Nivel de enlace y redes de área local

Eduard Lara Ochoa
Xavier Vilajosana Guillén
René Serral i Gracià
Miquel Font Rosselló

PID_00171192

Universitat Oberta de Catalunya
www.uoc.edu

Índice

Introducción

El nivel de enlace ha tenido un papel destacado a lo largo de la historia de las redes de ordenadores. Es un nivel que, a diferencia de otros niveles de la torre OSI, se ha tenido en cuenta en todas las arquitecturas de redes propietarias creadas durante los años sesenta y setenta, y se han realizado numerosos diseños de sus protocolos. Además, debido a su posición estratégica, está implementado en todos y cada uno de los nodos de una red, del mismo modo que el nivel de red.

Según la orientación clásica, el nivel de enlace permite establecer una conexión directa entre dos entidades para transmitir información: son los denominados enlaces punto a punto. No obstante, veremos que el nivel de enlace también permite establecer conexiones en medios de difusión, en los que participan más de dos entidades, hecho que ocurre generalmente en las redes de área local.

Por tanto, encontraremos el nivel de enlace implicado en diferentes contextos:

- En las conexiones locales de un ordenador con un periférico (por ejemplo, impresora).
- En las redes de área local.
- En las redes de acceso a WAN.
- En las redes de transporte WAN.

Este módulo sobre el nivel de enlace se ha estructurado de la siguiente manera:

1) En un primer gran apartado se abordarán las características generales del nivel de enlace. Se analizarán las funcionalidades del nivel de enlace agrupadas en 5 grandes bloques:

a) Gestión de las tramas, que engloba funciones como: entramado de la trama, sincronización, transparencia, numeración, multiplexación y direccionamiento.

b) Gestión del enlace, en el que se diferenciará entre los servicios de la capa de enlace orientados a conexión y los no orientados a conexión.

c) Control de flujo, en el que se estudiarán diferentes algoritmos desarrollados para compatibilizar la velocidad de recepción de las tramas con la velocidad de procesamiento en el receptor, entre los que podemos destacar: los mecanismos XON/XOFF y RTS/CTS, el protocolo *stop & wait* y la ventana deslizante.

d) Control de errores. Veremos que el nivel de enlace es el encargado de intentar resolver los errores de transmisión que introduce la utilización de canales físicos no perfectos. Las técnicas de control de errores se presentaran divididas en 3 categorías: la detección de errores, la corrección de errores y la retransmisión de tramas erróneamente recibidas.

e) Control de acceso al medio. Esta funcionalidad se encuentra ubicada dentro del siguiente gran apartado, dado que toma relevancia en las redes de área local.

Veremos que todas estas funcionalidades normalmente se encuentran implementadas en los dispositivos de nivel de enlace, también denominados tarjetas de red o NIC.

2) El segundo gran apartado de este módulo se centra en el estudio del nivel de enlace aplicado al contexto de las redes de área local. Históricamente, las redes de área local se han basado en medios compartidos (medios de difusión). Para asegurar un acceso equitativo de todos los terminales que comparten el medio, se ha diseñado una serie técnicas o protocolos de acceso al medio. Algunos de los que trataremos son: Aloha, Aloha ranurado, CSMA, CSMA/CD y CSMA/CA. También se estudiarán las tecnologías más utilizadas en las redes de área local, tanto en medios cableados, en los que la tecnología dominante es Ethernet IEEE 802.3, como en medios inalámbricos, en los que Wireless LAN IEEE 802.11 es el estándar elegido. Por último, se establecerá la clasificación de las tecnologías inalámbricas, según su extensión, y se incluirá el estudio de Wimax (IEEE 802.16).

