An Overview of Constraint-Based Path Selection Algorithms for QoS Routing," *IEEE Commun. Mag.*, pp. 50–55, 2002.
[96] Becerra L. Y.; and Padilla J. J., "Estudio de propuestas para soportar Ingeniería de tráfico en Internet," *Entre Cienc. e Ing.*, vol. 6, no. 11, pp. 53–76, 2012.
[97] Guerin R.; Orda A. and Williams D., "QoS Routing Mechanisms and OSPF Extensions," *IEEE/Globecom*, 1997.
[98] Wang Z.; and Crowcroft J., "Quality of Service Routing for Supporting Multimedia Applications," *IEEE JSAC*, vol. 14, no. 7, pp. 1228–1234, 1996.
[99] Khan J. A.; Alnuweiri H. M., "A Fuzzy Constraint-Based Routing Algorithm for Traffic Engineering," *IEEE Commun. Soc.*, vol. 3, pp. 1366–1372, 2004.
[100] Suri S.; Waldvogel M.; Bauer D.; and Ramesh P., "Profile based routing and traffic engineering," *Comput. Commun.*,

pp. 351–365, 2003.
[101] Wang J.; Patek S.; Wang H.; and Liebeherr J., "Traffic Engineering with AIMD in MPLS Networks," *Lect. Notes Comput. Sci.*, pp. 192–210, 2002.
[102] Cho H. Y.; Lee J. Y.; Kim B. C., "Multi-path Constraint-based Routing Algorithms for MPLS Traffic Engineering," *IEEE Int. conf Commun.*, vol. 3, p. 5, 2003.
[103] Tang J.; Siew C. K. and Feng G., "Parallel LSPs for constraint-based routing and load balancing in MPLS networks," *IEE Proc.-Commun.,* vol. 152, pp. 6–12, 2005.
[104] Deb K.; Agrawal A.; Pratab A. and Meyarivan T., "A fast elitist nondominated sorting genetic algorithm for multiobjective optimization: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.
[105] Zeng Y. and Luo Y., "A multi-contrained Routing Optimization Algorithm in the IP Networks," *1th Int. Conf. Nat. Comput. IEEE*, pp. 314–318, 2015.
[106] Mieghem P. V. and Kuipers F. A., "Concepts of exact QoS routing algoritms," *IEEE/ACM Trans. Netw.*, vol. 12, no. 5, pp. 851–864, 2004.
[107] Korkmaz T. and Krunz M., "Routing Multimedia Traffic with QoS guarantees," *IEEE Trans. Multimed.*, vol. 5, no. 3, pp. 429–443, 2003.
[108] Becerra L.Y.; Bañol J.L. and and Padilla J.J., "Un estudio sobre algoritmos basados en restricciones: objetivos ingeniería de tráfi co y calidad de servicio," *Entre Cienc. e Ing.*, vol. 11, no. 21, pp. 103–111, 2017.
[109] Nayak P. And Murty G.R., "Survey on constrained based path selection QoS routing algortihms: MCP and MCOP Problems," *J. Inf. Syst. Commun.*, vol. 4, no. 1, p. 6, 2013.
[110] Jaffe J., "Algorithms for finding paths with multiple constraints," *Networks,* vol. Volume 14, no. 1, pp. 95–116, 1984.
[111] Iwata A.; Izmailov R.; Lee D.-S.; Sengupta B.; Ramamurthy G. Suzuki H., "ATM routing algorithms with multiple QoS requirements for multimedia internetworking," *IEICE Trans. Commun. E79-B*, pp. 999–1006, 1996.
[112] De Neve H. and Van Mieghem P., "TAMCRA: A Tunable Accuracy Multiple Constraints Routing Algorithm," *Comp. Commun.*, vol. 23, pp. 667–79, 2000.
[113] Van Mieghem P.; De Neve H. and Kuipers F. A., "Hop- by-hop Quality of Service Routing," *Comp. Nets.,* vol. 37, pp. 407–23, 2001.
