

Vea discusiones, estadísticas y perfiles de autor para esta publicación en <https://www.researchgate.net/publication/338670233>

Algoritmo de control de congestión basado en la optimización de colonias de hormigas para la red MPLS

Documento de sesión · Enero 2020

CITAS
3

LEE
42

3 autores:



Dr. Rajagopalan S

13 PUBLICACIONES 13 CITAS

VER EL PERFIL



Naganathan Ealai Rengasari

Instituto de Simbiosis de Estudios e Investigaciones Informáticas

48 PUBLICACIONES 163 CITAS

VER EL PERFIL



Perianayagam Herbert Raj

Ministerio de Educación, Brunéi

6 PUBLICACIONES 10 CITAS

VER EL PERFIL

Algunos de los autores de esta publicación también están trabajando en estos proyectos relacionados:



Project

Procesamiento de imágenes [Ver Proyecto](#)



Project

Segmentación de imágenes hiperespectrales [Ver Proyecto](#)

Congestión basada en la optimización de colonias de hormigas

Algoritmo de control para red MPLS

S.Rajagopalan*, ERNaganathan**, P.Herbert Raj***

* Departamento de CSE, Universidad de Alagappa,
Karaikudi, India.

** *Profesor y Director*, Departamento de Aplicaciones
Informáticas, Velammal Engineering College, Chennai, India

*** Departamento de Educación Técnica,
Brunéi.

Abstracto.

La conmutación de etiquetas multiprotocolo (MPLS) es un mecanismo en redes de telecomunicaciones de alto rendimiento que dirige y transporta datos de un nodo de red al siguiente con la ayuda de etiquetas. MPLS facilita la creación de "enlaces virtuales" entre nodos distantes. Puede encapsular paquetes de varios protocolos de red. MPLS es un mecanismo de transmisión de datos altamente escalable e independiente del protocolo. Las decisiones de reenvío de paquetes se toman únicamente en función del contenido de esta etiqueta, sin necesidad de examinar el paquete en sí. Esto permite crear circuitos de extremo a extremo a través de cualquier tipo de medio de transporte, utilizando cualquier protocolo. Este artículo propone una técnica de optimización de colonias de hormigas para la gestión del tráfico en la red MPLS.

Palabras clave: Optimización de colonias de hormigas, red MPLS, gestión de tráfico

1 Introducción:

MPLS opera en una capa del modelo OSI que generalmente se considera que se encuentra entre las definiciones tradicionales de capa 2 y capa 3 y, por lo tanto, a menudo se la denomina protocolo de "capa 2.5". Fue diseñado para proporcionar un servicio de transporte de datos unificado tanto para clientes basados en circuitos como para clientes de conmutación de paquetes que proporcionan un modelo de servicio de datagramas. Se puede utilizar para transportar muchos tipos diferentes de tráfico, incluidos paquetes IP, así como tramas ATM, SONET y Ethernet nativas. MPLS actualmente está reemplazando algunas de estas tecnologías en el mercado. Es muy posible que MPLS reemplace por completo estas tecnologías en el futuro, alineando así estas tecnologías con las necesidades tecnológicas actuales y futuras.

En particular, MPLS prescinde del bagaje de protocolo de señalización y conmutación de celdas de ATM. MPLS reconoce que las celdas ATM pequeñas no son necesarias en el núcleo de las redes modernas, ya que las redes ópticas modernas (a partir de 2008) son tan rápidas (a 40 Gbit/s y más) que incluso los paquetes completos de 1500 bytes no incurren en un valor real significativo. -tiempo de demoras en las colas (la necesidad de reducir tales demoras, por ejemplo, para soportar el tráfico de voz fue la motivación para la naturaleza celular de ATM). MPLS fue propuesto originalmente por un grupo de ingenieros de IPSILON Networks, pero su tecnología de "conmutación de IP", que se definió solo para trabajar sobre ATM, no logró dominar el mercado. Cisco Systems, Inc., presentó una propuesta relacionada, no restringida a la transmisión ATM, denominada "Conmutación de etiquetas". Era una propuesta propietaria de Cisco y pasó a llamarse "

Una motivación original fue permitir la creación de conmutadores simples de alta velocidad, ya que durante un período de tiempo significativo fue imposible reenviar paquetes IP completamente en hardware. Sin embargo, los avances en VLSI han hecho posible este tipo de dispositivos. Por lo tanto, las ventajas de MPLS giran principalmente en torno a la capacidad de admitir múltiples modelos de servicio y realizar la gestión del tráfico. MPLS también ofrece un sólido marco de recuperación que va más allá de los simples anillos de protección de las redes ópticas síncronas (SONET/SDH). En 2000, un equipo de proyecto dirigido por Preston Poole de Schlumberger NIS implementó la primera iteración de IP-MPLS puro. A través de una serie de empresas conjuntas de investigación, este equipo diseñó, implementó y puso en servicio con éxito la primera red IP-MPLS comercial del mundo. Esta red, que originalmente constaba de 35 puntos de presencia (PoP) en todo el mundo, se diseñó inicialmente para servir a la comunidad de petróleo y gas mediante la entrega del conjunto de servicios DeXa. Las iteraciones posteriores de esta red IP-MPLS comercial incluyeron acceso a VSAT Satellite a través de conexiones estratégicas de telepuerto, acceso a aplicaciones financieras y bancarias, y centros de colaboración de perforación. Otros desarrollos en el campo IP-MPLS implementados por el equipo del Sr. Poole incluyeron la concepción matemática y el desarrollo de los algoritmos más utilizados para lo que hoy se conoce como ancho de banda bajo demanda (BoD), video bajo demanda (VoD) y servicios diferenciados para IP. MPLS. Las iteraciones posteriores de esta red IP-MPLS comercial incluyeron acceso a VSAT Satellite a través de conexiones estratégicas de telepuerto, acceso a aplicaciones financieras y bancarias, y centros de colaboración de perforación. Otros desarrollos en el campo IP-MPLS implementados por el equipo del Sr. Poole incluyeron la concepción matemática y el desarrollo de los algoritmos más utilizados para lo que hoy se conoce como ancho de banda bajo demanda (BoD), video bajo demanda (VoD) y servicios diferenciados para IP. MPLS. Las iteraciones posteriores de esta red IP-MPLS comercial incluyeron acceso a VSAT Satellite a través de conexiones de telepuerto estratégicas, acceso a aplicaciones financieras y bancarias, y centros de colaboración de perforación. Otros desarrollos en el campo IP-MPLS implementados por el equipo del Sr. Poole incluyeron la concepción matemática y el desarrollo de los algoritmos más utilizados para lo que hoy se conoce como ancho de banda bajo demanda (BoD), video bajo demanda (VoD) y servicios diferenciados para IP. MPLS.