1. Características generales del nivel de enlace

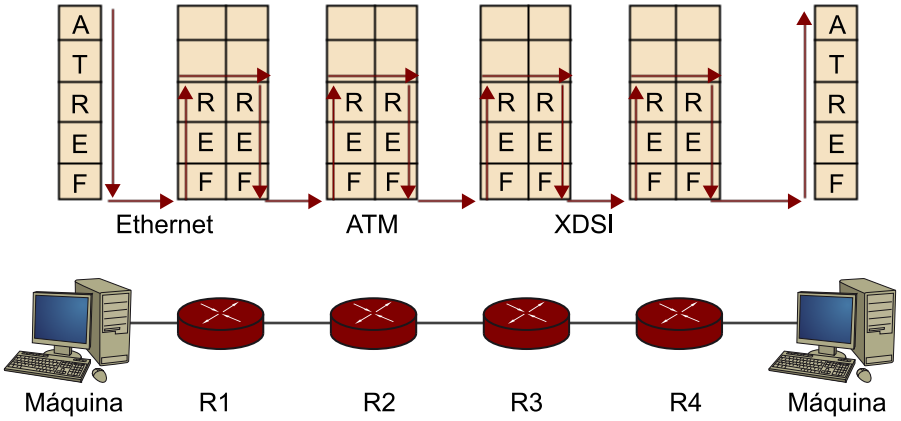
Hemos visto que la capa de red proporciona un servicio de comunicación entre dos máquinas, lo que establece diferentes rutas o caminos entre ellas. Cada ruta de comunicación está formada por una serie de enlaces que conectan la máquina origen con la de destino mediante unos dispositivos encaminadores intermedios. Cuando un datagrama del nivel de red sale de la máquina origen hacia la máquina destino atraviesa cada uno de estos enlaces individuales, que conforman el recorrido extremo a extremo.

En este sentido, es necesario una capa lógica adicional, situada inmediatamente debajo de la capa de red, que se encargue de gestionar cada enlace individual. Esta nueva entidad debe ofrecer a la capa de red un transporte de información fiable entre los diferentes enlaces que atraviesa a lo largo de un recorrido. La capa física no es capaz de aportar ninguno de los elementos necesarios para la transmisión efectiva de información en un enlace. La capa que realiza esta función recibe el nombre de nivel de enlace y se sitúa entre los niveles físico y de red.

El nivel de enlace consiste en dos programas o procesos que se ejecutan en ambos lados de un enlace y se comunican entre sí. Para que estos dos procesos se puedan comunicar, es necesario establecer un formato para la información que se intercambia y un conjunto de reglas de comportamiento o protocolos necesarios para la transmisión de datos.

El principal cometido de la capa de enlace es conseguir que la comunicación de datos en un enlace se realice correctamente en un medio físico de transmisión no perfecto, que puede introducir errores. Gráficamente, podemos decir que el nivel de enlace se encarga de establecer y mantener un puente de comunicación lo más fiable posible entre dos nodos vecinos, para que por encima puedan circular los datagramas de nivel superior, tal y como podemos observar en la figura 1.

Figura 1. Ruta de comunicación entre dos máquinas finales



Ruta de comunicación creada entre dos máquinas finales, formada por 5 enlaces: 2 enlaces comunican las máquinas finales con los encaminadores de la red y 3 enlaces internos intercomunican sólo encaminadores de la red

De la observación de dicha figura podemos destacar 2 características muy importantes del nivel de enlace:

1) A lo largo de un recorrido de comunicación los enlaces pueden utilizar diferentes protocolos y estar constituidos por tecnologías de base totalmente diferentes. Un encaminador puede disponer de diferentes enlaces y cada uno de ellos puede utilizar un protocolo de nivel de enlace distinto. En la figura 1 podemos observar cómo un datagrama enviado desde la máquina origen es manejado por Ethernet en el primer enlace, por el protocolo ATM[1] en el segundo enlace y, sucesivamente, por una tecnología distinta en cada nuevo enlace.

(1)ATM es la sigla de *asynchronous transfer mode*.

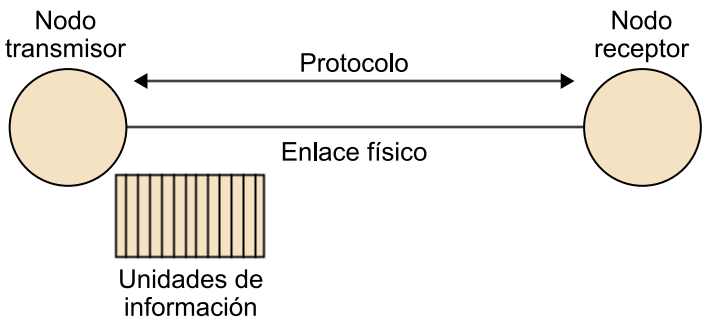
2) Una de las funcionalidades básicas del nivel de enlace consiste en encapsular/desencapsular los datagramas de la capa de red en unidades de información (PDU[2]) de la capa de enlace, denominadas tramas. Las flechas de la figura 1 indican el flujo que sigue la información a lo largo del recorrido. Cuando una trama llega a un encaminador desde un enlace de entrada, la capa de enlace desencapsula/extrae el datagrama de la trama recibida y se lo entrega a la capa de red. Una vez que la capa de red determina por dónde debe encaminar el datagrama, se lo envía al enlace de salida. En este punto, el datagrama es encapsulado según las normas del protocolo del enlace y preparado para ser enviado a través de él.

