

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Proyecto Fin de Carrera

Planificación de redes ópticas wavelength-routing con demanda dinámica de tráfico



AUTOR: M^a Belén García Manrubia
DIRECTOR: Pablo Pavón Mariño
CODIRECTOR: Ramón Aparicio Pardo

Noviembre / 2007





Autor	María Belén García Manrubia
E-mail del Autor	belen_gmanrubia@hotmail.com
Director(es)	Pablo Pavón Mariño
E-mail del Director	pablo.pavon@upct.es
Codirector(es)	Ramón Aparicio Pardo
Título del PFC	"Planificación de Redes Ópticas Wavelength-Routing con Demanda dinámica de Tráfico"
Descriptor(es)	Planificación de redes WDM. Herramientas de planificación Diseño de topologías virtuales en redes WDM
Resumen	<p>Este proyecto presenta una herramienta para la planificación de redes ópticas Wavelength Routing (WR) de manera dinámica. La herramienta ha sido desarrollada en MATLAB, y optimiza tanto el problema RWA (<i>Routing-and-Wavelength Assignment</i>), como el problema de encaminamiento de los flujos de capas altas sobre la topología virtual. El proyecto incluye una interfaz gráfica para el acceso al kernel de la herramienta, diseñada específicamente para facilitar la utilización en docencia. Se incluyen una serie de algoritmos de ejemplo y los resultados obtenidos para ejemplificar el uso de la aplicación.</p>
Titulación	Ingeniería de Telecomunicación
Intensificación	
Departamento	Tecnologías de la Información y las Comunicaciones
Fecha de Presentación	Noviembre del 2007





ÍNDICE

ÍNDICE.....	V
ÍNDICE DE FIGURAS.....	VII
ÍNDICE DE TABLAS	IX
CAPÍTULO 1: INTRODUCCIÓN.....	1
1 REDES ÓPTICAS WDM	1
1.1 <i>Introducción.....</i>	<i>1</i>
1.2 <i>Redes Ópticas</i>	<i>2</i>
1.3 <i>Wavelength Division Multiplexing.....</i>	<i>2</i>
2 EVOLUCIÓN DE LAS REDES WDM	6
2.1 <i>Sistemas WDM punto a punto.....</i>	<i>6</i>
2.2 <i>Wavelength Add/Drop Multiplexer (WADM)</i>	<i>7</i>
2.3 <i>Fibras y matrices de conmutación de longitudes de onda, estrellas pasivas, routers pasivos y switches activos</i>	<i>7</i>
3 REDES WDM WAVELENGTH-ROUTED	10
3.1 <i>Redes ópticas de área extendida wavelength-routed.....</i>	<i>10</i>
3.2 <i>Paradigma de las Redes WDM</i>	<i>13</i>
4 OBJETIVOS Y DISTRIBUCIÓN DEL DOCUMENTO.....	14
CAPÍTULO 2: PLANIFICACIÓN DINÁMICA DE REDES WAVELENGTH-ROUTING	15
1 PLANIFICACIÓN ESTÁTICA VERSUS DINÁMICA.....	15
2 HERRAMIENTAS CON PLANIFICACIÓN DINÁMICA.....	16
3 PROBLEMA RWA CON GROOMING	18
3.1 <i>Planificación de Flujos Dinámicos.....</i>	<i>20</i>
CAPÍTULO 3: DYN-MATPLANWDM	25
1 ESTRUCTURA DEL TOOLBOX	25
1.1 <i>Parámetros de Entrada.....</i>	<i>25</i>
1.2 <i>Event loop.....</i>	<i>26</i>
1.3 <i>Estadísticos calculados.....</i>	<i>26</i>



1.4 Estructura de Directorios	27
1.5 Estructura de EventLoop	31
1.6 Estructura del Cálculo de Estadísticos.....	32
2 MANUAL DE USUARIO	33
2.1 Instalación y arranque de la aplicación	33
2.2 Ventana Principal.....	33
2.3 Estructura del Fichero .phys	38
2.4 Estructura del Fichero .traff.....	39
2.5 Estructura de los Módulos Generador	40
2.6 Estructura de los Módulos Planificador.....	40
2.7 Estructura de las variables currentNetworkState	41
2.8 Extensión de la herramienta.....	42
3 HEURÍSTICOS IMPLEMENTADOS	42
3.1 Generación	42
3.2 Planificación.....	43
CAPÍTULO 4: RESULTADOS Y EJEMPLOS DE UTILIZACIÓN	47
CAPÍTULO 5: CONCLUSIONES Y LÍNEAS FUTURAS	55
REFERENCIAS	57



ÍNDICE DE FIGURAS

FIGURA 1 IMPLEMENTACIÓN TDM	1
FIGURA 2 IMPLEMENTACIÓN CDM	1
FIGURA 3 VENTANA ÓPTICA DE BAJA ATENUACIÓN.....	3
FIGURA 4 VARIOS CANALES WDM SE PROPAGAN POR UNA ÚNICA FIBRA	3
FIGURA 5 DIAGRAMA DE UN SISTEMA SIMPLE WDM	4
FIGURA 6 SISTEMA DE TRANSMISIÓN WDM PUNTO A PUNTO CON REGENERACIÓN.....	5
FIGURA 7 SISTEMA DE TRANSMISIÓN WDM CON AMPLIFICADORES Y CUATRO CANALES PUNTO A PUNTO	6
FIGURA 8 WAVELENGTH ADD/DROP MULTIPLEXER (WADM).....	7
FIGURA 9 ESTRELLA PASIVA 4 × 4.....	8
FIGURA 10 ROUTER PASIVO 4 × 4 (CUATRO LONGITUDES DE ONDA).....	9
FIGURA 11 SWITCH PASIVO 4 × 4 (CUATRO LONGITUDES DE ONDA)	9
FIGURA 12 RED WDM WAVELENGTH-ROUTED DE ÁREA EXTENDIDA.....	11
FIGURA 13 POSIBLES CAPAS EN UNA RED DE TRANSPORTE WDM.....	1
FIGURA 14 RED DE LIGHTPATHS SOBRE UNA RED WDM (LOS CÍRCULOS NEGROS INDICAN EXTREMOS DE UN LIGHTPATH)	13
FIGURA 15 INTERFAZ GRÁFICA DE USUARIO DE MERLÍN.....	16
FIGURA 16 VENTANA PRINCIPAL DE LA APLICACIÓN TSC DE GLASS	17
FIGURA 17 INTERFAZ GRÁFICA DE USUARIO DE METROWAND.....	18
FIGURA 18 RESTRICCIÓN DE LONGITUD DE ONDA Y BLOQUEO.....	1
FIGURA 19 DIAGRAMA DEL TOOLBOX DE DYN-MATPLANWDM	25
FIGURA 20 ESTRUCTURA DE DIRECTORIOS DE DYN-MATPLANWDM	27
FIGURA 21 ESTRUCTURA DEL MÓDULO EVENTLOOP.....	31
FIGURA 22 VENTANA PRINCIPAL DE DYN-MATPLANWDM	33
FIGURA 23 PANEL INPUT DATA.....	34
FIGURA 24 PANEL RESULTS INFORMATION	35
FIGURA 25 PANEL INPUT DATA CUANDO LA SIMULACIÓN ES DETENIDA	35



FIGURA 26 EDIT RESULTADOS	36
FIGURA 27 PANEL PHYSICAL TOPOLOGY	37
FIGURA 28 PANEL VIRTUAL TOPOLOGY.....	37
FIGURA 29 ARCHIVO .PHYS	1
FIGURA 30 ARCHIVO .TRAFF.....	1
FIGURA 31 CASUÍSTICAS DE UTILIZACIÓN DE UN NODO VECINO	44
FIGURA 32CREACIÓN DE DOS NUEVOS LIGHTPATHS	44
FIGURA 33 RED NSFNET DE 14 NODOS	47
FIGURA 34 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA	1
FIGURA 35 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA	50
FIGURA 36 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA	1
FIGURA 37 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA	1
FIGURA 38 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA	1
FIGURA 39 PROBABILIDAD DE BLOQUEO VS PARÁMETRO K.....	53
FIGURA 40 PROBABILIDAD DE BLOQUEO VS PARÁMETRO K.....	1



ÍNDICE DE TABLAS

TABLA 1 ALGORITMOS DE PLANIFICACIÓN DINÁMICOS PARA EL PROBLEMA RWA O SUS SUBPROBLEMAS.....	21
TABLA 2 RELACIÓN DE LAS PRINCIPALES FUNCIONES.....	28



CAPÍTULO 1: INTRODUCCIÓN

1 REDES ÓPTICAS WDM

1.1 INTRODUCCIÓN

Las tecnologías de fibra óptica tienen una capacidad potencialmente ilimitada en comparación con otras tecnologías existentes: gran ancho de banda (cercano a los 50 terabits por segundo (Tbps)), baja atenuación de la señal (0.2 dB/km), baja distorsión, bajo requerimiento de energía, requiere poco espacio y material, y presenta un bajo coste. El ancho de banda potencial de una fibra monomodo está cercano a los 50 Tbps, mientras que las actuales tasas de tecnología electrónica se sitúan entorno a decenas de gigabits por segundo (Gbps). Por esta razón, es claro que se debe invertir esfuerzo en explotar esta diferencia abismal entre el ancho de banda ofrecido por el dominio eléctrico y el óptico. En una red de comunicaciones ópticas la concurrencia puede implementarse mediante distintas técnicas como multiplexación de longitud de onda-frecuencia (WDM), ranuras temporales (TDM), forma de onda (espectro ensanchado, multiplexación por división de código (CDM)) o una combinación de las mismas.

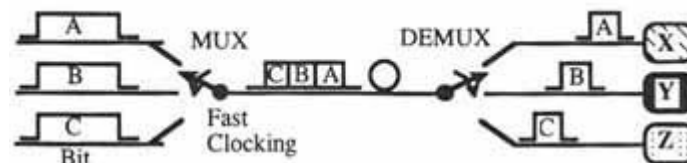


FIGURA 1 IMPLEMENTACIÓN TDM

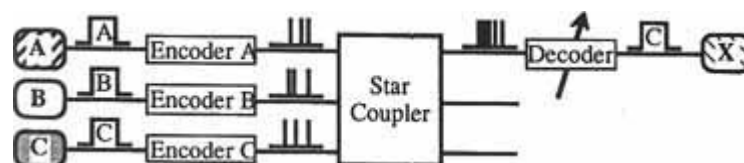


FIGURA 2 IMPLEMENTACIÓN CDM

Si se opta por la tecnología TDM, tendremos que cada usuario final sería capaz de sincronizarse dentro de una ranura (“slot”) temporal. La tasa de bit de una multiplexación TDM es la tasa agregada sobre todos los canales TDM del sistema, mientras que la tasa de chip CDM puede ser mucho más alta que la suma de cada usuario. Como resultado, las tasas de ambas tecnologías serán mucho más altas que la velocidad de procesamiento electrónico, lo que significa que alguna



de las partes de la red deben operar a una velocidad más alta que la velocidad eléctrica. Esta restricción hace que las tecnologías TDM y CDM sean relativamente menos atractivas que WDM.

Como consecuencia de lo dicho anteriormente, la tecnología óptica de multiplexación actualmente mejor posicionada en las redes de comunicaciones ópticas es WDM debido a que todos los equipos finales requieren trabajar sólo a la tasa de bit de un canal WDM, que puede ser elegida arbitrariamente, e.g., velocidad de pico en el procesado electrónico. Por consiguiente, los esfuerzos más importantes se están haciendo en el desarrollo y aplicación de WDM y en su comercialización.

1.2 REDES ÓPTICAS

Las redes ópticas son sensiblemente de mayor capacidad que las redes tradicionales, en las que el formato de los datos son electrones que viajan a través de cables de cobre. Las fibras ópticas se basan en el principio de la reflexión total, que fue descubierto en torno a 1850 y fue desarrollado con los primeros endoscopios a principios del siglo XX. La utilización de fibras de vidrio para comunicarse fue propuesta por primera vez por Kao y Hockham en 1966. La manufactura de fibras ópticas comenzó en la década de los 70. A finales de los 80 y principios de los 90 aparecieron una gran variedad de redes ópticas que utilizaban fibras ópticas en sustitución de cables de cobre para alcanzar velocidades más altas. Diversas interconexiones de computadores como ESCON (Enterprise Serial Connection), Fiber Channel y HiPPI (High Performance Parallel Interface) utilizan componentes ópticos de baja tasa de bit a bajo coste. FDDI (Fiber Distributed Data Interface) utiliza ambos, token rings de fibra óptica para proporcionar velocidades de 100-200 megabits por segundo (Mbps) en las transmisiones entre estaciones de trabajo. SONET/SDH (Synchronous Optical NETWORK en América del Norte, Synchronous Digital Hierarchy en Europa y Asia)¹, que conforma la base de las actuales redes troncales de alta velocidad y es uno de los estándar con más implantación en la industria de redes, permite tasas de transmisión similares a OC-768², entorno a 40 gigabits por segundo (Gbps).

1.3 WAVELENGTH DIVISION MULTIPLEXING

El ancho de banda teórico de la fibra óptica se sitúa entorno a los 25 terahercios (THz) en la banda de baja atenuación de 1,55 micras tal y como vemos en la Figura 3. Sin embargo, no hay que olvidar la limitación impuesta en la tecnología electrónica de los usuarios finales que sólo permite que las velocidades alcanzadas (tasas de datos) sean de unos pocos gigabits por segundo (Gbps). Así, este “cuello de botella” electrónico dificulta el aprovechamiento del ancho de banda potencial que ofrece una fibra mediante un solo canal (longitud de onda). Los avances recientes que han permitido alcanzar velocidades de terabits por segundo (Tbps) son el resultado de dos

¹ Conjunto de estándares para la transmisión digital de información sobre redes ópticas

² OC- n [*Optical Carrier- n*] especifica una tasa electrónica de $n \times 51.84$ Mbps aproximadamente; por lo tanto OC-48 y OC-192 se corresponde con velocidades de 2.5 Gbps y 10 Gbps respectivamente. OC-768 [40 Gbps] es el próximo objetivo para el techo de las velocidades electrónicas,



desarrollos diferentes. Por un lado, la tecnología *wavelength division multiplexing* (WDM) envía simultáneamente haces de luz a diferentes longitudes de onda a través del núcleo de una fibra óptica; y por otro, *erbium-doped fiber amplifier* (EDFA) amplifica simultáneamente un conjunto de señales a diferentes longitudes de onda, sin tener en cuenta su velocidad o tipo de modulación.

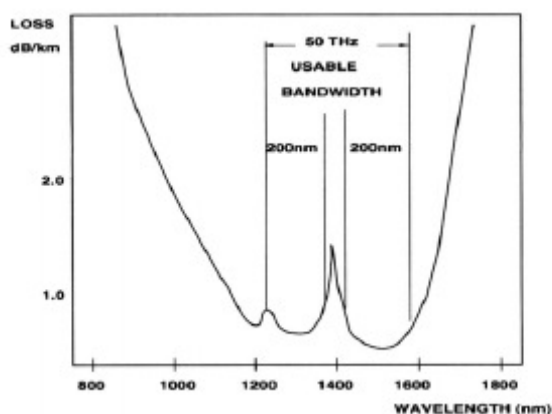


FIGURA 3 VENTANA ÓPTICA DE BAJA ATENUACIÓN

Conceptualmente se puede trazar un paralelismo entre WDM y FDM (*frequency division multiplexing*), pues varias señales de información modulan una señal óptica a diferentes longitudes de onda y las resultantes se combinan y transmiten a través de la misma fibra óptica.

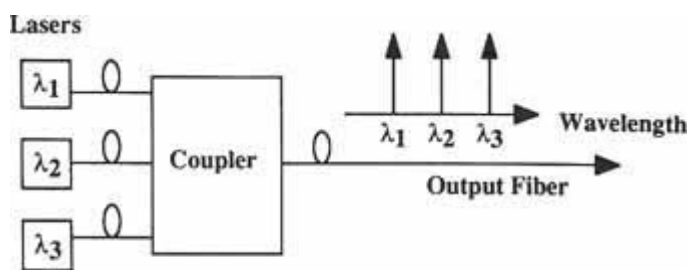


FIGURA 4 VARIOS CANALES WDM SE PROPAGAN POR UNA ÚNICA FIBRA

Los prismas y las rejillas de difracción conocidos se utilizan para combinar (multiplexar) o separar (demultiplexar) las longitudes de onda en distintas señales. A diferencia de las usadas en FDM, las rejillas de difracción en un sistema óptico WDM son totalmente pasivas y mucho más fiables. Además las ondas portadoras de cada canal WDM tienen una frecuencia un millón de veces mayor que uno FDM (THz vs MHz). Dentro de cada canal WDM es posible introducir un canal FDM mediante la subdivisión del ancho de banda en diferentes canales de radio, cada uno a su frecuencia correspondiente. Esta técnica se denomina *subcarrier multiplexing*. Una única longitud



de onda también puede ser dividida entre varios nodos de la red mediante *time division multiplexing* (TDM). Se puede comprobar como WDM elimina el “cuello de botella” electrónico dividiendo el espectro de transmisión (banda de 1.55 micras) en un conjunto de canales de forma que las longitudes de onda no se solapen y coexistan en una misma fibra; cada longitud de onda soporta un canal de comunicación que opera a la velocidad de pico electrónica. Gracias a WDM se puede disponer de un incremento sustancial de ancho de banda sin la necesidad de grandes inversiones en el despliegue de nuevas fibras.

A continuación vemos un ejemplo sencillo para apreciar la revolución de WDM. Para transmitir 40 Gbps sobre 600 Km en un sistema tradicional se requerían 16 pares de fibra y 224 regeneradores ($[600/40-1]16$), necesitando uno cada 40 Km. Con un sistema de transmisión WDM se puede usar un único par de fibra óptica con 16 canales y 4 amplificadores ($600/120 -1$), colocados cada 120 Km.

Comprobamos como WDM ha mejorado de forma notable la capacidad con sólo introducir dos, tres o cuatro longitudes de onda adicionales en los sistemas de transmisión punto a punto ya instalados. Ya están disponibles sistemas WDM de 16 longitudes de onda en *OC-48* y 32 longitudes de onda en *OC-192*, que proporcionan tasas agregadas de 40 y 320 Gbps respectivamente. La técnica *dense wavelength division multiplexing* (DWDM) permite incrementar de forma efectiva el número de canales en la fibra mediante la explotación intensiva de la banda espectral, es decir, disminuyendo la distancia entre canales (aumento de la densidad). El espaciado entre canales tradicional oscila de 0.4 a 4 nm, equivalente en frecuencia a un rango 50 – 500 GHz.

