

**Departamento de Organización Industrial y Gestión de  
Empresas  
Escuela Superior de Ingenieros  
Universidad de Sevilla**

**PROYECTO FIN DE CARRERA**

**ALGORITMO GENÉTICO PARA EL  
ENCAMINAMIENTO DINÁMICO DE TRÁFICO CBR  
SOBRE REDES ATM**

**Autor: Fernando del Puerto Pacheco  
Tutor: Pablo Cortés Achedad**

**Abril de 2007**



## ÍNDICE

<b>1</b>	<b>OBJETO DEL PROYECTO .....</b>	<b>7</b>
<b>2</b>	<b>EVOLUCIÓN DE LAS REDES DE COMUNICACIÓN.....</b>	<b>9</b>
<b>3</b>	<b>LA FIBRA ÓPTICA.....</b>	<b>11</b>
3.1	ORIGEN Y EVOLUCIÓN.....	11
3.2	CONCEPTOS BÁSICOS DE LA FIBRA ÓPTICA.....	12
3.3	MEDIO FÍSICO .....	13
3.4	FUNCIONAMIENTO Y CARACTERÍSTICAS ÓPTICAS .....	14
3.4.1	Ventajas .....	14
3.4.2	Inconvenientes .....	14
3.5	TIPOS DE FIBRA .....	15
<b>4</b>	<b>RED DIGITAL DE SERVICIOS INTEGRADOS.....</b>	<b>17</b>
4.1	DEFINICIÓN DE RDSI.....	17
4.2	PRINCIPIOS BÁSICOS DE LA RDSI.....	18
4.3	CANALES Y ACCESOS BÁSICOS EN LA RED DIGITAL DE SERVICIOS INTEGRADOS .....	19
4.3.1	Acceso Básico .....	20
4.3.2	Acceso Primario .....	20
4.4	RDSI-BA (RDSI-BANDA ANCHA).....	21
4.4.1	Servicios de Banda Ancha.....	22
<b>5</b>	<b>JERARQUÍA DE TRANSMISIÓN SÍNCRONA.....</b>	<b>27</b>
5.1	SDH .....	27
5.2	SONET .....	28
5.3	JERARQUÍA DE SEÑALES .....	29
5.4	JERARQUÍA DEL SISTEMA .....	29
<b>6</b>	<b>MODO DE TRANSFERENCIA ASÍNCRONO.....</b>	<b>33</b>
6.1	ANTECEDENTES DE ATM.....	33
6.2	COMPARATIVA ATM Y STM.....	33
6.3	TIPOS DE CONMUTACIÓN.....	34
6.3.1	Conmutación de circuitos .....	34
6.3.2	Conmutación de paquetes.....	35
6.4	DESCRIPCIÓN DE LA CÉLULA ATM.....	36
6.5	COMPOSICIÓN DE UNA CÉLULA ATM.....	37
6.5.1	Campo de ruta VPI/VCI.....	37
6.5.2	PTI. Identificador de carga útil.....	38
6.5.3	CLP. Identificación de prioridad de pérdida de células.....	38
6.6	CANALES VIRTUALES Y CAMINOS VIRTUALES .....	39
6.6.1	Canales y caminos virtuales. STM .....	40
6.6.2	Canales y caminos virtuales. ATM .....	41
6.7	PROCESO DE CONMUTACIÓN EN ATM.....	42
6.7.1	Probabilidad de pérdida de células .....	42
6.7.2	Retardo de conmutación .....	42
6.7.3	Tipos de conmutadores en ATM .....	43
6.8	GESTIÓN DE TRÁFICO Y CALIDAD DE SERVICIO EN ATM.....	46
6.8.1	Categorías de servicio.....	46
6.8.2	Gestión de tráfico.....	48
<b>7</b>	<b>ANTECEDENTES E HIPÓTESIS DEL PROBLEMA.....</b>	<b>51</b>
<b>8</b>	<b>ENCAMINAMIENTO EN UNA RED ATM.....</b>	<b>53</b>

8.1	JUSTIFICACIÓN DE ATM.....	53
8.2	ESCALAS DE TIEMPO .....	56
8.3	PÉRDIDAS PRODUCIDAS EN LOS CONMUTADORES ATM.....	58
8.4	CONMUTADORES CON COLAS .....	59
8.5	ECUACIONES DE BALANCE.....	64
8.6	COLAS INFINITAS .....	65
8.7	COLAS FINITAS.....	67
8.7.1	Tamaño de la cola.....	70
8.8	MODELO LINEAL DE ENCAMINAMIENTO ATM .....	70
8.8.1	Parámetros del modelo .....	70
8.8.2	Variables del modelo .....	71
8.8.3	Parámetros del modelo .....	73
8.8.4	Restricciones del modelo.....	74
8.8.5	Función Objetivo .....	77
8.8.6	Modelo.....	78
<b>9</b>	<b>RESOLUCIÓN DEL MODELO.....</b>	<b>81</b>
9.1	ALGORITMO GENÉTICO .....	81
9.1.1	Población en un algoritmo genético .....	82
9.2	CODIFICACIÓN DE LAS SOLUCIONES.....	83
9.2.1	Individuos .....	83
9.2.2	Población .....	84
9.3	OPERADORES .....	85
9.3.1	Operador cruce .....	86
9.3.2	Operador mutación .....	87
9.3.3	Reemplazo de individuos .....	87
9.3.4	Criterio de parada del algoritmo genético .....	89
9.4	APTITUD DE LOS INDIVIDUOS.....	90
9.4.1	Simplificación del modelo.....	91
9.5	ALGORITMO PARA LA OBTENCIÓN DE LA APTITUD DEL INDIVIDUO.....	93
9.6	ALGORITMO DE SELECCIÓN DE RUTA MÍNIMA.....	95
9.6.1	Pseudocódigo del algoritmo de selección de ruta mínima .....	95
<b>10</b>	<b>DESCRIPCIÓN DEL SOFTWARE DEL ALGORITMO DE OPTIMIZACIÓN DE ENCAMINAMIENTO ATM.....</b>	<b>101</b>
10.1	DATOS DE ENTRADA DEL ALGORITMO GENÉTICO.....	101
10.2	PSEUDOCÓDIGO DEL ALGORITMO GENÉTICO .....	102
10.3	VARIABLES GLOBALES DEL PROGRAMA .....	103
10.4	CONSTANTES SIMBÓLICAS DEL PROGRAMA .....	104
10.5	ESTRUCTURAS EMPLEADAS EN EL PROGRAMA.....	105
10.6	DIAGRAMA DE FLUJOS DEL PROGRAMA .....	108
10.6.1	Simbología.....	108
10.7	FUNCIÓN PRINCIPAL MAIN .....	109
10.7.1	Descripción de las funciones .....	110
10.8	OTRAS FUNCIONES.....	121
10.8.1	Función calcula_caminos .....	121
10.8.2	Función generar_individuo.....	123
10.8.3	Función mutación .....	126
10.8.4	Función cruce .....	129
10.8.5	Función calculaaptitud.....	133

<b>11</b>	<b>ANÁLISIS DE REDES DE ENSAYO. CALIBRACIÓN DE PARÁMETROS.</b>	<b>139</b>
11.1	ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED A ....	140
11.1.1	Estudio de las pérdidas medias frente al número de iteraciones del algoritmo .....	141
11.1.2	Estudio de la solución alcanzada respecto al tamaño del búfer.....	143
11.1.3	Estudio de la solución alcanzada respecto al grado de servicio .....	144
11.1.4	Estudio de la solución alcanzada frente al tamaño del búfer y grado de servicio .....	145
11.1.5	Estudio de la solución alcanzada en función del número de iteraciones del algoritmo.....	146
11.1.6	Estudio de la solución alcanzada en función del tráfico.....	147
11.2	ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED B ....	148
11.2.1	Estudio de las pérdidas medias frente al número de iteraciones del algoritmo .....	149
11.2.2	Estudio de la solución alcanzada en función del número de iteraciones del algoritmo.....	150
11.2.3	Estudio de la solución alcanzada en función del número de iteraciones del algoritmo. Para tamaños de población diez y veinte individuos. ....	152
11.2.4	Estudio de la solución alcanzada en función del número de iteraciones del algoritmo y del tamaño del búfer.....	153
11.2.5	Estudio de la solución alcanzada en función del tráfico.....	156
11.3	ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED C ....	157
11.3.1	Estudio de la pérdidas medias en función del número de iteraciones del algoritmo .....	158
11.3.2	Estudio de la solución alcanzada en función del número de de iteraciones del algoritmo.....	159
11.3.3	Estudio de la solución alcanzada en función del tráfico.....	160
11.3.4	Estudio de la solución alcanzada respecto al grado de servicio .....	161
<b>12</b>	<b>ESFUERZO COMPUTACIONAL .....</b>	<b>163</b>
12.1	ANÁLISIS DEL ESFUERZO COMPUTACIONAL PREVIO AL ALGORITMO GENÉTICO. ....	163
12.1.1	Tiempo de obtención de todas los posibles caminos entre pares origen-destino .....	164
12.1.2	Tiempo de generación de la población inicial de partida frente a diversas redes. ....	165
12.2	ANÁLISIS DEL ESFUERZO COMPUTACIONAL DEL ALGORITMO GENÉTICO .....	166
<b>13</b>	<b>ANÁLISIS DE RESULTADOS PARA LA BATERÍA DE PROBLEMAS..</b>	<b>167</b>
13.1	CONSTRUCCIÓN DE UNA BATERÍA DE REDES DE ENSAYO .....	167
13.2	SELECCIÓN DE PARÁMETROS PARA EL ALGORITMO GENÉTICO .....	168
13.3	COMPARACIÓN ALGORITMO GENÉTICO Y RUTA MÍNIMA .....	175
13.3.1	Resultados.....	175
<b>14</b>	<b>CONCLUSIONES .....</b>	<b>185</b>
<b>15</b>	<b>BIBLIOGRAFÍA .....</b>	<b>189</b>



# 1 OBJETO DEL PROYECTO

El presente documento se puede dividir en tres bloques bien diferenciados.

## *Primer bloque*

En el primero se hace un análisis de la actualidad de las telecomunicaciones, y se presentan las tecnologías en las que se basa el modelo realizado. Esto incluiría los capítulos 2, 3, 4 y 5.

En el capítulo dos se hace un pequeño análisis de la fibra óptica, viendo su origen y evolución, sus conceptos básicos, su funcionamiento y características ópticas, los tipos de fibra y algunas ventajas e inconvenientes de este medio físico.

En el capítulo tres y cuatro se hace un recorrido a través de la red digital de servicios integrados y la jerarquía digital síncrona respectivamente.

Ya en el apartado quinto, se hace un exhaustivo análisis del modo de transferencia síncrono, más concretamente ATM, que será el modo de transferencia idóneo para el problema que nos interesa y sobre el que se analizará el modelo de encaminamiento dinámico.

## *Segundo bloque*

En el segundo se plantea el modelo de optimización propiamente dicho, y se busca un método de resolución mediante la aplicación de algoritmos genéticos. También se incluye en este bloque la parte que se refiere a la programación del modelo, que se realiza en el lenguaje de programación en C. Incluye los capítulos 6, 7, 8 y 9.

En el capítulo seis se estudian los antecedentes e hipótesis del problema.

En la programación del modelo no sólo se incluirá la correspondiente al algoritmo genético, sino también la correspondiente al algoritmo de selección de ruta mínima. Incluye los capítulos 7, 8 y 9.

La comparación de los dos algoritmos y sus respectivos resultados serán analizados en el tercer bloque.

## *Tercer bloque*

El tercer, y último bloque, se refiere a la obtención de resultados y el pertinente análisis de los mismos. Incluye los capítulos 10, 11 y 12.

En el capítulo diez, se hace un análisis de calibración de los parámetros tanto de los elementos tecnológicos de diseño (búfer), como del propio algoritmo de optimización, para distintas redes de ensayo. Se estudian diversas propuestas de redes, variando para cada una de ellas las condiciones topológicas y de tráfico. Esto es: el número de conexiones establecidas durante cada intervalo de conexión, el grado de servicio y el número de nodos y de enlaces. Las redes de ensayo elegidas son: una red de 7 nodos y 18 arcos; una red de 10 nodos y 30 arcos, y una red de 15 nodos y 46 arcos. Para cada una de ellas se realizarán, entre otros, los siguientes estudios:

- Estudio de las pérdidas medias frente al número de iteraciones del algoritmo.
- Estudio de la solución alcanzada frente al tamaño del búfer.
- Estudio de la solución alcanzada frente al tamaño del búfer y el grado de servicio.
- Estudio de la solución alcanzada frente al número de iteraciones del algoritmo.
- Estudio de la solución alcanzada en función del tráfico.

En el capítulo once, se hace un estudio de tiempos de ejecución previos a la ejecución del algoritmo genético, y también para el propio algoritmo.

En el capítulo doce, y último, se toma de partida una batería de problemas, y se hace una propuesta de diseño para diversas topologías de red, así como la comparación entre el algoritmo genético y el algoritmo de selección de ruta mínima. Se variará tanto el número de nodos como el número de arcos. De este modo, se dispondrá desde una topología de red de diez nodos y 30 arcos, hasta una de cuarenta nodos y 108 arcos.

En primer lugar, se procederá a una selección de parámetros para el algoritmo genético. Estos parámetros serán *la probabilidad de cruce, la probabilidad de reemplazo y el tamaño del búfer idóneo para cada caso*.

En segundo lugar, se comparará el algoritmo genético con el algoritmo de selección de ruta mínima. Al igual que en el capítulo diez, se estudiará diversos casos:

- Solución alcanzada por parte de los dos algoritmos frente al número de conexiones. Aplicado a diversas redes.
- Solución alcanzada por parte de los dos algoritmos frente al número de enlaces. Aplicado a diversas redes.
- Solución alcanzada por parte de los dos algoritmos frente al número de conexiones y frente al número de enlaces. Aplicado a una red de diez nodos.

Un análisis de las conclusiones del proyecto viene descrito en el capítulo 13, terminando con la bibliografía utilizada para la realización del mismo, ya en el último capítulo 14.

## 2 EVOLUCIÓN DE LAS REDES DE COMUNICACIÓN

Hoy en día existen multitud de redes de comunicación basadas en diferentes filosofías y pensadas para determinados servicios. En el pasado no parecía viable el hecho de conseguir una red que pudiera soportar el amplio espectro de aplicaciones disponibles. Últimamente se están desarrollando nuevas tecnologías que allanan el camino hacia la quimera de una red integrada de servicios para telecomunicaciones.

Inicialmente, cada red de telecomunicaciones ha sido especialmente diseñada para diferentes servicios:

### 1. Red Telefónica

- Transmisión de voz
- Transmisión de datos

### 2. Redes de datos

- Transmisión de datos
- Comunicación entre ordenadores

### 3. Redes de distribución

- Transmisión de imágenes de televisión

Posteriormente, se pasa de una tecnología analógica a una tecnología digital. En 1959 se pensó en una integración de estas redes, pero no se realizó por no disponer de la tecnología necesaria y porque los usuarios no pedían la integración de servicios. En 1972 los mismos conmutadores y caminos digitales son usados para establecer las conexiones de diferentes servicios.

Estas nuevas aplicaciones requieren más recursos que las aplicaciones existentes. La aparición de nuevos servicios implica un incremento en la utilización de las redes. Todo por la necesidad de incrementar la velocidad de las redes. Las ventajas que tenían este incremento de la velocidad eran: hace que los usuarios utilicen más las aplicaciones, lo que a su vez vuelve a provocar la necesidad de incrementar la velocidad; Son necesarias velocidades del orden de 150 Mb/s para ofrecer un servicio y de 600 Mb/s para ofrecer varios simultáneamente. La realimentación entre los avances tecnológicos y los nuevos servicios provoca un cambio importante en las redes de telecomunicaciones. Las nuevas aplicaciones o servicios requieren nuevos requerimientos, diferentes entre sí, e implicaban una red universal que lo

proporcionasen. Todo esto se traducía en Red Digital de Servicios Integrados de Banda Ancha (B-ISDN). ATM empezaba a tomar cuerpo.

Hablaremos de banda ancha cuando los servicios o sistemas requieran canales capaces de soportar velocidades superiores a 2 Mb/s. Las ventajas de esta nueva idea son:

- La aparición de un servicio nuevo no implica el diseño de una nueva red. Se obtiene una disminución de los costes de implantación y de mantenimiento.
- Utilización más eficiente de los recursos de la red. Todos los recursos de la red pueden ser usados por todos los servicios.

Para la consecución de esto, es necesario diseñar un nivel de transporte y de protocolos que proporcionen un entorno en el que se puedan integrar todos los servicios actuales y los que puedan definirse en el futuro.

Como se dijo antes, uno de los objetivos principales que llevaron a comenzar la digitalización de las redes a partir de 1960 fue la posibilidad de aunar los servicios. La meta fijada era ambiciosa: conseguir incorporar cualquier tipo de tráfico, independientemente el ancho de banda o de las características de la información, en una misma red. La tasa básica de canal de 64Kbps y sus múltiplos se consideraron unidades suficientemente flexibles para satisfacer todo tipo de necesidades futuras. Sin embargo, estas premisas no pudieron mantenerse debido, fundamentalmente, a que estos canales de capacidad fija no se adaptaban frente a las distintas tasas de los servicios y tráfico intermitentes. Como consecuencia de esto, se originó una nueva tecnología derivada de la conmutación rápida de paquetes. Así surgieron las redes ATM como base para la tecnología de banda ancha.

La implantación definitiva de una red que aúne todos los servicios en el mercado se contempla en un futuro no demasiado lejano. Sin embargo, debido a que las tecnologías actuales llevan poco tiempo, corresponde a los científicos el reto y la satisfacción de encontrar soluciones a los nuevos problemas que surgen al implantar siempre una novedad en el mercado.

Actualmente ATM está todavía en estado de maduración, están por mostrarse todas las posibilidades que ofrece y todavía faltan por definir algunos estándares. Todos los organismos responsables se están esforzando para hacer realidad el futuro de los servicios integrados de banda ancha en las telecomunicaciones.

El resto del documento que se expone a continuación se centra en todo lo introducido anteriormente planteándose el estudio en las redes ATM. Más concretamente se estudiará el encaminamiento en una red basada en la tecnología ATM, buscando para ello un modelo de optimización, buscando asimismo un método de solución factible para el problema y resolviéndolo para diversas redes. Aunque la consecución de la ruta óptima a seguir por la información en redes de telecomunicaciones ha sido objeto de numerosos estudios, las redes ATM presentan características distintas que permiten un planteamiento diferente del problema, y que se exploran a lo largo del presente trabajo.

## 3 LA FIBRA ÓPTICA

### 3.1 ORIGEN Y EVOLUCIÓN

La historia de la comunicación por la fibra óptica es relativamente corta. En 1977, se instaló un sistema de prueba en Inglaterra; dos años después, se producían ya cantidades importantes de pedidos de este material.

Antes, en 1959, como derivación de los estudios en física enfocados a la óptica, se descubrió una nueva utilización de la luz, a la que se denominó rayo láser, que fue aplicado a las telecomunicaciones con el fin de que los mensajes se transmitieran a velocidades inusitadas y con amplia cobertura.

Sin embargo esta utilización del láser era muy limitada debido a que no existían los conductos y canales adecuados para hacer viajar las ondas electromagnéticas provocadas por la lluvia de fotones originados en la fuente denominada láser.

Fue entonces cuando los científicos y técnicos especializados en óptica dirigieron sus esfuerzos a la producción de un ducto o canal, conocido hoy como la fibra óptica. En 1966 surgió la propuesta de utilizar una guía óptica para la comunicación.

Esta forma de usar la luz como portadora de información se puede explicar de la siguiente manera: Se trata en realidad de una onda electromagnética de la misma naturaleza que las ondas de radio, con la única diferencia que la longitud de las ondas es del orden de micrómetros en lugar de metros o centímetros.

El concepto de las comunicaciones por ondas luminosas ha sido conocido por muchos años. Sin embargo, no fue hasta mediados de los años setenta que se publicaron los resultados del trabajo teórico. Estos indicaban que era posible confiar un haz luminoso en una fibra transparente flexible y proveer así un análogo óptico de la señalización por alambres electrónicamente.

El problema técnico que se había de resolver para el avance de la fibra óptica residía en las fibras mismas, que absorbían luz que dificultaba el proceso. Para la comunicación práctica, la fibra óptica debe transmitir señales luminosas detestables por muchos kilómetros. El vidrio ordinario tiene un haz luminoso de pocos metros. Se han desarrollado nuevos vidrios muy puros con transparencias mucho mayores que la del vidrio ordinario. Estos vidrios empezaron a producirse a principios de los setenta. Este gran avance dio ímpetu a la industria de fibras ópticas. Se usaron láseres o diodos emisores de luz como fuente luminosa en los cables de fibras ópticas. Ambos han de ser miniaturizados para componentes de sistemas fibro-ópticos, lo que ha exigido considerable labor de investigación y desarrollo. Los láseres generan luz "coherente" intensa que permanece en un camino sumamente estrecho. Los diodos emiten luz "incoherente" que ni es fuerte ni concentrada. Lo que se debe usar depende de los requisitos técnicos para diseñar el circuito de fibras ópticas dado.

### 3.2 CONCEPTOS BÁSICOS DE LA FIBRA ÓPTICA

Antes de explicar directamente que es la fibra óptica, es conveniente resaltar ciertos aspectos básicos de óptica. La luz se mueve a la velocidad de la luz en el vacío, sin embargo, cuando se propaga por cualquier otro medio, la velocidad es menor. Así, cuando la luz pasa de propagarse por un cierto medio a propagarse por otro determinado medio, su velocidad cambia, sufriendo además efectos de reflexión (la luz rebota en el cambio de medio, como la luz reflejada en los cristales) y de refracción (la luz, además de cambiar el módulo de su velocidad, cambia de dirección de propagación, por eso vemos una cuchara como doblada cuando está en un vaso de agua, la dirección de donde nos viene la luz en la parte que está al aire no es la misma que la que está metida en el agua).

Dependiendo de la velocidad con que se propague la luz en un medio o material, se le asigna un Índice de Refracción "n", un número deducido de dividir la velocidad de la luz en el vacío entre la velocidad de la luz en dicho medio. Los efectos de reflexión y refracción que se dan en la frontera entre dos medios dependen de sus Índices de Refracción. La ley más importante utilizada es la ley de Snell, como puede apreciarse en la Ecuación 1:

$$N1 * \text{sen}(\theta1) = N2 * \text{sen}(\theta2)$$

Ecuación 1

La ley de Snell nos dice que el índice de refracción del primer medio, por el seno del ángulo con el que incide la luz en el segundo medio, es igual al índice del segundo medio por el seno del ángulo con el que sale propagada la luz en el segundo medio. ¿Y esto para que sirve?, lo único que nos interesa aquí de esta ley es que dados dos medios con índices N1 y N2, si el haz de luz incide con un ángulo mayor que un cierto ángulo límite (que se determina con la anterior ecuación) el haz siempre se reflejara en la superficie de separación entre ambos medios. De esta forma se puede guiar la luz de forma controlada tal y como se ve en el dibujo de abajo (que representa de forma esquemática como es la fibra óptica).

Los circuitos de fibra óptica son filamentos de vidrio (compuestos de cristales naturales) o plástico (cristales artificiales), del espesor de un pelo (entre 10 y 300 micrones). Llevan mensajes en forma de haces de luz que realmente pasan a través de ellos de un extremo a otro, donde quiera que el filamento vaya (incluyendo curvas y esquinas) sin interrupción.

Las fibras ópticas pueden ahora usarse como los alambres de cobre convencionales, tanto en pequeños ambientes autónomos (tales como sistemas de procesamiento de datos de aviones), como en grandes redes geográficas (como los sistemas de largas líneas urbanas mantenidos por compañías telefónicas).

El principio en que se basa la transmisión de luz por la fibra es la reflexión interna total; la luz que viaja por el centro o núcleo de la fibra incide sobre la superficie externa con un ángulo mayor que el ángulo crítico, de forma que toda la luz se refleja sin pérdidas hacia el interior de la fibra. Así, la luz puede transmitirse a larga distancia reflejándose miles de veces. Para evitar pérdidas por dispersión de luz debida a impurezas de la superficie de la fibra, el núcleo de la fibra óptica está recubierto por una capa de vidrio con un índice de refracción mucho menor; las reflexiones se producen en la superficie que separa la fibra de vidrio y el recubrimiento.

### **3.3 MEDIO FÍSICO**

En los primeros tiempos de la fibra óptica, algunos puntos de vista se consideraban aventureros al asegurar que el nuevo medio de transmisión no serviría sólo para abaratar los costes de los servicios de telecomunicaciones del momento. Consideraban que la fibra tenía un gran potencial que permitiría mayores anchos de banda a más bajo coste. Estas predicciones se han cumplido y hoy la fibra óptica constituye el medio físico que soportará la red de servicios integrados de banda ancha. Algo que era inalcanzable con los medios de transmisión tradicionales.

Los sistemas de fibra óptica que fundamentalmente se aplican en la actualidad son los que enlazan las grandes centrales telefónicas. Se transmite una gran cantidad de información a largas distancias con lo que el coste por unidad de información transmitida es mucho menor que el que resulta de emplear los medios de transmisión tradicionales. Además, en el futuro se espera que cada fibra podrán transmitir muchísima más información ya que se conseguirán fuentes de luz suficientemente puras y dispositivos ópticos capaces de modularlas. No obstante, la limitación real de un sistema de comunicaciones que sustituyese totalmente los cables de cobre por cables de fibra óptica no está en el coste de la transmisión a larga distancia, sino en la transmisión desde el usuario a la central más próxima. El precio de los dispositivos transmisores y receptores de luz todavía tiene que disminuir bastante para que sea absolutamente rentable sustituir el par de cables de cobre del teléfono de cualquier abonado por un cable de fibra óptica.

En Estados Unidos y los diferentes países que componen la Europa Occidental tanto las compañías telefónicas como las de televisión por cable están estudiando la forma de introducir la fibra óptica en la red de acceso (bucle de abonado). Se pretende elegir una arquitectura que permita la implantación de la banda ancha sin tener que volver a tender otra infraestructura de cables.

### **3.4 FUNCIONAMIENTO Y CARACTERÍSTICAS ÓPTICAS**

Cada filamento consta de un núcleo central de plástico o cristal (óxido de silicio y germanio) con un alto índice de refracción, rodeado de una capa de un material similar con un índice de refracción ligeramente menor.

Los principios básicos de funcionamiento se justifican aplicando las leyes de la óptica geométrica, principalmente, la ley de la reflexión (principio de reflexión interna total) y la ley de Snell.

Su funcionamiento se basa en transmitir por el núcleo de la fibra un haz de luz, tal que este no atraviese el núcleo, sino que se refleje y se siga propagando. Esto se consigue si el índice de refracción del núcleo es mayor al índice de refracción del revestimiento, y también si el ángulo de incidencia es superior al ángulo límite.

#### **3.4.1 Ventajas**

Su ancho de banda es muy grande, hay sistemas de multiplexación que permiten enviar 32 haces de luz a una velocidad de 10Gb/s cada uno por una misma fibra, dando lugar a una velocidad total de 320Gb/s.

1. Su atenuación es muy baja.
2. Es inmune al ruido electromagnético
3. La materia prima con la que se fabrica es abundante.

#### **3.4.2 Inconvenientes**

A pesar de las ventajas antes enumeradas, la fibra óptica presenta una serie de desventajas frente a otros medios de transmisión, siendo las más relevantes las siguientes:

- La fragilidad de las fibras.
- Necesidad de usar transmisores y receptores más caros.

- Los empalmes entre fibras son difíciles de realizar, especialmente en el campo, lo que dificulta las reparaciones en caso de rotura del cable.
- No puede transmitir electricidad para alimentar repetidores intermedios.
- La necesidad de efectuar, en muchos casos, procesos de conversión eléctrica-óptica.
- La fibra óptica convencional no puede transmitir potencias elevadas.
- No existen memorias ópticas.

### 3.5 TIPOS DE FIBRA

Las diferentes trayectorias que puede seguir un haz de luz en el interior de una fibra se denominan modos de propagación. Y según el modo de propagación tendremos dos tipos de fibra óptica: multimodo y monomodo.

#### **Fibra Multimodo**

Una fibra multimodo es una fibra que puede propagar más de un modo de luz. Puede tener más de mil modos de propagación de luz. Las fibras multimodo se usan comúnmente en aplicaciones de corta distancia, menores a 1 km. Son simples de diseñar y económicas.

El núcleo de una fibra multimodo tiene un índice de refracción inferior, pero del mismo orden de magnitud, que el revestimiento (corteza). Debido al gran tamaño del núcleo de una fibra multimodo, es más fácil de conectar y tiene una mayor tolerancia a componentes de menor precisión.

Dependiendo el tipo de índice de refracción del núcleo, tenemos dos tipos de fibra multimodo:

- ***Salto de índice***: En este tipo de fibra, el núcleo tiene un índice de refracción constante en toda la sección cilíndrica.
- ***Índice gradual***: En este tipo de fibras, el índice de refracción no es constante.

## **Fibras Monomodo**

Una fibra monomodo es una fibra óptica en la que sólo se propaga un modo de luz. Se logra reduciendo el diámetro del núcleo de la fibra hasta un tamaño que sólo permite un modo de propagación. Se utiliza en aplicaciones de larga distancia, más de 300 km. Son bastante más caras.

## 4 RED DIGITAL DE SERVICIOS INTEGRADOS

### 4.1 DEFINICIÓN DE RDSI

La idea básica a tener en cuenta cuando se habla de la red Digital de Servicios Integrados es que cualquier tipo de información (voz, datos, imágenes, etc.), una vez codificado digitalmente puede ser tratado de idéntica manera, con la única diferencia de las velocidades requeridas. Una RDSI es integrada porque utiliza la misma infraestructura para muchos servicios que tradicionalmente requerían interfaces distintos (télex, voz, conmutación de circuitos, conmutación de paquetes...); es digital porque se basa en la transmisión digital, utiliza canales de 64 Kbps del MIC (G.732); y es una red porque proporciona transmisión y conmutación.

La digitalización de la red telefónica analógica ha dado lugar a la Red Digital Integrada (RDI), en la que lo único que no es digital son las líneas de acceso de los abonados (bucle de abonado).

Según la UIT-T podemos definir la Red Digital de Servicios Integrados (RDSI o ISDN en inglés) como: una red que procede por evolución de la Red Digital Integrada (RDI) y que facilita conexiones digitales extremo a extremo para proporcionar una amplia gama de servicios, tanto de voz como de otros tipos, y a la que los usuarios acceden a través de un conjunto de interfaces normalizados.

Y también (Recomendación I.120):

"Un elemento clave de la integración de servicios para una RDSI es proporcionar un abanico de servicios utilizando un conjunto limitado de tipos de conexión y disposiciones de interfaz usuario-red de propósito general".

Casi todas las redes que existen en la actualidad fueron diseñadas para servicios específicos como la telefonía, la distribución de televisión o la transferencia de datos.

Algunas aplicaciones, como el facsímil, no utilizan redes específicas (con el consiguiente ahorro en infraestructuras que esto supone), sino usan la red telefónica que está suficientemente extendida. El inconveniente de utilizar redes preexistentes y ya desplegadas es que fueron hechas a medida para su servicio, es decir, enfocaban todo su estudio en el tipo de servicio específico que ofertaban. Son redes especializadas para el servicio para el que fueron creadas, aunque las prestaciones ya no funcionan tan bien si deben soportar otro servicio. Por lo tanto, la transmisión de datos por la red telefónica no será todo lo idóneo que se quiera ya que dispondrá de una flexibilidad y anchura de banda escasa y de equipos para la transmisión de voz con calidad similar. La red telefónica fue creada para un servicio con ancho de banda constante, por lo que si se requiere su utilización para tráfico de datos con tasa de bits variable, y con cierta calidad, todo ello repercutirá en una adaptación costosa.

Debido a que, generalmente, la red telefónica básica (RTB) no se considera un soporte efectivo para otros servicios que no sean de transmisión de voz, se empezaron a crear otro tipo de redes para los distintos servicios. Las redes privadas que aparecen presentan en su mayoría interfaces y protocolos incapaces de ofrecer acceso a otras redes y usuarios. En estas situaciones son necesarios puentes si se quiere una conexión con el exterior, y su implementación resulta tediosa. La proliferación de redes singulares con interfaces propios y su posible expansión dan lugar al planteamiento de la posibilidad de una red única estandarizada. Son los inicios de la red digital de servicios integrados, también llamada RDSI.

La RDSI original constaba de una red telefónica digitalizada que se caracterizaba por tener canales de 64 Kbps (tasa que proviene de muestrear la señal de voz a 8 KHz por el teorema de Nyquist, y la posterior cuantificación con 8 bits). Esta red era básicamente una red de conmutación de circuitos, pero podía ofrecer también acceso a servicios que precisan conmutación de paquetes.

Entre las ventajas de la red digital de servicios integrados frente a las muchas de redes son: interfaz única usuario-red para el acceso a cualquier tipo de servicio, aumento de las prestaciones de la señalización, integración de los servicios en una sola red y provisión de servicios mejorados.

## 4.2 PRINCIPIOS BÁSICOS DE LA RDSI

***Soporte de aplicaciones.*** Soporta tanto de voz como de datos, utilizando un conjunto de aplicaciones estándar.

***Soporte para aplicaciones conmutadas y no conmutadas.*** RDSI admite tanto conmutación de circuitos como conmutación de paquetes. Además, RDSI proporciona servicios no conmutados con líneas dedicadas a ello.

***Dependencia de conexiones de 64 kbps.*** RDSI proporciona conexiones de conmutación de circuitos y de conmutación de paquetes a 64 kbps. Este es el bloque de construcción fundamental de la RDSI.

***Inteligencia en la red.*** Se espera que la RDSI pueda proporcionar servicios sofisticados por encima de la sencilla situación de una llamada de circuito conmutado.

***Arquitectura de protocolo en capas.*** Los protocolos para acceso a la RDSI presentan una arquitectura de capas que se puede hacer corresponder con la del modelo OSI.

**Variedad de configuraciones.** Es posible más de una configuración física para implementar RDSI. Esto permite diferencias en políticas nacionales, en el estado de la tecnología, y en las necesidades y equipos existentes de la base de clientes.

### 4.3 CANALES Y ACCESOS BÁSICOS EN LA RED DIGITAL DE SERVICIOS INTEGRADOS

Un canal es un conducto unidireccional por el que se transporta información. Puede transportar señales analógicas o digitales, y la información que lleva puede ser de datos o de señalización indistintamente. El flujo digital entre la central y el usuario RDSI se usa para llevar varios canales de comunicación. La capacidad del flujo, y por tanto el número de canales de comunicación, puede variar de un usuario a otro. Para la transferencia de información y señalización se han definido los siguientes canales:

- **Canal B.** es el canal básico de usuario. Es un canal a 64 kbps para transporte de la información generada por el terminal de usuario. Se puede usar para transferir datos digitales, voz digital codificada PCM, o una mezcla de tráfico de baja velocidad, incluyendo datos digitales y voz digitalizada descodificada a la velocidad antes mencionada de 64 kbps. Puede subdividirse en subcanales, en cuyo caso todos ellos deben establecerse entre los mismos extremos subcriptores. Puede soportar las siguientes clases de conexiones:

*Conmutación de circuitos:* Es el equivalente al servicio digital conmutado disponible en la RDI. El usuario hace una llamada y se establece una conexión de circuito conmutado con otro usuario de la red, con unos recursos dedicados. Cabe destacar que el diálogo de establecimiento de la llamada no tiene lugar en el canal B, sino en el D, que se define más tarde.

*Conmutación de paquetes:* El usuario se conecta a un nodo de conmutación de paquetes y los datos se intercambian con otros usuarios vía X.25. Los recursos no son dedicados.

*Permanentes:* No requiere un protocolo de establecimiento de llamada. Es equivalente a una línea alquilada. Se contrata un canal fijo, permanente.

- **Canal D.** Es un canal de señalización a 16 ó 64 kbps. Sirve para dos fines. Primero, lleva información de señalización para controlar las llamadas de circuitos conmutados asociadas con los canales B. Además el canal D puede usarse para conmutación de paquetes de baja velocidad mientras no haya esperando información de señalización.

- **Canal H.** Son canales destinados al transporte de flujos de información de usuario a altas velocidades, superiores a 64 kbps. En la RDSI están definidos los siguientes canales H:

- H0 velocidad 384kbps (equivalente a 6B).
- H10 velocidad 384kbps (equivalente a 23B).
- H11 velocidad 1536kbps (equivalente a 24B).
- H12 velocidad 1920 kbps (equivalente a 30B).

Todos estos canales se agrupan formando estructuras de transmisión que se ofrecen al usuario en la interfaz. Por ello, se definen dos accesos fundamentales: *acceso básico* y *acceso primario*.

#### 4.3.1 Acceso Básico

El acceso básico consiste en dos canales B full-duplex de 64 kbps y un canal D full-duplex de 16 kbps. Luego, la división en tramas, la sincronización, y otros bits adicionales dan una velocidad total a un punto de acceso básico de 192kbps (2B+D+señalización+sincronización+mantenimiento).

#### 4.3.2 Acceso Primario

El acceso primario está destinado a usuarios con requisitos de capacidad mayores, tales como oficinas con centralita (PBX) digital o red local. Debido a las diferencias en las jerarquías de transmisión digital usadas en distintos países, no es posible lograr un acuerdo en una única velocidad de los datos.

Estados Unidos, Japón y Canadá usan una estructura de transmisión basada en 1.544 Mb/s, mientras que en Europa la velocidad estándar es 2.048 Mb/s. Típicamente, la estructura para el canal de 1.544 Mb/s es 23 canales B más un canal D de 64 kbps y, para velocidades de 2.048 Mb/s, 30 canales B más un canal D de 64 kbps. A continuación, se muestran las distintas estructuras de transmisión en función de los países:

- 30B(64)+D(64)+señalización+sincronización(64) 2048 *Europa (E1)*
- 23B(64)+D(64)+señalización+sincronización(8) 1544 *Estados Unidos, Japón (T1).*

#### 4.4 RDSI-BA (RDSI-BANDA ANCHA)

Con la RDSI basada en una tasa de 64Kbps se puede ofrecer al usuario una capacidad de 1.544 o 2.048Mb/s como mucho. Sin embargo, hay aplicaciones que requieren velocidades mayores como ocurre en las conexiones de LANs o en la transmisión de imágenes en movimiento. Debido a esto, y para acabar con el problema, aparece la red digital de servicios integrados de banda ancha.

La definición de banda ancha aparece en la norma I.113 de la ITU-T como: ‘aquel servicio o sistema que requiere capacidades en los canales de transmisión que soporten tasas superiores a la tasa primaria’. Por ello, RDSI-BA además de ser equivalente a la RDSI-BE (RDSI o RDSI de banda estrecha) ofrece la posibilidad de utilizar tasas superiores de hasta cientos de megabits por segundo.

Las primeras restricciones de la RDSI-BA incluían una serie de condiciones que la red debería cumplir, entre las que destacan:

- Añadir nuevos canales de alta velocidad al espectro de canales existentes en aquel momento.
- Redefinir las nuevas interfaces entre el usuario y la red.
- Basarse en los protocolos existentes de la RDSI y modificarlos sólo en caso de que fuera absolutamente necesario hacerlo.

Como mejora de los canales de RDSI aparecen los nuevos canales H2, H3 y H4, que soportan tasas de hasta 140 Mb/s. Estas tasas de bits están orientadas a la de PDH (la jerarquía digital plesiócrona está definida por un conjunto de tasas de bits y de esquemas de multiplexación, de forma que se puedan multiplexar varios, no necesariamente síncronos, canales de RDSI-BE formando señales de tasa de bits más altas). Los canales H, sin embargo, presentan unas tasas demasiado rígidas como para ser aplicadas a todos los servicios futuros de banda ancha. Esto puso en duda la idoneidad de dichos canales como base de la RDSI-BA.

El inconveniente anterior fue resuelto cuando se pensó en la partición de la capacidad de información en ‘trozos’ denominados células. Así, cada célula podría llevar información asociado a diferentes conexiones, y por tanto, a diferentes tipos de servicios. La red ATM estaba tomando cuerpo.

#### 4.4.1 Servicios de Banda Ancha

El desarrollo de RDSI-BA sólo tiene razón de ser si surgen necesidades potenciales por parte de los clientes que no puedan ser solventadas con la red de banda estrecha, esto es, si los clientes requieren mayor banda ancha para sus aplicaciones, es decir, mayor velocidad. La banda ancha, en comparación con los medios de comunicación más comunes implica: la posibilidad de transmitir mayores volúmenes de información, que la información que se transmite sea variada (voz, vídeo, datos...). La banda estrecha permite también la transmisión combinada de varios tipos de información, pero su capacidad sólo permite la transmisión de vídeo lento de baja calidad, al menos en su configuración básica.

Una idea de la capacidad de la RDSI frente a la RDSI-BA la da la velocidad binaria que se necesita para codificar una imagen de vídeo. Con los algoritmos de compresión de la imagen de vídeo digital que se manejan actualmente, una imagen de calidad PAL (la que ofrece cualquier receptor de televisión doméstico) requeriría entre 3 y 6 Mb/s y una imagen de menos calidad, pero que diese una sensación de movimiento total bastante buena requeriría cerca de 2 Mb/s. Una aplicación basada en el acceso básico de la RDSI no permite asignar más de 64Kbps para la codificación del vídeo, es decir, como máximo se puede transmitir vídeo lento de muy baja calidad o una sucesión de imágenes fijas si la calidad que se necesita es mayor. En cambio, la RDSI-BA puede transportar servicios con tasa de bit constante y variable, datos, voz y sonido, imágenes en movimiento y aplicaciones multimedia que combinen voz, datos e imágenes.

Lógicamente, dado que la capacidad disponible en RDSI-BA es superior a la de RDSI-BE, el rango de servicios que puede soportar la primera será también superior. La ITU-T clasifica los servicios que pueden ser proporcionados por la RDSI-BA en:

***Servicios Interactivos.*** Se consideran aquellos en los cuales hay intercambio de información en los dos sentidos, ya sea entre dos usuarios o entre un usuario y un proveedor de servicio. Aquí se incluirán entonces los servicios de conversación: telefonía, videotelefonía, videoconferencia, transferencia de datos entre dos o más usuarios (ordenadores remotos...).

***Servicios de difusión.*** Son aquellos en los cuales la transferencia de información se da en un solo sentido de la comunicación: desde el proveedor de servicio al usuario. Aquí se incluyen por lo tanto la distribución de televisión, vídeo a la carta.

***Otros.*** Otra posible clasificación de los servicios de banda ancha consiste en agrupar los servicios en función de la funcionalidad de los mismos, quedando entonces dos categorías:

- Residenciales.

- De negocios

En el mundo de los negocios, los cambios que están teniendo lugar en las telecomunicaciones llevan a una continua demanda de comunicaciones de banda ancha. A ello influye el que se dé cada día más un uso creciente de aplicaciones que implican altos volúmenes de datos, incluyendo las de gráficos de alta resolución y procesamiento de imagen, así como la proliferación que están teniendo las LAN's en el mundo empresarial.

En lo que se refiere a la categoría residencial, los consumidores demandan servicios avanzados de telecomunicaciones de forma que aumenten las prestaciones telefónicas y de televisión por cable. De esta forma es posible augurar un crecimiento rápido del mercado dominado por las aplicaciones domésticas prácticas y de entretenimiento.

A continuación se describen algunos de los servicios más destacados:

#### **4.4.1.1 Videoconferencia**

Videoconferencia es la comunicación simultánea bidireccional de audio y vídeo, permitiendo mantener con grupos de personas situadas en lugares dejados entre sí. Adicionalmente, pueden ofrecerse facilidades telemáticas o de otro tipo como el intercambio de informaciones gráficas, imágenes fijas, transmisión de ficheros desde el PC, etc.

La videoconferencia proporciona importantes beneficios como el trabajo de colaboración entre personas distantes y una mayor integración entre grupos de trabajo.

A principios de los 80's utilizaron una tecnología conocida como codificación de la transformada discreta del coseno (DCT). Con estas tecnologías se analizan las imágenes de modo que si las áreas de la imagen que se parecen bastante pueden ser representadas con una misma secuencia. Con este método se logró un nivel de compresión de 60:1.

En los 90's, los equipos que se vendían eran ya, más ligeros y reducidos. Hoy en día, la razón de compresión es mayor empleada es de 1600:1 (56kps), los costes de establecimiento son insignificantes, que puede ir desde una llamada telefónica a una tarifa plana de Internet a través de llamadas IP. Las empresas involucradas en esta tecnología son muchas, como: Tanedrb, Sony, Aterra, etc.

#### **4.4.1.2 Televisión por cable**

La televisión por cable surge por la necesidad de llevar señales de televisión y radio, de índole diversa, hasta el domicilio de los abonados, sin necesidad de que éstos deban disponer de diferentes equipos, receptores, reproductores y sobre todo de antenas.

En España, las primeras redes de televisión por cable aparecieron a principios de los años 80, especialmente en la modalidad conocida como vídeo comunitario, donde en un edificio o grupo de edificios se enviaba a través del sistema de antena colectiva de televisión la señal de un reproductor de vídeo mediante el que el ‘administrador del sistema’ pasaba películas a cambio de un canon.

#### **4.4.1.3 Televisión de alta definición (HDTV)**

HDTV es el acrónimo inglés de High Definition Televisión (Televisión de alta definición). Es uno de los formatos que sumados a la televisión digital, se caracteriza por emitir las señales televisivas en una calidad superior a los demás sistemas.

Históricamente el término también fue aplicado a los estándares de televisión desarrollados en la década de 1930 para reemplazar modelos de prueba. También se aplicaba a modelos anteriores de alta definición, particularmente en Europa, llamados D2 Mac, y HD Mac, pero que no pudieron implantarse ampliamente.

#### **4.4.1.4 Servicio de pago por visión**

El pago por visión - en inglés *pay per view (PPV)*, o también conocido por pago por evento (PPE)- es una modalidad de televisión de pago, en la que el abonado paga por los eventos individuales que desea ver. Éstos pueden ser eventos deportivos, películas recién estrenadas, conciertos musicales importantes, etc. Habitualmente el sistema se comercializa como complemento a un paquete de canales de televisión que el abonado recibe continuamente de la forma tradicional, debiendo pagar, además de los eventos comprados, una cuota fija, y habitualmente un alquiler por el equipo necesario.

A diferencia de los sistemas de vídeo a la carta, la señal se transmite de forma simultánea para todos los compradores. El canal empleado puede ser tanto digital como analógico, y el usuario no recibe señal, o la recibe distorsionada en tanto que no efectúa ‘la compra’. La compra se puede realizar de forma automática, con el mando a distancia o por el propio cable si se trata de televisión por cable; también de forma manual, a través de un distribuidor o realizando una llamada telefónica.

La necesidad de decodificador supone un inconveniente cuando se desea ver distintos contenidos en varias televisiones del abonado.

El primer operador de televisión español que ofreció este servicio, bajo la marca “Taquilla”, fue canal Satélite Digital, desde el comienzo de su andadura en enero de 1997, comercializando por este sistema los partidos de liga española cuyos derechos había adquirido.

#### **4.4.1.5 Vídeo bajo demanda**

El vídeo a la carta o la televisión a la carta – en inglés video on demand (VOD) – es un sistema de televisión que permite al usuario el acceso a contenidos multimedia de forma personalizada. El usuario puede elegir en cualquier momento el programa que desea ver, sin depender de un horario fijo de programación; del mismo modo puede detener el programa y reanudarlo a voluntad. El usuario puede disponer de una oferta de programas para visualizar o realizar un pago por ciertos programas como en los sistemas de pago por visión.

El primer sistema de vídeo bajo demanda comercial fue puesto en marcha en Hong Kong, durante 1990. La tecnología no estaba madura. Ahora hay servicios de cable que permiten esta incipiente tecnología, aunque no se ha consolidado porque supone un cambio estructural muy importante en la actual industria del ocio.

En España, el sistema Imagenio envía en cada momento el programa seleccionado por el usuario sobre su línea telefónica empleando tecnologías DSL. Ono comercializa también un sistema de vídeo a la carta bajo la denominación Ojo.

#### **4.4.1.6 Radio a la carta**

Es un innovador servicio que permite escuchar desde Internet, en cualquier momento y con la mejor calidad de sonido, los programas emitidos en RADIOMEDIACIÓN, a través de conexiones de banda ancha (ADSL y Cable).

Una oferta de audio a la carta (Audio on Demand) a tu alcance y que aumentará día a día.

Con radio a la carta, RADIOMEDIACIÓN elimina las distancias geográficas y temporales y consolida una nueva manera de acceder a la radio, desde cualquier punto del mundo y sin limitaciones horarias. El usuario puede convertirse en un usuario activo que puede diseñar su programación a medida.

Radio a la carta es un primer paso, una iniciativa impulsada por Solomediación.com con el objetivo de poner al alcance de todo el mundo el trabajo que desarrolla que otras personas y entidades puedan compartir su actividad en torno a la mediación con personas de todo el mundo.



## 5 JERARQUÍA DE TRANSMISIÓN SÍNCRONA

### 5.1 SDH

SDH y el equivalente norteamericano SONET son las tecnologías dominantes en la capa física de transporte de las actuales redes de fibra óptica de banda ancha. Su misión es transportar y gestionar gran cantidad de tipos de tráfico diferentes sobre la infraestructura física.

Esencialmente, SDH es un protocolo de transporte (primera capa en el modelo OSI) basado en la existencia de una referencia temporal común (Reloj primario), que multiplexa diferentes señales dentro de una jerarquía común flexible, y gestiona su transmisión de forma eficiente a través de fibra óptica, con mecanismos internos de protección.