(2)PDU es la sigla de *protocol data unit*.

1.1. Terminología y definiciones

En el nivel de enlace identificamos los siguientes elementos:

Figura 2



- **Nodo**: es una máquina, que puede ser un terminal o un encaminador[3]. En la terminología clásica existen dos tipos de nodos en un enlace: el nodo transmisor o primario y el nodo receptor o secundario. No obstante, y dependiendo del medio, ambos pueden realizar funciones de transmisión y recepción.

(3)En inglés, *router*.

- **Enlace**: es el canal físico que conecta dos nodos adyacentes en el recorrido de la comunicación.

- **Protocolo de la capa de enlace**: es el modo de comunicarse entre los nodos para mover un datagrama sobre el enlace individual. Define el formato de la información intercambiada entre los nodos, así como las acciones tomadas por esos nodos cuando envían y reciben esas unidades de información.

- **Trama**: son las unidades de datos (PDU) intercambiadas por un protocolo de la capa de enlace. El nodo transmisor encapsula el datagrama de la capa de red en una trama de la capa de enlace y la transmite por el enlace. El nodo receptor recibe la trama y de él extrae el datagrama de nivel de red.

1.2. Tipos de enlaces

Básicamente, podemos destacar dos tipos de enlaces:

1) Enlaces de comunicación punto a punto: sólo participan dos entidades o puntos. Son enlaces 1 a 1, compuestos por un único nodo emisor en un extremo del enlace y un único nodo receptor en el otro. Ambos nodos utilizan en exclusiva el enlace, sin compartir el canal. Son considerados enlaces punto a punto:

- Bucle de abonado local, cable de dos hilos telefónico para acceso a Internet.

- Las redes de área local *fast Ethernet*.

- Las redes de área local *gigabit Ethernet*.

- PPP[4], HDLC[5] (como enlace), X.25 para red y TCP[6] como transporte (en este caso, también es extremo a extremo).

2) Enlaces de difusión[7] o canales de multidifusión: son enlaces 1 a *N*, en los que una serie de nodos están conectados al mismo canal físico de comunicación. La transmisión realizada por un nodo la reciben todos los nodos conectados al enlace. En este caso, son necesarias unas políticas de coordinación (o protocolos de acceso al medio) que permitan la compartición del único medio de manera eficiente, tratando de evitar al máximo las colisiones entre tramas. Son enlaces de difusión:

- Las redes de área local Ethernet (semidúplex).
- Las redes de área local sin cable (WIFI).
- Los enlaces con satélites.
- Las redes de acceso híbrido fibra-cable (HFC[8]).
- Las redes de área local *token ring*.
- Las redes de área local FDDI[9].
- Las redes metropolitanas (MAN[10]).

(4)PPP es la sigla de *point to point protocol*.

(5)HDLC es la sigla de *high level data link control*.

(6)TCP es la sigla de *transmission control protocol*.

(7)En ingles, *broadcast*.

(8)HFC es la sigla de *hybrid fibre coaxal*.

(9)FDDI es la sigla de *fiber distributed data interface*.

(10)MAN es la sigla de *metropolitan area networks*.

1.3. Servicios proporcionados por la capa de enlace

El servicio básico del nivel de enlace consiste en mover correctamente un datagrama de nivel de red, desde un nodo hasta otro adyacente, sobre un enlace de comunicación fijo en el recorrido.

Los posibles servicios que puede ofrecer un protocolo de la capa de enlace son:

1) Gestión de las tramas: el nivel de enlace se encarga de la organización y gestión de las tramas. Entre las diferentes funciones que engloba la gestión de tramas podemos destacar:

- Entramado o composición de la trama.
- Sincronización a nivel de trama.
- Transparencia de trama.
- Numeración y secuenciación.
- Multiplexación de tramas de niveles superiores.
- Direccionamiento.