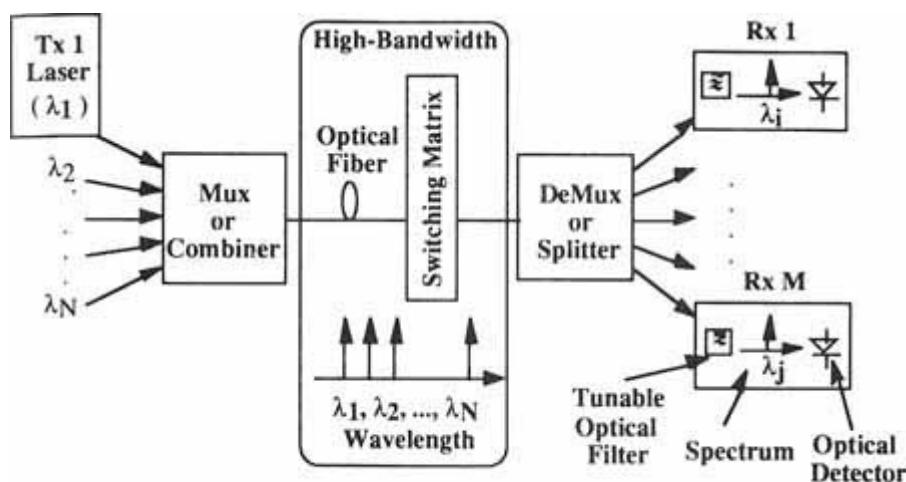


FIGURA 5 DIAGRAMA DE UN SISTEMA SIMPLE WDM

Como ya sabemos de la teoría óptica, cuando los problemas de atenuación son resueltos, los efectos de la dispersión se hacen importantes. De forma similar, cuando se controlan los



problemas asociados a la dispersión, los efectos no lineales³ se hacen dominantes, éstos limitan potencialmente la tasa de transmisión máxima de los sistemas de comunicación WDM. Las causas de los efectos no lineales son la interacción entre la señal óptica y las moléculas de sílice de la fibra o entre aquélla y las ondas acústicas, también es importante la dependencia del índice de refracción con la intensidad óptica de las señales que atraviesan la fibra. Estas no linealidades se pueden controlar en las fibras con la elección cuidadosa de la potencia y el espaciado de los canales.

La llegada de EDFA permite la comercialización de los sistemas WDM, encontrando una vía para amplificar todas las longitudes de onda al mismo tiempo, independientemente de sus tasas de transmisión individuales, tipos de modulación o niveles de potencia. Antes de la aparición de los EDFAs, los efectos de las pérdidas ópticas eran compensados cada 10 Km por regeneradores electrónicos, que requerían una conversión opto-electrónica y viceversa. La limitación fundamental proviene de que los regeneradores electrónicos trabajan sólo para la tasa especificada a una longitud de onda, sin considerar retardos ni consumo. El amplificador EDFA contiene varios metros de fibra de vidrio de sílice que ha sido dopada con iones de erbio, metal lantánido ubicado en las “tierras raras”. A través de un láser de bombeo óptico se activan los iones de erbio, que amplifican las señales ópticas que atraviesan la fibra. Es una providencial coincidencia que la banda de amplificación del EDFA (1,53 – 1,54 micras) sea la misma que la banda de transmisión de las fibras ópticas (1,55 micras).

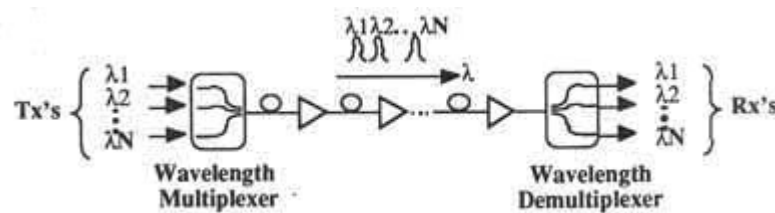


FIGURA 6 SISTEMA DE TRANSMISIÓN WDM PUNTO A PUNTO CON REGENERACIÓN

En la práctica existen tres tipos de regeneradores de señal: los regeneradores de 3R, que excluye las operaciones de *reshaping* y *reclocking*⁴; los regeneradores de 2R, que sólo ejecuta la operación de *reshaping* y los regeneradores de 1R, que sin operaciones de *reshaping* y *reclocking*, llevan a cabo una amplificación simple usando EDFAs u otros amplificadores.

Se llama transponedor a un elemento de red que combina el receptor óptico, varios niveles de regeneración y un transmisor óptico. Si un transponedor es capaz de transmitir a una longitud de

³ Dispersión estimulada Raman, dispersión estimulada Brillouin, modulación self- y cross-phase y el mezclado de cuatro ondas.

⁴ *Reshaping* de la señal reproduce el pulso original de cada bit, eliminando la mayoría del ruido. *Reclocking* de la señal sincroniza la señal con el patrón de tiempo original y la tasa de bit.



onda diferente a la de recepción, se lleva a cabo una conversión de longitud de onda como resultado de la regeneración.

2 EVOLUCIÓN DE LAS REDES WDM

2.1 SISTEMAS WDM PUNTO A PUNTO

La tecnología WDM está siendo desplegada por varias empresas de telecomunicación en forma de comunicaciones punto a punto. Este despliegue está siendo impulsado por la incremento en la demanda de ancho de banda. Cuando los niveles de demanda superen las capacidades ofrecidas por las fibras existentes, WDM se convertirá en la alternativa más rentable. Así, un estudio comparó los costes relativos de actualizar los enlaces de transmisión punto a punto de *OC-48* a *OC-192* con tres alternativas diferentes:

- i. Solución “multi-fibra”. Instalación de fibras y equipamientos adicionales.
- ii. Solución WDM de “cuatro-canales” (ver Figura 7). Se trata de un multiplexador WDM que combina cuatro fuentes de datos, cada una en una única longitud de onda y los envía a través de la fibra. En recepción un demultiplexador separa los cuatro canales a sus flujos originales.
- iii. Solución *OC-192* “alta velocidad electrónica”.

Diversos análisis demuestran que para distancias menores de 50 Km, la solución “multifibra” es la menos cara, pero para distancias del enlace de transmisión superiores a 50 Km, el coste de la solución WDM es el menor, seguido de cerca por la solución “alta velocidad electrónica”.

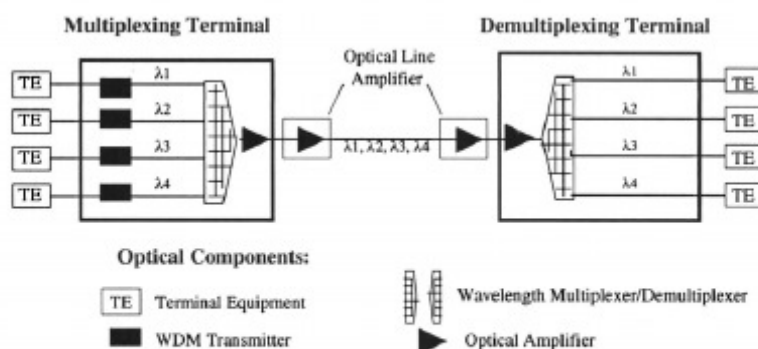


FIGURA 7 SISTEMA DE TRANSMISIÓN WDM CON AMPLIFICADORES Y CUATRO CANALES PUNTO A PUNTO



2.2 WAVELENGTH ADD/DROP MULTIPLEXER (WADM)

Seguidamente (Figura 8) se muestra la estructura de un *Wavelength Add/Drop Multiplexer* (WADM) consistente en un demultiplexor seguido de una etapa de *switches* 2×2 con un *switch* por longitud de onda y terminando en un multiplexor. El WADM se puede insertar en un enlace físico de fibra. Si todos los *switches* 2×2 están en estado no activo, todas las longitudes de onda atraviesan el WADM sin alteración. En cambio, en el caso de estar alguno de los *switches* en estado activo (como la longitud de onda λ_i en la Figura 8), la señal de la correspondiente longitud de onda es “extraída” de forma local y un flujo de datos entrante se puede “añadir” con la misma longitud de onda en el nodo donde se ubica el WADM. En el caso de que la interfaz del WADM tenga el hardware necesario y la capacidad de procesamiento se pueden “añadir y extraer” más de una longitud de onda.

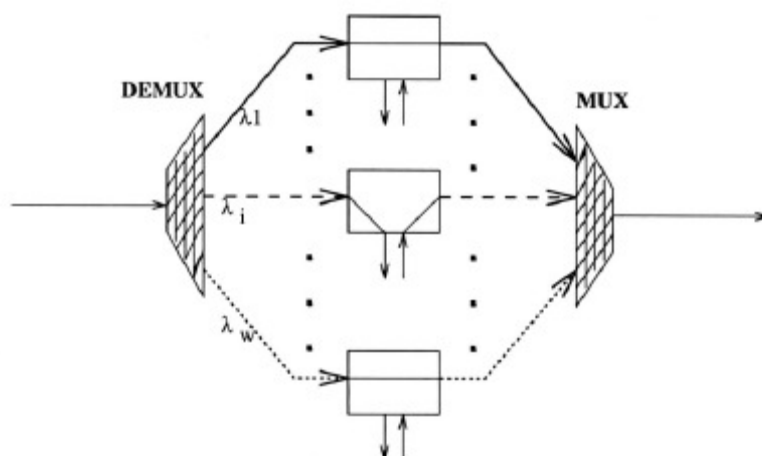


FIGURA 8 WAVELENGTH ADD/DROP MULTIPLEXER (WADM)

2.3 FIBRAS Y MATRICES DE CONMUTACIÓN DE LONGITUDES DE ONDA, ESTRELLAS PASIVAS, ROUTERS PASIVOS Y SWITCHES ACTIVOS

Para manejar una red de enlaces de fibra con distintas longitudes de onda, se requieren dispositivos de interconexión adecuados, que se agrupan en tres categorías:

- i. Estrella pasiva (ver Figura 9)
- ii. Router pasivo (ver Figura 10)
- iii. Switch activo (ver Figura 11)



La “estrella pasiva” recibe una señal en uno de sus puertos de entrada y a una longitud de onda determinada, divide la potencia de la señal entre todos los puertos salida y a la misma longitud de onda que la recibió la envía por todos ellos. En la Figura 9 las señales con longitudes de onda λ_1 y λ_4 de los puertos *Input Fiber 1* y *Input Fiber 4* respectivamente, son enviadas a todos los puertos de salida en una operación de *broadcast*. Es posible dar lugar a una colisión en el caso de que dos o más señales de entrada salgan de la estrella a la misma longitud de onda. Si se asume que hay tantas longitudes de onda como puertos, una estrella pasiva $N \times N$ puede enrutar N conexiones simultáneas.

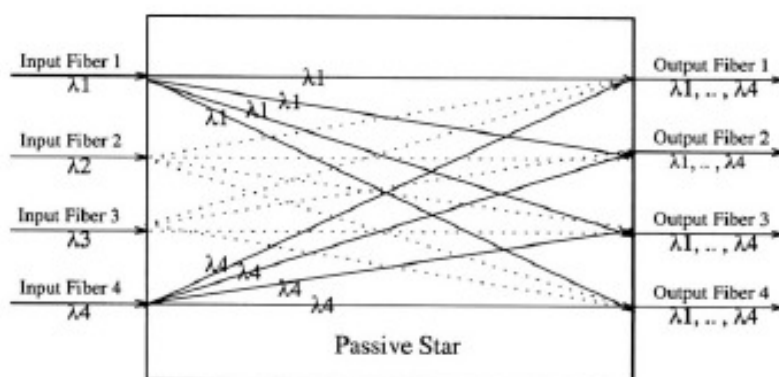


FIGURA 9 ESTRELLA PASIVA 4×4

Un “router pasivo” puede enrutar de forma separada cada una de las longitudes de onda entrantes en distintos puertos de salida a la misma longitud de onda, e.g., las longitudes de onda λ_1 , λ_2 , λ_3 y λ_4 entrantes en el *Input Fiber 2* se enrutan de forma alternada en los cuatro puertos de salida (ver Figura 10). El “router pasivo” permite la reutilización de longitudes de onda, es decir, la misma longitud de onda se puede usar en diversas conexiones a través de *router*. La “matriz de enrutamiento” caracteriza al *router* y determina que longitud de onda entrante se enruta al correspondiente puerto de salida, esta matriz está conformada por las conexiones internas entre el demultiplexor y multiplexor. La matriz de enrutamiento es estática y no admite cambios. Esta familia de *routers* están disponibles en el mercado y son conocidos como *Latin routers*, *waveguide grating routers* (WGRs), *wavelength routers* (WRs), etc. Si se asume que hay tantas longitudes de onda como puertos, un “router pasivo” puede enrutar N^2 conexiones simultáneas (en comparación con las N que podía enrutar la “estrella pasiva”), si bien carece de la capacidad *broadcast* de aquella.

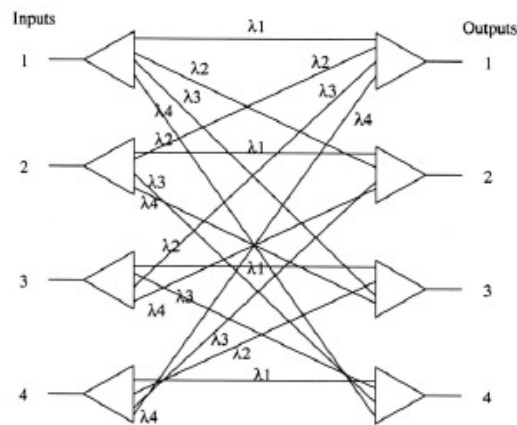


FIGURA 10 ROUTER PASIVO 4 × 4 (CUATRO LONGITUDES DE ONDA)

De forma similar el “switch activo” permite la reutilización de longitudes de onda y puede enrutar N^2 conexiones simultáneas. Además tiene una ventaja importante sobre el “router pasivo”, su matriz de enrutamiento puede ser reconfigurada mediante control electrónico según la demanda, presentando como inconveniente su necesidad de ser alimentado por una fuente externa. El “switch activo” se denomina de forma mayoritaria *wavelength-routing switch* (WRS) aunque también es referido como *wavelength selective crossconnect* (WSXC).

Se puede añadir una funcionalidad adicional al “switch activo”, la posibilidad de convertir la longitud de onda antes de entrar en la etapa de multiplexado, de forma que la señal se transmite a la salida con una longitud de onda diferente a la entrante. Los “switches activos” que presentan esta propiedad se denominan *wavelength-convertible switch* o *wavelength interchanging crossconnect* (WIXC).

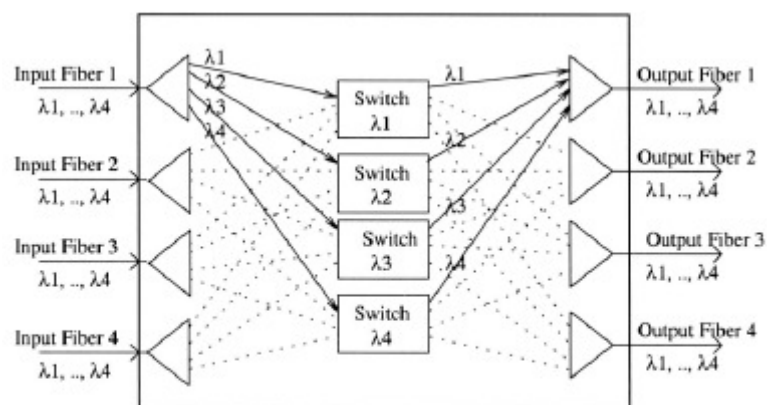


FIGURA 11 SWITCH PASIVO 4 × 4 (CUATRO LONGITUDES DE ONDA)



Las aplicaciones de los tres dispositivos vistos son diferentes: la “estrella pasiva” se usa en entornos locales WDM, el “router pasivo” tiene distintas aplicaciones como dispositivo mux/demux y el “switch activo” se emplea en redes ópticas de área extensa.

3 REDES WDM WAVELENGTH-ROUTED

3.1 REDES ÓPTICAS DE ÁREA EXTENDIDA WAVELENGTH-ROUTED

Las redes WDM *wavelength-routed* presentan la peculiaridad de evitar los tres problemas principales de las redes *broadcast* vistas en el apartado anterior: reutilización de longitudes de onda, pérdidas de potencia por división de señal y escalabilidad a redes de área extensas (WANs). En esencia una red *wavelength-routed* consiste en un conjunto de nodos de enrutamiento (WXC) interconectados por enlaces de fibra punto a punto dentro de una topología de red arbitraria. Cada nodo final está conectado a un WXC por un enlace de fibra. El conjunto de nodos finales y su correspondiente WXC se referencia como nodo de la red. Cada nodo está equipado con un conjunto de transmisores y receptores, susceptibles de ser sintonizados a la longitud de onda deseada, para enviar y recibir datos a/de la red.

En una *red wavelength-routed* los mensajes se envían de un nodo a otro usando una longitud de onda continua denominada camino de luz o *lightpath*, que no requiere ningún mecanismo de reconversión opto-electrónica ni *buffers* en los nodos intermedios. Este proceso se llama *enrutamiento de longitud de onda* y en él, los nodos intermedios encaminan los *lightpaths* en el dominio óptico mediante sus respectivos WXC. Los nodos finales de los *lightpath* acceden usando transmisores y receptores sintonizados a la longitud de onda que opera.

Un *lightpath* se puede definir como una vía de comunicación en el dominio óptico entre dos nodos, establecida con la misma longitud a través de la ruta que sigue la señal transmitida. Por lo tanto constituye una vía de gran ancho de banda, capaz de transmitir datos a varios gigabits por segundo y está caracterizada de forma unívoca por una ruta física y una longitud de onda. La restricción de que sea la misma longitud de onda utilizada a lo largo de toda la ruta física se conoce como *restricción de continuidad de longitud de onda*. Conviene destacar sin embargo que dos *lightpaths* pueden usar la misma longitud de onda si utilizan un conjunto de enlaces disjuntos. Esta propiedad se conoce como *reutilización de longitud de onda*.

En comparación con otras arquitecturas de redes ópticas, las redes *wavelength-routed* presentan ventajas considerables. La propiedad de reutilizar las longitudes de onda en conjuntos disjuntos de enlaces asegura una mayor escalabilidad de la red, respecto a las redes *broadcast-and-select* vistas en el apartado anterior. Una característica importante, que permite a las redes *wavelength-routed* extenderse hasta grandes distancias, es que la potencia de los *lightpaths* no se divide en destinos irrelevantes.

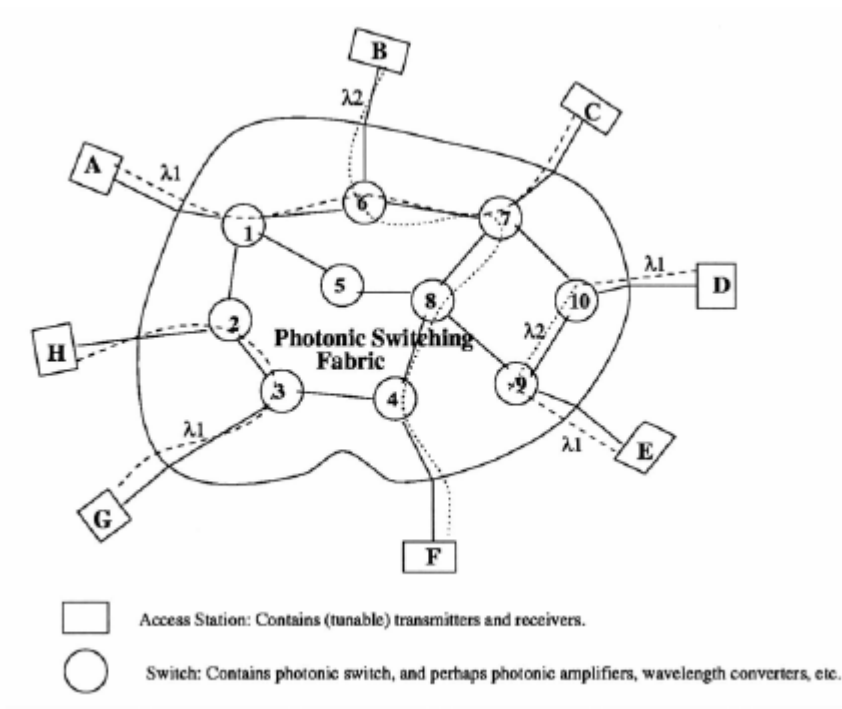


FIGURA 12 RED WDM WAVELENGTH-ROUTED DE ÁREA EXTENDIDA

En este escenario, dada una red WDM, el problema fundamental se centra en el enrutamiento y la asignación de longitudes de onda a los *lightpaths*. Existen diversos métodos y algoritmos que abordan la resolución de ambos problemas y a la vez minimizan el número de longitudes de onda utilizadas. El número de longitudes de onda que admite una fibra óptica está entre 4 y 32, aunque las líneas actuales de investigación aseguran hasta 100 longitudes de onda por fibra, y esta variable juega un papel fundamental en la resolución de ambos problemas en función de los recursos disponibles.