2 Modelo de trabajo MPLS:

OSI Model - 7 Layers

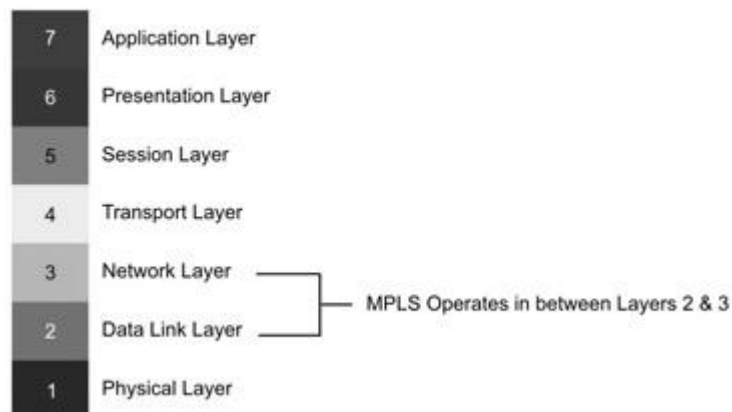


Figura 1. Operación de MPLS en Capa OSI

Esto asegura circuitos de extremo a extremo sobre CUALQUIER tipo de medio de transporte utilizando CUALQUIER protocolo de capa de red. En vista del hecho de que MPLS es compatible con las versiones revisadas del Protocolo de Internet (IPv4 e IPv6), IPX, AppleTalk en Layer3; Ethernet, Token Ring, Interfaz de datos distribuidos por fibra (FDDI), Modo de transferencia asíncrona (ATM), Frame Relay y PPP (protocolo punto a punto) en la capa 2, se denomina "protocolo de capa 2.5".

MPLS funciona prefijando paquetes con un encabezado MPLS, que contiene una o más "etiquetas". Esto se llama pila de etiquetas. Cada entrada de la pila de etiquetas contiene cuatro campos:

- Un valor de etiqueta de 20 bits.
- un campo de clase de tráfico de 3 bits para prioridad QoS (calidad de servicio) (experimental) y ECN (notificación de congestión explícita).
- un indicador de fondo de pila de 1 bit. Si está configurado, significa que la etiqueta actual es la última en la pila.
- un campo TTL (tiempo de vida) de 8 bits.

Estos paquetes etiquetados con MPLS se conmutan después de una búsqueda/cambio de etiqueta en lugar de una búsqueda en la tabla de IP. Como se mencionó anteriormente, cuando se concibió MPLS, la búsqueda de etiquetas y el cambio de etiquetas eran más rápidos que una tabla de enrutamiento o una búsqueda RIB (Base de información de enrutamiento) porque podían tener lugar directamente dentro de la estructura conmutada y no en la CPU. Los puntos de entrada y salida de una red MPLS se denominan enrutadores de borde de etiqueta (LER), que, respectivamente, insertan una etiqueta MPLS en un paquete entrante y la extraen del paquete saliente. Los enrutadores que realizan el enrutamiento basándose únicamente en la etiqueta se denominan enrutadores de conmutación de etiquetas (LSR). En algunas aplicaciones, el paquete

presentado al LER ya puede tener una etiqueta, de modo que el nuevo LER inserte una segunda etiqueta en el paquete. Para obtener más información, consulte el estallido del penúltimo salto.

Las etiquetas se distribuyen entre LER y LSR utilizando el "Protocolo de distribución de etiquetas" (LDP). Los enrutadores de conmutación de etiquetas en una red MPLS intercambian regularmente información de accesibilidad y etiquetas entre sí utilizando procedimientos estandarizados para crear una imagen completa de la red que luego pueden usar para reenviar paquetes. El operador de la red establece rutas de conmutación de etiquetas (LSP) para una variedad de propósitos, como crear redes privadas virtuales IP basadas en la red o enrutar el tráfico a lo largo de rutas específicas a través de la red. En muchos aspectos, los LSP no son diferentes de los PVC en las redes ATM o Frame Relay, excepto que no dependen de una tecnología de capa 2 en particular. Cuando un paquete sin etiquetar ingresa al enrutador de ingreso y necesita pasar a un túnel MPLS, el enrutador primero determina la clase de equivalencia de reenvío (FEC) en la que debe estar el paquete y luego inserta una o más etiquetas en el encabezado MPLS recién creado del paquete. Luego, el paquete se pasa al enrutador del siguiente salto para este túnel. Cuando un enrutador MPLS recibe un paquete etiquetado, se examina la etiqueta superior. En función del contenido de la etiqueta, se puede realizar una operación de intercambio, inserción (imposición) o extracción (eliminación) en la pila de etiquetas del paquete. Los enrutadores pueden tener tablas de búsqueda prediseñadas que les indiquen qué tipo de operación realizar en función de la etiqueta superior del paquete entrante para que puedan procesar el paquete muy rápidamente. Cuando un enrutador MPLS recibe un paquete etiquetado, se examina la etiqueta superior. En función del contenido de la etiqueta, se puede realizar una operación de intercambio, inserción (imposición) o extracción (eliminación) en la pila de etiquetas del paquete. Los enrutadores pueden tener tablas de búsqueda prediseñadas que les indiquen qué tipo de operación realizar en función de la etiqueta superior del paquete entrante para que puedan procesar el paquete muy rápidamente. Cuando un enrutador MPLS recibe un paquete etiquetado, se examina la etiqueta superior. En función del contenido de la etiqueta, se puede realizar una operación de intercambio, inserción (imposición) o extracción (eliminación) en la pila de etiquetas del paquete. Los enrutadores pueden tener tablas de búsqueda prediseñadas que les indiquen qué tipo de operación realizar en función de la etiqueta superior del paquete entrante para que puedan procesar el paquete muy rápidamente.