[114] Chen S. and Nahrstedt K., "On finding multi-constrained Paths," *IEEE*, pp. 874–879, 1998.
[115] Korkmaz T. and Krunz M., "A randomized algorithm for finding a path subject to multiple QoS constraints," *Comput. Networks*, pp. 251–268, 2001.
[116] Korkmaz T. and Krunz M., "Multi Contrained Optimal Path Selection," *IEEE INFOCOM*, pp. 834–843, 2001.
[117] Liu G. and Ramakrishnan K., "A*Prune: an algorithm for finding k shortest path subject to multiple constraint," *IEEE INFOCOM*, pp. 743–749, 2001.
[118] Feng G., "The revisit of QoS routing based on non-linear lagrange relaxation," *Int. J. Commun. Syst.*, pp. 9–22, 2005.
[119] Li Y.; Harms J. and Holte R., "Fast Exact Multiconstraint Shortest Path Algorithms," *Proc. IEEE ICC*, pp. 123–130, 2007.
[120] Chen S.; Song M. and Sahni S., "Two Techniques for fast computation of constrained shortest paths," *IEEE/ACM Trans. Netw.*, pp. 105–115, 2008.
[121] Liu W.; Lou W. and Fang Y., "An efficient quality of service routing algorithm for delay-sensitive applications," *Comput. Networks*, vol. 47, no. 1, pp. 87–104, 2005.
[122] Baradaran M. and Yaghmaee H., "A Constraint Based Routing algorithm For Multimedia Networking," *IAENG Int. J. Comput. Sci.*, vol. 33, no. 2, p. 8, 2007.
[123] Prakash P. and Selvan S., "A Feasible Path Selection QoS Routing Algorithm with two constraints in packet switched networks," *World Acad. Sci. Eng. Technol.*, pp. 444–450, 2008.
[124] Handler G. Y. and Zang I., "A dual algorithm for the constrained shortest path problem," *Networks*, pp. 293–310, 1980.
[125] Guo L. and Matta I., "Search space reduction in QoS routing," *Comput. Networks*, vol. 41, no. 1, pp. 73–88, 2003.
[126] Yuan X. and Liu X., "Heuristic algorithm for multiconstrined quality of service routing," *IEEE INFOCOM*, pp. 844–853, 2001.
[127] Kuipers F. A. and Mieghem P. V., "Bi-directional Search in QoS Routing," *Lect. Notes Comput. Sci.*, pp. 102–111, 2003.
[128] Anjali T. and Scoglio C., "Traffic routing in MPLS Networks Based on QoS Estimation and Forecast," *IEEE Glob. Telecommun. Conf.*, vol. 2, pp. 1135–1139, 2004.
[129] Lin H.; Xue-wu D. and Jin X., "Multi-Constrained Routing Based on Tabu Search," *IEEE Int. Conf. Control Autom.*, pp. 157–161, 2007.
[130] Vincent G. and Sasipraba T., "An Efficient Routing Algorithm for Improving the QoS in Internet," *Int. Conf. Emerg. Trends Robot. Commun. Technol.*, pp. 381–387, 2010.
[131] Dragos S. M., "Hierarchical QoS Routing by Using Multi-Constrained Macro-Routing," *7th Int. Conf. Next Gener.*

Web Serv. Pract., pp. 105–112, 2011.
[132] Cui H.; Li J.; Liu X. and Cai Y., "Particle Swarm Optimization for Multi-constrained Routing in Telecommunication Networks," *I.J.Computer Netw. Inf. Secur.*, pp. 10–17, 2011.
[133] Zhang B.; Hao J. and Mouftah H. T., "Bidirectional Multi-Constrained Routing Algorithms," *IEEE Trans. Comput.*, vol. 63, no. 9, pp. 2174–2186, 2014.
[134] Phuong Thanh C. T.; Nam H. H. and Hung T. C., "A heuristic algorithm for bandwidth delay constrained routing," *Int. Conf. Adv. Technol. Commun.*, pp. 99–104, 2014.