La conmutación de paquetes sobre redes *wavelength-routed* se puede modelar mediante sistemas mono-salto o multi-salto, de forma similar a las redes *broadcast-and-select*. En la aproximación multi-salto, una topología virtual⁵ se superpone sobre la topología física mediante los WXC's en los nodos. Sobre esta topología virtual los paquetes se encaminan a través de los nodos intermedios antes de alcanzar el destino final. En cada uno de los nodos intermedios el paquete se convierte al dominio electrónico para su procesamiento, siguiendo la operación inversa y su retransmisión en otra longitud de onda.

Una red de transporte WDM se puede descomponer en tres capas: capa de medios físicos, capa óptica y capa del cliente, tal y como se muestra en el esquema de la Fig. 1.14. El desarrollo de las tecnologías WDM ha introducido la capa óptica entre la inferior netamente física y la superior de acceso a los usuarios finales. Como se ha comentado anteriormente un conjunto de *lightpaths* forma la capa óptica (topología virtual); proveyendo a los clientes de la capa superior transparencia de protocolos y conmutación de circuitos. Esto es posible por la flexibilidad que presentan los

⁵ Conjuntos de *lightpaths* establecidos de forma lógica. También se denomina *red de lightpaths* o *topología lógica*



lightpaths de transportar los paquetes a distintas tasas de bit y sobre distintos protocolos y, consecuentemente, a distintos clientes de forma concurrente. Por ejemplo, dentro de una misma red algunos *lightpath* pueden transportar flujos de SONET, mientras que otros lo harán con paquetes IP o celdas ATM. Una red con una capa óptica puede ser configurada para reenrutar los *lightpaths* por vías alternativas de forma automática, en caso de haber un fallo en el sistema, mecanismo que proporciona fiabilidad a la red.

A su vez, y de acuerdo con Recomendación G.872 de la Unión Internacional de Telecomunicaciones – Sector de Estandarización de la Telecomunicaciones (UIT-T), la capa óptica se puede descomponer en tres subcapas: la capa de canal óptico, la capa de multiplexación óptica y la capa de transmisión óptica. El objetivo de la capa de canal óptico es proveer la conectividad entre puntos finales de los canales ópticos, de forma que sea transparente a los clientes. La capa de multiplexación óptica está relacionada con el agregado a la red de las señales ópticas con distintas longitudes de onda. La capa de transmisión óptica se refiere a comunicación de las señales ópticas por distintos medios físicos, como por ejemplo las transmisiones monomodo o multimodo.

Las características mencionadas –reutilización de longitud de onda, transparencia de protocolos y alta fiabilidad- hacen de las redes *wavelength-routed* idóneas para las redes de área extensa. Estas redes exigen el tratamiento de tres cuestiones fundamentales: diseño, reconfiguración, corrección de fallos. El diseño de la capa óptica parte de la demanda de tráfico y es fundamental para aprovechar eficientemente los recursos de la red (fibras y transmisores/receptores) así como para proveer calidad de servicio a los usuarios. La reconfiguración de la capa óptica se deriva de las variaciones de las demandas de tráfico. El hecho de que grandes cantidades de tráfico se transporten por la capa óptica, obliga de forma crítica a disponer de un servicio de recuperación en caso de fallos. El objeto fundamental de estudio de este Proyecto será el diseño de la capa óptica.

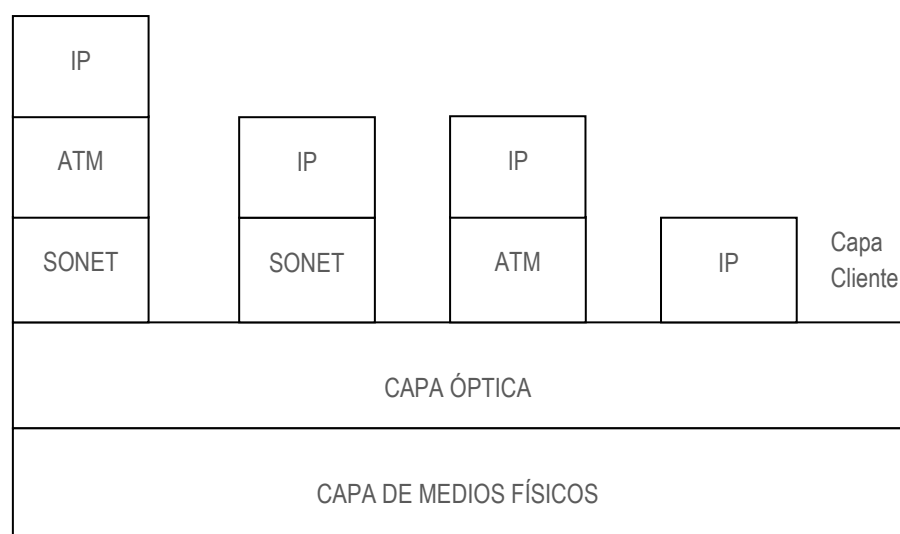


FIGURA 13 POSIBLES CAPAS EN UNA RED DE TRANSPORTE WDM



3.2 PARADIGMA DE LAS REDES WDM

Como se ha descrito en la sección 3.1, los usuarios finales en una red troncal basada en fibra se comunican entre ellos por medio de canales en el dominio óptico que hemos llamado *lightpaths*. Un *lightpath* atraviesa varios enlaces de fibra para interconectar dos nodos con alta carga de tráfico mediante una técnica de “conmutación de circuitos”, además el *lightpath* se establece con independencia de la proximidad física de los nodos. Cada nodo intermedio actúa de catalizador en el dominio óptico para el establecimiento y mantenimiento de los *lightpaths*. En la Figura 14 se muestra un ejemplo de red de *lightpaths*.

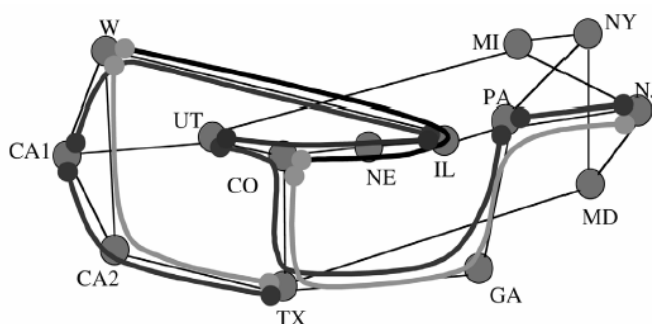


FIGURA 14 RED DE LIGHTPATHS SOBRE UNA RED WDM (LOS CÍRCULOS NEGROS INDICAN EXTREMOS DE UN LIGHTPATH)

En una red de N nodos, si cada nodo está equipado con $N-1$ transceptores⁶ y hay disponibles suficientes longitudes de onda en cada enlace de fibra, todos los nodos se pueden conectar mediante *lightpaths* sin requerir resolución de problema alguno. Las limitaciones de recursos conocidos en todos los sistemas de comunicaciones también se aplican a las redes ópticas: los transceptores son caros por lo que se limita su número en cada nodo, y el número de canales disponibles en cada enlace de fibra se limita a W (como ya hemos dicho pueden llegar a 100). En consecuencia, tan sólo unos pocos *lightpaths* se pueden establecer en la red, de lo que se deriva un problema de optimización.

De forma simplificada y bajo las condiciones citadas, el problema se describe así: un conjunto finito de *lightpaths* se tienen que establecer sobre una topología física de red y bajo una demanda determinada de tráfico. Para ello se tienen que determinar las rutas físicas que atravesarán los *lightpaths* y qué longitud de onda ocupará en cada una; todo con el objetivo de que se establezcan el máximo número *lightpaths* posibles. Desde una visión parcial de cada *lightpath* es deseable que se elijan las rutas con menor número de saltos, aunque este criterio se puede variar para posibilitar que se establezcan un número mayor de *lightpaths*. Los *lightpaths* que no se puedan establecer debido a las restricciones sobre las rutas y las longitudes de onda, se bloquean; por lo que el problema de optimización se centra en minimizar la probabilidad de bloqueo.

⁶ El término transceptor se refiere a los transmisores (láseres) y los receptores (fotodiodos)



A las condiciones del problema se añade la *restricción de continuidad de longitud de onda*, descrita en el apartado 3.1, de forma que si dos *lightpaths* comparten el mismo enlace de fibra no pueden usar la misma longitud de onda. Esta restricción desaparece en el caso de introducir nodos con capacidad de conversión de longitud de onda, de forma que se pueda conmutar la longitud de onda a lo largo de la ruta del *lightpath*. Estos nodos deben contener dispositivos llamados conversores de longitud de onda sintonizable (*Tunable Wavelength Converter*, TWC), que son capaces de modificar la longitud de onda de un canal WDM, sin conversión a electrónico.

Este problema particular se denomina RWA por sus siglas en inglés: *Routing and Wavelength Assignment* y engloba todo el planteamiento descrito.

Los diseñadores de la próxima generación de redes ópticas tienen que ser conscientes de las propiedades y limitaciones de fibras y dispositivos para que los protocolos y algoritmos diseñados aprovechen las redes WDM en todo su potencial. Con frecuencia los profesionales han trabajado bajo asunciones ideales o con una visión tradicional de las redes ópticas que arrojan resultados imprácticos o incluso irrealizables.

4 OBJETIVOS Y DISTRIBUCIÓN DEL DOCUMENTO

Como se ha visto en el presente capítulo de introducción, la tecnología óptica WDM articula un modelo de capas y requiere la convergencia de diversas áreas de conocimiento en el estudio clásico de redes además de conceptos propios e inherentes a las redes ópticas.

En este marco, el presente proyecto aborda el desarrollo de la herramienta *Dyn-MatPlanWDM* con fines educativos y orientada a la planificación de redes WDM wavelength-routing. Esta herramienta considera demanda dinámica de tráfico y provee un algoritmo de planificación que dinámicamente decide como encaminar el flujo teniendo en cuenta el estado de la red en ese momento, así como la topología física de dicha red. Además también incluye un generador para simular la llegada dinámica de flujos. *Dyn-MatPlanWDM* ha sido implementada sobre MATLAB de forma que pueda ser extendida sin gran esfuerzo, con sólo incluir los algoritmos y generadores en sus correspondientes librerías.

En el segundo capítulo se define el RWA y los tipos de problemas e instancias que incluye, siendo la principal el diseño de topologías virtuales. Además también se habla de los precedentes de la herramienta de planificación con demanda dinámica de tráfico.

El tercer capítulo se centra en la descripción de la herramienta desarrollada. Primero vemos la estructura del toolbox con todas sus funciones documentadas y después se incluye un manual de usuario de *Dyn-MatPlanWDM* para acercarnos al manejo de la herramienta.

En el capítulo 4 se presentan los resultados obtenidos sobre distintos escenarios, donde se ha querido incluir un amplio abanico de casuísticas que nos permita conocer la herramienta. Por último, y para enmarcar el Proyecto en las líneas de investigación citadas al comienzo se obtendrán las conclusiones de este trabajo y su idoneidad como aplicación docente, al mismo tiempo que se indicará su evolución futura.



CAPÍTULO 2: PLANIFICACIÓN DINÁMICA DE REDES WAVELENGTH-ROUTING

1 PLANIFICACIÓN ESTÁTICA VERSUS DINÁMICA

La demanda de tráfico sobre una red puede ser estática o dinámica lo que implica la existencia de dos tipos de estrategias de optimización cuando abordamos el proceso de planificación de una red WDM. Tendremos por una parte la planificación estática, también llamada off-line, y, por otro lado, la planificación dinámica, conocida como on-line.

En el caso de tráfico estático, los requerimientos de las conexiones a establecer se conocen a priori. Este tráfico se puede explicitar para cada par de nodos origen-destino, en una matriz de tráfico. Cada coordenada de la matriz puede elegirse a partir de una estimación del tráfico generado entre el par de nodos. El objetivo se focaliza en asignar rutas y longitudes de onda a toda la demanda de tráfico, minimizando una cierta medida de coste, y maximizando la demanda satisfecha. Las dos instancias del problema RWA para tráfico estático se encuadran dentro del problema SLE (*static lightpath establishment*). Se ha demostrado que el problema SLE es NP-completo⁷ por lo tanto se opta por los algoritmos de tiempo polinomial cuya solución se aproxima a la óptima. Así, podemos concluir que en la planificación estática partimos de una demanda de tráfico modelada como una matriz de tráfico $N \times N$, donde N es el número de nodos en la red. Cada coordenada (i, j) de la matriz de tráfico recoge el tráfico total esperado desde el nodo i al nodo j .

En caso de demanda de tráfico dinámica, las peticiones de conexión llegan y salen de la red de forma aleatoria. Los *lightpaths* establecidos lo hacen de forma temporal. La demanda de tráfico por tanto modela esta evolución temporal, y ya no puede ser únicamente una matriz estática. A diferencia del problema RWA estático, cualquier solución para un escenario de tráfico dinámico tiene que ser computacionalmente simple, ya que las peticiones y modificaciones tienen que ser procesadas de forma instantánea. Cuando se produce una nueva petición, se asigna una longitud de onda y ruta teniendo como objetivo maximizar el número de peticiones aceptadas (o, análogamente, minimizar el número de peticiones rechazadas).

Por tanto, para la planificación dinámica tenemos que la demanda de tráfico está modelada con mayor detalle, es decir, como una secuencia temporal de peticiones de flujos a más alto nivel. Por tanto, la planificación dinámica implica la toma de decisiones cada vez que se produce la llegada de una demanda de tráfico, y posiblemente también deberá decidir que hacer cuando un flujo existente termina. En el caso general, los nuevos flujos pueden ser acomodados en *lightpaths*

⁷ Computacionalmente intratable. Los algoritmos conocidos necesitan tiempo exponencial en el peor caso para encontrar una solución óptima.



existentes o implicar el establecimiento de un o más nuevos lightpaths. También, la llegada de nuevos flujos puede implicar la finalización o la redistribución de los lightpaths existentes. Aunque no es común, el fin de un flujo podría conllevar la redistribución o la finalización de flujos existentes y/o lightpaths.

Si comparamos ambas formas de llevar a cabo la planificación de la red, podemos decir que la planificación dinámica permite una evaluación más detallada de las características de la red, además de la observación de indicadores realistas que caracterizan la red tales como la probabilidad de bloqueo de las demandas de flujo.

2 HERRAMIENTAS CON PLANIFICACIÓN DINÁMICA

En este apartado veremos brevemente una revisión de algunas herramientas educativas relevantes que abordan la planificación dinámica. En primer lugar, hablaremos de “The GMPLS Lightwave Agile Switching Simulator” (GLASS) [1]. Se trata de una herramienta educativa construida como una evolución del proyecto “MERLiN” (Modeling Evaluation and Research of Lightwave Network), el cual fue suspendido en 2001 [2].

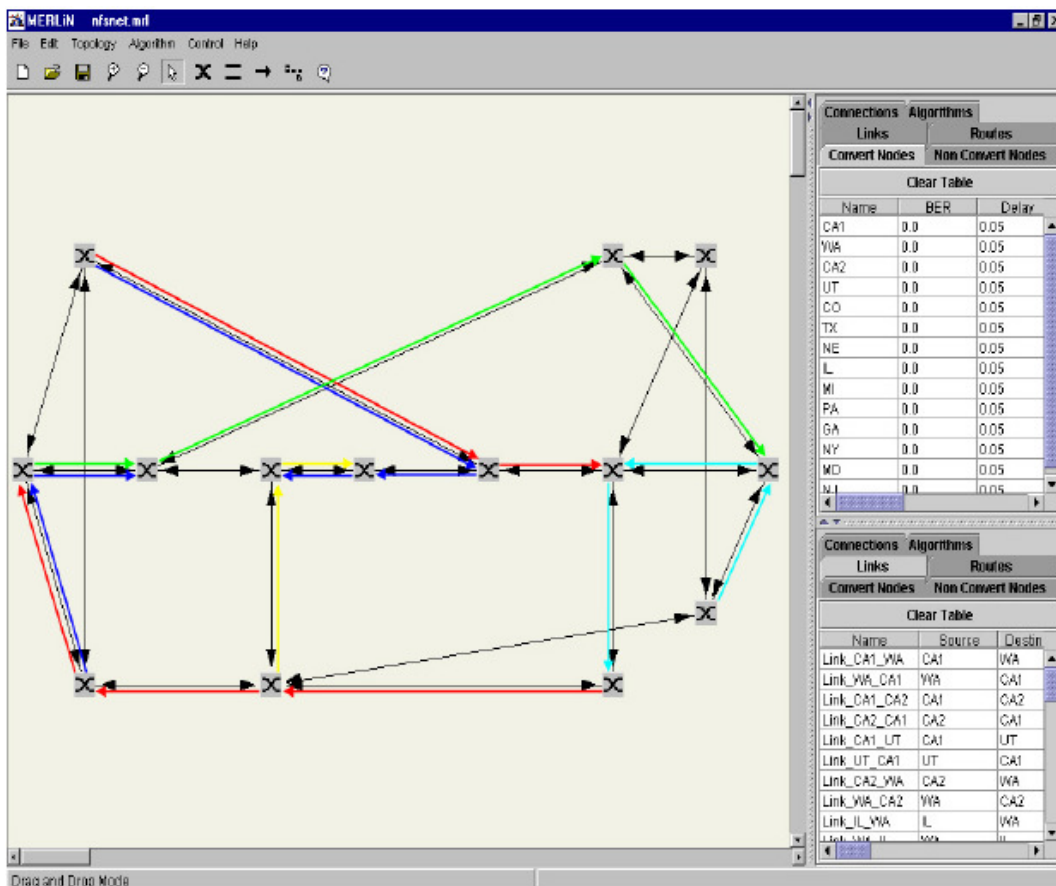


FIGURA 15 INTERFAZ GRÁFICA DE USUARIO DE MERLiN



La herramienta GLASS fue desarrollada en el Instituto Nacional de Estándares y Tecnología (NIST, National Institute of Standards and Technology). Está implementada en Java y diseñada sobre SSF/SSFNet [1] con un framework de simulación de eventos discretos. GLASS emplea el Data Modeling Language (DML) [1] para diseñar la topología and obtiene scripts para los diferentes escenarios de simulación sus componentes. El núcleo de GLASS está diseñado como un simulador por línea de comandos que lee su estructura de simulación de un script para posteriormente volcar la salida en archivos binarios que pueden ser procesados por lectores construidos según unas especificaciones. “The Towson State University” desarrolló una interfaz gráfica denominada “Topology and Simulation Creator” (TSC) con el fin de facilitar esta tarea.