Usando como referencia el modelo OSI, SDH es comúnmente visto como un protocolo de nivel uno, es decir, un protocolo de la capa física de transporte. En este papel, actúa como el portador físico de aplicaciones de nivel 2 a 4, esto es, es el camino en el cual tráfico de superiores niveles tales como IP o ATM es transportado. En palabras simples, podemos considerar a las transmisiones SDH como tuberías las cuales portan tráfico en forma de paquetes de información. Estos paquetes son de aplicaciones tales como PDH, ATM o IP.

SDH permite el transporte de muchos tipos de tráfico tales como voz, video, multimedia, y paquetes de datos como los que genera IP. Para ello, su papel es, esencialmente, el mismo: gestionar la utilización de la infraestructura de fibra. Esto significa gestionar el ancho de banda eficientemente mientras porta varios tipos de tráfico, detectar fallos y recuperar de ellos la transmisión de forma transparente para las capas superiores.

Las principales características que encontramos en cualquier sistema de red de transporte SDH implementado a día de hoy son las siguientes:

- ***Multiplexión digital***: Éste término fue introducido hace 20 años y permitió que las señales de comunicaciones analógicas sean portadas en formato digital sobre la red. El tráfico digital puede ser portado mucho más eficientemente y permite monitorización de errores, para propósitos de calidad.
- ***Fibra óptica***: Éste es el medio físico comúnmente desplegado en las redes de transporte actuales. Tiene una mayor capacidad de portar tráfico que los coaxiales o los pares de cobre lo que conduce a una disminución de los costes asociados al transporte de tráfico.

- **Esquemas de protección:** Éstos han sido estandarizados para asegurar la disponibilidad del tráfico. Si ocurriera una falla o una rotura de fibra, el tráfico podría ser conmutado a una ruta alternativa, de modo que el usuario final no sufriera interrupción alguna en el servicio.
- **Topologías en anillo:** Éstas están siendo desplegadas cada vez en mayor número. Esto es porque, si un enlace se perdiera, hay un camino de tráfico alternativo por el otro lado del anillo. Los operadores pueden minimizar el número de enlaces y fibra óptica desplegada en la red. Esto es muy importante ya que el coste de colocar nuevos cables de fibra óptica sobre el terreno es muy caro.
- **Gestión de red:** La gestión de estas redes desde un único lugar remoto es una prestación importante para los operadores. Se ha desarrollado software que permite gestionar todos los nodos y caminos de tráfico desde un único computador. Un operador puede ahora gestionar una variedad grande de funciones tales como el provisionamiento de capacidad en respuesta a la demanda de clientes y la monitorización de la calidad de una red.
- **Sincronización:** Operadores de red deben proporcionar temporización sincronizada a todos los elementos de la red para asegurarse que la información que pasa de un nodo a otro no se pierda. La sincronización es de creciente concierto entre los operadores, con avances tecnológicos cada vez más sensibles al tiempo. La sincronización se está convirtiendo en un punto crítico, proveyendo a SDH un camino ideal de filosofía de red.

## 5.2 SONET

SONET define una tecnología para transportar muchas señales de diferentes capacidades a través de una jerarquía óptica síncrona y flexible. Esto se logra por medio de un esquema de multiplexado por interpolación de bytes. La interpolación de bytes simplifica la multiplexación y ofrece una administración de la red extremo a extremo. El primer paso en el proceso de la multiplexación de SONET implica la generación de las señales del nivel inferior de la estructura de multiplexación. En SONET la señal básica la conocemos como señal de nivel 1 o también STS-1 (Synchronous Transport Signal level 1). Está formada por un conjunto de 810 bytes distribuidos en 9 filas de 90 bytes. Este conjunto es transmitido cada 125 microsegundos, correspondientes a la velocidad del canal telefónico básico de 64 Kbps, por lo que la velocidad binaria de la señal STS-1 es 51,84 Mb/s.

### 5.3 JERARQUÍA DE SEÑALES

La especificación de SONET define una jerarquía de tasas estandarizadas de transmisión datos. El nivel más bajo, que se conoce como STS-1 es de 51.84 Mb/s. Esta tasa puede ser utilizada para transportar un único DS-3, o un conjunto de señales de tasa menor, como DS1, DS2... además de las tasas definidas por la ITU-T para PDH (e.g. 2048 Mb/s).

Para SDH, la tasa de transmisión más baja es de 155.52 Mb/s y se designa con las siglas STM-1. Esta tasa se corresponde con lo que en el estándar americano sería un STS-3. La razón de ésta discrepancia entre estándares es que el STM-1 es la tasa de transmisión más baja, que sin dejar de ser compatible con el estándar americano, puede acomodar una señal del nivel cuatro del estándar de ITU-T (139.264 Mb/s).

La gran aportación de este estándar es la creación de una red de transporte compatible con las actuales redes plesiócronicas y con las futuras arquitecturas de red como la RDSI-BA. En la Tabla 1 se puede apreciar la jerarquía de señales:

**Tabla 1. Tabla de la jerarquía de señales SDH/SONET**

<b>Estándar SONET</b>	<b>Estándar SDH</b>	<b>Tasa de línea (Mbps)</b>
STS-1		51.84
STS-3	STM-1	155.52
STS-9	STM-3	466.56
STS-12	STM-4	622.08
STS-18	STM-6	933.12
STS-24	STM-8	1244.16
STS-36	STM-12	186.24
STS-48	STM-16	2488.32

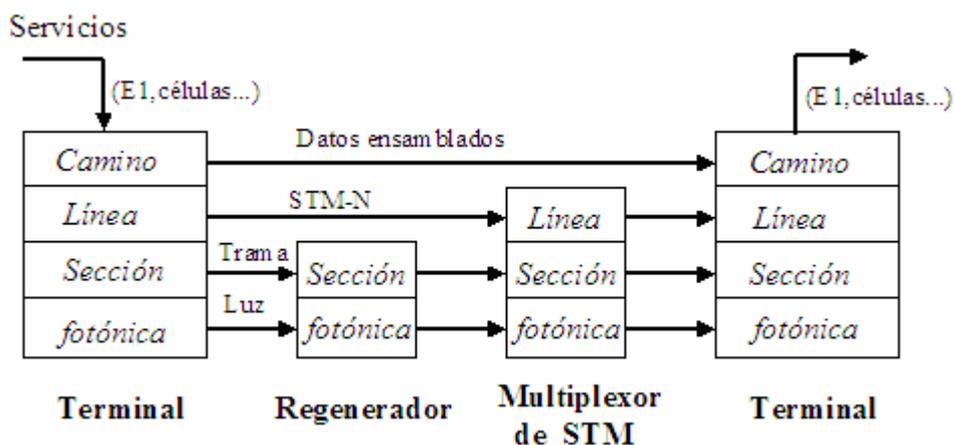
### 5.4 JERARQUÍA DEL SISTEMA

El sistema responde a una estructura de cuatro capas: la capa de trayecto, la capa de línea, la capa de sección y la capa fotónica. Véase la Tabla 2:

**Tabla 2. Tabla representativa de las distintas capas del sistema**

Descripción /Capas	Descripción general
<b>Capa fotónica</b>	Es la capa física. Incluye especificaciones sobre el tipo de fibra que debe ser utilizada, y detalles sobre los emisores ópticos como la potencia mínima requerida y la característica de dispersión, así como la sensibilidad necesaria en el receptor.
<b>Capa de sección</b>	En esta capa se crean las tramas básicas SDH. Se convierten las señales eléctricas en fotónicas. También se tienen funciones de monitorización.
<b>Capa de línea</b>	Es la responsable de la sincronización, de la multiplexación de datos en las tramas SDH, así como de la protección y de las funciones de mantenimiento y de la conmutación
<b>Capa de camino</b>	Es la capa responsable del transporte extremo a extremo de los datos a la velocidad apropiada.

No todos los equipos necesitan implementar las cuatro capas. En un repetidor sólo son necesarias la capa de sección y la fotónica, y en los terminales que no extraen o insertan señales no es necesaria la capa de trayecto. Esto se aprecia en la Figura 1.



**Figura 1. Jerarquía SDH**

Para distancias cortas, el cable puede conectarse directamente entre dos unidades terminales, pero en el caso de distancias más largas hacen falta unidades de regeneración. Los regeneradores o repetidores aceptarán la señal digital de datos que le llega por la entrada, la regenera y la repite uno a uno los bits regenerados por la salida. Los multiplexores y conmutadores pueden añadir y sacar canales de la línea. Por último el terminal receptor recoge la señal cerrando así el circuito. Los datos que se envían se ensamblan en la capa de camino y no se desensamblan hasta que llegan a la capa de transporte del destino.

El SDH o SONET se está introduciendo muy rápidamente debido a la agilidad en los procesos de estandarización. Existen tres tendencias de introducción de SDH/SONET, como aparecen en la Tabla 3:

**Tabla 3. Representación de las tendencias de introducción de SDH/SONET**

<b>SDH/SONET / Tendencias</b>	<b>Descripción general</b>
<b>De abajo a arriba</b>	Influida por la demanda de nuevos servicios en la zona del bucle del abonado
<b>De arriba a abajo</b>	Cubre la demanda de mayor capacidad de transmisión, conmutación de protección y flexibilidad.
<b>Mixta</b>	Es la tendencia que actúa a varios niveles simultáneamente. La ejercen los operadores que no tienen infraestructura previa y pretenden ofrecer servicios en competencia



## 6 MODO DE TRANSFERENCIA ASÍNCRONO

### 6.1 ANTECEDENTES DE ATM

Tres letras - ATM - se repiten cada vez más en estos días en los ambientes Informáticos y de Telecomunicaciones. La tecnología llamada Asynchronous Transfer Mode (ATM) Modo de Transferencia Asíncrona es el corazón de los servicios digitales integrados que ofrecerán las nuevas redes digitales de servicios integrados de Banda Ancha (B-ISDN), para muchos ya no hay cuestionamientos; el llamado tráfico del "Cyber espacio", con su voluminoso y tumultuoso crecimiento, impone a los operadores de redes públicas y privadas una voraz demanda de anchos de banda mayores y flexibles con soluciones robustas. La versatilidad de la conmutación de paquetes de longitud fija, denominadas celdas ATM, son las tablas más calificadas para soportar la cresta de esta "Ciberola" donde los 'surfeadores' de la banda ancha navegan.

Algunos críticos establecen una analogía de la tecnología ATM con la red digital de servicios integrados o ISDN por sus siglas en inglés. Al respecto se escuchan respuestas de expertos que desautorizan esta comparación aduciendo que la ISDN es una gran tecnología que llegó en una época equivocada, en términos de que el mercado estaba principalmente en manos de actores con posiciones monopolísticas.

Se dice en todos los medios que ATM será la solución tecnológica que permitirá el desarrollo de las autopistas de la información pero en la práctica también es posible transmitir con la misma velocidad con tecnología síncrona. ATM sin embargo ofrece un mecanismo importante de control de información, o lo que es lo mismo un mecanismo de tarificación de información y el objetivo de las autopistas de la información es proveer un servicio sujeto al pago.

### 6.2 COMPARATIVA ATM Y STM

Asynchronous Transfer Mode. Tecnología de transmisión digital que permite una asignación estadística de canales en función del tráfico existente en una red.

El concepto de ATM es más fácil de entender si se comprende STM. STM (Synchronous Transfer Mode) es una tecnología que permite la multiplexación varios canales de información asignando a cada uno de ellos un intervalo de tiempo concreto en una trama. En una trama STM constituida por N canales, (digamos 32) se reserva un intervalo de tiempo para que transmita el canal 1, luego el canal 2,... el canal 31, el canal 32, y otra vez el canal 1... De esta forma la trama STM ha de fluir 32 veces más rápido que uno de los canales individuales.

ATM permite aprovechar mas eficientemente el ancho de banda al realizar siempre una ocupación completa de canal, y adaptándose dinámicamente a las necesidades de tráfico.

La desventaja de ATM es que ha de existir una lógica de control adicional capaz de gestionar el tráfico, que puede ser incluso contraproducente para aplicaciones en tiempo real. Mientras que en STM tenemos nuestra velocidad de canal asegurada, en ATM se corre el riesgo de perder el flujo constante de información.

### **6.3 TIPOS DE CONMUTACIÓN**

La comunicación entre un origen y un destino habitualmente pasa por nodos intermedios que se encargan de encauzar el tráfico. Por ejemplo, en las llamadas telefónicas los nodos intermedios son las centralitas telefónicas y en las conexiones a Internet, los routers o encaminadores. Dependiendo de la utilización de estos nodos intermedios, se distingue entre conmutación de circuitos, de mensajes y de paquetes.

#### **6.3.1 Conmutación de circuitos**

En la conmutación de circuitos se establece un camino físico entre el origen y el destino durante el tiempo que dure la transmisión de datos. Este camino es exclusivo para los dos extremos de la comunicación: no se comparte con otros usuarios (ancho de banda fijo). Si no se transmiten datos o se transmiten pocos se estará desperdiciando el canal. Las comunicaciones a través de líneas telefónicas analógicas (RTB) o digitales (RDSI) funcionan mediante conmutación de circuitos.

Normalmente la capacidad total se reparte con TDM (multiplexación por división en el tiempo), técnica que asigna a cada conexión la totalidad del ancho de banda cada cierto tiempo, se dice entonces que a un par de nodos le corresponde una rendija de tiempo en la que no podrá transmitir más que el par perteneciente a esa conexión (quedando vacío el enlace si no tienen nada que transmitir).

El camino elegido entre el origen y el destino probablemente atravesará varios conmutadores y utilizará distintos enlaces a través de la red. El encaminamiento utilizado se llama 'fijado en el origen', ya una vez que se decide la ruta a seguir por toda la información desde el nodo origen ya no varía. Las redes basadas en conmutación de circuitos presentan algunas características ventajosas derivadas de esta forma de encaminar, como puede ser la coherencia en la recepción de información. Es decir, dado que todas las unidades de información siguen el mismo trayecto entre origen y destino, no es posible que una unidad que se transmitió con anterioridad llegue más tarde que otra que fue transmitida con posterioridad. Si por el contrario siguieran distintas rutas existiría la posibilidad de que esto ocurriese, ya que una unidad de información podría ser enviada por un enlace más congestionado que el que toma la siguiente unidad transmitida, tardando así más tiempo en alcanzar el destino, con lo que llegarían

desordenadas. Incluso puede darse el caso de que caiga algún enlace, con lo que las unidades que estuvieran circulando por un camino que contuviese ese enlace se perderían, mientras que las que fueran por otros caminos llegarían al destino. Este fenómeno ocurre en la conmutación de paquetes.

### **6.3.2 Conmutación de paquetes**

Finalmente, la conmutación de paquetes es la que realmente se utiliza cuando hablamos de redes. Los mensajes se fragmentan en paquetes y cada uno de ellos se envía de forma independiente desde el origen al destino. De esta manera, los nodos (encaminadores) no necesitan una gran memoria temporal y el tráfico por la red es más fluido. Nos encontramos aquí con una serie de problemas añadidos: la pérdida de un paquete provocará que se descarte el mensaje completo; además, como los paquetes pueden seguir rutas distintas puede darse el caso de que lleguen desordenados al destino. Esta es la forma de transmisión que se utiliza en Internet: los fragmentos de un mensaje van pasando a través de distintas redes hasta llegar al destino.

Los paquetes llevan una etiqueta que los identifica como parte de una conexión, de esta forma es posible multiplexar distintos paquetes que pertenecen a distintas conexiones por una misma línea sin necesidad de dedicar canales como ocurría en la conmutación de circuitos. De esta forma, se ahorra capacidad en la red. Esta multiplexación se llama multiplexación etiquetada. Además de la etiqueta de identificación, los paquetes controlan sus errores gracias a que llevan un código de corrección de error. Esto permite que los paquetes erróneos sean retransmitidos. Así se puede evitar que la calidad en la transmisión en este tipo de redes sea demasiado baja.

Tanto la etiqueta como el código de control son campos de longitud fija comunes a todos los paquetes. A este conjunto de campos se les llama cabecera. Huelga decir que la existencia de la cabecera como campo fijo a transmitir en todos los paquetes reduce la tasa de bits disponible para la transmisión de información de usuario.

Normalmente, para elegir el camino hacia un destino, el nodo origen envía cada paquete a un conmutador y éste se encargará de enviarlo a otro, y éste a otro, así hasta alcanzar el destino. Esta forma de encaminamiento se llama 'salto a salto', ya que en cada nodo se decide cual será el siguiente nodo de la ruta, o el siguiente salto. Esta forma de encaminar también presenta sus ventajas, de hecho, la posibilidad de variar la ruta en función de la variación del estado de la red (p.e la variación de tráfico por los enlaces) puede proporcionar una mayor velocidad a la transmisión.

Otra característica de las redes de conmutación de paquete es que la red siempre puede continuar aceptando paquetes sin asegurarse de que vaya a poder trabajar aceptablemente con el tráfico extra. Es decir, un nodo puede transmitir siempre que la línea esté libre los paquetes tan grandes como quiera (dentro del límite de tamaño de los mismos). Esta forma de trabajar ocasionará grandes retrasos que se verán incrementados a medida que la carga de la red aumenta. Cuando la carga se acerca a la capacidad de la red, las colas en los conmutadores se llenan con lo que los paquetes que llegan se pierden, esto origina la retransmisión de los mismos por parte de los nodos origen, con

lo que la situación se empeora cada vez más yendo en detrimento del rendimiento de la red. Para evitar llegar a estas situaciones extremas se usan controles de congestión que limitan la transmisión de paquetes por parte de los nodos, utilizándose cuando la red se carga demasiado. Habrá, por lo tanto, en la cabecera de los paquetes un campo relacionado con el control de congestión.

#### 6.4 DESCRIPCIÓN DE LA CÉLULA ATM

ATM combina las técnicas de conmutación de circuitos y de paquetes de forma muy simple; mantiene la estructura de canales (reflejada en la división por ranuras de tiempo) de la conmutación de circuitos y mantiene la estructura de paquetes definida en conmutación de paquetes. Sin embargo, a pesar de que toma los conceptos de ‘canal’ y de ‘paquete’ de las técnicas anteriores, hace variaciones sobre los mismos. De esta forma, en ATM un paquete no sólo es una agrupación de octetos en la que una parte es fija, (cabecera) y otra variable (datos útiles de información de usuario), sino que añade una longitud fija a los paquetes. De igual forma, los canales se corresponden con ranuras pertenecientes a usuarios. Estos, podrán utilizar la capacidad del enlace durante el intervalo de tiempo que les corresponda, al igual que antes, pero en ATM se añade la característica de que estas rendijas no tienen posición fija dentro de una trama, por lo que no todos los usuarios tendrán canales de la misma capacidad asignados.

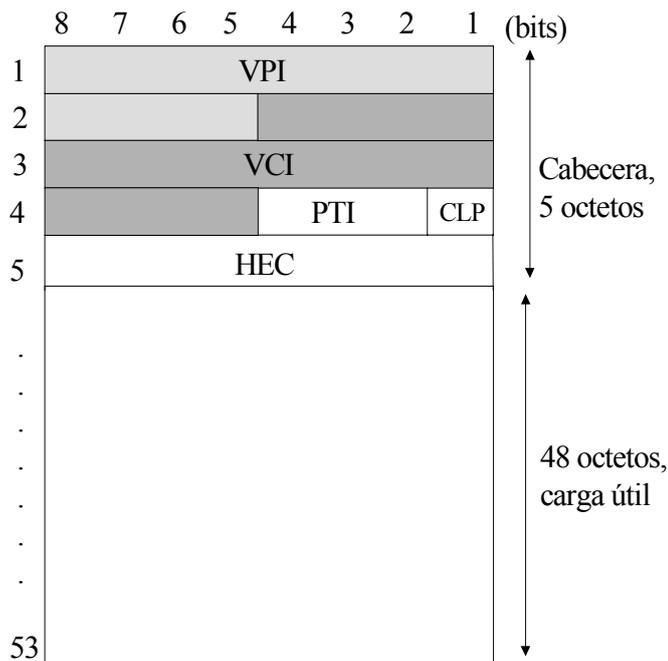
Son estructuras de datos de 53 bytes compuestas por dos campos principales. Se muestra en la Tabla 4.

**Tabla 4. Representación de los dos campos principales de una célula ATM**

<b>Descripción / Dos campos principales</b>	<b>Descripción general</b>
<b>Header (cabecera)</b>	Sus 5 bytes tienen tres funciones principales: identificación del canal, información para la detección de errores y si la célula es o no utilizada. Eventualmente puede contener también corrección de errores, número de secuencia.
<b>Payload (datos útiles)</b>	Tiene 48 bytes fundamentalmente con datos del usuario y protocolos AAL que también son considerados como datos del usuario

Dos de los conceptos más significativos del ATM, Canales Virtuales y Rutas Virtuales, están materializados en dos identificadores en el header de cada célula (VCI y VPI) ambos determinan el encaminamiento entre nodos. Existen dos formatos de células: la UNI (User Network Interface) utilizado en el interfaz red/usuario y la NNI (Network Interface) cuando circulan por la red.

La estructura de la célula se puede ver en la Figura 2:



**Figura 2. Célula ATM**

La Figura 2 muestra los distintos campos existentes en la cabecera de la célula, la descripción de los mismos es la que se explica a continuación.

## 6.5 COMPOSICIÓN DE UNA CÉLULA ATM

En los siguientes apartados, se va a proceder a la descripción de cada una de las partes que componen una célula ATM.

### 6.5.1 Campo de ruta VPI/VCI

VPI y VCI son las siglas inglesas correspondientes a Virtual Path Identifier, Virtual Channel Identifier (identificador de camino virtual y de canal virtual respectivamente).

El primero indica la ruta que deberá seguir la célula para llegar al destino, identifica el camino virtual asignado a la conexión. El segundo marca el canal virtual en el que debe transmitirse la célula según al servicio que pertenezca. Todas las células de un mismo servicio se transportan dentro del mismo VCI. Dentro de un camino virtual pueden existir varios canales virtuales. Con los dos indicadores quedará perfectamente clara la ruta de una conexión. Pudiéndose soportar teóricamente 4.096 caminos virtuales y 65536 canales virtuales.

Hay que hacer una puntualización a la figura anterior, ya que corresponde a una célula que fluye por el NNI (interfaz red- red, Network-Network interface), es decir entre los conmutadores de la red. En caso de que fuera una célula del interfaz UNI (interfaz usuario- red, User-Network Interface) el campo VPI sería menor (pasaría a ser de 8 bits) porque los cuatro primeros bits de la cabecera formarían otro campo llamado GFC, control de flujo genérico (Generic Flow Control en inglés), que permite establecer control sobre el flujo que se emite en el interfaz UNI.

### **6.5.2 PTI. Identificador de carga útil**

Los tres bits que forman el campo PTI (Payload Type Identifier) permitirán identificar el tipo de carga que lleva la célula: tráfico normal o tráfico de control (señalización u operación y mantenimiento de la red).

### **6.5.3 CLP. Identificación de prioridad de pérdida de células**

Este campo puede tomar los valores 0 o 1. Con ello, se indica una prioridad mayor o menor en la célula para ser desechada en cualquier momento por la red. Así, se está asegurando los recursos para mantener una buena comunicación, ya que la red marcará aquellas células que incumplan el contrato o que se puedan perder causando un menor perjuicio a los servicios y serán éstas las células que se pierdan en caso de que la sobrecarga de la red así lo requiera. De este modo se protege al resto de las células.

#### **6.5.3.1 HEC. Detección y corrección de errores en la cabecera.**

El nombre del campo viene de las siglas inglesas de Header Error Detection. Su función es la de detectar y corregir los posibles errores que puedan producirse en la transmisión de cualquier campo de la cabecera de la célula. La información que transporta la cabecera es de suma importancia, ya que de ella depende el que las células se entreguen en el lugar de destino correcto y que se las maneje según la prioridad que tengan asignada. Esa es la razón de que se dedique un octeto completo sólo a la detección de errores en la cabecera.

## 6.6 CANALES VIRTUALES Y CAMINOS VIRTUALES

Las células generadas en la UNI por distintos servicios se transmiten a través de un único enlace al siguiente nodo de la red. Todas las células son distribuidas dentro de este único enlace en distintos canales virtuales (VC). Cada canal virtual contiene las células de un servicio. De esta forma distintas conexiones con distintas características se transmiten por un enlace físico. Varios canales virtuales agrupados forman un camino virtual (VP), se puede ver como un conjunto de VC en paralelo. El canal virtual es unidireccional y está identificado por su VCI y por el VPI del camino virtual que lo soporta. Todas las células de una misma conexión tienen el mismo VPI/VCI que se mantiene constante hasta llegar a algún nodo de conmutación. De esta forma se pueden establecer rutas lógicas entre dos nodos cualesquiera. Se llama conexión de canal virtual (VCC) a la concatenación de una serie de canales virtuales entre el transmisor de la información y el receptor de la misma.

Como se mencionó anteriormente, tanto el VPI como el VCI se transmiten en la cabecera de la célula. Estos identificadores no tienen significado extremo a extremo, esto es, su significado se limita a cada enlace concreto de la conexión; identifican un canal lógico dentro de un enlace físico. Cuando la célula llega a un elemento de conmutación se busca el VPI/VCI en la tabla de encaminamiento y se envía la célula por el puerto de salida que le corresponda, cambiando el valor del identificador que fuera necesario. Las tablas de encaminamiento se van creando a medida que se van estableciendo conexiones y permanecen constantes hasta que éstas se cierran.

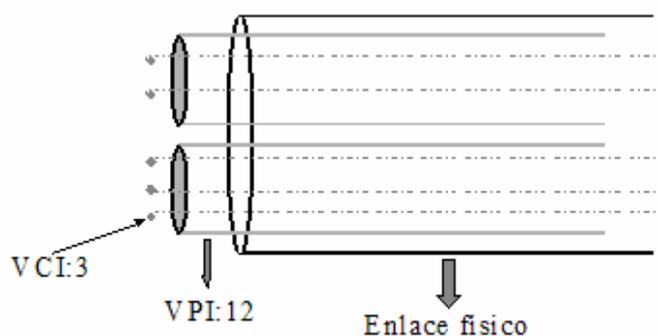


Figura 3. Camino Virtual y canal virtual

Cuando se habla de VC en ATM debe quedar claro que se trata de una abstracción mediante la cual se designa un transporte unidireccional de células asociado a un identificador: el VCI. Tanto el VCI como el VPI indicarán por lo tanto, un flujo ordenado de células asociado a una conexión concreta. El concepto de VCC puede parecer sin embargo un poco más complejo. Para facilitar su comprensión se incluye una figura en la que se puede apreciar claramente una conexión de caminos virtuales.

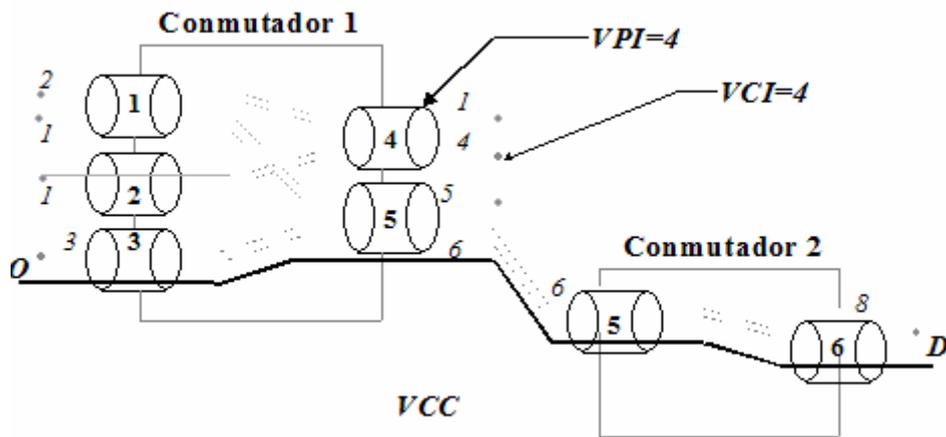


Figura 4. Concatenación de caminos virtuales

### 6.6.1 Canales y caminos virtuales. STM

Desde el punto de vista de transmisión se denomina STDM (multiplexación por división en el tiempo síncrona). En este modo de transferencia se asigna a cada usuario una ranura de tiempo periódica. Todas las ranuras son de igual duración. Así, se forman tramas con ranuras de repetición periódica y queda la capacidad del canal dividida en canales de tamaño constante  $C/n$ . Donde  $C$  es la capacidad total del canal y  $n$  es el número de usuarios que lo comparten.

La denominación de ‘síncrono’ para este modo de transferencia se debe a que la información de un determinado usuario se encuentra periódicamente en el canal. El usuario tiene reservada una cierta ranura en el canal cada periodo de tiempo  $T$ . Si no tuviese nada que transmitir en el intervalo que tiene designado, la ranura quedaría vacía.

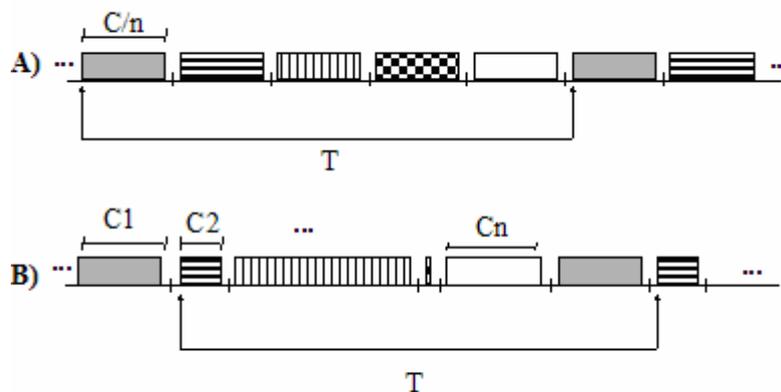


Figura 5. Trama STM (A) y STM multicanal(B)

En la Figura 5, se ha incluido un esquema de STM multicanal. Es una variación de STM en la que los canales no tienen la misma capacidad. Como ventaja presenta la posibilidad de multiplexar tráfico que requieran distintos anchos de banda. Como inconvenientes, además del desaprovechamiento de capacidad cuando un usuario no transmite, presenta unos requerimientos particulares en la conmutación, ya que las distintas capacidades requieren velocidades de conmutación diferentes.

### 6.6.2 Canales y caminos virtuales. ATM

En ATM se asignan las capacidades requeridas por los servicios de forma dinámica mediante paquetes de longitud fija. La multiplexión por división en el tiempo es asíncrona. Esto es, los usuarios no van a tener una ranura de tiempo asignada cada periodo de tiempo. Ahora, la forma de determinar al usuario en el canal no es gracias a la identificación de la ranura de tiempo que le corresponde sino gracias a la cabecera de las células. Así, un usuario no aparece de forma síncrona en el canal sino que lo hará en función de la capacidad que requiera dependiendo del tipo de tráfico que lleve. Por esto se le da la denominación de ‘asíncrono’ a este modo de transferencia. No debe pensarse que la transmisión es asíncrona, sólo lo es la multiplexión de la información. Como los usuarios no tienen reservadas ranuras fijas, ya no se desaprovecha tanto la capacidad del canal, y debido a que el tamaño asignado a cada usuario no es constante, se puede multiplexar tráfico de distintos tipos de fuentes.

Como se ha visto, STM y ATM se rigen por diferentes reglas a la hora de asignar la capacidad utilizable en los canales. Mientras que STM asigna a cada canal una ranura de tiempo periódica, en ATM la capacidad utilizable (ancho de banda) se segmenta en unidades de longitud fija orientadas a la información (células). La estructura de un interfaz ATM consiste en un grupo de canales etiquetados, pero sin una posición fija. Dichos canales no están limitados a un pequeño número de valores fijos de tasa de bit (como ocurre en STM multicanal). Los servicios se mezclan y las tasas de transferencia de la información se separan en el conmutador. No se requiere una asignación de ranuras de tiempo a los canales ante un establecimiento de llamada.

En comparación con los equipos para STM, los multiplexadores y conmutadores ATM son menos dependientes de las tasas de bits de los diferentes servicios. Los equipos ATM pueden soportar gran variedad de servicios con diferentes tasas de transferencia sin las complicaciones asociadas con la demultiplexación de canales con una posición predeterminada y fija dentro de una trama. Todo ello hace innecesarios los circuitos dependientes de la tasa de bits, lo que permite una mayor integración de la red.

Por su propia filosofía, el modo de transferencia síncrono conlleva una cierta rigidez. Se basa en transmitir la información en tramas periódicas. Dentro de cada trama un canal tiene su situación específica y una tasa de bits concreta. Todo ello limita sus posibilidades como estructura base en la red de banda ancha donde, lo que se pretende, es transmitir cualquier tipo de información (voz, datos, vídeo) a cualquier velocidad. Esto explica que se descartara STM dejando paso a ATM como el modo de transferencia estándar en la RDSI-BA.

## **6.7 PROCESO DE CONMUTACIÓN EN ATM**

Un conmutador es un elemento de la red cuya función primordial es trasladar la información de un canal lógico entrante al canal lógico saliente que le corresponda. Es importante seleccionar correctamente el canal de salida para obtener un buen funcionamiento de la red. Este procedimiento de elección de ruta de salida se conoce como enrutado.

Las células ATM portan un campo destinado a la corrección de errores singulares que se producen sobre la información de la cabecera (campo HEC). Sin embargo, no hay ningún tipo de protección para los datos de usuario, y si fuese necesaria una retransmisión, ésta se produciría extremo a extremo. De ahí la importancia que tiene en los conmutadores ATM poder proporcionar a las células recibidas un lugar donde puedan esperar hasta ser reenviadas. El proceso de almacenamiento temporal de células en los búferes de tamaño finito del conmutador se llama encolado.

El funcionamiento de los conmutadores ATM se caracteriza por el volumen de información que pueden procesar, la tasa de bits erróneos, el retardo de conmutación, la probabilidad de células perdidas y la fluctuación en el retardo.

En los conmutadores ATM, el volumen de información que se puede procesar y la tasa de bits erróneos los determinan principalmente la tecnología y las dimensiones del sistema. El uso de tecnologías de alta velocidad como BICMOS o ECL proporcionan fácilmente cientos de Mb/s con tasas de error aceptables. A continuación se definen los parámetros que merecen especial atención en el caso de los conmutadores ATM.

### **6.7.1 Probabilidad de pérdida de células**

En los conmutadores ATM La pérdida de una célula se produce cuando muchas células son destinadas al mismo enlace, dándose la situación de que el número de células es superior al de estados libres en la cola. Siendo imposible almacenarlas todas. La probabilidad de pérdida debe mantenerse dentro de unos límites, valores típicos los sitúan entre  $10^{-8}$  y  $10^{-11}$ .

### **6.7.2 Retardo de conmutación**

Es el tiempo que tarda una célula en atravesar el conmutador hacia su destino. Valores típicos varían entre 10 y  $1000\mu\text{s}$  con una oscilación de  $100\mu\text{s}$  aproximadamente.

### 6.7.3 Tipos de conmutadores en ATM

La conmutación en ATM puede ser realizada a distintos niveles, ya que normalmente se tienen redes jerárquicas. Sin embargo, la arquitectura de los conmutadores, independientemente del nivel jerárquico en el que se encuentre trabajando el conmutador, es la misma.

Los conmutadores pueden ser clasificados en función de la arquitectura de la fábrica de conmutación.

- **Matricial**

Como su nombre indica, la red de interconexión interna de este tipo de conmutadores se forma gracias a una matriz de interconexión. De esta forma se puede conectar cualquier entrada con cualquier salida.

- **Medio compartido(bus, anillo)**

En el caso del bus todas las células se transportan en paralelo por el mismo medio. Cada puerto de entrada podrá transferir su célula hacia el destino correspondiente antes de que llegue la siguiente célula, por lo que no harían falta buffers a la entrada de este tipo de conmutadores. No obstante, no sucede lo mismo en los puertos de salida, ya que existe la posibilidad de que a un mismo puerto vayan dirigidas varias células. Sí se requieren de este modo colas a la salida.

- **De memoria central**

Se basa en la existencia de una memoria central compartida por todos los puertos de entrada y todos los de salida. Cualquier puerto de entrada puede escribir en la memoria que puede ser leída por cualquiera de los puertos de salida.

*Fábrica de conmutación* es la traducción española del término switching fabric, con el que se denomina, en tecnología ATM, a la parte del conmutador que se encarga del encaminamiento de las células desde su punto de entrada al de la salida.

### 6.7.3.1 Tipos de conmutadores matriciales

Los conmutadores con matriz de interconexión (matriciales) pueden a su vez clasificarse en función de la ubicación de la memoria de almacenamiento de células. A continuación, se presentan los diversos tipos.

#### 6.7.3.1.1 Colas a la entrada del conmutador

Las células se almacenan en la entrada evitando así que se produzcan pérdidas en esta parte del conmutador. A la hora de ser servidas, podrán serlo aleatoriamente o por turnos definidos anteriormente (FIFO, LIFO...). El mayor problema que presentan las colas a la entrada es el denominado en la bibliografía como bloqueo en la cabecera de la línea (traducción del término inglés Head of line blocking). En este fenómeno, debido a que a la salida no hay colas, el conmutador podrá enviar más de una célula a una determinada salida y en caso de que muchas células quieran ir a la misma salida, y como no pueden ser enviadas más que de una en una, deberán ser retenidas a la entrada bloqueando a otras células que, aunque comparten la cola con ellas, se dirigen a una salida libre de congestión. De esta forma no sólo sufren retraso las células dirigidas a la salida que da origen al problema, sino todas las células del sistema. El esquema del conmutador con colas a la entrada se puede ver en la

Figura 6 :

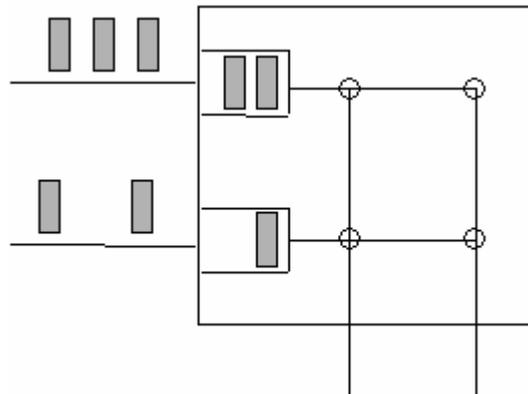
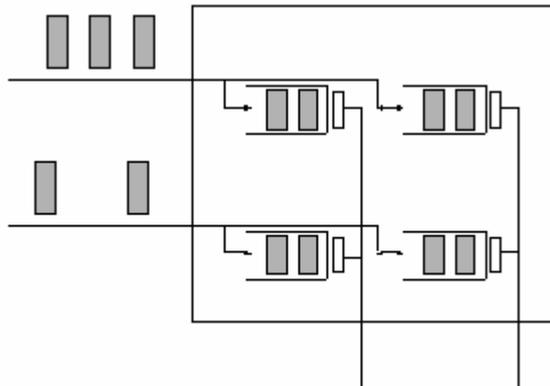


Figura 6. Conmutador con colas a la entrada

#### 6.7.3.1.2 Colas en los puntos de cruce

En esta arquitectura se colocará una cola por cada punto de cruce de la matriz. Así se evita el bloqueo en la cabecera de la línea, pero sus requerimientos de memoria

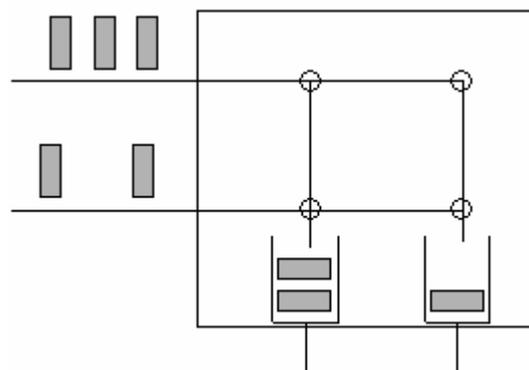
son muy superiores. También es necesaria la utilización de un filtro a la entrada que reconozca la salida a la que va dirigida cada célula para ubicarla en la cola correspondiente. El esquema se muestra en:



**Figura 7. Conmutadores con colas en los puntos de cruce.**

### 6.7.3.1.3 Colas a la salida del conmutador

Es la técnica de ubicación de memoria que obtiene una mejor relación retraso-rendimiento. La colocación de las colas da lugar a que en una rendija de tiempo se puedan transferir más de una célula a una misma salida. Para asegurar que no se pierde ninguna célula la transferencia debe ser  $N$  veces más rápida que la entrada de las mismas. Donde  $N$  es el número de entradas. Las colas a la salida también evitan que se produzca el bloqueo en la cabecera de la línea. En la Figura 8 esquematiza este tipo de conmutador:



**Figura 8. Conmutadores con colas a la salida**

La elección de una arquitectura determinada no sólo depende de la tecnología hardware disponible sino también de las características de la red. Por ello, la selección de la tecnología apropiada estará influenciada por muchos factores y cambiará con el tiempo.

## **6.8 GESTIÓN DE TRÁFICO Y CALIDAD DE SERVICIO EN ATM**

La tecnología ATM se caracteriza por su capacidad de soportar simultáneamente tráfico proveniente de distintas fuentes con características dispares.

Una aplicación desarrolla sobre ATM deberá especificar a la red el ancho de banda que requiere para la transmisión de los datos así como la calidad de servicio que pretende obtener de la red. Esto lo realiza por medio de un 'contrato de tráfico'. En él se especifican los siguientes componentes:

- *Categoría de servicio.*
- *Calidad de servicio requerida(GOS)*
- *Características de tráfico de la conexión.*

La red y el usuario negociarán estos parámetros en función de los recursos disponibles de la red.

### **6.8.1 Categorías de servicio**

Se han definido una serie de categorías de servicio de forma que con ellas se puedan cubrir el espectro de todas las aplicaciones existentes usando las redes ATM. Dependiendo del tipo de tráfico que configure el servicio, éste se asociará a una categoría u otra. Las categorías de servicio son:

- *Servicio de tasa de bits constante (CBR).*
- *Servicio de tasa de bits variable (VBR).*
- *Servicio de tasa de bits disponible (ABR).*

- **Servicio de tasa de bits no especificada (UBR).**

A los servicios CBR y VBR el ancho de banda se les asigna cuando la conexión es admitida. Esta asignación se considera vigente durante todo el tiempo que dure la conexión aunque haya momentos en que ésta lo infrutilice. La red requerirá maximizar la utilización del ancho de banda si quiere ser realmente efectiva. Para ello, deberá aprovechar los intervalos en los que las conexiones que tienen ancho de banda asignado permanecen ociosas. ABR y UBR utilizan el ancho de banda desaprovechado por otras conexiones. Se les asigna de forma dinámica, por ello también se les conoce como servicios de ancho de banda en demanda. Debe quedar claro que las conexiones CBR o VBR no utilizan constantemente el ancho de banda que les fue asignado en el establecimiento de la conexión, pero cuando lo requieran deberán tenerlo siempre a su disposición, pudiendo ser utilizado en el resto de las situaciones para lo que la red disponga.

#### **6.8.1.1 Servicio CBR (Constant Bit Rate)**

Es el tipo de servicio elegido para soportar aplicaciones en tiempo real. Estas aplicaciones son fundamentalmente aquellas que contienen información audio o de vídeo. En ellas los requerimientos de retardo son muy estrictos y el sistema final podría ser muy sensible a las variaciones del retraso extremo a extremo. La aplicación puede que sea intermitente por naturaleza pero el ancho de banda se le asigna durante toda la conexión aunque no sea necesario.

El servicio CBR proporciona una conexión con ancho de banda dedicado con una probabilidad de pérdida extremadamente baja, así como un retraso predecible. El tiempo entre llegadas de las células es constante y puede ser caracterizado en función de un tiempo mínimo entre llegadas que se podrá conocer en función de la tasa pico de emisión (PCR).

#### **6.8.1.2 Servicio VBR (Variable Bit Rate)**

Esta categoría de servicio va dirigida fundamentalmente a soportar aplicaciones de vídeo y tráfico del tipo FR( *frame relay*), así como cualquier otra aplicación que utilice tráfico con características de intermitencia conocidas o predecibles. El tráfico del servicio VBR se caracteriza principalmente por los parámetros PCR y SCR (tasa media de transmisión). El ancho de banda que se le asignará será menor que el ancho de banda pico y mayor o igual que el valor medio del mismo.

Esta categoría de servicio está a su vez dividida en dos subcategorías basadas en los requerimientos de retardo de las aplicaciones. Son la VBR de tiempo real (rt-VBR) y

VBR sin requerimientos de tiempo real (nrt-VBR). La primera tiene requerimientos muy estrictos de retardo, luego no debe ser almacenada extensivamente en su recorrido por la red. La nrt-VBR no garantiza ninguna cota del retardo así que puede ser almacenada con mayor libertad.

#### **6.8.1.3 Servicio ABR ( Available Bit Rate)**

Se utiliza para aplicaciones que no conocen a priori las características de su tráfico. Estas serán aplicaciones moderadamente sensibles a la pérdida de células y no tendrán las restricciones de tiempo que presentan las aplicaciones en tiempo real.

El servicio ABR garantizará un mínimo de ancho de banda y puede limitar la transmisión a la tasa pico. La fuente de tráfico ABR participará activamente en los protocolos de control para minimizar las pérdidas de células por la red.

#### **6.8.1.4 Servicio UBR ( Unspecified Bit Rate)**

Es un servicio muy simple desde el punto de vista de control de tráfico, ya que cualquier aplicación UBR puede usar tanto ancho de banda como quiera mientras esté disponible. Pero la red no garantiza ni siquiera un mínimo de ancho de banda a este tipo de conexiones. Se utiliza para los mismos casos que ABR pero siempre que la aplicación pueda soportar grandes pérdidas de células. También llamado servicio best effort.

### **6.8.2 Gestión de tráfico**

El objetivo que se busca es prestar servicios con la calidad pedida por los usuarios manteniendo una buena utilización de los recursos de la red. Para ello es necesario establecer una buena gestión del tráfico que fluye por la red. Esto se consigue gracias a las funciones que se describen en este apartado.

#### **Control de admisión de conexiones (CAC)**

Se considera la primera forma de protección de la red contra una carga excesiva. Cuando un usuario pide una nueva conexión, debe especificar las características del tráfico que requerirá la misma. También seleccionará el grado de servicio (QOS) de entre los posibles grados de servicio que la red puede ofrecer. La red debe examinarse a sí misma contemplando la posibilidad de aceptar la nueva conexión y el efecto que tendría sobre las demás conexiones ya aceptadas. La red aceptará la nueva conexión siempre que pueda soportar la nueva carga con la calidad pretendida por el usuario sin

que vaya en detrimento de las demás conexiones existentes. Si se accede a la petición, la red se compromete a mantener el servicio con la calidad pactada.

### **Control de parámetros de uso (UPC)**

También llamado control policía. Durante la fase de transferencia, la red comprueba si el tráfico de las conexiones se ajusta a lo pactado en el contrato que se hizo en la CAC. El objetivo principal del UPC es proteger los recursos de la red de las conexiones que pudieran saturar los recursos por incumplimiento del contrato. Cuando la red detecta alguna violación por parte de una conexión tomará medidas que irán desde marcar sus células como ‘células de descarte preferente’ (usando el campo CLP de la cabecera) hasta descartar directamente las células, o incluso liberar la conexión en caso de que tuviera problemas en la red.

### **Control de prioridad (PC)**

Este control entra en juego cuando el sistema debe descartar células para conseguir un buen funcionamiento de la red. El objetivo estará entonces en eliminar aquellas células que tienen prioridad menor. Hay que reparar en que la red no es capaz de distinguir entre aquellas células que fueron marcadas de baja prioridad desde la fuente y las que son marcadas por la función UPC.

### **Conformado de tráfico (traffic shaping)**

Esta función va ligada al control policía pero se lleva a cabo por parte del usuario de la red. Este, para evitar las medidas del UPC, evalúa el cumplimiento del contrato de forma periódica de forma que si detecta violaciones del mismo rectifica la conducta de la fuente hasta ajustarse a lo pactado con la red.

### **Control de flujo**

Sólo se usa para servicios ABR. Se basa en la realimentación desde la red al usuario. Indica al usuario el estado de la red para que éste controle su tasa ajustándola al ancho de banda disponible en cada momento. Se utilizarán un tipo de células especiales llamadas RM que son células para la gestión de recursos (del inglés Resource Management) que tienen campos específicos para poder controlar el tráfico.



## 7 ANTECEDENTES E HIPÓTESIS DEL PROBLEMA

El problema de optimización de ruta en redes de telecomunicaciones ha sido tratado en multitud de ocasiones, ya sea en redes con servicio orientado a conexión (como la RTB) o en redes con servicio no orientado a conexión (por ejemplo una red IP). ATM tiene similitudes con los dos tipos de redes indicados anteriormente, sin embargo, no se pueden considerar correctos los algoritmos de encaminamiento que se utilizan para ninguna de ellas, debido a sus diferencias.

En ATM existen dos parámetros fundamentales de referencia a la hora de determinar si la información recibida por el nodo destino cumple con el grado de servicio contratado con la red: el número de células perdidas por la red antes de llegar al destino y el retardo sufrido por la información que finalmente llega.

En este proyecto se pretende estudiar la posibilidad de encaminamiento dinámico de la información en las redes que utilizan tecnología ATM, utilizando como icono para ello la pérdida de células producida en la red. Sin embargo, no se deja de lado el otro parámetro fundamental (el retardo), ya que se procede a buscar el encaminamiento óptimo minimizando la pérdida de células, a la vez que se incorporan unas condiciones mínimas, a satisfacer, sobre el retraso en las comunicaciones a través de red.