En una operación de intercambio, la etiqueta se intercambia con una nueva etiqueta y el paquete se reenvía a lo largo de la ruta asociada con la nueva etiqueta. En una operación de inserción, se inserta una nueva etiqueta encima de la etiqueta existente, "encapsulando" efectivamente el paquete en otra capa de MPLS. Esto permite el enrutamiento jerárquico de paquetes MPLS. En particular, esto es utilizado por MPLS VPN. En una operación pop, la etiqueta se elimina del paquete, lo que puede revelar una etiqueta interna debajo. Este proceso se llama "desencapsulación". Si la etiqueta extraída fue la última en la pila de etiquetas, el paquete "sale" del túnel MPLS. Durante estas operaciones, no se examinan los contenidos del paquete debajo de la pila de etiquetas MPLS. De hecho, los enrutadores de tránsito generalmente solo necesitan examinar la etiqueta superior en la pila. El reenvío del paquete se realiza en función del contenido de las etiquetas, lo que permite el "reenvío de paquetes independiente del protocolo" que no necesita mirar una tabla de enrutamiento dependiente del protocolo y evita la costosa coincidencia de prefijo IP más largo en cada salto. En el enrutador de salida, cuando se ha extraído la última etiqueta, solo queda la carga útil. Puede ser un paquete IP o cualquier otro tipo de paquete de carga útil. Por lo tanto, el enrutador de salida debe tener información de enrutamiento para la carga útil del paquete, ya que debe reenviarlo sin la ayuda de las tablas de búsqueda de etiquetas. Un enrutador de tránsito MPLS no tiene tal requisito. En algunos casos especiales, la última etiqueta también se puede quitar en el penúltimo salto (el salto antes del enrutador de salida). Esto se llama Penúltimo Hop Popping (PHP). Esto puede ser interesante en los casos en que el enrutador de salida tiene muchos paquetes que salen de los túneles MPLS y, por lo tanto, gasta cantidades excesivas de tiempo de CPU en esto. Mediante el uso de PHP, los enrutadores de tránsito conectados directamente a este enrutador de salida lo descargan de manera efectiva, sacando la última etiqueta ellos mismos.

MPLS puede utilizar la red ATM existente o la infraestructura de retransmisión de tramas, ya que sus flujos etiquetados se pueden asignar a identificadores de circuitos virtuales de retransmisión de tramas o ATM, y

viceversa. MPLS no se puede comparar con IP como una entidad separada porque funciona junto con IP y los protocolos de enrutamiento IGP de IP. Los dispositivos LSR proporcionan funciones de ingeniería de tráfico que se pueden definir mediante

- configuración explícita de salto a salto,
- enrutado dinámicamente por el algoritmo de ruta más corta restringida primero (CSPF), o
- configurado como una ruta suelta que evita una IP particular o que es en parte explícita y en parte dinámica.

En una red IP pura, se elige la ruta más corta a un destino incluso cuando se vuelve más congestionado. Mientras tanto, en una red IP con enrutamiento CSPF de ingeniería de tráfico MPLS, también se pueden considerar restricciones como el ancho de banda RSVP de los enlaces atravesados, de modo que se elegirá la ruta más corta con ancho de banda disponible. MPLS Traffic Engineering se basa en el uso de extensiones TE para OSPF o IS-IS y RSVP. Además de la restricción del ancho de banda RSVP, los usuarios también pueden definir sus propias restricciones especificando atributos de enlace y requisitos especiales para túneles. enrutar (o no enrutar) sobre enlaces con ciertos atributos.

3 Arquitectura MPLS

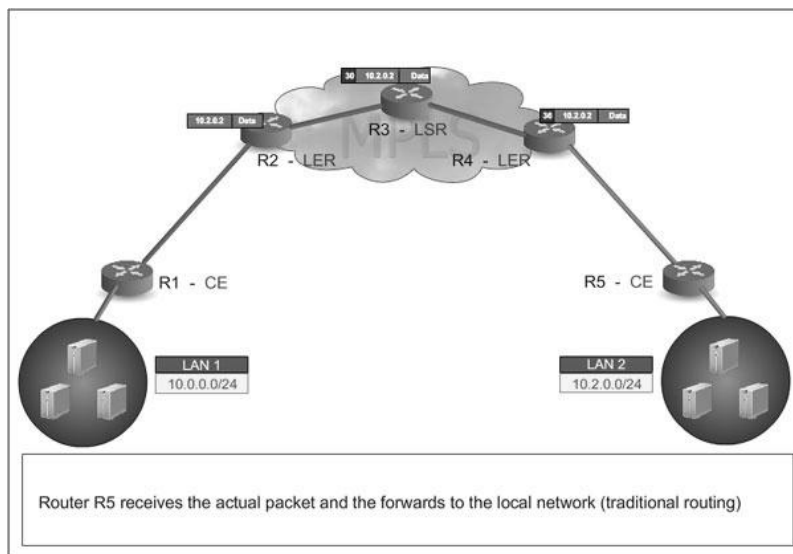


Figura 2. Arquitectura de muestra de la red MPLS

El cruce de paquetes en la figura 1 es Sitio A (10.2.0.0/24) – R1 – R2 (MPLS Ingress LER)- R3 (LSR) - R4 (LSR) - R5 (Egress LER) – R6 – Sitio B (10.0.0.0. 0.0/24). Aquí R1-

