

## XVII

# APLICACIÓN DE LAS REDES NEURONALES ARTIFICIALES A LAS TELECOMUNICACIONES

A. Díaz Estrella; A. García Lopera; A. Jurado Díez; F. Sandoval  
*Universidad de Málaga*

## 1. INTRODUCCIÓN

Una de las primeras aplicaciones de las redes neuronales artificiales (RNAs) llevadas a cabo con éxito en los años 60 y profusamente utilizada hoy en día, fue en el campo de las telecomunicaciones. Nos referimos a los filtros transversales adaptativos usados en los ecualizadores adaptativos y canceladores de eco de las líneas telefónicas. Este filtro (una neurona) ajusta sus coeficientes (pesos sinápticos) automáticamente con el algoritmo de aprendizaje LMS (*Least Mean Square*) de Widrow y Hoff (la denominada regla delta en el campo de las RNAs). Sin embargo, esta potente herramienta no sirvió de incentivo en años posteriores para el desarrollo de otro tipo de aplicaciones. Ha sido a finales de los 80, cuando el gran interés despertado por las RNAs en conjunción con la aparición de nuevos retos en el campo de las telecomunicaciones, de difícil solución mediante técnicas convencionales, ha puesto de nuevo sobre la mesa las capacidades de las RNAs. Problemas de optimización de recursos como enrutamientos de mensajes a través de redes de conmutación de paquetes o asignamiento de canales en redes móviles celulares tienen solución mediante redes tipo Hopfield, la eliminación de ruidos, interferencias o distorsiones en canales de comunicaciones se puede resolver mediante PMF (Perceptrones Multicapa *Feedforward*) o RFBRs (Redes de Funciones de Base Radial) demostrando su superioridad frente a métodos convencionales, sobre todo cuando la transmisión es de alta velocidad y la distorsión es fuertemente no lineal. El PMF también ha demostrado su utilidad en predicción y control adaptativo de tráfico multimedia, donde la alta velocidad, la dificultad de modelado y la necesidad de

adaptación a nuevos servicios de características desconocidas, hacen a los métodos tradicionales muy poco eficientes. En resumen, las RNAs están demostrando ser unas buenas herramientas en el diseño y control de los nuevos sistemas de telecomunicaciones.

Este capítulo pretende ofrecer una panorámica de los trabajos que se han llevado a cabo sobre RNAs en el mundo de las Comunicaciones. Para ello, se propone una clasificación general de aplicaciones, que se desarrolla a continuación. Para cada aplicación, se plantea el problema o la necesidad del sistema en cuestión, se comentan brevemente las soluciones convencionales aportadas hasta la fecha y las ventajas que pueden ofrecer los métodos neuronales. Finalmente, se muestra una solución neuronal al problema planteado. El capítulo se cierra con un apartado de conclusiones sobre la viabilidad y el futuro de las RNAs en este campo.

## 2. CLASIFICACIÓN DE APLICACIONES

En la figura 1 se establece una posible clasificación general de las aplicaciones de las RNAs en las Comunicaciones. Se distinguen tres grandes categorías: Optimización de funciones, proceso de señal y control adaptativo. Por cada categoría, se muestran algunas de las aplicaciones que ya se han llevado a cabo.

## 3. PROBLEMAS DE OPTIMIZACIÓN

### 3.1. Introducción

El problema a resolver puede considerarse genéricamente como un problema de optimización discreta con una complejidad combinatoria en que el vector solución puede tomar únicamente valores discretos, típicamente  $\{\pm 1\}$  ó  $\{0,1\}$ . En todos estos problemas existe un conjunto de posibles soluciones muy amplio, aunque finito, de entre las que queremos encontrar una que optimice globalmente la función de coste. Generalmente, si el problema de optimización combinatoria es de tamaño  $n$ , el número de posibles soluciones son del orden  $e^n$  ó  $n!$ . Esta naturaleza combinatoria de los problemas a tratar, implica que el tiempo necesario para resolverlos crece de una forma exponencial, por lo que llegan a ser algorítmicamente intratables para tamaños moderados o incluso bajos del conjunto de soluciones posibles. De hecho, la obtención de la solución óptima da lugar a problemas del tipo NP-completos [Hérault y Niez, 1991]. Para este tipo de problemas no existe algoritmo capaz de aportar una solución en un tiempo que sea polinómico respecto del tamaño del problema.

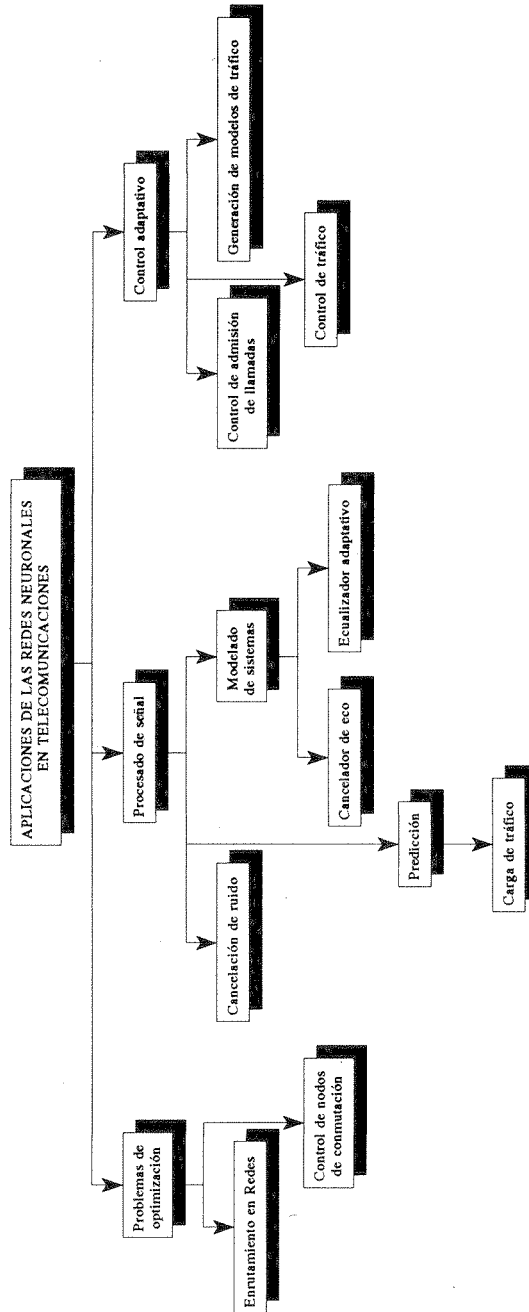


Figura 1. Clasificación de aplicaciones.

En los últimos años se han desarrollado algunos algoritmos heurísticos que ofrecen soluciones cuasi-óptimas en un tiempo de orden polinómico respecto de  $n$  [Simeone, 1989]. Pero existen dos graves inconvenientes que los hacen inapropiados para nuestra aplicación. En primer lugar, los algoritmos conocidos se han desarrollado para aplicaciones muy específicas que dificultan su implementación en ordenadores de procesamiento paralelo. En segundo lugar, y aún más grave, se presenta el inconveniente de los requerimientos de tiempo necesarios para las aplicaciones en comunicaciones. Efectivamente, una característica común a todas las aplicaciones inicialmente expuestas es su necesidad de obtener respuesta en tiempo real. Y este tiempo de respuesta será casi siempre del orden inferior a los milisegundos, llegando en muchos casos a los muy pocos microsegundos. Podemos imaginar que no existe actualmente ningún ordenador ni algoritmo con velocidad suficiente para garantizar-nos dichos tiempos de respuesta. La aproximación neuronal al problema, con su tratamiento analógico paralelo, ofrece posibilidades muy interesantes para llegar a obtener soluciones eficientes y en tiempo real.

### 3.2. Funciones de energía

Todas las aplicaciones indicadas pueden generalizarse como un problema de asignación de recursos sometidos a restricciones. En el paradigma de optimización, los elementos de un conjunto  $\{R_i: 1 \leq i \leq M\}$ , recursos, han de ser asignados a los elementos de otro conjunto  $\{C_j: 1 \leq j \leq N\}$ , consumidores. Una asignación está determinada mediante una matriz binaria  $X = [x_{ij}]$  donde  $x_{ij} = 1$  si  $R_i$  se asocia a  $C_j$  y  $x_{ij} = 0$  en caso contrario. El coste total de la asignación seleccionada está dado por:

$$\text{COSTE TOTAL} = \sum_{i=1}^M \sum_{j=1}^N Q_{ij} x_{ij}$$

donde  $[Q_{ij}]$  es la matriz de costes de asignación de las parejas  $R_i C_j$ . Naturalmente esta asignación no es arbitraria; existen unas restricciones «de bloque» que especifican el número de elementos de  $\{C_j\}$  que pueden ser asociados con cada elemento  $R_i$ , y el número de elementos de  $\{R_i\}$  que pueden ser asociados con cada  $C_j$ . Las restricciones más frecuentemente encontradas son del tipo uno a uno, aunque algunas aplicaciones generan restricciones del tipo uno a varios, o varios a uno.

Las redes neuronales para la resolución de este tipo de problemas han sido extensamente utilizadas desde que Hopfield y Tank [Hopfield, 1985] las aplicaran inicialmente. La idea básica de la aplicación de las redes neuronales en optimización puede resumirse como sigue. Es posible asociar las soluciones de un problema específico de optimización con los mínimos locales de la función de

energía de Liapunov definida por Hopfield para su red aditiva de neuronas interconectadas:

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n T_{ij} V_i V_j - \sum_{i=1}^n V_i I_i$$

donde:

- n** número de neuronas,
- $V_i, V_j$**  estado de la neurona  $V_i, V_j$ ,
- $T_{ij}$**  conexión entre las neuronas «i» y «j»,
- $I_i$**  polarización externa a la neurona «i».

La red resuelve el problema automáticamente convergiendo desde un estado inicial hasta un mínimo local, que correspondería a una solución local del problema. Si la convergencia se realizara hasta un mínimo global, correspondería a una solución global. Como función de activación de las neuronas usualmente se toma la tangente hiperbólica con forma de sigmoide que, en el caso de máxima pendiente, correspondiente a máxima ganancia de la función de activación, se transforma en la función escalón con dos valores posibles de salida de la neurona: «-1» y «1». Estos valores binarios de salida son los correspondientes al tipo de solución de optimización discreta que vamos buscando. El hecho de utilizar neuronas cuya función de activación es de tipo sigmoide nos mejorará el funcionamiento de la red obteniendo una convergencia más rápida y fiable, y con una mayor probabilidad de que corresponda a un mínimo global. El modelo de neurona de McCulloch-Pitts con histéresis ha sido también utilizado reportando buenos resultados [Takefuji, 1993].

El problema se reduce ahora a expresar nuestra aplicación bajo la forma de una función objetivo que sea función cuadrática de energía cuyos mínimos representen nuestras soluciones. Identificando ambas funciones, podremos obtener el valor de los pesos  $T_{ij}$  de las interconexiones de la red de Hopfield y de sus polarizaciones externas  $I_i$ .

### 3.3. Aplicaciones

Presentamos aquí el uso de las técnicas neuronales para la optimización del enrutamiento de mensajes en las redes de telecomunicación basadas en la conmutación y transmisión de paquetes. El enrutamiento presenta una doble problemática que necesita de un tratamiento diferenciado. De un lado, identificamos el problema consistente en la selección óptima de enlaces a través de los nodos de conmutación que conforman una red de comunicaciones, de modo que consigamos mandar información eficientemente entre las diferentes estaciones que la conforman. Como

segundo campo de aplicaciones, identificamos el enrutamiento de paquetes internamente a través de los propios nodos de la red, considerados como conmutadores, o dicho de otro modo, el control mismo de los nodos de conmutación. Un tercer aspecto que comparte ambas problemáticas se está tratando más recientemente desde un enfoque neuronal: la asignación dinámica de canales de comunicación y frecuencias en las modernas Redes Móviles Celulares. Muy semejante a él, y como tal lo consideramos incluido en nuestro estudio, está el problema de la asignación de ventanas de tiempo en los sistemas de Acceso Múltiple por División en el Tiempo (TDMA), tecnología que se utiliza en las redes de comunicación de paquetes vía radio. En todos los casos expuestos el objetivo que se pretende es minimizar los costes de conexión así como optimizar los recursos de la red de comunicaciones.

### *3.3.1. Futuras redes de Comunicaciones*

Los nuevos servicios de comunicaciones se diferencian de los servicios telefónicos convencionales en tres aspectos esenciales. Pueden ser multimedia, multipunto y multivelocidad. Una conexión multimedia puede comunicar audio, datos, imagen fija, video en movimiento o cualquier combinación de entre ellos. Una conexión multipunto implica la posibilidad de interconectar a más de dos abonados, como pueden ser los casos de teleconferencia en que varios abonados se conectan simultáneamente con varios, o la retransmisión de películas de vídeo desde una fuente de programa, en que un servidor se conecta simultáneamente a muchos abonados. Cada tipo de servicio conectado requiere de unas velocidades de transmisión de la información muy diferentes, por lo que la red deberá ser lo suficientemente flexible como para adaptarse a estas variaciones. En contraste con esta situación, pensemos que la telefonía convencional establece comunicaciones de sólo la voz humana, con sólo dos teléfonos por conexión y utiliza circuitos de velocidad fija.

El conjunto de tecnologías necesarias para poder suministrar todos estos servicios se integran en lo que se denomina Red Digital de Servicios Integrados de Banda Ancha (RDSI-BA). La información se transmite en forma digital. Esta información digitalizada se agrupa en paquetes previamente a su transmisión por el canal de comunicaciones. Estos paquetes de información pueden ser de longitud variable, caso de transmisión en modo de transferencia síncrono (STM), o de longitud fija, caso de transferencia en modo asíncrono (MTA). Los estándares últimos se han decantado por ésta última para las comunicaciones futuras. A cada paquete de información, también llamado «celda», se le agrega una cabecera que contiene diversa información entre la que destaca la relativa a las direcciones origen y destino de esa celda (ver figura 2).<sup>1</sup> La técnica utilizada para conseguir que todos estos paquetes «viajen» por un mismo canal de comunicaciones es la de multiplexarlos por división

en el tiempo. En este tipo de multiplexación, el tiempo se divide en «ventanas», («*slots*» en terminología inglesa) y cada paquete de los posibles terminales que quieren transmitir al canal en un momento dado contiene para utilizar cada ventana (figura 3). Debe pues existir un gestor que establezca prioridades de una manera dinámica y decida cada ventana asignando paquetes a *slots* con el mínimo retardo.