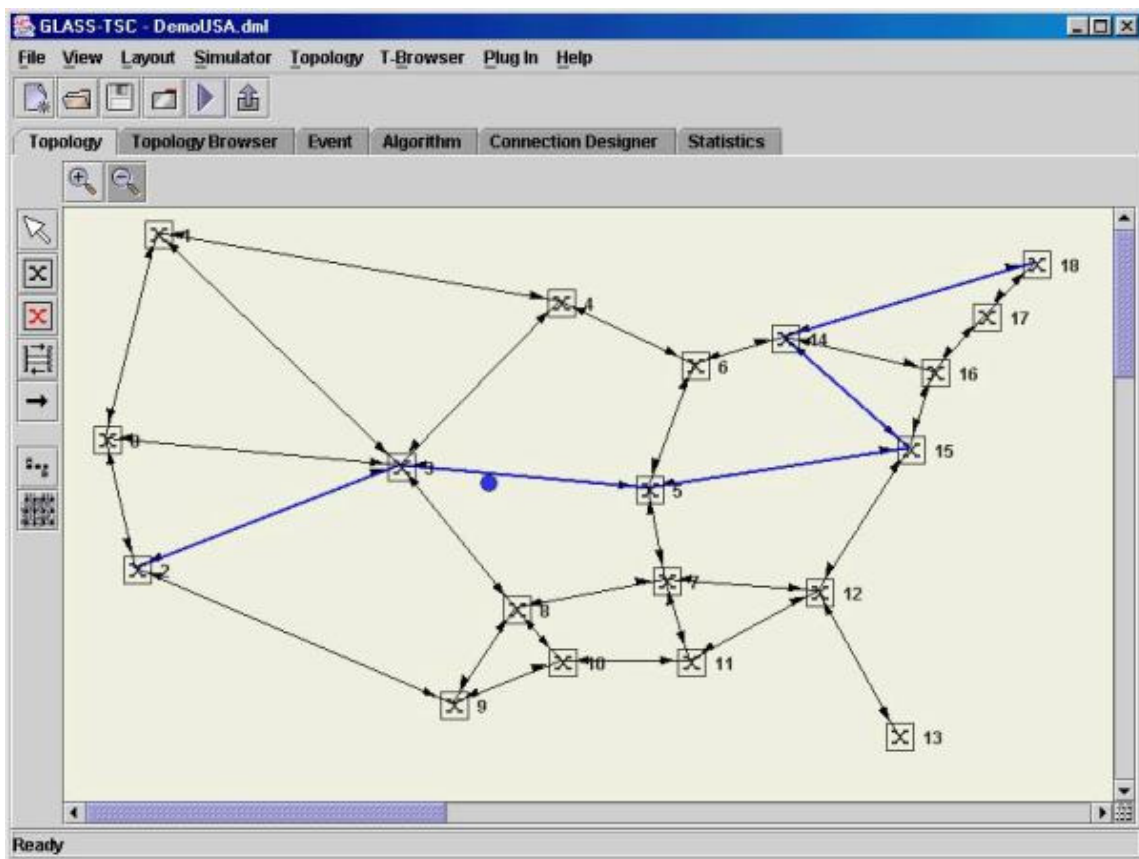


FIGURA 16 VENTANA PRINCIPAL DE LA APLICACIÓN TSC DE GLASS

GLASS es de uso gratuito y su última versión actualizada se publicó en Marzo de 2006. La propuesta de *Dyn-MatPlanWDM* ofrece algunas ventajas cuando la comparamos con la herramienta GLASS. En primer lugar, GLASS necesita el aprendizaje previo de una jerarquía compleja de clases de Java que nos permita la definición y la posterior prueba de nuevos algoritmos de optimización. En segundo lugar, el lenguaje Java no nos ofrece la funcionalidad matemática que MATLAB proporciona permitiendo una rápida implementación de algoritmos complejos.



Hablamos ahora de la herramienta *Optical WDM Network Simulator (OWns)* [3] que fue desarrollada en Washington University basándose en los conocidos simuladores de redes *ns*. Esta herramienta facilita el estudio de los algoritmos dinámicos RWA en redes WDM. Desafortunadamente, la actualización de *OWns* se detuvo en 2001. Si comparamos *Dyn-MatPlanWDM*, los beneficios de nuestra propuesta están basados de nuevo en la potente plataforma que MATLAB nos proporciona, que además nos ayuda a simplificar la implementación de algoritmos y a aclarar al estudiante los conceptos de la planificación.

Por supuesto, también existe un amplio rango de utilidades comerciales para el dimensionamiento de redes WDM bajo demanda dinámica de tráfico [4-6], tales como *MetroWAND* y *OptiSystem*.

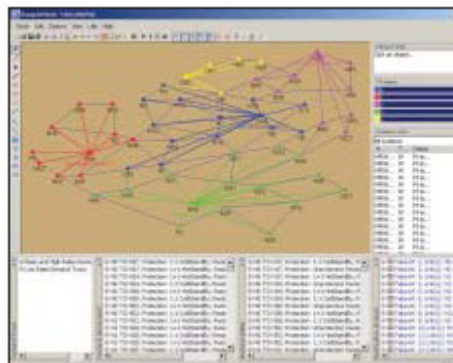


FIGURA 17 INTERFAZ GRÁFICA DE USUARIO DE METROWAND

No obstante, estas herramientas no están diseñadas con objetivos educacionales, ya que no es posible acceder a los detalles subyacentes de los algoritmos y tampoco pueden ser extendidas por investigadores o estudiantes. Por lo tanto, quedan fuera ámbito de esta investigación.

3 PROBLEMA RWA CON GROOMING

Teniendo en cuenta lo visto en el apartado 3.2 sabemos que en una red *Wavelength Routed WDM*, una conexión es llevada a cabo por un *lightpath* y para establecer una conexión entre un par fuente-destino, se debe encontrar una ruta continua de longitud de onda entre el par de nodos. Un algoritmo que se usa para seleccionar rutas y longitudes de onda para establecer *lightpaths* se conoce como algoritmo *RWA* (*Routing and Wavelength Assignment*). Muchos problemas en redes *Wavelength Routed WDM* tienen *RWA* como un subproblema.

Por tanto, es fundamental usar un buen algoritmo *RWA* para establecer *lightpaths* de manera eficiente. La asignación de longitudes de onda es la única característica en las redes *Wavelength Routed WDM* que las distingue de las redes convencionales. La restricción de continuidad de longitud de onda impone que la longitud de onda utilizada en todos los enlaces que el *lightpath* atraviesa tiene que ser la misma. Esta restricción es privativa de las redes WDM y



degrada considerablemente el rendimiento de la red, incrementando la probabilidad de bloqueo. La probabilidad de bloqueo se define como el porcentaje de peticiones rechazadas.

Un algoritmo *RWA* tiene dos componentes, selección de ruta y selección de longitud de onda. La ruta es elegida según criterios como el número de saltos de la ruta, el retraso total de la ruta o la congestión, intentando minimizarlos. La longitud de onda se elige a partir de criterios como el porcentaje de uso de la longitud de onda en la red completa. Estos dos métodos pueden ser usados en cualquier orden, uno detrás de otro o conjuntamente, pero el orden de selección afecta al rendimiento de los mismos.

En un entorno de tráfico dinámico, donde las peticiones de conexiones entran y salen de la red de forma aleatoria, puede ocurrir la siguiente situación: una petición de conexión para el par de nodos $\langle s, d \rangle$ encuentra una ruta disponible, pero se bloquea al no encontrar una longitud de onda común a toda ella. Se ilustra en la Fig. 2.1. La figura muestra dos *lightpaths* p_1 y p_2 con longitudes de onda w_0 y w_1 , respectivamente. Se asume que hay dos longitudes de onda por fibra. Supongamos que llega una petición al nodo 0 para conectarse al nodo 2 y que, por lo tanto, existe una ruta disponible, pero para esta ruta en el primer enlace (0-1) está libre w_1 y en el segundo enlace (1-2) está libre w_0 . La petición es rechazada por no existir continuidad de longitud de onda.

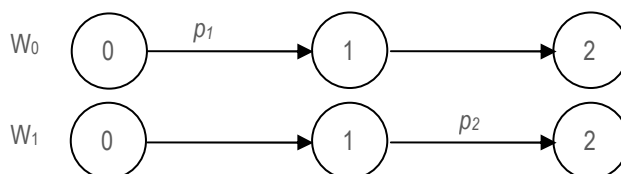


FIGURA 18 RESTRICCIÓN DE LONGITUD DE ONDA Y BLOQUEO

Si el nodo 1 estuviera equipado con capacidad de conversión de longitud de onda, la petición podría ser aceptada eligiendo en el primer salto la longitud de onda w_0 y en el segundo w_1 . En el caso de que todos los nodos de la red tuvieran esta capacidad, el funcionamiento sería el mismo que una red de conmutación de circuitos convencional.

En el ejemplo anterior no hemos tenido en cuenta que los *lightpaths* proporcionan un ancho de banda grande, que no suele ser aprovechado por los usuarios individuales. Por esta razón, deben de ser vistos como canales de transporte en la red *backbone*, donde se multiplexa el tráfico de diversas aplicaciones de la red de acceso. En cierto sentido, esta es la razón que justifica la inclusión del problema de enrutamiento de tráfico en el diseño de la topología virtual, ya que el tráfico de capa superior debe de ser encaminado por la topología virtual propuesta, y de esta forma los *lightpaths* transportan el tráfico obtenido por agregación de los flujos de baja de velocidad. El patrón de multiplexación de tráfico sobre los *lightpaths* afecta directamente a la eficiencia del reenvío óptico que se produce en los *Wavelength Routers* (WR), ya que toda la información que contiene un *lightpath* se somete a una conversión electro-óptica y a un posterior enrutamiento electrónico en el caso de que el flujo de tráfico finalice en el nodo intermedio. Este efecto se refleja en el coste de los componentes de red (número y capacidades). Para poder computar los efectos del



diseño, se especifican diversas topologías físicas, cuyos resultados se comparan tomando como variables relativas al coste, el número de transceptores por nodo, número de longitudes de onda, número medio de salto para los *lightpaths*, o la caracterización de los patrones de tráfico. Por todo ello el *traffic grooming* se está convirtiendo en una de las áreas dentro del diseño óptico más estudiadas en las líneas de investigación actuales.

3.1 PLANIFICACIÓN DE FLUJOS DINÁMICOS

Los algoritmos de planificación On-line reaccionan ante las peticiones de conexión a más alto nivel tanto llegadas como eventos fin de flujo. El problema total implica decidir sobre la evolución de la topología virtual y el encaminamiento de los flujos. Los heurísticos propuestos para este tipo de problemas están normalmente compuestos de otros algoritmos heurísticos que dividen el problema en otros subproblemas. Por ejemplo:

- (a) Algoritmos Routing Lightpaths: para una petición de lightpath entre un par de nodos origen y destino, y partiendo de una topología física, el algoritmo obtiene el camino en saltos físicos para el lightpath. Por ejemplo, las referencias: [7], [8], [9] y [10].
- (b) Algoritmos Wavelength Assignment (WA): para una petición de lightpath entre un par de nodos origen y destino partiendo de una topología física y de la ruta del lightpath, el algoritmo decide sobre la asignación de la longitud de onda en cada salto con y sin restricción de continuidad de longitud de onda. Por ejemplo, las referencias: [7], [8], [11] y [12].
- (c) Algoritmos Routing and Wavelength Assignment (RWA): para una petición de lightpath entre un par de nodos origen y destino partiendo de una topología física, el algoritmo calcula una ruta para el lightpath incluyendo la asignación de la longitud de onda en cada salto, con y sin restricción de continuidad de longitud de onda. Por ejemplo, las referencias: [7], [12], [13], [14] y [15].
- (d) Algoritmos Rerouting and Wavelength Assignment (RRWA): este problema es parecido al anterior RWA. La única diferencia se encuentra en la posibilidad de reencaminar lightpaths existentes en la red, para acomodar la nueva petición de lightpath. Este tipo de algoritmos están normalmente diseñados para redes WR con restricción de continuidad de longitud de onda, y buscan una solución que minimice el coste de rerouting. Por ejemplo, las referencias: [7] y [16].
- (e) Algoritmos Flow routing: para una petición de flujo entre un par de nodos origen y destino partiendo de la topología virtual existente, el algoritmo obtiene un camino de saltos virtuales para el flujo. Por ejemplo, las referencias: [17] y [18].

A continuación se incluye una relación de algoritmos encontrados de los distintos tipos que hemos explicado anteriormente incluyendo su función objetivo, así como si tienen en cuenta la conversión de longitud de onda.


TABLA 1 ALGORITMOS DE PLANIFICACIÓN DINÁMICOS PARA EL PROBLEMA RWA O SUS SUBPROBLEMAS

ALGORITMO	TIPO	CONVERSIÓN DE WAVELENGTH	FUNCIÓN OBJETIVO	REFERENCIAS
Fixed Routing (FR)	Routing	-	-Mínimo número de saltos -Mínimo tiempo de establecimiento de conexión	[7] [8]
Fixed Alternate Routing (FAR)	Routing	-	Mínimo número de saltos	[7] [8]
Exhaust Routing (ER)	Routing	-	- Mínimo número de saltos -Mínima Probabilidad de Bloqueo	[7]
Least Congested Path Routing (LCPR)	Routing	-	Mínima congestión de la ruta	[7] [8]
Weighted Least-Congestion Routing (WLCR)	Routing	-	Mínima congestión de la ruta "Weighted"	[9] [10]
Random Order (RN)	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[7] [8]
First-Fit (FF) / Fixed Order (FX)	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[7] [8]
Least-Used (LU) / SPREAD	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[7] [8]
Most-Used (MU) / PACK	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[7] [8]
Max-Sum (MΣ) / Total Capacity Loss (TCL)	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[8]
Relative Capacity Loss (RCL)	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[8] [11]
Distributed Relative Capacity Loss (DRCL)	Wavelength Assignment	Sin conversión	Mínima Probabilidad de Bloqueo	[8]
First-Fit Wavelength First (FFW)	Wavelength Assignment	Con conversión	Mínima Probabilidad de Bloqueo	[12]



LEast Converter First (LEC)	Wavelength Assignment	Con conversión	Mínimo número de conversores usados	[12]
Least Conversion Cost First (LCC)	Wavelength Assignment	Con conversión	Mínima función de coste de conversión	[12]
Joint Wavelength-Route Selection (JWRS)	Routing and Wavelength Assignment (RWA)	Sin conversión	Mínima función de coste basada en el estado de uso de las longitudes de onda, número de saltos y congestión	[7]
Hop Count and Available Wavelengths (HAW)	Routing and Wavelength Assignment (RWA)	Sin conversión	Mínima Probabilidad de Bloqueo	[13]
Hop Count and Available Wavelengths (HAW)	Routing and Wavelength Assignment (RWA)	Sin conversión	Mínima Probabilidad de Bloqueo	[13]
Total Wavelengths and Available Wavelengths (TAW)	Routing and Wavelength Assignment (RWA)	Sin conversión	Mínima Probabilidad de Bloqueo	[12] [13]
Hop Count and Total Wavelengths and Available Wavelengths (HTAW)	Routing and Wavelength Assignment (RWA)	Sin conversión	Mínima Probabilidad de Bloqueo	[13]
Total Routing and Wavelength Assignment (TRWA)	Routing and Wavelength Assignment (RWA)	Con conversión	- Mínima Probabilidad de Bloqueo - Mínimo número de conversores usados	[12] [14]
Wavelength- and Converter- Aware Algorithm (WCA)	Routing and Wavelength Assignment (RWA)	Con conversión	- Mínima Probabilidad de Bloqueo - Mínimo número de conversores usados	[12]
-	Routing and Wavelength Assignment (RWA)	Sin conversión	- Máximo Tráfico encaminado en un salto óptico - Mínima Probabilidad de Bloqueo	[15]



Minimum Blocking Probability First (MBPF)	Converter Placement	Con conversión	Mínima Probabilidad de Bloqueo	[9] [10]
Weighted Maximum Segment Length (WMSL)	Converter Placement	Con conversión	Mínima suma de la máxima longitud del segmento	[10]
Auxiliary Graph (AG)	Wavelength ReRouting Algorithm	Sin conversión	Mínimo número de lightpaths reacomodados “Weighted”	[7] [16]
Minimum Weighted-Count Path-Preserving Graph (MWPG)	Wavelength ReRouting Algorithm	Sin conversión	Mínimo número de lightpaths reacomodados “Weighted”	[7]



CAPÍTULO 3: DYN-MATPLANWDM

1 ESTRUCTURA DEL TOOLBOX

La herramienta Dyn-MatPlanWDM ha sido implementada como un toolbox de MATLAB. Todas las funciones del toolbox están documentadas y están abiertas al uso. La siguiente figura nos ayuda a describir la estructura del toolbox.

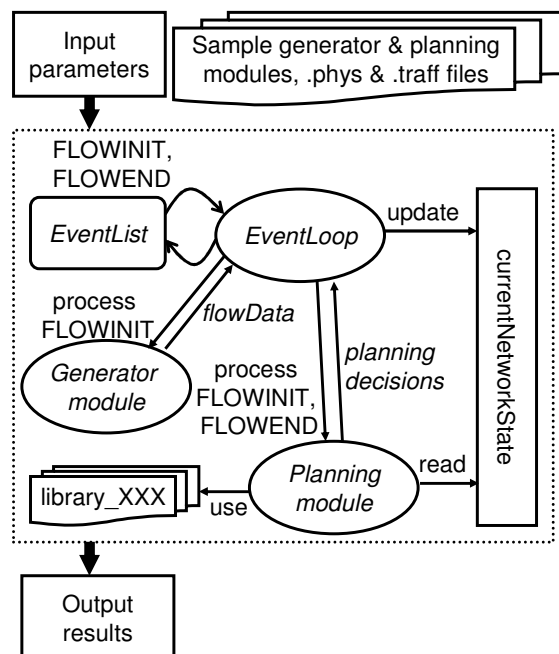


FIGURA 19 DIAGRAMA DEL TOOLBOX DE DYN-MATPLANWDM

1.1 PARÁMETROS DE ENTRADA

Los parámetros de entrada para la simulación son especificados en tres archivos: (1) un archivo de texto que describe la topología física, recogiendo la información por nodo (número de transmisores E/O, receptores O/E, conversores de longitud de onda (TWC), el número de longitudes de onda por enlace, y la capacidad de los lightpaths en Gbps), (2) un archivo de texto que contenga la matriz de tráfico que tomaremos para la generación de eventos en la red, (3) un archivo de MATLAB que implementa el módulo generador, y (4) un archivo de MATLAB que implementa



el módulo de planificación. Una vez definidos los parámetros de entrada, el usuario puede pasarlos a los dos módulos mencionados anteriormente para la inicialización.

1.2 EVENT LOOP

El módulo *EventLoop* es el motor del simulador. Éste controla una lista de eventos y además es el único módulo capaz de añadir y eliminar eventos de la lista. En dicha lista nos encontraremos sólo dos tipos de eventos: (a) nueva llegada de flujo (FLOWINIT), (b) y final de un flujo existente (FLOWEND). Estos dos eventos son procesados como se describe a continuación:

- (a) FLOWINIT: Cuando se procesa un evento de este tipo, el módulo *Event Loop* primero invoca al módulo Generador. Este último debe devolver la información acerca del flujo que llega (nodos origen y destino del flujo, tasa media del flujo en Gbps, duración del flujo en segundos) y además el instante de llegada del próximo flujo (*timeToNextFlow*). El módulo *EventLoop* empleará la información del tiempo de llegada del próximo flujo con el fin de añadirlo como un evento FLOWINIT a la lista de eventos. De esta forma se programa la llegada de un nuevo flujo cada vez que se produce un evento de este tipo. Una vez hecho esto, *EventLoop* llama al módulo de Planificación pasándole la información del flujo como parámetro de entrada, así como la estructura *currentNetState*. El algoritmo de planificación decide si el nuevo flujo será cursado o no.