R2- R3- R4- R5- R6 se refiere a los enrutadores utilizados para la operación de conmutación de paquetes. Los elementos del interruptor en capas son:

- **Clase de equivalencia directa (FEC):** Esta clase incluye un grupo de paquetes de una aplicación específica reenviados en su ruta de conmutación por la misma ruta (con el mismo tratamiento de reenvío). Cada paquete de una clase en particular tiene el mismo requisito de servicio. A cada tipo de tráfico de datos se le asigna un nuevo FEC y se realiza una sola vez mientras ingresan a la nube MPLS.
- **Enrutador de borde de etiqueta de entrada (LER):** Existe en el perímetro de una nube mpls y es un punto de entrada donde el paquete de datos se origina desde su fuente. Este enrutador de borde impone etiquetas (PUSH) y reenvía paquetes al destino a través del dominio. Después de configurar LSP y asignar etiquetas, este enrutador de borde de entrada inicia el proceso de reenvío de paquetes en la red central MPLS.
- **Enrutador de borde de etiqueta de salida (LER):** Existe en el perímetro de una red mpls y es un punto de salida donde el paquete de datos llega a su destino. Este enrutador perimetral realiza la disposición o eliminación de etiquetas (POP) y reenvía el paquete IP al destino. Elimina la etiqueta del paquete que llega solo cuando el indicador de la parte inferior de la pila identifica si la etiqueta encontrada es la etiqueta inferior de la pila o no.
- **Enrutador de interruptor de etiquetas (LSR):** Este enrutador recibe un paquete etiquetado, lo intercambia con uno saliente y reenvía el nuevo paquete a una interfaz apropiada. Según su ubicación en el dominio MPLS, este enrutador realiza la disposición de etiquetas (eliminación, POP), la imposición de etiquetas (adición, PUSH) o el intercambio de etiquetas (reemplazo de la etiqueta superior en una pila con un nuevo valor de etiqueta saliente). Cuando el flujo de datos (archivos o tráfico multimedia) llega desde la red de acceso al núcleo MPLS, se segrega en FEC separado en este enrutador. Como reconocimiento de los enlaces de etiquetas, LSR crea entradas en la base de información de etiquetas. Esta tabla se compone de puertos de E/S y etiquetas de puertos de E/S que indican la asignación de etiqueta-FEC.
- **Ruta de cambio de etiqueta (LSP):** LSP es la ruta atravesada por un paquete desde el origen hasta el destino a través de una red habilitada para MPLS. El trayecto es de tipo símplex o de sentido único. Esto permite que los paquetes se cambien de un borde a otro atravesando varios enrutadores de conmutación intermedios. Cada ubicación de red necesita que se establezcan LSP para la transferencia de datos. Por ejemplo, los datos del paquete de LER1 atraviesan varios nodos intermedios a LER2 utilizando LSP1, luego se establece otra ruta LSP2 para la transferencia de paquetes directamente al otro extremo, que es la ruta más corta para llegar al destino. Sin embargo, la conmutación de ruta se deriva de la información de enrutamiento IGP y puede diferir de la ruta preferida del Protocolo de puerta de enlace interior a la red de destino.

3.1 Encabezado MPLS



Fig. 3. Formato de encabezado MPLS

Un encabezado MPLS de 32 bits contiene el siguiente campo:

- **Etiqueta (20 bits):** Es un campo corto de longitud fija con un identificador físicamente contiguo, que se utiliza para clasificar una FEC de importancia local. Después de esta clasificación, el paquete recibe un valor de etiqueta, que se utiliza para el reenvío de paquetes. Estas etiquetas se crean en función de la topología de la red, la solicitud y el tráfico entrante. Dado que están asociados con FEC, se unen. Tanto el proceso de vinculación como el de distribución de etiquetas se realizan mediante enrutadores descendentes. Las etiquetas se eliminan a la salida y el paquete se envía al destino.
- **Campo Clase de servicio (CoS) o EXP (Experimental):** Este campo de 3 bits influye en los algoritmos de cola y de descarte que se aplican al paquete cuando se transmite a través de la red. Dado que el campo CoS tiene 3 bits, hay ocho clases de servicio distintas en uso.
- **Campo de pila (S):** Una pila de etiquetas es un conjunto ordenado de etiquetas donde cada una opera con una función específica. Si un enrutador de entrada impone más de una etiqueta en un solo paquete IP, forma una pila de etiquetas. Aquí, el indicador de fondo de la pila (S) se crea para mencionar el final de la pila de etiquetas. Como se mencionó anteriormente, LSR intercambia solo la etiqueta superior en una pila de etiquetas. Un LSR de salida continúa con la disposición de etiquetas desde la pila hasta que encuentra el indicador de fondo de pila (S). Después de esto, se realiza una búsqueda de ruta utilizando los detalles del encabezado de la capa 3 de IP y el enrutador del conmutador reenvía el paquete hacia el destino.
- **Tiempo de vida (TTL):** Un campo TTL de 8 bits realiza la misma función que el TTL de IP, donde el paquete se descarta cuando TTL es '0' para evitar bucles en la red. Cada vez que un paquete etiquetado atraviesa LSR, el valor se reduce en uno.
- **Río arriba y río abajo:** El concepto de flujo descendente y ascendente es fundamental para comprender la distribución de etiquetas (plano de control) y el reenvío de paquetes en el dominio MPLS. Tanto aguas abajo como aguas arriba son una referencia a la red. Los datos son aguas abajo para una red de destino, ya que es el punto de destino. Las actualizaciones relacionadas con un prefijo específico siempre se propagan en sentido ascendente (protocolo de enrutamiento o distribución de etiquetas, LDP/TDP).
- **Protocolo de distribución de etiquetas (LDP):** Este protocolo distribuye etiquetas en una red MPLS en función de la información de enrutamiento de IGP. La base de información de reenvío (FIB) del enrutador determina una ruta de salto por salto en toda la red.