Se trata, por lo tanto, de trabajar con una red de telecomunicaciones con una topología dada, de forma que la aplicación del algoritmo desarrollado permita elegir por qué camino se deben establecer las conexiones para conseguir el funcionamiento más óptimo y eficaz que se pueda.

Para alcanzarlo, se estudia la función de pérdidas que siguen los procesadores de conmutación utilizados y se aplica a un modelo matemático de optimización de rutas, el cual deberá ser analizado y resuelto. Finalmente, aplicará el algoritmo sobre distintas topologías de red y se analizará la bondad de los resultados que se obtengan.

Los puntos principales de partida del proyecto que se desarrolla se muestra en la Tabla 5:

**Tabla 5. Tabla representativa de las premisas fundamentales.**

<b>Descripción / Premisas fundamentales</b>	<b>Descripción general</b>
<b>Tráfico CBR</b>	En ATM se utilizan tráficos distintos con diferentes características, de emisión, de grado de servicio que contratan, todo esto hace muy difícil su implementación conjunta en un modelo. El tráfico CBR es

<b>Descripción / Premisas fundamentales</b>	<b>Descripción general</b>
	el que presenta las restricciones más estrictas en cuanto a calidad, y es el tipo de tráfico elegido para el desarrollo que se realiza.
<b>Conmutadores con colas</b>	Procesadores de conmutación con colas a la salida: Actualmente existe una gran gama de tipos de conmutadores. El utilizado para el modelo en cuestión es el conmutador con colas a la salida
<b>Enlaces STM-1</b>	Enlaces STM-1: Las redes consideradas están formadas por enlaces con capacidad de 155,52 Mb/s.

## 8 ENCAMINAMIENTO EN UNA RED ATM

### 8.1 JUSTIFICACIÓN DE ATM

ATM fue adoptado por el ITU-T como el medio de transporte sobre el cual se debería basar la RDSI- BA. La necesidad de un encaminado óptimo de células por la red es evidente, ya que las prestaciones que pueda ofrecer la red dependerán en gran medida de la bondad del encaminamiento del tráfico que circule por la misma.

El problema de optimización de ruta - encaminado, encaminamiento o enrutado en redes de telecomunicaciones ha sido tratado en multitud de ocasiones, sea en redes con servicio orientado a conexión (como la RTB) o en redes con servicio no orientado a conexión (por ejemplo una red IP). ATM tiene similitud con los dos tipos de redes indicados anteriormente sin embargo, no se puede considerar como válidos los algoritmos de encaminamiento que se utilizan para ninguna de ellas, debido a sus diferencias evidentes.

Una red ATM se puede representar mediante un grafo  $G = (N, E)$  donde  $N$  sea el conjunto de nodos (conmutadores) y  $E$  el conjunto de enlaces de comunicación. Cada nodo estaría compuesto por un conjunto de búferes y una 'fábrica de interconexión'. Por búfer se entiende un lugar de almacenamiento para células, normalmente con capacidad finita. Se usa siempre que es necesario compartir los recursos entre usuarios, ya que todos no pueden ser servidos a la vez, y de otra forma serían desechados sin ser servidos. Con 'fábrica de interconexión' se denomina al conjunto de conexiones internas que existen en el conmutador de forma que cualquier célula que entre en el sistema pueda ser conmutada a la salida que le corresponda. Esto se puede implementar de formas muy diversas, ya sea por medio de una matriz de interconexión, por memoria central o por un bus o un anillo de interconexión. Cada una de ellas presenta ventajas e inconvenientes, por esto se utiliza una implementación u otra en función del tipo de red.

Se representará cada buffer como una cadena de Markov y se buscará un método para calcular las pérdidas. Tomando como base la probabilidad de pérdida de célula se expone un modelo de encaminamiento de tráfico para una red ATM. Las pérdidas que se tendrán en cuenta serán únicamente las que se producen en el interior de los conmutadores.

Las redes ATM poseen varias características que las hacen merecedoras de la gran aceptación que están teniendo en el mundo de las telecomunicaciones. Sobre todo, porque son capaces de integrar sobre ellas cantidades variables de flujos con características muy distintas, y a alta velocidad. He ahí la característica fundamental.

ATM es una tecnología de alta velocidad. Así, es capaz de transferir unidades de información entre nodos extremos de una comunicación con una velocidad elevada. Es por ello que en una red ATM los requerimientos de retardo son muy estrictos, ya que se debe conseguir la gran velocidad de transmisión que se espera de ella. Para que una red de estas características pueda funcionar satisfactoriamente será imprescindible un medio

físico capaz de permitir dicha velocidad. El medio físico que se utiliza es la fibra óptica. Sin embargo, no bastará tener una red potencialmente veloz, también será necesario buscar el camino óptimo por el que enviar los datos de cada conexión, de forma que estos ocupen los recursos de la red durante el menor tiempo posible. Dicho de otra forma, la importancia de un buen encaminamiento radica en que, partiendo de la misma topología y datos de demanda, dependiendo del reparto que se haga del flujo, se puede obtener una buena transmisión del flujo o bien una transmisión lenta y/o defectuosa de los datos. Esto último implicaría que quedarán algunos enlaces demasiado cargados y otros infrautilizados. Por ello es tan importante encaminar correctamente.

Desde un principio, se ha indicado la importancia del retraso para ATM, se podría entonces determinar como función objetivo la minimización del retardo para encaminar óptimamente por la red. Esto ya se ha contemplado en ocasiones, por ejemplo, en Amiri et al (1998). Sin embargo, no se debe ni se puede olvidar que en ATM se garantizan los servicios. Cuando se abre una conexión, la red negocia con el usuario unos parámetros de servicio, y se compromete a garantizarlos durante todo el tiempo que la conexión permanezca abierta. La red, antes de aceptar una conexión deberá, por lo tanto, analizar los recursos de los que dispone en ese momento. Sabrá qué recursos necesitan utilizar las conexiones que hay abiertas en el sistema. Si no se concluye que se pueden mantener los requerimientos de la nueva conexión, no la acepta. Si la acepta es necesario asegurar la calidad de la conexión mediante una serie de parámetros. Entre ellos está el retraso medio de las células por conexión, pero también otros, como la variación del retraso de célula y la tasa de células perdidas. Se considera por ello, que la pérdida de información en la red es igualmente importante, si no más, que el exceso de retardo. De forma que si el número de células que se pierde aumenta, aun cuando la red sea sumamente rápida, no se podría asegurar que sus prestaciones fueran buenas. Las células que llegaran lo harían con retardos pequeños, pero no llegaría toda la información que requiere el servicio. Por lo tanto, realmente no se estarían soportando los servicios con la calidad requerida. Por esta razón se procede a buscar el encaminamiento óptimo minimizando la pérdida de células, a la vez que se incorporan unas condiciones mínimas, a satisfacer, sobre el retraso en las comunicaciones de red.

En las redes ATM, para conseguir rapidez, se han eliminado tanto el control de flujo por ventana entre nodos adyacentes, como la retransmisión de células. Estas dos formas de corrección frente a pérdidas dan lugar a mucho tráfico por la red, tráfico que se eliminará en ATM, toda vez que el medio de transmisión será de pérdidas muy pequeñas. La utilización de un medio físico como la fibra óptica permitiría la transmisión de información con una tasa de pérdidas prácticamente nula. En este contexto no sería, por tanto, necesario considerar la minimización de pérdida de células como objetivo a considerar en la optimización del encaminamiento de la red. Pero este razonamiento no resulta válido, pues de hecho se producen pérdidas en ATM, sobre todo en los conmutadores de células - o switches en inglés-.

En ATM las células se pierden fundamentalmente de cuatro maneras, o por cuatro causas distintas, que se detallan en la Tabla 6:

**Tabla 6. Representación de los distintos tipos de pérdidas en una red ATM**

<b>Descripción</b> <b>/Tipos de pérdidas</b>	<b>Descripción general</b>
<b>Errores de transmisión en las líneas</b>	Estas pérdidas son muy pequeñas y por ello no se tienen en cuenta
<b>Debido al control UPC</b>	Se realizan por la red en los conmutadores
<b>Falta de espacio en los búferes de los conmutadores</b>	Se producen al almacenar las células que llegan
<b>Limitaciones que existen en la propia fábrica de interconexión</b>	Se encarga de encaminar las células que entran por cualquier entrada hacia la salida correspondiente.

Como se ha indicado, el modelo buscará la asignación óptima del flujo atendiendo a los requerimientos de pérdidas de células en cada conexión, y de forma que se pierda el menor número de células posibles

Se partirá de una serie de datos como son: la topología de la red, la capacidad de los enlaces de la misma y la demanda entre cada par de nodos que quieran establecer una conexión. También será dato el máximo número de células perdidas por conexión, que determinará el grado de servicio de la misma.

Como se ha indicado con anterioridad, el objetivo principal del modelo planteado será buscar la ruta óptima para cada flujo de datos atendiendo a las pérdidas, no al retraso sufrido por una célula al atravesar la red que se impondrá como restricción de diseño. Esta será la forma de enfocar el problema de encaminamiento aun sabiendo que los requerimientos respecto a retrasos en ATM son muy estrictos. La explicación que sostiene este enfoque resulta simple si analizamos las fuentes de retraso que existen en la red. A continuación, en la Tabla 7, se muestran los diferentes tipos de retraso.

**Tabla 7. Tabla representativa de los diferentes tipos de retraso.**

<b>Descripción /</b> <b>Tipos de retrasos</b>	<b>Descripción general</b>
<b>Conversión señal óptica en eléctrica</b>	Este retraso no afecta a la red troncal, que es la que se maneja, ya que en ella la señal ya se encuentra convertida y no hacen falta posteriores conversiones de un modo a otro.

<b>Descripción / Tipos de retrasos</b>	<b>Descripción general</b>
<b>Propagación señal en medio físico</b>	El medio físico utilizado es la fibra óptica, entre cuyas características se encuentra un retraso de transmisión de la señal despreciable.
<b>Cola conmutador</b>	Las células que no pueden ser servidas inmediatamente tienen que esperar hasta que llegue su turno, esto lógicamente conlleva un retraso sobre el tiempo de transmisión ideal de la célula.

De las cuatro fuentes de retraso que vemos que existen, las dos primeras introducen un retraso despreciable. Las otras dos sí provocan retardos en las células, si bien debemos recordar que debido a las características que presenta ATM estos retardos son bastante menores que los que presentan los conmutadores de otro tipo de redes.

Se puede concluir que el único retardo que aparece se introduce en los conmutadores; el retraso depende en exclusiva del número de conmutadores que atraviesan los datos, y esto depende a su vez del encaminamiento utilizado. Por lo tanto, bastaría con determinar un número máximo de saltos concedidos a cada paquete, o lo que es lo mismo, un número máximo de nodos que cada unidad de información puede atravesar para llegar al destino. Así, se podría introducir una restricción para limitar el retraso, lo que se pediría al sistema es que el número de conmutadores atravesados entre origen y destino de una conexión fuera el mínimo posible. Esto se atendería mediante la determinación de una cota superior sobre el número de saltos de conmutadores. En el modelo propuesto se busca minimizar las pérdidas, pero se encuentra que las fuentes de pérdidas significativas están, al igual que las de retraso, en los conmutadores. Si se quieren minimizar las pérdidas, el número de conmutadores que atraviesen las células de una conexión deberá ser el menor posible. Será pues lógico pensar que con el modelo propuesto se está buscando de forma indirecta la minimización del número de saltos de las células por la red, y como consecuencia, se estará limitando el retardo de la información sin necesidad de incluir una restricción específica al respecto.

## 8.2 ESCALAS DE TIEMPO

Cualquier análisis basado en la variación del tráfico que fluye por una red de telecomunicaciones tiene que tener una ubicación temporal. Esto es, el análisis deberá centrarse en un periodo de tiempo que resulte significativo para el estudio que se pretende realizar. Además, si se pretende representar la red con un conjunto de variables

será importante determinar la relación que éstas guardan con el tiempo. Para poder establecer esta última relación, es fundamental conocer las escalas de tiempo que se adecuan al comportamiento del sistema. De esta forma se seleccionarán las escalas de tiempo en las cuales sea posible identificar la ocurrencia de algún proceso estocástico relevante para el análisis.

Para el modelo de Erlang (universalmente aceptado para telefonía tradicional con circuitos conmutados) se considera una única escala de tiempos. Es la escala de conexión, también conocida como connection scale o call scale en inglés. Esta escala de tiempos viene determinada por la llegada de una llamada al sistema. Sin embargo, cuando se habla de tráfico de banda ancha no es posible limitarse a la escala de conexión. Esto se debe a que se producen otros procesos importantes en otras escalas de tiempo más pequeñas que no se pueden ignorar. En Medova (1998) se realiza un análisis detallado al respecto considerando tres niveles o escalas de tiempo. A saber: call level, burst level y cell level que corresponden en castellano al nivel de conexión, de ráfaga y de célula respectivamente. Estos niveles corresponden a tres niveles jerárquicos para la gestión de redes de telecomunicaciones de banda ancha.

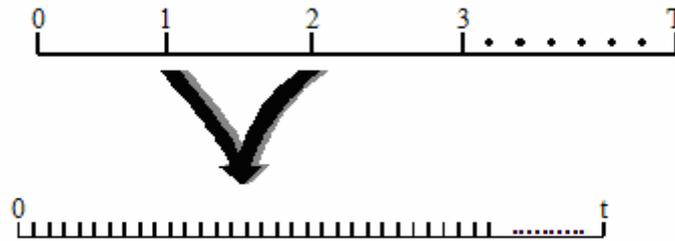
En éste proyecto, se trabajará con dos de estos tres niveles, o escalas jerárquicas. El nivel de conexión y el de célula.

Trabajar con el nivel de célula resulta indispensable si se tiene en cuenta que el análisis de pérdidas en los conmutadores debe hacerse célula a célula. Los conmutadores se encargan de encaminar las células pertenecientes a distintas conexiones por el camino correcto. El conmutador, por lo tanto, trabaja a nivel de célula y para poder contabilizar las pérdidas en los conmutadores las variables de flujo deberán estar referidas a esta escala de tiempo.

El encaminamiento de una llamada, por tratarse de una red ATM, se mantendrá durante todo el tiempo que dure la misma. De esta forma, todas las células que pertenecen a una conexión entre dos nodos recorrerán el mismo camino desde el origen al destino. Según esto, si se intenta buscar el encaminamiento entre dos nodos habrá que tener en cuenta que las variables que indiquen el camino a recorrer por las células deberán estar referidas a una conexión. Se moverían en otra escala de tiempos distinta a la de las variables de flujo. Este sería el nivel de conexión o de llamada.

En la Figura 9 que se incluye a continuación se pueden observar los distintos niveles que se utilizan y la relación que guardan entre sí:

*Nivel de llamada: cada ranura representa el tiempo medio de duración de una llamada,  $t$ .*



*Nivel de célula: cada llamada se divide en un conjunto de células. Cada una de ellas ocupa un tiempo fijo.*

**Figura 9. Diferentes escalas de tiempo.**

### 8.3 PÉRDIDAS PRODUCIDAS EN LOS CONMUTADORES ATM

La función de pérdidas utilizada, guarda una gran relación con la estructura interna de los conmutadores de la red. Un conmutador ATM debe encaminar las células que le llegan hacia la salida apropiada. Se han desarrollado gran variedad de conmutadores distintos para el encaminamiento en ATM. Cada uno de ellos presenta una serie de ventajas e inconvenientes dependiendo de la distinta ubicación de los búferes y del tipo de interconexión utilizada. Sin embargo, actualmente, la forma mas usada es la que utiliza malla o red de interconexión con colas a la salida.

Así, el número de células perdidas en un conmutador ATM dependerá del tipo de conmutador, habiéndose mostrado que los que mejor rendimiento ofrecen son aquellos que presentan las colas a la salida. Por ello, se considerará la red diseñada con este tipo de conmutadores.

Los conmutadores ATM son sistemas ranurados en los que cada ranura, o slot en literatura inglesa, ocupa el tiempo de transmisión de una célula (aproximadamente  $2,8\mu\text{seg}$  si la velocidad de transmisión es de  $155,52\text{ Mb/s}$ ). Para poder analizar el comportamiento de las células en el conmutador hay que tener en cuenta los instantes de tiempo en los que se produce cada suceso en el búfer de salida. En una red ATM existe sincronización, de esta forma, el suceso de partida de una célula abandona el buffer al final de la ranura de tiempo y para ser servida se sincroniza con el comienzo de la siguiente ranura. Las llegadas de células se pueden producir en cualquier instante durante todo el tiempo que dura una ranura. No es relevante el instante de llegada de cada célula, sin embargo, se establece que cualquiera que sea el instante de una llegada, ésta se producirá antes del instante de partida de la célula que esté en la cabecera del

buffer (siempre que no esté vacío). Esta forma de gestión del buffer se llama de ‘llegadas primero’. También se supone que si una célula llega durante la ranura  $n$  tendrá que esperar como mínimo a la  $n+1$  para ser servida.

Con la palabra puerto se designará a cada una de las entradas que llegan al conmutador, así como cada una de las salidas que parten de él. Se diferencian por lo tanto dos tipos de puertos: de entrada y de salida del conmutador. Cada puerto de salida llevará a un enlace de salida y cada puerto de entrada será la terminación de un enlace de entrada al conmutador. Al nombrar un enlace de salida, teniendo en cuenta que a cada uno de ellos hay asociada una cola y un servidor que desemboca en un puerto de salida, se podrán utilizar independientemente los términos de enlace de salida, salida, puerto de salida o cola del sistema.

A continuación, en la Figura 10, se incluye un esquema del sistema según lo expuesto anteriormente:

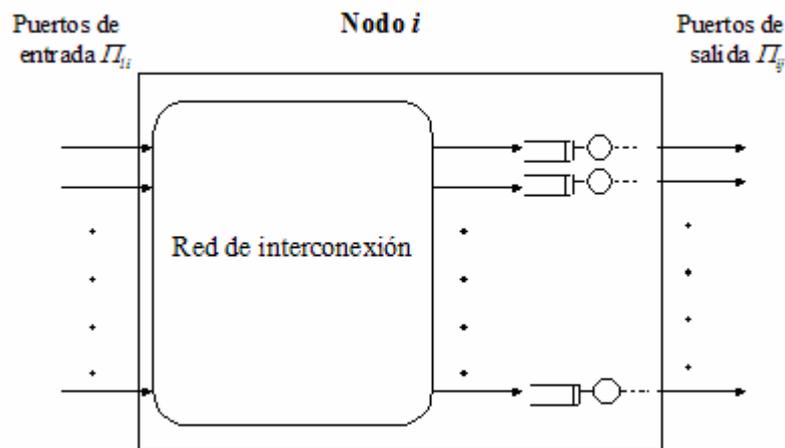


Figura 10. Red de interconexión.

#### 8.4 CONMUTADORES CON COLAS

No se pretende en este apartado hacer un análisis exhaustivo de la teoría de colas, sino más bien introducir aquellos aspectos de la misma más relevantes para la justificación de la expresión de la función de pérdidas utilizada.

Se presenta la teoría de colas para tratar las pérdidas en los conmutadores debido a que los recursos que se tienen en los mismos son limitados. ATM establece

conexiones, sin embargo, no hay que perder de vista el hecho de que toda conexión consiste en un flujo de células que debe atravesar la red compartiendo los recursos de la misma, también los conmutadores. Para el análisis, éstos no serán más que una fábrica de interconexión y un conjunto de colas.

Los clientes, en este caso células, llegarán al sistema – el conmutador o nodo - para ser conmutados, esperando en la cola si no pueden ser servidos inmediatamente. Cuando les llegue el turno, serán servidos y abandonarán el sistema.

Un sistema de colas se describe por una serie de parámetros: el patrón de servicio, el número de servidores, la capacidad del sistema, la disciplina de cola y el patrón de llegadas:

- El patrón de servicio vendrá descrito por el tiempo requerido para servir a un cliente,  $s$ , o bien, por la tasa de servicio  $\mu$ . Normalmente, si el sistema está vacío se dice que el servidor está ocioso, pero en ATM, aunque el buffer de células esté vacío se seguirá transmitiendo, ya que el flujo debe ser siempre continuo. Es por ello que los servidores con los que se trata son sincronizados y deterministas. El tiempo de servicio que se toma es siempre el mismo e igual a una ranura de tiempo, que es lo que tarda en transmitirse una célula.
- Con el número de servidores determinamos cuantos canales hay en servicio, o cuantos clientes pueden ser servidos simultáneamente. Cuando hay más de un servidor, cada uno puede tener su propia cola, o bien, que el sistema sea de cola compartida de forma que sólo haya una cola en el sistema. Para el caso que se considera, cada servidor se encargará de ofrecer servicio a los clientes que tiene en su cola.
- La capacidad del sistema consiste tanto en la zona de espera (capacidad de la cola), como en los canales de servicio. Puede ser finita o infinita. Evidentemente, para sistemas reales debe ser finita, sin embargo, muchas veces el suponer capacidad infinita proporcionará simplificaciones enormemente útiles que ayudarán a describir el comportamiento del sistema aun a riesgo de perder cierta exactitud en los resultados.
- El patrón de llegadas se refiere a las llegadas de clientes al sistema, puede ser especificado con el número medio de llegadas en la unidad de tiempo ( $\lambda$ ) o bien con el tiempo medio entre llegadas de los clientes. Si se considera que las células no llegan de una en una sino en lotes - grupos, o batches, en literatura anglosajona - debemos tener en cuenta que el número de células que llegan en cada ranura podría variar.

La notación usada comúnmente para describir los sistemas de colas, a partir de los parámetros ya descritos es la de Kendall: A/B/X/Y/Z:

- A especifica la distribución del tiempo entre llegadas al sistema.
- B especifica la distribución del tiempo de servicio.
- X especifica el número de servidores.
- Y especifica la capacidad del sistema.
- Z especifica la disciplina de cola

Por defecto, se toma:

$$Y = \infty$$

$$Z = FIFO$$

Para obtener el valor de las pérdidas primero hay que determinar cual será el comportamiento del sistema. Teniendo en cuenta que la tasa de servicio es constante e igual al tiempo de duración de una ranura y suponiendo la distribución de llegadas exponencial, así como el tamaño de cola finito K, se concluye que el modelo de colas que mejor se ajusta al sistema que se estudia será el M/D/1/K.

Se considera por lo tanto, que la disciplina de cola es FIFO. Esto es, el primer elemento que entra en la cola es el primero que se sirve (first in first out en literatura inglesa). Esta suposición lleva consigo una simplificación del sistema para facilitar el análisis, ya que el comportamiento real no se ajustará siempre a esta disciplina de cola. En las redes ATM existirán células con distintas prioridades que requerirán distinto trato a la hora de esperar a ser servidas. Entre estas células se encuentran las de control y gestión de la red que tendrán mayor prioridad que las de datos.

El modelo M/D/1 está ampliamente estudiado y las fórmulas de probabilidad de estado son conocidas y aparecen en numerosos textos. No ocurre lo mismo para el caso M/D/1/K en el que las colas dejan de ser infinitas. Es por esta razón que a continuación se hace un pequeño estudio hasta obtener la probabilidad de pérdida del sistema, análisis que se hace se basa fundamentalmente en el flujo de células a través de un buffer ATM. En el caso que se está considerando el tiempo entre llegadas viene dado por una distribución exponencial negativa (Pitts et al, 1996). Es conveniente señalar que esta es la misma distribución que se toma cuando se habla del comportamiento del usuario a la hora de establecer llamadas en una red de circuitos conmutados. Sin embargo, aunque se esté tomando el mismo patrón, en realidad se están tratando dos cosas totalmente distintas. Las escalas de tiempo también lo son. En el caso que se estudia, la escala de

tiempo que interesa es de una ranura, el tiempo de transmisión de una célula. Si continuamos con el patrón de llegadas, se pueden considerar dos distribuciones distintas según se considere un eje temporal continuo o discreto.

### Tiempo Continuo

En tiempo continuo la distribución de llegadas según una exponencial negativa forma un proceso de Poisson donde la probabilidad de que se produzcan k llegadas en un intervalo continuo de tiempo T, siendo  $\lambda$  la tasa de llegada de células, viene dada por:

$$P(k) = \frac{(\lambda T)^k}{k!} e^{-\lambda T} \quad \text{Ecuación 2}$$

La variable aleatoria k indica el número de llegadas que se producen en el periodo de tiempo T. Por lo tanto, como se determina en la Ecuación 2, la variable k se distribuye según una Poisson de parámetros  $\lambda T$ , donde  $\lambda T$  es la tasa de llegada de células durante todo el periodo de tiempo considerado: T.

### Tiempo Discreto

En tiempo discreto, el patrón de llegadas forma un proceso de Bernoulli en el que la probabilidad de que haya una llegada en una ranura es p y la probabilidad de que la ranura esté vacía es de 1-p. Si en lugar de considerar una sola ranura consideramos N ranuras, entonces se configura una distribución Binomial, donde la probabilidad de que se den k llegadas en N slots será:

$$P(k) = \frac{N!}{(N-k)!k!} (1-p)^{N-k} p^k \quad \text{Ecuación 3}$$

La distribución tiende a una Poisson a medida que N aumenta.

Para cualquier tipo de distribución se pueden considerar las llegadas en lotes. Entonces, en lugar de considerar que se produzca la llegada de una célula con probabilidad p, se considerará que con esta probabilidad se produce la llegada de un grupo con varias células (a priori se desconoce cuantas). Esta forma de ver las llegadas

es especialmente útil cuando se trata de modelar las llegadas a los conmutadores, ya que a las colas de los mismos podrán llegar células desde cualquiera de los puertos de entrada. Los conmutadores considerados tendrán las colas en los puertos de salida, pudiendo un búfer recibir en una ranura de tiempo un número variable de células, dependiendo del número de puertos que tenga a la entrada. Si se considera el número de puertos a la entrada igual a M, el número de células que se pueden recibir en un lote variará de 0 a M, incluyendo ambos extremos. De forma que ahora p será la probabilidad de que haya una llegada en grupo en una ranura de tiempo, y 1-p será la probabilidad de que no la haya. La probabilidad de que haya k células en un lote cuando se produzca una llegada en racimo, se llamará b(k). Para este tipo de procesos con llegadas en lotes la distribución del número de células por ranura a(k) se puede describir de la forma:

$$\begin{aligned}
 a(0) &= 1 - p \\
 a(1) &= pb(1) \\
 &\dots\dots\dots \\
 a(k) &= pb(k) \quad \forall 0 < k \leq M
 \end{aligned}$$

Si se intenta buscar la distribución del número de llegadas por intervalo de tiempo a(k) para un proceso de Poisson de tasa  $\lambda$  células por unidad de tiempo, y suponiendo el caso concreto en el que el intervalo de tiempo T tiene una duración igual a una ranura de tiempo. Podría entonces escribirse:

$$a(k) = \frac{\lambda^k}{k!} e^{-\lambda} \quad \text{Ecuación 4}$$

Análogamente, para la distribución Binomial, encontramos que la probabilidad de que se produzcan k llegadas en una única ranura de tiempo, suponiendo M puertos de entrada, cada uno de ellos con probabilidad p de producir la llegada de una célula en cualquier slot, es la que sigue:

$$a(k) = \frac{M!}{(M-k)!k!} (1-p)^{M-k} p^k \quad \text{Ecuación 5}$$

La tasa total de llegadas es de Mp células por ranura de tiempo. La distribución Binomial tiende a la de Poisson a medida que M aumenta y p disminuye.

## 8.5 ECUACIONES DE BALANCE

Para poder estudiar el comportamiento del sistema es necesario plantear las ecuaciones de balance del mismo. Para poder hacerlo se considerarán dos casos: el de un sistema sin pérdidas (con capacidad ilimitada) y de un sistema real. Para el estudio se tomarán las llegadas como procesos de Poisson.

Se considera que cada nodo tiene, asociado a cada puerto de salida, un buffer de tamaño finito  $b$ . Por lo tanto, cada puerto de salida se comportará como un servidor con una cola de tamaño  $b$ . Se tomará cada cola como una cadena de Markov con  $b$  estados. El cual, si el sistema se encuentra en el estado  $j$  querrá decir que hay  $j$  células esperando en cola al final de la ranura de tiempo considerada. Se entenderá por  $s(j)$  que el sistema esté en el estado  $j$ .

Si se analiza el sistema para llegar a las ecuaciones de balance, se encuentra que el hecho de que el número de llegadas de células a cada sistema sea en lotes, i.e. cada llegada se produce con un número aleatorio de células, se puede que pasar a un estado  $j$  no sólo a partir de los estados inmediatamente anterior,  $j-1$ , y posterior,  $j+1$ , sino que también se podrá hacer a partir de cualquiera de los estados anteriores al  $j$ . Esto queda reflejado en la Figura 11 que se muestra a continuación. En ella se pueden ver las posibles transiciones hacia un estado cualquiera  $i$ . Siendo  $a(k)$  la probabilidad de que lleguen  $k$  células en una misma ranura de tiempo.

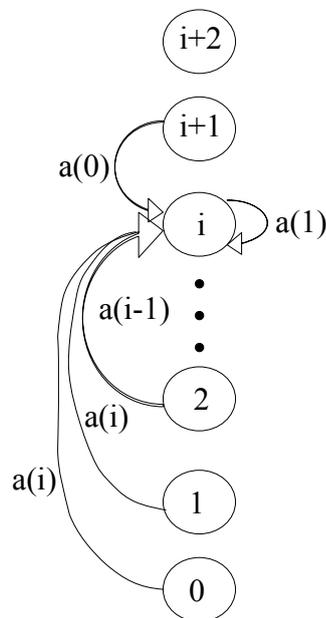


Figura 11. Diagrama de estado del sistema

Hay que notar sin embargo que, como queda reflejado en el diagrama, el sistema si bien puede recibir en un intervalo de tiempo un grupo de células, sólo es capaz de servir una célula en cada ranura de tiempo. Por esto, si estamos en el estado  $j$ , podremos pasar al  $j-1$  en caso de que no llegue ninguna célula nueva, y a cualquiera de los estados superiores  $j+1, j+2, \dots, b$  en función de las células que lleguen, en caso de que sí lleguen células.

## 8.6 COLAS INFINITAS

Se ha señalado que para sistemas reales no es posible considerar recursos infinitos. Sin embargo, para llegar a la distribución de la probabilidad de estado de los búferes de salida, primero se considerará que su capacidad es ilimitada, para luego, con ayuda de algunos de los resultados obtenidos, poder hacer un análisis más exacto del sistema con capacidad finita.

Para obtener los valores de las variables de estado  $s$  se analizarán estado a estado cómo puede ser alcanzado cada uno de ellos. Por ejemplo, para determinar la ecuación del estado 0, éste se puede alcanzar desde el estado 1 o el 0. Si el sistema estaba al final de la ranura  $n-1$  en el estado 0, para mantenerse en el mismo estado en la ranura  $n$  no debe haber recibido ninguna célula durante este periodo de tiempo. Si se encontraba en el estado 1, durante la ranura  $n$  tampoco debe haber recibido ninguna célula para poder pasar al estado 0. Estas son todas las formas posibles de llegar al estado 0 en el sistema. Esto se puede escribir en forma de ecuación:

$$s(0) = s(0)a(0) + s(1)a(0) \qquad \text{Ecuación 6}$$

Toda vez que  $a(k)$  es la probabilidad de que lleguen  $k$  células en una ranura de tiempo cualquiera. Reescribiendo la Ecuación 6, es posible obtener la ecuación correspondiente para  $s(1)$ :

$$s(1) = \frac{1 - a(0)}{a(0)} s(0) \qquad \text{Ecuación 7}$$

Análogamente, se puede escribir la ecuación de balance para  $s(1)$  que llevará a la expresión correspondiente para  $s(2)$  como se ve en la Ecuación 9:

$$s(1) = s(0)a(1) + s(1)a(1) + s(2)a(0) \quad \text{Ecuación 8}$$

$$s(2) = \frac{s(1) - s(0)a(1) - s(1)a(1)}{a(0)} \quad \text{Ecuación 9}$$

Haciendo lo mismo hasta llegar al estado general  $x$ , a partir de la ecuación de balance de  $x-1$  se obtiene

$$s(x) = \frac{s(x-1) - s(0)a(x-1) - \sum_{i=1}^{x-1} s(i)a(x-i)}{a(0)} \quad \text{Ecuación 10}$$

Todavía no se ha utilizado el hecho de que la cola sea infinita. Para calcular  $s(0)$  si será necesaria esta condición. Todo sistema cumple la ecuación de balance, esto es, el tráfico cursado es igual al tráfico ofrecido al sistema, menos las pérdidas que se producen en el mismo. O lo que es lo mismo: se cursan todas las células que llegan al sistema menos las que se pierden. Escrito matemáticamente queda:

$$C = A - L \quad \text{Ecuación 11}$$

Donde  $C$  es el tráfico que se cursa,  $A$  es el tráfico ofrecido al sistema y  $L$  es el volumen de tráfico que se pierde. Si el sistema considerado tiene capacidad ilimitada, no se perderá ninguna célula por desbordamiento del búfer, con lo que  $L=0$ , o escrito de otra manera:  $C=A$ . Considerando que las células llegan a tasa  $\lambda$  (cel/seg), y que el tiempo de servicio es  $s$ , igual a una ranura de tiempo, se puede escribir:

$$\lambda = \frac{\rho}{s} \quad \text{Ecuación 12}$$

Donde  $\rho$  será la utilización del sistema. Si se denomina  $E(k)$  al número medio de llegadas por ranura, es posible identificarlo con  $\rho$ . De forma matemática se obtiene que  $E(k)=\rho$ , ya que se está suponiendo un sistema sin pérdidas. Siguiendo con el razonamiento, se puede decir que la utilización no es más que la probabilidad de que el sistema no esté vacío. Con todo esto se llega a que

$$E(k) = \rho = 1 - s(0) \Rightarrow s(0) = 1 - E(k)$$

**Ecuación 13**

De esta forma, sólo a partir de la tasa de llegadas se llega a determinar la probabilidad de que el sistema esté vacío. Una vez obtenida  $s(0)$  ya es posible analizar el comportamiento del sistema.

Es importante remarcar que la tasa media de llegada de usuarios a la cola  $E(k)$  no sobrepasará normalmente la unidad. Si  $E(k)>1$  el sistema tiende a hacerse inestable de forma que se satura. El que  $E(k)>1$  significará que llegan, como media, al sistema más de una célula por ranura de tiempo - o más de 155Mb/s que es la capacidad máxima de transmisión de ATM si se implementa sobre un STM-1- como consecuencia, es normal que sature. Otra cosa distinta es que en un slot determinado concurren dos o más células a la entrada de la cola, debido a la aleatoriedad de las llegadas. Si no se dieran estas coincidencias en las llegadas no tendría sentido la utilización de los búferes en los conmutadores.

## 8.7 COLAS FINITAS

Para esta situación, sí se considerarán pérdidas en el sistema de colas. Los búferes serán de tamaño limitado  $b$ . Se puede, como se hizo en el caso de colas infinitas, buscar recursivamente las ecuaciones de balance del sistema, obteniendo las mismas ecuaciones para  $s(x)$ . Sin embargo, ahora existe una dificultad añadida, ya que no será posible encontrar  $s(0)$  a partir de la ecuación de balance de tráfico. Esto es debido a que existen pérdidas cuyo valor ignoramos, ya que es precisamente el número de células perdidas la incógnita buscada.

Para resolver la cuestión se propone un cambio de variables, (la variable  $s(x)$  por  $u(x)$ ). Se define la nueva variable  $u(x)$  del modo:

$$u(x) = \frac{s(x)}{s(0)}$$

**Ecuación 14**

de forma que para  $x=0$  sabemos que  $u(0)=1$ . Si hacemos el cambio de variable en la Ecuación 14, obtenemos:

**Ecuación 15**

$$u(1) = \frac{1 - a(0)}{a(0)}$$

Análogamente, para la Ecuación 15, obtenemos:

$$u(x) = \frac{u(x-1) - a(x-1) - \sum_{i=1}^{x-1} u(i)a(x-i)}{a(0)} \quad \text{Ecuación 16}$$

La cola considerada es de tamaño finito, luego debe aparecer en el análisis un caso adicional que no se tenía en cuenta en el caso ideal, será aquel estado en el que la cola esté llena al final de la ranura de tiempo considerada - estado b -. Este será el último estado posible del sistema. Con la estrategia de gestión de colas que se considera, de ‘llegadas primero’, resulta que para que se pueda afirmar que el sistema se encuentra en el estado b, tiene que ocurrir que el búfer esté vacío al comienzo de la ranura, y que lleguen durante esa ranura de tiempo, al menos, b células.

Inicialmente se podría pensar que para alcanzar el último estado es indiferente que el sistema se encuentre inicialmente vacío y que basta con que llegue un número suficiente de células como para llenar el búfer. No obstante, si se considera el caso de que la cola no comience la ranura vacía, y suponiendo que llegasen suficientes células como para llenarla, cuando llegara el final de la ranura se produciría una salida de la cola: la célula de la cabecera, que ya llevaría al menos una ranura de tiempo esperando, se sirve. De esta forma, de nuevo la cola dejaría de estar completa, no habría b clientes esperando. Con lo cual no estaría en el estado b. Llegando a esta conclusión es posible escribir la ecuación del estado b de la forma:

$$s(b) = s(0)A(b) \quad \text{Ecuación 17}$$

Donde A(k) es la probabilidad de que lleguen al menos k células a la cola en una ranura de tiempo.

$$A(k) = 1 - a(0) - a(1) - \dots - a(k-1) \quad \text{Ecuación 18}$$

Aplicando para la Ecuación 17 el cambio de variable indicado en la Ecuación 14 se obtiene que  $u(b)=A(b)$ . Una vez llegados a este punto, ya será posible evaluar  $u(x)$  para todos los valores de  $x$  desde 0 a  $b$ , ambos inclusive.

Sin embargo con ello no se ha llegado a la expresión buscada. Para obtenerla será necesario utilizar la ecuación de balance de flujo, sabiendo que ahora no todo lo que llega se cursa. La utilización ' $\rho$ ' será el número medio de células servidas por ranura de tiempo. De esta forma

$$L = E(k) - \rho = E(k) - (1 - s(0)) = E(k) + s(0) - 1 \quad \text{Ecuación 19}$$

De nuevo,  $L$  se refiere a la tasa de células perdidas,  $\rho$  será la utilización del sistema y  $E(k)$  el número esperado de células que lleguen.

Para resolver esta última ecuación es necesario conocer el valor de  $s(0)$ , así bastará considerar que la suma de todas las probabilidades de estado,  $s(i)$ , tiene que ser igual a la unidad. Con esto se obtiene

$$s(0) = \frac{1}{\sum_{i=0}^b u(i)} \quad \text{Ecuación 20}$$

y para el resto de los  $s(x)$ , a partir de la Ecuación 14:

$$s(x) = s(0)u(x) \quad \text{Ecuación 21}$$

Con todo esto se llega a la obtención de las pérdidas,  $L$ , que indicará el número de células perdidas en cada puerto de salida del conmutador, supuestas colas de longitud  $b$ , durante una ranura de tiempo de tamaño  $s$ , y medido en células/ranura.

Para el modelado del problema, se tomará la función de pérdidas obtenida mediante este desarrollo. El valor de  $L$ , calculada en un nodo  $i$  determinado, representa las pérdidas obtenidas en el nodo para un puerto de salida concreto a partir de un tráfico ofrecido al mismo,  $\lambda$ . Estas pérdidas pasarán a ser denominadas  $\mathcal{L}_{ij}^t(\lambda)$ . La notación indica con el subíndice que se refiere al puerto de salida del conmutador  $i$  que está asociado a la salida, o puerto de salida,  $j$ , y con el superíndice se indica la localización

dentro del horizonte temporal. Esta función de pérdidas, dependerá del tráfico ofrecido a cada puerto de salida. De ahí que el parámetro que se le asocia sea  $\lambda$ .

### 8.7.1 Tamaño de la cola

Una vez concluido el análisis anterior y sabiendo la probabilidad de estado del sistema resulta sencillo obtener la probabilidad de pérdida del sistema. Ahora bien, para poderlo hacer hay que determinar el tamaño que ha de dársele a la cola. La asignación de este parámetro no resulta indiferente ni a nivel económico ni de buen funcionamiento del sistema.

Cada conmutador ATM debe tener una cola de tamaño suficiente para que el número de pérdidas no supere como norma a los valores estándares permitidos. Sin embargo, si las colas son excesivamente grandes, aunque el número de células perdidas se reduzca enormemente ocurrirá que el retraso acumulado por las unidades de información será cada vez mayor. Esto puede provocar un funcionamiento de pésima calidad, ya que en multitud de ocasiones la llegada de información con excesivo retraso puede tener efectos aún peores que los que provocaría la pérdida de los mismos (basta pensar en lo que ocurriría en una conversación telefónica si se perdiese una sílaba de una palabra y compararlo con lo que ocurriría si llegasen con retardos variables).

Dada la importancia de determinar el tamaño de los búferes correctamente se intentará al menos acotar el valor del parámetro en función del comportamiento del sistema. Es importante destacar que éste último variará dependiendo del tipo de tráfico que tenga que ser transportado. Una de las hipótesis con la que se trabajará en este documento es que el tráfico que fluye por la red es exclusivamente del tipo CBR, es decir, de tasa de bits constante

## 8.8 MODELO LINEAL DE ENCAMINAMIENTO ATM

### 8.8.1 Parámetros del modelo

Tabla 8. Representación del modelo a utilizar

Descripción/ Parámetros del modelo	Descripción general
N	Nodos de la red
E	Enlaces de la red

<b>Descripción/ Parámetros del modelo</b>	<b>Descripción general</b>
<b>M</b>	Conjunto de pares origen-destino que pueden establecer la conexión
<b>H(m)</b>	Conjunto de caminos posibles que pueden establecer una conexión entre los nodos del par
<b>T</b>	Horizonte temporal a nivel de conexión. Estará compuesto por divisiones temporales de duración de una llamada $t \in T$ . Este último será el parámetro que se utilizará a nivel de célula, $\tau \in t$ .
<b>N(h)</b>	Conjunto de nodos que forman parte del camino h
<b>E(h)</b>	Conjunto de enlaces del camino h
<b>A(i)</b>	Conjunto de nodos antecesores al nodo i.
<b>D(i)</b>	Conjunto de nodos sucesores al nodo i.

### 8.8.2 Variables del modelo

**Tabla 9. Representación de las variables del modelo**

<b>Descripción / Variables del modelo</b>	<b>Descripción general</b>
$P_h^{m,t}$	Variable binaria que indica si la conexión entre los nodos origen y destino del par m - O(m) y D(m) respectivamente- se hace por el camino h. Referido al intervalo de tiempo $t \in T$ . Por lo tanto es una variable referida al nivel de conexión.
$X_{h,ij}^{m,\tau}$	Flujo asociado al par m que va por el

Descripción / Variables del modelo	Descripción general
	enlace (i,j), y en esa dirección (de i a j), donde (i,j) es un enlace del camino h. Referido al periodo $\tau \in t$ , con lo que esta variable se considera a nivel de célula. Se expresa en células por ranura.
$I_{h,ij}^{m,\tau}$	Pérdida de células que se produce en el enlace (i,j) debido al nodo i perteneciente al camino h. Se refiere al par m, en el intervalo $\tau \in t$ (variable del nivel de célula). Expresado en células por ranura de tiempo.
$F_{ij}^{\tau}$	Flujo total de células desde el nodo i que se debería encaminar hacia el nodo j. Está referido a un intervalo de tiempo $\tau \in t$ (variable considerada a nivel de célula). Se mide en células por ranura.

La diferencia entre esta última variable y la variable de flujo  $X_{h,ij}^{m,\tau}$ , radica en que mientras que  $F_{ij}^{\tau}$  se refiere al flujo total de células que idealmente deberían atravesar el enlace (i,j) suponiendo que no se produjeran pérdidas,  $X_{h,ij}^{m,\tau}$  es el flujo real de células que se encaminan hacia j atravesando el enlace (i,j). Esta explicación se puede ilustrar con un ejemplo: Sean los nodos v, w, y, z los nodos antecesores a un nodo cualquiera i. Sean los nodos j, k los nodos sucesores al nodo en cuestión i. Supóngase que en un instante t hay flujo en dirección a i desde los cuatro nodos anteriores al mismo. El flujo proveniente de v, z debe ser encaminado hacia el nodo j, y el que llega de w y de y será encaminado hacia el conmutador k. Con estas premisas a continuación se incluye un esquema en el que se aprecia claramente la diferencia entre las dos variables  $F_{ij}^{\tau}$  y  $X_{h,ij}^{m,\tau}$

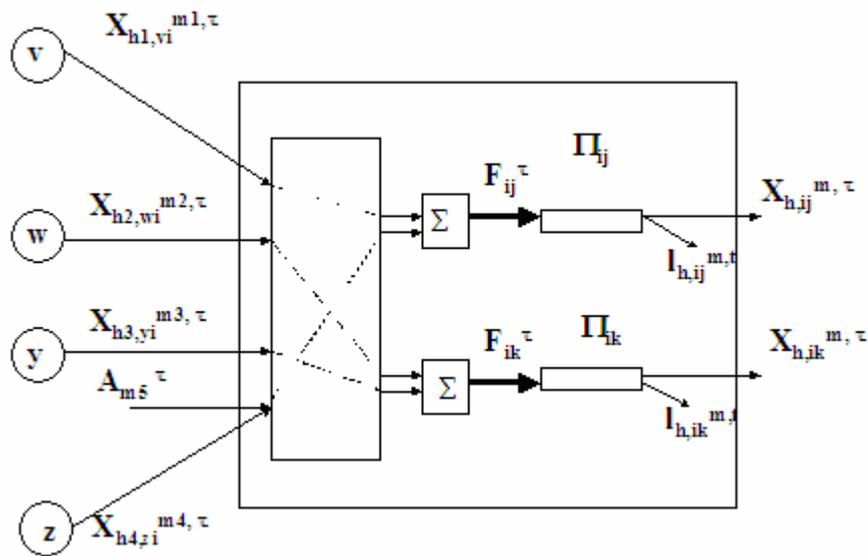


Figura 12. Representación de las distintas variables

### 8.8.3 Parámetros del modelo

Tabla 10. Tabla de parámetros del modelo

Descripción / Parámetros del modelo	Descripción general
$Q_{ij}$	Capacidad del enlace (i,j) (que une los nodos i y j), expresado en células por ranura de tiempo.
$A_m^{\tau}$	Demanda del par m en el intervalo $\tau$ . Esto es, referida al nivel de célula. Se expresa en células/ranura.
$GOS_{ cel}^m$	Máximo número de células perdidas para la conexión m, (células/ranura)

### 8.8.4 Restricciones del modelo

$$(1) \quad \sum_{h \in H(m)} P_h^{m,t} = 1 \quad \forall m \in M, \forall t \in T$$

#### Descripción de la restricción

Esta restricción obliga a que cada conexión entre el par de nodos  $O(m) - D(m)$ , se realice a través de sólo un camino. De esta forma, de entre todos los caminos posibles para abrir una conexión entre los extremos de cada par  $m$ , se tomará uno y sólo uno de acuerdo a la conmutación en modo ATM. Esto será para cada periodo de tiempo  $t$  del horizonte  $T$ . O dicho de otra manera, si  $h$  es el camino único para establecer la conexión entre dos nodos lo será durante todo el tiempo que dure la llamada.

La ecuación representa el balance de flujo sobre todo nodo de la red. El término de la izquierda muestra el flujo a través del enlace  $(i,j)$  que forma parte del camino  $h$  que une el par origen - destino  $m$ , más las pérdidas de células que se producen en el buffer de almacenamiento de salida del nodo  $i$ . El término de la derecha refleja dos situaciones: En la primera el nodo  $i$  es el nodo origen de comunicación, con lo cual incluye todo el tráfico creado en el origen y que se ha enviado por el camino  $h$ ; en la segunda, el nodo  $i$  es un nodo intermedio del camino  $h$ , así la expresión recoge todo el flujo que llega al nodo  $i$  procedente del nodo  $k$ , precedente en el camino - enlace  $(k,i) -$ . El tráfico que llega proveniente de los nodos anteriores lleva acumulado un retraso debido a los conmutadores. A pesar de que el modelo se basa en las pérdidas de los nodos de conmutación para buscar el encaminamiento apropiado, y no en el retraso introducido por los mismos, la consideración de un retraso mínimo de conmutación es imprescindible a la hora de implementar el sistema.

Nótese cómo con esta ecuación se ligan las variables de pérdidas  $l_{h,ij}^{m,\tau}$  con el camino elegido  $h$ , de forma que, si  $P_h^{m,t}$  es igual a cero (lo cual indica que  $h$  no es el camino elegido para encaminar la conexión), la correspondiente variable de pérdidas también lo será. Este balance de flujo se hace a nivel de célula.