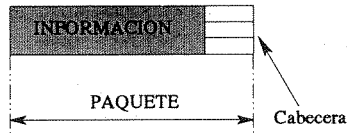
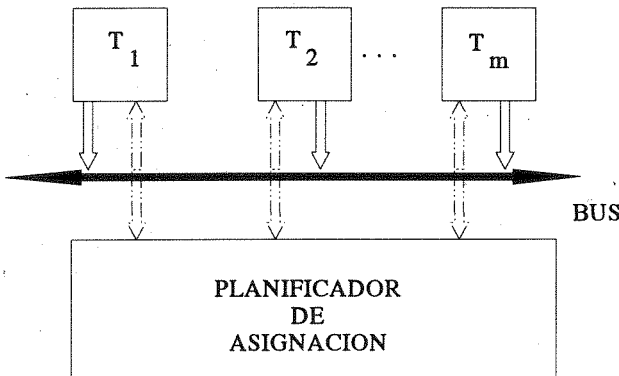


Figura 2. Unidad de información: paquete o celda en MTA.



Figura 3. Multiplexación en el tiempo.



### 3.3.2. Enrutamiento en Redes de Comunicaciones

Al hablar de Red de Comunicaciones podemos entender desde una red instalada en una pequeña oficina, hasta una red de información extendida por todo un país o incluso a nivel mundial. En toda red debemos distinguir los nodos de comunicación y los enlaces (canales de comunicación) que los interconectan. La complejidad de estas redes puede llegar a ser muy respetable con miles de nodos y enlaces interconectados gestionando transferencias de información del orden de Gigabits por segundo. Los enlaces pueden ser de muy diferentes tipos: fibra óptica,

cable coaxial, radio vía satélite, etc., (figura 4). Ello implica una capacidad diferente de canal, diferentes retardos en la transmisión de paquetes, costes diferentes, etc. Los nodos de la red se comportan como centrales de conmutación. Les llegan paquetes con sus direcciones de origen y destino y deben enrutarlos seleccionando el mejor camino entre cada dos nodos. Este enrutamiento es una de las más importantes tareas en la red. Existen mecanismos de enrutamiento adaptativo basados en estrategias predefinidas que fundamentalmente utilizan tablas dinámicas de rutas orientados a optimizar la utilización de recursos, evitar sobrecarga de enlaces o nodos, incrementar la velocidad de transferencia de información o reducir los costes de las comunicaciones. Pero a medida que la complejidad de la red aumenta y los servicios introducidos necesitan de mayor ancho de banda o velocidad de transferencia, los algoritmos convencionales se revelan incapaces de resolver estos problemas de gran carga computacional y con unas especificaciones de tiempo muy estrictas. Se detecta, pues, la necesidad de un gestor centralizado que dinámicamente asigne rutas dentro de una red de comunicaciones optimizando unos parámetros predefinidos.

Para ilustrar el problema, consideremos la red de la figura 4 donde los nodos pueden representar diferentes localidades geográficas y las líneas que los unen representan los enlaces entre ellas con un coste asociado. El coste del enlace puede ser tabulado en función de la capacidad del enlace, tarifas comerciales, retardo de propagación, nivel de tráfico instantáneo, probabilidad de congestión, etc. En general, cualquier conjunto de parámetros que nos determine una métrica del enrutamiento y consecuentemente influya en la elección de ruta. Podrán existir aplicaciones específicas de enrutamiento con su métrica específica que darán lugar a una elección de rutas específica. El enrutamiento en la red requiere encontrar el camino a través de nodos y enlaces desde un nodo origen **O** a un nodo destino **D** en un instante dado que minimice una función de coste de acuerdo con la métrica asignada.

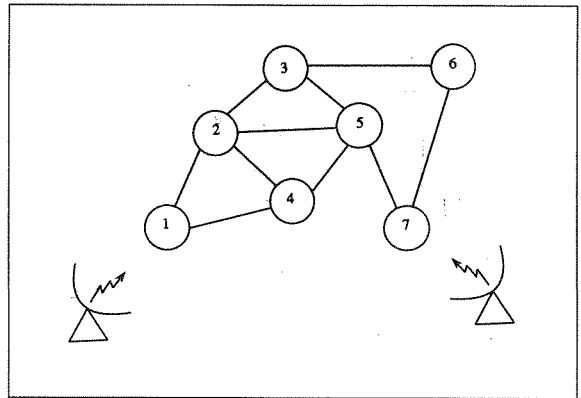


Figura 4. Red de Comunicaciones.

Esta optimización requiere de un conocimiento previo de los patrones de tráfico de la red. Así, las compañías actuales de comunicación utilizan tablas de rutas

que establecen las mejores opciones en cada caso de acuerdo con una experiencia previa. El conocimiento adquirido sobre la situación instantánea del tráfico de la red, serviría para actualizar periódicamente dichas tablas de ruta. Pero esta actualización periódica estaría limitada, pues la tarea misma de adquirir y transmitir la información del estado de la red implica una sobrecarga para la misma. Los algoritmos de enrutamiento basados en tablas de ruta son, pues, algoritmos estáticos incapaces de reaccionar a los cambios instantáneos de tráfico. Hemos de añadir a ello que, como ya comentamos, la complejidad del algoritmo precisa de un tiempo para su ejecución superior al tiempo de respuesta requerido en los modernos sistemas de comunicaciones. Por contra, los algoritmos adaptativos reaccionan a los cambios instantáneos de tráfico producidos en la red, seleccionando en cada caso el camino de «mínimo coste».

La solución neuronal a este problema, en general, utiliza la red asociativa de Hopfield analógica como optimizador según el modelo ya estudiado [Rauch y Winarske, 1988; Zhang y Thomopoulos, 1989; Cavalieri y col., 1993]. Dada una red de  $N$  nodos y definidos unos costes de enlace  $w_{ij} = w_{ji}$ , la matriz simétrica  $[w_{ij}]$  de  $N \times N$  elementos representaría la matriz de costes de la red. Para aquellos nodos entre los que no hubiera enlace se asignaría un coste infinito (valor muy alto). El camino óptimo a través de la red se define por una matriz de permutación  $N \times N$  donde las filas representan los nodos de la red y las columnas la posición del nodo en la ruta. Vemos que el problema está muy relacionado con el problema del viajante. Pero, a diferencia de él, una ruta elegida puede incluir menos de los  $N$  nodos. Por lo tanto se añaden pseudoenlaces de coste cero a cada nodo de forma que garanticemos que en cada columna de la matriz de permutación exista un «1» sin que se incremente el coste total de la ruta seleccionada. De la descripción de la matriz de permutación, deducimos que debe cumplir las dos siguientes condiciones: 1)  $N$  de sus términos exactamente deben tomar el valor «1» y 2) Cada columna contiene exactamente un término con valor «1». La función cuadrática de energía que particulariza nuestra aplicación sería:

$$\begin{aligned}
 E = & \frac{A}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=2}^{N-1} w_{ij} V_{ik} (V_{jk+1} V_{jk-1}) && \text{Coste del enlace} \\
 & + \frac{B}{2} \sum_{k=2}^{N-1} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N V_{ik} V_{jk} && \text{Restricción de columna} \\
 & + \frac{C}{2} \left( \sum_{i=1}^N \sum_{j=1}^N V_{ij} - N \right)^2 && \text{Número de 1's igual a } N
 \end{aligned}$$

La resolución del algoritmo implica que se conocen los nodos origen y destino. Por lo tanto, al inicio de la optimización, se fijan a «1» los valores de las columnas primera y última en la matriz de neuronas para los nodos origen y destino, y a «0» el resto de valores de dichas columnas. Todas las otras columnas se fijan con valores aleatorios teniendo en cuenta que la suma total de todas las entradas a la matriz debe valer  $N$

Diversos autores han detectado problemas inherentes a este tipo de aproximación neuronal al enrutamiento. Fundamentalmente, al igual que en el problema del viajante, la elección de los multiplicadores de Lagrange  $A, B, C$  debe ser heurística e influye considerablemente en la calidad de la solución. Igualmente la convergencia de la red no está garantizada cuando el número de nodos se incrementa. ¿Convergerá nuestro modelo cuando el número de nodos de la red aumente a 1000 o incluso más?

Actualmente se exploran nuevas alternativas neuronales más prometedoras [Zhang, 1993; Littman y Boyan, 1993; Martínez y Rudolph, 1993] basadas en modelos de aprendizaje por refuerzo y redes neuronales autoasociativas de Kohonen [Kohonen, 1989]. Estas redes se han mostrado muy eficientes en la solución del problema del viajante [Angeniol y col., 1988]. En nuestro problema, la topología de la red y sus características estadísticas espaciales y temporales pueden ser reflejadas en un mapa autoorganizativo que se aplica al enrutamiento. Esta solución trabaja bien incluso en redes de gran tamaño donde el paralelismo de la red neuronal se muestra más ventajoso. Igualmente presenta la ventaja de no necesitar formalizar expresamente las condiciones de enrutamiento para una aplicación específica mediante la definición de una función objetivo.

### 3.3.3. Control de matrices de conmutación

La configuración interna de un nodo de conmutación está basada en la interconexión modular de varias matrices de interconexión. Esencialmente una matriz de interconexión es un módulo conmutador con  $m$  líneas de entrada y  $n$  líneas de salida capaz de seleccionar a la salida una permutación de las entradas (figura 5). A las entradas llegan secuencialmente las celdas que deben ser dirigidas a diferentes salidas según la información que contienen en su cabecera. Dado que la tasa de llegada de paquetes por las líneas de entrada puede ser mayor que la capacidad del conmutador para transferirlos de entrada a salida, existen unas colas donde se almacenan previamente (figura 6). En cada ventana de tiempo será necesario seleccionar un conjunto de celdas, una por cada línea de entrada, para ser transferidas a la salida. Los paquetes seleccionados deberán poder propagarse a través del conmutador sin que haya colisión en ningún camino interno. Igualmente no podrán seleccionarse dos paquetes que deseen acceder a la misma dirección de salida en el

mismo instante. Al igual que en los casos anteriores, los algoritmos de competición secuencial diseñados actualmente, son incapaces de transferir paquetes a la velocidad requerida por la red. Para esta aplicación se han desarrollado controladores neuronales centralizados que ofrecen interesantes resultados.

Existen diferentes tipos de matrices de conexión que se utilizan en los nodos de interconexión de red. El conmutador básico es el de barras cruzadas o «*crossbar*». Conceptualmente consiste en una rejilla de  $N \times N$  hilos con un contacto en cada punto de cruce que puede estar abierto o cerrado, conectando de esta forma cualquier línea de entrada a cualquier línea de salida. Este conmutador permite establecer en las líneas de salida cualquier permutación de las líneas de entrada en cualquier instante. Es el prototipo de los conmutadores que llamamos estrictamente no bloqueables. Es, indudablemente, el más deseable, pero lo hace a costa de  $N^2$  puntos de cruce. El número de puntos de cruce debe ser minimizado si queremos disminuir su complejidad, y por ello, aumentar su fiabilidad y reducir su coste. Una forma de conseguirlo es construyendo conmutadores a partir de etapas formadas por otros más pequeños de tipo barras cruzadas. Estas son las llamadas Redes de Interconexión Multietapa. Pero esta reducción de puntos de cruce se consigue a costa de perder el carácter de estrictamente no bloqueable. En ellas, dos paquetes procedentes de entradas diferentes y con destinos a salidas diferentes, pueden entrar en conflicto al necesitar compartir algún camino interno del conmutador, común a las rutas fijadas. A esta situación se la denomina de bloqueo interno. Otro tipo de bloqueo que puede darse es el bloqueo a la salida. En esta situación, al menos dos paquetes intentan acceder simultáneamente a la misma línea de salida. Este tipo de bloqueo es independiente del tipo de conmutador y únicamente puede evitarse mediante una apropiada estrategia de selección de los paquetes que esperan en las líneas de entrada.

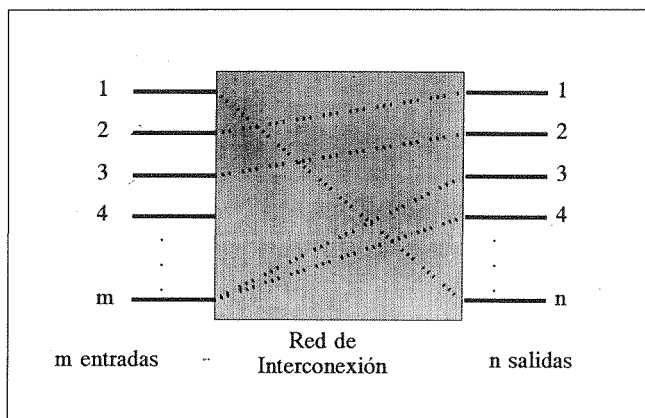


Figura 5. Red de interconexión.

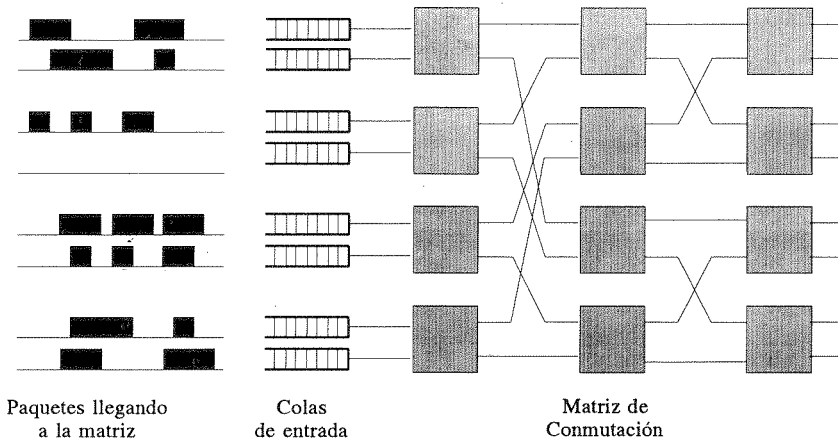


Figura 6. Matriz de conmutación. Colas de entrada.