A diferencia de las rutas diseñadas para el tráfico que usan restricciones y rutas explícitas para configurar rutas de conmutación de etiquetas de extremo a extremo, LDP se usa solo para señalar los LSP de mejor esfuerzo. El descubrimiento LDP permite que un Label Switch Router determine el potencial de sus pares. Los mecanismos básicos, como el descubrimiento y el descubrimiento extendido, se utilizan para ubicar los LSR a nivel de enlace.

4 Ingeniería de tráfico y control de congestión en MPLS:

La ingeniería de tráfico es un método para optimizar el rendimiento de una red de telecomunicaciones mediante el análisis, la predicción y la regulación dinámica del comportamiento de los datos transmitidos por esa red. La ingeniería de tráfico también se conoce como ingeniería de teletráfico y gestión de tráfico. Las técnicas de ingeniería de tráfico se pueden aplicar a redes de todo tipo, incluida la PSTN (red telefónica pública conmutada), LAN (redes de área local), WAN (redes de área amplia), redes de telefonía celular, negocios propietarios e Internet.

La teoría de la ingeniería de tráfico fue concebida originalmente por AK Erlang, un matemático danés que desarrolló métodos de medición del tráfico de señales a principios del siglo XX. La ingeniería de tráfico hace uso de un concepto estadístico conocido como la ley de los grandes números (LLN), que establece que a medida que se repite un experimento, la frecuencia observada de un resultado específico se acerca a la frecuencia teórica de ese resultado en toda una población. En términos de telecomunicaciones, la LLN dice que el comportamiento general de una red grande se puede predecir con certeza razonable incluso si no se puede predecir el comportamiento de un solo paquete.

Cuando el nivel de tráfico de la red se acerca, alcanza o supera el máximo de diseño, se dice que la red está congestionada. En una red telefónica, el tráfico se mide en segundos de llamada (CCS) o erlangs. Un CCS es igual a 100 segundos de tiempo telefónico. Un erlang equivale a una hora o 36 CCS de tiempo telefónico. En una red congestionada, puede suceder una de estas tres cosas cuando un suscriptor intenta enviar un mensaje o realizar una llamada:

- El usuario recibe una señal de ocupado u otra indicación de que la red no puede realizar una llamada en ese momento.
- Un mensaje se coloca en una cola y finalmente se entrega de acuerdo con los parámetros especificados.
- Un mensaje es rechazado, devuelto o perdido.

Cuando las colas de mensajes se vuelven inaceptablemente largas o la frecuencia de las señales de ocupado se vuelve inaceptablemente alta, se dice que la red está en una condición de pérdida alta. Un objetivo principal de la ingeniería de tráfico es minimizar o eliminar las situaciones de grandes pérdidas. En particular, el número de mensajes rechazados o llamadas fallidas

los intentos deben estar lo más cerca posible de cero. Otro objetivo de la ingeniería de tráfico es equilibrar la QoS (Calidad de servicio) con el costo de operación y mantenimiento de la red.

En términos generales, los modelos de diseño de red clásicos consideran un gráfico con n nodos que describen la red y una matriz que proporciona la cantidad de tráfico que se entregará para cada par de nodos. El problema es seleccionar las capacidades de enlace y nodo para encontrar la red de costo mínimo capaz de enrutar todo el tráfico desde el origen hasta el destino. Este enfoque clásico del diseño de redes supone que todos los nodos pueden enrutar los flujos de tráfico y, por lo general, se lo conoce como diseño de una sola capa. Sin embargo, el efecto de multiplexación descrito anteriormente requiere que se consideren modelos más complejos donde algunos nodos (posiblemente no todos los nodos) pueden agregar flujos de tráfico y hay un costo asociado con esta operación de multiplexación. El enfoque más apropiado en este caso se basa en la descripción de la red multicapa, donde los nodos de la capa lógica superior pueden cambiar los flujos de tráfico entre flujos agregados, mientras que los nodos de la capa física inferior solo pueden cambiar los flujos agregados. Se pueden usar dos gráficos para representar la red, un gráfico físico y uno lógico. Los enlaces en el gráfico lógico se mapean en rutas de enlaces físicos. El problema aquí no es solo dimensionar el gráfico físico sino también diseñar la topología del gráfico lógico.

La mayoría de los trabajos previos sobre modelos de dos capas se centran en la optimización de la agregación y el enrutamiento de flujo y, en particular, en las redes de multiplexación por división de longitud de onda (WDM). En este contexto, el problema considerado suele denominarse problema de preparación, ya que el objetivo principal es agregar flujos para aprovechar mejor la gran capacidad disponible en cada longitud de onda. Para definir la topología lógica de la red WDM, también se deben asignar longitudes de onda a los caminos de luz y los convertidores ubicados donde sea necesario. Se pueden considerar diferentes variantes del problema, incluida la agregación de flujo jerárquico, pero se ha demostrado que incluso para topologías de red simples donde el enrutamiento es trivial, el problema de preparación es inherentemente difícil.

En las redes WDM, la resistencia a las fallas también es una característica importante y las técnicas de protección y restauración en diferentes capas se pueden aplicar y optimizar de manera conjunta. Desde la perspectiva de la tecnología de red, la integración de la capa óptica con las capas electrónicas dentro de una infraestructura óptica de datos convergente, basada en arquitecturas IP clásicas o GMPLS (MPLS generalizadas) modernas, es un elemento clave en la tendencia actual en la evolución de la red de banda ancha. Los problemas de diseño de redes de dos capas, donde también se incluye el dimensionamiento de enlaces y nodos en el modelo, se han considerado recientemente. Algunos trabajos consideran específicamente la tecnología MPLS y algunos abordan el problema de la ubicación del nodo MPLS. Dada la complejidad de los modelos de optimización, varios autores confían en las formulaciones de ruta y la generación de columnas junto con ramificación y límite, métodos de generación de filas y columnas conjuntas, ramificación y corte con desigualdades de conjuntos de corte o enfoques de descomposición basados en LP. Para redes medianas y grandes, la solución elegida sigue siendo los algoritmos heurísticos, que brindan una solución factible en un tiempo limitado.