(3)

$$F_{ij}^{\tau} = \sum_{m \in M} \sum_{\substack{h \in H(m), \\ (q,i) \in E(h): j \in N(h)}} \sum_{q \in A(i)} X_{h,qi}^{m,\tau-1} + \sum_{m \in M / i=0(m)} A_m^{\tau} \quad \forall (i,j) \in E, \forall \tau \subset t, \forall t \in T$$

## Descripción de la restricción

En la ecuación anterior se aplica la definición de la variable  $F_{ij}^\tau$ . Como se puede ver en la igualdad, El flujo total que comprende la variable incluirá los flujos procedentes de aquellos nodos predecesores al  $i$  que deban ser encaminados hacia el  $j$  para poder alcanzar sus respectivos destinos así como los generados en ese mismo nodo por ser origen de una conexión. Se vuelve a considerar el retraso producido por el conmutador anterior en el flujo entrante de la red. La explicación de este retraso ha sido detallada anteriormente.

$$(4) \quad \sum_{m \in M} \sum_{h \in H(m)} l_{h,ij}^{m,\tau} \geq \mathcal{L}_{ij}^\tau (F_{ij}^\tau) \quad \forall i \in N, \forall (i, j) \in E, \forall \tau \subset t, \forall t \in T$$

## Descripción de la restricción

Con esta restricción se obliga a que las variables  $l_{h,ij}^{m,\tau}$  reflejen al menos las pérdidas debidas al desbordamiento de las colas de los puertos de salida. Se utiliza la expresión gramatical ‘al menos’ porque, como se puede ver, la restricción indica que las pérdidas reales contabilizadas para cada puerto de salida deben ser mayores o iguales que las pérdidas estimadas para ese puerto por la función de pérdidas  $\mathcal{L}_{ij}^\tau$ . Así, se consideran otras posibles causas de pérdidas de células en las redes ATM, que aunque no son ni con mucho tan importantes como las pérdidas debidas a desbordamiento en los búferes, también existen y ya las hemos indicado anteriormente. No obstante, en las cercanías del óptimo la holgura de la restricción tiende a reducirse para los caminos que constituyen la solución básica.

A continuación, se pasa a detallar los sumatorios que componen la ecuación (3) del modelo. La restricción se aplica nodo a nodo, y dentro de cada nodo se aplica en cada puerto de salida, de ahí que se escriba  $\forall i \in N, \forall (i, j) \in E$ .

En el término de la izquierda, se suma el valor total de las pérdidas en el nodo  $i$  con un doble sumatorio, el primero es para todos los caminos posibles de una conexión, de forma que sea cual sea el camino, si la conexión pasa por el nodo que se está considerando, se contabilicen las pérdidas. En el segundo sumatorio, se termina de contabilizar las pérdidas producidas en el puerto de salida  $\Pi_{ij}$  del nodo  $i$ , ya que se suma para todas las conexiones abiertas. Es decir, todas las conexiones que pasen por el puerto  $\Pi_{ij}$  contribuirán con sus pérdidas a las pérdidas totales del puerto.

En el término de la derecha, se calculan las pérdidas que se producen para una de las colas de salida del nodo que se está considerando, la que lleva al nodo  $j$ . Utiliza como parámetro el tráfico que pasa por el nodo  $i$ , y tiene como siguiente salto el nodo  $j$ ,

independientemente de cual sea su destino final k y del camino h correspondiente que esté utilizando para alcanzarlo.

La función  $L$ , que ya fue definida anteriormente, está muy relacionada con la estructura interna de los conmutadores. Dependerá de si estos tienen un tamaño de buffer que pueda ser considerado infinito o no, y sobre todo de la situación de las colas contemplada. En este desarrollo se ha considerado que las colas se encuentran a la salida de los conmutadores, ya que ofrecen mejor relación retraso - rendimiento.

$$(5) \quad \sum_{m \in M} \sum_{h \in H(m): (i,j) \in E(h)} X_{h,ij}^{m,\tau} \leq Q_{ij} \quad \forall (i,j) \in E, \forall \tau \subset t, \forall t \in T$$

### Descripción de la restricción

Con esta restricción se impone que el flujo total por un enlace cualquiera (i,j), que une los nodos i y j, no supere la capacidad del mismo.

$$(6) \quad \sum_{\tau \in t} \sum_{h \in H(m)} \sum_{(i,j) \in E(h): j \in D(i)} I_{h,ij}^{m,\tau} \leq GOS_{cel}^m \quad \forall m \in M, \forall t \in T$$

### Descripción de la restricción

Condición restrictiva sobre las pérdidas. Se impone el grado de servicio contratado por una conexión como límite máximo de células perdidas en todo el camino que recorre el flujo de una conexión. De esta forma, para cada par m se cumplirá que el conjunto de pérdidas que se producen a lo largo de todo el trayecto h no supere la cota impuesta. Debe quedar claro, que no se impone la condición para cada nodo, sino para cada conexión m abierta. Se va sumando lo perdido en cada nodo del camino.

Se argumenta, a continuación que no es necesaria la inclusión de la variable multiplicando a la variable de pérdidas, de forma que se sumaran pérdidas sólo en caso de que el camino h considerado fuera el camino real por el que van las células de la conexión. En cuyo caso la ecuación quedaría como sigue:

$$\sum_{h \in H(m)} \sum_{(i,j) \in E(h): j \in D(i)} I_{h,ij}^{m,\tau} P_h^{m,t} \leq GOS_{cel}^m \quad \forall m \in M, \forall t \in T$$

En efecto, la inclusión de las variables  $P_h^{m,t}$  sólo llevaría a considerar una no linealidad no necesaria. Las variables  $l_{h,ij}^{m,\tau}$  se relacionan con el camino  $h$  seguido y con la conexión  $m$  que se trata en cada momento, de forma que, cuando la ruta seguida para una conexión no sea la correcta, la correspondiente variable  $l_{h,ij}^{m,\tau}$  tendrá valor nulo. De esta forma, cuando se suma para un camino  $h$  que no pertenezca a la ruta válida de la conexión  $m$ , la variable de pérdidas correspondiente no aportará nada al sumatorio. Estará fijada a cero y verificará la restricción. Así, se concluye que la inclusión de la no linealidad indicada en la ecuación es innecesaria.

Finalmente, se plantean las especificaciones del modelo (7), (8) y (9):

$$(7) \quad l_{h,ij}^{m,\tau} \geq 0 \quad \forall (i, j) \in E, \forall m \in M, \forall \tau \subset t, \forall t \in T$$

$$(8) \quad X_{h,ij}^{m,\tau} \geq 0 \quad \forall (i, j) \in E, \forall m \in M, \forall \tau \subset t, \forall t \in T$$

Evidentemente, ni las pérdidas ni el flujo de células pueden ser negativos

$$(9) \quad P_h^{m,t} \in (0,1) \quad \forall m \in M, \forall h \in H(m), \forall t \in T$$

Como ya se había dicho,  $P_h^{m,t}$  es una variable binaria.

### 8.8.5 Función Objetivo

En el caso discutido, el encaminamiento óptimo se determina atendiendo a las pérdidas en los conmutadores, intentando minimizarlas. Por lo tanto, si se tienen en cuenta todas las conexiones que soportará la red, la función objetivo requerida es la que sigue:

$$MIN \sum_{t \in T} \sum_{\tau \in t} \sum_{(i,j) \in E} \sum_{m \in M} \sum_{h \in H(m)} l_{h,ij}^{m,t}$$

Como se puede leer, se pretende minimizar las pérdidas totales en todos los enlaces y para todo el horizonte.

La función objetivo busca una red en la cual el número de pérdidas totales sea mínimo. Para conseguirlo, se minimizan las pérdidas nodo a nodo y para cada puerto de salida. Para conseguir esto, habrá que contabilizar como sigue:

1. Sumando para todos los caminos  $h$  disponibles para la conexión  $m$ . De esta forma se asegura la contabilización de las pérdidas, ya que sólo se establece un camino por conexión.
2. Sumando para todas las conexiones  $m$  abiertas que atraviesan el nodo. Así se contabilizan las pérdidas que se producen en un puerto de salida independientemente de su origen y destino final. Contabilizándose todas las pérdidas para cada puerto de salida en cada uno de los nodos
3. Sumando para todos los enlaces que salen de cada nodo, así se tienen en cuenta las pérdidas debidas a cualquiera de los puertos de salida (o colas) de cada uno de los nodos.
4. Sumando para cada uno de los instantes que componen el horizonte temporal (esto se hace con los dos sumatorios, uno para cada una de las escalas de tiempo).

### 8.8.6 Modelo

El modelo final queda plasmado en la Tabla 11.

**Tabla 11. Tabla representativa del la función objetivo y las respectivas restricciones del modelo**

Ecuaciones /Modelo	Descripción de las restricciones y la Función Objetivo
<b>Función Objetivo</b>	$MIN \sum_{t \in T} \sum_{\tau \in T} \sum_{(i,j) \in E} \sum_{m \in M} \sum_{h \in H(m)} l_{h,ij}^{m,t}$
<b>R1</b>	$\sum_{h \in H(m)} P_h^{m,t} = 1 \quad \forall m \in M, \forall t \in T$

Ecuaciones /Modelo	Descripción de las restricciones y la Función Objetivo
R2	$X_{h,ij}^{m,\tau} + l_{h,ij}^{m,\tau} = \begin{cases} A_m^\tau P_h^{m,t} & \text{si } i = O(m) \\ X_{h,ki}^{m,\tau-1} & \text{si } i \neq O(m) \end{cases}$ $\forall \tau \subset t, \forall t \in T, \forall h \in H(m), \forall m \in M, \forall i \in N: \forall (i,j) \in E(h), \forall (k,i) \in E(h), i,j,k \in N(h)$
R3	$F_{ij}^\tau = \sum_{m \in M} \sum_{\substack{h \in H(m), \\ (q,i) \in E(h); j \in N(h)}} \sum_{q \in A(i)} X_{h,qi}^{m,\tau-1} + \sum_{m \in M / i=O(m)} A_m^\tau \quad \forall (i,j) \in E, \forall \tau \subset t, \forall t \in T$
R4	$\sum_{m \in M} \sum_{h \in H(m)} l_{h,ij}^{m,\tau} \geq \mathcal{L}_{ij}^\tau(F_{ij}^\tau) \quad \forall i \in N, \forall (i,j) \in E, \forall \tau \subset t, \forall t \in T$
R5	$\sum_{m \in M} \sum_{h \in H(m); (i,j) \in E(h)} X_{h,ij}^{m,\tau} \leq Q_{ij} \quad \forall (i,j) \in E, \forall \tau \subset t, \forall t \in T$
R6	$\sum_{\tau \in t} \sum_{h \in H(m)} \sum_{(i,j) \in E(h); j \in D(i)} l_{h,ij}^{m,\tau} \leq GOS_{cel}^m \quad \forall m \in M, \forall t \in T$
R7	$l_{h,ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t, \forall t \in T$
R8	$X_{h,ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t, \forall t \in T$
R9	$P_h^{m,t} \in (0,1) \quad \forall m \in M, h \in H(m), t \in T$



## 9 RESOLUCIÓN DEL MODELO

### 9.1 ALGORITMO GENÉTICO

Como método de solución se ha optado por la utilización de algoritmos genéticos. Así, se obtendrá una buena solución para el problema de encaminamiento en tiempos de ejecución razonables.

Los algoritmos genéticos (A.G) son algoritmos que pretenden ofrecer soluciones aproximadas a problemas de gran complejidad computacional mediante un proceso de 'evolución simulada'. Se basan en la búsqueda y comparación de individuos para solucionar los problemas que se les plantean. Inicialmente, el algoritmo busca un número de individuos que cumplan las restricciones del sistema (soluciones posibles), incorporándolos como población inicial. Luego, basándose en la teoría de la evolución de las especies y por medio de la comparación de sus individuos es capaz de generar otros nuevos que se asemejarán a los mejores y los peores serán eliminados de la población.

Siguiendo el razonamiento anterior, a medida que se avanza en la resolución irán desapareciendo de la población los peores individuos, propiciándose la aparición de individuos cada vez mejores, hasta llegar al óptimo. Sin embargo, en los algoritmos genéticos se introduce una componente de aleatoriedad a la hora de reemplazar los individuos, característica que se presenta también en la selección natural. De esta forma, los nuevos individuos que aparecen en cada generación no siempre sustituirán a los peores de la generación que les precedió. Se considera por lo tanto la posibilidad de que los individuos reemplazados presenten mejores aptitudes que los que se mantienen.

Así, se concluye que mediante la utilización de los A.G la obtención de una buena solución está asegurada. Sin embargo, ésta no tiene porqué ser la solución óptima del problema.

Como se ha indicado los algoritmos genéticos tratan de explotar los mecanismos evolutivos presentes en la naturaleza. Utilizan asimismo conceptos presentes en la evolución natural para explorar el espacio de búsqueda. De este modo, incorporan el concepto de cromosoma como elemento descriptor del individuo, que lo caracteriza y que a su vez contiene toda la información necesaria sobre el mismo. De esta forma, en el algoritmo genético resulta equivalente hablar de un individuo que de la carga genética que lo caracteriza y determina; esto es, el cromosoma.

Asimismo, existen otros términos genéticos incorporados por los A.G de entre los que se destacan la mutación y el cruce de individuos. Ambos son mecanismos asociados a la variación del material genético de los padres conformando el de los hijos. Las mutaciones aleatorias proveen cierta variación, y en ocasiones introducen alteraciones beneficiosas en los cromosomas. El cruce es el mecanismo mediante el cual las características de los padres se combinan cuando éstos se reproducen dando lugar a los genes del hijo. Es este último mecanismo el que gobierna la eficiencia de los algoritmos genéticos.

En los A.G deben aparecer una serie de características que se dan en los procesos de selección natural, como aparece en la Tabla 12.

**Tabla 12. Tabla representativa de las características básicas de los algoritmos genéticos**

<b>CARACTERÍSTICAS BÁSICAS DE LOS ALGORITMOS GENÉTICOS</b>
Los cromosomas que codifican estructuras con éxito se reproduzcan más frecuentemente que aquellos que no lo hacen
La evolución es un proceso que opera en los cromosomas en lugar de en los seres vivos que estos codifican
Las mutaciones deben ser consideradas como fuentes causantes de diferencias entre los cromosomas de los hijos y de sus progenitores.
Los procesos de recombinación podrán crear cromosomas bastante diferentes en los hijos por la combinación de material genético de los dos padres
La evolución biológica no tiene memoria

Como ocurre en la evolución biológica, la evolución simulada que llevan a cabo los algoritmos genéticos estará diseñada para encontrar cada vez mejores cromosomas mediante una manipulación ciega de sus contenidos. El término ciega se refiere al hecho de que el proceso no tiene ninguna información acerca del problema que está tratando de resolver, exceptuando el valor de la función objetivo. La función objetivo es la única información por la que se evalúa la aptitud de un cromosoma (fitness en inglés). Por todo ello, se considera de importancia fundamental implementar el cromosoma del individuo de una forma sencilla y de manera que sea posible la evaluación del mismo, esto es, de calidad de la solución admisible codificada.

### **9.1.1 Población en un algoritmo genético**

Durante todo el proceso evolutivo del algoritmo se mantiene una población de cromosomas en la que todos los individuos serán potencialmente padres de la nueva población que compondrá la siguiente generación. A cada uno de ellos se le adjudica un

valor de aptitud que surgirá de evaluar la función objetivo a optimizar, así se establece un ranking en el que cada individuo ocupa un puesto más alto o menos alto en función de su aptitud. Cada cromosoma representará por lo tanto un punto en el espacio de búsqueda del problema.

Dos cromosomas (que no serán más que cadenas o matrices binarias), son seleccionados para ser los padres de una nueva solución mediante el mecanismo de cruce. La elección de los padres puede ser en función de su posición en el ranking de los individuos o por métodos aleatorios. La operación de cruce dará lugar a dos estructuras de hijo de la cual se elegirá sólo una (de nuevo la elección de la misma puede ser después de su evaluación y en función de la aptitud de ambas o bien por métodos heurísticos) la cual sustituirá a un individuo de la población elegido al azar.

El algoritmo implementado, realiza la elección de los padres antes de un cruce de forma aleatoria. Para la sustitución de un individuo ante la llegada de un nuevo hijo utiliza un método basado en las aptitudes de los individuos.

## 9.2 CODIFICACIÓN DE LAS SOLUCIONES

### 9.2.1 Individuos

La codificación de un individuo representa la estructura de rutas elegida para cada conexión dentro de las rutas posibles existentes en la red de telecomunicaciones. Cada individuo codificado se evalúa en base a un criterio de pérdida de células por la red y se combina con los demás individuos buscando la mejora de la especie.

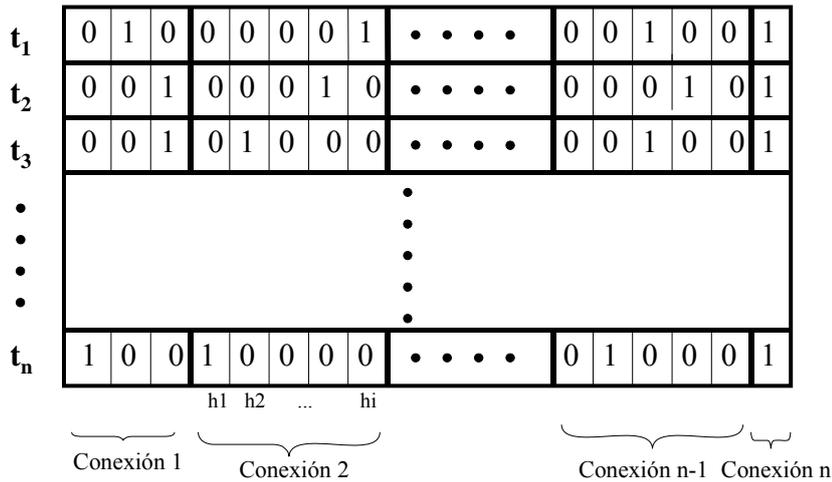
El código genético asociado a cada individuo vendrá determinado por una matriz con elementos binarios, como puede verse en Tabla 13.

**Tabla 13. Descripción del cromosoma del individuo**

<b>DESCRIPCIÓN DEL CROMOSOMA DEL INDIVIDUO</b>
Cada individuo contenga en su matriz de código genético tantas filas como periodos de conexión se tomen en el intervalo de tiempo considerado. En principio, $t$ se considera el tiempo medio de duración de una conexión ATM.
A su vez, cada una de las filas está compuesta por un determinado número de campos, $M$ , que representan los distintos pares origen-destino que pueden dar lugar a una conexión. Esto depende de la propia red.
El campo asociado a un par se forma con tantos elementos como caminos posibles se hayan encontrado entre ese origen concreto y el correspondiente destino de la

## DESCRIPCIÓN DEL CROMOSOMA DEL INDIVIDUO

comunicación. Cada uno de estos elementos representa a una ruta posible. Sólo tomará el valor 1 o 0.



**Figura 13. Representación del código genético del individuo**

En la Figura 13 se observa que puede existir más de un elemento con valor uno en una fila de la matriz que codifica al individuo, pero sólo puede haber un elemento distinto de cero para cada conexión dentro de una fila (o instante  $t$ ), la explicación para esta limitación es evidente si se tiene en cuenta la estructura de la matriz: Cada fila de la matriz representa un instante de tiempo  $t$ . Dentro de cada fila, se consideran las distintas conexiones que se pueden establecer durante ese periodo de tiempo. A su vez, las distintas conexiones consideran los distintos caminos por los que pueden llegar al destino y sólo se elige una de ellos como ruta de la conexión. Es por lo tanto imposible que dos columnas dentro de una misma conexión y para el mismo instante de tiempo estén marcadas como activas (a uno). Cada una de las rutas activas de las conexiones se evaluarán teniendo en cuenta la influencia del tráfico de otras rutas activas, esto se debe a que los arcos pueden estar compartidos.

### 9.2.2 Población

Para que el algoritmo genético pueda evolucionar hacia una buena solución debe trabajar con una población de individuos capaces de combinar su material genético de forma que resulten nuevos individuos hijos con mejor aptitud. El algoritmo trabajará

siempre con una población de tamaño constante. El tamaño de la población se elige de forma que no sea demasiado pequeña, de forma que se permita la existencia de riqueza genética en el material genético de los individuos presentes, ni excesivamente grande, ya ambos extremos supondrían una carga computacional innecesaria. Se toma un tamaño de población N que cumpla estas premisas.

La población de partida del algoritmo genético se generará de forma aleatoria.

Si se tiene en cuenta la estructura del código genético de un individuo, el de la población completa para cada iteración del algoritmo se representará como la Figura 14:

Individuo 1	$t_1$	0010	010	10000	10	•••	0001	01
	$t_2$	0001	010	00100	01	•••	0100	01
	$t_3$	1000	100	01000	01	•••	0010	10
					•			
Individuo 2	$t_n$	0100	001	00010	10	•••	0100	10
	$t_1$	0010	010	10000	10	•••	0001	01
	$t_2$	0010	100	00010	01	•••	1000	10
	$t_3$	1000	001	01000	01	•••	0010	01
Individuo N					•			
					•			
					•			
	$t_n$	0010	001	10000	10	•••	0100	01

Figura 14. Representación de población de N individuos

### 9.3 OPERADORES

Los operadores que se utilizan sobre los individuos de la población son el cruce y la mutación. El algoritmo elige en cada iteración el método de generación del nuevo individuo hijo y lo hará de forma aleatoria. La probabilidad de que se produzca cruce se denominará pc, y la probabilidad de mutación será por lo tanto 1- pc. Si la probabilidad de cruce es alta, el algoritmo tiende a explotar a fondo las soluciones del espacio de búsqueda en el que se encuentra, sin tender a cambiar la zona de búsqueda. Por el contrario, con la mutación se tiende a explorar nuevas regiones del espacio de soluciones admisibles.

Habitualmente se utiliza una probabilidad de mutación de individuos menor que la de cruce entre individuos.

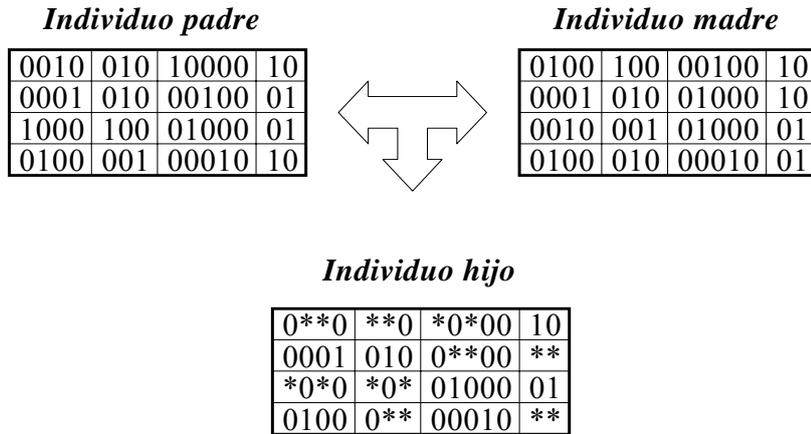
### 9.3.1 Operador cruce

En caso de que se produzca un cruce, el nuevo hijo se obtiene a partir de la combinación del material genético de los individuos padres. En el algoritmo implementado, la selección de los padres se hace de forma aleatoria, propiciando de esta forma el enriquecimiento genético y una mayor exploración de la región.

Una vez elegidos los padres, se adopta como mecanismo de cruce el cruce uniforme, de forma que los genes de cada padre se transmiten al hijo con la misma probabilidad que los del otro. Así, si ocurriese que los dos individuos padres presentaran una misma ruta activa para el mismo intervalo de conexión  $t$  y para el mismo par de conexión  $m$ , el individuo hijo resultante también presentaría esta ruta activa para esa conexión en ese intervalo. En caso de que los padres tengan para un mismo instante  $t$  y para una misma conexión  $m$  distinta ruta activa, el hijo podría presentar cualquiera de las dos rutas activas (nunca las dos). La elección en este caso sería de nuevo aleatoria. El individuo hijo presentará activa la misma ruta que su ‘padre’ con probabilidad  $p=0.5$ , y con idéntica probabilidad presentará la de su ‘madre’.

Al realizar cruces entre individuos se contempla la prevención de incesto. De esta forma se controla el mecanismo de elección de progenitores evitando el cruce entre individuos muy parecidos. Con esto se fomenta un enriquecimiento del material genético con el que trabaja el algoritmo. Por la misma razón, habrá que evitar el hecho de que el individuo hijo generado sea una réplica de alguno de sus progenitores. Esto podría ocurrir debido a la característica aleatoria de la selección del material genético que hereda el nuevo individuo de cada uno de los padres.

La Figura 15 que se incluye a continuación representa el funcionamiento del operador de cruce de individuos:



*Aquel campo donde aparecen \* \* es con 50% de probabilidad igual al campo de la madre, con la misma probabilidad igual al padre*

**Figura 15. Representación del operador cruce**

### 9.3.2 Operador mutación

La mutación da lugar a un nuevo individuo a partir de un cambio en la información genética de otro individuo ya existente en la población. Como se ha mencionado, mediante este operador se introducen en el algoritmo nuevas áreas de exploración enriqueciendo así el material genético de los individuos de la población.

Si el algoritmo concluye que se produzca una mutación en lugar de un cruce será necesaria la elección de un individuo, cuyos genes han de ser mutados. La elección del individuo, al igual que en el caso del cruce, será aleatoria.

En el caso de la mutación, se cambiará la ruta activa de dos conexiones para cada una de las filas que componen la matriz genética del individuo. Esto es, la mutación será fuerte: Cambiará la ruta elegida para una conexión cualquiera en cada uno de los intervalos de conexión existentes. Se ha elegido este tipo de mutación en lugar de una más débil debido a que de esta forma el salto, o cambio, de región de búsqueda es mayor dotando así a los individuos de una mayor riqueza genética.

### 9.3.3 Reemplazo de individuos

Como ya se indicó, el algoritmo genético que se ensaya trabaja con una población constante. Debido a ello, cada vez que se genera un hijo éste sustituye a

alguno de los individuos preexistentes en la población. Se implementa así un algoritmo genético incremental.

La estrategia de reemplazo elegida es distinta según el individuo hijo haya sido obtenido por medio de un cruce o de una mutación. En el caso de la mutación, se opta porque el individuo mutado sustituya al que le dio origen, mientras que en el caso del cruce, el criterio de reemplazo de individuos es distinto y está basado en la aptitud de los individuos.

La ley que sigue la probabilidad de reemplazo de los individuos en función del ranking es una ley hipergeométrica truncada cuyo esquema se representa a continuación:

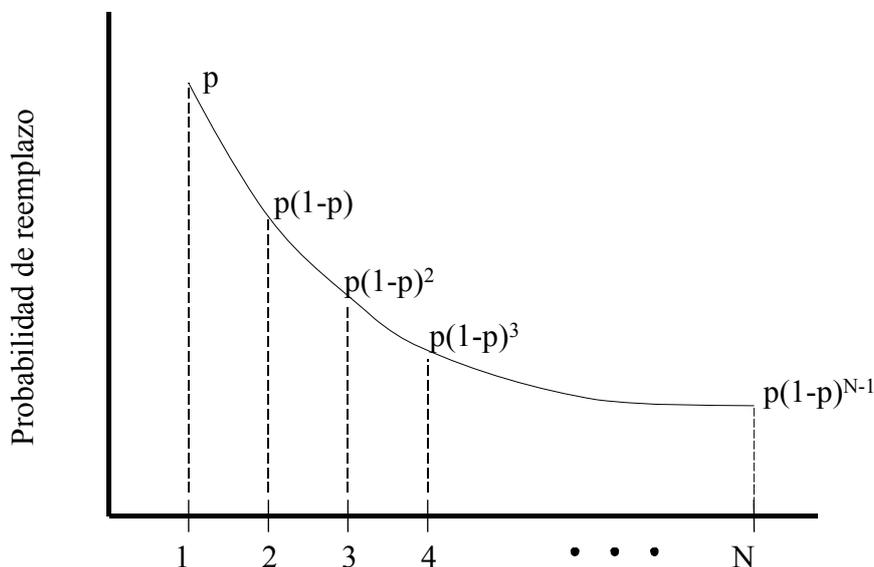


Figura 16. Representación de la probabilidad de reemplazo de los individuos

Por ser truncada, existe una probabilidad residual que se sumará al individuo de menor aptitud. La probabilidad residual se obtiene sabiendo que:

$$\sum_{n=0}^{\infty} p(1-p)^n = 1$$

**Ecuación 22**

Como en el caso que se trata la función está truncada en N, la suma hasta N de la función de probabilidad será:

**Ecuación 23**

$$\sum_{n=0}^n p(1-p)^n = 1 - (1-p)^{N+1}$$

Por lo que la probabilidad residual quedaría:

$$p_{res} = (1-p)^{N+1}$$

**Ecuación 24**

Como se aprecia en la gráfica, el orden de los individuos comienza en el 1 y no en el 0. Si se mantiene la suposición de que la lista de elementos ordenados se extiende hasta N la fórmula que calcula la probabilidad residual queda:

$$p_{res} = (1-p)^N$$

**Ecuación 25**

Podría haberse tomado una probabilidad de selección del individuo a reemplazar proporcional a la aptitud, ya que también cumpliría que un individuo con alta aptitud tuviera menos posibilidades de ser sustituido por el nuevo individuo que aquel con una aptitud baja. Sin embargo, la ley hipergeométrica elegida se basa en el ranking de individuos, de forma que dos individuos próximos en aptitud tengan probabilidades de reemplazo muy distintas. Esto es útil sobre todo en los últimos escalones del algoritmo, donde las aptitudes de los individuos tenderán a ser más parecidas

#### **9.3.4 Criterio de parada del algoritmo genético**

La detención del algoritmo puede tener lugar debido a dos causas: que se haya completado el número de iteraciones admisibles, N\_MAX, o que el grado de semejanza de los individuos sea muy elevado.

El número de iteraciones máxima será el criterio de parada por defecto, de forma que una vez fijado N\_MAX el algoritmo se detendría evitando que se superase este límite preestablecido. Por otra parte, en el algoritmo se lleva a cabo un control de semejanza de los individuos que se haya altamente relacionado con el control de prevención de incesto que se ha comentado anteriormente, de forma que si el grado de semejanza entre los individuos crece demasiado, se considera que se ha conseguido un

cierto grado de convergencia dando lugar a la detención del procedimiento, independientemente del número de iteraciones llevadas a cabo.

#### 9.4 APTITUD DE LOS INDIVIDUOS

Calcular la aptitud de cada individuo implica la evaluación la solución que este representa tomando como referencia la función objetivo del modelo que se quiere resolver. Esto es, se pretende saber ‘cómo de buena es una solución’ para poder establecer un criterio de comparación entre individuos, y de esta forma poder elegir el mejor.

Al evaluar la aptitud del individuo se pretende establecer la calidad de los caminos  $P_h^{m,t}$  desde el punto de vista de la pérdida de células .

El modelo equivalente una vez fijado  $P_h^{m,t}$  queda mucho más sencillo, siendo posible resolver el problema de pérdidas de la red fácilmente mediante la evaluación de la función objetivo para cada una de las soluciones obtenida. Así, el modelo una vez fijado  $P_h^{m,t}$  quedaría como puede apreciarse en la Tabla 14:

**Tabla 14. Tabla representativa de las restricciones del modelo**

Ecuaciones / Restricciones	Ecuaciones de las restricciones del modelo
Restricción	$X_{ij}^{m,\tau} + I_{ij}^{m,\tau} = \begin{cases} A_m^\tau & \text{si } i = O(m) \\ X_{ki}^{m,\tau-1} & \text{si } i \neq O(m) \end{cases}$ $\forall \tau < t, \forall m \in M, \forall i \in N: \forall (i, j) \in E(h), \forall (k, i) \in E(h), / i, j, k \in N(h)$
Restricción	$F_{ij}^\tau = \sum_{m \in M} \sum_{q \in A(i)} X_{qi}^{m,\tau-1} + \sum_{m \in M / i=O(m)} A_m^\tau \quad \forall (i, j) \in E(h): q \in A(i), \forall \tau < t$
Restricción	$\sum_{m \in M} I_{ij}^{m,\tau} \geq \mathcal{L}_{ij}^\tau(F_{ij}^\tau) \quad \forall i \in N, \forall (i, j) \in E(h), \forall \tau < t$

<b>Ecuaciones / Restricciones</b>	<b>Ecuaciones de las restricciones del modelo</b>
<b>Restricción</b>	$\sum_{m \in M} X_{ij}^{m,\tau} \leq Q_{ij} \quad \forall (i,j) \in E(h), \forall \tau \subset t$
<b>Restricción</b>	$\sum_{\tau \in t} \sum_{(i,j) \in E(h)} l_{ij}^{m,\tau} \leq GOS_{cel}^m \quad \forall m \in M$
<b>Restricción</b>	$l_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$
<b>Restricción</b>	$X_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$

Antes, la primera restricción representaba la asignación un único camino a cada par. Evidentemente, una vez fijado el camino, pierde este significado. Ahora, la primera restricción representa un balance asociado al caso en el cual el camino  $h$  se elige para transportar la comunicación del par  $m$ .

#### 9.4.1 Simplificación del modelo

Antes de abordar la resolución del modelo se tienen en cuenta una serie de suposiciones que permiten la simplificación del mismo.

Lo primero que se supone es que la capacidad de los enlaces siempre superará la posible demanda de utilización de los mismos. Esto es, se consideran todos los enlaces dimensionados de forma que a la hora de enviar tráfico, nunca se supere la capacidad disponible. Con ello, la cuarta restricción desaparece. Esta suposición no excesiva, ya que normalmente las redes de telecomunicaciones están sobredimensionadas, de forma que la capacidad real disponible supera con creces la demanda. La razón fundamental que lleva al sobredimensionamiento de este tipo de redes es la previsión de futuros incrementos en el tráfico. Esto puede ocurrir bien porque tengan lugar nuevas altas de usuarios, o bien debido a la aparición de nuevos servicios que requieran mayor ancho de banda. Si por algún motivo la red no tuviese capacidad sobrante para acoger este aumento en el tráfico, habría que volver a abrir las zanjas del suelo e insertar nuevos cables. Algo que requeriría una muy alta inversión económica por parte de la empresa explotadora.

**Tabla 15. Tabla representativa del modelo simplificado**

<b>Ecuaciones / Restricciones</b>	<b>Ecuaciones de las restricciones del modelo</b>
<b>Restricción</b>	$X_{ij}^{m,\tau} + l_{ij}^{m,\tau} = \begin{cases} A_m^\tau & \text{si } i = O(m) \\ X_{ki}^{m,\tau-1} & \text{si } i \neq O(m) \end{cases}$ $\forall \tau \subset t, \forall m \in M, \forall i \in N: \forall (i,j) \in E(h), \forall (k,i) \in E(h), /i,j,k \in N(h)$
<b>Restricción</b>	$F_{ij}^\tau = \sum_{m \in M} \sum_{q \in A(i)} X_{qi}^{m,\tau-1} + \sum_{m \in M / i=O(m)} A_m^\tau \quad \forall (i,j) \in E(h): q \in A(i), \forall \tau \subset t$
<b>Restricción</b>	$\sum_{m \in M} l_{ij}^{m,\tau} \geq \mathcal{L}_{ij}^\tau (F_{ij}^\tau) \quad \forall i \in N, \forall (i,j) \in E(h), \forall \tau \subset t$
<b>Restricción</b>	$l_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$
<b>Restricción</b>	$X_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$

Una nueva simplificación se produce al excluir la quinta ecuación. Ésta resulta ser prescindible al evaluar la aptitud si se tiene en cuenta que actúa únicamente como un ‘control de admisibilidad de soluciones’. Luego, para la obtención de las soluciones del problema de encaminamiento, se puede obviar esta restricción como tal y considerarla al final sometiendo las soluciones al debido examen. Concretamente, toda solución que no cumpla la restricción que se refiere al número máximo de pérdidas por trayecto, no podrá ser considerada como admisible. Esto puede quedar reflejado en el algoritmo genético penalizando la aptitud del individuo en cuestión.

Considerando las dos suposiciones anteriores, las restricciones del modelo a resolver se reducen a:

Agrupando las restricciones 2 y 3, el problema a resolver quedaría:

**Tabla 16. Tabla representativa del modelo final**

Ecuaciones / Restricciones	Ecuaciones de las restricciones del modelo
R1	$X_{ij}^{m,\tau} + I_{ij}^{m,\tau} = \begin{cases} A_m^\tau & \text{si } i = O(m) \\ X_{ki}^{m,\tau-1} & \text{si } i \neq O(m) \end{cases}$ $\forall \tau \subset t, \forall m \in M, \forall i \in N: \forall (i,j) \in E(h), \forall (k,i) \in E(h), / i,j,k \in N(h)$
R2	$\sum_{n \in M} I_{ij}^{n,\tau} \geq L_{ij}^\tau \left( \sum_{m \in M} \sum_{q \in A(i)} X_{qi}^{m,\tau-1} + \sum_{m \in M / i=O(m)} A_m^\tau \right) \quad \forall i \in N, \forall (i,j) \in E(h): q \in A(i), \forall \tau \subset t$
R3	$I_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$
R4	$X_{ij}^{m,\tau} \geq 0 \quad \forall (i,j) \in E, \forall m \in M, \forall \tau \subset t$

## 9.5 ALGORITMO PARA LA OBTENCIÓN DE LA APTITUD DEL INDIVIDUO

Se pretenden evaluar las soluciones obtenidas mediante el algoritmo genético, para cada t del intervalo T considerado y para cada par de nodos m.

Los pasos seguidos en el algoritmo para el cálculo de las aptitudes serán:

Para cada  $\tau$  contenido en t:

$$\text{- Calcular } \sum_{m \in M} \sum_{\substack{q \in A(i) \\ q,i,j \in h}} X_{qi}^{m,\tau-1} + \sum_{m \in M / i=O(m)} A_m^\tau = F_{ij}^\tau$$

- Llamado  $l_{ij}^\tau = \sum_{m \in M} l_{ij}^{m,\tau}$  y utilizando como parámetro de la función

de pérdidas la variable calculada en 1.1, se obtiene cada  $l_{ij}^\tau = \mathcal{L}_{ij}^\tau (F_{ij}^\tau)$ .

- Una vez calculada en el paso anterior la variable de pérdidas, se obtienen las pérdidas que se producen para cada conexión suponiendo que éstas se producen proporcionalmente a la demanda del par. Con esto

$$l_{ij}^{m,\tau} = l_{ij}^\tau \left( \frac{A_m^\tau}{\sum_{i,j \in h(m)} A_m^\tau} \right)$$

- Se pueden ir calculando los nuevos  $X_{ij}^{m,\tau}$  mediante la ecuación 2 de balance de flujo a partir de los valores del instante anterior y sustituyendo los valores de las pérdidas obtenidas:

$$X_{ij}^{m,\tau} + l_{ij}^{m,\tau} = \begin{cases} A_m^\tau & \text{si } i = O(m) \\ X_{ki}^{m,\tau-1} & \text{si } i \neq O(m) \end{cases}$$

$$\forall i \in N: \forall (i,j) \in E(h), \forall (k,i) \in E(h), / i,j,k \in N(h)$$

Calcular la aptitud como:

$$\sum_{t \in T} \sum_{\tau \in T} \sum_{(i,j) \in E(h)} \sum_{m \in M} l_{ij}^{m,\tau}$$

Comprobar que se verifica

$$\sum_{\tau \in T} \sum_{(i,j) \in E(h)} l_{ij}^{m,\tau} \geq GOS_{cel}^m$$

Si se cumple es que la solución es admisible, si no lo es, se penalizará al individuo. La penalización se establece incrementando un 80% las pérdidas obtenidas por la ruta.

Se supone que al iniciar la búsqueda la red está vacía, por lo que las variables que contienen el valor del tráfico enviado de un nodo a otro tienen valor cero.

## **9.6 ALGORITMO DE SELECCIÓN DE RUTA MÍNIMA**

Entendemos por mejor ruta aquella que cumple las siguientes condiciones:

1. Presenta el menor retardo medio de tránsito.
2. Consigue mantener acotado el retardo entre pares de nodos de la red.
3. Se obtienen las pérdidas más bajas.

El criterio más sencillo es elegir el camino más corto, es decir la ruta que pasa por el menor número de nodos. Una generalización de este criterio es el de “coste mínimo”.

El algoritmo de selección de ruta mínima trata de identificar las rutas más cortas a nivel de número de conmutadores, aunque no tiene en cuenta el grado de congestión de estos.

A diferencia del algoritmo genético, el algoritmo de ruta mínima no parte de una población inicial e itera para la consecución de individuos mejores propios de la selección natural, sino que es un único individuo cuyas rutas activas entre pares para la formación del cromosoma se eligen por el método anteriormente expuesto, es decir, la ruta más corta.

### **9.6.1 Pseudocódigo del algoritmo de selección de ruta mínima**

#### **9.6.1.1 Simbología**

En la Figura 17 se observa la simbología que se va a utilizar para la realización del pseudocódigo del algoritmo de selección de ruta mínima:

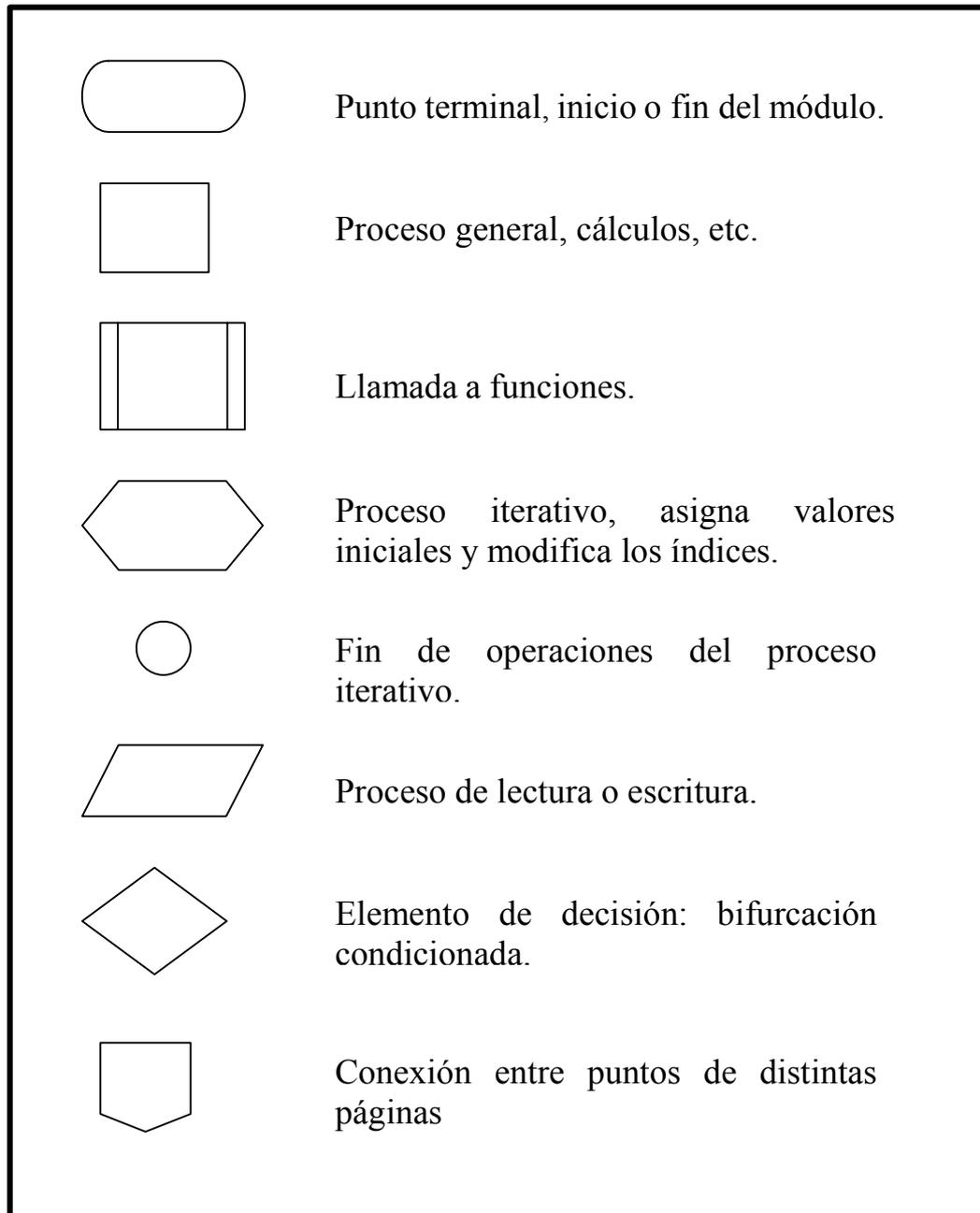


Figura 17. Simbología

### 9.6.1.2 Función principal

En la Figura 18 se observa el diagrama de flujo de la función principal ‘main’.

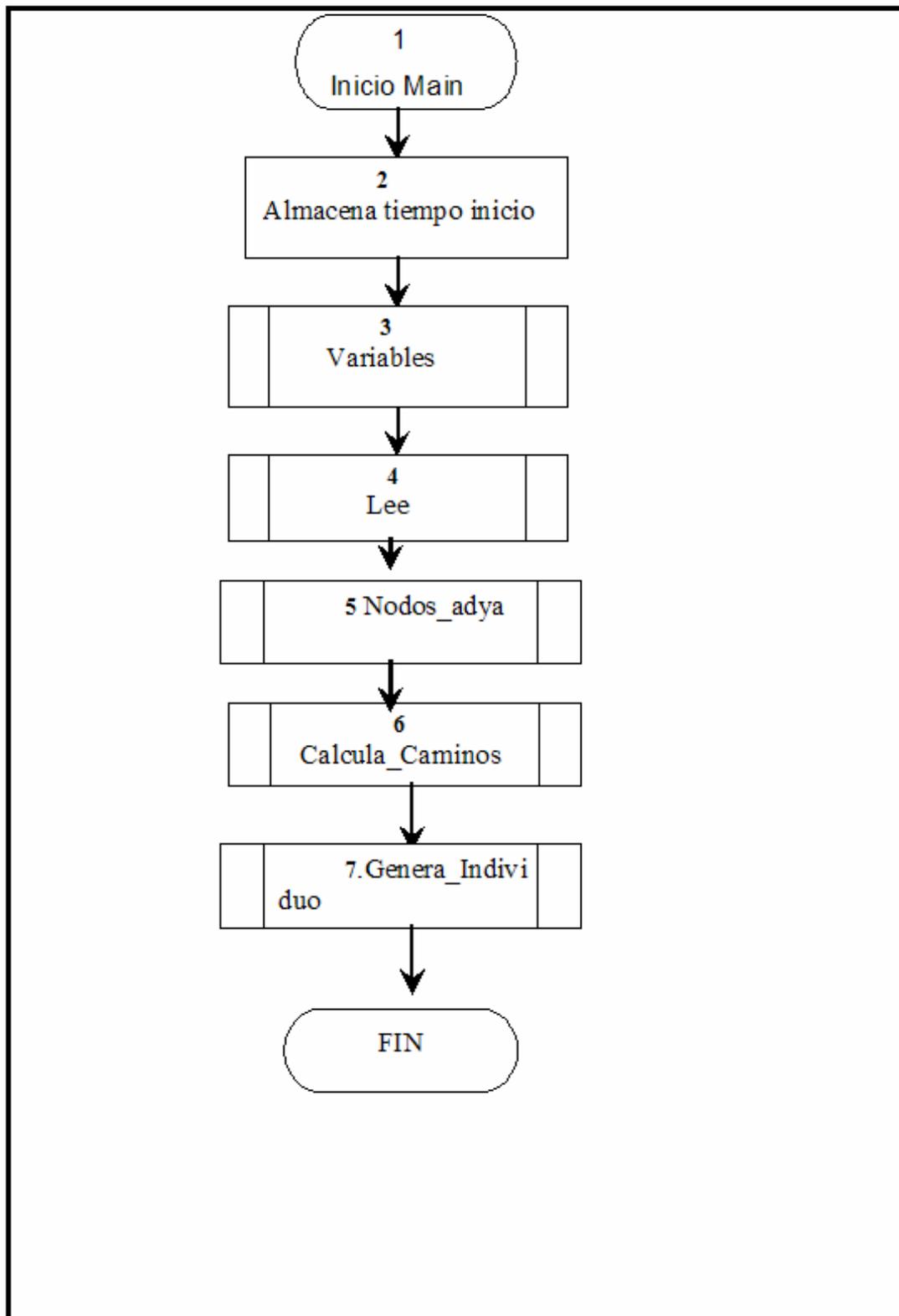


Figura 18. Diagrama de flujo de la función principal 'main'.

### 9.6.1.3 Otras funciones

#### Función LEE

Como su nombre indica, esta función se encarga de leer los datos del fichero de entrada “GeneraFichero.txt” y de guardar los datos en las variables pertinentes.

- **Prototipo.** *ARCOS \* lee (char m[], DEMANDA \*\*p).*
- **Parámetros de entrada.** Se le pasa la variable m, de tipo string, y un puntero a puntero para crear una lista con los valores de las demandas de cada par y para cada instante.
- **Parámetros de salida.** *ARCOS \**. Devuelve la dirección de memoria del primer elemento de una lista que contiene los arcos que constituyen, en parte, la topología de la red.

#### Función VARIABLES

Esta función se encarga de dar valor a todos los parámetros con los que trabaja el algoritmo. Los parámetros con los que trabaja esta función están definidos algunos como variables globales.

#### Función NODOS\_ADYACENTES

Esta función se encarga de calcular para cada uno de los nodos de la red sus correspondientes adyacentes.

- **Prototipo.** *void nodos\_adyacentes(ADYA \*\*pa, ARCOS \*parco);*
- **Parámetros de entrada.** Se le pasa como primer argumento la dirección de memoria de un puntero que almacenará una lista con todos los adyacentes correspondientes a cada uno de los nodos; como segundo argumento, y último, se le pasa una lista con todos los arcos de la red, que nos viene dado por el fichero de entrada.
- **Parámetros de salida.** No devuelve nada.

## Función CALCULA\_CAMINOS

Función que obtiene todos los caminos posibles entre todos los pares origen-destino que componen la red.

- **Prototipo.** *void calcula\_caminos(CAMINOS \*\*ca, ADYA \*p, int nodoo, int camino[], int indice)*
- **Parámetros de entrada.** Como primer argumento se le pasa la dirección del primer elemento de una lista que contiene todos los caminos entre todos los pares; como segundo argumento le pasa una lista con la información de todos los adyacentes entre nodos; como tercer argumento se le pasa el nodo padre denominado hondo; como cuarto argumento se le pasa una cadena llamada camino para guardar el camino.
- **Parámetros de salida.** No devuelve nada.