Por último, la información que el algoritmo devuelve al módulo *EventModule* es la estructura *currentNetworkState* actualizada y una variable *infoWhatHappenedWithTheFlow* que nos informa de lo ocurrido con el flujo. Una vez el módulo *Event Loop* recupera el control, comprueba si el flujo ha sido cursado y si es así programa el final del flujo añadiéndolo a la lista de eventos como FLOWEND.

- (b) FLOWEND: A la hora de procesar un evento de este tipo, el módulo *Event Loop* llama al módulo de Planificación. Éste finaliza el flujo actualizando la estructura *currentNetState* y comprueba si algún *lightpath* queda libre, cuando es así lo elimina liberando los recursos.

En cuanto a la estructura *currentNetState* cabe decir que se trata de un conjunto de variables que describen el estado de la red en un instante dado. Como hemos visto, el módulo de Planificación actualiza esta estructura en cada ciclo. Este módulo emplea la estructura *currentNetState* para obtener la información que necesita acerca de los *lightpath* existentes con sus correspondientes rutas, los flujos cursados y sus rutas, las longitudes de onda disponibles en cada enlace y el número de recursos disponibles en un nodo (transmisores, receptores y TWCs).

1.3 ESTADÍSTICOS CALCULADOS

La herramienta automatiza el cálculo de un gran número de estadísticos, que pueden ser guardados en un archivo de texto con el fin de poder procesarlos. DynMATPlanWDM traza la



evolución a lo largo del tiempo de los siguientes indicadores, proporcionando el valor mínimo, el valor máximo y el promedio temporal de todos los estadísticos que a continuación indicaremos:

- Estadísticos asociados al coste de la red: número de transmisores, receptores y TWCs empleados por nodo, número de longitudes de onda usadas por enlace, cantidad de tráfico procesado electrónicamente en cada nodo (Gbps).
- Estadísticos asociados al rendimiento percibido por el tráfico cursado: número medio de saltos físicos de los lightpaths sobre los enlaces, número medio de saltos virtuales de los flujos sobre los lightpaths, tráfico encaminado en un único salto virtual, la media del retardo de propagación experimentado por un mensaje en la red, congestión de la red (definida como la utilización del tráfico en un lightpath frente a la máxima utilización del tráfico).

Como estadísticos asociados a los cálculos de bloqueo: obtiene la probabilidad de bloqueo, porcentaje de tráfico bloqueado.

1.4 ESTRUCTURA DE DIRECTORIOS

En este apartado veremos la estructura relativa a los directorios que componen la herramienta Dyn-MatPlanWDM y sus funciones. En primer lugar incluimos un esquema con los directorios que componen la herramienta.

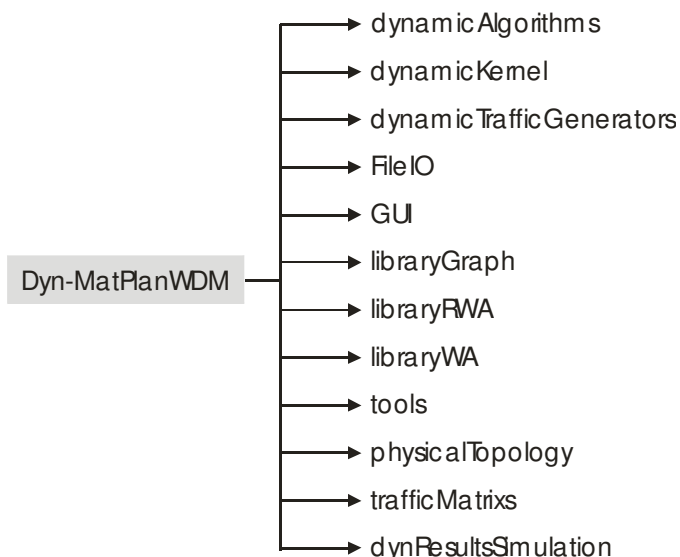


FIGURA 20 ESTRUCTURA DE DIRECTORIOS DE DYN-MATPLANWDM



A continuación se incluye una tabla que aporta una relación detallada de las principales funciones que constituyen la herramienta incluyendo información sobre su funcionalidad y su signatura.

TABLA 2 RELACIÓN DE LAS PRINCIPALES FUNCIONES

NOMBRE	SIGNATURA	FUNCIONALIDAD	DIRECTORIO
dynamicAlgorithm	[netState,infoWhatHappenedWithTheFlow] = dynamicAlgorithm (eventType,netState, phys,simTime , flowAverageRate, serialNumOfThisFlow, pairIngressEgress, algorithmParameters, flowDuration)	Procesa los eventos tanto de fin como de llegada de un flujo y posteriormente actualiza las variables de estado de la red.	dynamicAlgorithms
dynamicKernel	[dynStatistics] = dynamicKernel (maxSimulationTime, transitoryTime, physicalTopologyFile,generator , generatorParameters, algorithmMethod, algorithmParameters, batchSimulationFlag, handles)	Controla la lista de eventos pendientes, llama al generador y al algoritmo de planificación.	dynamicKernel
generateFlow	[timeToNextEvent , flowDuration, flowAverageRate,pairIngressEgress] = generateFlow (generatorParameters, simTime, netState, phys)	Devuelve la información del nuevo flujo y el tiempo de llegada del siguiente flujo.	dynamicTrafficGenerators
readPhysFile	[phys] = readPhysFile (fullpathname)	Lee un archivo .phys situado en el path que se le pasa como entrada y devuelve la topología física del archivo.	FileIO
writePhysFile	writePhysFile(phys , fullpathname)	Crea un archivo .phys con la topología física que se la pasa como entrada y en el path que se le indica.	FileIO
readTraffFile	[traff_trafficMatrix] = readTraffFile(fullpathname)	Lee un archivo .traff situado en el path que se le pasa como entrada y devuelve la matriz de	FileIO



		tráfico del archivo.	
writeTraffFile	writeTraffFile(phys , fullpathname)	Crea un archivo .traff con la matriz de tráfico que se la pasa como entrada y en el path que se le indica.	FileIO
libraryGraph_capacityShortestPath	[sequenceOfSPFLinkIds,sequenceOfSPFNodeIds , totalCost] = libraryGraph_capacityShortestPath (linkTable, ingressNode, egressNode, minimumCapacityInEachLink)	Calcula el camino de ingressNode a egressNode con el peso total (suma de pesos de enlaces) mínimo. Solo sirven los enlaces cuya capacidad sea mayor a la marcada en minimumCapacityInEachLink.	libraryGraph
libraryGraph_kshortestPath	[cellOfKPaths] = libraryGraph_kshortestPath (k,linkTable,ingressNode,egressNode)	Calcula k caminos distintos shortest path segun los pesos de los enlaces.	libraryGraph
libraryGraph_shortestPath	[sequenceOfSPFLinkIds,sequenceOfSPFNodeIds , totalCost] = libraryGraph_shortestPath (linkTable, ingressNode, egressNode)	Calcula el camino de ingressNode a egressNode con el peso total (suma de pesos de enlaces) mínimo. En caso de no existir camino, devuelve matrices vacias	libraryGraph
libraryRWA_kShortestRWA	[exitFlag sequenceOfSPFLinkIds sequenceOfSPFNodeIds sequenceOfWavelengths newLightpathEntry] = libraryRWA_kShortestRWA(k, ingressNode, egressNode, phys, netState)	Calcula un camino y asigna la longitud de onda (RWA) para un par de nodos intentando establecer un lightpath entre ellos; primero elige las 'k' rutas más cortas, y luego prueba a asignar longitudes de onda a la ruta más corta con mínimo número de conversores (TWCs). Si no es posible lo intenta con la siguiente ruta más corta.	libraryRWA



libraryWA_graphTransformation	[ingressTransfId egressTransfId transfNodeTable transfLinkTable transfTWCTable] = libraryWA_graphTransformation (sequenceOfLinkIDs , sequenceOfNodeIDs, phys, netState)	Calcula un grafo auxiliar que permite realizar la asignación de longitud de onda para un camino dado y con el mínimo número de conversores (TWCs)	libraryWA
libraryWA_minConv	[exitFlag sequenceOfWavelengths newLightpathEntry] = libraryWA_minConv(sequenceOfLinkIDs, sequenceOfNodeIDs , phys, netState)	Realiza la asignación de la longitud de onda para un camino dado con el mínimo número de conversores (TWCs).	libraryWA
calculateDynamicNetMetrics	[dynStatisticsOutput] = calculateDynamicNetMetrics (dynStat)	Obtiene el promedio temporal de los estadísticos de la estructura dynStat.	Tools
acumulateDynamicNetMetrics	[dynStatisticsOutput] = acumulateDynamicNetMetrics (dynStatistics , simTime, lastSimTime, netState, phys, verbose)	Realiza la suma de los estadísticos multiplicada por el tiempo del ciclo para posteriormente poder obtener el promedio temporal.	Tools
plotTopology	plotTopology (nodesPlaceMatrix, edgeTable, type, topologyTitle, showNrEdgesPerPair)	Dibuja una topología de nodos y enlaces partiendo de la información de entrada.	Tools
plotLightpath	plotLightpath (nodesPlaceMatrix, edgeTable, lightpathID, lightpathRoutingMatrix, lightpathTitle)	Dibuja un lightpath sobre una topología de nodos y enlaces que viene dada.	Tools
plotTrafficFlow	plotTrafficFlow (nodesPlaceMatrix, edgeTable, flowID, flowTable, flowRoutingMatrix, topologyTitle)	Dibuja un flujo sobre la topología dada.	Tools
addEventToListOfPendingEvents	listOfPendingEvents=addEventToListOfPendingEvents (eventType , eventTime , serialNumber,listOfPendingEvents)	Añade un evento a la lista de eventos pendientes con la información que se le pasa.	Tools



nextPendingEvent	[eventType , eventTime , serialNumber] = nextPendingEvent(listOfPendingEvents)	Devuelve el siguiente evento pendiente.	Tools
removeTopPendingEvent	[listOfPendingEvents,serialNumber]=removeTopPendingEvent(listOfPendingEvents)	Borra el primer evento pendiente en el tiempo y devuelve la información de este evento.	Tools

1.5 ESTRUCTURA DE EVENTLOOP

En la siguiente figura vemos la estructura general del módulo *EventLoop* que después vamos a describir con más detalle.

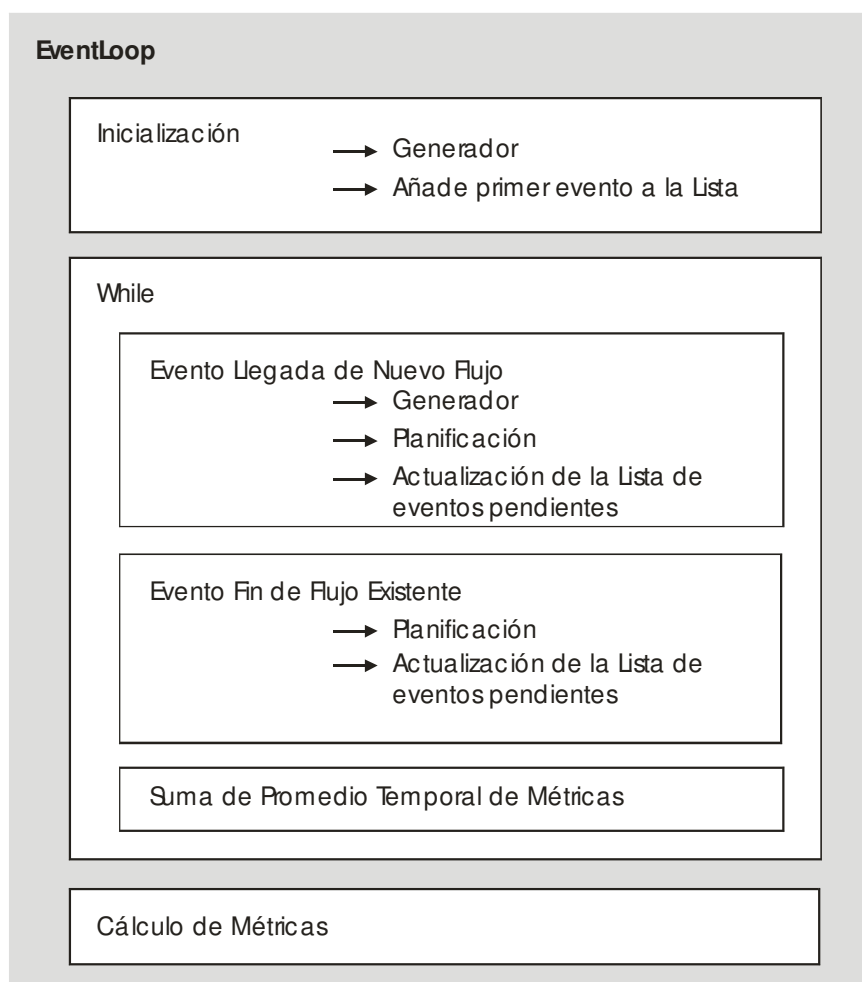


FIGURA 21 ESTRUCTURA DEL MÓDULO EVENTLOOP

En primer lugar, en la parte de inicialización el módulo realiza una llamada al generador, el cual creará en ese momento una lista de eventos interna en la que almacena para cada par de nodos



el tiempo de llegada del siguiente flujo entre ellos. En este punto añade el primer evento de llegada de nuevo flujo a la lista de eventos pendientes.

Después de la inicialización, comienza el bucle que gestiona la llegada de eventos. Si se trata de un evento de llegada de nuevo flujo, realiza una llamada al generador para obtener la información de este flujo así como el tiempo de llegada al siguiente flujo para añadirlo a la lista de eventos pendientes. Con estos datos, llama al módulo de Planificación para intentar cursar el flujo de las distintas maneras que mencionamos en el punto 1.2 y posteriormente este módulo devuelve el estado de la red actualizado. Para terminar el evento es eliminado de la lista. Si ha sido cursado se programa su evento fin de flujo y, en cualquier caso, siempre se añade una llegada de nuevo flujo a la lista de eventos.

Si se trata de un evento fin de flujo, el bucle pasa el control al módulo Planificación para que elimine el flujo de la red y actualice la estructura de estado de la red. Dentro del mismo bucle, cada vez que se produce un evento se obtienen las métricas y se realiza la suma acumulada de cada una de ellas por el tiempo del ciclo para el posterior cálculo del promedio temporal. Por último, la lista de eventos es actualizada eliminando el evento fin de flujo.

Una vez cumplido el tiempo máximo de simulación, el módulo *EventLoop* sale del bucle y termina realizando el cálculo del promedio temporal de todas las métricas.

1.6 ESTRUCTURA DEL CÁLCULO DE ESTADÍSTICOS

Las métricas son calculadas como promedios temporales siguiendo la siguiente expresión:

$$\bar{x} = \frac{\sum_i x_i \cdot (t_i - t_{i-1})}{T}$$

Donde : \bar{x} es el promedio temporal de la métrica,

x_i es el valor de la métrica en el tiempo de simulación t_i ,

t_i es el instante de simulación i ,

T es el tiempo total de simulación.

Por tanto, cada vez que se produce la llegada de un evento las métricas son recalculadas (x_i) por la función *calculateNetMetrics* para ser multiplicadas por la duración del ciclo ($t_i - t_{i-1}$) y posteriormente sumadas al valor acumulado. Este paso es realizado por la función *accumulateDynamicNetMetrics*.

Además del promedio temporal de cada métrica, la función *accumulateDynamicNetMetrics* guarda los valores máximos y mínimos que toma cada métrica en un instante determinado.



Cuando la simulación termina cada una de las sumas acumuladas se divide por el tiempo de simulación total (T) en el que no se tiene en cuenta el tiempo de transitorio. Este punto es llevado a cabo por la función *calculateDynamicNetMetrics*.

2 MANUAL DE USUARIO

2.1 INSTALACIÓN Y ARRANQUE DE LA APLICACIÓN

Para la instalación de la aplicación únicamente será necesario descomprimir el archivo de extensión *.rar*. No hará falta guardar la carpeta en un directorio especial, será suficiente con elegir en la pestaña *Current Directory* de MATLAB el path de la carpeta *Dyn-MatPlanWDM*.

En cuanto a la ejecución de la aplicación, basta con ejecutar el comando *startup* en la línea de comandos de MATLAB y se abrirá la ventana principal de *Dyn-MatPlanWDM*.

2.2 VENTANA PRINCIPAL

Una vez hemos ejecutado *startup* aparece la ventana principal de la aplicación como la que vemos a continuación:

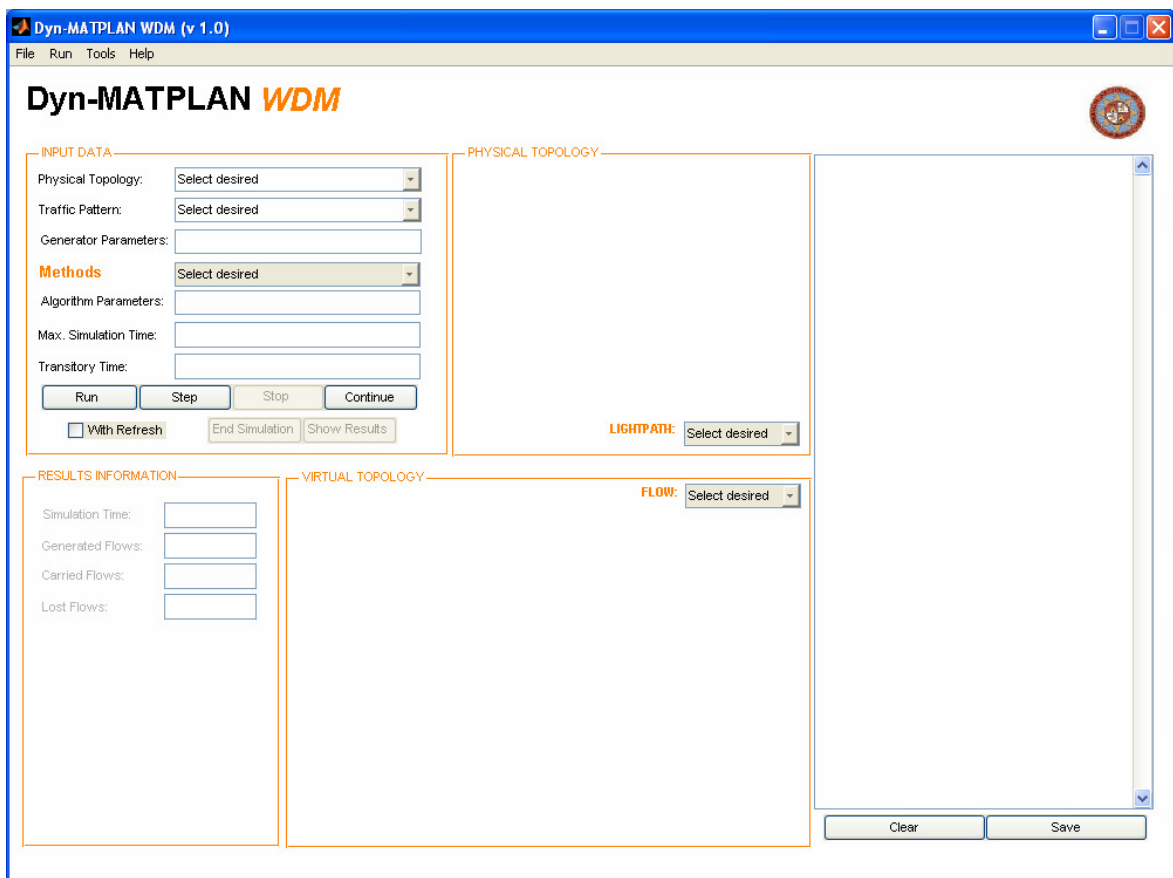


FIGURA 22 VENTANA PRINCIPAL DE DYN-MATPLANWDM



Observando la figura anterior comprobamos que la ventana está constituida por cinco partes principalmente. Seguidamente vemos que nos ofrece cada una de ellas.