Sin embargo, hasta donde sabemos, el efecto de la multiplexación estadística no se ha considerado previamente en tales modelos de enrutamiento y diseño de red.

Chun Tung Chou [1] propuso una arquitectura de red privada virtual usando MPLS, que permite granularidad y equilibrio de carga. Este artículo muestra un resultado factible desde el punto de vista de la utilización del enlace, pero la función multiobjetivo propuesta por el autor no es óptima en otros aspectos como el tiempo de respuesta y la pérdida de paquetes. Shekhar et al [2] introdujeron un factor de distorsión para flujos heterogéneos en la ingeniería de tráfico de redes troncales MPLS en presencia de túneles y restricciones de capacidad mediante la formulación de un problema de optimización discreto no lineal consciente de la distorsión. El autor presentó un enfoque heurístico de dos fases para resolver esta formulación de manera eficiente.

Francesco Palmieri [3] explica que la arquitectura jerárquica MPLS para redes conmutadas por etiquetas se puede utilizar para abordar todas las funciones requeridas de las redes convergentes/unificadas, desde la autenticación y configuración inicial del nivel de IP, la seguridad, el control de sesión, la reserva de recursos, el control de admisión y la calidad del servicio. y administración de políticas, mejorada solo donde sea necesario para abordar las idiosincrasias del entorno inalámbrico móvil. Esta arquitectura que abarca principalmente protocolos estandarizados IETF (Grupo de trabajo de ingeniería de Internet), aprovecha la flexibilidad de MPLS para abordar los requisitos específicos inalámbricos, como la micromovilidad, así como los requisitos específicos no inalámbricos.

Bosco et al [4] analizaron el rendimiento de una estrategia de ingeniería de tráfico (TE) para una red basada en MPLS, descrita en [5] se lleva a cabo. Concretamente se ha investigado y realizado la implementación basada en un plano de control distribuido (similar a Internet) mediante un banco de pruebas donde se han implementado protocolos de señalización real (RSVP-TE) y protocolos de enrutamiento (OSPF-TE). Todos estos trabajos previos brindan viabilidad pero un resultado óptimo en función de los requisitos actuales de los usuarios de Internet, como un menor tiempo de respuesta y una menor pérdida de paquetes.

5 Trabajo propuesto

El sistema propuesto involucra inteligencia de enjambre. La inteligencia de enjambre: la colonia de hormigas se utiliza para un control óptimo de la congestión. Los algoritmos de colonias de hormigas [6], [7] se han inspirado en el comportamiento de la colonia de hormigas real. El algoritmo puede encontrar la solución óptima generando hormigas artificiales. Mientras las hormigas reales buscan comida en su entorno, las hormigas artificiales buscan el espacio de solución. El movimiento probabilístico de las hormigas en el sistema les permite explorar nuevos caminos y volver a explorar los viejos caminos visitados. La fuerza del depósito de feromonas dirige a las hormigas artificiales hacia los mejores caminos y la evaporación de feromonas permite que el sistema olvide viejos

información y evitar la convergencia rápida a soluciones subóptimas. La selección probabilística de los caminos permite buscar un gran número de soluciones. ACO se ha aplicado con éxito a problemas de optimización discretos, como el problema del viajante de comercio [8], el enrutamiento [9] y el equilibrio de carga [10]. En [11] y [12] se pueden encontrar varias pruebas de la convergencia al camino óptimo del ACO. La implementación del sistema propuesto [13] [14] en el entorno cableado que proporciona un resultado óptimo y un enrutamiento libre de tráfico sugerido.

Los agentes hormiga se mueven en la red aleatoriamente para escanear una gran cantidad de nodos de red. Mientras se mueve, recopila información sobre la red y la entrega a los nodos de la red. Los algoritmos de esta categoría no utilizan los agentes para optimizar las rutas como en S-ACO o S-ACO meta-heurística [6]. Solo se usa para entregar información más actualizada sobre la red a los nodos de la red, lo que acelera el proceso de optimización. Esta categoría de algoritmo de agente móvil tipo hormiga ya se implementó con éxito en el algoritmo de enrutamiento tipo hormiga/GPS para redes ad hoc móviles y en el protocolo de enrutamiento híbrido de vector de distancia bajo demanda (AODV) antiadhoc.

Cada nodo de la red puede funcionar como nodo de origen, nodo de destino y/o nodo intermedio. Cada nodo tiene una tabla de feromonas y una tabla de enrutamiento. La tabla de enrutamiento se puede construir en función de la regla de transición de estado y la política de actualización de feromonas. Se aplica la siguiente regla proporcional aleatoria como regla de transición de estado: para el destino D, en el nodo i, la probabilidad de seleccionar un vecino j es

$$probabilidad(D, i, j) = \frac{Divertida(DT, i, j)}{\sum_{j \in N} Divertida(DT, i, j)} \quad (1)$$

Donde TD es el valor de la feromona correspondiente al vecino j en el nodo i y $0 < TD < 1$ es el valor heurístico local del enlace (i, j) o nodo j. $0 < \eta(i, j) < 1$ es el valor que puede representar la información del vecino (es decir, el retraso en la cola del vecino, la energía restante de la batería, la potencia de procesamiento, la relación señal-ruido del enlace, el ancho de banda del enlace, la tasa de error de bits, etc.). Fun (TD, i, j, -) es una función en TD y - (el valor de esta función es alto cuando TD y - son altos). N es el conjunto de todos los nodos vecinos factibles definidos por la información de la hormiga y las restricciones de enrutamiento (es decir, la garantía de bucle libre). Suponiendo que en un momento dado m1 hormigas han utilizado el primer puente y m2 el segundo, la probabilidad p1 de que una hormiga elija el primer puente es:

$$Divertida(DT, r, s) = \frac{\tau(r, s) \cdot \eta(r, s)}{\sum_{s \in N} \tau(r, s) \cdot \eta(r, s)} \quad (2)$$