## Función GENERAR\_INDIVIDUO

Función que se encarga de generar la población inicial de un individuo.

- **Prototipo.** *MATINDIV \* genera\_individuo (int total, DEMANDA \*d, CAMINOS \*c, ARCOS \*a, CAMINOORDENADO \*\*cor, ADYA \*adya, RUTAACTIVA \*\*pruta).*
- **Parámetros de entrada.** Se le pasa como primer argumento el número total de caminos hallados entre todos los pares; como segundo argumento se le pasa una lista con todos los caminos entre pares; como tercer argumento se le pasa una lista con los arcos que definen, en parte, la topología de la red; como cuarto argumento se le pasa una lista con los caminos ordenados por par; como quinto argumento se le pasa una lista con los adyacentes a cada nodo de la red; como quinto argumento se le pasa la dirección de un puntero que contendrá de todos los caminos, sólo las rutas activas.
- **Parámetros de salida.** Devuelve una lista con el cromosoma codificado del individuo y sus correspondientes pérdidas.

Básicamente, la diferencia entre el algoritmo genético y el de selección de ruta mínima, se encuentra en la función de generar\_individuo. A continuación, en la Figura 19, se detalla explícitamente la función:

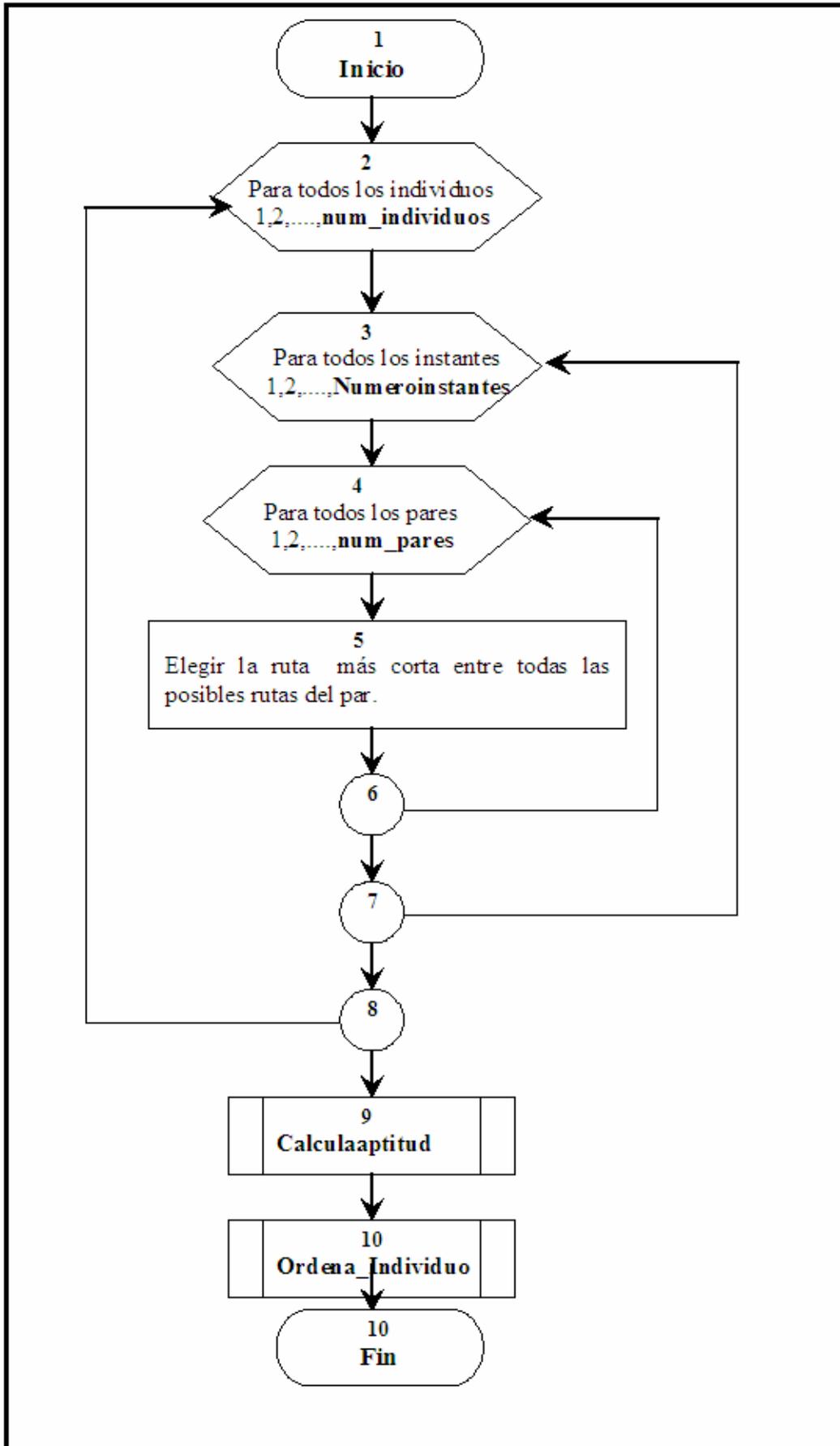


Figura 19. Diagrama de flujos de la función 'generar\_individuo'.

## 10 DESCRIPCIÓN DEL SOFTWARE DEL ALGORITMO DE OPTIMIZACIÓN DE ENCAMINAMIENTO ATM

El estudio del modelo matemático planteado para el encaminamiento de células ATM se realiza gracias a la implementación de un programa informático capaz de resolverlo. En este caso la programación ha sido llevada a cabo utilizando el lenguaje de programación C.

### 10.1 DATOS DE ENTRADA DEL ALGORITMO GENÉTICO

Estos datos son, principalmente, el número de nodos y arcos de la red y la demanda de los pares. También se incluye una descripción de la red en la que se asigna un número a cada uno de los arcos y los nodos. El formato definido deberá indicar:

1. El número de nodos de la red a considerar.
2. El número de arcos de la misma
3. El número de intervalos de conexión.
4. Para cada uno de los arcos:
  - Número de nodo origen
  - Número de nodo destino
5. Para cada uno de los pares:
  - La demanda en cada uno de los instantes de conexión considerados

Para cada uno de los ficheros de entrada, el programa utilizado usa como lenguaje de programación C.

El número de intervalos de conexión- $t$ - utilizado para toda la batería de problemas es cuatro, y el número de instantes de conexión- $\tau$ - es diez. En éste proyecto, se trabajará con dos niveles, o escalas jerárquicas. El nivel de conexión- $t$ - y el de célula- $\tau$ .

Trabajar con el nivel de célula resulta indispensable si se tiene en cuenta que el análisis de pérdidas en los conmutadores debe hacerse célula a célula.

La generación aleatoria del número de nodos para la batería de problemas, tiene como restricción el ser mayor o igual a diez, y menor o igual a cuarenta.

La generación aleatoria del número de conexiones establecidas durante cada intervalo oscilará entre un mínimo de diez conexiones y como máximo el combinatorio definido por:

$$\binom{\text{NumeroNodos}}{2} = \text{NumeroNodos!} / 2! * (\text{NumeroNodos} - 2)! \quad \text{Ecuación 26}$$

## 10.2 PSEUDOCÓDIGO DEL ALGORITMO GENÉTICO

A grandes rasgos, los pasos seguidos por el algoritmo código que implementa el algoritmo genético se reducen a las incluidas en la tabla que se presenta a continuación:

1. Generar aleatoriamente una población inicial compuesta por N individuos. Esta población dependerá en gran medida de la topología de la red.
2. Evaluar la aptitud de los individuos y proceder a su ordenación en función de la misma.
3. Determinar aleatoriamente un número p, entre 0 y 1. En función de este número se ha de decidir el siguiente paso del algoritmo:
  - Si  $p \geq p_c$  entonces se origina un cruce entre dos individuos para generar un nuevo elemento de la población.
  - Si  $p \leq p_c$  entonces lo que se origina es una mutación de los genes de un individuo, dando lugar, igualmente, a un nuevo elemento de la población.
4. Calcular la aptitud del nuevo individuo generado.
5. Calcular la probabilidad de reemplazo de todos los individuos que componían la población anterior.
6. Se procede al reemplazo de algún individuo de la población por el nuevo individuo generado, según las reglas de reemplazo descritas anteriormente.
7. Volver a 3.

Todo este proceso se realiza hasta que se llega a un número máximo de generaciones del algoritmo prefijado de antemano, o bien, los individuos llegan a un grado de similitud elevado.

### 10.3 VARIABLES GLOBALES DEL PROGRAMA

El número de variables globales del programa se ha intentado de reducir al máximo. A continuación, muestro una tabla con las distintas variables globales del programa y su correspondiente explicación. Véase la Tabla 17.

**Tabla 17. Tabla representativa de las variables globales del sistema**

<b>Descripción / Variables globales del sistema</b>	<b>Descripción general</b>	<b>Tipo de dato</b>
<b>B</b>	Tamaño del búfer	Int
<b>Pares</b>	Número de pares de la red. (Viene dado por el fichero de entrada).	Int
<b>Num_arcos</b>	Número de enlaces de la red. (Viene dado por el fichero de entrada).	Int
<b>Num_nodos</b>	Número de nodos de la red. (Viene dado por el fichero de entrada).	Int
<b>Num_individuos</b>	Número de individuos de la población.	Int
<b>Numeroinstantes</b>	Número de intervalos de conexión	Int
<b>Lim_incesto</b>	Determina el límite de incesto entre dos individuos antes de aplicar el operador cruce.	Int
<b>Num_ciclos</b>	Determina el número de iteraciones	Int

<b>Descripción / Variables globales del sistema</b>	<b>Descripción general</b>	<b>Tipo de dato</b>
	máximas del algoritmo genético	
<b>GoS</b>	Determina el grado de servicio	Float

#### 10.4 CONSTANTES SIMBÓLICAS DEL PROGRAMA

**Tabla 18. Tabla representativa de las constantes simbólicas del programa**

<b>Descripción/ Constantes simbólicas del sistema</b>	<b>Descripción general</b>
<b>TAM</b>	Constante simbólica para cadenas
<b>TAMADYA</b>	Constante simbólica que define el número máximo de adyacentes a un nodo
<b>RES_SALTOS</b>	Constante simbólica que define el número máximo de procesos de conmutación permitidos entre par origen y par destino
<b>LIM_INCES</b>	Constante simbólica que controla el grado de semejanza entre individuos
<b>CONEX</b>	Constante simbólica que define el número de conexiones abiertas dadas en el fichero de entrada pasadas al algoritmo

## 10.5 ESTRUCTURAS EMPLEADAS EN EL PROGRAMA

Las estructuras más importantes utilizadas para la realización del programa, además de variables y arrays, son las denominadas estructuras dinámicas. De entre ellas, la elegida han sido las listas.

```
typedef struct caminos
{
    int comorigen;
    int cam[TAMA];
    struct caminos *sig;
}CAMINOS;
```

### *Descripción*

Lista que almacenará todos los caminos posibles entre cada par de la red.

```
typedef struct caminoordenado
{
    int origenordenado;
    int destinoordenado;
    int marcas;
    int caminoorden[TAMA];
    struct caminoordenado *sig;
}CAMINOORDENADO;
```

### *Descripción*

Lista que almacenará todos los posibles caminos de la red pero ordenados por cada par.

```
typedef struct adya {
    int n_nodo;
    int cuantos;
    int saleenlace[TAMADYA];
    int entraenlace[TAMADYA];
    struct adya *sig;
}ADYA;
```

### ***Descripción***

Lista que almacenará para todos los nodos, sus respectivos adyacentes de la red, ya sea el arco de salida o de entrada.

```
struct pares
{
    int origen;
    int destino;
    int num_par;
    int n_instante;
    int instantegrande;
};
```

### ***Descripción***

Lista que almacenará todos los posibles pares de la red.

```
typedef struct matindiv
{
    int ind;
    float fitness;
    float fitnessverdad;
    int *infoindiv;
    struct matindiv *sig;
}MATINDIV;
```

### ***Descripción***

Lista que almacenará el cromosoma de cada individuo, y su correspondiente valor de pérdidas o fitness.

```
typedef struct demanda
{
    float demanda_par;
    struct pares p;
    struct demanda *sig;
}DEMANDA;
```

### ***Descripción***

Lista que almacenará para todos los pares su correspondiente valor de demanda para cada instante de conexión  $\tau$  y cada intervalo de conexión  $t$ . Los datos son recogidos del fichero de entrada.

```
typedef struct rutaactiva
{
    int paro;
    int pard;
    int ruta[TAMA];
    int instante;
    int marca;
    struct rutaactiva *sig;
}RUTAACTIVA;
```

### ***Descripción***

Lista que almacenará todas las rutas activas para cada intervalo de conexión  $t$  y para cada individuo que compone la población.

```
typedef struct conexasarcos
{
    int cualarco;
    int ins;
    int co;
    int cd;
    int tau;
    struct conexasarcos *sig;
}CONEXARCOS;
```

### ***Descripción***

Lista que almacenará todas las conexiones abiertas que atraviesan cada arco para cada instante de conexión  $\tau$ , para cada intervalo de conexión  $t$ , y para un individuo concreto de la población.

## 10.6 DIAGRAMA DE FLUJOS DEL PROGRAMA

### 10.6.1 Simbología

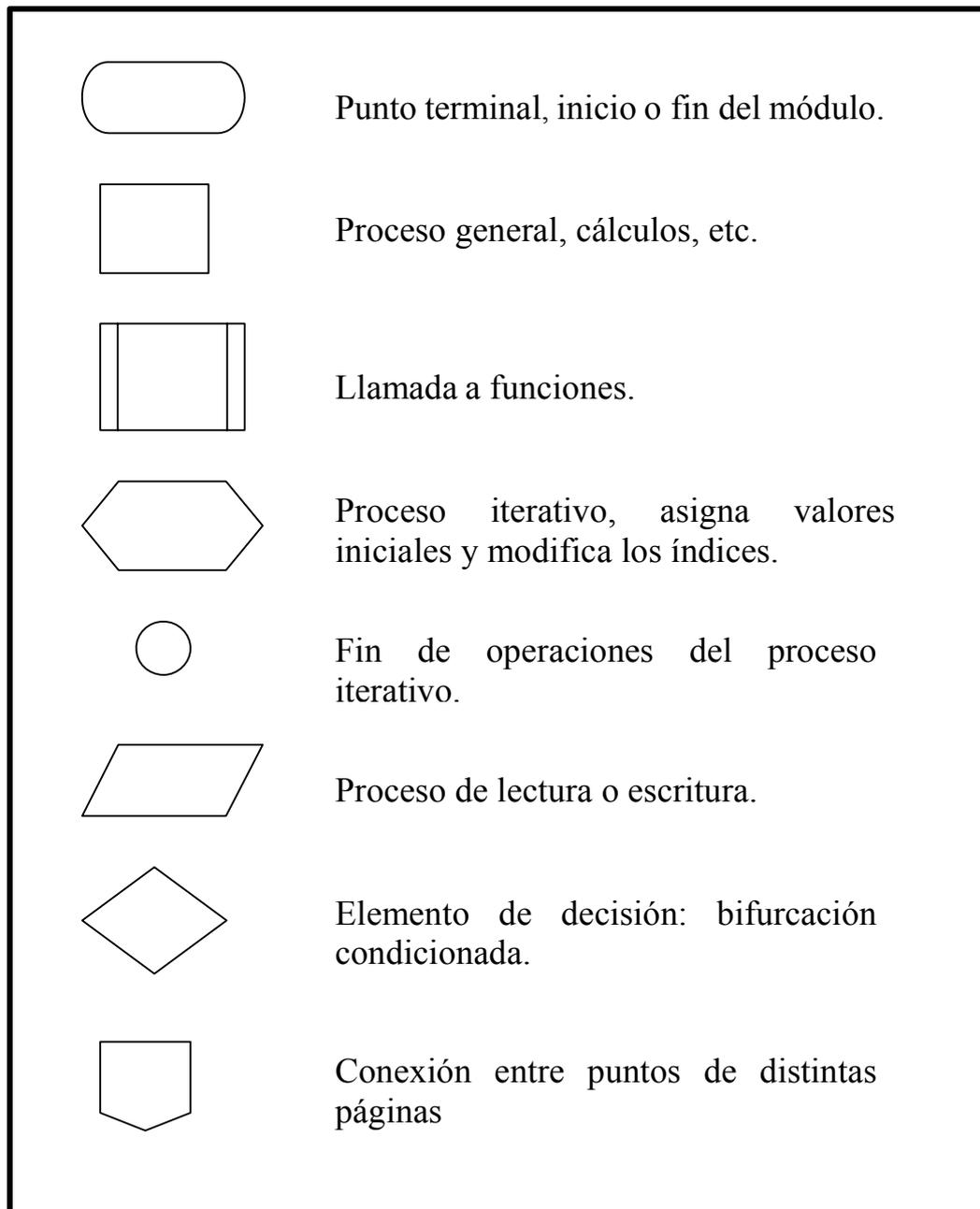


Figura 20. Representación de la simbología.

## 10.7 FUNCIÓN PRINCIPAL MAIN

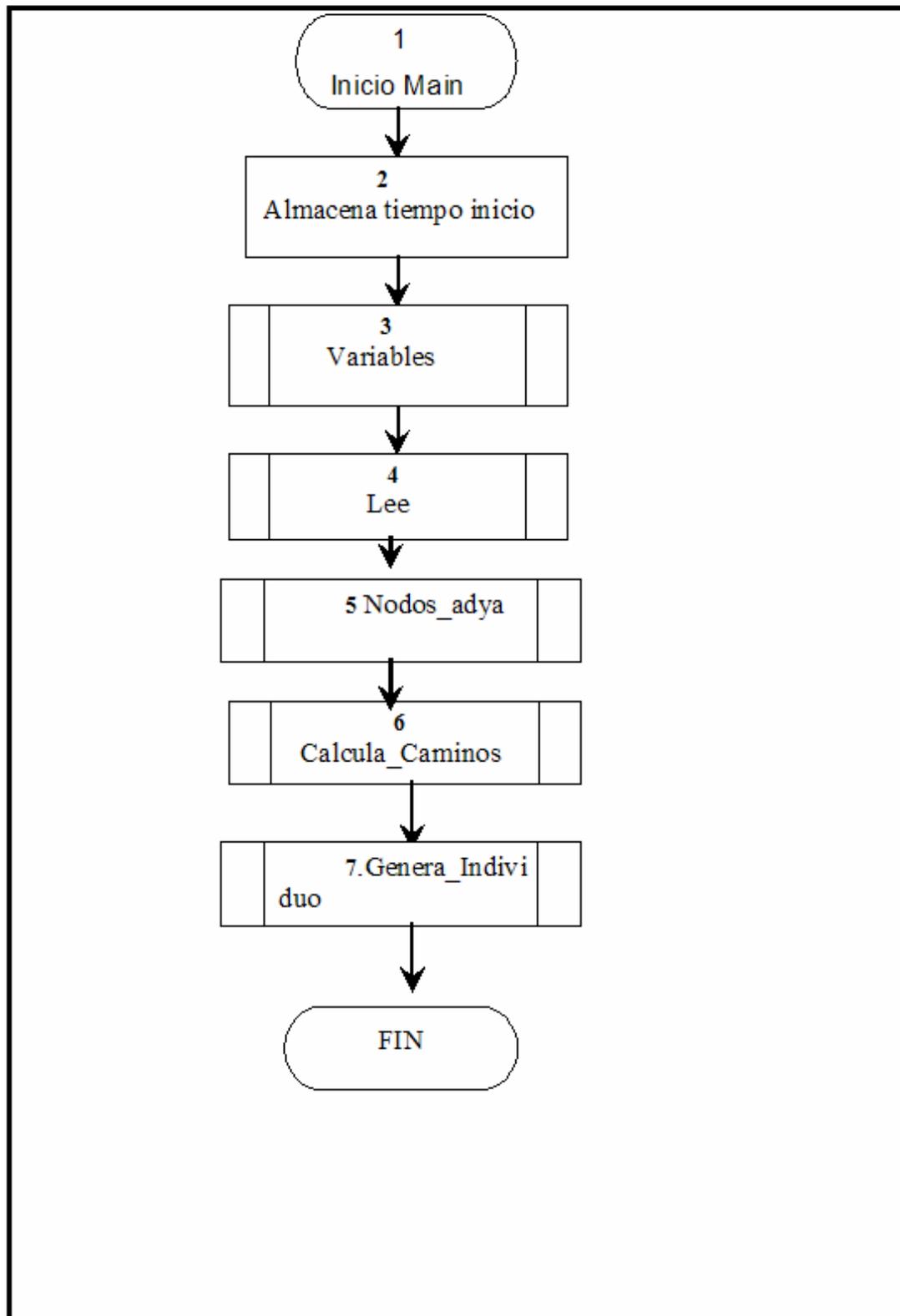


Figura 21. Representación de la función principal 'main'.

## 10.7.1 Descripción de las funciones

### 10.7.1.1 Función lee

Tabla 19. Tabla representativa de la función lee

Descripción /Función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Lee	ARCOS* lee (char m[], DEMANDA **p).	1)char *m 2)DEMANDA **p	ARCOS *	Como su nombre indica, esta función se encarga de leer los datos del fichero de entrada "GeneraFichero.txt" y de guardar los datos en las variables pertinentes.

#### *Parámetros de entrada*

Se le pasa la variable m, de tipo string, y un puntero a puntero para crear una lista con los valores de las demandas de cada par y para cada instante de conexión  $\tau$ .

#### *Parámetros de salida*

Devuelve la dirección de memoria del primer elemento de una lista que contiene los arcos que constituyen, en parte, la topología de la red.

### 10.7.1.2 Función nodos\_adyacentes

Tabla 20. Tabla representativa de la función nodos\_adyacentes.

Descripción /Función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Nodos_adyacente	void	1)ADYA	void	Esta

Descripción /Función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
s	nodos_adyacentes(ADYA **pa,ARCOS *parco);	**pa 2) ARCOS *parco		función se encarga de calcular para cada uno de los nodos de la red sus nodos adyacentes

#### *Parámetros de entrada*

Se le pasa como primer argumento la dirección de un puntero que almacenará una lista con todos los nodos adyacentes a cada nodo; como segundo argumento se le pasa una lista con todos los arcos de la red que nos vienen descritos en el fichero de entrada, necesarios para calcular los adyacentes.

#### *Parámetros de salida*

No devuelve nada.

### 10.7.1.3 Función calcula\_caminos

Tabla 21. Tabla representativa de la función calcula\_caminos.

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Calcula_caminos	void calcula_caminos(CAMINOS **ca, ADYA *p, int nodoo, int camino[], int indice)	1)CAMINOS ** ca 2) ADYA *p	void	Función que obtiene todos los caminos posibles entre todos

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
		3)int noodo 4)int camino[] 5)int indice		los pares origen-destino que componen la red.

#### *Parámetros de entrada*

Como primer argumento se le pasa la dirección del primer elemento de una lista que contiene todos los caminos entre todos los pares; como segundo argumento le pasa una lista con la información de todos los adyacentes entre nodos; como tercer argumento se le pasa el nodo padre denominado hondo; como cuarto argumento se le pasa una cadena llamada camino para guardar el camino.

#### *Parámetros de salida*

No tiene.

### 10.7.1.4 Función generar\_individuo

**Tabla 22. Tabla representativa de la función generar\_individuo**

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Generar_Individuo	MATINDIV * genera_individuo (int total, DEMANDA *d, CAMINOS *c, ARCOS *a, CAMINOORDEN	1) int total 2)DEMANDA *d 3)CAMINOS *c 4)ARCOS *a	MATINDIV *	Función que se encarga de generar la población

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
	ADO **cor, ADYA *adya, RUTAACTIVA **pruta).	5)CAMINOORDEN ADO **cor 6)ADYA *adya 7)RUTAACTIVA **pruta		n inicial de individuos

#### *Parámetros de entrada*

Se le pasa como primer argumento el número total de caminos hallados entre todos los pares; como segundo argumento se le pasa una lista con todos los caminos entre pares; como tercer argumento se le pasa una lista con los arcos que definen, en parte, la topología de la red; como cuarto argumento se le pasa una lista con los caminos ordenados por par; como quinto argumento se le pasa una lista con los adyacentes a cada nodo de la red; como sexto argumento se le pasa la dirección de un puntero que contendrá de todos los caminos posible entre par origen y destino, sólo las rutas activas.

#### *Parámetros de salida*

Devuelve una lista con el cromosoma codificado del individuo y sus correspondientes pérdidas.

### 10.7.1.5 Función cruce

Tabla 23. Tabla representativa de la función cruce.

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Cruce	Int cruce(MATINDIV **p, DEMANDA *d, RUTAACTIVA	1)MATINDIV **p 2)DEMANDA *d	int	Función que se encarga de cruzar a

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
	**r,CAMINOORDENADO **cor,CAMINOSPAR *ca,int total,int *muchos_incestos,ARCOS *a,ADYA *adya);	3)RUTA ACTIVA **r 4)CAMINO ORDENADO **cor 5)CAMINOSPAR *ca 6)int total 7)int *muchos_incestos 8)ARCOS *a 9)ADYA *adya		dos individuos obteniendo uno nuevo con características heredadas de sus progenitores. Esta función incluye prevención de incesto y control de semejanza entre los individuos. Esta función se detalla posteriormente.

### *Parámetros de entrada*

Se le pasa como primer argumento la dirección de memoria de un puntero a la lista que contiene el valor del fitness (pérdidas) de cada individuo y sus correspondientes cromosomas; como segundo argumento se le pasa una lista con los valores de las demandas entre pares; como tercer argumento se le pasa la dirección de memoria de un puntero que contiene una lista con todas las rutas activas de cada individuo; como cuarto argumento se le pasa la dirección de memoria de un puntero que contiene una lista con todos los caminos ordenados por par; como quinto argumento se le pasa el número total de caminos posibles entre todos los pares de la red; como sexto argumento se le pasa la dirección de una variable para controlar la semejanza entre individuos; como séptimo argumento se le pasa una lista con todos los arcos existentes en la red; como octavo argumento se le pasa una lista con los nodos adyacentes a cada nodo de la red.

*Parámetros de salida*

Devuelve un entero que vale cero si todo el proceso de cruce ha ido bien, y uno si ocurre lo contrario.

**10.7.1.6 Función mutación**

**Tabla 24. Tabla representativa de la función mutación**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
Mutación	int mutacion(MATINDIV **p, DEMANDA *d, RUTAACTIVA **r, CAMINOORDENADO **cor, CAMINOSPAR *ca, int total, ARCOS *a, ADYA * adya);	1)MATINDIV **p 2)DEMANDA *d 3)RUTAACTIVA **r 4)CAMINORDENADO **cor 5)CAMINOSPAR *ca 6)int total 7)ARCOS *a 8)ADYA *adya	int	Función que se encarga de mutar un individuo de forma que se genere otro nuevo que ocupará su lugar en la población.

*Parámetros de entrada*

Se le pasa como primer argumento la dirección de memoria de un puntero a la lista que contiene el valor del fitness (pérdidas) de cada individuo y sus correspondientes cromosomas; como segundo argumento se le pasa una lista con los valores de las demandas entre pares; como tercer argumento se le pasa la dirección de memoria de un puntero que contiene una lista con todas las rutas activas de cada individuo; como cuarto argumento se le pasa la dirección de memoria de un puntero que contiene una lista con todos los caminos ordenados por par; como quinto argumento se le pasa el número total de caminos posibles entre todos los pares de la red; como sexto

argumento se le pasa una lista con todos los arcos existentes en la red; como séptimo argumento se le pasa una lista con los nodos adyacentes a cada nodo de la red.

*Parámetros de salida*

Devuelve un entero que vale cero si todo el proceso de mutación ha ido bien, y uno si ocurre lo contrario.

**10.7.1.7 Función restricción\_saltos**

**Tabla 25. Tabla representativa de la función restricción\_saltos**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
Restricción_saltos	int restriccion_saltos(CAMINOS **p);	1)CAMINOS ** p	int	Función que limita el número máximo de procesos de conmutación entre par origen y destino

*Parámetros de entrada*

Se le pasa una lista con todos los posibles caminos entre todos los pares de la red.

*Parámetros de salida*

Devuelve un entero cuyo valor es el número de caminos excluidos.

### 10.7.1.8 Función controlsemejanza

Tabla 26. Tabla representativa de la función controlsemejanza

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Controlsemejanza	int controlsemejanza(MATINDIV *p,int total);	1)MATINDIV * p 2) int total	int	Función que controla en cada generación que la semejanza de los individuos

#### *Parámetros de entrada*

Se le pasa como primer argumento una lista con los distintos cromosomas de los individuos de la población y sus correspondientes pérdidas; como segundo argumento se le pasa el número total de caminos entre todos los pares de la red.

#### *Parámetros de salida*

Devuelve un entero cuyo valor es cero si todavía los individuos no son muy semejantes, y uno en caso contrario con la inmediata finalización de los individuos.

### 10.7.1.9 Función determina\_límites

Tabla 27. Tabla representativa de la función controlsemejanza

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Determina_límite	int	void	int	Función que

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
s	determina_limites() ;			determina el límite de semejanza entre individuos. Es utilizada por la función controlsemejanza

*Parámetros de entrada*

No tiene.

*Parámetros de salida*

Devuelve un entero cuyo valor es el setenta y cinco por ciento de todos los pares posibles de la red.

**10.7.1.10 Funciones de representación de resultados**

Todas las funciones mostradas en la Tabla 28, representan los distintos resultados válidos para la realización de estudios en distintos ficheros de salida.

**Tabla 28. Tabla representativa de las funciones de representación de resultados**

Descripción / Funciones	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Escribe1	void escribe1(MATINDIV *p,int total);	1)MATINDIV *p 2)int total	void	Función que escribe el valor de pérdidas de cada

Descripción / Funciones	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
				individuo de la población
Escribe2	void escribe2(CAMINOSPARR *hp);	1)CAMINOSPARR *hp	void	Función que escribe todos los caminos posibles de cada par existente en la red
Escribe3	void escribe3(RUTAACTIVA *pr);	1)RUTAACTIVA *pr	void	Función que escribe todas las rutas activas de cada individuo
Escribe4	void escribe4(CONEXARCOS *p);	1)CONEXARCOS *p	void	Función que escribe las conexiones abiertas que atraviesan cada arco

## Escribe1

### *Parámetros de entrada*

Se le pasa como primer argumento una lista con los cromosomas de cada individuo y el correspondiente valor de las pérdidas.

*Parámetros de salida*

No devuelve nada.

### **Escribe2**

*Parámetros de entrada*

Se le pasa una lista con todos los posibles caminos ordenados para cada par.

*Parámetros de salida*

No devuelve nada.

### **Escribe3**

*Parámetros de entrada*

Se le pasa una lista con todas las rutas activas de cada individuo.

*Parámetros de salida*

No devuelve nada.

### **Escribe4**

*Parámetros de entrada*

Se le pasa como argumento una lista cuyo contenido son las conexiones abiertas que atraviesan cada arco en cada intervalo de conexión-  $t$ - y para cada instante de conexión-  $\tau$ -.

*Parámetros de salida*

No devuelve nada.

## **10.8 OTRAS FUNCIONES**

### **10.8.1 Función calcula\_caminos**

Se encarga de calcular todos los caminos posibles entre un origen y un destino teniendo en cuenta la limitación impuesta en el número de saltos (presente en la variable res\_salto).

Puede verse en la Figura 22:

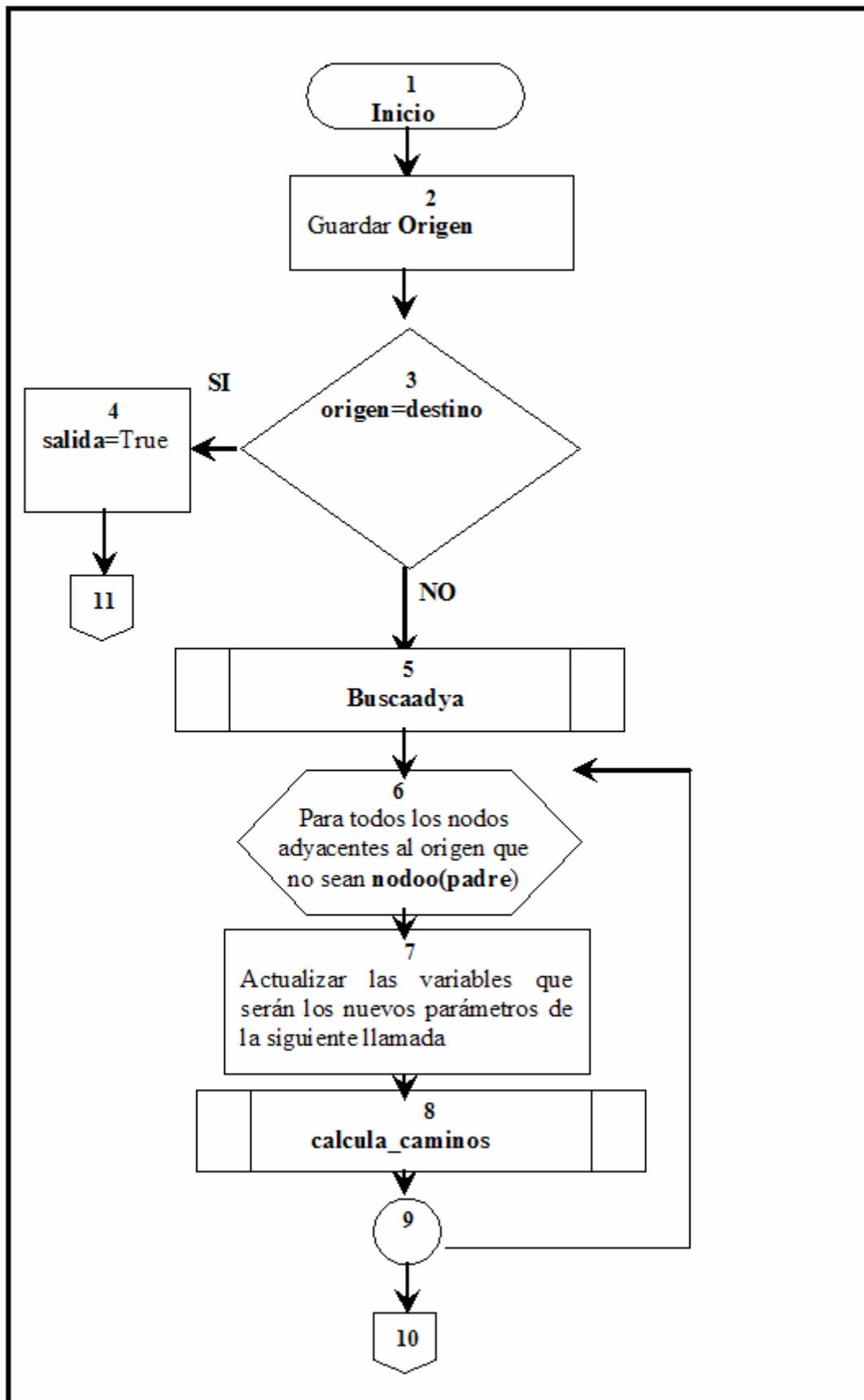


Figura 22. Representación del diagrama de flujo de la función 'calcula\_caminos'

A su vez, esta función hace referencia a buscaadyaa.

## Función BUSCAADYA

Tabla 29. Tabla Representativa de la función buscadadya

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
buscaadya	ADYA* buscaadya(ADYA *p,int origen);	1)ADYA *p 2)int origen	ADYA*	Función que determina todos los nodos adyacentes a uno dado pasado como argumento

### *Parámetros de entrada*

Se le pasa como primer argumento una lista con todos los adyacentes a todos los nodos de la red; como segundo argumento se le pasa el nodo del cual se quieren saber sus adyacentes.

### *Parámetros de salida*

Devuelve el nodo de la lista buscado.

## 10.8.2 Función generar\_individuo

En la Figura 23 se puede ver la función generar\_individuo:

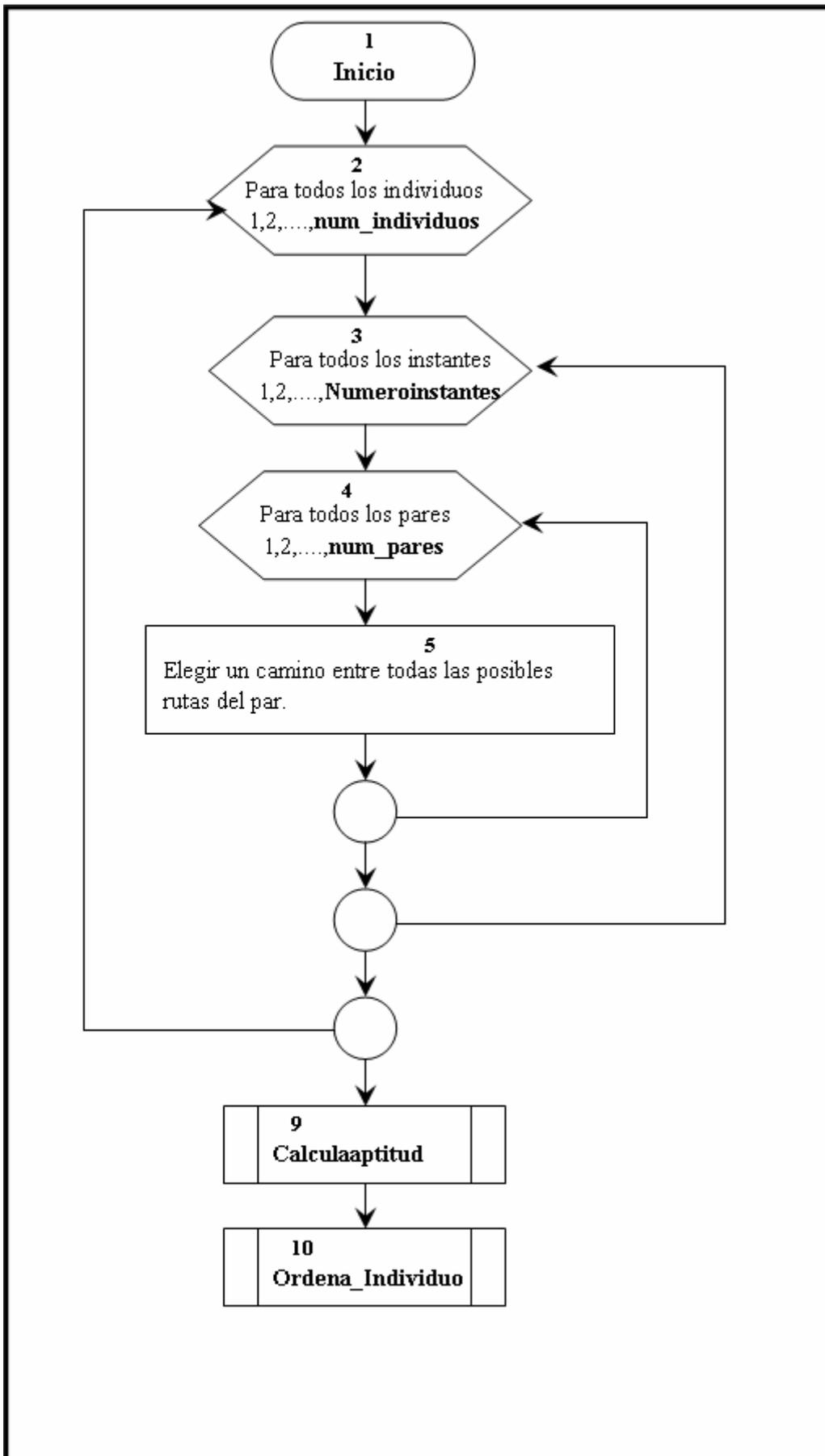


Figura 23. Representación del diagrama de flujo de la función 'generar\_individuo'.

A su vez, dentro de la función generar\_individuo, se hace referencia a diversas funciones.

### Función CALCULAAPTITUD

Tabla 30. Tabla representativa de la función calculaaptitud

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
calculaaptitud	float calculaaptitud(ARCOS *pa, DEMANDA *d, RUTAACTIVA **pr, ADYA *adya, int indre, CAMINOORDEN ADO **cor, float * pf);	1)ARCOS *pa 2)DEMANDA * d 3)RUTAACTIVA **pr 4)ADYA *adya 5)int indre 6)CAMINOORDEN ADO **cor 7)float *pf	float	Función que determina el fitness de cada individuo (pérdidas). Devuelve la aptitud.

#### Parámetros de entrada

Se le pasa como primer argumento una lista con todos los arcos de la red, que vienen dados por el fichero de entrada; como segundo argumento se le pasa una lista con las demandas entre todos los pares para cada instante de conexión  $\tau$ ; como tercer argumento se le pasa la dirección de memoria de un puntero para guardar todas las rutas activas de cada individuo; como cuarto argumento se le pasa un lista con los adyacentes de cada nodo de la red; como quinto argumento se le pasa una variable índice que indica el orden del individuo pasado como argumento; como sexto argumento se le pasa la dirección de memoria de un puntero que contendrá una lista con todos los caminos ordenados por par existente en la red; como séptimo, y último argumento, se le pasará la dirección de una variable para almacenar las pérdidas ficticias de cada individuo.

### *Parámetros de salida*

Devuelve una variable de tipo flota cuyo contenido es la aptitud del individuo.

### **Función ordena\_individuo**

**Tabla 31. Tabla representativa de la función ordena\_individuo**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
Ordena_individuo	void ordena_individuo1(MATINDIV *p,float fit[]);	1)MATINDIV *p 2)float fit[]	void	Función que ordena de mayor a menor pérdidas los correspondientes individuos de la población

### *Parámetros de entrada*

Se le pasa como primer argumento una lista con el cromosoma de cada individuo y su correspondiente aptitud; como segundo argumento se le pasa una tabla donde se almacenará en orden descendente el valor de las aptitudes de los individuos.

### *Parámetros de salida*

No devuelve nada.

### **10.8.3 Función mutación**

En la Figura 24, se observa el diagrama de flujo de la función mutación:

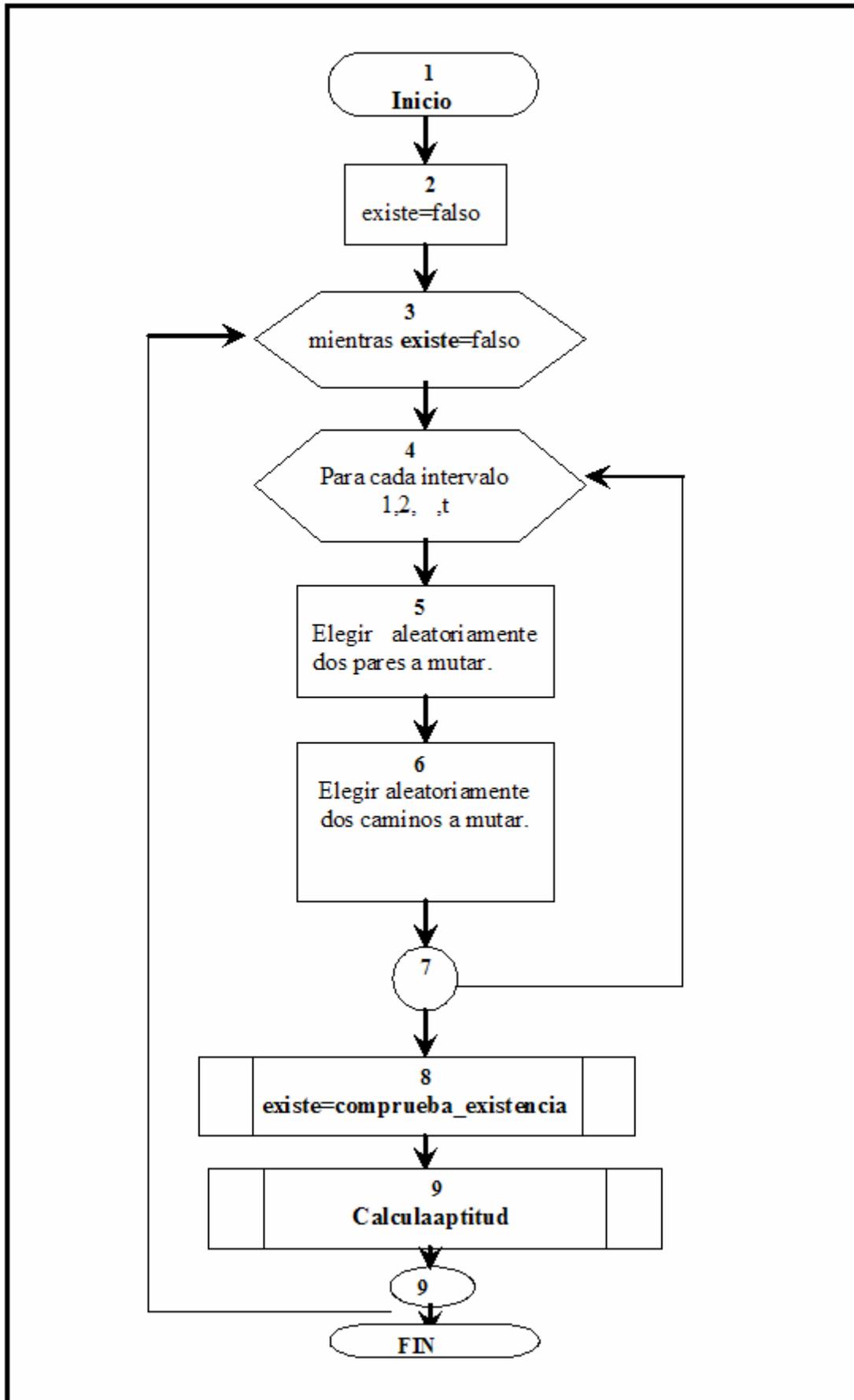


Figura 24. Representación del diagrama de flujo de la función 'mutación'.

A su vez, dentro de esta función, se hace referencia a `comprueba_existencia`.