La utilización de las Redes de Interconexión Multietapas (**RIM**) como conmutadores espaciales en los nodos de conmutación en Modo de Transferencia Asíncrona (**MTA**) ha sido aceptada como la arquitectura más extendida [Minzer, 1988]. Una **RIM** de  $N \times N$  consiste de  $\log_2 N$  etapas de conmutación. Cada etapa consiste de  $N/2$  elementos de conmutación tipo *crossbar* de  $2 \times 2$ . Dentro de las diversas topologías existentes, existen dos clases bien diferenciadas de ellas atendiendo al tipo de enrutamiento de un paquete a su través. De un lado Goke y Lipovski [Goke y Lipovski, 1973] definieron las redes tipo *Banyan* que se caracterizan por su propiedad de autoenrutado. Para un paquete situado en un puerto de entrada y queriendo acceder a una dirección de salida, existe una única ruta posible; la dirección del puerto de salida es la única información que precisan los conmutadores de las sucesivas etapas para enrutar el paquete desde la entrada en que se encuentra a la salida deseada. Otro tipo de **RIM** son aquellas que presentan varias rutas posibles para un paquete con un determinado par de direcciones Entrada/Salida. Las Redes de Benes [Benes, 1965] son un ejemplo de ellas. En este caso, para enrutar un paquete se precisa seleccionar el estado de los conmutadores internos apropiados para la ruta elegida de entre las posibles.

En estas **RIM** pueden presentarse diferentes condiciones de bloqueos. Ello acarrearía una pérdida de información, posibilidad inherente y admitida en **MTA**. El problema del control de tráfico en estas redes es el de minimizar estas pérdidas de información, o de otro modo, maximizar el número de paquetes interconectados libres de bloqueo en cada ventana de tiempo. Este es un problema de complejidad

NP-Completa por lo que ningún algoritmo podría encontrar una solución óptima en un tiempo razonable para los retardos y anchos de banda de conmutación requeridos en MTA. Para las **RIM** autoenrutables o tipo *Banyan*, la solución consiste en elegir de entre todos los paquetes esperando en los puertos de entrada, un subconjunto de ellos que no se bloqueen en la red en cada ranura de tiempo en **MTA**. Dicho subconjunto será el que la Red pueda conmutar de Entrada a Salida sin pérdida de información. En la figura 7 se muestra una Red de tipo *Banyan* con un paquete en la línea de entrada **110** accediendo a la línea de salida **011**. En el caso de las Redes tipo *Benes* o multiruta, para un conjunto de paquetes en los puertos de entrada queriendo acceder a determinados puertos de salida, será preciso generar una «tabla de rutas» para todos los paquetes, seleccionando determinados caminos de entre todos los posibles y de modo que no se produzcan bloqueos en el conmutador. En general, el problema de enrutamiento en una **RIM** puede ser considerado como un problema de optimización sujeto a restricciones [Brown, 1990]. La capacidad de proceso paralelo y la similitud de topologías entre las Redes Neuronales y las matrices de conmutación, han inducido a numerosos investigadores a resolver el problema de control desde una perspectiva neuronal. T.X. Brown y K.H. Liu [Brown y Liu, 1990] proponen una solución para matrices *autorouting* tipo *Banyan*. A.M. Marrakchi y T. Troudet [Marrakchi y Troudet, 1989] lo hacen para matrices *Crossbar*. Goudreau y Giles [Goudreau y Giles, 1991] tratan el problema de redes no *autorouting* y de estructura irregular. En todos estos casos, el cuello de botella que representa el tiempo de proceso necesario para, bien seleccionar los paquetes a presentar a la red, bien generar los bits de control que definan la tabla de rutas, se supera mediante un

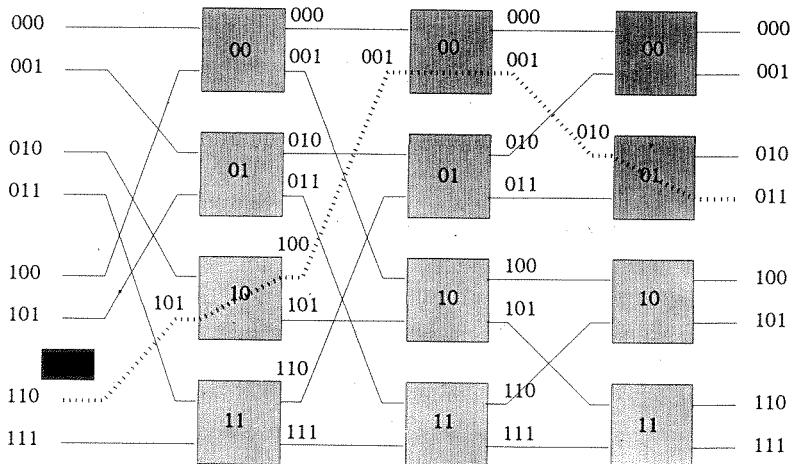


Figura 7. Red Banyan.

proceso paralelo analógico ejecutado por un Controlador Neuronal Centralizado (CNC).

En la figura 8 se representa el diagrama de bloques de un controlador neuronal genérico para un conmutador espacial en MTA. Cada elemento  $a_{ij}$  de la Matriz de Entrada representa el par de direcciones  $E_i/S_j$  de Entrada/Salida de un paquete esperando en los *buffers* de entrada. La Matriz de peticiones es una matriz cuadrada binaria cuyos elementos  $b_{ij}$  los obtenemos mapeando sobre ella la Matriz de Entrada, de forma que  $b_{ij}$  es igual a 1 si existe al menos un correspondiente  $a_{ij}$ . Caso contrario,  $b_{ij}$  es igual a -1. La función del controlador neuronal es seleccionar para cada *slot* de tiempo, y ante cada matriz de peticiones instantánea, un subconjunto de paquetes en espera en los *buffers* de entrada que no generen bloqueos internos ni de salida.

Para un conmutador de  $N \times N$  el controlador Neuronal propuesto es una Red de Hopfield de  $N \times N$  neuronas, interconectadas en forma de circuitos múltiplemente solapados del tipo Ganador-Toma-Todo (WTA). Esta estructura ha sido propuesta y estudiada en [Brown y Liu, 1989; Majani y col., 1989].

Las Redes de tipo *Banyan* pertenecen al tipo de camino único, es decir, existe un único camino para cada par de direcciones Entrada/Salida a su través. Debido a ésta propiedad de camino único, no es necesaria la búsqueda de caminos libres para un paquete en espera en sus *buffers* de entrada. Ello libera de una gran carga de trabajo al control del conmutador permitiéndole trabajar a velocidades MTA. Pero como contrapartida, implica que es bloqueable y esto puede tener un impacto no admisible en cuanto a pérdidas de celdas, especialmente para el caso de tráfico no constante.

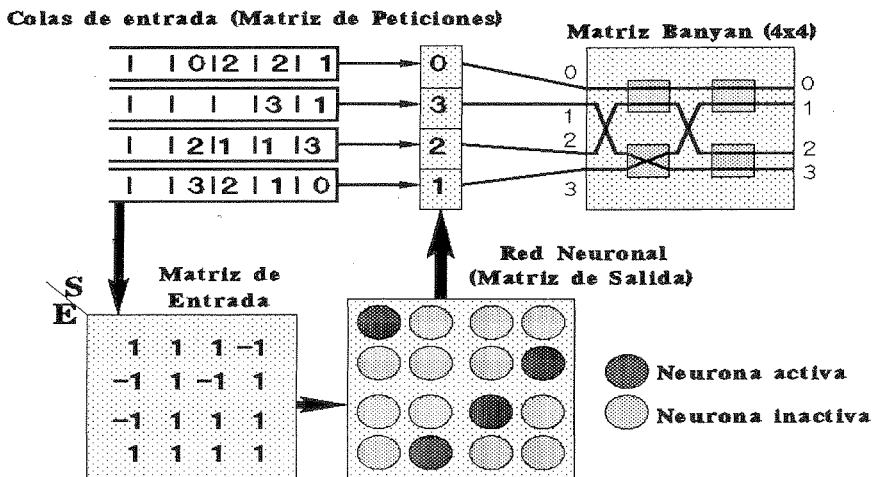


Figura 8. Controlador neuronal de matriz de conmutación.

Para compensar este problema de bloqueo, manteniendo la capacidad de *autorouting*, se han estudiado diversas estructuras de asociaciones de Redes Bloqueables [Shyy, 1991]. Fundamentalmente se busca generar una Red estrictamente no bloqueable mediante la combinación en vertical y horizontal de estructuras bloqueables. La estructura así creada presenta más de un camino de entrada/salida, y sigue siendo autoenrutado. Como ventaja adicional, presentan tolerancia a fallos, al permitir caminos alternativos en el caso de que uno o más de los conmutadores internos presente un fallo. García Lopera [García-Lopera y col., 1993] propone una estructura de interconexión en paralelo de los controladores neuronales correspondientes a una asociación de matrices de este tipo.

En la figura 9 representamos una asociación de Matrices de Conmutación en paralelo cada una con su Controlador Neuronal asociado. Los *buffers* de entrada son comunes a todas las Matrices simultáneamente, de modo que una misma celda MTA puede alcanzar la salida deseada por cualquiera de las Matrices. Dado que varias celdas pueden alcanzar la misma salida procedente de diferentes entradas en un mismo *slot*, será preciso establecer *buffers* de salida con la longitud mínima adecuada. En principio ésta no debe ser inferior al número de matrices en paralelo. Observamos que al sistema se le presenta una única Matriz de Entrada. El problema planteado ahora es, partiendo de los controladores individuales de cada Conmutador, ya estudiados, cómo interconectarlos de modo que actúen como un único controlador de la asociación. La capacidad de enrutamiento del sistema deberá mejorar drásticamente al existir varias rutas posibles simultáneamente. Asimismo el controlador deberá presentar un esquema tolerante a fallos en el sentido de que deberá ser capaz de no utilizar aquellas rutas que presenten conmutadores internos con fallo, encaminando los paquetes en espera a través de las rutas alternativas existentes.

En la figura 10 representamos la interconexión entre cada controlador individual de la asociación en paralelo de Matrices de conmutación. Se representan las matrices de petición correspondiente a cada conmutador ( $k=1, k=2, \dots, k=n$ ). Los elementos  $\mathbf{a}_{ij1}, \mathbf{a}_{ij2}, \dots, \mathbf{a}_{ijn}$ , corresponden a un mismo par Entrada/Salida en la Matriz de Entrada, común a todos los conmutadores. La celda de direcciones respectivas  $\mathbf{E}_i/\mathbf{S}_j$ , debe ser direccionada a través de uno sólo de los conmutadores en paralelo. Para ello conectamos todas las correspondientes neuronas asociadas a los elementos  $\mathbf{a}_{ij1}, \mathbf{a}_{ij2}, \dots, \mathbf{a}_{ijn}$  en un circuito del tipo Ganador-Toma-Todo (WTA) solapado con los propios de cada controlador individual. Así, la salida de cada neurona inhibe a las demás. La neurona ganadora evita que el resto de ellas gane en su respectivo WTA solapado evitando de este modo que la célula  $\mathbf{E}_i/\mathbf{S}_j$ , sea seleccionada por más de un Controlador. De esta forma la Matriz de Peticiones es subdividida en  $n$  máximos conjuntos disjuntos solapándose con el conjunto  $\mathbf{C}$  que forman todas las posibles direcciones de Entrada/Salida. Cada controlador neuronal envía una Matriz de Salida a su correspondiente Matriz de Conmutación. Por lo tanto, la salida después de la

computación paralelo neuronal será un máximo de  $k$  subconjuntos disjuntos de celdas no bloqueables. Se consigue con esta configuración una disminución sensible de las colas de entrada necesarias para pérdidas cero, incluso en el caso de fallo de conmutadores internos de las matrices.

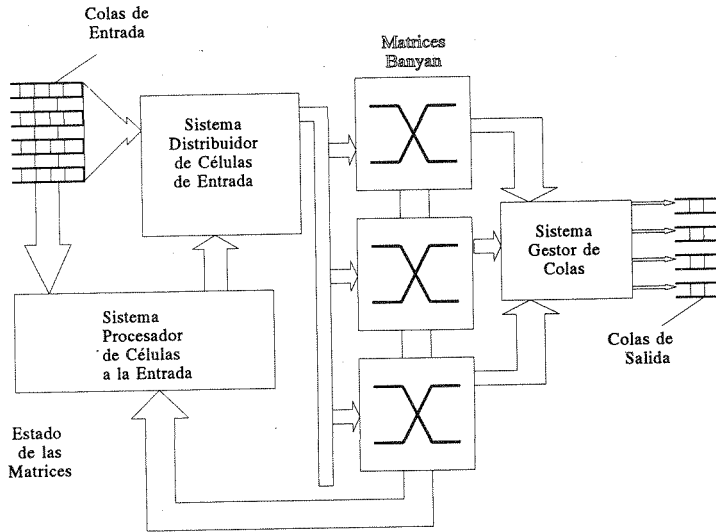


Figura 9. Controlador de matrices en paralelo.

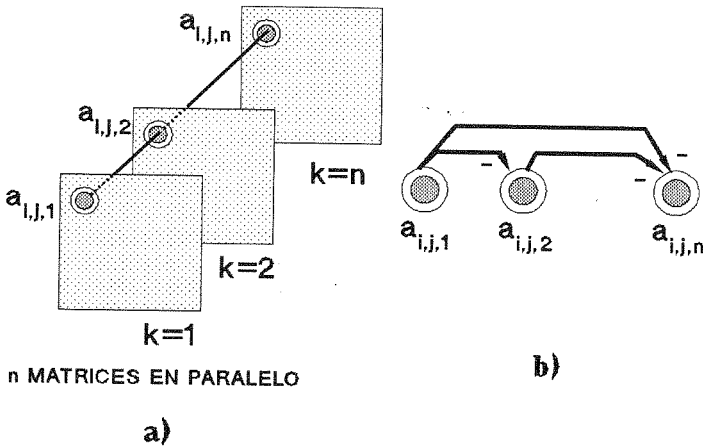


Figura 10. Asociación de controladores.