0 - de lo contrario

donde $\tau(r, s)$ es la feromona depositada en el camino entre 'r' y 's', $\eta(r, s)$ es el valor heurístico correspondiente que es el inverso de la longitud del camino particular. β es un parámetro que determina la importancia relativa de la feromona frente a la distancia ($\beta > 0$). La política de actualización de feromonas es la siguiente:

$$\tau(r, s) = (1 - \alpha) \cdot \tau(r, s) + \alpha \cdot \tau(r, s) \quad (3)$$

$$T_k(r, s) = \begin{cases} \frac{1}{L_k} & \text{Si ruta fundar} \\ 0 & \text{de lo contrario} \end{cases} \quad (4)$$

Donde L es la duración del recorrido, α es el parámetro de decaimiento de feromonas que se encuentra entre '0 y 1'. Los valores de feromonas de cada entrada en la tabla se pueden inicializar a valores iguales, lo que proporciona una búsqueda imparcial de la mejor ruta. Si hay alguna información disponible sobre la mejor ruta, los valores de las feromonas de la entrada se pueden establecer en valores más cercanos al óptimo, acelerando así el algoritmo.

El algoritmo de optimización de colonias de hormigas:

1) //Fase de Inicialización

para cada par (r, s) , el valor de $T(r, s) := 0$ Fin de

Parak := 1 a m hacer

Dejar (r, k_1) ser la ciudad de partida para una hormiga k

$j_k(r_{k1}) := \{1, \dots, n\} - r_{k1}$

// $j_k(r_{k1})$ es el conjunto de ciudades aún por visitar para // ant k en Ciudad r_{k1}

$r_k := r_{k1}$

// r_k es la ciudad donde hormiga k se encuentra

Fin de

2) //Esta es la fase en la que las hormigas construyen sus recorridos. El recorrido de la hormiga k se almacena en //

Recorrido k

Parayo := 1 a m hacer

Si $y_o < n$

Entonces

Parak := 1 a m hacer

Elige la siguiente ciudad S_k

$j_k(S_k) := j_k(r_k) - S_k$

Excursión k(yo) := (r_k, S_k)

Final para

Demás

Parak := 1 a m hacer

//En este ciclo todas las hormigas regresan a la ciudad inicial r_{k1}

$S_k := r_{k1}$

Excursión k(yo) := (r_k, S_k)

Fin de

Terminara si

//En esta fase se produce la actualización local y se actualiza la feromona

Parak := 1 a m hacer

$T(r, s) = (1 - \tau) T(r, s) + \tau T(r, s) \quad r_k := S_k$ // Nueva

ciudad para hormiga k

Fin de

Fin de

3) //En esta fase se produce la actualización global y se actualiza la feromona

Para $k := 1$ a m hacer

Calcular $L_k // L_k$ es la duración del recorrido realizado por ant k

Fin de

Calcular L_{mejor}

/*Actualizar aristas pertenecientes a L_{mejor}

Para cada borde (r, s)

$\tau(r,s) - (1-\alpha) - \tau(r,s) - (1-\alpha) - \tau(r,s)$

Fin de

4) //En esta fase asigna prioridad

Para $k := 1$ a m hacer

Ordenar la tabla de enrutamiento según los valores de feromonas {

Asigna alta prioridad a la ruta de mayor densidad de feromonas Elija la

mejor ruta en función de la prioridad en la red cableada y la disponibilidad

y prioridad en el caso de la red inalámbrica

}

Fin de

Este enrutamiento propuesto se simula en NS2. El análisis de resultados y rendimiento se describe en la siguiente sección.

Tabla 1: Tabla de enrutamiento del nodo 'S' en caso de algoritmo de enrutamiento de ruta única

Destino	Siguiente salto	Prioridad	Feromona Densidad	Disponibilidad	Estado
D	I_1	1	20	Disponible	Ruta seleccionada
D	I_2	2	15	Disponible	Ruta en lista de espera
D	I_3	∞	15	No disponible	Rutas eliminadas

Tabla 2: Tabla de enrutamiento del nodo 'S' en caso de algoritmo de enrutamiento de rutas múltiples

Destino	próximo Brincar	Prioridad	Feromona Densidad	Probabilidad (feromona)	Disponibilidad	Estado
D	I_1	1	20	0.7	Disponible	70% paquetes transmitidos
D	I_2	2	15	0.3	Disponible	30% de paquetes transmitidos
D	I_4	∞	15	- -	No disponible	Rutas eliminadas

6 Análisis de resultados y rendimiento

Tabla 3. Tiempo de respuesta en enrutamiento de ruta única (en ms) bajo carga normal

	OSPF	ROTURA	COA
--	------	--------	-----

8Nodos	47	44	38
20Nodos	50	48	40
50Nodos	56	54	42
75Nodos	59	60	45
100Nodos	62	68	47
200Nodos	66	91	50

Tabla 4. Tiempo de respuesta en enrutamiento de ruta única (en ms) bajo carga media

	OSPF	ROTURA	COA
8Nodos	83	80	sesenta y cinco
20Nodos	88	86	68
50Nodos	97	97	72
75Nodos	104	108	78
100Nodos	108	123	81
200Nodos	116	164	86

Tabla 5. Tiempo de respuesta en enrutamiento de ruta única (en ms) bajo carga pesada

	OSPF	ROTURA	COA
8Nodos	145	144	112
20Nodos	153	155	116
50Nodos	170	175	122
75Nodos	182	194	133
100Nodos	189	222	139
200Nodos	202	295	148

Tabla 6. Rendimiento en enrutamiento de ruta única cableado (en kbps) en carga normal