### **Función `COMPRUEBA_EXISTENCIA`**

**Tabla 32. Tabla representativa de la función `comprueba_existencia`**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
Comprueba_existencia	<code>int comprueba_existencia(MATINDIV *p,int individuo[],int total_caminos);</code>	1) <code>MATINDIV *p</code> 2) <code>int individuo[]</code> 3) <code>int total_caminos</code>	<code>int</code>	Función que se encarga de comprobar que el nuevo individuo generado que se le pasa como argumento no existe

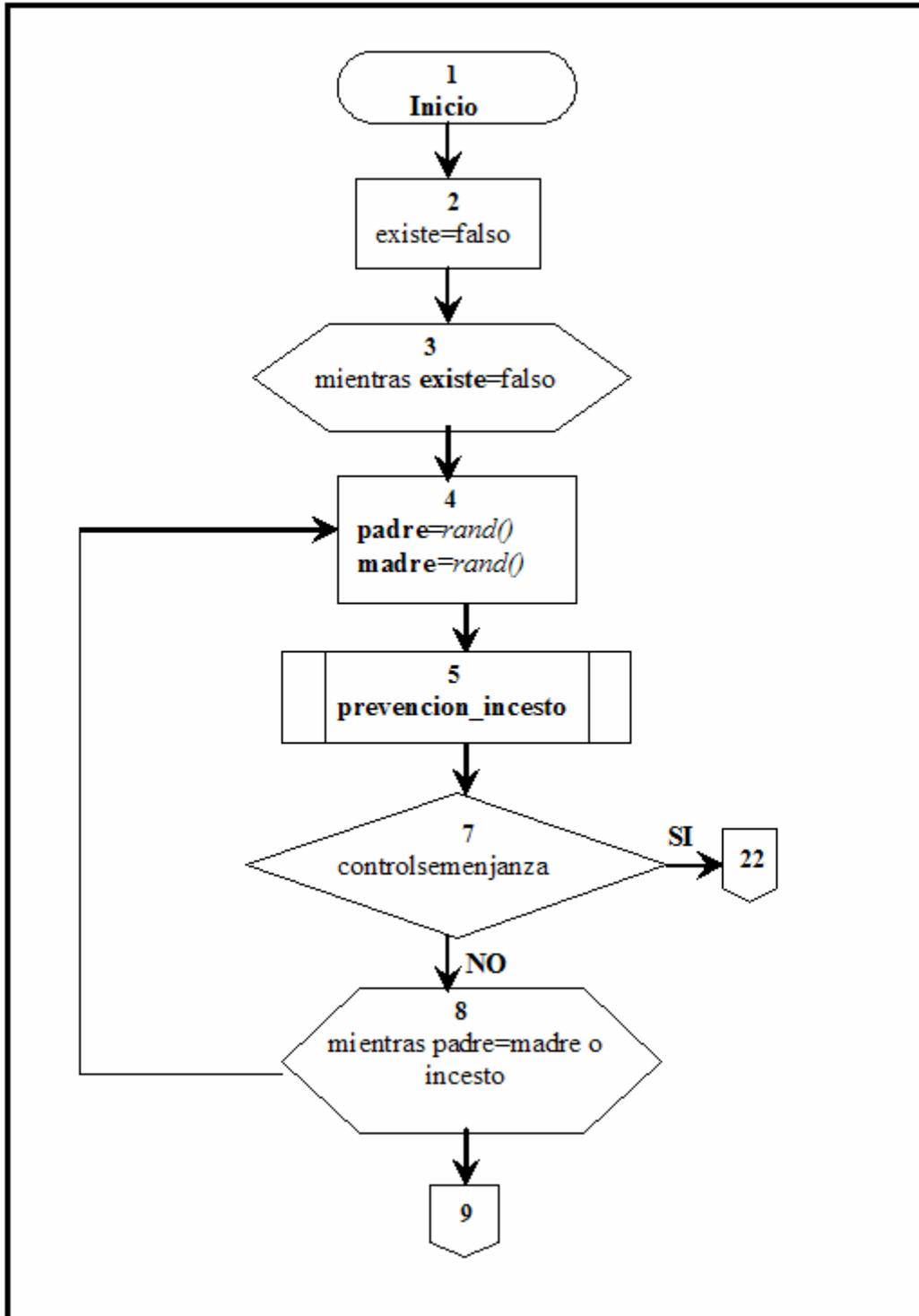
#### *Parámetros de entrada*

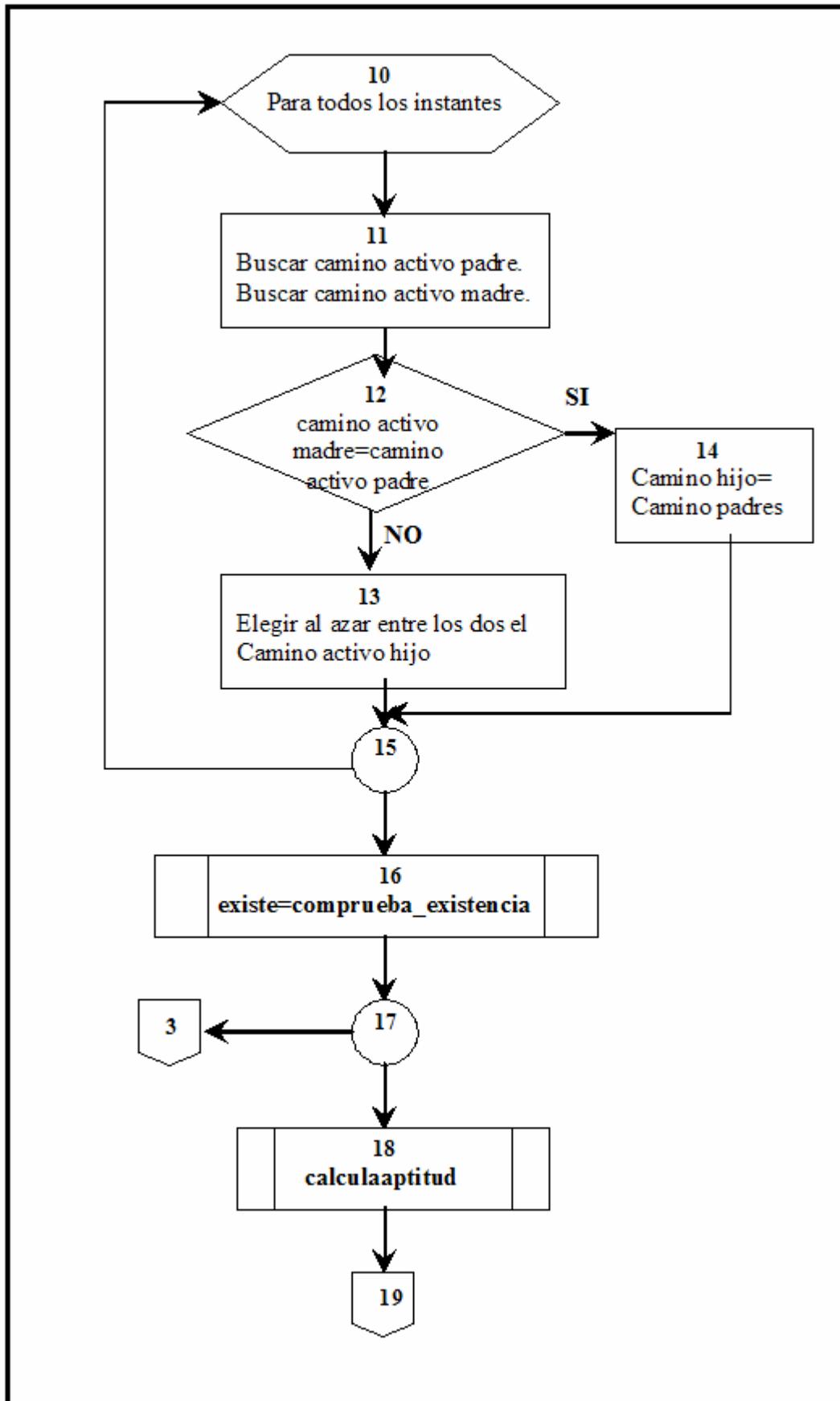
Se le pasa como primer argumento una lista con todos los cromosomas de los individuos y sus respectivas pérdidas; como segundo argumento se le pasa una cadena que contiene el cromosoma del individuo a probar su existencia; como tercer, y último argumento, se le pasa el número total de caminos de la red.

#### *Parámetros de salida*

Devuelve un entero cuyo valor es cero si la comprobación da negativo, es decir, el individuo no existiese. Si da positivo, se volvería a llamar a la función para la elección de otro individuo a mutar.

### 10.8.4 Función cruce





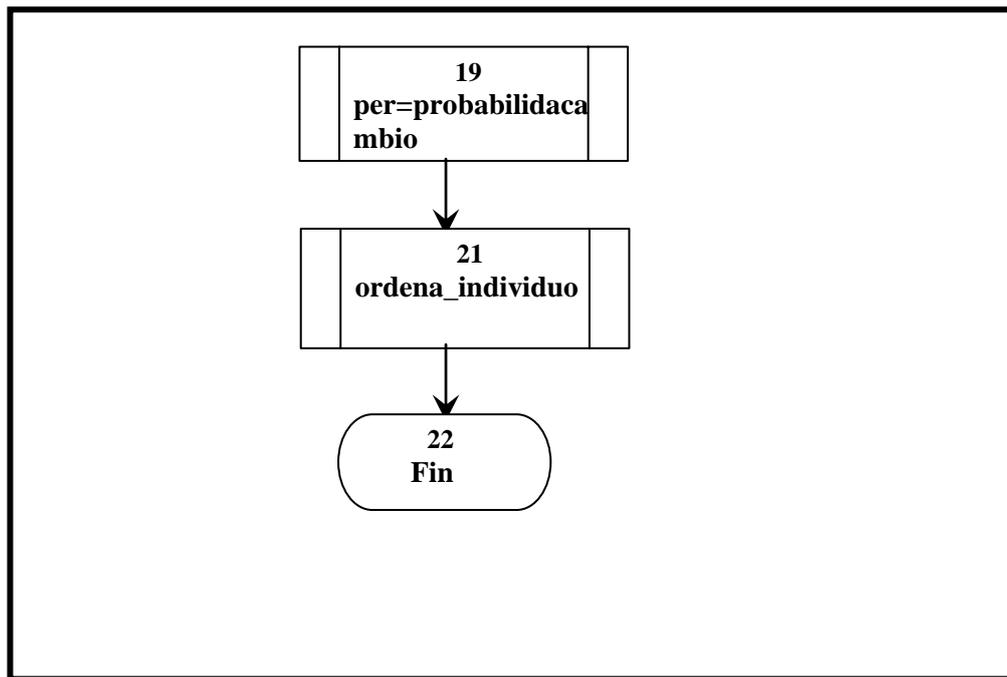


Figura 25. Representación del diagrama de flujo de la función 'cruce'.

A su vez, esta función hace referencia a distintas funciones.

### Función PREVENCIÓN\_INCESTO

Tabla 33. Tabla representativa de la función prevención\_incesto

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
Prevencción_incesto	int prevencion_incesto(MATINDIV *padre,MATINDIV *madre,int total);	1) MATINDIV *padre 2) MATINDIV *madre 3) int total	int	Función que se ocupa de comparar los dos individuos a cruzar de forma que avise si se parecen mucho.

### *Parámetros de entrada*

Se le pasa como primer argumento el padre que quiere ser cruzado; como segundo argumento se le pasa la madre que quiere ser cruzada; como tercer argumento se le pasa el número total de caminos posible entre todos los pares de la red.

### *Parámetros de salida*

Cuando esta variable toma el valor 0 indica que no se considera incesto el cruce entre los dos individuos que se indican a la entrada. Por el contrario, si toma el valor 1 indica que el cruce esas dos variables no debe ser efectuado. La función recibe este valor y actúa en consecuencia.

## **Función PROBABILIDADCAMBIO**

**Tabla 34. Tabla representativa de la función probabilidad\_cambio**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
Probabilidad_cambio	float probabilidad_cambio(MATI NDIV *p);	1) MATIND I *p	float	Función que determina las probabilidades de cambio de cada uno de los individuos de la población

### *Parámetros de entrada*

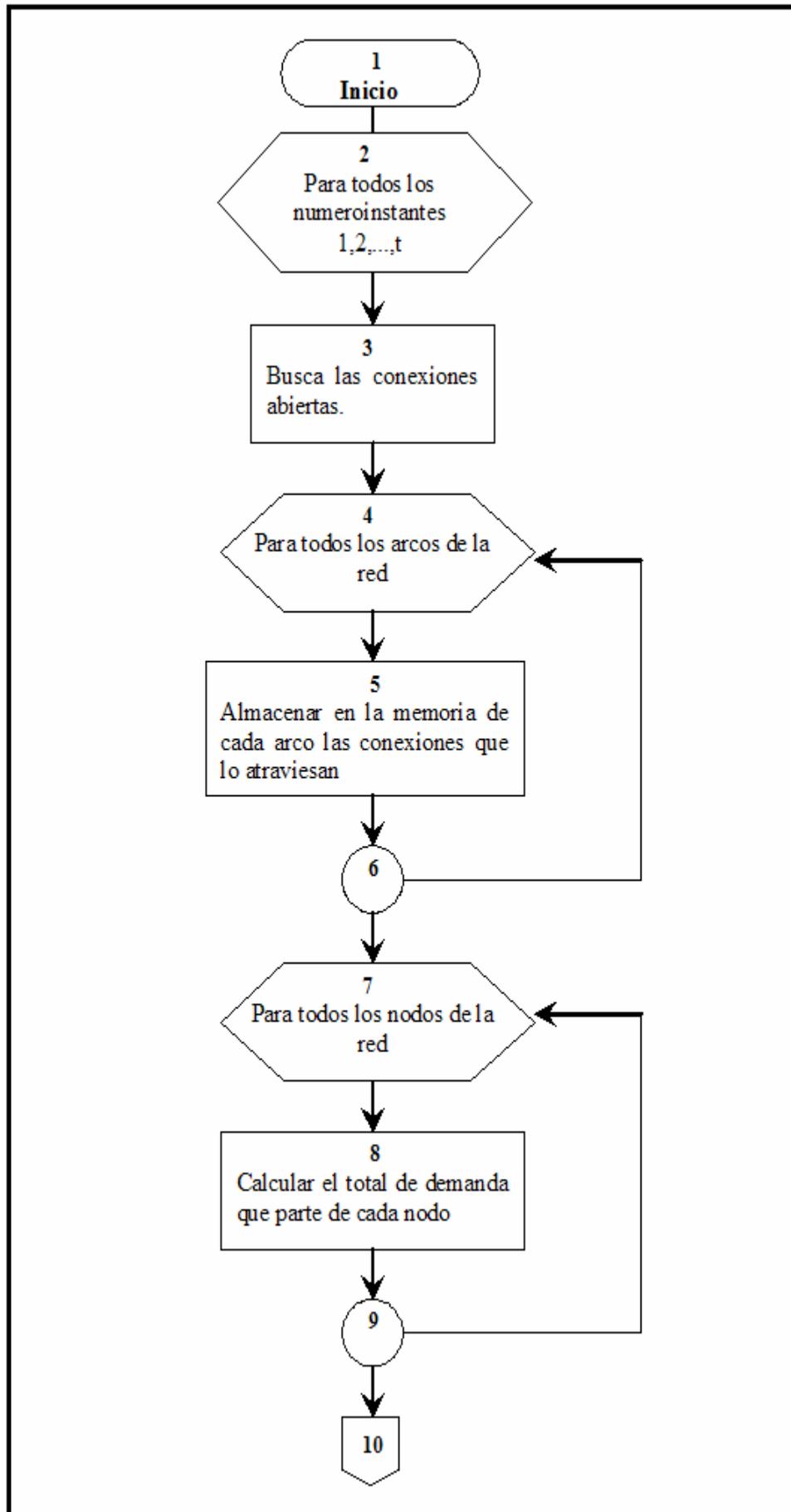
Se le pasa como argumento una lista con todos los cromosomas de los individuos de la población y sus correspondientes pérdidas.

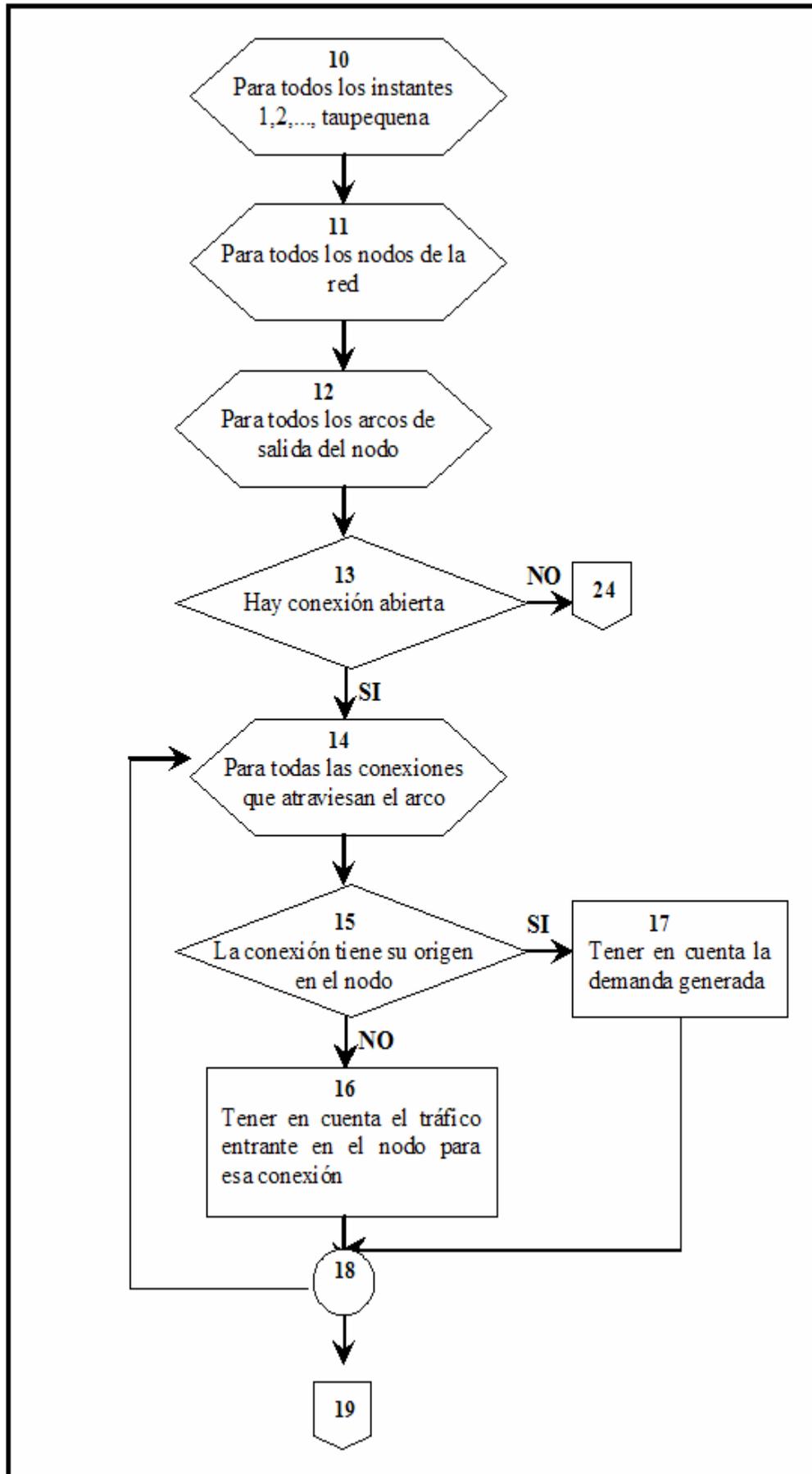
### *Parámetros de salida*

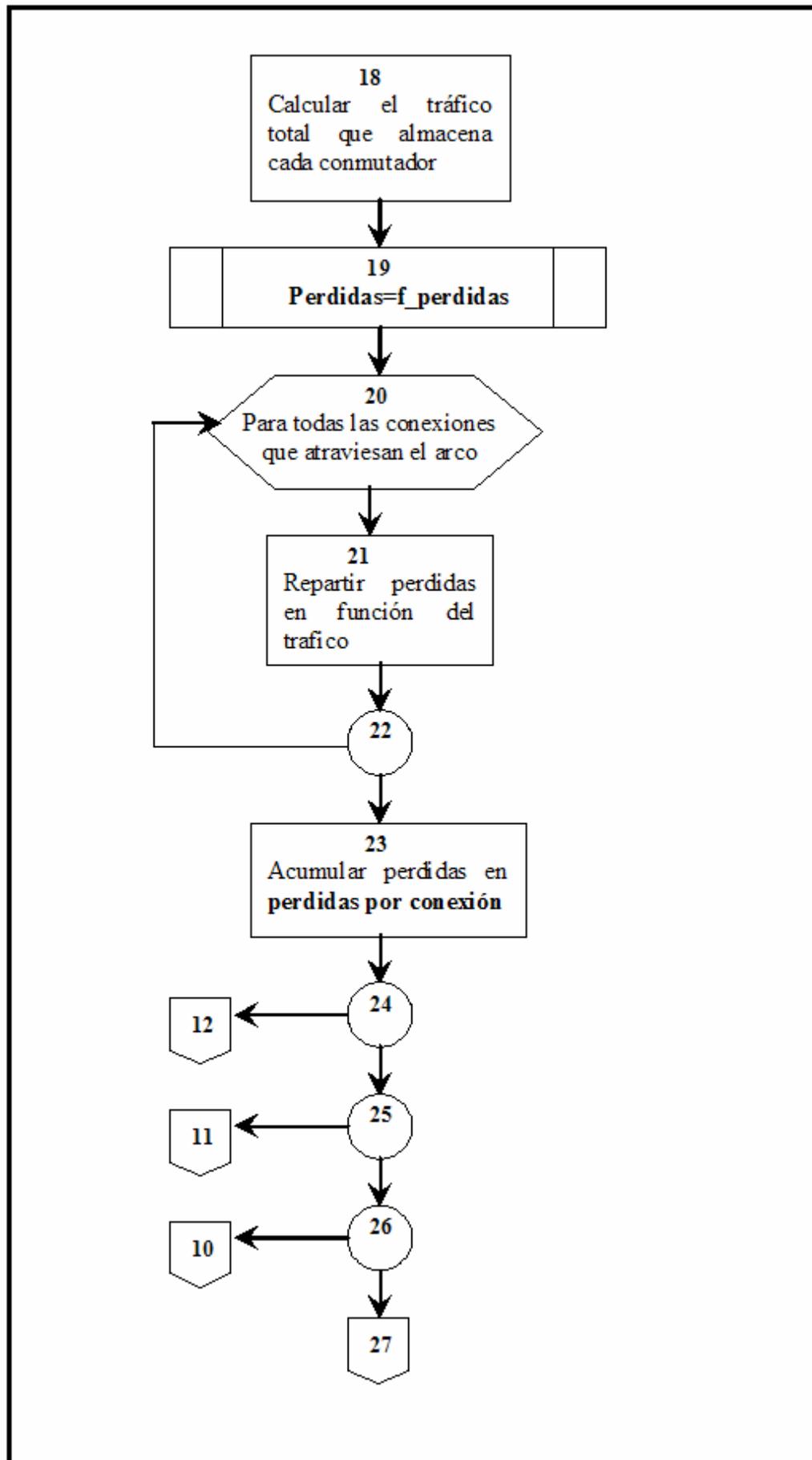
Devuelve una variable tipo flota cuyo valor es la aptitud (fitness) del individuo a cambiar.

#### **10.8.5 Función calculaaptitud**

Esta función se encarga de evaluar a cada uno de los individuos en función de las pérdidas que se producen en los caminos elegidos. Es una función importante dentro del algoritmo genético, ya que es la encargada de calcular las aptitudes. También se ocupa de encontrar las conexiones abiertas en cada intervalo de tiempo, ya que es necesario tener esta información antes de calcular las pérdidas. Puede verse en la Figura 26:







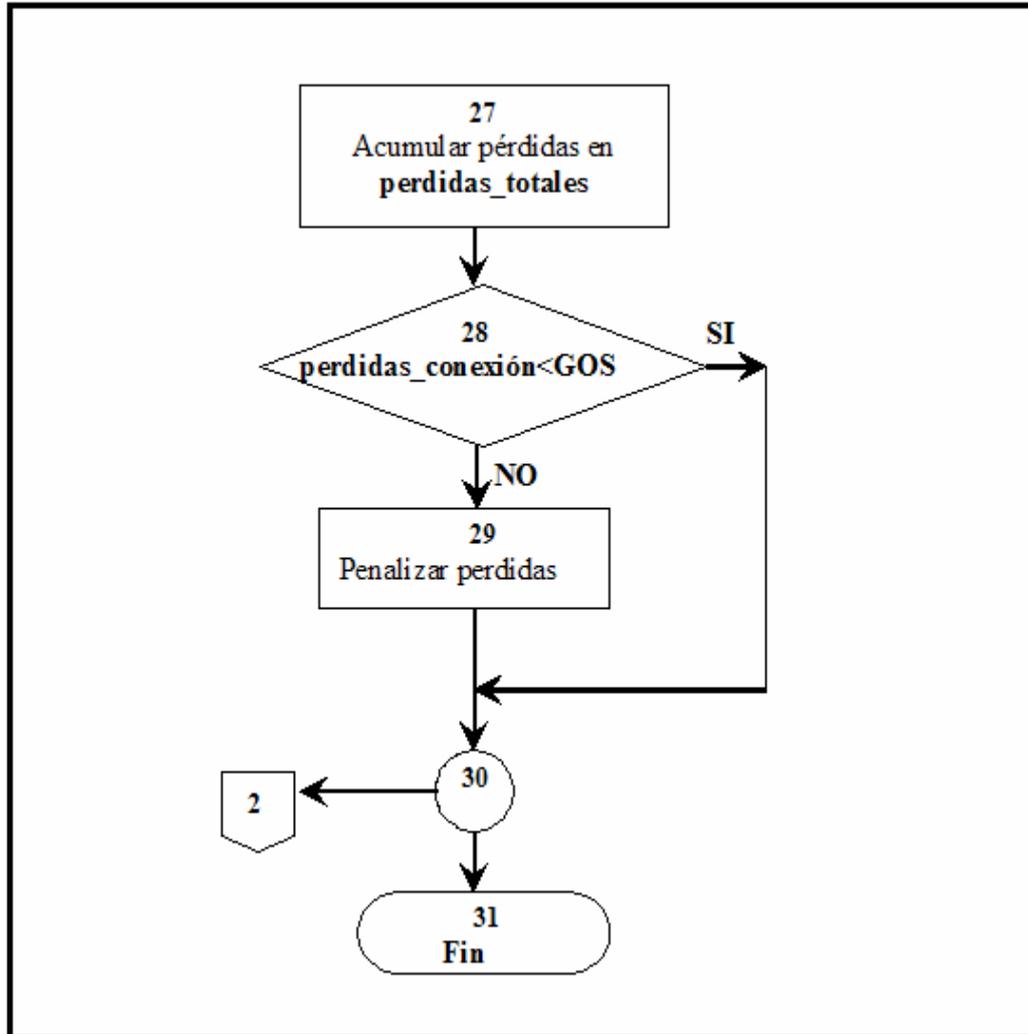


Figura 26. Representación del diagrama de flujo de la función 'calculaaptitud'

A su vez, esta función hace referencia a la función *f\_pérdidas* y a la función *poisson*.

### Función F\_PÉRDIDAS

Tabla 35. Tabla representativa de la función *f\_pérdidas*

Descripción / función	Prototipo	Parámetros de entrada	Parámetros de salida	Descripción de la función
F_pérdidas	Float f_perdidas(float trafico);	1) float tráfico	float	Función que calcula las

				pérdidas que se producen en los conmutadores
--	--	--	--	--

#### *Parámetros de entrada*

Se le pasa como argumento de entrada el tráfico de células de entrada al conmutador.

#### *Parámetros de salida*

Devuelve las pérdidas producidas que vendrán representadas en una variable de tipo float. Estas pérdidas son las pérdidas totales que se producen en un instante de tiempo  $\tau$ , no se diferencia entre las distintas conexiones que utilizan el conmutador.

### **Función POISSON**

**Tabla 36. Tabla representativa de la función de poisson.**

<b>Descripción / función</b>	<b>Prototipo</b>	<b>Parámetros de entrada</b>	<b>Parámetros de salida</b>	<b>Descripción de la función</b>
poisson	float poisson(int numcel, float lamda);	1) int numcel 2) float lamda	float	Función que calcula las pérdidas a través de la distribución de poisson

#### *Parámetros de entrada*

Se le pasa como primer argumento el número de células necesarias para calcular la distribución de *poisson*; como segundo argumento se le pasa el tráfico de entrada al conmutador en células/ranura.

#### *Parámetros de salida*

Devuelve en una variable flotante el valor de pérdidas.

## 11 ANÁLISIS DE REDES DE ENSAYO. CALIBRACIÓN DE PARÁMETROS.

A continuación, se van a estudiar diversas propuestas de redes, variando para cada una de ellas las condiciones topológicas y de tráfico. Esto es: el número de conexiones establecidas durante cada intervalo de conexión, el grado de servicio y el número de nodos y de enlaces.

El estudio se presenta para una red de siete, una de diez y una de quince nodos. En la Tabla 37 se pueden apreciar diversas topologías y los correspondientes parámetros que irán variando para el análisis. En la Tabla 38, se observan los distintos parámetros del algoritmo genético a tener en cuenta para el estudio de las tres redes.

Obsérvese que, para los tres casos propuestos, se utilizará una probabilidad de cruce de 0.85 y una probabilidad de reemplazo de 0.6, como puede observarse en la Tabla 38:

**Tabla 37. Tabla representativa de las diversas redes de ensayo y sus correspondientes topologías**

Parámetros / Diversas redes	Datos Topológicos		Tráfico	
	Nodos	Enlaces	Número de conexiones	GoS
<b>Red A</b>	7	18	2-3	2-6
<b>Red B</b>	10	30	4-10-20	2-6
<b>Red C</b>	15	46	10	2-6

**Tabla 38. Tabla representativa de los distintos parámetros del algoritmo para la realización de gráficas**

Parámetros / Diversas redes	q(Probabilidad de reemplazo)	Pc(Probabilidad de cruce)	Tamaño de la población	Intervalo variación de generaciones para la convergencia(ciclos)
<b>Red A</b>	0.6	0.85	10	(30-120)

Parámetros / Diversas redes	q(Probabilidad de reemplazo)	Pc(Probabilidad de cruce)	Tamaño de la población	Intervalo variación de generaciones para la convergencia(ciclos)
Red B	0.6	0.85	5-10-15-20	(50-80)
Red C	0.6	0.85	10	(50-100)

### 11.1 ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED A

Red de 7 nodos y 18 enlaces. Su topología se muestra en la se muestra a continuación.

Figura 27, que

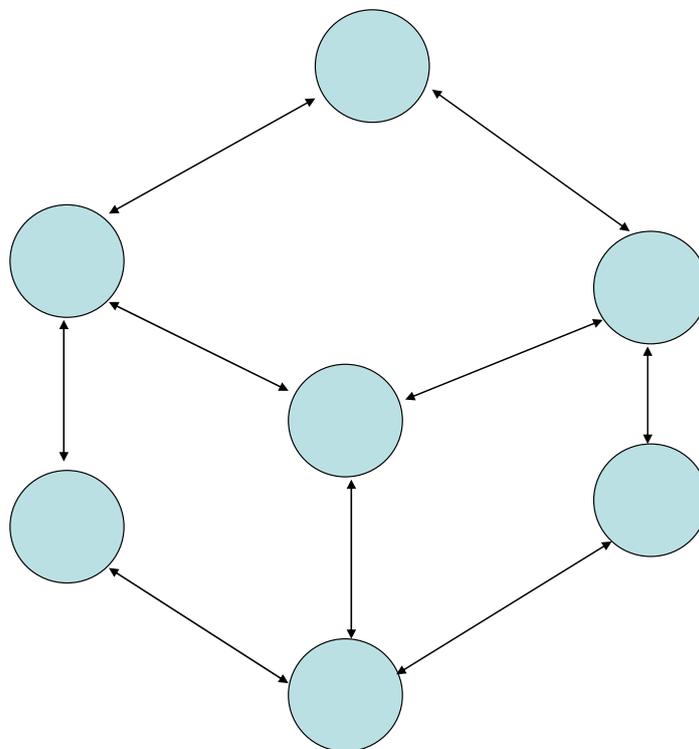


Figura 27. Topología de la red de 7 nodos y 18 enlaces bidireccionales.

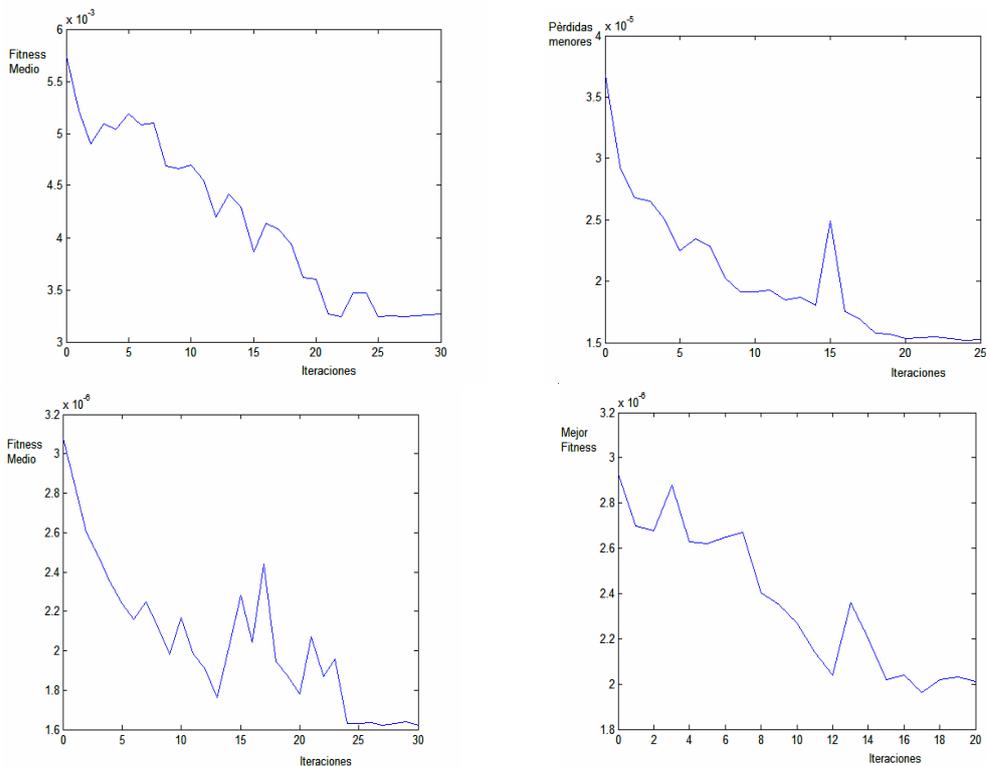


**Nodo de Conmutación**  
**Enlaces bidireccionales**

**Figura 28. Representación de la simbología utilizada para la realización de la red**

### 11.1.1 Estudio de las pérdidas medias frente al número de iteraciones del algoritmo

En las siguientes gráficas, se representa el valor medio de las pérdidas totales alcanzadas respecto a cada generación del algoritmo genético para el caso de dos conexiones establecidas durante cada intervalo de conexión y distintos tamaños de búfer. Se representan cuatro casos en la Figura 29:



**Figura 29. Representación de las pérdidas medias respecto al número de iteraciones del algoritmo. (Superior izquierda) Caso de B=1. (Superior derecha) Caso de B=2. (Inferior izquierda) Caso de B=3. (Inferior derecha) Caso de B=4.**

## Interpretación de los resultados

En la Figura 29, formada por cuatro gráficas, cada una para un tamaño de búfer distinto, puede verse como a medida que aumenta el búfer las pérdidas medias de la población tiende a disminuir.

Se observa que la función no es monótonamente decreciente, sino que debido a la componente de aleatoriedad del algoritmo, algunas veces se crean individuos que generan puntualmente en una determinada población mayores pérdidas provocando picos en las gráficas. A pesar de ello, la tendencia de la gráfica es descendente, como cabía esperar, apreciándose la convergencia del algoritmo.

A continuación, en la Tabla 39, se compara el orden de las pérdidas para las cuatro gráficas anteriores.

**Tabla 39. Comparación del orden de las pérdidas para cada uno de los cuatro casos diferentes.**

<b>Parámetros / Red de 7 nodos variando B</b>	<b>Orden de magnitud de las pérdidas</b>	<b>Tamaño de la población</b>
<b>7 nodos( B=1)</b>	$10^{-3}$	10
<b>7 nodos(B=2)</b>	$10^{-5}$	10
<b>7 nodos(B=3)</b>	$10^{-6}$	10
<b>7 nodos(B=4)</b>	$10^{-6}$	10

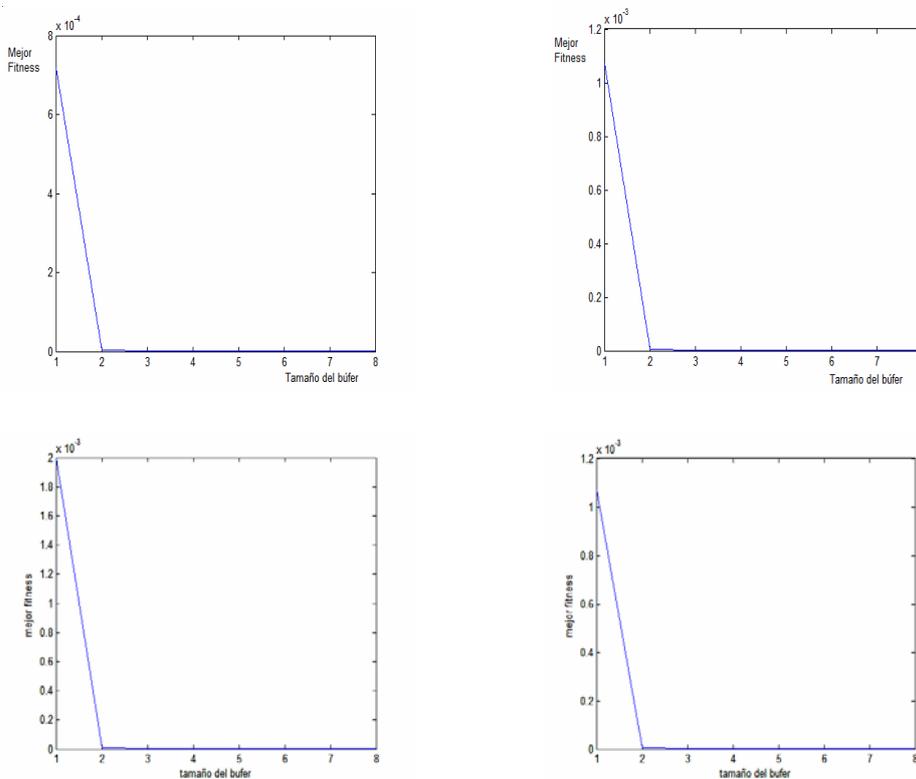
En la Tabla 39 se observa que, a medida que aumenta el tamaño del búfer, las pérdidas disminuyen considerablemente. Puede verse que las pérdidas para un tamaño de búfer de una célula son del orden de  $10^{-3}$ , mientras que para un tamaño de búfer de tres células son del orden de  $10^{-6}$ . Para un tamaño de búfer mayor a tres células, aunque siguen siendo del orden de  $10^{-6}$ , son mayores que para un tamaño de búfer igual a tres células.

La explicación que tienen estos datos es que, para un número de conexiones establecidas baja, dos en este caso, con un tamaño de búfer relativamente pequeño se obtienen pérdidas muy bajas. La elección más adecuada estaría, en este caso, en un búfer de tres células.

### 11.1.2 Estudio de la solución alcanzada respecto al tamaño del búfer

**Tabla 40.** Tabla representativa de los parámetros utilizados para la realización de las gráficas

Parámetros / Casos	Orden de magnitud de las pérdidas para B=1	Tamaño de la población
GoS=3	$10^{-4}$	10
GoS=4	$10^{-3}$	10
GoS=5	$10^{-3}$	10
GoS=6	$10^{-3}$	10



**Figura 30.** Representación de la mejor solución alcanzada respecto al tamaño de búfer. (Superior izquierda) Caso para GoS=3. (Superior derecha) Caso para GoS=4. (Inferior izquierda) Caso para GoS=5. (Inferior derecha) Caso para GoS=6.

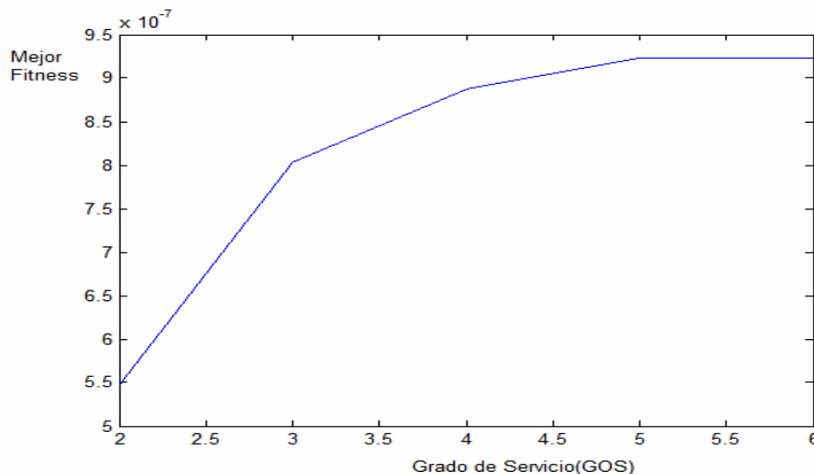
## Interpretación de los resultados

En la Figura 30, compuesta de cuatro gráficas, representan la solución alcanzada frente al tamaño del búfer para distintos valores del grado de servicio. El grado de servicio determina el número de procesos de conmutación que se pueden producir entre cada origen y destino. Actúa como un elemento reductor del retardo de comunicación, a la vez que sobre las pérdidas en la comunicación. Éste número de procesos de conmutación se ha variado desde tres hasta seis.

Se observa que, a medida que aumenta el grado de servicio, las pérdidas aumentan. Las mayores se dan para un grado de servicio de seis, y las menores para un grado de servicio de tres.

Todas coinciden, como se describió antes, que a partir de un tamaño de búfer de tres procesos de conmutación, la solución óptima no mejora, debido al escaso número de conexiones establecidas durante cada intervalo de conexión.

### **11.1.3 Estudio de la solución alcanzada respecto al grado de servicio**



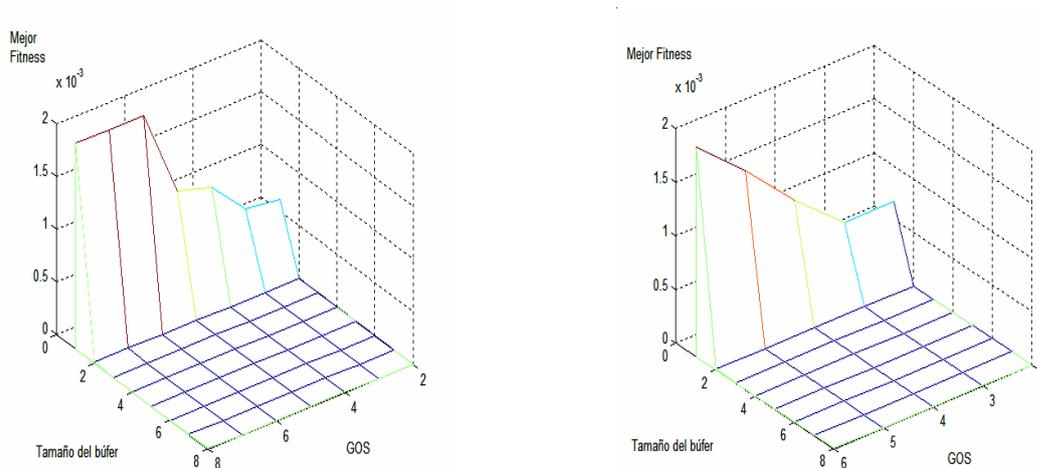
**Figura 31. Representación de la mejor solución respecto al grado de servicio. Caso de B=3 y 2 conexiones establecidas durante cada intervalo de conexión.**

## Interpretación de los resultados

Conforme aumenta número de procesos de conmutación permitidos, la solución alcanzada empeora. Es decir, las pérdidas aumentan como puede verse en la Figura 31.

Se observan dos fuertes cambios de pendiente para un grado de servicio de tres y de cinco.

#### 11.1.4 Estudio de la solución alcanzada frente al tamaño del búfer y grado de servicio



**Figura 32. Representación mejor solución frente a tamaño de búfer y GoS. (Izquierda) Evolución de la solución para 2 conexiones simultáneas y (derecha) para 3 conexiones simultáneas.**

#### Interpretación de los resultados

Para tamaño de búferes mayores, la solución alcanzada mejora disminuyendo las pérdidas. Se observa que a partir de un tamaño de búfer de tres células la solución alcanzada no mejora. Un tamaño de búfer de tres o cuatro células sería el adecuado.

Al restringir menos el grado de servicio permitiendo caminos más largos (mayor número de saltos entre par origen y par destino), las pérdidas empiezan a crecer considerablemente.

La franja azul de las gráficas es del orden de  $10^{-6}$  y  $10^{-7}$ . Para un grado de servicio alto y un tamaño de búfer de dos células se alcanza las pérdidas más altas.

Para la gráfica de tres conexiones, los cambios de pendiente son más suaves que para la de dos conexiones. Es decir, a medida que aumenta el número de conexiones,

para cambios de tamaño de búfer y grado de servicio, las soluciones alcanzadas varían más suavemente.

### 11.1.5 Estudio de la solución alcanzada en función del número de iteraciones del algoritmo.

Tabla 41. Tabla representativa de los parámetros utilizados para la realización de las gráficas

Parámetros / Tamaño de Búfer	GoS (Grado de Servicio)	Orden de magnitud de las pérdidas	Generaciones en la que se mejora función Objetivo	Conexiones establecidas
B=1	5	$10^{-3}$	3,5 y 17	3
B=2	5	$10^{-5}$	6,14 y 34	3
B=2	6	$10^{-5}$	4,17,23,33 y 47	3
B=3	6	$10^{-6}$	15,62 y 103	3

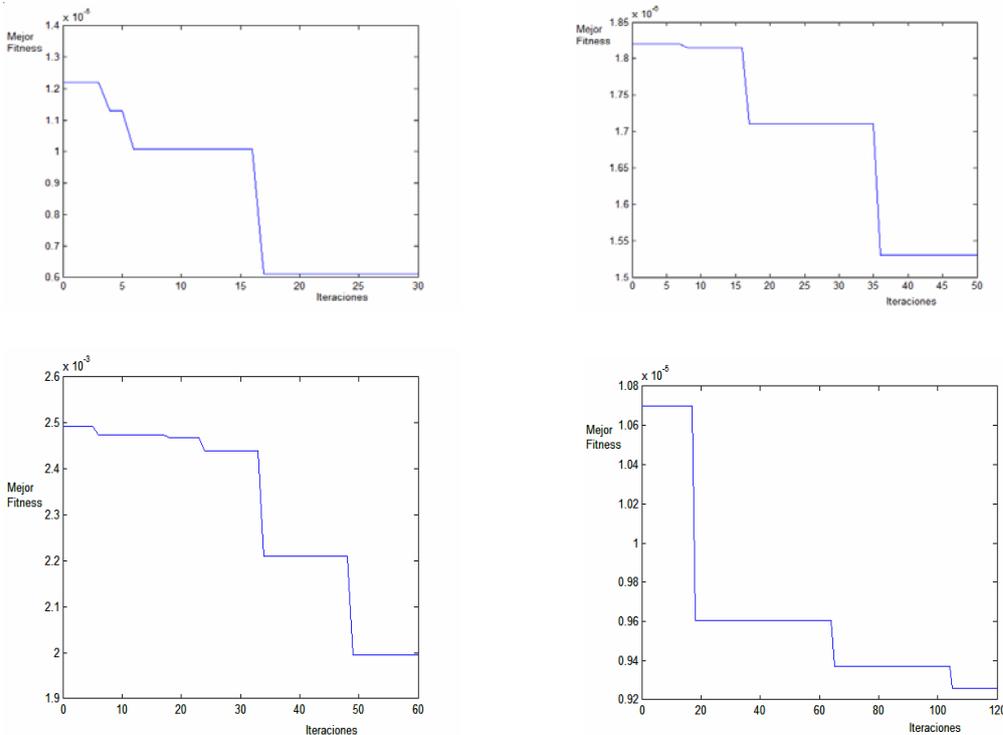


Figura 33. Representación de la solución alcanzada en función del número de iteraciones del algoritmo. (Superior izquierda) Caso B=2 y GoS=5. (Superior derecha) Caso B=3 y GoS=5. (Inferior izquierda) Caso B=1 y GoS=6. (Inferior derecha) Caso B=2 y GoS=6.

## Interpretación de los resultados

A medida que el algoritmo genético se ejecuta, la solución alcanzada mejora, es decir, disminuye hasta una cierta generación a partir de la cual no mejora más, haciéndose las gráficas planas.

Como puede verse en la Figura 33, cada gráfica converge en ciclos diferentes. Eso es debido al criterio de parada del algoritmo genético. La detención del algoritmo puede tener lugar debido a dos causas: que se haya completado el número de iteraciones admisibles, N\_MAX, o que el grado de semejanza de los individuos sea muy elevado. Pues bien, para caso concreto, estas circunstancias se dan en ciclos diferentes.

### 11.1.6 Estudio de la solución alcanzada en función del tráfico

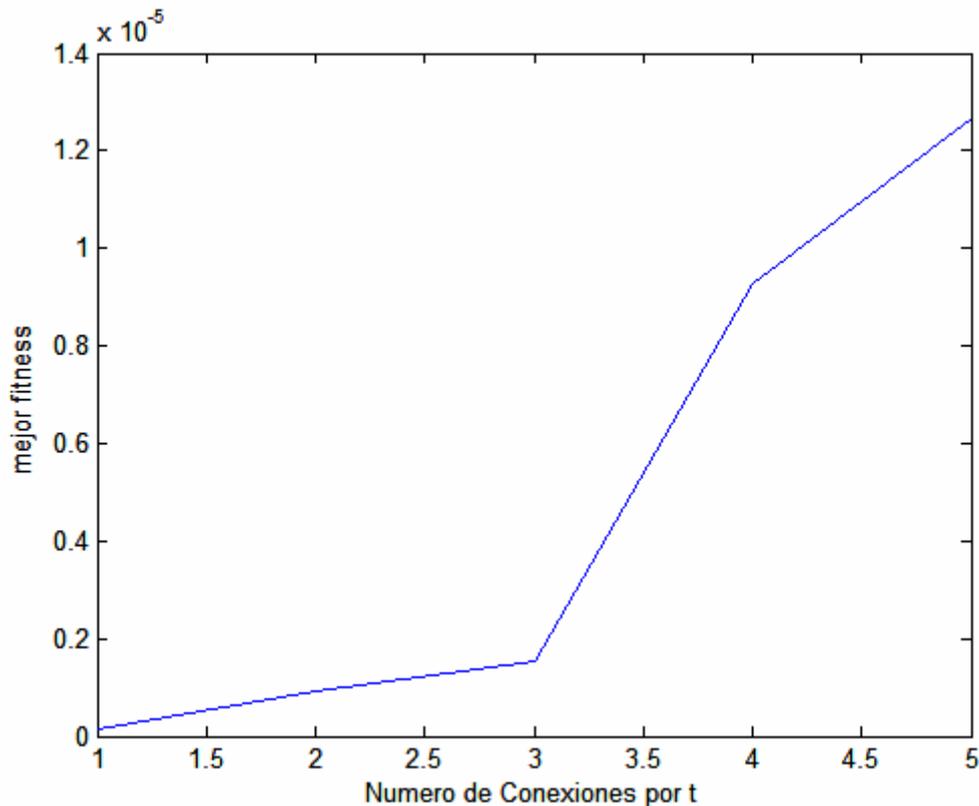


Figura 34. Representación de la solución alcanzada en función del tráfico. Caso B=3

## Interpretación de los resultados

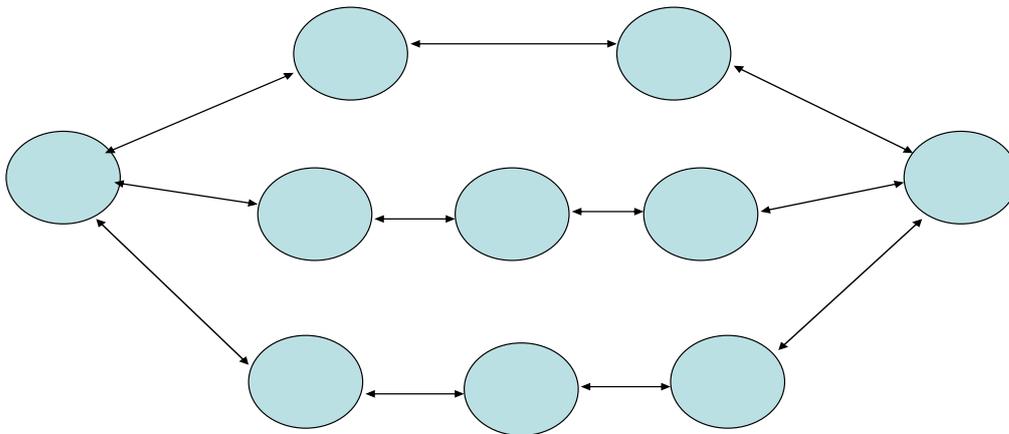
Cuando incrementa el tráfico – número de conexiones- la solución alcanzada empeora debido a la congestión de la red. Consecuentemente, las pérdidas se hacen más grandes.