2) Gestión del enlace: coordinación y gestión de los procesos de inicialización, mantenimiento y finalización del enlace. Varía en función del tipo de servicio que la capa de enlace suministra a la capa de red.

3) Control de errores: se trata de una de las funciones básicas del nivel de enlace. Se asume que el medio de transmisión físico que está "por debajo" no es perfecto e introduce errores de transmisión. Es necesario destinar una parte de los bits que se intercambian a la detección y a la posterior gestión de los errores para controlar que no se produzcan errores de transmisión. El control de errores distingue tres categorías de técnicas:

- Detección de errores (utilización de códigos detectores de errores).
- Corrección de errores (utilización de códigos correctores de errores).
- Retransmisión de tramas (implementación de la entrega fiable).

4) Control de flujo: funcionalidad que permite que la estación emisora y la receptora se pongan de acuerdo en el ritmo de transmisión de datos. Si la estación receptora recibe las tramas más rápidamente de lo que es capaz de procesarlas, el nivel de enlace remoto debe "frenarlas" para evitar que se sature la memoria intermedia o temporal que almacena las tramas pendientes de procesar.

5) Control de acceso al medio: esta funcionalidad adquiere relevancia en los enlaces de acceso múltiple o enlaces *broadcast*, en los que un número determinado de nodos comparten el mismo medio físico.

IEEE divide la capa de enlace en dos subniveles:

- LLC[11]
- MAC[12]

(11)LLC es la sigla de *logical link layer*.

(12)MAC es la sigla de *medium access control*.

El subnivel MAC es el encargado de especificar las reglas con las que se transmite una trama sobre el enlace. Su función es garantizar que los usuarios accedan correctamente al medio de transmisión en condiciones de igual prioridad, velando por que el acceso no sea simultáneo. Cuando los accesos sean simultáneos, intentará solucionar el conflicto entre los nodos. En los enlaces punto a punto los protocolos de acceso al medio dejan de tener sentido.

1.4. Adaptadores y dispositivos de red

Los nodos o encaminadores se conectan a los enlaces mediante un adaptador, conocido como tarjeta de interfaz de red o NIC[13].

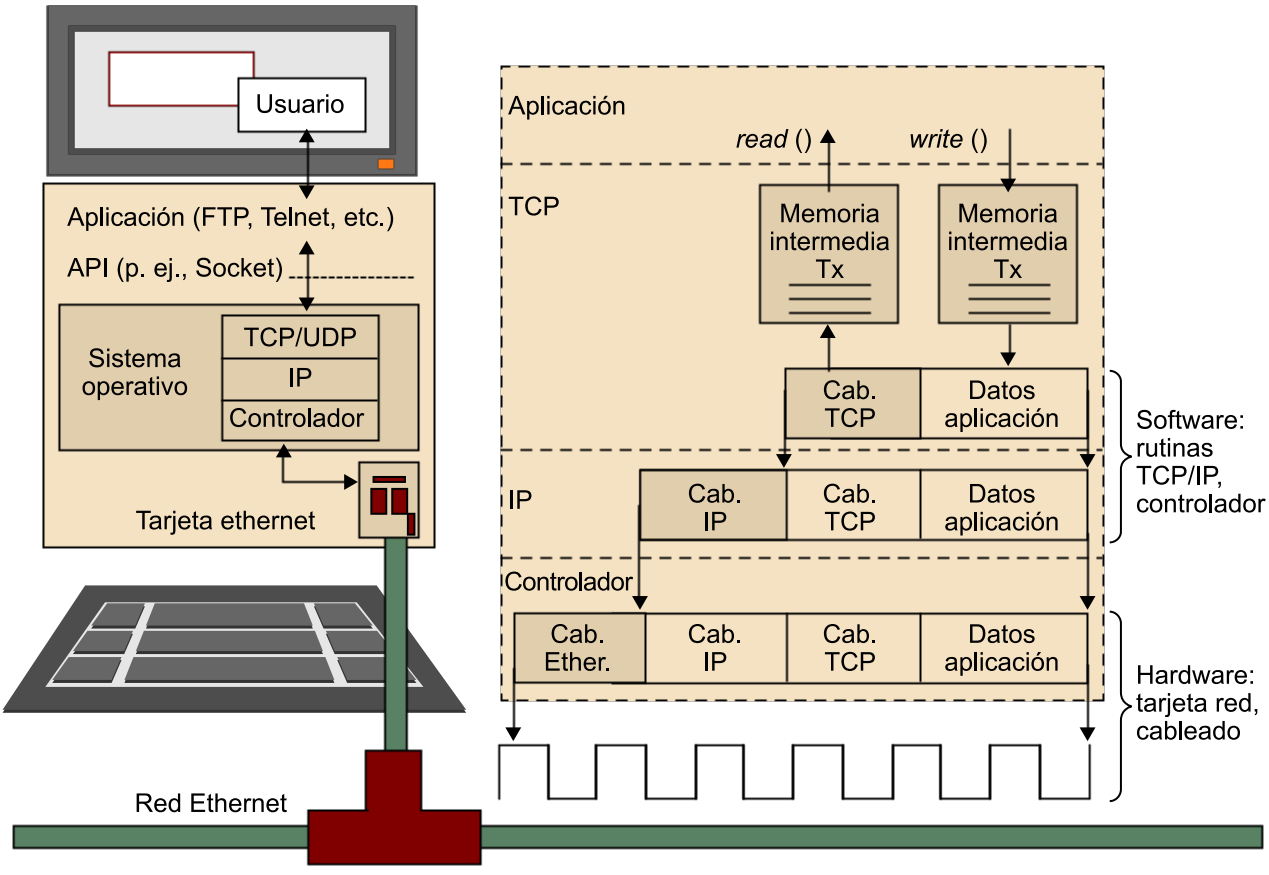
(13)NIC es la sigla de *network interface card*.