[135] Avallone S. and Ventre G., "Q-BATE: A QoS Constraint-based Traffic Engineering Routing Algorithm," *Conf. Next Gener. Internet Des. Eng. 2006.*, p. 8, 2006.
[136] Kulkarni S.; Sharma R. and Mishra I., "New Bandwidth Guaranteed QoS Routing Algorithm for MPLS Networks," *J. Emerg. Trends Comput. Inf. Sci.*, vol. 3, no. 3, pp. 384–389, 2012.
[137] Kher V.; Arman A. and Saini D. S., "Hybrid evolutionary MPLS Tunneling Algorithm based on high priority bits," *Int. Conf. Futur. trend Comput. Anal. Knowl. Manag.*, pp. 495–499, 2015.
[138] Perkins C.; Jhonson D. and Arkko J., "Mobility Support in IPv6," *IETF RFC6275*, 2011.
[139] Soliman H.; Catelluccia C; ElMalki K. and Bellier L., "Hierarchical Mobile IPv6 (HMIPv6) Mobility Management," *IETF RFC5380*, 2008.
[140] Koodli R., "Mobile IPv6 Fast Handovers," *IETF RFC5568*, 2009.
[141] Conta A. and Deering S., "Generic Packet Tunneling in IPv6 Specification," *IETF RFC2473*, 1998.
[142] McGovern M. and Ullmann R., "CATNIP: Common Architecture for the Internet," *IETF RFC1707*, 1994.
[143] Hinden R., "Simple Internet Protocol Plus White Paper," *IETF RFC1710*, 1994.
[144] Metzler J. and Hauth S., "An end-to-end usage of the IPv6 flow label," *Work Prog.*, 2000.
[145] Conta A. and Carpenter B., "A proposal for the IPv6 Flow Label Specification," *IETF Internet-Draft*, 2001.
[146] Conta A. and Rajahalme J., "A Model for Diffserv use of the IPv6 Flow Label Specification," *IETF Internet-Draft*, 2001.
[147] Hagino J., "Socket API for IPv6 Flow Label Field," *IETF Internet-Draft*, 2001.
[148] Rajahalme J.; Conta A.; Carpenter B. and Deering S., "IPv6 Flow Label Specification," *IETF RFC3697*, 2004.
[149] Amante S.; Carpenter B.; Jiang S. and Rajahalme J., "Ipv6 Flow Label Specification," *IETF RFC6437*, 2011.
[150] Schulzrinne H. and Hancock R., "GIST: General Internet Signalling Transport," *IETF RFC5971*, 2010.
[151] Lin C.; Tseng P. and Hwang W., "End-to-End QoS Provisioning by Flow Label in IPv6," *ICIS*, 2006.
[152] Prakash B., "Using the 20 bit flow label field in the IPv6 header to indicate desirable quality of service on the internet," *Univ. Color. (MSc Thesis)*, 2004.
[153] Lee I. and Kim S., "A QoS Improvement Scheme for Real-Time Traffic Using IPv6 Flow Labels," *Lect. Notes Comput. Sci. Vol 3043*, 2004.
[154] Banerjee R.; Malhotra S. and M. M, "A Modified Specification for use of the IPv6 Flow Label for providing An efficient Quality of Service using a hybrid approach," *Work Prog.*, 2002.
[155] Black D; Brim S. and Le Faucheur F., "Per Hop Behavior Identification Codes," *IETF RFC3140*, 2001.
[156] Zheng He et al., "The Optimization of QoS Path-Selection Using Flow Label Based On Overlay Network," *IEEE Int. Conf. Commun. Technol. Appl.*, pp. 241–245, 2009.
[157] Xiao-Hong Huang et. al., "Packet-switch Mechanism Based on QoS Class Mapping Using Flow Label for Ovelay Network," *J. China Univ. Post Telecommun. Sci. Direct*, vol. 17, pp. 17–23, 2010.
[158] Padilla J. and Paradells J., "Intserv6: an approach to support QoS over IPv6 wired and wireless networks," *Eur. Trans. Telecommun. Wiley Intersci.*, 2007.