El panel INPUT DATA nos permite elegir los datos de entrada de la simulación a realizar: la topología física, el generador, los parámetros del generador, el algoritmo de planificación, los parámetros del algoritmo, el tiempo máximo de simulación y tiempo del transitorio.

FIGURA 23 PANEL INPUT DATA

El menú desplegable *Physical Topology* mostrará las topologías físicas que se encuentran en el directorio *physicalTopology* entre las que tendremos que elegir una. El siguiente menú *Traffic Pattern* nos muestra los generadores disponibles que son los que están en la carpeta *dynamicGenerators*. Lo mismo sucede con el menú *Methods* pero en este caso los algoritmos serán los del directorio *dynamicAlgorithms*.

En cuanto al edit *Generator Parameters*, habrá que tener en cuenta el formato que espere recibir el generador seleccionado. Por ejemplo, en la figura anterior vemos que para el caso del generador *generateFlow* el formato de los parámetros es: tasa media del flujo, duración media del flujo, indicador de truncamiento y nombre del archivo de tráfico que va a emplear el generador; separados entre ellos por espacios. En la misma situación nos encontramos con el edit *Algorithm Parameters*, que también tendrá un formato que debe ser acorde con el esperado por el algoritmo de planificación. En el ejemplo de la figura vemos que el algoritmo *dynamicAlgorithm* tiene como parámetro el valor de k que es el número de caminos más cortos calculados durante la planificación. Los dos últimos edits nos permiten elegir el máximo tiempo de simulación y el tiempo transitorio.

En la parte inferior del panel tenemos los botones que controlan la simulación. El usuario puede comenzar la simulación haciendo click sobre el botón *Run*. Si el checkbox “with refresh”



está activado, la información del panel RESULTS INFORMATION se actualiza durante la simulación.

RESULTS INFORMATION	
Simulation Time:	9738.7538
Generated Flows:	370
Carried Flows:	370
Lost Flows:	0

FIGURA 24 PANEL RESULTS INFORMATION

Una vez iniciada la simulación, ésta se puede detener momentáneamente pulsando el botón *Stop* y puede finalizarse por completo con el botón *End Simulation*. Cuando se ha pulsado el botón *Stop*, el botón *Step* se activa dando la posibilidad de consumir un evento cada vez que se hace click en él.

INPUT DATA	
Physical Topology:	physFile_L8_TW0.phys
Traffic Pattern:	generateFlow.m
Generator Parameters:	0.1 43200 1 nsfnet10.traff
Methods	dynamicAlgorithm.m
Algorithm Parameters:	2
Max. Simulation Time:	5000000
Transitory Time:	50000
<input type="button" value="Run"/> <input type="button" value="Step"/> <input type="button" value="Stop"/> <input type="button" value="Continue"/>	
<input checked="" type="checkbox"/> With Refresh <input type="button" value="End Simulation"/> <input type="button" value="Show Results"/>	

FIGURA 25 PANEL INPUT DATA CUANDO LA SIMULACIÓN ES DETENIDA

Para continuar con la simulación es suficiente con pulsar el botón *Continue* y si queremos ver los valores de los estadísticos hasta ese momento podemos hacer click en *ShowResults* que mostrará los resultados el edit situado a la derecha.



```

#####
Simulation date: 23/11/2007 - 13:42
#####

-----
INPUT DATA
-----
Physical Topology : physFile_L8_T000.phys
Traffic Pattern : generateFlow.m
Generator Parameters : 0.1 43200 1 nsfnct10.traff
Method : dynamicAlgorithm.m
Algorithm Parameters : 2
Max. Simulation Time : 5000000
Transitory Time : 50000

-----
RESULTS
-----
Carried Flows: 370
Lost Flows: 0
Blocking Probability: 0
Percent of Blocked Traffic: 0
Average Used Transmitters Per Node: 3.5575 1.9696
1.8693 2.9021 2.9208 0.95769 1.5277 1.6327
1.8837 1.5251 1.8983 2.7281 0.99951
1.9326
Average Used Receivers Per Node: 2.8232 1.7694
0.95992 1.883 0.87245 0.75814 0.73782
0.99951 1.4595 1.8374 0.82787 3.6221
5.8924 3.8617
Average Used Wavelengths Per Fiber Link: 0.87245 1.7181
0.96698 0.99572 0.99322 1.8488 0.93844
1.7694 2.6823 0.99572 0.87245 4.8236 2.9237
0.9799 0.73782 1.7694 0.73782 0
3.8617 0.98284 0.54482 0.88905 0 3.1445
1.8886 1.8374 0.79528 1.9616 1.5251 0
0.90018 2.7911 3.9308 1.9147 1.7281 0
0.99951 0 0 0.98215 0.95048 0
Average Used Physical Hops a Lightpath: 2.057
Average Link Traffic: 0.33002 1.2838 0.36587
0.95727 0.25128 1.3488 0.76328 1.4504
0.90737 0.95727 0.33002 4.4955 1.6745
1.0671 0.22256 1.4504 0.22256 0 3.1751
0.43827 0.057627 0.68315 0 0.64768
2.1091 0.96824 0.093804 1.1461 0.36283 0
0.88719 2.5527 3.7777 2.0289 0.92461 0
1.4259 0 0 1.3102 0.76832 0
Average Total offered traffic to the network: 17.883
    
```

FIGURA 26 EDIT RESULTADOS

Para borrar los datos mostrados en el edit de la figura anterior basta con hacer click sobre el botón *Clear*. Y si lo que queremos es guardar los datos que aparecen en el edit podemos pulsar el botón *Save*. Aparecerá una ventana emergente para elegir el nombre del archivo de extensión *dynResults* y el directorio en el que se quiere guardar.

Cuando la simulación está detenida, podemos ver gráficamente el estado de la red en el panel *physical topology* y el panel *virtual topology*. El usuario puede escoger en el menú del panel *physical topology* el *lightpath* que quiere representar sobre la topología física. Además de mostrar el *lightpath* veremos que longitud de onda le ha sido asignada en cada salto.

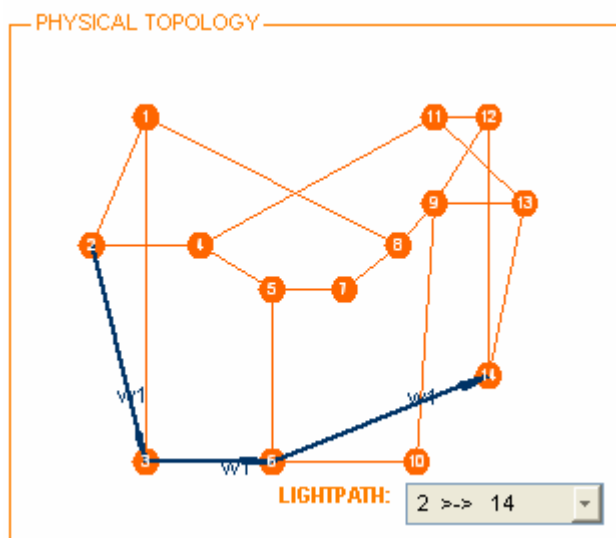


FIGURA 27 PANEL PHYSICAL TOPOLOGY

En el ejemplo de la figura anterior vemos un lightpath del nodo 2 al 14 sobre la topología física. Este lightpath sigue la secuencia de saltos físicos {2, 3, 6, 14} y la secuencia de longitudes de onda $\{w_1, w_1, w_1\}$.

El panel *virtual topology* representa la topología virtual cuando la simulación está parada y sobre ella podemos representar los flujos existentes. Disponemos de un menú para elegir el flujo a representar y veremos así que lightpath atraviesa.

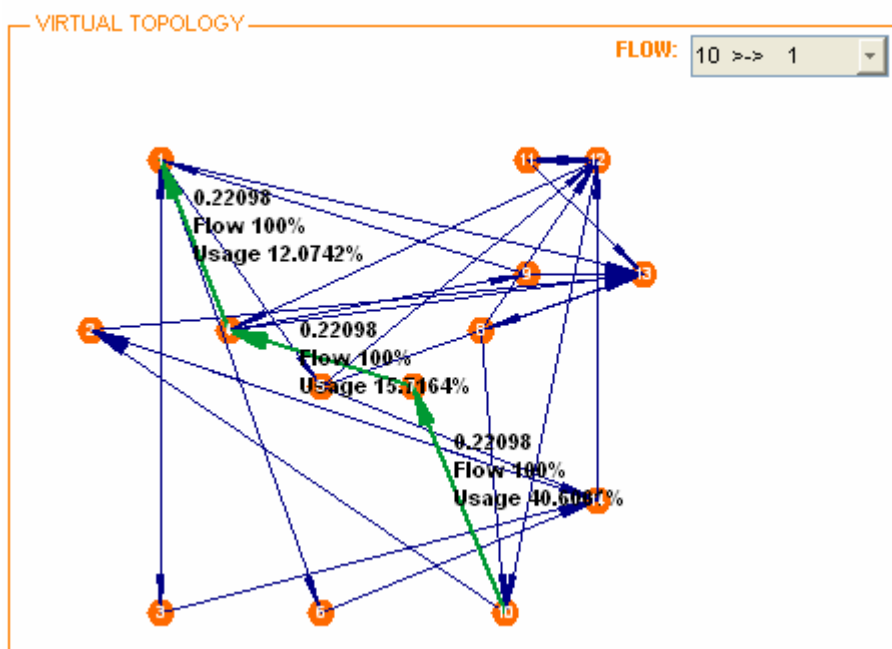


FIGURA 28 PANEL VIRTUAL TOPOLOGY



En la figura vemos que hemos optado por representar el flujo desde el nodo 10 al nodo 1. Este flujo atraviesa la siguiente secuencia de lightpaths {(10,7);(7,4);(4,1)} que vemos en verde. También se muestra la tasa del flujo en Gbps y la fracción que ocupa en cada lightpath. Si el lightpath sólo es atravesado por ese flujo tendremos un porcentaje *Usage* de valor 100%.

2.3 ESTRUCTURA DEL FICHERO .PHYS

Los archivos .phys permiten guardar los datos de la topología física en un archivo de texto que luego podremos utilizar para la simulación. Veamos el formato que deben tener estos archivos y la información que deben incluir.

```
// Information File

//Nr      Nr      Nr
//Nodes   Links   Levels
4         6         1

//Node Information Table - Physical Topology

//X      Y      Population  Lvl  NTx  NRx  NTWC
95.01  89.13   8393      1    4    4    1
23.11  76.21   5002      1    4    4    1
60.68  45.65   6539      1    4    4    1
48.60  1.85    8127      1    4    4    1

//Link Information Table - Physical Topology

//Ori Dest  Nr
//Nod Nod   Wavelengths
1     2     2
2     1     2
1     3     2
3     1     2
1     4     2
4     1     2

//Level Information Table - Physical Topology

1

//Lightpath Capacity

3.000

*****END_OF_FILE_.PHYS*****
*
```

FIGURA 29 ARCHIVO .PHYS



En la primera parte del archivo .phys debemos indicar el número de nodos, el número de enlaces y de niveles de la topología física. Seguidamente tenemos una tabla con la información de los nodos. En cada fila tenemos un nodo con sus coordenadas (x,y), la población, el nivel, el número de transmisores, de receptores y de conversores TWC.

Después tenemos una tabla con la información de los enlaces. En cada fila tenemos la siguiente información: nodo origen, nodo destino y número de longitudes de onda del enlace. A continuación se incluye una tabla con la información de los niveles en la que cada coordenada (x,y) es el peso del nivel entre un nodo con nivel 'x' y otro nodo con nivel 'y'. Por último, el archivo incluye un campo con la capacidad de los lightpath expresada en Gbps.

2.4 ESTRUCTURA DEL FICHERO .TRAFF

Los archivos .traff permiten guardar los datos de la matriz de tráfico en un archivo de texto que luego podremos utilizar para la simulación. Veamos el formato que deben tener estos archivos y la información que deben incluir.

```
// Information File

//Number of Nodes
4

//Traffic Table - Traffic

0.000 1.317 0.636 0.629
1.136 0.000 0.838 0.617
0.771 0.895 0.000 0.840
0.891 1.224 0.613 0.000

*****END_OF_FILE_.TRAFF*****
```

FIGURA 30 ARCHIVO .TRAFF

Vemos en la figura anterior que el archivo consta únicamente de dos campos. El primero de ellos indica el número de nodos de la red y el segundo es la matriz de tráfico propiamente dicha, donde cada coordenada (x,y) de la matriz es el flujo de tráfico medio entre los nodos 'x' e 'y' medido en número de bits por segundo.



2.5 ESTRUCTURA DE LOS MÓDULOS GENERADOR

En este punto veremos que estructura debe tener un modulo Generador. En primer lugar, su signatura debe ser de la siguiente manera:

```
[timeToNextEvent, flowDuration, flowAverageRate, pairIngressEgress] = generateFlow(generatorParameters, simTime, netState, phys)
```

Sus parámetros de entrada son:

- *generatorParameters*, variable de tipo string introducida en la aplicación en el campo *Generator Parameters*.
- *simTime*, instante actual de simulación
- *netState*, estructura de variables de estado de la red
- *phys*, estructura de variables de la topología física de la red

El módulo se encargará de realizar la traducción de la variable *generatorParameters* según se haya especificado desde el string a las variables correspondientes que espere.

En cuanto a los parámetros de salida, el módulo Generador tiene que devolver cuatro variables:

- *timeToNextEvent*, instante de llegada del próximo evento de nuevo flujo
- *flowDuration*, duración del nuevo flujo
- *flowAverageRate*, tasa media del nuevo flujo
- *pairIngressEgress*, par de nodos origen-destino del nuevo flujo.

2.6 ESTRUCTURA DE LOS MÓDULOS PLANIFICADOR

Veamos cual es la estructura que debe presentar un módulo Planificador. Empezamos por su signatura que debe ser como sigue:

```
[netState, infoWhatHappenedWithTheFlow] = dynamicAlgorithm(eventType, netState, phys, simTime, flowAverageRate, serialNumOfThisFlow, pairIngressEgress, algorithmParameters, flowDuration)
```

Sus parámetros de entrada son:

- *eventType*, si toma el valor '0' se trata de un evento de llegada de nuevo flujo y si se trata de un '1' es un evento fin de un flujo existente
- *netState*, estructura de variables de estado de la red
- *phys*, estructura de variables de la topología física de la red.
- *simTime*, instante actual de simulación



- *flowAverageRate*, tasa media del nuevo flujo. Si se trata de un evento fin, esta variable no es utilizada.
- *serialNumOfThisFlow*, número de serie que identifica al flujo de forma inequívoca
- *pairIngressEgress*, vector de identificadores de nodo origen y destino del flujo
- *algorithmParameters*, variable de tipo string introducida en la aplicación en el campo *Algorithm Parameters*
- *flowDuration*, duración del nuevo flujo.

El módulo se encargará de realizar la traducción de la variable *algorithmParameters* según se haya especificado desde el string a las variables correspondientes que espere.

En cuanto a los parámetros de salida, el módulo Planificador tiene que devolver el siguiente par de variables:

- *netState*, estructura de variables del estado de la red actualizada
- *infoWhatHappenedWithTheFlow*, si se trata de un evento de llegada de nuevo flujo, será '0' si el flujo ha podido ser cursado y '1' si no ha sido posible. Si es un evento fin de flujo, la variable siempre será '0'.

2.7 ESTRUCTURA DE LAS VARIABLES CURRENTNETWORKSTATE

Se trata de una estructura de variables que definen el estado de la red en un instante determinado. La estructura consta de los siguientes campos:

- *netState_lightpathTable* (Lx2): Cada fila es un lightpath, donde la primera columna es el nodo origen del lightpath y la segunda, el nodo destino. (L es el número de lightpaths en la red).
- *netState_lightpathRoutingMatrix* (LxM): Cada fila i es un lightpath, donde un elemento (i,j) es la longitud de onda del lightpath i en la fibra j (0 si el lightpath i no pasa por la fibra j). Hay tantas columnas como enlaces físicos. Debe garantizarse que la fila i de esta matriz haga referencia al mismo LP que la fila i de lightpathTable. (L es el número de lightpaths en la red y M es el número de enlaces físicos).
- *netState_flowTable*: (Fx5): Cada fila es un flujo y cada columna representa los siguientes campos para un flujo dado:
 - Número de serie del flujo. Este número es necesario únicamente para que el evento "fin de flujo existente" sepa a qué flujo se refiere. Tiene que ser un número consecutivo y único que el bucle principal asigna a cada flujo.
 - Nodo Origen del flujo



- Nodo Destino del flujo
- Tasa media del flujo
- Instante inicial del flujo
- Duración del flujo

(F es el número de flujos cursados).

- *netState_flowRoutingMatrix* (F×L): Cada fila i es un flujo, donde un elemento (i,j) es el número de Gbps del flujo i que circula por el lightpath j. Hay tantas columnas como lightpaths. Debe garantizarse que la fila i de esta matriz haga referencia al mismo flujo que la fila i de flowTable. (F es el número de flujos cursados y L es el número de lightpaths en la red).
- *netState.numberOfWorkAvailableTWCs*(1×N): Cada elemento i es el número de conversores TWC disponibles en el nodo i. (N es el número de nodos en la red).

2.8 EXTENSIÓN DE LA HERRAMIENTA

La herramienta puede ser extendida fácilmente de varias maneras. Podremos probar cualquier topología física simplemente creando su archivo *.phys* con el formato explicado en el punto 2.3 y guardándolo en el directorio *physicalTopology*. Otra forma de extender la herramienta es creando nuevos módulos del tipo Generador y Planificador como funciones de MATLAB con la signatura que es descrita en los puntos 2.6 y 2.7. Los módulos Generador se deben guardar en el directorio *dynamicGenerators* y el módulo Planificador en el directorio *dynamicAlgorithms*.

3 HEURÍSTICOS IMPLEMENTADOS

3.1 GENERACIÓN

El generador implementado en este Proyecto está constituido por dos partes. En primer lugar, hay una parte de inicialización que se encarga de construir la lista de eventos pendientes interna al generador cuando es la primera vez que es llamado. La lista de eventos del generador contiene el instante de llegada del siguiente flujo entre cada par de nodos posible.

Para construir la lista de eventos calculamos la media del tiempo entre llegadas de nuevos flujos (IAT) para cada par de nodos (i,j). Con este fin se emplea la siguiente expresión:

$$IAT(i,j) = Tasa\ Media\ (Gbps) * Duración\ Media\ (s) / Matriz\ de\ Tráfico\ (i,j)$$

Con estos valores para la media del tiempo entre llegadas obtenemos los tiempos hasta la siguiente llegada para cada par de nodos siguiendo una distribución exponencial.



En la segunda parte del generador obtenemos los valores de la tasa del flujo, la duración de éste y el instante del siguiente evento llegada. Tanto la tasa como la duración son obtenidas de una distribución exponencial cada una con la media correspondiente. En el caso de haber elegido truncamiento, el generador comprueba que la tasa del flujo no excede el valor de la capacidad del lightpath y si es así le asigna un nuevo valor hasta que se cumpla la condición.