La Figura 34 se refiere al caso concreto de un tamaño de búfer de tres células y dos conexiones, debido a que para la topología de red estudiada, resultaba el más adecuado como ya se indicó.

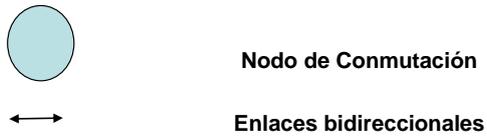
### **11.2 ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED B**

Red de 10 nodos y 30 enlaces. Su topología se muestra en la 35, implementada a continuación:

Figura

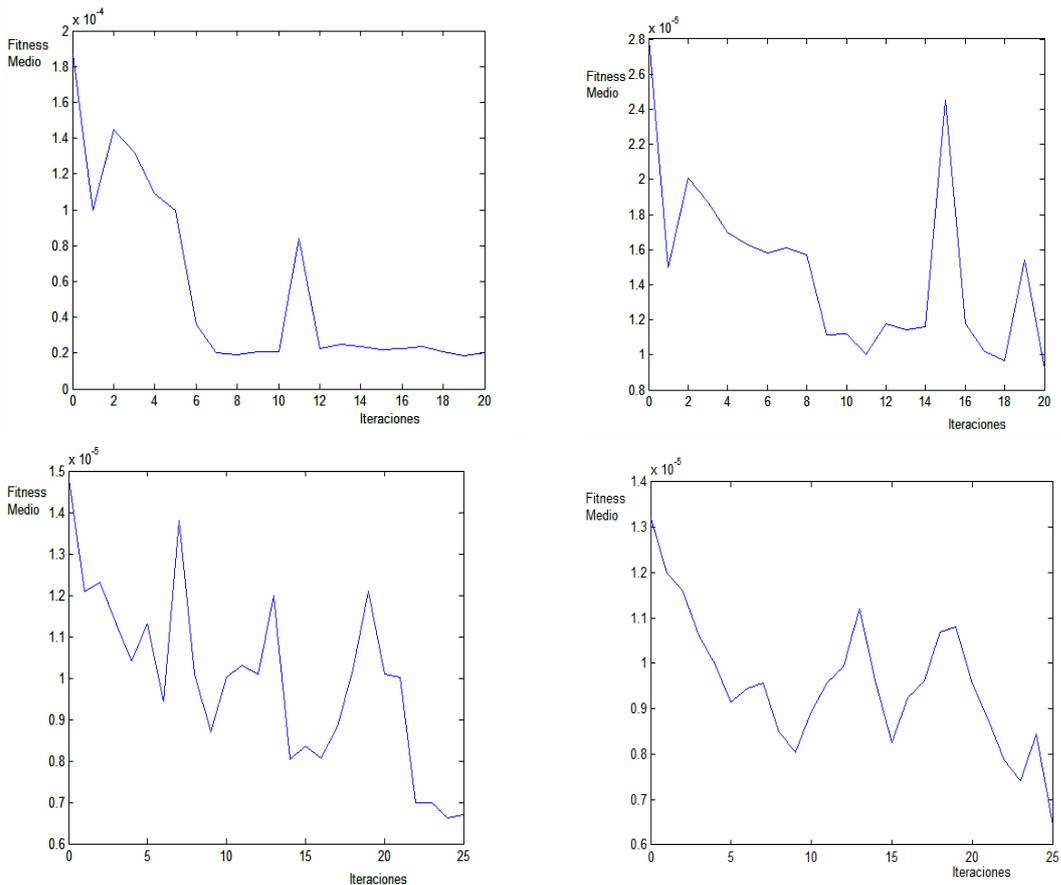


**Figura 35. Topología de la red de 10 nodos y 30 enlaces bidireccionales.**



**Figura 36.** Representación de la simbología utilizada para la realización de la red

### 11.2.1 Estudio de las pérdidas medias frente al número de iteraciones del algoritmo



**Figura 37.** Representación de las pérdidas medias respecto al número de iteraciones del algoritmo. (Superior izquierda) Caso de B=3. (Superior derecha) Caso de B=4. (Inferior izquierda) Caso de B=5. (Inferior derecha) Caso de B=6.

## Interpretación de los resultados

En la Figura 37, dividida en cuatro gráficas, cada una para un tamaño de búfer distinto, puede verse como a medida que aumenta el búfer las pérdidas medias disminuyen.

Se observa que la función no es monótonamente decreciente, sino que debido a la componente de aleatoriedad del algoritmo, algunas veces se crean individuos de mayores pérdidas provocando picos en las gráficas. A pesar de ello, la tendencia de la gráfica es descendente.

En la Tabla 42 se compara el orden de pérdidas para las cuatro gráficas anteriores.

**Tabla 42. Comparación del orden de las pérdidas para cada uno de los cuatro casos.**

<b>Parámetros / Red de 7 nodos variando B</b>	<b>Orden de magnitud de las pérdidas</b>	<b>Tamaño de la población</b>
<b>10 nodos( B=3)</b>	$10^{-5}$	10
<b>10 nodos(B=4)</b>	$10^{-6}$	10
<b>10 nodos(B=5)</b>	$10^{-6}$	10
<b>10 nodos(B=6)</b>	$10^{-6}$	10

### **11.2.2 Estudio de la solución alcanzada en función del número de iteraciones del algoritmo**

**Tabla 43. Tabla representativa de los parámetros utilizados para la realización de las gráficas**

<b>Parámetros / Tamaño de Búfer</b>	<b>GoS (Grado de Servicio)</b>	<b>Orden de magnitud de las pérdidas</b>	<b>Generaciones de mejora de función objetivo</b>	<b>Conexiones establecidas</b>
<b>B=3</b>	6	$10^{-5}$	3 y 50	4

Parámetros / Tamaño de Búfer	GoS (Grado de Servicio)	Orden de magnitud de las pérdidas	Generaciones de mejora de función objetivo	Conexiones establecidas
<b>B=4</b>	6	$10^{-6}$	4,41 y 48	4
<b>B=5</b>	6	$10^{-6}$	6,8,24 y 32	4
<b>B=6</b>	6	$10^{-6}$	3,7 y 28	4

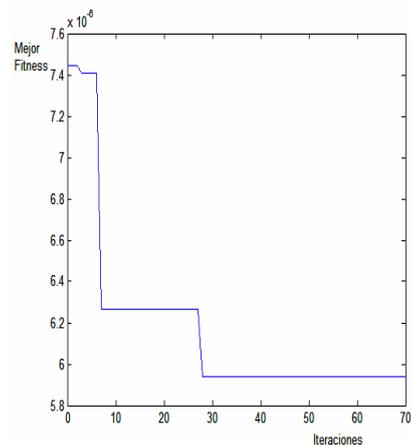
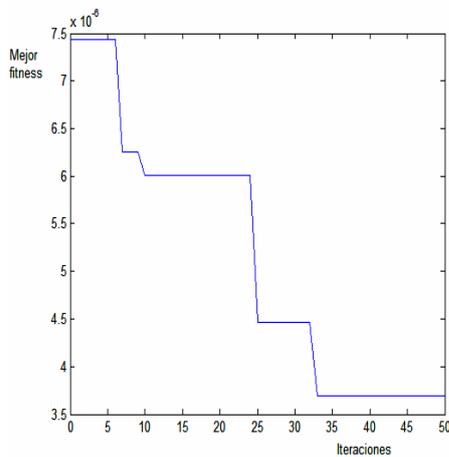
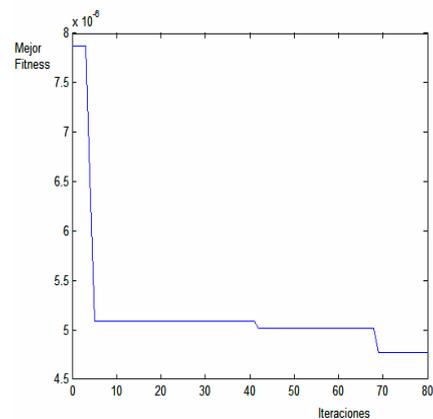
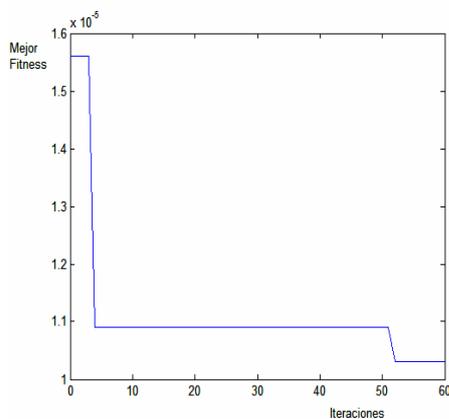


Figura 38. Representación de la solución alcanzada en función del número de iteraciones del algoritmo. (Superior izquierda) Caso B=3 y GoS=6. (Superior derecha) Caso B=3 y GoS=6. (Inferior izquierda) Caso B=5 y GoS=6. (Inferior derecha) Caso B=6 y GoS=6.

## Interpretación de los resultados

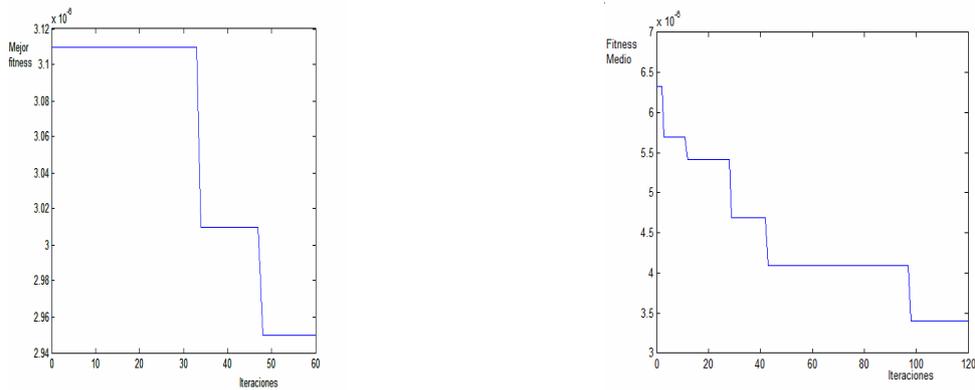
Al igual que para la red A, el algoritmo genético va tendiendo hacia individuos cada vez mejores.

Al observar las cuatro gráficas de la Figura 38 y la Tabla 43, se comprueba que conforme aumenta el tamaño del búfer no sólo disminuyen las pérdidas, sino que se producen más cambios de pendiente. Para el caso de un tamaño de búfer de tres células se producen dos cambios de pendiente y para un tamaño de búfer de cinco células se producen cuatro. Para un tamaño de búfer de seis células se puede observar que la solución alcanzada es peor, es decir, las pérdidas más pequeñas alcanzadas por un tamaño de búfer de seis células son mayores que utilizando uno de cinco. Más concretamente, para un tamaño de búfer de seis células es de  $5,93^{-6}$ , y para un tamaño de búfer de cinco células es de  $3,64^{-6}$ . Este fenómeno se debe a la aleatoriedad intrínseca de los algoritmos genéticos.

### **11.2.3 Estudio de la solución alcanzada en función del número de iteraciones del algoritmo. Para tamaños de población diez y veinte individuos.**

**Tabla 44. Tabla representativa de los parámetros utilizados para la realización de las gráficas**

<b>Parámetros / Tamaño de Búfer</b>	<b>GoS (Grado de Servicio)</b>	<b>Orden de magnitud de las pérdidas</b>	<b>Tamaño de la Población</b>	<b>Generaciones de mejora de la función Objetivo</b>	<b>Conexiones establecidas</b>
<b>B=4</b>	4	$10^{-6}$	10	31 y 47	4
<b>B=4</b>	6	$10^{-6}$	20	3,12,29,43 y 99	4



**Figura 39. Representación de la solución alcanzada en función del número de iteraciones del algoritmo. (Izquierda) Caso de un tamaño de población igual a 10 individuos. (Derecha) Caso de un tamaño de población igual a 20 individuos.**

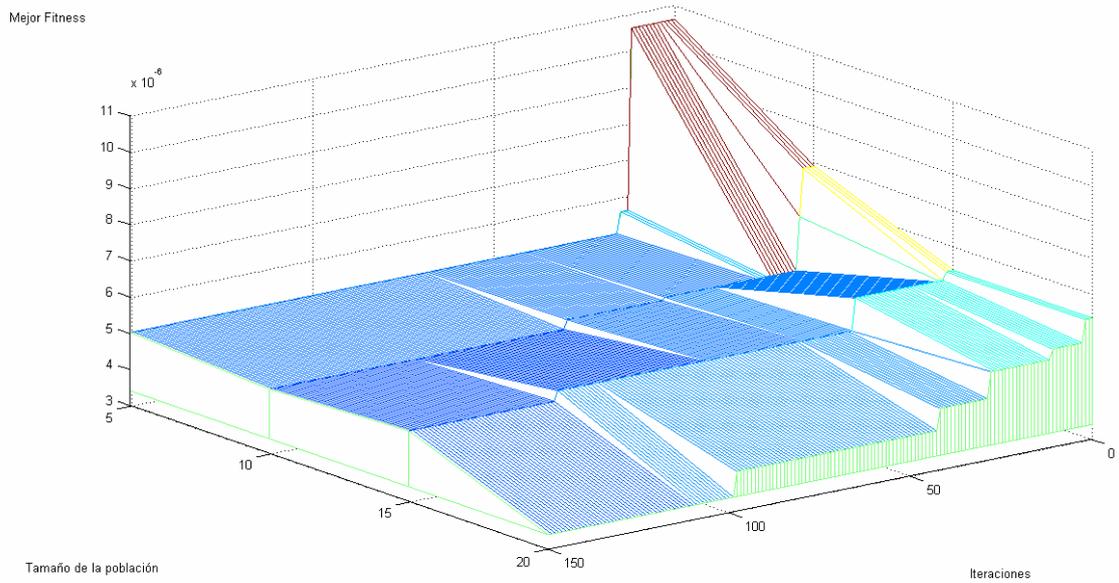
### Interpretación de los resultados

Al igual que cuando aumenta el tamaño del búfer se producen más cambios de pendiente, cuando aumenta el tamaño de la población no converge tan rápido el algoritmo genético, sino se producen más cambios de pendiente. Puede verse en la Figura 39.

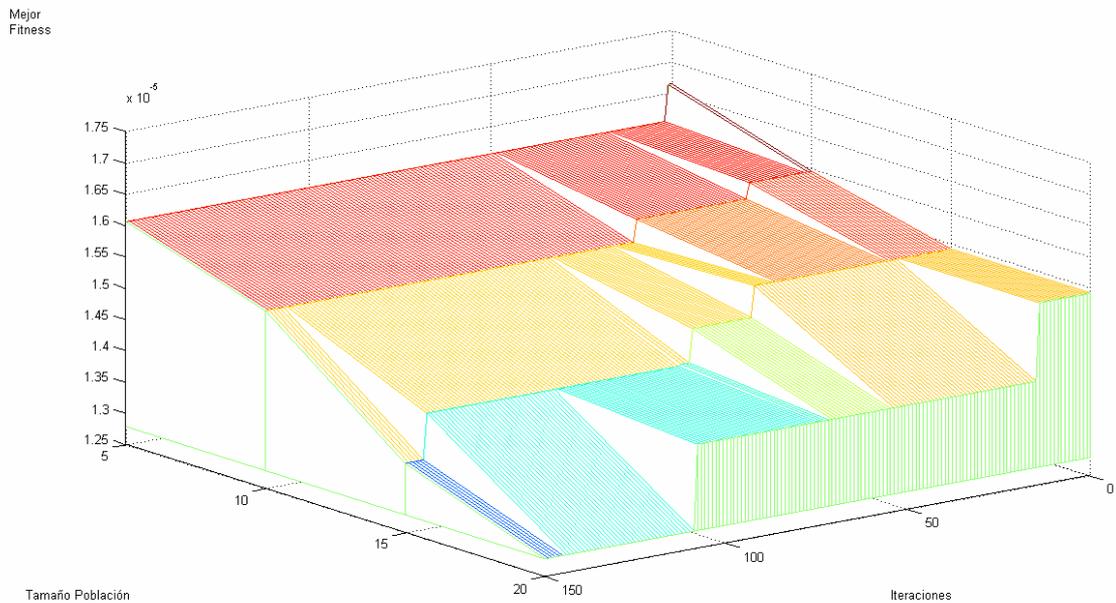
### **11.2.4 Estudio de la solución alcanzada en función del número de iteraciones del algoritmo y del tamaño del búfer.**

**Tabla 45. Representación de los parámetros utilizados para la realización de las gráficas**

<b>Parámetros Casos</b>	<b>Tamaño población</b>	<b>Tamaño de búfer</b>	<b>Orden de magnitud de las pérdidas</b>
<b>10 conexiones</b>	5-10-15-20	4	$10^{-6}$
<b>20 conexiones</b>	5-10-15-20	6	$10^{-5}$



**Figura 40. Representación de la mejor solución encontrada en función del tamaño de la población y de las iteraciones. Evolución de la solución para el caso de 10 conexiones establecidas durante cada intervalo de conexión**



**Figura 41. Representación de la mejor solución encontrada en función del tamaño de la población y de las iteraciones. Evolución de la solución para el caso de 20 conexiones establecidas durante cada intervalo de conexión.**

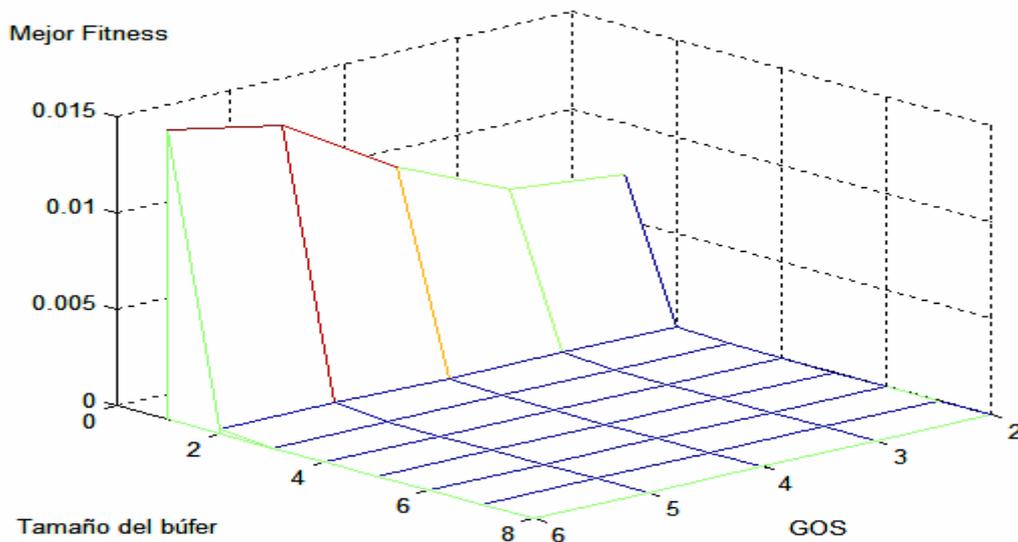
### Interpretación de los resultados

La Figura 40 y la Figura 41 representan la solución alcanzada frente al número de iteraciones y al tamaño de la población.

Para el caso de diez conexiones establecidas durante cada intervalo de conexión, el tamaño de la población no resulta tan importante a medida que aumenta el número de iteraciones del algoritmo genético.

Un tamaño de población de diez individuos podría ser adecuado. Conforme aumenta el número de individuos, se producen más cambios de pendiente, y, por otro lado, una mayor solución final.

A diferencia del caso de diez conexiones, para el caso de veinte, no sólo las pérdidas son mayores-se incrementa el número de conexiones- sino que el algoritmo genético mejora mucho a medida que incrementa el tamaño de la población, lo cual es esperable. Se observa la mejora progresiva de las pérdidas desde un tamaño de población de cinco individuos hasta uno de veinte.



**Figura 42. Representación de la solución alcanzada frente al tamaño del búfer y al grado de servicio. Caso de 4 conexiones establecidas durante cada intervalo.**

### Interpretación de los resultados

En la Figura 42 se puede observar que, para tamaño de búferes mayores, la solución alcanzada mejora. Puede observarse que, a partir de un tamaño de búfer cinco, la solución alcanzada no mejora más. Un tamaño de búfer de cuatro o cinco células sería el adecuado.

Al restringir menos el grado de servicio, las pérdidas empiezan a crecer considerablemente.

La franja azul de las gráficas es del orden de  $10^{-6}$  y  $10^{-7}$ . Para un grado de servicio bajo y un tamaño de búfer de una o dos células se alcanza las pérdidas más altas.

### 11.2.5 Estudio de la solución alcanzada en función del tráfico

Tabla 46. Representación de los parámetros para la realización de gráficas

Parámetros	Tamaño Población	GoS (Grado de Servicio)	Tamaño del búfer
Caso	10 Individuos	6	5

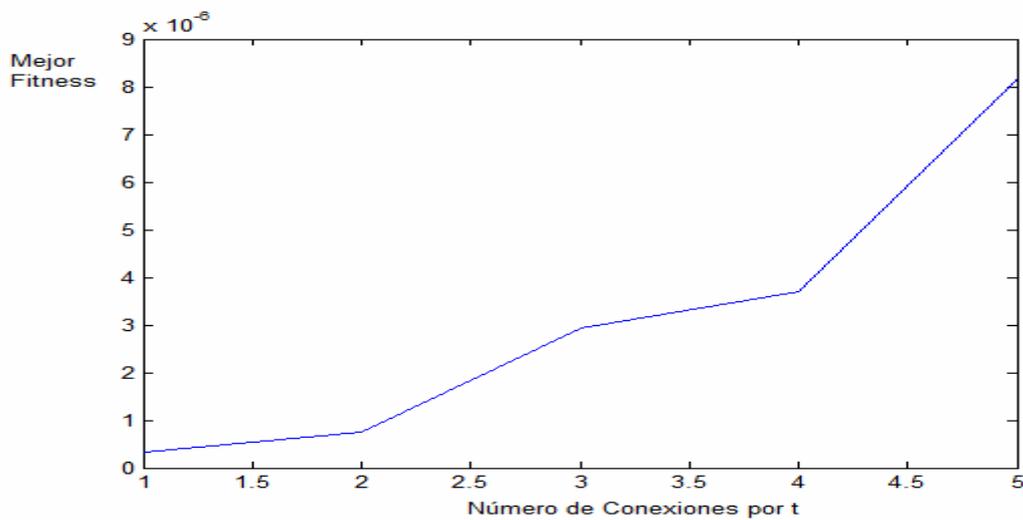


Figura 43. Representación de la mejor solución alcanzada en función del número de conexiones establecidas durante cada intervalo de conexión.

#### Interpretación de los resultados

Se representa y se hace el análisis para el caso de un tamaño de búfer de cinco células, que es el adecuado.

Conforme aumenta el tráfico- numero de conexiones-, la solución alcanzada empeora debido a que se está sobrecargando la red, como puede verse en la Figura 43.

### 11.3 ANÁLISIS DE LOS PARÁMETROS DE DISEÑO PARA LA RED C

Red de 15 nodos y 46 enlaces. Su topología se muestra en la Figura 44, implementada a continuación:

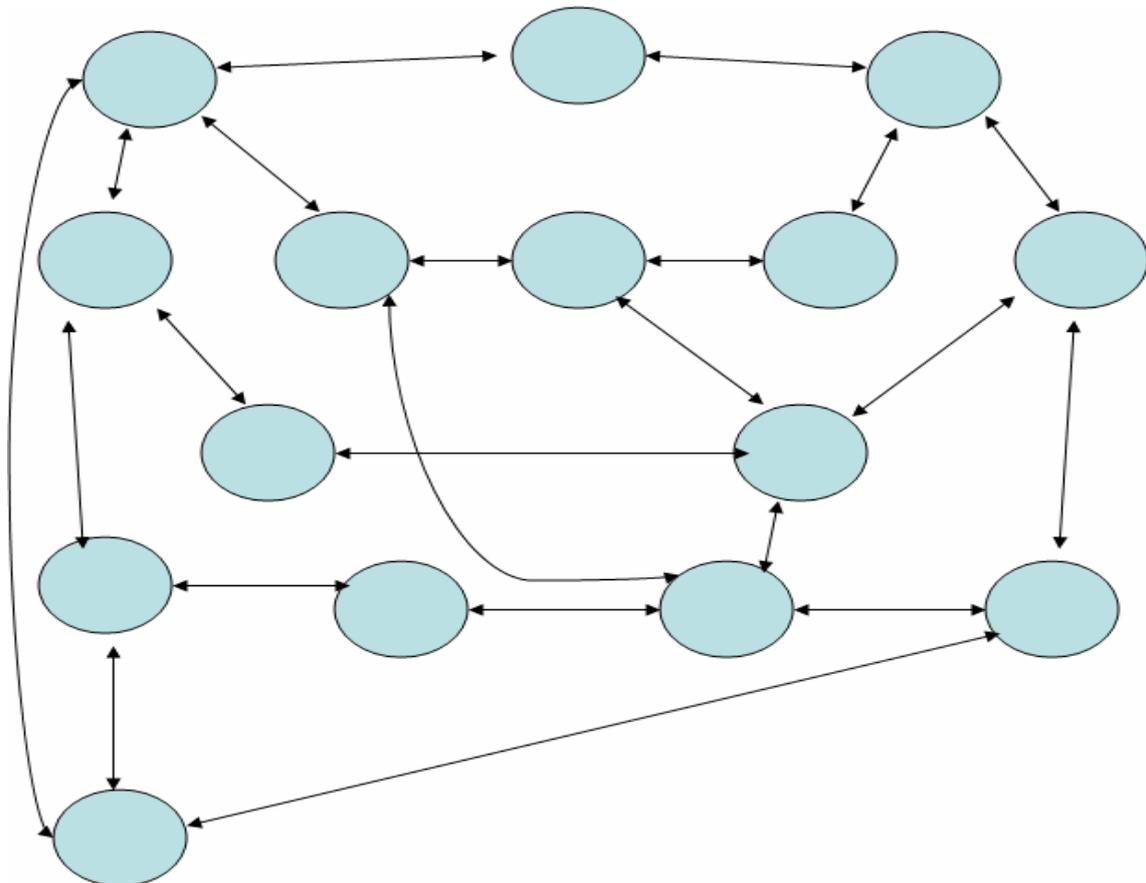
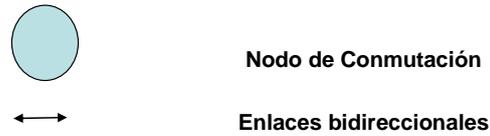


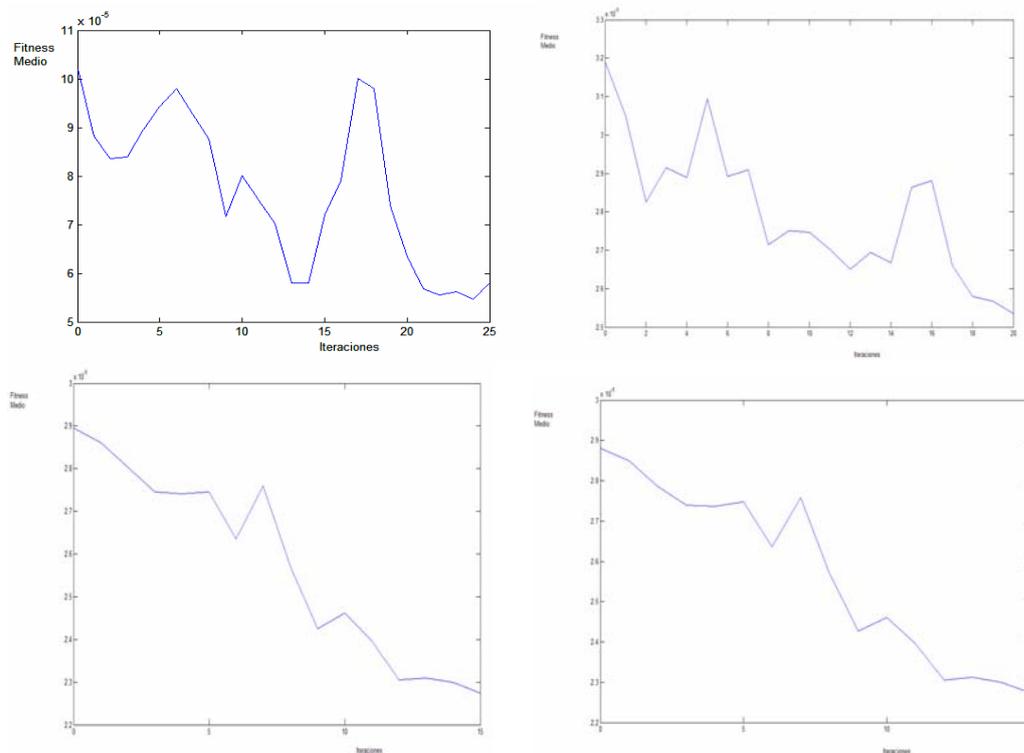
Figura 44. Topología de la red de 15 nodos y 46 enlaces bidireccionales.



**Figura 45.** Representación de la simbología utilizada para la realización de la red

### 11.3.1 Estudio de la pérdidas medias en función del número de iteraciones del algoritmo

En la Figura 46, formada por cuatro gráficas, se representa la media de las pérdidas respecto al número de generaciones del algoritmo genético para diez conexiones establecidas durante cada intervalo de conexión y distintos tamaños de búfer



**Figura 46.** Representación de las pérdidas medias respecto al número de iteraciones del algoritmo. (Superior izquierda) Caso de B=3. (Superior derecha) Caso de B=4. (Inferior izquierda) Caso de B=5. (Inferior derecha) Caso de B=6.

## Interpretación de resultados

En las tres figuras anteriores, cada una para un tamaño de búfer distinto, puede verse como a medida que aumenta el búfer las pérdidas medias de la población tienden a disminuir.

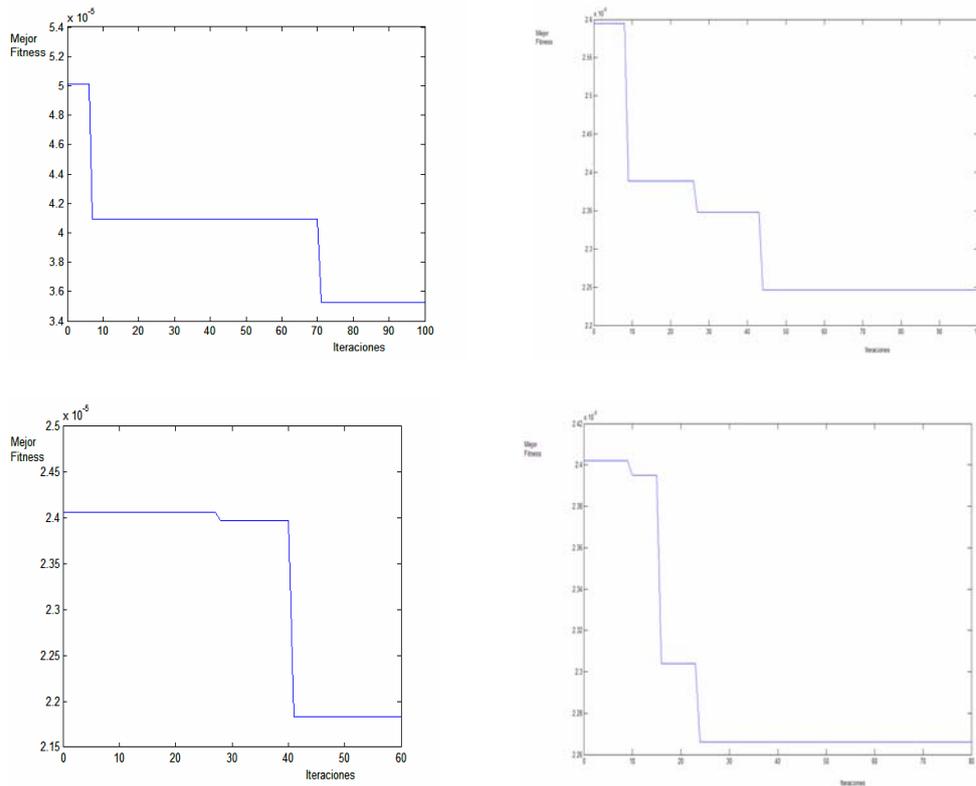
Se observa que la función no es monótonamente decreciente, sino que debido a la componente de aleatoriedad del algoritmo, algunas veces se crean individuos que generan puntualmente en una determinada población mayores pérdidas provocando picos en las gráficas. A pesar de ello, la tendencia de la gráfica es descendente, como cabía esperar, apreciándose la convergencia del algoritmo.

Obsérvese que las gráficas para un tamaño de búfer de cinco y de seis células son prácticamente iguales. Las pérdidas medias para la de un tamaño de búfer de seis células, son ligeramente inferiores a la de cinco células. Por esta razón, la elección para la red bajo estudio de un tamaño de búfer de seis células sería apropiada.

### **11.3.2 Estudio de la solución alcanzada en función del número de de iteraciones del algoritmo.**

**Tabla 47. Representación de los parámetros utilizados para la realización de las gráficas.**

<b>Parámetros / Tamaño del búfer</b>	<b>GoS( Grado de Servicio)</b>	<b>Mejor solución alcanzada</b>	<b>Tamaño de la población</b>	<b>Generaciones de mejora de la función objetivo</b>	<b>Número de Conexiones</b>
<b>B=3</b>	6	$3,53^{-5}$	10	7 y 71	10
<b>B=4</b>	6	$2,24^{-5}$	10	9, 27 y 44	10
<b>B=5</b>	6	$2,18^{-5}$	10	28 y 41	10
<b>B=6</b>	6	$2,17^{-5}$	10	10, 16, 24 y 75	10



**Figura 47. Representación de la solución alcanzada en función del número de iteraciones del algoritmo. (Superior izquierda) Caso B=3 y GOS=6. (Superior derecha) Caso B=3 y GOS=6. (Inferior izquierda) Caso B=5 y GOS=6. (Inferior derecha) Caso B=6 y GOS=6.**

### Interpretación de resultados

Al igual que para la red A, el algoritmo genético va tendiendo hacia individuos cada vez mejores.

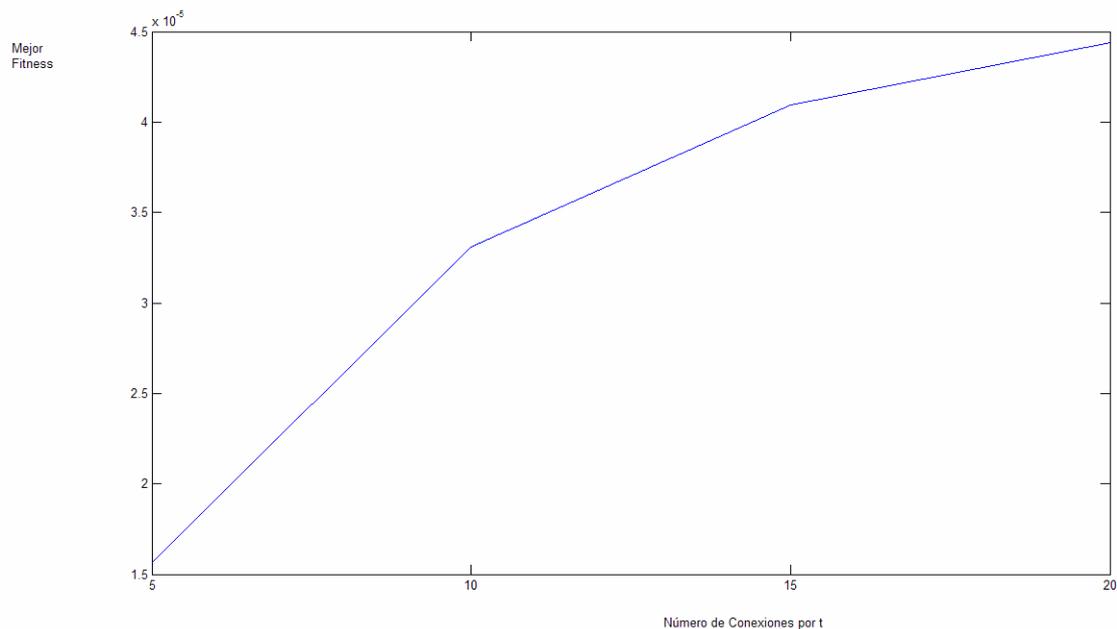
Al observar las cuatro gráficas de la Figura 47 y la Tabla 47, se comprueba que conforme aumenta el tamaño del búfer no sólo disminuyen las pérdidas, sino que se producen más cambios de pendiente. Para el caso de un tamaño de búfer de tres células se producen dos cambios de pendiente y para un tamaño de búfer de cinco células se producen cuatro.

### **11.3.3 Estudio de la solución alcanzada en función del tráfico**

Puede verse en la Tabla 48:

**Tabla 48. Representación de los parámetros para la realización de las gráficas**

<b>Parámetros</b>	<b>Tamaño Población</b>	<b>GoS (Grado de Servicio)</b>	<b>Tamaño del búfer</b>
<b>Caso</b>	10 Individuos	6	6



**Figura 48. Representación de la mejor solución alcanzada en función del número de conexiones establecidas durante cada intervalo de conexión.**

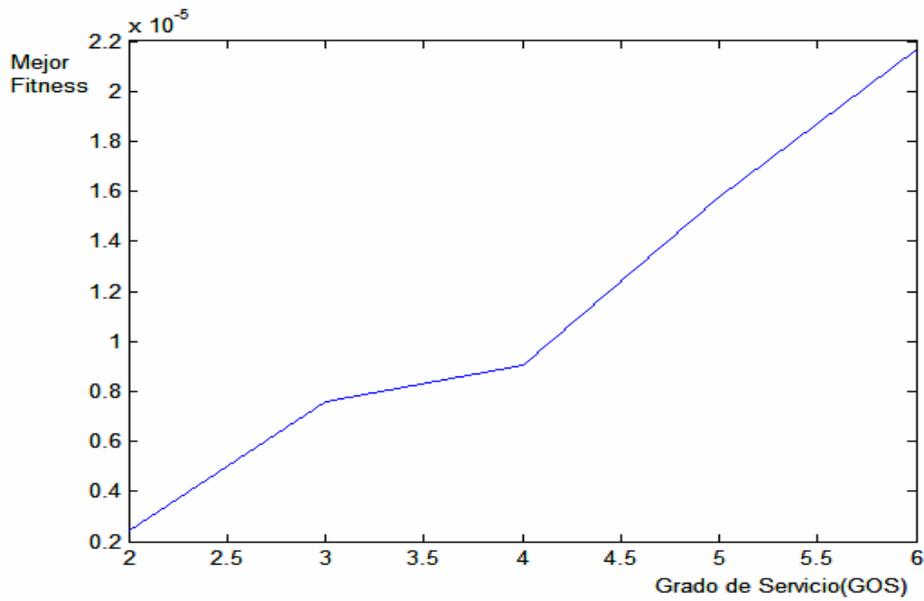
### Interpretación de resultados

Se representa y se hace el análisis para el caso de un tamaño de búfer de seis células, que es el adecuado.

Conforme aumenta el tráfico- numero de conexiones-, la solución alcanzada empeora debido a que se está sobrecargando la red.

### **11.3.4 Estudio de la solución alcanzada respecto al grado de servicio**

Puede verse en la Figura 49:



**Figura 49. Representación de la mejor solución respecto al grado de servicio. Caso de B=6 y 10 conexiones establecidas durante cada intervalo de conexión.**

### Interpretación de resultados

Conforme aumenta número de procesos de conmutación permitidos, la solución alcanzada empeora. Es decir, las pérdidas aumentan como puede verse en la Figura 49. Se observan dos fuertes cambios de pendiente para un grado de servicio de tres y de cuatro, tendiendo la gráfica a una relación lineal a partir de aquí.

## 12 ESFUERZO COMPUTACIONAL

### 12.1 ANÁLISIS DEL ESFUERZO COMPUTACIONAL PREVIO AL ALGORITMO GENÉTICO.

En este apartado se van a utilizar diversas redes de telecomunicación para la el estudio. A continuación, se muestra en la Tabla 49 las correspondientes topologías propuestas.

**Tabla 49. Tabla representativa de las diversas redes de telecomunicación y sus correspondientes parámetros.**

<b>Parámetros / Diversas redes</b>	<b>Nodos</b>	<b>Enlaces</b>	<b>Tamaño de la población</b>
<b>Red A</b>	10	30	10
<b>Red B</b>	11	34	10
<b>Red C</b>	12	38	10
<b>Red D</b>	13	46	10
<b>Red E</b>	14	48	10
<b>Red F</b>	15	54	10
<b>Red G</b>	16	58	10
<b>Red H</b>	17	64	10
<b>Red I</b>	18	50	10
<b>Red J</b>	19	58	10
<b>Red K</b>	20	62	10

Parámetros / Diversas redes	Nodos	Enlaces	Tamaño de la población
Red L	21	60	10
Red M	22	66	10
Red N	23	70	10
Red Ñ	24	76	10
Red O	25	84	10

### 12.1.1 Tiempo de obtención de todas los posibles caminos entre pares origen- destino

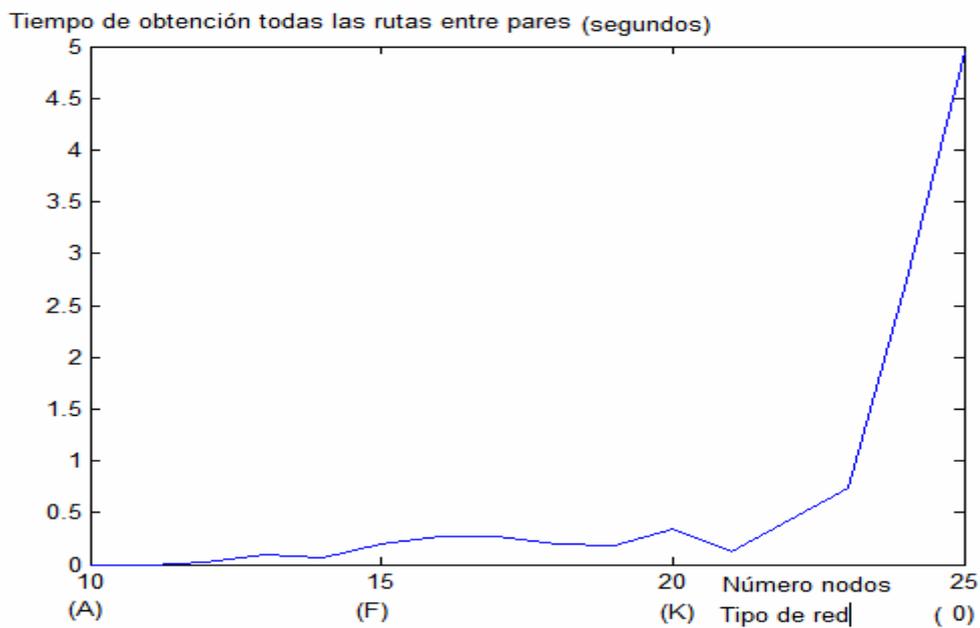


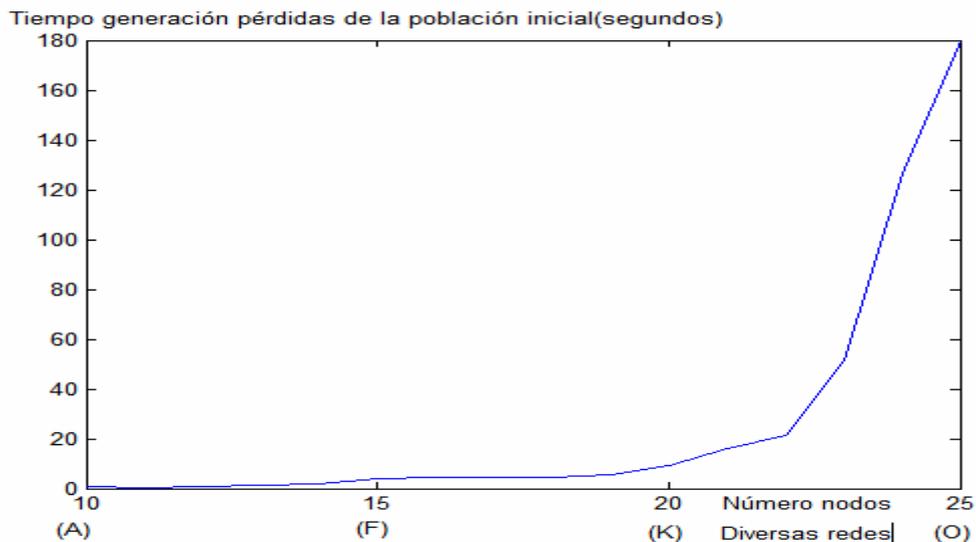
Figura 50. Representación del tiempo de cálculo de todas las rutas entre pares frente a diversas redes de telecomunicación

## Interpretación de resultados

En la Figura 50, se aprecia que el tiempo preciso para el cálculo de caminos aumenta progresivamente a medida que incrementa el número de nodos y arcos. Es razonable que, a medida que aumenta el número de nodos haya más caminos, y por tanto, ocupe más tiempo, pero también dependerá del número de enlaces que tenga la red en cuestión.

Hasta la red C el tiempo en segundos es prácticamente nulo. Hasta la red M nodos se mantiene por debajo de un segundo dando un salto bastante importante a partir de aquí.

### **12.1.2 Tiempo de generación de la población inicial de partida frente a diversas redes.**



**Figura 51. Representación del tiempo preciso para el cálculo de las pérdidas de la población inicial frente a diversas redes.**

## Interpretación de resultados

A medida que aumenta el número de nodos y el número de arcos, el tiempo de generación de las pérdidas de la población inicial aumenta.

## 12.2 ANÁLISIS DEL ESFUERZO COMPUTACIONAL DEL ALGORITMO GENÉTICO

Si se comparan los tiempos de CPU utilizados por el algoritmo para la resolución de los distintos problemas, se observa como el algoritmo tarda lo mismo en resolver cualquier problema para una topología de red dada, independientemente de los datos de tráfico con los que trabaje. No obstante, a medida que aumenta el número de nodos de la red el tiempo de CPU aumenta. A continuación, en la Tabla 50, se representan los tiempos medios de CPU utilizados por el algoritmo en función de la topología del problema.

**Tabla 50.** Tabla representativa de diversas redes de telecomunicación y sus correspondientes tiempos de ejecución

Tiempos de ejecución / Redes		Intervalo de variación de tiempo ejecución de CPU(seg)	Tiempo calcular caminos(seg)	Tiempo generar población inicial(seg)	Tamaño de la población
<b>10 nodos</b>	<b>30 arcos</b>	(5,80)	(0,0.3)	(0.531,5)	10
<b>21 nodos</b>	<b>60 arcos</b>	(200,2500)	(0.4,3)	(5,180)	15
<b>30 nodos</b>	<b>82 arcos</b>	(700,3000)	(3,4)	(180,224)	15
<b>40 nodos</b>	<b>104 arcos</b>	(1600,3700)	(5,6)	(400,600)	15

# 13 ANÁLISIS DE RESULTADOS PARA LA BATERÍA DE PROBLEMAS

## 13.1 CONSTRUCCIÓN DE UNA BATERÍA DE REDES DE ENSAYO

El paquete informático encargado de la resolución del problema necesita recibir de un fichero de tipo texto correspondiente a los datos de trabajo. Estos datos son, principalmente, el número de nodos y arcos de la red y la demanda de los pares. También se incluye una descripción de la red en la que se asigna un número a cada uno de los arcos y los nodos. El formato definido deberá indicar:

1. El número de nodos de la red que se va a considerar.
2. El número de arcos de la misma.
3. El número de intervalos de tiempo de conexión que se van a considerar en el problema.
4. Para cada uno de los arcos:
  - Número de nodo origen
  - Número de nodo destino
5. Para cada uno de los pares
  - La demanda en cada uno de los instantes de conexión considerados.

Para cada uno de los ficheros de entrada, el programa utilizado usa como lenguaje de programación C.

El número de intervalos de conexión- $t$ - utilizado para toda la batería de problemas es cuatro, y el número de instantes de conexión- $\tau$ - es diez. En éste proyecto, se trabajará con dos niveles, o escalas jerárquicas. El nivel de conexión- $t$ - y el de célula- $\tau$ .

Trabajar con el nivel de célula resulta indispensable si se tiene en cuenta que el análisis de pérdidas en los conmutadores debe hacerse célula a célula.

La generación del número de enlaces, número de nodos y número de conexiones establecidas durante cada intervalo de tiempo-t- que tendrán las diversas topologías de red, serán aleatorias.

La generación aleatoria del número de nodos para la batería de problemas, tiene como restricción el ser mayor o igual a diez, y menor o igual a cuarenta.