[159] Braden R.; Clark D. and Shenke S., "Integrated Services in the Internet Architecture: an Overview," *IETF RFC1633*, 1994.
[160] Chakravorty S., "Challenges of IPv6 Flow Label implementation," *Proc IEEE MILCOM2008*, 2008.
[161] Chakravorty S.; Bush J. and Bound J., "IPv6 Label Switching Architecture," *Work Prog.*, 2008.
[162] Beckman M., "IPv6 Dynamic Flow Label Switching (FLS)," *IETF Internet-Draft*, 2007.
[163] Roberts L. and Harford J., "In-Band QoS Signaling for IPv6," *Work Prog.*, 2005.
[164] Chin-Ling Chen, "A Study of IPv6 Labeling Forwarding Model Supporting Diffserv," *Procedia Eng. Sci. Elseiver*, vol. 15, no. 5590–5594, 2011.
[165] Doan H. et. al., "Flow-based forwarding scheme and performance analysis in mobile IPv6 networks.," *8th Int. Conf. Adv. Commun. Technol.*, vol. 3, pp. 1490–1496, 2006.
[166] Zheng Tao; Wang Lan and Gu Daqing, "A flow Label Based QoS Scheme for End-to-End Mobile Services," *ICNS 2012-Ehte Eighth Int. Conf. Netw. Serv.*, pp. 169–174, 2012.
[167] Ouellette S. and Pierre S., "HPMRSVP-TE: a Hierarchical Proxy Mobile Resource Reservation Protocol for Traffic engineering," *IEEE 64th Veh. Technol. Conf. 2006*, pp. 1–5, 2006.
[168] Yee Tai W.; Eng Tan Ch. and Ping Lau S., "Towards Utilizing Flow Label IPv6 in Implicit Source Routing for Dynamic

Source Routing (DSR) in Wireless Ad Hoc Network.," *IEEE Symp. Comput. Informatics*, pp. 101–106, 2012.
[169] Donley C. and Erichsen K., "Using the Flow Label with Dual-Stack Lite," *Work Prog.*, 2011.
[170] Carpenter B. and Amante S., "Using the IPv6 flow label for equal cost multipath routing and link aggregation in tunnels," *RFC6438*, 2011.
[171] Hopps C., "Analysis of an Equal-Cost Multi-Path Algorithm," *IETF RFC2992*, 2000.
[172] Blake S., "Use of the IPv6 Flow Label as a Transport- Layer Nonce to Defend Against Off-Path Spoofing Attacks," *Work Prog.*, 2009.
[173] Balbinot L.; Andrade M.; De Tarouco L. and Roesler V., "IP Next Generation Label Switching," *IEEE Work. IP Oper. Manag.*, pp. 21–25, 2002.
[174] Davie B. S. and Farrel A., *MPLS: Next Steps*. 2008.
[175] Cerrutti I. and Castoldi P., "Influence of Label Stack Depth on the performance of MPLS Networks," *IEEE Globecom*, pp. 1–5, 2006.
[176] Vanaubel Y.; Mérindol P.; Pansiot J.J. and Donnet B., "A Brief History of MPLS Usage in IPv6," in *Int. Conf. Passiv. Act. Netw. Meas. Link.*, 2016, pp. 359–370.
[177] Hardekopf B.; Taylor R.; Jasleen K.; Mudigonda J.; Dahlin M. and Harrick V., "Impact of Network Protocols on Programmable Router Architectures," *Citeseerx*, 2002.
[178] Padilla J. J., "Contribución al soporte de Calidad del Servicio en Redes Móviles," 2007.
[179] Bonald T. and Roberts J. W., "Congestion at flow level and the impact of user behaviour," *Comput. Networks*, vol. 42, pp. 521–536, 2003.
[180] Intel, "Intel® IXP2800 Network Processor," 2002.
[181] Skorepa M. and Klugl R., "Enhanced analytical method for IP mobility handover schemes cost evaluation," *Telecommun. Syst.*, vol. 52, pp. 1573–1582, 2013.