### 3.3.4. Aplicaciones a las Radiocomunicaciones

Recientemente se están explorando las posibilidades de aplicación de los paradigmas neuronales en el campo de las comunicaciones vía radio. Un problema estudiado [Funabiki, 1992; Nakano, 1991] es el de la asignación de canales en las Redes Móviles Celulares. Se plantea como un problema de asignación de canales y frecuencias para cada célula en que se divide la red, debiendo satisfacerse las restricciones de frecuencias representadas en una matriz de compatibilidad. Otra aplicación estudiada [Funabiki, 1993] es el de la sincronización de estaciones radiotransmisoras en una red de radiodifusión para comunicación de paquetes. La red utiliza canales de radio como medio de interconexión entre sus nodos. La tecnología de acceso es TDMA (Acceso Múltiple por División en el Tiempo). Ello permite a varios usuarios compartir un mismo canal mediante el acceso sincronizado y supervisado a «ventanas» de tiempo (*slot*). Las condiciones de conexión de la red imponen una matriz de restricciones al gestor que debe asignar cada paquete a un determinado *slot*.

Como vemos, todas estas aplicaciones pueden contemplarse como un problema de optimización combinatoria discreta en que hemos de determinar una asignación concreta sujeta a unas restricciones determinadas. El tratamiento que los autores citados hacen pasa por establecer una función objetivo que describa el problema, e identificarla con la función de energía de Hopfield, tal como hemos estudiado.

## 4. PROCESO DE SEÑAL

### 4.1. Introducción

Son muy variadas las aplicaciones de las RNAs en el proceso de señal. En particular, para Comunicaciones son de gran interés los campos de cancelación de ruido, modelado de sistemas y predicción de series temporales.

### 4.2. Cancelación de ruido

La eliminación de ruido o interferencias asociadas a la señal es uno de los problemas a resolver más típicos en Comunicaciones. Las técnicas convencionales utilizan filtros óptimos de Wiener o de Kalman [Kailath, 1981] que no pueden evitar cierta distorsión de la señal y algún ruido remanente. Las RNAs ofrecen atractivas soluciones en este campo [Bijani, 1990], ya que pueden comportarse como potentes filtros adaptativos no lineales.

En la figura 11 se presenta una técnica adaptativa típica [Widrow y Stearns, 1985], basada en una RNA. La técnica necesita disponer de una entrada de referencia que contenga un ruido  $n_1$  correlacionado con el ruido original  $n_0$ , que se quiere eliminar. La RNA recibe el ruido de referencia, lo filtra, y a continuación, lo resta del resultado de la entrada primaria de ruido  $s+n_0$ . En la RNA, la entrada ruidosa actúa como respuesta deseada y la salida del sistema como error del filtro.

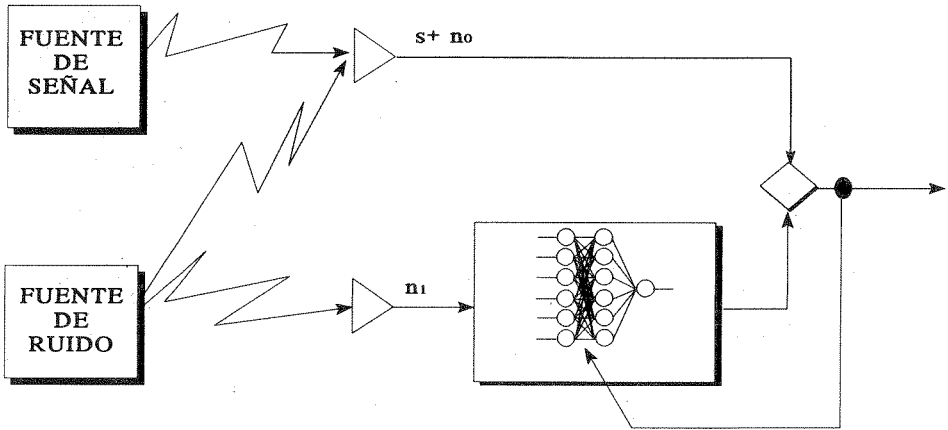


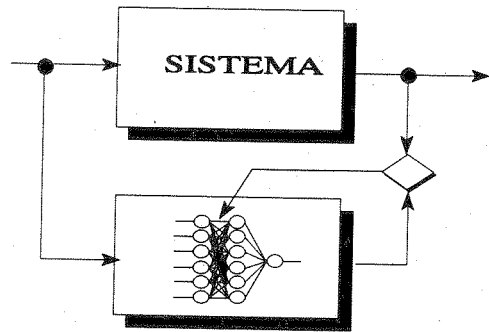
Figura 11. Cancelación de ruido.

### 4.3. Modelado de sistemas

El modelado de sistemas es de gran utilidad en Comunicaciones. Sus aplicaciones más típicas se encuentran en el control de sistemas y en la identificación de canales de comunicaciones. Su objetivo básico es obtener la función de transferencia de un sistema, supuestamente desconocido. Las RNAs son una buena solución en el modelado de sistemas dinámicos no lineales y de sistemas completamente desconocidos. Varios autores [Hornik y col., 1989], [Funahashi, 1989] han demostrado que cualquier mapeado continuo sobre un dominio compacto puede ser aproximado con tanta seguridad como sea necesaria por medio de una red neuronal *feedforward* con una capa oculta. Se han propuesto muchos tipos de RNAs para modelado [Sofge y White, 1992] entre los que destacan el perceptrón multicapa *feedforward* (PMF), el CMAC (*Cerebellar Model Articulation Controller*) y las redes de función de base radial.

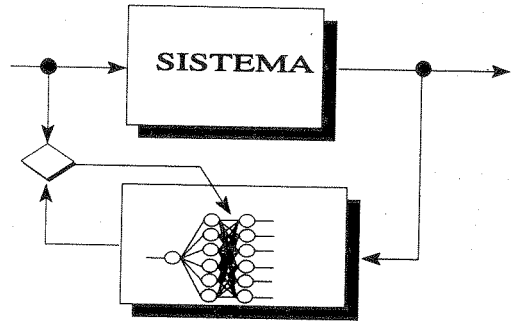
En la figura 12 se ilustran los dos esquemas básicos de modelado basado en RNAs: el modelo directo y el modelo inverso.

En el modelo directo, o de identificación del sistema, la RNA aprende, por observación, el comportamiento de un sistema desconocido. Normalmente, la RNA tiene las mismas entradas y salidas que el sistema. La RNA intenta minimizar, para cada entrada, la diferencia entre su salida y la del sistema. Después del período de entrenamiento, la RNA se comportará igual que el sistema, siendo capaz de predecir la salida del sistema para cualquier entrada.



a) Modelo directo

En el modelo inverso, las salidas (entradas) del sistema son las entradas (salidas) de la RNA. Esto permite obtener la inversa de la función de transferencia del sistema desconocido. En el entrenamiento, la RNA intenta minimizar la diferencia entre su salida y la entrada del sistema. Esta diferencia será mínima cuando el conjunto en cascada formado por el sistema desconocido y la RNA tenga una función de transferencia unidad.



b) Modelo inverso

Figura 12. Modelos de sistemas.

#### 4.4. Predicción de series temporales

En muchas aplicaciones de Comunicaciones, se plantea la necesidad de predecir el futuro de una serie temporal de muestras, a partir del conocimiento de las muestras anteriores. Esta predicción de futuro en tiempo real está íntimamente relacionada con el control adaptativo y el modelado de sistemas. La solución ideal es, siempre, descubrir la ley que gobierna la dinámica del sistema y describirla analíticamente (mediante un conjunto de ecuaciones diferenciales). Con esta ley será fácil determinar el futuro, siempre y cuando se conozcan las condiciones iniciales. Desafortunadamente, no siempre se tiene toda la información necesaria sobre el

sistema dinámico, de forma que la predicción no se puede basar sobre un modelo analítico conocido. Otra solución es intentar descubrir fuertes regularidades empíricas mediante observación de la serie temporal. Por ejemplo, las muestras de una serie temporal de un proceso periódico se pueden modelar como una superposición de sinusoides, generadas por un conjunto de ecuaciones diferenciales de segundo orden. Por desgracia, en muchos problemas reales las regularidades están enmascaradas por el ruido. Es más, hay fenómenos descritos por series temporales caóticas donde los datos parecen aleatorios sin ninguna periodicidad aparente. Hay que destacar que no son posibles las predicciones a largo plazo de series temporales caóticas, ya que el error (incertidumbre de la predicción) se incrementa exponencialmente con el tiempo. No obstante, si es posible hacer predicciones a corto plazo para estas series, habiéndose obtenido resultados prometedores con soluciones basadas en RNAs. En el campo de las Comunicaciones se han realizado algunas aportaciones muy interesantes [Connors, 1993; Tarraf y col., 1993].

Hay varios modelos [Narendra, 1992] para la predicción de series temporales, que pueden ser aplicados mediante RNAs. A continuación, se comentan brevemente algunos de ellos.

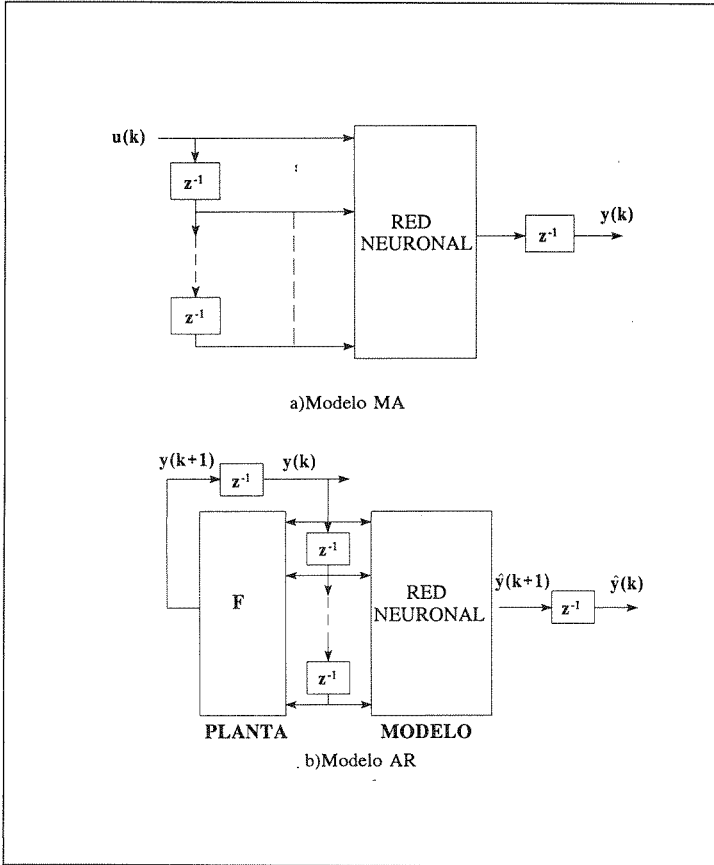
El modelo MA (*moving average*) predice una salida en función de las entradas presentes y pasadas. Se puede implementar con RNAs, mediante una estructura como la mostrada en la figura 13a. La RNA estima el valor de  $y(k+1)$ , según la ecuación,  $y(k+1)=f(u(k+1), u(k), \dots, u(k-n), \mathbf{W})$ , siendo  $\mathbf{W}$  la matriz de pesos y  $u(k)$  la entrada en el instante  $k$ . La elección de  $n$  depende del grado de exactitud deseado.

En el modelo AR (*autoregressive*), la salida, en cada instante, depende de las  $n$  salidas anteriores. Se puede implementar mediante RNAs, tal como se ve en la figura 13b. En este caso, la RNA estima el valor de  $y(k+1)$ , según la ecuación  $y(k+1)=f(y(k), y(k-1), \dots, y(k-n), \mathbf{W})$ .

El modelo ARMA combina MA y AR. Su salida, en un instante dado, se puede aproximar mediante sus salidas y entradas anteriores. La RNA estima el valor de  $y(k+1)$ , según la ecuación  $y(k+1)=f(y(k), y(k-1), \dots, y(k-n), u(k+1), u(k), \dots, u(k-m+1), \mathbf{W})$ .

#### 4.5. Aplicaciones

A continuación, se presentan algunas aplicaciones que ilustran la viabilidad del uso de RNAs en los campos de proceso de señal mencionados anteriormente.



**Figura 13.** Modelos de predicción. a) Modelo MA. b) Modelo AR.

#### 4.5.1. Cancelación de eco

Una aplicación típica de modelado directo es el cancelador de eco. En telefonía, se utiliza un circuito denominado híbrida o duplexor que permite la transmisión y recepción simultánea con sólo dos hilos. Se denomina eco cercano a una réplica (distorsionada y atenuada por el duplexor) de la señal transmitida, que aparece a la entrada del receptor. También existe un eco lejano, originado en la híbrida de la central remota. Se define el canal de eco como el camino que sigue el eco desde que sale del transmisor hasta que entra en el receptor. En transmisión de datos mediante *modems*, el eco cercano puede llegar a ser 200 veces mayor que la señal recibida, mientras que el eco lejano es mucho más pequeño, aunque llega al *modem* con mucho retraso y afectado de distorsiones no lineales. Ambos ecos imposibilitan una correcta recepción de los datos. Para evitar estos ecos indeseables

se utilizan los denominados canceladores de eco que consisten en unos filtros adaptativos que ajustando apropiadamente sus coeficientes, son capaces de simular el canal de eco.

En la figura 14 se ilustra un esquema de un bucle local de abonado con cancelador de eco local. Tomando como entrada del filtro la señal transmitida, obtendrás a la salida una replica del eco que se ha producido. Si este eco sintetizado se le resta a lo que llega al receptor (eco + señal recibida) se podrá obtener una señal recibida libre de interferencias. Para bajas velocidades de transmisión, se han utilizado, como canceladores de eco, filtros lineales con resultados aceptables, sin embargo, conforme aumenta la velocidad de los datos, se acentúan los efectos no lineales y estas soluciones no dan los niveles de cancelación requeridos. Hay por tanto, que ir a otro tipo de soluciones como las memorias adaptativas, series de Volterra o RNAs [Weruaga y Figueiras, 1992; Dali, 1992].