La generación aleatoria del número de conexiones establecidas durante cada intervalo oscilará entre un mínimo de diez conexiones y como máximo el combinatorio definido por:

$$\binom{\text{NumeroNodos}}{2} = \text{NumeroNodos!} / 2! * (\text{NumeroNodos} - 2)! \quad \text{Ecuación 27}$$

### 13.2 SELECCIÓN DE PARÁMETROS PARA EL ALGORITMO GENÉTICO

Se va a proceder a estudiar una propuesta de parámetros para cada uno de los siguientes casos:

- Caso 1: Red 10-17 nodos y 25-50 arcos
- Caso 2: Red 18-25 nodos y 50-70 arcos
- Caso 3: Red 25-35 nodos y 70-90 arcos
- Caso 4: Red 35-40 nodos y 90-110 arcos

Para cada uno de los cuatro casos descritos, se verá cómo responden dichas topologías ante cambios en una serie de parámetros. Los parámetros de red que se modificarán son: número de nodos, número de enlaces y número de conexiones establecidas durante cada intervalo. Modificación de tráfico no sólo se refiere al aumento del número de conexiones, sino que también se probarán las diversas topologías ante tres valores de demanda: baja, media y alta.

Los parámetros de diseño del algoritmo son: probabilidad de reemplazo, probabilidad de cruce y tamaño de búfer.

Para todos los casos, la probabilidad de reemplazo 'q' se ha hecho variar siempre entre 0.4 y 0.8. La probabilidad de reemplazo está relacionada con el operador

cruce, cuyo criterio de reemplazo de individuos es distinto al de mutación y está basado en la aptitud de los individuos. La probabilidad de reemplazo de los individuos se calcula mediante una ley hipergeométrica en función del fitness. Se podría haber tomado proporcional, pero con esto se consigue que dos individuos próximos en aptitud tengan probabilidades de reemplazo diferentes. Aunque se ha estudiado para una probabilidad de reemplazo entorno a 0.4 y 0.8, un buen criterio de diseño para los diversos casos propuestos están entorno a 0.4 y 0.6, aunque dependerá del caso de estudio.

El algoritmo elige en cada iteración el método de generación del nuevo individuo y lo hace de forma aleatoria. Si la probabilidad de cruce es alta, el algoritmo tiende a explorar a fondo las soluciones del espacio de búsqueda. Por el contrario, el operador mutación tiende a explorar nuevas regiones del espacio de soluciones admisibles.

Habitualmente se utiliza una probabilidad de cruce mayor que la de mutación. Debido a esto, como límite inferior la probabilidad de cruce será de 0.6, y como límite superior 0.9.

Para el criterio del tamaño del búfer, éste varía en función del caso que se estudia. Cada conmutador ATM debe tener una cola de tamaño suficiente para que el número de pérdidas no supere como norma a los valores estándares permitidos.

Sin embargo, si las colas son excesivamente grandes, aunque el número de células perdidas se reduzca enormemente ocurrirá que el retraso acumulado por las unidades de información será cada vez mayor. Esto puede provocar un funcionamiento de pésima calidad (basta pensar en lo que ocurriría en una conversación telefónica si se perdiese un mínimo sonido y compararlo con lo que ocurriría si llegasen con retardos variables).

A continuación, se muestra una interpretación para los cuatro casos:

*Propuesta general para los cuatro casos.* A medida que aumenta el número de conexiones abiertas, la red necesita un tamaño de búfer mayor de células, es decir, mayor.

Se aplica la misma tendencia a los distintos valores de demanda. Para demanda alta se necesita un tamaño de búfer mayor que para demanda baja. Los valores de demanda media son del orden de cinco veces superior a los demanda baja, y los de demanda alta diez veces superior a los de demanda baja. Los valores pueden verse en la Tabla 51.

**Tabla 51. Tabla representativa del orden de los distintos valores de demanda**

<b>Valor Demanda / Tipo de demanda</b>	<b>Valor de la demanda en células/ranura (Órdenes de magnitud)</b>
<b>Demanda baja</b>	0.0145 células/ranura

<b>Valor Demanda / Tipo de demanda</b>	<b>Valor de la demanda en células/ranura (Órdenes de magnitud)</b>
<b>Demanda media</b>	0.0252 células/ranura
<b>Demanda alta</b>	0.056 células/ranura

*Propuesta particular para caso 1.* La elección del búfer apropiado varía entre tres y nueve células, dependiendo del número de conexiones establecidas durante cada intervalo de conexión  $t$ . Hasta un rango de conexiones simultáneas establecidas de diez, la elección de un tamaño de búfer de tres o cuatro células sería buena. Para mayor número de conexiones, se necesita un búfer mayor.

Desde diez conexiones simultáneas hasta cuarenta, el tamaño del búfer aumenta en una célula más, partiendo de cuatro y acabando en nueve.

Desde valores de demanda hasta valores de demanda alta, el tamaño del búfer aumenta también en una célula más, partiendo de seis y acabando en ocho.

En conclusión, para una demanda considerable, la elección de un tamaño de búfer de siete u ocho células sería apropiada.

Puede verse en la Tabla 52.

**Tabla 52. Tabla de diseño de distintos parámetros para diversas topologías de red. Caso 1**

<b>Parámetros / Diversas redes</b>		<b>Probabilidad de reemplazo</b>	<b>Probabilidad de Cruce</b>	<b>Tamaño del Búfer</b>
<b>10-17nodos</b>	<b>30-50 arcos</b>	<b><math>q \in (0.4,0.8)</math></b>	<b><math>P_{\text{cruce}} \in (0.6,0.85)</math></b>	<b><math>B \in (3,9)</math></b>
<b>Conexiones abiertas</b>	<b>10</b>	0.6	0.6	4
<b>Conexiones abiertas</b>	<b>20</b>	0.4	0.6	6
<b>Conexiones abiertas</b>	<b>30</b>	0.7	0.6	7
<b>Conexiones</b>	<b>40</b>	0.4	0.6	9

Parámetros / Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
abiertas				
<b>Demanda</b>	<b>baja</b>	0.6	0.6	6
<b>Demanda</b>	<b>media</b>	0.6	0.7	7
<b>Demanda</b>	<b>alta</b>	0.7	0.85	8

*Propuesta particular para caso 2.* A diferencia del caso anterior, la elección apropiada del tamaño del búfer varía entre siete y quince células, dependiendo del número de conexiones establecidas durante cada intervalo de conexión  $t$ . Hasta un rango de conexiones simultáneas establecidas de diez, la elección de un tamaño de búfer de siete células sería buena. Para mayor número de conexiones, se necesita un búfer mayor.

Desde diez conexiones simultáneas hasta cincuenta, el tamaño del búfer aumenta en una célula más, partiendo de siete y acabando en quince.

Desde valores de demanda hasta valores de demanda alta, el tamaño del búfer aumenta también en una célula más, partiendo de ocho y acabando en doce.

Para el caso de cincuenta conexiones simultáneas durante cada intervalo de conexión, la solución alcanzada mejora hasta un tamaño de búfer de quince células, e incluso dieciséis. Aunque el número de células perdidas se reduzca enormemente ocurrirá que el retraso acumulado por las unidades de información será cada vez mayor. Realmente una buena elección para el tamaño del búfer será de diez u once células.

Puede verse en Tabla 53.

**Tabla 53. Tabla de diseño de distintos parámetros para diversas topologías de red. Caso 2**

Parámetros / Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
<b>18-25 nodos</b>	<b>50-70 arcos</b>	<b><math>q \in (0.4,0.8)</math></b>	<b><math>P_{cruce} \in (0.6,0.85)</math></b>	<b><math>B \in (7,15)</math></b>
<b>Conexiones abiertas</b>	<b>10</b>	0.6	0.7	7
<b>Conexiones</b>	<b>20</b>	0.6	0.6	8

Parámetros Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
<b>abiertas</b>				
<b>Conexiones abiertas</b>	<b>30</b>	0.4	0.7	10
<b>Conexiones abiertas</b>	<b>40</b>	0.4	0.7	10-11
<b>Conexiones abiertas</b>	<b>50</b>	0.6	0.7	10-15
<b>Demanda</b>	<b>baja</b>	0.6	0.7	7
<b>Demanda</b>	<b>media</b>	0.6	0.7	8
<b>Demanda</b>	<b>alta</b>	0.4	0.6	10-12

*Propuesta particular para caso 3.* La elección apropiada del tamaño del búfer varía entre ocho y quince células, dependiendo del número de conexiones establecidas durante cada intervalo de conexión  $t$ . Hasta un rango de conexiones simultáneas establecidas de diez, la elección de un tamaño de búfer de ocho sería buena. Para mayor número de conexiones, se necesita un búfer mayor.

Desde diez conexiones simultáneas hasta sesenta, el tamaño del búfer aumenta en una célula más, partiendo de ocho y acabando en quince.

Desde valores de demanda hasta valores de demanda alta, el tamaño del búfer aumenta también en una célula más, partiendo de ocho y acabando en doce.

Para el caso de cincuenta conexiones simultáneas durante cada intervalo de conexión, la solución alcanzada mejora hasta un tamaño de búfer de quince células, e incluso dieciséis. Aunque el número de células pérdidas se reduzca enormemente ocurrirá que el retraso acumulado por las unidades de información será cada vez mayor. Realmente una buena elección para el tamaño del búfer será de diez u once células.

Puede verse en la Tabla 54.

**Tabla 54. Tabla de diseño de distintos parámetros para distintas topologías de red. Caso 3**

Parámetros Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
<b>25-35</b>	<b>70-90</b>	<b><math>q \in (0.4, 0.8)</math></b>	<b><math>P_{cruce}</math></b>	<b><math>B \in (8, 15)</math></b>

Parámetros Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
<b>nodos</b>	<b>arcos</b>		∈ (0.6,0.85)	
<b>Conexiones abiertas</b>	<b>10</b>	0.4	0.85	8
<b>Conexiones abiertas</b>	<b>20</b>	0.4	0.7	9
<b>Conexiones abiertas</b>	<b>30</b>	0.6	0.6	10
<b>Conexiones abiertas</b>	<b>40</b>	0.6	0.85	11
<b>Conexiones abiertas</b>	<b>50</b>	0.6	0.7	10-13
<b>Conexiones abiertas</b>	<b>60</b>	0.4	0.7	10-15
<b>Demanda</b>	<b>baja</b>	0.4	0.85	8
<b>Demanda</b>	<b>media</b>	0.4	0.7	9-10
<b>Demanda</b>	<b>alta</b>	0.4	0.6	9-12

*Propuesta particular para caso 4.* La elección apropiada del tamaño del búfer varía entre ocho y quince células, dependiendo del número de conexiones establecidas durante cada intervalo de conexión  $t$ . Hasta un rango de conexiones simultáneas establecidas de diez, la elección de un tamaño de búfer de ocho u nueve sería buena. Para mayor número de conexiones, se necesita un búfer mayor.

Desde diez conexiones simultáneas hasta cincuenta, el tamaño del búfer aumenta en una célula más, partiendo de cuatro y acabando en nueve.

Desde valores de demanda hasta valores de demanda alta, el tamaño del búfer aumenta también en una célula más, partiendo de ocho y acabando en dieciséis.

Para el caso de cincuenta conexiones simultáneas durante cada intervalo de conexión, la solución óptima mejora hasta un tamaño de búfer de quince procesos, e incluso dieciséis. Aunque el número de células pérdidas se reduzca enormemente

ocurrirá que el retraso acumulado por las unidades de información será cada vez mayor. Realmente una buena elección de tamaño de búfer será de diez u once células.

Puede verse en Tabla 55.

**Tabla 55. Tabla de diseño de distintos parámetros para diversas topologías de red. Caso 4**

Parámetros Diversas redes		Probabilidad de reemplazo	Probabilidad de Cruce	Tamaño del Búfer
<b>35-40 nodos</b>	<b>90-110 arcos</b>	<b><math>q \in (0.4,0.8)</math></b>	<b><math>P_{cruce} \in (0.6,0.85)</math></b>	<b><math>B \in (8,16)</math></b>
<b>Conexiones abiertas</b>	<b>10</b>	0.7	0.7	8-9
<b>Conexiones abiertas</b>	<b>20</b>	0.6	0.6	9-11
<b>Conexiones abiertas</b>	<b>30</b>	0.4	0.7	9-12
<b>Conexiones abiertas</b>	<b>40</b>	0.7	0.7	9-12
<b>Conexiones abiertas</b>	<b>50</b>	0.6	0.7	9-16
<b>Demanda</b>	<b>baja</b>	0.7	0.7	8-9
<b>Demanda</b>	<b>media</b>	0.4	0.7	9-14
<b>Demanda</b>	<b>alta</b>	0.6	0.85	9-16

Para cada caso, se muestra el esfuerzo computacional, tanto previo a la ejecución del algoritmo como el tiempo que ocupa a la CPU la ejecución del mismo. Los parámetros propuestos en la tabla son:

*Tcaminos*. Tiempo de obtención de todas las rutas posibles entre pares que forman la red. Se muestran los intervalos de variación.

*TpoblaciónInicial*. Tiempo de cálculo de las pérdidas de la población inicial de partida del algoritmo genético. Se muestran los intervalos de variación.

*TejecuciónAG*. Tiempo empleado por la CPU en la ejecución del algoritmo genético. Se muestran los intervalos de variación.

Se puede observar que, conforme aumenta el número de nodos y arcos, aumenta el esfuerzo computacional. Se puede ver en la Tabla 56.

Por último, se indica para cada caso, el tamaño de la población utilizado, siendo de diez el primero y de quince individuos para el resto.

**Tabla 56. Tabla representativa de tiempos de ejecución para los cuatro casos propuestos**

<b>Tiempos de ejecución / Diversos casos</b>	<b>Intervalo Tcaminos (segundos)</b>	<b>Intervalo TpoblaciónInicial (segundos)</b>	<b>Intervalo TejecuciónAG (segundos)</b>
<b>Caso1</b>	€(0,0.3)	€(0.531,5)	€(5,80)
<b>Caso2</b>	€(0.4,3)	€(5,180)	€(200,2500)
<b>Caso3</b>	€(3,4)	€(180,224)	€(2500,3000)
<b>Caso4</b>	€(5,6)	€(400,600)	€(3000,3700)

### 13.3 COMPARACIÓN ALGORITMO GENÉTICO Y RUTA MÍNIMA

#### 13.3.1 Resultados

En la Tabla 57, se representan las diversas topologías de red utilizadas para la realización de las gráficas posteriores

**Tabla 57. Problemas de la batería de problemas**

<b>Parámetros / Diversas redes</b>	<b>Nodos</b>	<b>Enlaces</b>	<b>Tamaño de la población</b>	<b>Número de conexiones establecidas</b>
<b>ENDIN1</b>	10	30	10	10
<b>ENDIN2</b>	11	34	10	10
<b>ENDIN3</b>	12	38	10	10
<b>ENDIN4</b>	13	46	10	10
<b>ENDIN5</b>	14	48	10	10
<b>ENDIN6</b>	15	54	10	10
<b>ENDIN7</b>	16	58	10	10
<b>ENDIN8</b>	17	64	10	10
<b>ENDIN9</b>	18	50	10	10
<b>ENDIN10</b>	19	58	10	10
<b>ENDIN11</b>	20	62	10	10
<b>ENDIN12</b>	21	60	10	10
<b>ENDIN13</b>	22	66	10	10
<b>ENDIN14</b>	23	70	10	10
<b>ENDIN15</b>	24	76	10	10
<b>ENDIN16</b>	25	84	10	10
<b>ENDIN17</b>	26	82	10	10
<b>ENDIN18</b>	27	84	10	10

<b>Parámetros / Diversas redes</b>	<b>Nodos</b>	<b>Enlaces</b>	<b>Tamaño de la población</b>	<b>Número de conexiones establecidas</b>
<b>ENDIN19</b>	28	86	10	10
<b>ENDIN20</b>	29	72	10	10
<b>ENDIN21</b>	30	82	10	10
<b>ENDIN22</b>	31	80	10	10
<b>ENDIN23</b>	32	86	10	10
<b>ENDIN24</b>	33	88	10	10
<b>ENDIN25</b>	34	90	10	10
<b>ENDIN26</b>	35	94	10	10
<b>ENDIN27</b>	36	98	10	10
<b>ENDIN28</b>	37	90	10	10
<b>ENDIN29</b>	38	90	10	10
<b>ENDIN30</b>	39	96	10	10
<b>ENDIN31</b>	40	100	10	10

### 13.3.1.1 Resultados en tablas.

Puede verse en la Tabla 58:

**Tabla 58. Comparativa del algoritmo genético con el algoritmo de selección de ruta mínima**

	Ruta Mín	G1	G2	G3	G4	G5	G mejor	TRM(seg)	% Mejora	TAG(seg)
ENDIN1	4,42E-06	4,4E-06	3,62E-06	4,18E-06	3,85E-06	3,82E-06	3,62E-06	0,531	13,22	1,359
ENDIN2	5,26E-06	4,75E-06	5,26E-06	5,15E-06	5,07E-06	4,84E-06	4,75E-06	1,844	10,73	7,031
ENDIN3	6,63E-06	6,46E-06	6,63E-06	5,49E-06	5,07E-06	6,55E-06	5,49E-06	2,578	17,19	6,953
ENDIN4	2,36E-06	2,08E-06	2E-06	2,09E-06	5,07E-06	2,36E-06	2,00E-06	2,781	15,25	19,858
ENDIN5	6,64E-06	6,34E-06	5,15E-06	5,17E-06	5,07E-06	6,42E-06	5,15E-06	2,781	22,43	23,431
ENDIN6	5,81E-06	5,8E-06	5,8E-06	5,81E-06	5,07E-06	5,70E-06	5,70E-06	3,813	1,89	25,016
ENDIN7	4,89E-06	4,65E-06	4,43E-06	4,77E-06	5,07E-06	4,89E-06	4,51E-06	4,813	7,77	35,453
ENDIN8	4,24E-06	3,73E-06	4,11E-06	4,24E-06	5,07E-06	4,20E-06	3,73E-06	5,406	12,02	50,7843
ENDIN9	5,68E-06	4,89E-06	5,45E-06	5,59E-06	5,07E-06	5,40E-06	4,89E-06	4,187	13,9	70,423
ENDIN10	5,76E-06	5,72E-06	5,76E-06	5,64E-06	5,07E-06	5,75E-06	5,64E-06	5,078	1,9	200,31
ENDIN11	5,9E-06	5,9E-06	5,82E-06	5,25E-06	5,07E-06	5,90E-06	5,25E-06	9,265	11,01	300,434
ENDIN12	7,46E-06	7,11E-06	6,84E-06	7,46E-06	5,07E-06	7,36E-06	6,84E-06	15,5	8,31	580
ENDIN13	7,97E-06	7,55E-06	7,77E-06	7,38E-06	5,07E-06	7,97E-06	7,55E-06	11,031	5,26	790
ENDIN14	7,77E-06	7E-06	7,77E-06	7,7E-06	5,07E-06	7,35E-06	7,00E-06	22,188	9,09	821
ENDIN15	6,63E-06	5,75E-06	6,55E-06	6,59E-06	5,07E-06	6,00E-06	5,75E-06	63,391	13,27	1038,34
ENDIN16	6,74E-06	6,74E-06	6,56E-06	6,15E-06	5,07E-06	6,15E-06	6,15E-06	150,516	8,75	2313,34
ENDIN17	6,63E-06	6,63E-06	6,61E-06	6,12E-06	6,62E-06	5,98E-06	5,98E-06	27,017	9,8	2364,213
ENDIN18	1,27E-05	7,91E-05	7,95E-05	8,01E-05	8,02E-05	7,70E-05	9,74E-06	83,453	3,99	2595,1
ENDIN19	8,51E-06	8,51E-06	8,2E-06	8,34E-06	8,1E-06	8,32E-06	8,10E-06	173,156	4,81	2783,43
ENDIN20	1,27E-05	1E-06	8,9E-06	9,98E-06	9,12E-06	1,13E-06	9,92E-06	137,875	21,88	2574,45
ENDIN21	1,29E-06	1,15E-06	1,22E-06	1,1E-06	1,28E-06	1,23E-05	1,10E-06	4,797	14,7	2557,32
ENDIN22	1,25E-05	1,25E-06	1,24E-06	9,96E-06	1,06E-06	9,92E-06	9,92E-06	19,748	20,64	2600
ENDIN23	1,01E-05	1E-05	1,01E-05	9,99E-06	8,54E-06	9,26E-06	8,54E-06	8,656	15,44	2650,12
ENDIN24	1,07E-05	1,01E-05	1,03E-05	1,07E-06	9,93E-06	1,00E-05	9,93E-06	134,484	7,19	2671,432
ENDIN25	1,3E-05	1,21E-05	1,29E-05	1,3E-06	9,91E-06	1,14E-06	9,91E-06	191,719	24,3	2784,6
ENDIN26	8,22E-06	8,22E-06	8,18E-06	7,6E-06	7,76E-06	7,90E-06	7,60E-06	263,109	7,54	2987,43
ENDIN27	7,91E-06	6,63E-06	6,61E-06	6,12E-06	5,89E-06	5,98E-06	6,62E-06	430,953	16,3	3124,34
ENDIN28	8,68E-06	8,68E-06	8,65E-06	8,23E-06	8,12E-06	8,03E-06	8,03E-06	905,25	7,48	3345,213
ENDIN29	1,73E-05	1,71E-05	1,48E-06	1,73E-05	1,32E-05	1,43E-05	1,32E-05	10,969	24,11	3124,56
ENDIN30	2,36E-05	2,14E-05	1,84E-05	1,79E-05	2,29E-05	2,36E-06	1,79E-05	135,172	24,15	3576,89
ENDIN31	1,15E-05	1,15E-06	1,06E-06	9,68E-06	1,13E-06	9,87E-06	9,68E-06	252,281	15,82	3672,32

Anteriormente, se ha realizado la comparativa entre los dos algoritmos para diversas topologías de red y un mismo número de conexiones establecidas para cada intervalo, diez en particular.

También resulta interesante compara ambos algoritmos para cuatro topologías de red diferentes con diferente número de nodos y enlaces. Para cada una de ellas, se ramificarán en dos casos con diferente número de enlaces, y a su vez, en diez, veinte y treinta conexiones establecidas durante cada intervalo de conexión. Puede verse en la Tabla 59:

**Tabla 59. Comparativa del algoritmo genético con el algoritmo de selección de ruta mínima**

Redes	Número Conexiones	Algoritmo Ruta Min	Algoritmo Genético	%Mejora
10 nodos	10	4,17E-06	3,62E-06	13,18
30 enlaces	20	8,49E-06	8,05E-06	5,18
	30	1,08E-05	1,04E-05	3,7
10 nodos	10	1,3E-06	9,93E-07	23,61
44 enlaces	20	5,55E-06	5,34E-06	2,9
	30	5,4E-06	5,30E-06	1,85
12 nodos	10	6,63E-06	6,24E-06	5,88
38 enlaces	20	1,26E-05	1,14E-05	9,52
	30	1,57E-05	1,47E-05	6,36
12 nodos	10	2,95E-06	2,82E-06	4,6
52 enlaces	20	8,07E-06	8,02E-06	0,61
	30	1,04E-05	1,02E-05	1,92
14 nodos	10	8,71E-06	8,25E-06	5,28
34 enlaces	20	2,12E-05	1,96E-05	7,54
	30	2,56E-05	2,2E-05	14
14 nodos	10	6,64E-06	5,15E-06	22,43
48 enlaces	20	1,36E-05	1,22E-05	10,29
	30	1,61E-05	1,43E-05	11,18
17 nodos	10	7,17E-06	6,95E-06	3,06
48 enlaces	20	2,53E-05	2,47E-05	2,37
	30	2,61E-05	2,6E-05	0,38
17 nodos	10	4,24E-06	3,73E-06	12
64 enlaces	20	9,19E-06	9,1E-06	9,79
	30	1,3E-05	1,29E-05	0,76
20 nodos	10	1,63E-05	1,26E-05	22,69
54 arcos	20	1,61E-05	1,37E-05	14,9
	30	2,27E-05	1,96E-05	13,65
20 nodos	10	1,08E-05	1,03E-05	4,62
62 arcos	20	1,59E-05	1,23E-05	22,62
	30	2,21E-05	1,79E-05	19
25 nodos	10	9,26E-06	8,85E-06	4,42
72 arcos	20	2,4E-05	2,31E-05	3,75
	30	2,76E-05	2,65E-05	3,77
25 nodos	10	1,02E-05	1,01E-05	0,98
80 arcos	20	2,36E-05	2,2E-05	6,77
	30	2,79E-05	2,37E-05	15,05
30 nodos	10	1,81E-05	1,43E-05	20,99
74 arcos	20	2,7E-05	2,22E-05	18,51
	30	3,88E-05	3,28E-05	15,46
30 nodos	10	1,34E-05	9,88E-06	26,26
82 arcos	20	2,41E-05	2,14E-05	11,19
	30	9,39E-05	8,67E-05	7,66
40 nodos	10	1,55E-05	1,12E-05	27,74
93 arcos	20	4,84E-05	4,35E-05	10,12
	30	6,35E-05	6,21E-05	2,2
40 nodos	10	1,4E-05	9,98E-06	28,7
97 arcos	20	2,34E-05	2,03E-05	13,24
	30	5,66E-05	5,24E-05	7,42

### 13.3.1.2 Resultados en gráficas

#### 13.3.1.2.1 Mejor solución encontrada frente a diversas redes

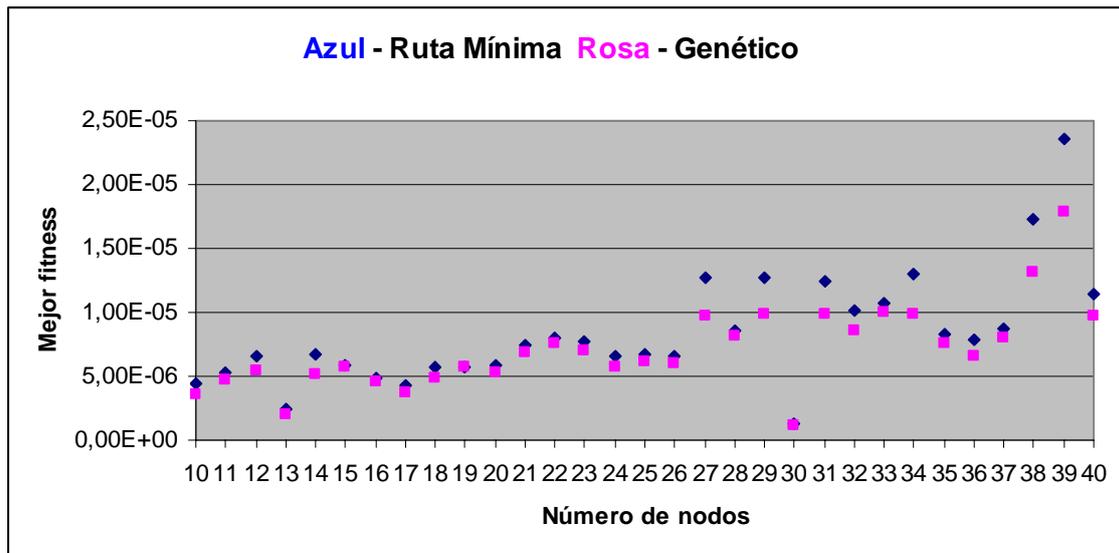


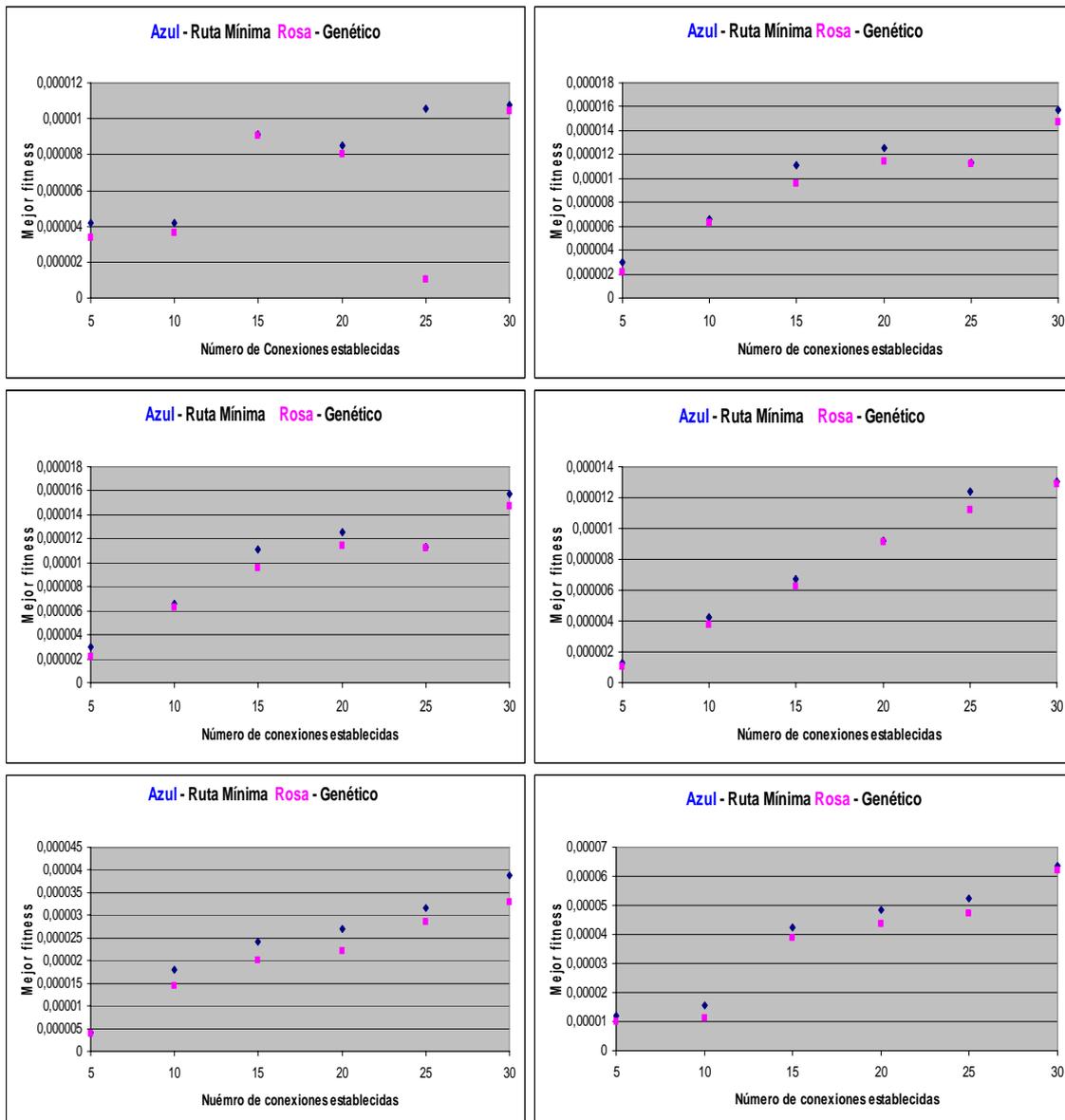
Figura 52. Comparación del algoritmo de ruta mínima con el algoritmo genético. Desde 10 hasta 40 nodos.

#### Interpretación de resultados

Tanto en Figura 52, se puede observar que, el algoritmo genético siempre ofrece una mejor solución frente al algoritmo de ruta mínima frente a diversas topologías de red. La nube de puntos rosa siempre está por debajo de la nube de puntos azul. Se encuentra dividida en dos figuras independientes, para así visualizar mejor las nubes de puntos.

#### 13.3.1.2.2 Mejor solución encontrada frente al número de conexiones establecidas durante cada intervalo de conexión t. Comparación de ambos algoritmos.

Puede verse en la Figura 53:



**Figura 53.** Comparación del algoritmo genético frente al de selección de ruta mínima. (Superior izquierda) Caso de 10 nodos y 30 enlaces. (Superior derecha) Caso de 12 nodos y 38 enlaces. (Central izquierda) Caso de 14 nodos y 48 enlaces. (Central derecha) Caso de 17 nodos y 64 enlaces. (Inferior izquierda) Caso de 30 nodos y 74 enlaces. (Inferior derecha) Caso de 40 nodos y 93 enlaces.

### Interpretación de resultados

En la Figura 53, formada por cuatro gráficas diferentes, se representa la mejor solución encontrada frente al número de conexiones establecidas durante un intervalo de tiempo  $t$  para distintas topologías de red y para los dos algoritmos; el genético y el de selección de ruta mínima

Se aprecia como la nube de puntos rosa se encuentra debajo, o al menos igual, que la nube de puntos azul. A medida que aumenta el número de conexiones

establecidas durante cada intervalo de conexión, la solución encontrada es cada vez peor, como cabe esperar.

### 13.3.1.2.3 Mejor solución encontrada frente al número de enlaces. Comparación de ambos algoritmos.

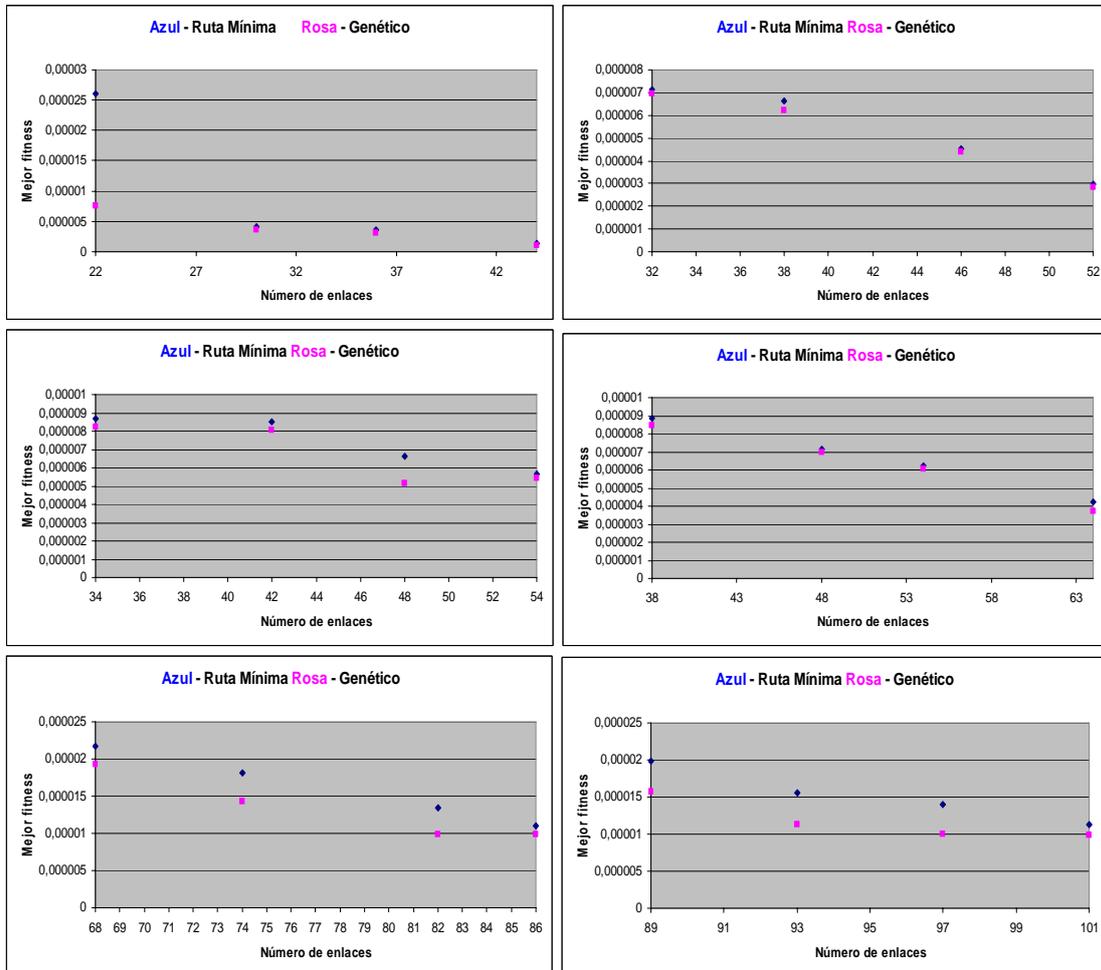


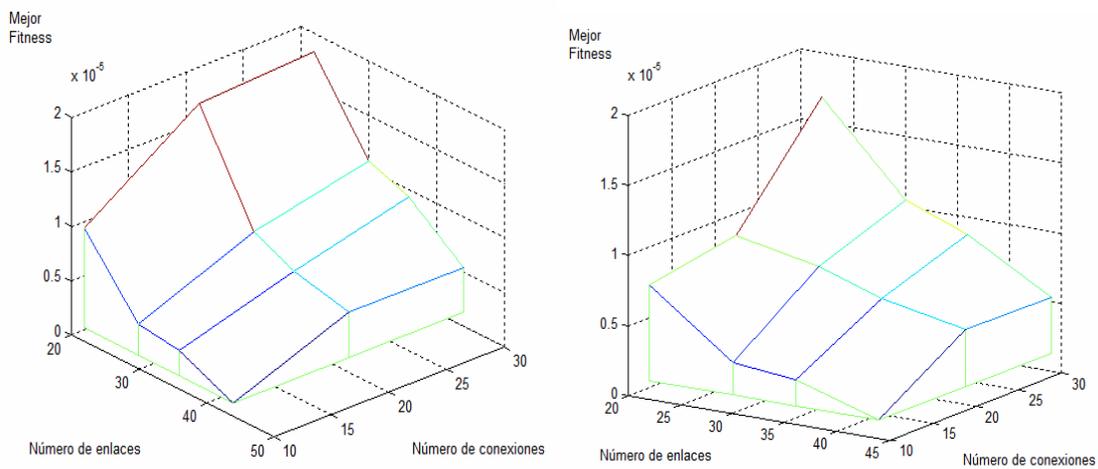
Figura 54. Comparación algoritmo genético con el de selección de ruta mínima. (Superior izquierda) Caso de 10 nodos. (Superior derecha) Caso de 12 nodos. (Central izquierda) Caso de 14 nodos. (Central derecha) Caso de 17 nodos. (Inferior izquierda) Caso de 30 nodos. (Inferior derecha) Caso de 40 nodos.

### Interpretación de resultados

En la Figura 54, formada por cuatro redes diferentes, se representa la mejor solución encontrada frente a la variación del número de enlaces.

Se aprecia como, a medida que aumenta el número de enlaces para una red dada, se encuentra una mejor solución para ambos algoritmos.

### 13.3.1.2.4 Mejor solución encontrada frente al número de enlaces y al número de conexiones establecidas durante cada intervalo de conexión t. Comparación de ambos algoritmos.



**Figura 55. Representación de la mejor solución encontrada frente al número de conexiones establecidas durante cada intervalo de tiempo t y frente al número de enlaces para una red de 10 nodos. (Izquierda) Algoritmo de selección de ruta mínima. (Derecha) Algoritmo genético.**

#### Interpretación de los resultados

En la Figura 55, se compara para los dos algoritmos la mejor solución encontrada frente al número de conexiones establecidas durante cada intervalo de tiempo t y frente al número de enlaces para una red de 10 nodos.

Se observa que, para la gráfica de la derecha – algoritmo de ruta mínima - , la solución encontrada es siempre ligeramente mejor, o al menos igual, que para el algoritmo de selección de ruta mínima.

A medida que aumenta el número de conexiones y disminuye el número de enlaces de la red, la solución que ofrecen ambos algoritmos es peor.



## 14 CONCLUSIONES

En este proyecto se ha propuesto un modelo matemático que permite la evaluación de la bondad de las rutas de una red de telecomunicaciones en cuanto a pérdidas se refiere. El modelo presentado, parte de una topología de red concreta y un nivel de demanda dado descrito en un fichero de entrada y mediante la aptitud de los individuos, se pretende establecer la calidad de las rutas desde el punto de vista de las pérdidas de células. Posteriormente, para el estudio de dichas pérdidas se han utilizado dos algoritmos: Un algoritmo genético y el algoritmo de selección de ruta mínima. En el algoritmo genético, para evaluar la bondad de una solución, se calcula la aptitud de los individuos. Así, de este modo, la propia evolución del método consigue establecer un criterio de comparación entre individuos y de esta forma elegir la mejor.

El modelo matemático, después de un extenso análisis sobre las características de las redes ATM, parte de la consideración de una serie de premisas básicas: Se utilizará tráfico CBR, que es el que presenta las restricciones más estrictas en cuanto a calidad, y es el tipo de tráfico elegido para el desarrollo que se realiza; Procesadores de conmutación con colas a la salida. Actualmente existe una gran gama de tipos de conmutadores. El utilizado para el modelo en cuestión es el conmutador con colas a la salida; Enlaces STM-1. Las redes consideradas están formadas por enlaces con capacidad de 155,52 Mb/s; Por último. Se han contemplado dos niveles de trabajo: el nivel de célula  $\tau$  - y el nivel de conexión  $t$ -.

Como se ha indicado a lo largo de este documento, en el análisis que se presenta se ha pretendido estudiar la posibilidad de encaminamiento de la información en las redes basadas en tecnología ATM, utilizando como icono para ello la pérdida de células producida por la red. Como se ha dicho anteriormente, En ATM existen dos parámetros fundamentales de referencia a la hora de determinar si la información recibida por el nodo destino cumple con el grado de servicio contratado con la red: el número de células perdidas por la red antes de llegar al destino y el retardo sufrido por la información que finalmente llega. En nuestro estudio, la pérdida de información en la red se considera igualmente importante, si no más, que el exceso de retardo. Sin embargo, no se ha menospreciado totalmente la consideración de limitaciones en el retardo de la información, ya que se ha procedido a buscar el encaminamiento óptimo minimizando la pérdida de células, a la vez que se han incorporado unas condiciones mínimas, a satisfacer, sobre el retraso en las comunicaciones de red.

Con todas estas hipótesis, se formula un modelo matemático de optimización para el encaminamiento dinámico que simula razonablemente fiel el comportamiento real de la red, y cuya resolución se puede acometer de una forma relativamente sencilla gracias a la codificación eficiente del cromosoma de los individuos del algoritmo genético, la cual permite una rápida y simple evaluación del fitness de los individuos. Dicho modelo ha sido implementado mediante lenguaje de programación C.

Con el método de evaluación que se expone a lo largo del proyecto es posible, partiendo de unos datos fundamentales como son la topología de red – nodos y enlaces – y valores de demanda para cada nivel de célula  $\tau$ - seleccionar el encaminamiento a través de las mejores rutas de la red y desechando, por el contrario, las peores

soluciones. Esto constituye una gestión de gran importancia en las tareas de operación y en tiempo real en de telecomunicaciones.

Tras un destacable número de ensayos con diversas topologías de red, los resultados pertinentes presentados demuestran que para un tipo de demanda uniforme entre los pares de la red, los factores fundamentales que determinan la obtención de una mayor o menor cantidad de pérdidas por la red es el control del número de nodos en la misma y la cantidad de conexiones establecidas simultáneamente en un intervalo de tiempo – t-. Se llega a una conclusión similar a la vista de los resultados relacionados con el análisis del esfuerzo computacional, tanto previos al algoritmo como del algoritmo genético. Como cabe esperar, el aumento del número nodos y número de enlaces – condiciones topológicas – aumenta el esfuerzo computacional de ambos algoritmos. Para el caso del esfuerzo computacional previo al algoritmo, los resultados nos indican que para redes por debajo del umbral de veintidós, veintitrés nodos, el tiempo de obtención de todas las rutas entre pares de la red se mantienen por debajo de un segundo, incrementando linealmente a partir de ahí. Para el caso del esfuerzo computacional del algoritmo genético, los resultados pertinentes nos indican que hasta un umbral de veintidós, veintitrés nodos, el tiempo de cálculo de pérdidas de la población inicial se mantiene por debajo de los veintidós segundos aproximadamente, incrementando linealmente a partir de aquí, al igual que el caso anterior. Es decir, se puede concluir que, en ambos casos, coincide el umbral de veintidós, veintitrés nodos.

Aunque la solución encontrada mediante el algoritmo genético sea mejor a medida que aumenta el tamaño del búfer, se llega a la conclusión de que utilizar un tamaño de búfer de más de diez, once o doce células no mejora los resultados. A partir de diez u once células, la solución encontrada en las diversas topologías de red analizadas, no mejora mucho más y, por el contrario, sí el retraso acumulado en los búferes. Para redes menores de diez nodos, un tamaño de búfer de tres, cuatro, células sería el apropiado, siempre dependiendo de las condiciones topológicas y de tráfico. Desde redes de diez nodos hasta redes de diecisiete nodos aproximadamente, un tamaño de búfer de seis o siete células sería el apropiado. A partir de aquí, el compromiso pérdidas pequeñas – retraso acumulado en el búfer – hace apropiado búferes de diez, once o doce células, como dije anteriormente.

También se han analizado tras una gran cantidad de ensayos, los intervalos de variación de los distintos parámetros del algoritmo genético para la obtención de una menor cantidad de pérdidas por red. De esta manera, una buena elección del parámetro probabilidad de cruce se situará entre un valor de 0.7 y un valor de 0.85, aunque siempre dependiendo del caso bajo estudio. En la mayoría de los casos, una buena elección del parámetro probabilidad de reemplazo, suele ser 0.4, o bien, 0.6. El parámetro tamaño de la población - número de individuos - influye de manera crucial en el comportamiento del algoritmo genético. Cuando se aumenta el número de individuos, además de incrementar el número de generaciones de mejora de la función objetivo, se obtienen pérdidas menores en la red, e incluso el número de generaciones para la convergencia del algoritmo genético también crece.

Después un proceso de comparación entre el AG – algoritmo genético- y ARM –algoritmo de selección de ruta mínima- para una batería de problemas suficientemente grande, hasta redes de cuarenta nodos, se llega a la conclusión de la mejor solución presentada por el algoritmo genético ofreciendo pérdidas menores. Generalmente, el

tanto por ciento de mejora que presenta el genético frente al algoritmo de selección de ruta mínima suele estar entre un 5% y un 20%, aunque depende de los factores topológicos y de tráfico de cada red.

A diferencia del algoritmo genético, después de analizar los resultados pertinentes del algoritmo de selección de ruta mínima, el factor primordial que determina la obtención una cantidad mínima de pérdidas es el control del número de enlaces, a parte del número de nodos y el número de conexiones establecidas durante cada intervalo de tiempo  $t$ . Así, a medida que las redes tienen mayor número de enlaces, dicho algoritmo ofrece mejores soluciones.

Debido a la importancia que presentan para el transporte de información el cumplimiento de los grados de servicio establecidos, una posible línea de futuro sería hacer un análisis profundo sobre el comportamiento de los conmutadores, en cuanto a retraso de información se refiere, y completar el modelo aquí presentado con otras limitaciones más fuertes sobre el retraso. Limitaciones que estarían determinadas por los grados de servicios contratados, y que quedan como línea de trabajo abiertas.

Otra línea posible de estudio que se plantean para futuras investigaciones consiste en la inclusión de comportamientos relacionados con otros tipos de tráfico característicos de este tipo de red como VBR, ABR y UBR que no han sido incluidos en el análisis que aquí se presenta.

La disciplina de la cola utilizada para representar los conmutadores ATM es FIFO – primero en entrar, primero en salir –. Dicha suposición trae como consecuencia una simplificación del sistema que no es totalmente realista. En ATM existen células con diferentes prioridades que requerirán distinto trato a la hora de ser servidas, como por ejemplo, las de control y gestión de red. Una posible línea de futuro sería tratar el sistema de colas mediante un análisis mucho más realista, es decir, tratar a las células con diferentes prioridades.



## 15 BIBLIOGRAFÍA

Amiri, A., Pirkul, H. (1999): "Routing and capacity assignement in backbone communication networks under time varying traffic conditions". European Journal of Operational Research. vol. 117, no. 1, pp. 15-29(15).

P. Cortés, J. Muñuzuri, J. Larrañeta y L. Onieva (2002). "A genetic algorithm based on cell loss for dynamic routing in ATM networks". In Soft Computing in Industry -Recent Applications. Springer-Verlag, Berlin, pp. 627-640.

Giroux, N., Ganti, S. (1999): "Quality of service in ATM networks", Prentice Hall.  
Händel, R., Huber, M.N. y Schöder, S. (1994): ATM Networks, concepts, protocols, applications, Addison-Wesley.

Lee, M-J., Yee J.R. (1994): "An algorithm for optimal minimax routing in ATM networks". Annals of Operations Research 49, 185-206.

Medova, E. (1998): "Chance-constrained stochastic programming for integrated services network management". The Judge Institute of Management Studie, University of Cambridge, England CB2 1AG.

Ministerio de Obras Públicas, Transportes y Medio Ambiente. Dirección General de Telecomunicaciones. (1995): Plan Nacional de Telecomunicaciones (1991-2002), Centro de Publicaciones del Ministerio.

Ministerio de Obras Públicas, Transportes y Medio Ambiente. Dirección General de Telecomunicaciones. (1994): Política de Telecomunicaciones durante el periodo (1994-1998), Centro de Publicaciones del Ministerio.

Ott, T.J., Krishnan K.R. (1992): "Separable routing: a scheme for state-dependent routing of circuit switche telephone traffic". Annals of Operations Research 35, 43-68.

Pitts, J.M., Schormans, J.A. (1996): "Introduction to ATM design and performance", Wiley.

Rahman, M.A. (1998): "Guide to ATM systems and technology", Artech House Publishers.

Redoli, J. (1997): “Redes metropolitanas de banda ancha. Presente y futuro”, Secretariado de publicaciones e intercambio científico. Universidad de Valladolid.

Stallings, W. (1995): “ISDN and broadband ISDN with frame relay and ATM”, Prentice Hall.

Sexton, M., Reid, A. (1997): “Broadband Networking. ATM, SDH and SONET”, Artech House Publishers.

Sexton, M., Reid, A. (1992): “Transmission Networking: SONET and the Synchronous Digital Hierarchy”, Artech House Publishers.