Por último, mencionar el formato que deben tener la variable *generatorParameters* que llega de la aplicación. Ha de ser del siguiente tipo: *0.1 43200 1 nsfnet.traff*; donde el primer número es la tasa media de los flujos, el segundo es el valor de la duración media del flujo, el tercero indica que elegimos una exponencial truncada cuando es '1' y que no escogemos truncamiento si es '0', y el último es el nombre del fichero *.traff* del que tomará la matriz de tráfico.

3.2 PLANIFICACIÓN

En este Proyecto se implementa un algoritmo de planificación que intenta encaminar el flujo de cuatro formas diferentes cuando se produce un evento de llegada de nuevo flujo. Los pasos que realiza el algoritmo ante un evento de este tipo son los siguientes:

- 1) En primer lugar, comprueba si es posible acomodar el flujo en los lightpaths existentes buscando el camino más corto que además tenga capacidad para cursarlo.
- 2) Si no ha sido posible encaminar el flujo, prueba a crear un nuevo lightpath entre los nodos origen y destino del flujo. Para ello obtiene los 'k' caminos más cortos y para cada uno de ellos intenta el encaminamiento.
- 3) Cuando no es posible crear un nuevo lightpath, entonces el algoritmo intenta acomodar el flujo utilizando un lightpath existente y un nuevo lightpath valiéndose de un nodo intermedio.
 - a) Primero intenta acomodar el flujo empleando un lightpath existente entre el nodo origen y un nodo vecino. Para los lightpaths entre el origen y sus nodos vecinos que pueden soportar el nuevo flujo, prueba a establecer un nuevo lightpath entre el nodo vecino y destino. (Figura 20 (a)).
 - b) Si no es posible la opción anterior, intenta emplear ahora un lightpath existente entre los vecinos del destino y este. Al igual que antes, se queda con los lightpaths capaces de transportar el flujo y para ellos prueba el establecimiento de un lightpath entre el nodo origen y el vecino del destino. (Figura 20 (b)).

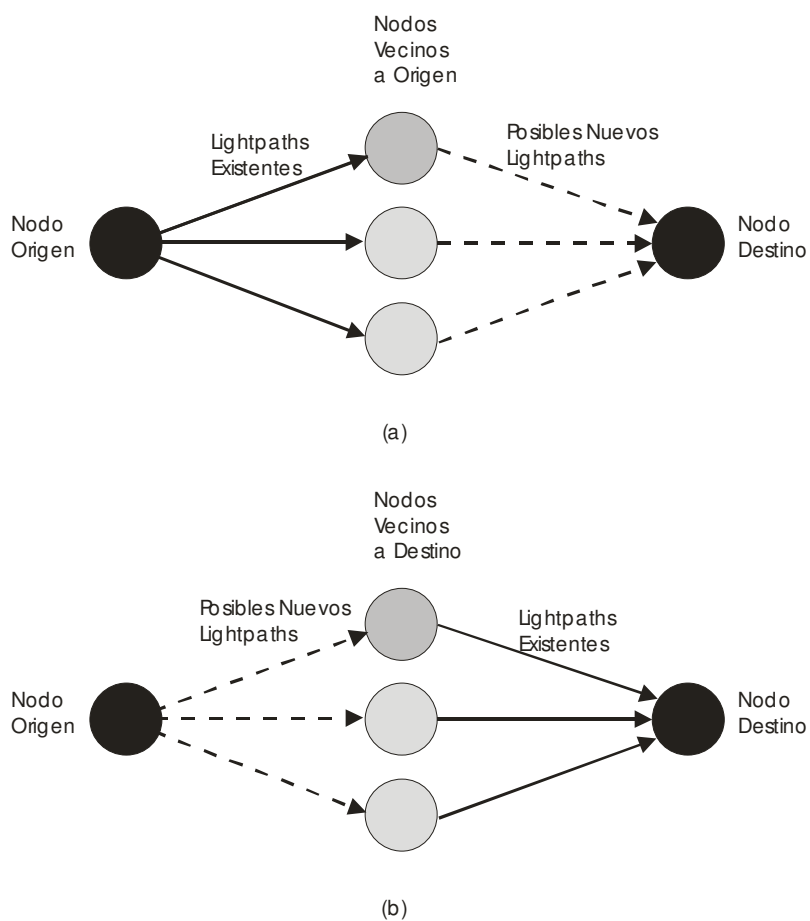


FIGURA 31 CASUÍSTICAS DE UTILIZACIÓN DE UN NODO VECINO

- 4) Cuando no son posibles ninguna de las opciones anteriores, entonces el algoritmo de Planificación intenta el establecimiento de dos nuevos lightpaths. Para ello, prueba de manera exhaustiva con cada uno de los posibles nodos intermedios.

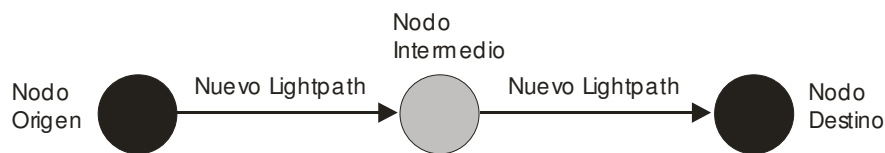


FIGURA 32 CREACIÓN DE DOS NUEVOS LIGHTPATHS

Si pasa por alguna de estas opciones actualiza de forma apropiada la estructura de variables de estado de la red.



Cuando se trata de un evento fin de flujo, el planificador simplemente se encarga de eliminar el flujo de la red para lo que actualiza la estructura de estado de la red y también comprueba si algún lightpath está vacío para poder liberarlo.



CAPÍTULO 4: RESULTADOS Y EJEMPLOS DE UTILIZACIÓN

En este capítulo se recogen y presentan algunos resultados para ilustrar el empleo de la herramienta. La topología física que hemos probado es una red NSFNET de 14 nodos, sin conversión, con un número infinito de transmisores E/O y receptores O/E por nodo y con una capacidad por lightpath de 10 Gbps. El módulo generador empleado toma la duración del flujo de una distribución exponencial con duración media 12 horas, y la tasa del flujo de otra distribución exponencial con una tasa media que variaremos en las distintas simulaciones llegando hasta 1 Gbps. En cuanto al tiempo entre llegadas de dos flujos desde el mismo nodo origen al mismo nodo destino también está distribuido exponencialmente. La media del tiempo entre llegadas desde el nodo i al nodo j puede ser diferente para cada par (i, j) . Para ello se toma el tráfico medio de la coordenada (i, j) de una matriz de tráfico dada.

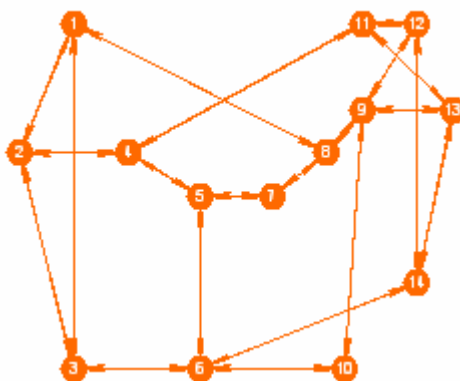


FIGURA 33 RED NSFNET DE 14 NODOS

En este caso se han elegido siete matrices de tráfico para las simulaciones: $T_1, T_2, T_3, T_4, T_5, T_6, T_7$. Todas estas matrices han sido calculadas a partir de una matriz de tráfico propuesta por la red NSFNET [19] (que denotamos por T). El patrón de tráfico sólo válido para esta red de 14 nodos se construye de la siguiente manera: 42 pares escogidos aleatoriamente tienen un tráfico tomado de una distribución uniforme $U(0,100)$ y a los pares de nodos restantes les corresponde un volumen de tráfico tomado de una distribución uniforme $U(0,1)$. La matriz de tráfico T_X se obtiene por un proceso de normalización que multiplica la matriz de tráfico inicial T por el factor que haga que el máximo tráfico generado desde cualquier nodo i a otro nodo cualquiera j sea igual a X Gbps.

El algoritmo de planificación empleado para las simulaciones está basado en la búsqueda de los k caminos más cortos (k -shortest path) empleando una implementación de este algoritmo en



su versión *path removing* [20] incluida en el directorio *library_Graph* del toolbox. El algoritmo trabaja de la siguiente manera:

- Ante la llegada de un flujo: primero, intenta acomodar el flujo empleando la topología virtual existente y minimizando el número de saltos virtuales. Con este fin se emplea el algoritmo *Capacitated Shortest Path* incluido en *library_Graph*. Si no es posible, intenta resolver el problema estableciendo un nuevo lightpath. Primero, prueba un lightpath directo entre el nodo origen y el nodo destino del flujo. Cuando esto no funciona, calcula el conjunto de nodos S_{in} que están conectados al nodo origen por un lightpath con capacidad suficiente. Para cada nodo S_{in} , intenta el establecimiento de un lightpath entre ese nodo y nodo destino. Cuando esto tampoco es posible, obtiene el conjunto de nodos S_{eg} que están conectados al nodo destino por un lightpath con capacidad suficiente. Para cada uno de los nodos en S_{eg} prueba a establecer un lightpath desde el nodo origen hasta ese nodo. Si esto no es posible, intenta el establecimiento de dos lightpaths nuevos empleando un nodo intermedio. Cuando nada de esto funciona, el flujo es bloqueado. Cada búsqueda de un lightpath desde el nodo origen al destino ha sido implementada primero obteniendo los k caminos más cortos en número de saltos físicos. Para cada camino posible, una longitud de onda disponible es seleccionada siguiendo el esquema *first-fit* implementado en *library_WA*.
- Ante un evento fin de flujo: el módulo de planificación borra el flujo y comprueba si, a consecuencia de esto, alguno de los lightpaths atravesados por este flujo han quedado vacíos. Si es así, los lightpaths también son eliminados liberando los recursos correspondientes.

A continuación se muestran varias gráficas que representan la probabilidad de bloqueo obtenida para distintas simulaciones, cuyos datos de entrada detallaremos y analizaremos los resultados obtenidos. Todas ellas han sido obtenidas empleando un tiempo de simulación total de 60 horas y un tiempo transitorio de 12 horas.

Realizamos una primera simulación con los siguientes datos de entrada: tasa media de flujo de 1 Gbps, valor de $k=1$, sin conversión de longitud de onda en los nodos, con las matrices de tráfico antes mencionadas, y para dos 8 y 16 longitudes de onda. La representación obtenida es la siguiente:

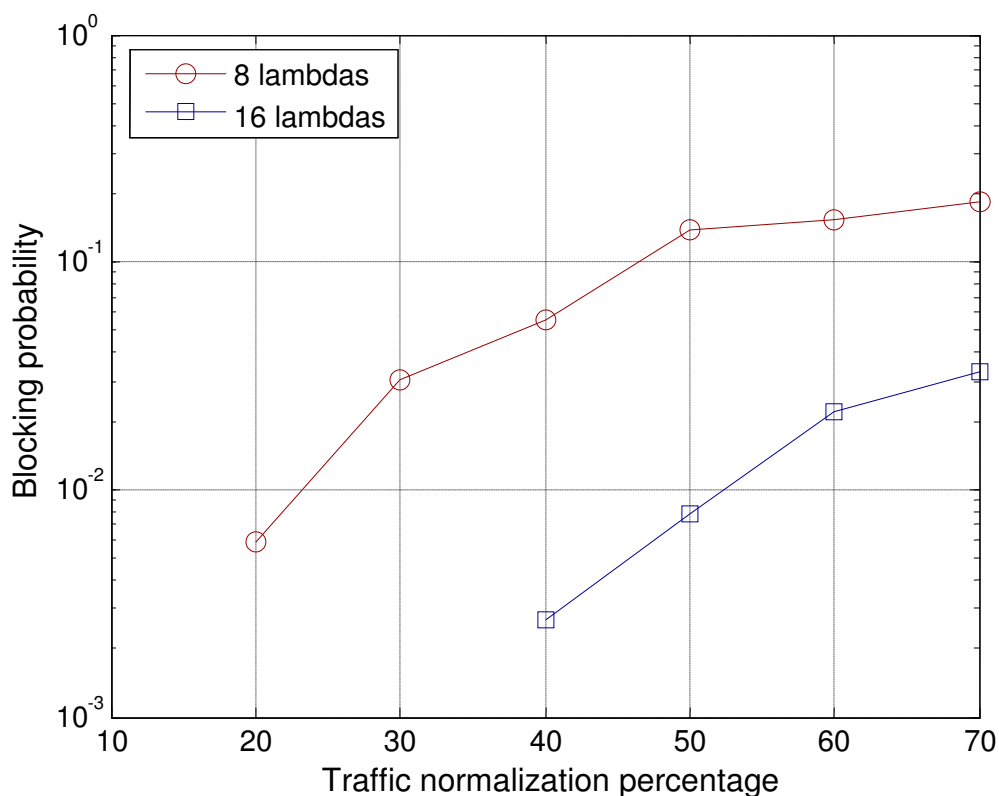


FIGURA 34 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA

En la figura anterior podemos comprobar que la probabilidad de bloqueo es superior para el caso en el que disponemos de 8 longitudes de onda. Además podemos observar en ambas curvas un aumento del bloqueo con el aumento del porcentaje de normalización.

A continuación volvemos realizamos la simulación pero esta vez cambiamos el valor de la tasa media del flujo a 0.5 Gbps. El resultado ahora es:

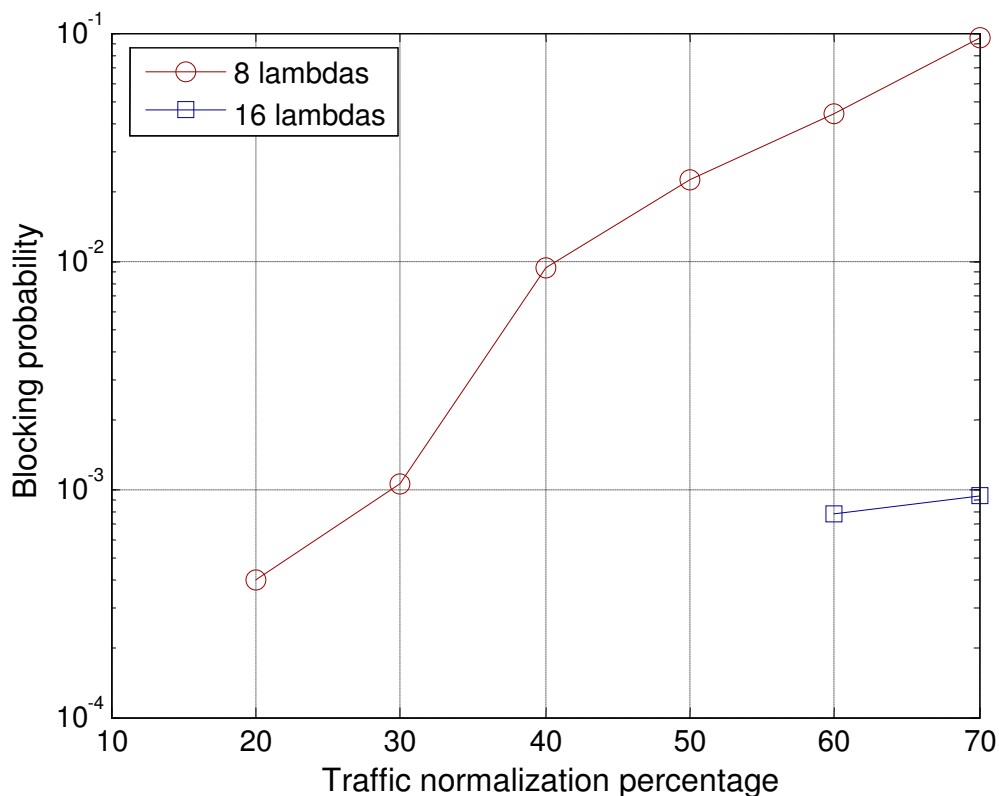


FIGURA 35 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA

Comprobamos que al reducir el valor de la tasa media del flujo, en el caso de disponer de 16 longitudes de onda la probabilidad de bloqueo se reduce considerablemente. En cuanto a la otra curva, vemos que para valores de normalización pequeños el bloqueo también ha disminuido. Sin embargo, con el aumento del porcentaje de normalización del tráfico observamos un incremento más rápido de la probabilidad de bloqueo aunque sin alcanzar los valores del caso anterior.

Seguidamente aumentamos el valor de k manteniendo el resto de parámetros con los valores de la primera simulación. La gráfica que obtenemos para los valores de $k=\{2, 3,4\}$ son las que se muestran seguidamente:

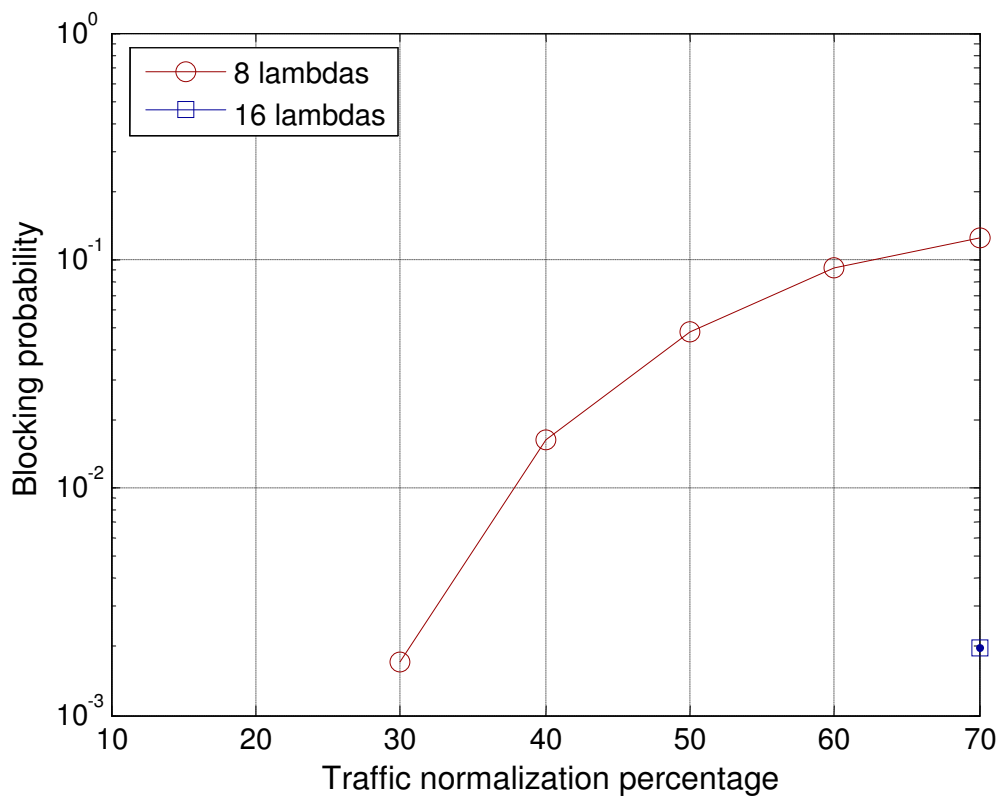


FIGURA 36 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA

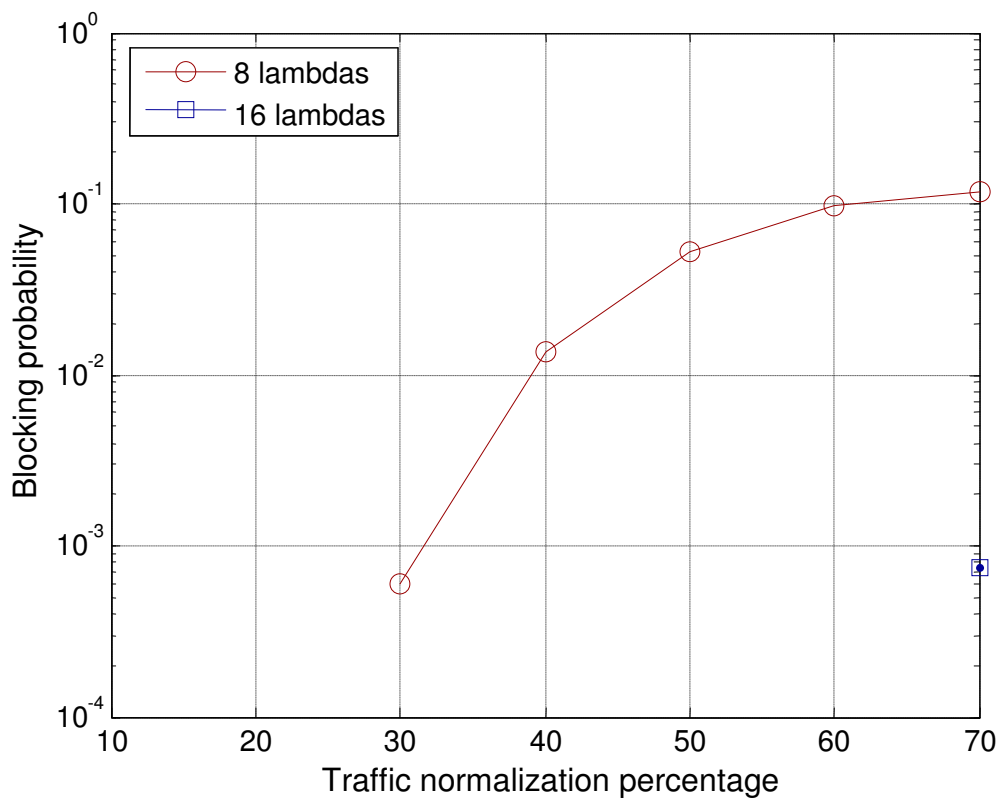


FIGURA 37 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA

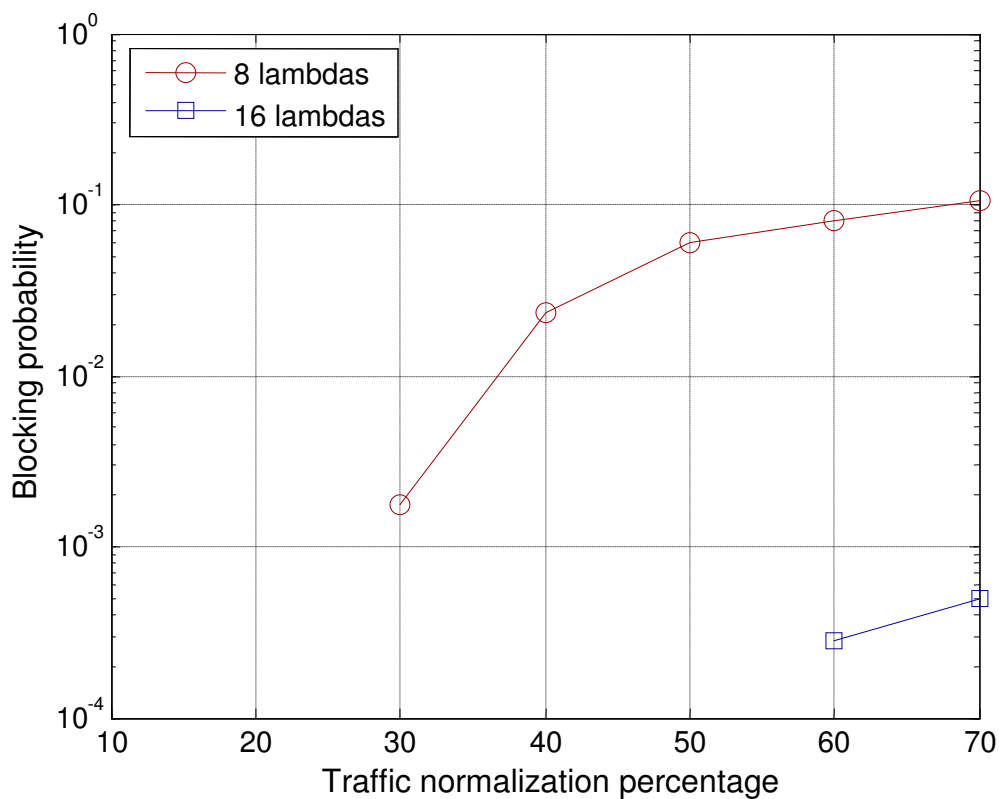


FIGURA 38 PROBABILIDAD DE BLOQUEO VS NÚMERO DE LONGITUDES DE ONDA

Si comparamos estas tres representaciones con la primera gráfica obtenida para $k=1$, comprobamos como al aumentar k a 2 la probabilidad de bloqueo disminuye de forma notable. Si continuamos incrementando k hasta 3 vemos que el bloqueo sigue reduciéndose. Sin embargo, para $k=4$ no observamos una mejora sensible respecto a los valores con $k=3$.

A continuación representamos la probabilidad de bloqueo para 8 longitudes de onda y los cuatro valores de k s reuniendo las simulaciones adjuntadas anteriormente.

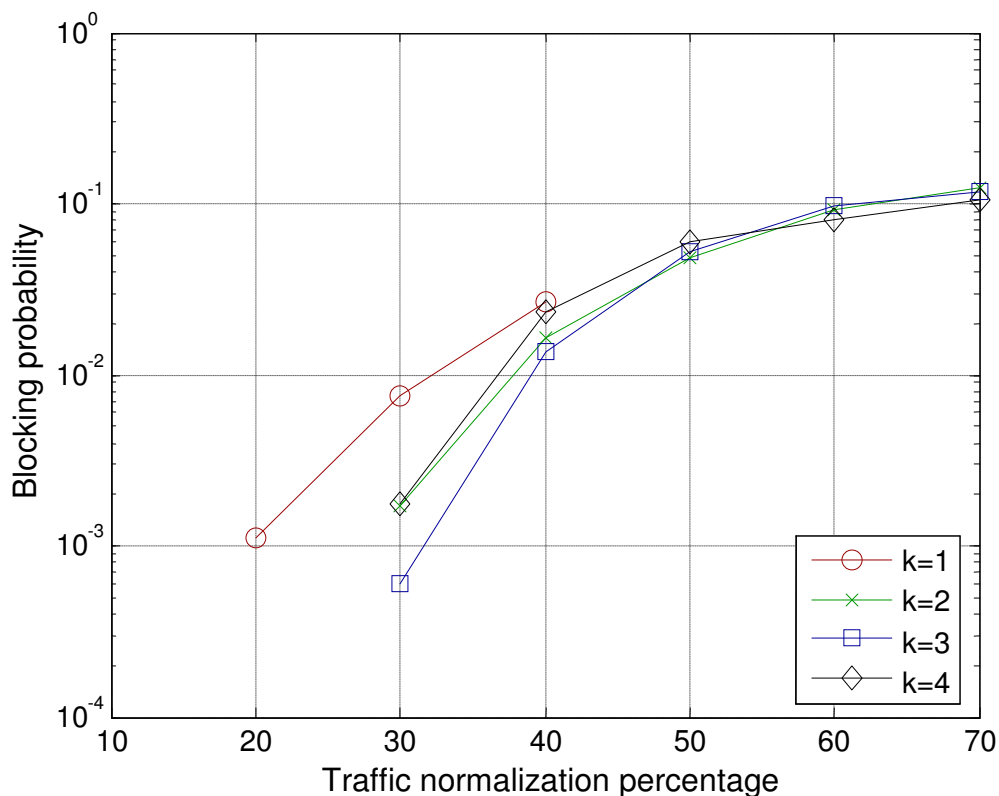


FIGURA 39 PROBABILIDAD DE BLOQUEO VS PARÁMETRO K

En la representación anterior comprobamos que para los valores más pequeños del porcentaje de normalización, la diferencia entre las probabilidades de bloqueo para los distintos k s es notable, mientras que conforme los valores de normalización aumentan las diferencias entre el bloqueo se reducen de forma notable.

Vemos ahora cuales son los resultados obtenidos para la probabilidad de bloqueo para los valores de $k=\{1,2,3,4\}$, 8 longitudes de onda y una tasa media de flujo de 0.3 Gbps. También elegimos otro rango de porcentajes de normalización.

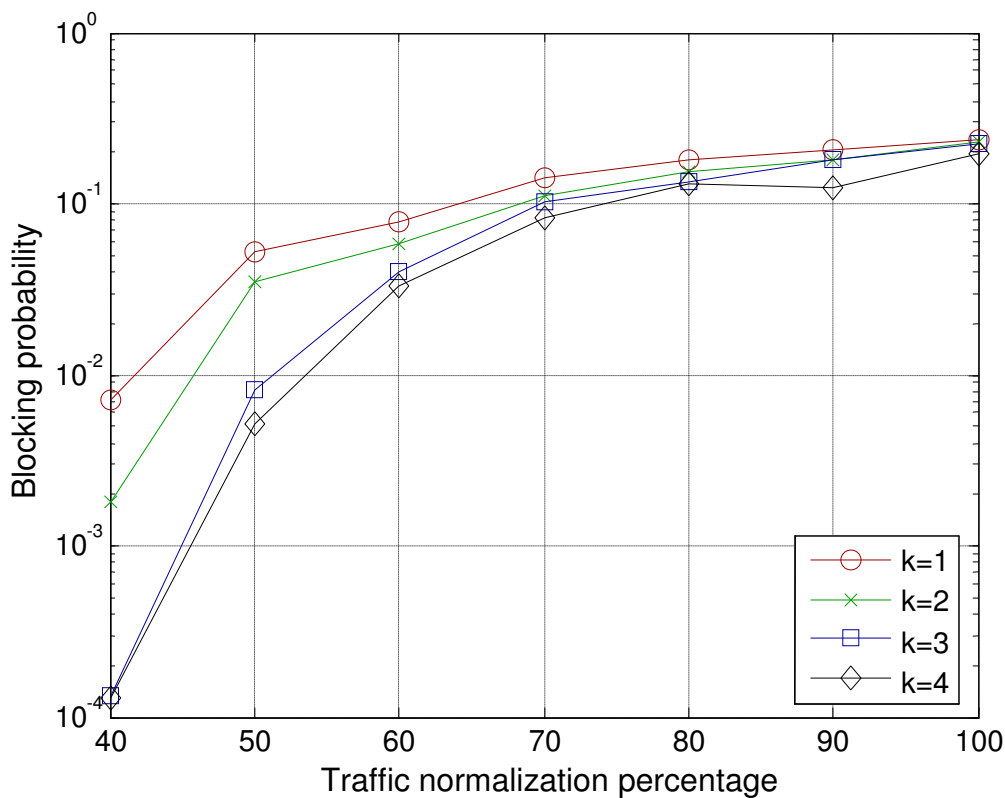


FIGURA 40 PROBABILIDAD DE BLOQUEO VS PARÁMETRO K

Nos encontramos ante una figura similar a la anterior. Podemos ver tal y como comentábamos antes que cuando los porcentajes de normalización son más altos la diferencia entre las probabilidades de bloqueo para los distintos ks se reduce, mientras que si los valores de normalización son menores las diferencias aumentan.



CAPÍTULO 5: CONCLUSIONES Y LÍNEAS FUTURAS

Actualmente las redes de fibra óptica constituyen uno de los sectores de mayor desarrollo en el sector de las comunicaciones. Ofrecen una serie de prestaciones frente a los medios de transmisión tradicionales (elevados anchos de banda, bajísimas tasas de error, bajos niveles de atenuación y distorsión de señal, bajos requerimientos de potencia,...) que las hacen especialmente indicadas como soporte físico para redes de troncales de larga distancia. Pero a la hora de la aplicación práctica de esta tecnología en redes de comunicación se nos plantea el problema del llamado “cuello de botella electrónico”. Este cuello de botella consiste en el desajuste existente entre las elevadas tasas de transmisión que la fibra óptica permite (tasas de hasta 40 gigabits por segundo (OC-768) para un canal sencillo de longitud de onda en sistemas *SONET/SDH* de 160 longitudes por fibra) y la baja velocidad de procesamiento electrónico a la que los datos son tratados en los nodos de la red (nodos de enrutamiento o nodos de acceso de un usuario final). Esto significa que el factor que limita actualmente las tecnologías basadas en la fibra óptica no es el ancho de banda de los enlaces de fibra, sino las limitaciones de velocidad del equipamiento electrónico de los nodos.

Para superar el cuello de botella electrónico se han propuesto diferentes soluciones, pero una de las más interesantes son las redes *WDM* (*Wavelength Division Multiplexing*) de conmutación de *lightpaths*, también llamadas redes *WR* (*Wavelength Routed*). Estas redes *WDM* intentan eliminar o disminuir la necesidad de procesamiento electrónico en los nodos intermedios de enrutamiento de la red (no se puede eliminar el procesamiento electrónico en los nodos de acceso de la red). Para disminuir esta necesidad de equipamiento electrónico, se establece una serie de rutas continuas de longitud de onda entre los nodos, sin requerimiento de conversión óptica-electrónica-óptica y almacenamiento electrónico en los nodos intermedios de las rutas. Estas rutas son llamadas *lightpaths* y son, por tanto, caminos de comunicación todo-óptico, con lo que se elimina la necesidad de electrónica en los nodos intermedios, ya que la conmutación y el enrutamiento será realizado por conmutadores ópticos. En cualquier caso, puede resultar que un flujo de tráfico deba cambiar de *lightpath* para llegar a su destino, en tal caso, en el nodo donde conmute de *lightpath* tendrá lugar un procesamiento electrónico y las necesarias conversiones opto electrónicas. Estos nodos son también los puntos de *grooming* de la red (los nodos donde se puede introducir o sacar datos de los *lightpaths*).

La pregunta que nos podemos hacer llegados a este punto es como establecemos estos *lightpaths* sobre nuestra red para reducir al máximo las conversiones opto electrónicas y el procesamiento electrónico. El conjunto de todos los *lightpaths* establecidos sobre una red recibe el nombre de topología virtual. El objetivo del presente Proyecto Fin de Carrera es el diseño de estas topologías virtuales sobre la topología física, real, de nuestra red. Además el Proyecto se enfocó la planificación de redes *WDM wavelength-routing* considerando demanda dinámica de tráfico.



Los distintos módulos de la herramienta se han implementado en MATLAB. Quizá en términos de tiempo de computación no sea el entorno idóneo, pero por otra parte nos ofrece una amplia funcionalidad matemática que implica la implementación de nuevos algoritmos y generadores sea más sencilla. Además la herramienta se puede extender con gran facilidad con sólo añadir las nuevas funciones con la signatura adecuada en los correspondientes directorios. Cabe decir también que el lenguaje empleado por MATLAB es sencillo con lo que requiere menos entrenamiento para el usuario. Todas estas razones hacen que este entorno se haya elegido para el desarrollo del Proyecto frente a otras opciones como la implementación en C++.

La aplicación GUI resultante proporciona una herramienta atractiva y accesible para poder caracterizar en diversos modos, y con varias opciones de simulación un algoritmo. Además la aplicación puede ser orientada a fines educacionales.

En cuanto a las posibilidades futuras de la herramienta *Dyn-MatPlanWDM*, será interesante la integración con *MatPlanWDM* con el fin de reunir en una sola aplicación la posibilidad de realizar planificación con demanda de tráfico estática y dinámica. También se pueden dirigir esfuerzos al desarrollo de algoritmos de planificación con demanda de tráfico dinámica con la finalidad de poder obtener comparativas.

Las líneas futuras para este trabajo son muchas. Por un lado, avanzar en la interfaz gráfica para el soporte de esta herramienta en la docencia de contenidos asociados a la optimización en redes. Por otro lado, la utilización de la misma para la investigación de algoritmos de planificación, y su evaluación. En concreto, una línea futura para el corto plazo será la evaluación comparativa de los algoritmos existentes para escenarios de tráfico *multi-hour* donde la demanda de tráfico varía según la hora del día.



REFERENCIAS

- [1] <http://www-x.antd.nist.gov/glass/> (last access: 19th November, 2007)
- [2] F. Mouveaux, F. Lapeyrere, N. Golmie. (2000, February). *MERLIN, User's Manual and Programmer's Guide*. Vs 1.0. National Institute of Standards and Technology. Available: http://w3.antd.nist.gov/Hsntg/prd_merlin.html.
- [3] B. Wen et al., "Optical Wavelength Division Multiplexing (WDM) Network Simulator (OWns): Architecture and Performance Studies," *SPIE Optical Networks Magazine Special Issue on "Simulation, CAD, and Measurement of Optical Networks"*, Mar. 2001.
- [4] <http://www.opnet.com> (last access: 19th November, 2007)
- [5] http://web1.rsoftdesign.com/products/network_modeling/MetroWAND/ (last access: 19th November, 2007)
- [6] <http://www.optiwave.co.kr/product/optisystem.htm> (last access: 19th November, 2007)
- [7] C.S.R. Murthy and M. Gurusamy, *WDM Optical Networks (Concepts, Design and Algorithms)*. Upper Sadle River, NJ: Prentice Hall PTR, 2002, ch 2.
- [8] H. Zang, J.P. Jue and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Networks Magazine*, vol. 1, no. 1, pp 47-60, Jan. 2000.
- [9] X. Chu, B. Li, Z. Zhang, "A Dynamic RWA Algorithm in a Wavelength-Routed All-Optical Network with Wavelength Converters," in *22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, Mar. 2003, vol. 3, pp.1795-1804
- [10] B. Li, X. Chu, K. Sohraby, "Routing and wavelength assignment versus wavelength converters placement in all-optical networks," *IEEE Communications Magazine*, vol. 41, no. 8, pp. S22-S28, Aug. 2003
- [11] X. Zhang, C. Qiao, "Wavelength Assignment for Dynamic Traffic in Multi-Fiber WDM Networks," in *7th International Conference on Computer Communications and Networks (ICCCN '98)*, 1998, pp. 479-485
- [12] Y. Zhang, K. Taira, H. Takagi, S.K. Das, "An efficient Heuristic for Routing and Wavelength Assignment in Optical WDM Networks," *IEEE International Conference on Communications*, vol. 5, pp. 2734- 2739 , 2002



- [13] T. Fabry-Asztalos, N.M. Bhide, K.M. Sivalingam, "Routing Mechanisms Employing Adaptive Weight Functions for Shortest Path Routing in Optical WDM Networks," *Photonic Network Communications*, vol. 3, no. 3, pp. 227-236, Jul. 2001
- [14] K.C. Lee and V.O.K. Li. "A wavelength-convertible optical network," *Journal of Lightwave Technology*, vol. 11, no. 5/6, pp. 962-970, May/June. 1993
- [15] Z. Zhang, A.S. Acampora, "A heuristic wavelength assignment algorithm for multihop WDM networks with wavelength routing and wavelength re-use," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 281-288, Jun. 1995.
- [16] C.S.R. Murthy and M. Gurusamy, "A Time Optimal Wavelength Rerouting Algorithm for Dynamic Traffic in WDM Networks," *Journal of Lightwave Technology*, vol. 17, no. 3, pp 406-417, Mar. 1999
- [17] R. Ramaswami, K.N. Sivarajan, "Design of Logical Topologies for Wavelength-routed Optical Networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 5, pp 840-851, Jun 1996
- [18] V.R. Konda, T.Y. Chow, "Algorithm for Traffic Grooming in Optical Networks to Minimize the Number of Transceivers," *IEEE Workshop on High Performance Switching and Routing*, pp. 218-221, 2001