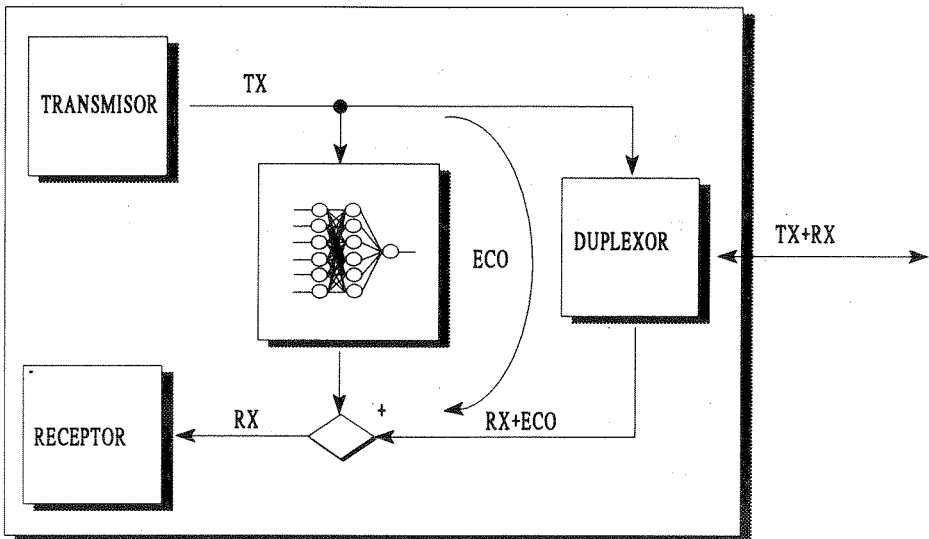


Figura 14. Cancelación de eco.

Weruaga y Figueiras [1992] proponen un filtro supertransversal de Hetch-Nielsen formado por un filtro transversal lineal de  $n$  coeficientes y un PMF en paralelo, solapado parcialmente con el anterior (ver figura 15). El algoritmo de entrenamiento consta de dos fases: fase lineal y fase conjunta. En la fase lineal, sólo se adapta el filtro transversal (mediante un algoritmo LMS (*Least Mean Square*)). Cuando

el eco se reduce lo suficiente como para que empiece a influir el efecto no lineal, se pasa a la fase conjunta donde se entrenan el filtro transversal y el PMF. El PMF usa el algoritmo de aprendizaje *backpropagation* con término momento. El filtro consigue cancelar correctamente el eco aunque su tiempo de convergencia es lento. Comparado con canceladores convencionales tiene ventajas e inconvenientes. Por ejemplo, el número de pesos o coeficientes es un orden de magnitud inferior que el de un filtro lineal en paralelo con una memoria adaptativa y aprendizaje LMS, sin embargo, el coste computacional es mayor.

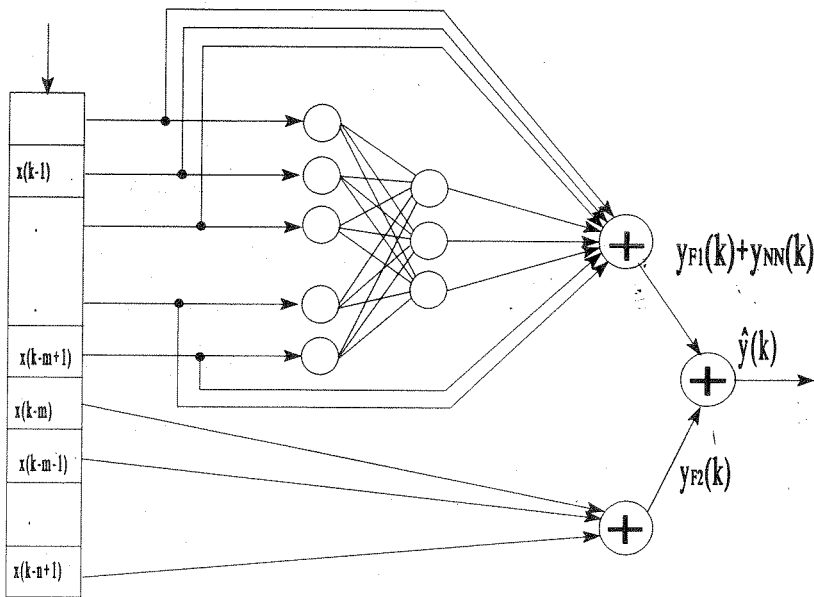


Figura 15. Estructura de un cancelador neuronal de eco.

#### 4.5.2. Ecuación adaptativa

La ecualización adaptativa es un ejemplo de modelo indirecto. Cuando se envían datos digitales a alta velocidad, a través un canal telefónico o de radio, suele aparecer un fenómeno indeseable, denominado interferencia entre símbolos, debido a que el canal no tiene un comportamiento ideal (su respuesta en frecuencia y fase no es plana ni lineal, respectivamente). La interferencia entre símbolos puede llegar a provocar tasas de error de bit del orden de  $10^{-1}$ .

El método ideal para recuperar los datos transmitidos es el MLSE (*Maximum Likelihood Sequence Estimation*) [Proakis, 1989], sin embargo, su complejidad es, algunas veces, inaceptable en ciertos sistemas de comunicaciones. De hecho, muchos ecualizadores prácticos emplean métodos más sencillos que hacen las decisiones símbolo a símbolo. La técnica más utilizada se basa en el modelo inverso. La figura 16 presenta un modelo, discreto en el tiempo, de un sistema de telecomunicaciones digital, donde la secuencia de datos a transmitir  $q(t)$  se modula, generando un conjunto finito de símbolos. La secuencia de símbolos obtenidos  $s(t)$  se transmite a través de un canal dispersivo. A la salida del canal se añade ruido  $n(t)$ , obteniéndose, finalmente, una secuencia recibida  $r(t)$  que se aplica al ecualizador adaptativo. En una primera fase de entrenamiento, el ecualizador ajustará sus coeficientes con el objeto de minimizar la diferencia entre su salida  $\hat{q}(t)$  y  $q(t)$ . Inicialmente, la salida del ecualizador adaptativo será muy ruidosa, sin embargo, con el tiempo el ruido irá disminuyendo y la salida se irá aproximando a su valor deseado, obteniéndose tasas de error del orden de  $10^{-6}$  o menores.

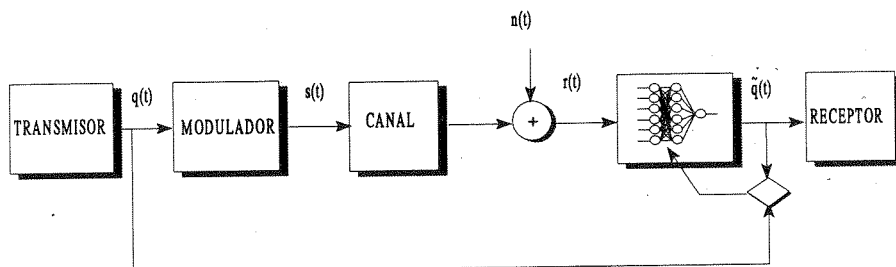


Figura 16. Ecualización adaptativa.

Normalmente, el ecualizador adaptativo es un filtro lineal, denominado ecualizador transversal lineal (LTE), sin embargo, el uso de PMFs [Meyer y Pfeiffer, 1993; Brown, 1993], filtros polinómicos [Chen y col., 1990] y redes de función de base radial [Chen y col., 1993; Cid y Figueiras, 1993] como ecualizadores adaptativos mejora el rendimiento ofrecido por un LTE, ya que estos ecualizadores no lineales son capaces de aproximar implícitamente la solución óptima del ecualizador.

Meyer y Pfeiffer [1993] simulan un sistema de telecomunicaciones digital con modulación QPSK (que genera un símbolo por cada dos bits transmitidos). La secuencia de símbolos obtenidos  $s(t)$  se transmite a través de un canal dispersivo con función de transferencia del tipo:

$$H(z) = \sum_{i=1}^n h_i z^{-i}$$

La secuencia  $s(t)$  sólo toma valores de +1 y -1. A la salida del canal se añade ruido gaussiano blanco y distorsión no lineal, obteniéndose una secuencia recibida  $r(t)$  que se aplica a un PMF *backpropagation* actuando como ecualizador adaptativo. El PMF elimina las perturbaciones originadas en el canal de transmisión y demodula la secuencia recibida. La función de error usada es la diferencia entre  $\tilde{q}[t]$  y  $q[t]$ , donde  $\tilde{q}[t]$  es la salida del PMF. La arquitectura del PMF *backpropagation* es la de la figura 17. Cada símbolo recibido se descompone en sus componentes de fase  $r_I(t)$  y cuadratura  $r_Q(t)$  ( $r(t) = r_I(t) + j r_Q(t)$ ). La salida del PMF son las componentes de fase  $\tilde{q}_I(t)$  y cuadratura  $\tilde{q}_Q(t)$  del símbolo estimado, y las entradas, una historia de los últimos símbolos recibidos  $r(t)=[r(t), r(t-T), r(t-2T), \dots]$  y de las últimas salidas obtenidas  $\tilde{q}(t)=[\tilde{q}(t), \tilde{q}(t-T), \tilde{q}(t-2T), \dots]$ . ( $T$  es el intervalo entre símbolos). Los resultados obtenidos (medidos en función de la tasa de error de bit conseguida) son mejores que los observados en un ecualizador convencional LTE, sobre todo cuando aumenta la no linealidad del canal.

Sin embargo, el PMF tiene problemas de lenta convergencia y sus soluciones son impredecibles durante el entrenamiento. Además, no hay un método claro que obtenga los parámetros del PMF, incluso cuando se conoce la función de transferencia del canal, los símbolos y la estadística del ruido. De hecho, los parámetros se calculan normalmente mediante la técnica de prueba y error. Además, el modelo inverso ignora, completamente, el hecho de que  $s(t)$  es binario. Otro inconveniente, es que un aumento del orden del ecualizador no mejora la ecualización, ya que también aumenta el ruido.

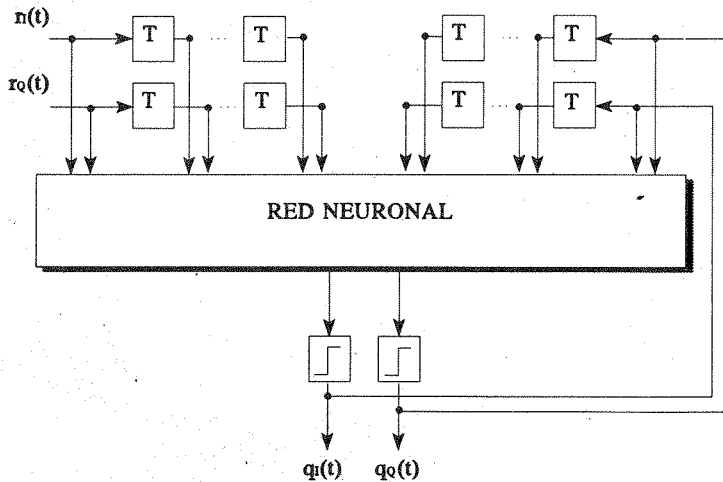


Figura 17. Estructura de un ecualizador adaptativo neuronal.

Varios autores [Chen y col., 1993; Cid y Figueiras, 1993] han propuesto otro tipo de RNAs para resolver los problemas anteriores. Se trata de las redes de funciones de base radial (FBR). Chen y col. [1993] demuestran que la solución óptima de un ecualizador adaptativo, utilizando la teoría de decisión de Bayes, tiene una estructura idéntica a la de una red FBR. Esto significa, que para un canal conocido, se pueden especificar explícitamente los parámetros de la FBR, lo cual evidencia una gran ventaja sobre los métodos anteriores.

#### 4.5.3. Predicción de tráfico multimedia en MTA

La nueva Red Digital de Servicios Integrados de Banda Ancha (RDSI-BA) soportará una gran variedad de servicios (voz, video, datos, etc.) con diferentes características de tráfico y distintos requerimientos de calidad de servicio. El modo de transmisión utilizado es el MTA (Modo de Transferencia Asíncrono), donde el tráfico multimedia se agrupa en pequeños paquetes de información, denominados celdas, que se transmiten a muy alta velocidad a través de la RDSI-BA. La caracterización del tráfico generado en la RDSI-BA es esencial en el diseño, control y desarrollo de la Red, sin embargo, es muy difícil de obtener debido a su complejidad y a que puede cambiar, conforme se vayan introduciendo nuevos servicios en la Red.

[Tarraf y col., 1993] presentan una novedosa herramienta para caracterizar y modelar el tráfico multimedia (voz, video y datos) usando RNAs. Para ello, utilizan un PMF con aprendizaje *backpropagation*, que caracteriza y predice las variaciones estadísticas del proceso de llegada de celdas, provocado por la superposición de N fuentes de video y M fuentes de voz. La exactitud de los resultados se verifica comparando varios parámetros estadísticos (el índice de dispersión de cuenta, la varianza y la autocorrelación de los datos) predichos por la RNA y medidos en la Red.

La figura 18 muestra la arquitectura del sistema utilizado, que se define mediante la ecuación:

$$\mathbf{b}(i+m)=f(\mathbf{b}(i),\mathbf{W})$$

donde  $f$  es la función de transferencia del PMF,  $\mathbf{W}$  la matriz de pesos del PMF y  $\mathbf{b}(i)$ , un vector de la forma  $\mathbf{b}(i)=[b(i),b(i-1),\dots,b(i-m-1)]^T$ , donde  $b(i)$  son el número de celdas por segundo que llegan en el instante  $i$ . El PMF estima un vector para los próximos  $m$  intervalos de la forma:  $\mathbf{b}(i+m)=[b(i+m),b(i+m-1),\dots,b(i+1)]^T$ , donde  $m$  representa el número neuronas de las capas de entrada y salida. Los resultados que se obtiene son satisfactorios y demuestran el potencial de las RNAs para la caracterización de complejos procesos matemáticos.

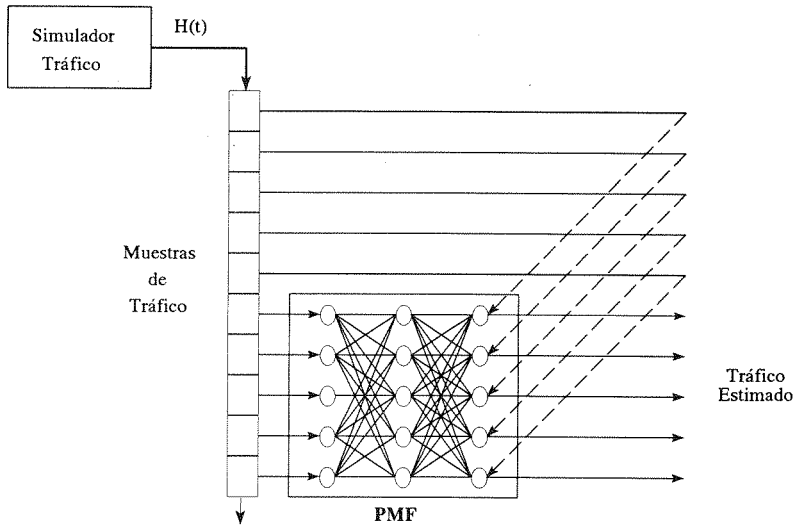


Figura 18. Estructura de un predictor neuronal de tráfico MTA.