	OSPF	ROTURA	COA
8Nodos	4	2	7
20Nodos	4	2	8
50Nodos	5	3	8
75Nodos	5	3	9
100Nodos	5	3	9
200Nodos	6	3	10

Tabla 7. Rendimiento en enrutamiento de ruta única (i n kbps) con carga media

	OSPF	ROTURA	COA
8Nodos	7	4	12

20Nodos	7	4	13
50Nodos	7	5	14
75Nodos	8	5	15
100Nodos	8	5	dieciséis
200Nodos	9	5	17

Tabla 8. Rendimiento en enrutamiento de ruta única (en kbps) bajo carga pesada

	OSPF	ROTURA	COA
8Nodos	12	8	21
20Nodos	12	8	22
50Nodos	13	8	24
75Nodos	14	9	26
100Nodos	15	9	27
200Nodos	dieciséis	9	29

Tabla 9. Pérdida de paquetes en enrutamiento de ruta única bajo carga normal (núm. número de paquetes)

	OSPF	ROTURA	COA
8Nodos	1	1	0
20Nodos	1	1	0
50Nodos	1	2	1
75Nodos	1	19	1
100Nodos	1	22	1
200Nodos	1	30	1

Tabla 10. Pérdida de paquetes en enrutamiento de ruta única (número de paquetes) bajo carga media

	OSPF	ROTURA	COA
8Nodos	2	1	1
20Nodos	2	1	1
50Nodos	2	4	1
75Nodos	2	35	1
100Nodos	2	40	1
200Nodos	3	53	1

Tabla 11. Pérdida de paquetes en enrutamiento de ruta única u bajo carga pesada (número de paquetes)

	OSPF	ROTURA	COA
8Nodos	3	2	1
20Nodos	3	2	1

50Nodos	4	6	1
75Nodos	4	63	2
100Nodos	4	72	2
200Nodos	5	96	2

Tabla 12. Tamaño del paquete de enrutamiento (en bytes)

OSPF	ROTURA	COA
44	44	dieciséis

Tabla 13. Sobre carga de enrutamiento en enrutamiento de ruta única con cable (en %)

	OSPF	ROTURA	COA	RACÓ
8Nodos	98	98	97	96
20Nodos	97	96	97	95
50Nodos	97	94	96	93
75Nodos	96	93	96	92
100Nodos	94	92	95	91
200Nodos	93	92	93	91

7 Referencias:

- [1] Chun Tung Chou, —Ingeniería de tráfico para redes privadas virtuales basadas en MPLS||, Computer Networks, Vol 44, pp319–333, 2004
- [2] Shekhar Srivastava, Appie van de Liefvoort, Deep Medhi, —Ingeniería de tráfico de redes troncales MPLS en presencia de flujos heterogéneos||, Computer Networks, Vol 53, pp2688–2702, 2009
- [3] Francesco Palmieri, —Una arquitectura basada en MPLS para QoS escalable e ingeniería de tráfico en redes IP móviles multiservicio convergentes||, Computer Networks, vol. 47, págs. 257–269, 2005
- [4] A. Boscoa, A. BottaB, G.ConteB, P. Iovannaa, R. Sabellaa, S.SalsanoC, —Control similar a Internet para la ingeniería de tráfico basada en MPLS: evaluación del rendimiento||, Evaluación del rendimiento, vol. 59, págs. 121 a 136, 2005
- [5] P. Iovanna, R. Sabella, M. Settembre, Estrategia de ingeniería de tráfico para redes multicapa basadas en el paradigma GMPLS, IEEE Netw. Vol.17 (2), págs. 28–37, 2003.
- [6] Di Caro G. y Dorigo M., "AntNet: un enfoque de agentes móviles para el enrutamiento adaptable", Tech. Rep. IRIDIA/97-12, Univ. Libre de Bruxelles, Bruselas, Bélgica, 1997
- [7] Schoonderwoerd R., Holland O., Bruten J., 'Agentes similares a hormigas para equilibrar la carga en redes de telecomunicaciones', en actas del primer int. conferencia sobre agentes autónomos, Nueva York, pp209-216, ACM Press 1997
- [8] Haibin Duan, Xiufen Yu, —Optimización de colonias de hormigas híbridas usando un algoritmo memético para el problema del vendedor ambulante||, Actas del simposio IEEE sobre programación dinámica aproximada y aprendizaje por refuerzo, P92-95, 2007

- [9] Subramanian D., Druschel P. y Chen J., 'Hormigas y aprendizaje por refuerzo: un estudio de caso sobre el enrutamiento en redes dinámicas', Actas del 15^o En t. conferencia conjunta sobre inteligencia artificial, San Francisco, pp823-838, Morgan Kaufmann, 1997
- [10] Kwang Mong Sim y Weng Hong Sun, —Optimización de colonias de hormigas para enrutamiento y equilibrio de carga: Encuesta y nuevas direcciones||, IEEE Transactions on Systems, Man, and Cybernetics, pp560-572, VOL. 33, núm. 5, 2003
- [11] Li-Ning Xing, Ying-Wu Chen, Peng Wang, Qing-Song Zhao, Jian Xiong, "Una optimización de colonias de hormigas basada en el conocimiento para problemas de programación de talleres flexibles", Applied Soft Computing, pp888-896, Vol 10 , 2010
- [12] Manuel Lopez-Ibanez, Christian Blum, —Beam ACO for the travelling sales man problem with time windows||, Computers & Operations Research, pp1570-1583, Vol 37, 2010
- [13] B. Chandra Mohan, R. Sandeep y D. Sridharan, —A Data Mining Approach for Predicting Reliable Path for Congestion Free Routing Using Self-motivated Neural Network||, Studies in Computational Intelligence, Springer, pp237-246, vol. 149, 2008
- [14] B. Chandra Mohan, R. Baskaran, —Algoritmo de evitación de enlaces redundantes para mejorar la eficiencia de la red||, International Journal of Computer Science Issues, Vol. 7, Número 3, mayo de 2010