## 5. CONTROL ADAPTATIVO

### 5.1. Introducción

El control adaptativo es necesario en muchos sistemas de comunicaciones. Por ejemplo, las características de los canales de transmisión o la carga de tráfico de una red de comunicaciones pueden variar con el tiempo. Se hace, por tanto, necesario un control que se adapte continuamente a los posibles cambios en el sistema, con el objeto de mantener ciertos requerimientos de calidad.

Las RNAs pueden resolver muchos de los problemas que se plantean. El PMF *backpropagation* se ha utilizado bastante en el control multidimensional dinámico de plantas [Schiffmann y Gefers, 1993]. Las redes de FBR también se han revelado como una buena solución para aprendizaje rápido en el caso de pocas variables de entrada [Narendra, 1992].

### 5.2. Tipos de control adaptativo neuronal

En la figura 19 se presentan algunas arquitecturas básicas de control adaptativo con RNAs. La figura 19a presenta un típico esquema de control directo basado en

RNA. La salida del sistema  $y(t)$  se resta de la entrada  $r(t)$  para obtener una señal de error  $e(t)$ , que se aplica al controlador. El controlador intenta obtener una salida que haga  $e(t) = 0$ . El control se conseguirá cuando  $r(t) \approx y(t)$  y, por supuesto, la función de transferencia entrada/salida sea estable. Sin embargo, este control exige conocer la matriz jacobiana del sistema desconocido (la derivada parcial de las salidas del sistema con respecto a sus entradas), con el objeto de retropropagar el error  $e(t)$  a través del sistema y ajustar los pesos de la RNA. Si no se conoce la matriz jacobiana, se puede identificar el sistema mediante otra RNA (modelo directo) (ver figura 19b). Una vez, esta segunda RNA, sea capaz de emular al sistema, es decir, genere las mismas salidas que el sistema para las mismas entradas, podrá retropropagar el error  $e(t)$  hacia la RNA de control, para que ésta ajuste sus pesos adecuadamente, es lo que se denomina control indirecto. Finalmente, otra posible solución hace uso del modelo inverso, donde la RNA aprende la función de transferencia inversa del sistema y se aplica, a posteriori, como RNA de un control directo.

### 5.3. Aplicaciones

Todas las aplicaciones que se presentan se hacen sobre la futura Red Digital de Servicios Integrados de Banda Ancha (RDSI\_BA). Esta red [CCITT, 1990] soportará servicios de voz, video y datos, con características de tráfico y requerimientos de calidad de servicio muy variados. Para satisfacer a todos estos servicios se usará como técnica de transporte el Modo de Transferencia Asíncrono (MTA), que tolera multiplexación estadística, la cual permite compartir los recursos de la Red. Sin embargo, esta mejora, en la utilización de la Red, se ve penalizada con una probabilidad de congestión no nula. Esto obliga al estudio y desarrollo de nuevos controles de congestión [Hughes y Bradlow, 1991], ya que los controles tradicionales no son eficaces, debido a las altas velocidades que se manejan en Banda Ancha, al desconocimiento, en muchos casos, de las características del tráfico generado y a la dificultad en obtener controles sencillos y eficientes a la vez.

A continuación, se presentan tres aplicaciones que ilustran la viabilidad del uso de RNAs en este campo: Control de admisión de llamadas, control de tráfico y generación de modelos de tráfico. También se hace referencia a las posibilidades de las RNAs de implementar un control distribuido de toda una red de Banda-Ancha.

#### 5.3.1. Control de admisión de llamadas en MTA

El control de admisión de llamadas en MTA acepta o rechaza una nueva llamada en función de las características de la nueva llamada solicitada, de la carga en curso de la Red y de los requerimientos de calidad de servicio de todas las llamadas conectadas y solicitadas.

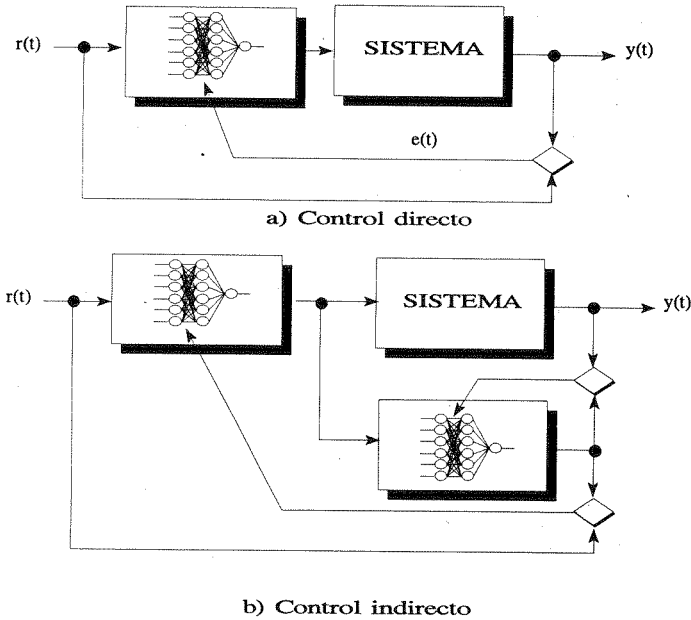


Figura 19. Tipos de control adaptativo.

En la RDSI-BA se han desarrollado muchos métodos analíticos para obtener el control [Murase y col., 1991; Mase y Shioda, 1991], pero todavía no se ha llegado a soluciones definitivas, sobre todo en el caso de tráfico heterogéneo, donde el coste computacional es enorme y obliga a realizar muchas suposiciones. Además, estos métodos no suelen ser adaptativos, característica importante en MTA ya que se prevé la continua introducción de nuevos servicios después del diseño e instalación de la Red. También, hay métodos basados en la monitorización de tráfico que intentan resolver algunos de los problemas anteriores, encontrándose entre ellos los métodos basados en el aprendizaje mediante RNAs.

Han sido varios los autores que han investigado el control de admisión de llamadas para la red MTA mediante RNAs [Hiramatsu, 1990; Morris, 1991; Bolla y col., 1992; Tran-Gia y Groppo, 1992; Nordstrom, 1993; Díaz y col., 1993].

Hiramatsu [1990] hace uso del modelo directo. Para ello, define una función  $r=f(n)$ , donde  $r$  es la tasa de pérdidas de un nodo MTA y  $n$  el número de llamadas conectadas. Hiramatsu utiliza un PMF *backpropagation* para identificar esta función. Una vez la red ha aprendido, el control es inmediato: Si la tasa de pérdidas está por encima de cierto umbral, la llamada se rechaza, en caso contrario se acepta (ver figura 20). El control se realiza en tiempo real, es decir la fase de aprendizaje y de control

se realizan en paralelo. Esto trae como consecuencia, el que no se disponga, a priori, de todos los posibles patrones de entrenamiento y que el control afecte al conjunto de patrones que son presentados a la RNA. Además, la RNA debe adaptarse a nuevas condiciones de tráfico. La solución adoptada por Hiramatsu para obtener un buen aprendizaje consiste en un método de selección de patrones de aprendizaje (basado en unas grandes tablas de almacenamiento) que independiza la fase de aprendizaje de la fase de control. Hiramatsu demuestra que la RNA es capaz de aprender características de tráfico, tales como la velocidad media y el *burstiness* (relación entre la velocidad de pico y la velocidad media de las fuentes de velocidad variable), y de adaptarse a nuevas condiciones de tráfico.

Morris [1991] realizó un estudio más riguroso mediante la simulación de un nodo MTA con fuentes de tráfico heterogéneas. Su método permite controlar el retardo medio en la cola del multiplexador. Es adaptativo frente a cambios de tráfico y permite manejar fácilmente distintos requerimientos para cada clase de servicio. También utiliza una tabla de patrones de entrenamiento y un PMF, aunque con un algoritmo de aprendizaje más elaborado que evita la caída en mínimos locales. Morris preprocesa los datos medidos en la cola del multiplexador antes de generar los patrones de aprendizaje, con el objeto de reducir el tiempo de convergencia.

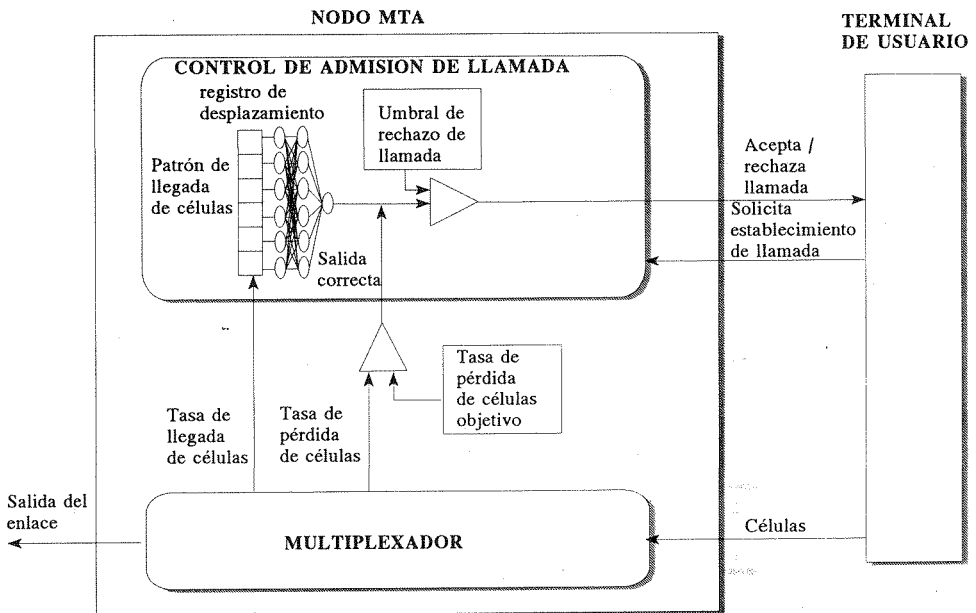


Figura 20. Control neuronal monoservicio de admisión de llamadas.

Díaz y col. [1993] proponen un controlador híbrido entre las dos aproximaciones anteriores. Este método divide el espacio de patrones en zonas, donde cada zona es una tabla de aprendizaje. Las ventaja fundamental obtenida es que sólo se necesita un conjunto de entrenamiento muy pequeño (tablas muy pequeñas), sin perder la propiedad de independencia entre la fase de aprendizaje y la de control. En la figura 21 se puede ver un diagrama de bloques del controlador neuronal utilizado y de su ubicación dentro del modelo de simulación. El controlador neuronal consta de: una tabla de patrones donde se guardan los patrones de aprendizaje con los que se entrenará a la RNA, un generador de patrones de entrada a la RNA y la propia RNA. El funcionamiento del sistema es el siguiente: Cada vez que hay un cambio en el número de llamadas conectadas, se almacena en la tabla de patrones la combinación de llamadas conectadas anterior y su tasa de calidad de servicio medida (pérdidas, retardo, etc.). Por otro lado, cada vez que se solicita una llamada, el generador de patrones obtiene un patrón suma de llamadas conectadas y solicitadas y lo aplica a la RNA. La salida de la RNA es una predicción de la tasa de calidad de servicio que se obtendría en el caso de aceptar la nueva llamada. Esta salida se compara con una tasa de calidad de servicio objetiva y se decide si se acepta o se rechaza la llamada. El aprendizaje es completamente independiente del proceso anterior, y se realiza cada cierto tiempo, durante todo el período de vida del controlador. El controlador toma aleatoriamente un patrón de aprendizaje de la tabla de patrones y lo presenta a la RNA. La RNA modificará sus pesos internos con el objeto de aprender dicho patrón.

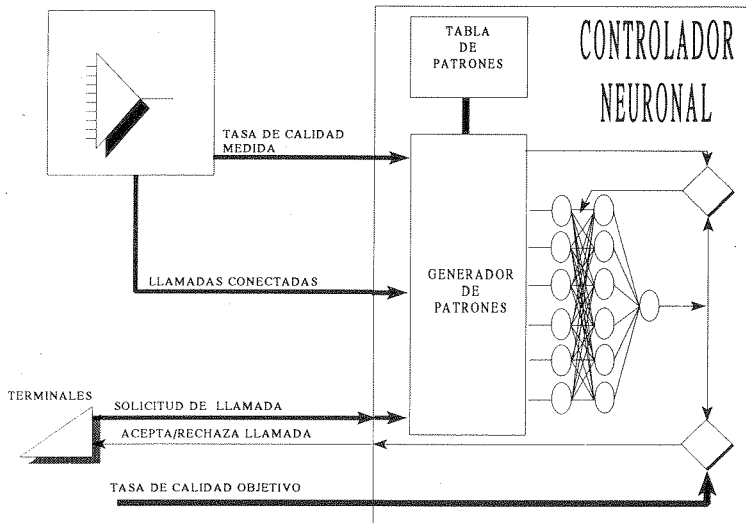


Figura 21. Control neuronal multiservicio de admisión de llamadas.

La figura 22 presenta los resultados obtenidos para dos clases de servicio. Las coordenadas representan el número de llamadas conectadas de la clase 1 en función del número de llamadas conectadas de la clase 2. Los valores están normalizados (con respecto a la capacidad del enlace y la velocidad media de cada clase). Las dos curvas que se presentan son la frontera de admisión óptima (obtenida por simulación) y la estimada (obtenida por la RNA). El controlador necesitó 100 segundos para aprender la frontera correcta. A efectos de comparación, también se presentan las fronteras de admisión de dos controladores analíticos (ancho de banda de pico y ancho de banda virtual). Como se ve, sus regiones de admisión son muy pequeñas, debido a que, al no conocer muy bien el funcionamiento del sistema, adoptan un comportamiento muy conservativo que asegura la ausencia de pérdidas (aunque incrementa la tasa de rechazo de llamadas). Hay que destacar, que la RNA sólo ha podido ser entrenada con parte de todas las posibles combinaciones de llamadas conectadas y que sin embargo ha sido capaz de obtener una buena generalización.

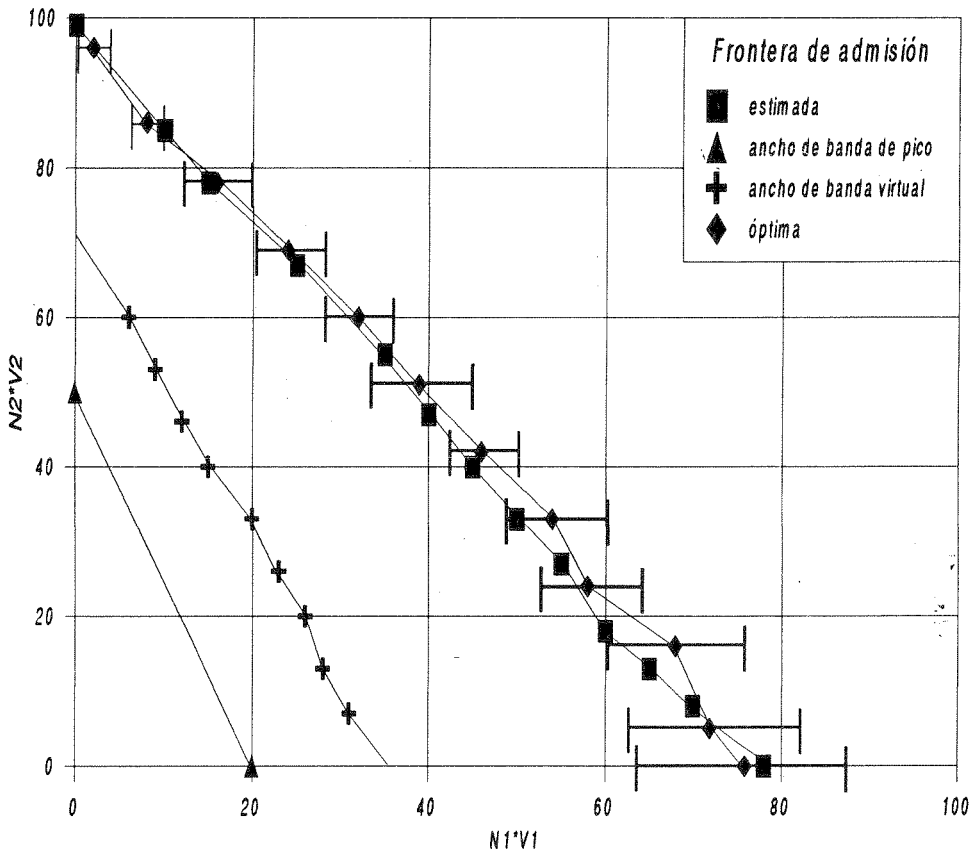


Figura 22. Región de admisión del control neuronal multiservicio de admisión de llamadas.

### 5.3.2. Control de tráfico en MTA

El control de tráfico en MTA intenta regular dinámicamente la carga de tráfico dentro de la Red para garantizar ciertos requerimientos de calidad de servicio (pérdidas, retardo, etc.). Los métodos convencionales hacen algunas suposiciones sobre la naturaleza de las fuentes de tráfico y su superposición, obteniendo modelos matemáticos que permiten predecir la carga de tráfico. Sin embargo, los métodos más elaborados no son viables para tráfico multiservicio y decisiones en tiempo real debido a su complejidad y tiempo de cálculo, mientras que los más simples no aprovechan eficientemente las ventajas de la multiplexación estadística. Chen y Leslie [1992] proponen un control neuronal adaptativo a la entrada de la red MTA (ver figura 23) que utiliza control indirecto y una estructura predictiva. La RNA de control calcula qué porción  $\eta(n)$  ( $0 < \eta(n) \leq 1$ ) del tráfico  $\lambda'(n)$ , que hay a la entrada de la Red en el instante  $n$ , puede admitirse, sin superar cierta tasa de calidad de servicio  $d'(n)$ . El tráfico admitido  $\lambda(n)$  vale  $\eta(n) \cdot \lambda'(n)$ . La RNA de emulación, con estructura ARMA, identifica al sistema y aprende a estimar, monitorizando la Red, la tasa de calidad de servicio  $d'(n)$ , en función de las cargas de tráfico admitido  $\lambda(t)$  y tasas de calidad asociadas  $d(t)$  anteriores. La función de error de la RNA de control vale  $e_c(n) = d'(n) - d(n)$ , así que el algoritmo *backpropagation* no puede ser aplicado directamente ya que no se conoce la matriz jacobiana de la Red MTA. La RNA de emulación solventa esta dificultad usando un modelado directo de la Red MTA. Esto permitirá retropropagar el error  $e_c(t)$  a través de la RNA de emulación y ajustar los pesos de la RNA de control.

Hay que destacar, que no se requiere un conocimiento previo de las características del tráfico y que mediante la monitorización de la carga de tráfico y sus correspondientes tasas de calidad asociadas, el control neuronal es capaz de aprender el comportamiento del sistema dinámico y, por lo tanto, de reaccionar correctamente frente a cualquier cambio que se produzca.

### 5.3.3. Generación de modelos de tráfico

Una aplicación original del control indirecto es la propuesta por [Jurado y col., 1993] para la generación de modelos de tráfico en RDSI-BA con técnica de transporte MTA. Como ya se mencionó, la obtención de modelos de tráfico para MTA presenta dificultades en el caso de ambientes multiservicio y decisiones en tiempo real debido a su complejidad y tiempo de cálculo. En este trabajo, se demuestra cómo complejos modelos matemáticos pueden ser sustituidos por RNAs para determinar los parámetros de un modelo de superposición de muchas fuentes heterogéneas de tráfico MTA. Con el sistema propuesto, se pueden obtener los parámetros de modelos de tráfico, en función del número de fuentes conectadas de cada clase (voz, video o datos).

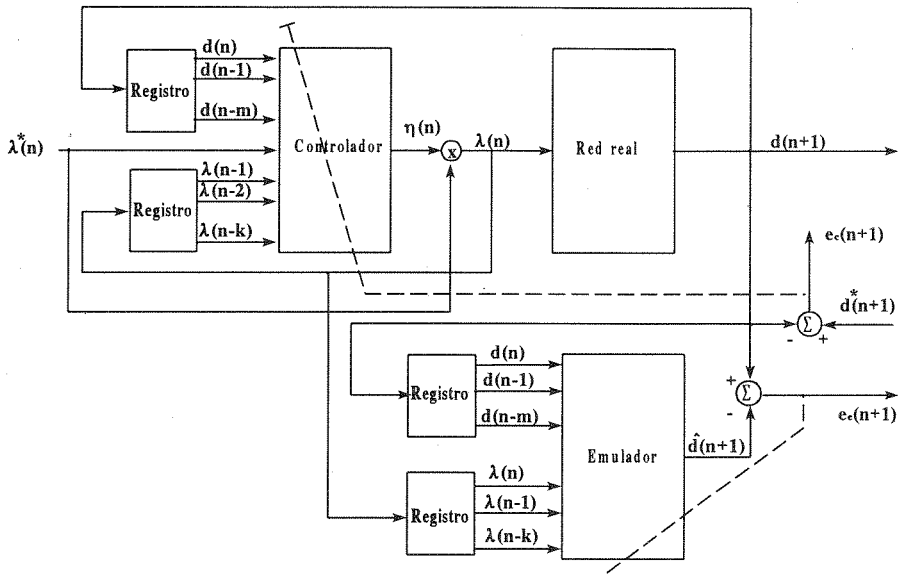


Figura 23. Control neuronal de tráfico MTA.

La arquitectura de este generador de modelos se puede ver en la figura 24, en la que se distinguen 4 módulos. El módulo **S** es un enlace de la red MTA, donde, en función de las fuentes conectadas, se mide un conjunto de parámetros estadísticos (vector  $\mathbf{p}^{[S]}$ ) que definen el tráfico en curso. El módulo **C**, denominado Red Controladora, es un PMF que estima los parámetros del modelo, en función de las fuentes conectadas de cada clase. Este módulo calcula un vector  $\mathbf{k} = \mathbf{F}(\mathbf{n}, \mathbf{W})$ , siendo  $\mathbf{n} = (n_1, n_2, \dots, n_N)$ , donde  $n_i$  es el número de fuentes de la clase  $i$  ( $i=1, 2, \dots, N$ ), y  $\mathbf{W}$  la matriz de pesos. Desafortunadamente se desconocen las componentes del vector  $\mathbf{k}$ , y por tanto es imposible la realización de la fase de aprendizaje de la Red Controladora. Para solucionar este problema se hace uso del control indirecto, introduciendo dos nuevos módulos (**K** y **E**). El módulo **K** calcula los parámetros estadísticos del modelo, en función de los parámetros del modelo, definidos por el vector  $\mathbf{k}$ , obteniéndose el vector  $\mathbf{p}^{[K]}$ . El módulo **E** es otro PMF que calcula también los parámetros estadísticos del modelo. Este último módulo dará a su salida el vector  $\mathbf{p}^{[E]}$ , con los mismos parámetros estadísticos del sistema real **S**. La Red Emuladora aprenderá a identificar el módulo **K**, estimando un vector  $\mathbf{p}^{[E]} = \mathbf{G}(\mathbf{k}, \mathbf{W})$ , donde  $\mathbf{V}$  es la matriz de pesos. Este módulo ajustará sus pesos para minimizar el error  $e^{[E]} = 1/2(\mathbf{p}^{[K]} - \mathbf{p}^{[E]})^T(\mathbf{p}^{[K]} - \mathbf{p}^{[E]})$ , usando el gradiente:

$$\frac{\partial e^{[E]}}{\partial \mathbf{V}} = -(\mathbf{p}^{[K]} - \mathbf{p}^{[E]}) \frac{\partial \mathbf{p}^{[E]}}{\partial \mathbf{V}}$$

La red Emuladora es necesaria para retropropagar el error hacia la Red Controladora. Este módulo debe minimizar el error  $e^{[C]} = 1/2(\mathbf{p}^{[S]} - \mathbf{p}^{[K]})(\mathbf{p}^{[S]} - \mathbf{p}^{[K]})^T$ , por medio del gradiente:

$$\frac{\partial e^{[C]}}{\partial \mathbf{W}} = -(\mathbf{p}^{[S]} - \mathbf{p}^{[K]}) \frac{\partial \mathbf{p}^{[K]}}{\partial \mathbf{W}} = -(\mathbf{p}^{[S]} - \mathbf{p}^{[K]}) \frac{\partial \mathbf{p}^{[E]}}{\partial \mathbf{W}}$$

donde  $\partial \mathbf{p}^{[K]}/\partial \mathbf{W}$  es sustituido por  $\partial \mathbf{p}^{[E]}/\partial \mathbf{W}$  ya que  $\mathbf{p}^{[E]}$  tiende asintóticamente hacia  $\mathbf{p}^{[K]}$ . Cuando el proceso de aprendizaje ha terminado, la Red Controladora calcula el vector  $\mathbf{k}$  en función del número de fuentes conectadas (vector  $\mathbf{n}$ ). Este vector identifica un modelo que tiene las mismas características estadísticas seleccionadas que produce el tráfico en una red MTA real.

### 5.3.4. Control distribuido mediante RNAs

La propiedad adaptativa de las RNAs puede ser muy útil en el desarrollo de un control distribuido para la RDSI-BA. [Hiramatsu, 1990; Keskiner y Pandya, 1991] esperan que las RNAs con aprendizaje *backpropagation* puedan aplicarse a todos los niveles de control en la RDSI-BA, incluso a sistemas expertos para el diseño de la Red. La figura 25 presenta un esquema de esta idea.

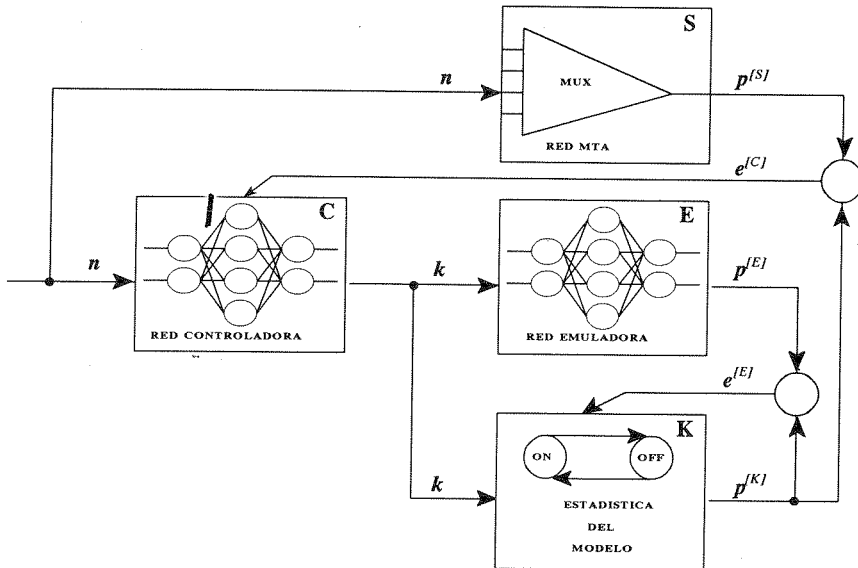


Figura 24. Generador neuronal de modelos de tráfico MTA.

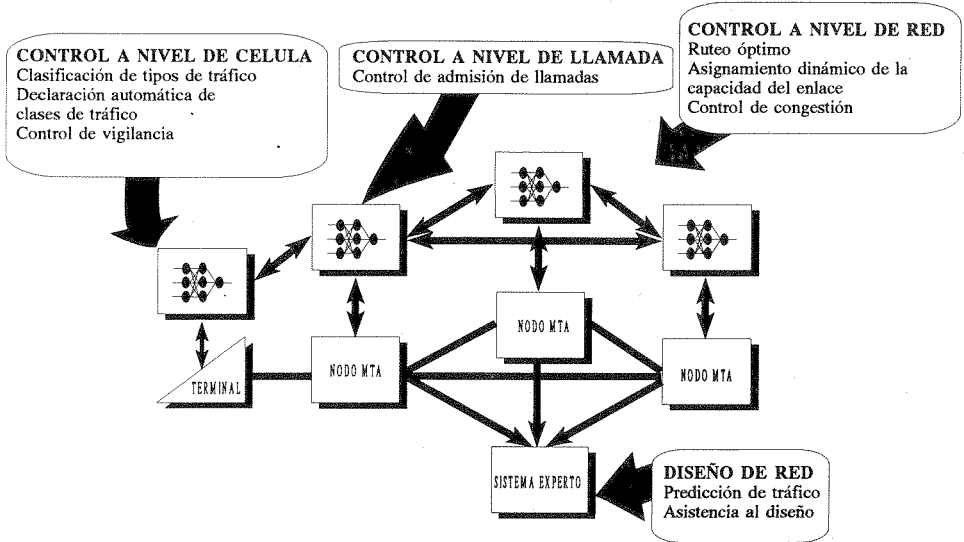


Figura 25. Control neuronal distribuido en una Red MTA.

Neves y col. [1993] han propuesto una función de calidad de servicio global de toda la Red MTA que permite establecer una estrategia para el control de admisión y ruteo de las conexiones. Su función de calidad facilita la partición dinámica del ancho de banda entre clases, el equilibrio entre probabilidad de bloqueo de llamadas de cada clase y el control de diferentes requerimientos de calidad de servicio. Neves y col., demuestran que el control es muy flexible y que se puede adaptar a las necesidades específicas de cada caso.

## 6. CONCLUSIONES

Las RNAs presentan unas propiedades nada desdeñables en el diseño de nuevos sistemas de telecomunicaciones y en la mejora de los sistemas actuales. La característica fundamental de aprendizaje adaptativo de las RNAs les permite controlar e identificar sistemas casi o completamente desconocidos, simplemente observando su funcionamiento. Su elevada velocidad de computación, debido al paralelismo masivo de su implementación *hardware*, es imprescindible en el desarrollo de sistemas complejos en tiempo real de muy alta velocidad. Su capacidad de generalización es particularmente útil en ambientes dinámicos de control donde las observaciones pueden ser incompletas. Finalmente, su característica innata de tolerancia a fallos ofrece una gran ventaja sobre muchas soluciones convencionales.

La flexibilidad y alta velocidad de la RDSI\_BA exigirá complicados mecanismos de control y gestión de tráfico si se quieren optimizar los recursos de la red. La RDSI\_BA trabajará en un entorno multidimensional, ya que tendrá que manejar muchos tipos de servicios, cada uno de ellos con diferentes características de tráfico y distintos requerimientos de calidad de servicio, tendrá que ser adaptativa si quiere ser una red universal que englobe todo tipo de servicios presentes y futuros, requerirá control en tiempo real y necesitará manejar fuentes de tráfico de difícil modelado y parcialmente conocidas. Todos estos requerimientos pueden ser satisfechos mediante RNAs y ya hay empresas y compañías telefónicas que empiezan a interesarse por esta tecnología. Por otro lado, el aprovechamiento de la red telefónica conmutada está conduciendo al diseño y desarrollo de *modems* de gran consumo y muy alta velocidad que necesitan ecualizadores y canceladores de eco cada vez más potentes. Las RNAs son capaces de obtener soluciones óptimas, a diferencia de técnicas convencionales que tienen dificultades cuando aumenta la velocidad y aparecen fenómenos no lineales.

No obstante, antes de que las aplicaciones propuestas puedan llevarse a cabo con éxito, las RNAs deberán resolver sus problemas típicos de lenta convergencia, mínimos locales, algoritmos de dimensionamiento, dificultad en la implementación física, etc. En este sentido, hay que decir que cada vez son más los investigadores que están prestando atención a las posibilidades que ofrecen las RNAs en el campo de las telecomunicaciones y confiamos que esto ayudará a acelerar la solución de todos los problemas que se vienen planteando.

## REFERENCIAS

- Angeniol, B., Croix, V.G., y Texier, J. Le, «Self-organizing feature Maps and the Travelling Salesman Problem», *Neural Networks*, 1, (1988), 11-19.
- Ansari, N. y Chen Y., «A neural network model to configure maps for satellite communications network», *Proceedings of GLOBECOM '90*, Vol. 2, (1990), 1042-1046.
- Benes, V.E., *Mathematical Theory of Connecting Networks and Telephone Traffic*, McGraw-Hill, New York, 1965.
- Bijjani, B. y Das, P.K., «Rejection of Narrowband Interference in PN Spread-Spectrum Systems using Neural Networks», *Proceedings of GLOBECOM '90*, Vol. 2, (1990), 1037-1041.
- Bolla, R., Davoli, F., Maryni, P. y Parisini, T., «A Neural Strategy for Optimal Multiplexing of Circuit-and Packet-switched Traffic», *Proceedings of GLOBECOM '92*, Vol. 3, (1992), 1324-1330.
- Brown, T.X. y Liu, K.H., «Neural Network Design of a Banyan Network Controller», *IEEE J. Select. Areas Commun.*, 8, (1990), 1428-1438.
- Brown, T.X., «Neural Networks for Switching», *IEEE Commun. Mag.*, 27, (1989), 72-81.
- Brown, T.X., «Neural Networks for Adaptive Equalization», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 27-33.
- Cavalieri, S., Di Stefano, A., y Mirabella, O., «Hopfield Neural Network for Routing», *Proc. del International Workshop on Artificial Neural Networks IWANN'93*, Springer Verlag, (1993), 583-593.
- CCITT, «B-ISDN MTA Functional Characteristics», *Study Group XVIII*, draft Rec. I.150, Génova, (1990).
- Chen, S., Gibson, G.J. y Cowan, C.F.N., «Adaptive Channel equalization using a polynomial-perceptron structure», *IEE Proceedings-I*, Vol. 137, N. 5, (1990), 257-264.
- Chen, X. y Leslie, I.M., «Neural Adaptive Congestion Control for Broadband MTA Networks», *IEE Proceedings-I*, Vol. 139, N.3, (1992), 233-244.
- Chen, S., Mulgrew, B. y Grant, P.M., «A Clustering Technique for Digital Communications Channel Equalization Using Radial Basis Function Networks», *IEEE Transactions On Neural Networks*, Vol. 4, N. 4, (1993), 570- 579.

- Cid-Sueiro, J. y Figueiras-Vidal, A.R., «Improving Conventional Equalizers with Neural Networks», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 20-26.
- Connors, J.T., «Prediction of Acces Line Growth», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 232-238.
- Dali, Y., «An echo canceller based on the multilayer LTBP neural network», *Proceedings of GLOBECOM '92*, Vol. 3, (1992), 1298-1302.
- Díaz Estrella, A., Jurado Díez, A., García Lopera, A. y Sandoval, F., «Control de admisión de llamadas multiservicio de un nodo de acceso ATM mediante redes neuronales», *TELECOM '93*, Madrid, (1993), 311-319.
- Funahashi, K., «On the approximate realization of continuous mappings by neural networks», *Neural Networks*, Vol. 2, (1989), 183-192.
- Funabiki, N. y Takefuji, Y., «A Neural Network Parallel Algorithm for Channel Assignment Problems In Cellular Radio Networks», *IEEE Trans. on Vehicular Technology*, 41, 4, (1992), 430-437.
- Funabiki, N. y Takefuji, Y., «A Parallel Algorithm for Broadcast Scheduling Problems in Packet Radio Networks», *IEEE Trans. on Communications*, 41, 6, (1993), 828-831.
- García Lopera, A., García Oller, F., Diaz Estrella, A. y Sandoval, F., «Neural Network Routing Controller for Communication Parallel Multistage Interconnection Networks», *Proc. del International Workshop on Artificial Neural Networks IWANN'93*, Springer Verlag, 1993, 593-598.
- Goke, L.R. y Lipovski, G.J., «Banyan Networks for Partitioning Multiprocessor Systems», *Proc. 1st Annu. Int. Symp. Comput. Architecture*, (1973), 21-28.
- Goudreau, M.W. y Giles, C.L., «Neural Network Routing for Multiple Stage Interconnection Networks», *Proc. IJCNN 91*, Vol II, (1991), A-885.
- Hérault, L. y Niez, J.J., «Neural Networks and combinatorial optimization, a study of NP-complete graph problems», en Gelenbe, E. (Ed.), *Neural Networks: Advances and Applications*, Elsevier, Amsterdam, 1991, 165-213.
- Hiramatsu, A., «ATM Communications network Control by Neural Networks», *IEEE Transactions on Neural Networks*, Vol. 1, N. 1, (1990), 122-130.

- Hopfield, J.J. y Tank, D.W., «Neural Computation of Decisions in optimization problems», *Biolog. Cybernetics*, 52, (1985), 141-152.
- Hornik, K., Stinchcomb, M. y White, H., «Multilayer Feedforward Networks are Universal Approximators», *Neural networks*, Vol.2, (1989), 359-366.
- Hughes, D.A. y Bradlow, H.S., «Congestion Control in a ATM Network», en Jensen, A. y Iversen, V.B. (Eds.), *Teletraffic and Data traffic*, Elsevier Science Publishers, 1991, 835-840.
- Jurado Díez, A., Díaz Estrella, A. y Sandoval, F., «Estudio de modelado de tráfico MTA mediante redes neuronales», *URSI'93*, (1993), 69-73.
- Kailath, T., *Lectures on Wiener and Kalman filtering*, Springer Verlag, NewYork, 1981.
- Keskiner, K. y Pandya, S., «Neuro-controllers in Communications», *Proceedings of IJCNN'91*, Vol. 2, (1991), 1005.
- Kil, R.M. y Choi, J.Y., «Time-Series Prediction Based on Global and Local Estimation Models», *Proceedings of WCCN'93*, Vol. 4, (1993), 617-621.
- Kohonen, T., *Self-Organization and Associative Memory*, Springer-Verlag, Berlín, 1989.
- Littmann, Michael y Boyan, Justin, «A Distributed Reinforcement Learning Scheme for Network Routing», *Proc. of the Int. Workshop on Applications of Neural Networks to Telecommunications*, Princeton, USA, (1993), 45-62
- Majani, E., Erlanson, R. y Abu-Mostafa, Y., «On the K-Winners-Take-All Network», en Touretzky, D.S. (Ed.), *Advances in Neural Information Processing Systems 1*, Morgan Kauffman, 1989, 634-642.
- Marrakchi, A.M. y Troudet, T., «A Neural Net Arbitrator for Larger Crossbar Packet Switches», *IEEE Trans. on Circ. and Sys.*, 36, 7, (1989), 1039-1041.
- Martínez, T. y Rudolph, G.L., «A Learning Model for Adaptive Network Routing», *Proc. of the Int. Workshop on Applications of Neural Networks to Telecommunications*, Princeton, USA, (1993), 183-188.
- Mase, K. y Shioda, S., «Real time Network Management for ATM Networks», en Cohen, J.W. y Pack, C.D. (Eds.), *Queueing, performance and control in ATM*, Elsevier Science Publishers, 1991, 129-136.
- Meyer, M. y Gerd, Pfeiffer, «Multilayer Perceptron bases Equalizers applied to Nonlinear Channels», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.),

*Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 188-195.

Minzer, S.E., «Broadband ISDN and Asynchronous Transfer Mode», *IEEE Communications Mag.*, 27, (1989), 127-138.

Morris, R.J.T., «Prospects for Neural Networks in Broadband Networks Resources Management», en Jensen, A. y Iversen, V.B. (Eds.), *Teletraffic and Data traffic*, Elsevier Science Publishers, 1991, 335-340.

Murase, T., Suzuki, H., Sato, S. y Takeuchi, T., «A Call Admission Control Scheme for ATM Networks Using a Simple Quality Estimate», *IEEE Journal on Selected Areas in Communications*, Vol. 9, N. 9, (1991), 1461-1470.

Nakano, K., Sengoku, M., Shinoda, S., Yamaguchi, Y. y Abe, T., «Channel Assignment in Cellular Mobile Communication Systems using Neural Networks», *Communications Systems: Toward Global Integration*, (1991), 531-534.

Narendra, K.S., «Adaptive Control of Dynamical Systems using Neural Networks», en White, D.A. y Sofge, D.A. (Eds.), *Handbook of Intelligent Control*, Van Nostrand Reinhold, New York, 1992, 141-185.

Neves, J.E., De Almeida, L.B. y Leitao, M.J., «ATM Call Control by Neural Networks», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 210-217.

Nordstrom, E., «A Hibrid Admission Control Scheme for Broadband ATM Traffic», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 77-83.

Proakis, J.K., *Digital Communications*, MacGraw-Hill International Editions, 1989.

Rauch, H.E. y Winarske, T., «Neural Networks for Routing Communication Traffic», *IEEE Control Systems Magazine*, (1988), 26-30.

Schiffmann, W.H. y Gefers, H.W., «Adaptative Control of Dynamic Systems by Back-Propagation Networks», *Neural Networks*, Vol. 6, (1993), 517-524.

Shyy, D. y Lea, C.T., « $\log_2(N,m,p)$  strictly Nonblocking Networks», *IEEE Transactions on Comm.*, 39, 10, (1991), 1502-1509.

Simeone, B., *Combinatorial Optimization*, Springer-Verlag, New-York, 1989.

- Sofge, D.A. y White, D.A., «Applied learning: Optimal Control for manufacturing», en White, D.A. y Sofge, D.A. (Eds.), *Handbook of Intelligent Control*, Van Nostrand Reinhold, New York, 1992, 259-282.
- Takefuji, Yoshiyasu, *Neural Network Parallel Computing*, Kluwer Academic, Norwell, 1993, USA, 180-195.
- Tarraf, A.A., Habib, I.W. y Saadawi, T.N., «Neural Networks for ATM Multimedia Traffic Prediction», en Alspector, J., Goodman, R. y Brown, T.X. (Eds.), *Applications of Neural Networks to Telecommunications*, Lawrence Erlbaum Associates Publishers, 1993, 85-91.
- Tran-Gia, P. y Groppo, O., «Performance of a Neural Net used as Admission Controller in ATM Systems», *Proceedings of GLOBECOM '92*, Vol. 3, (1992), 1303-1309.
- Widrow, B. y Stearns, S.D., *Adaptive Signal Processing*, Prentice-Hall, Inc., 1985.
- Weruaga Prieto, L. y Figueiras Vidal, A.R., «Redes Neuronales en Cancelación de Ecos», *URSI '92*, Málaga, Vol. 2, (1992), 989-993.
- Zhang, L. y Thomopoulos, S.C.A., «Neural Network Implementation of the Shortest Path Algorithm for Traffic Routing in Communication Networks», *Poster presentation at the 1<sup>st</sup> International Joint Conference on Neural Networks*, Washington, (1989).
- Zhang, Chen-Xiong, «Optimal Traffic Routing Using Self-Organization Principle», *Proc. of the Int. Workshop on Applications of Neural Networks to Telecommunications*, Princeton, USA, (1993), 225-232.