La importancia de una tarjeta de red radica en que en ésta se encuentran implementadas mayoritariamente las funciones del protocolo de la capa de enlace. Si un protocolo de la capa de enlace proporciona detección de errores, entrega fiable (numeración y reconocimientos) o acceso aleatorio, estas funcionalidades están implementadas completamente en los adaptadores.

Físicamente, un adaptador es una placa hardware (o una tarjeta PCMCIA) que contiene todos los elementos de un pequeño ordenador: memoria RAM, chip DSP, interfaz de bus con la maquina y otro interfaz para el conexionado con el enlace. Normalmente, se encuentra alojado en la misma caja física que el resto del nodo, compartiendo la alimentación y los buses.

Los componentes principales de un adaptador son la interfaz del bus y la del enlace. La interfaz del bus es responsable de comunicar con el nodo padre del adaptador. Transfiere datos e información de control entre el adaptador y el nodo padre. La interfaz del enlace es responsable de implementar el protocolo de la capa de enlace. También incluye la circuitería de transmisión y recepción.

Figura 3



Un adaptador tiene un cierto grado de autonomía:

- En recepción: cuando recibe una trama determina si ésta contiene errores. Si es así, la desecha sin notificarlo a su nodo padre. Si es correcta, desencapsula el datagrama de la capa de red e interrumpe a su nodo padre para pasarlo arriba de la pila de protocolos.

- En transmisión: cuando un nodo pasa un datagrama de abajo de la pila de protocolos a un adaptador, delega totalmente en el adaptador la tarea

de transmitir el datagrama sobre el enlace. El adaptador encapsula el datagrama en una trama y transmite la trama en el enlace de comunicación.

Ejercicios

1. Según lo visto, ¿creéis que todos los protocolos de nivel de enlace ofrecen todos los servicios de la capa de enlace descritos?

Solución ejercicio 1

No todos los servicios están implementados en todos los protocolos. Cada protocolo específico de la capa de enlace define una serie de servicios y desecha otros.

2. Indicad qué posibles servicios de la capa de enlace también son ofrecidos por las capas de red o transporte a sus respectivos niveles superiores.

Solución ejercicio 2

- **Entrega fiable**: tanto la capa de enlace como la de transporte pueden proporcionar entrega fiable. La capa de trasporte proporciona entrega fiable entre dos procesos extremo a extremo; en cambio, la capa de enlace proporciona entrega fiable entre dos nodos conectados por un único enlace.
- **Control de flujo**: la capa de transporte también puede proporcionar control de flujo. En este caso, proporciona control de flujo extremo a extremo, mientras que en un protocolo de la capa de enlace, aquél se proporciona en una base de nodo a nodo adyacente.
- **Detección de errores**: son ofrecidos también en la capa de transporte y en la capa de red.

3. Indicad si las siguientes tecnologías de nivel de enlace implementan los servicios de nivel de enlace.

	Entramado	Acceso al medio	Detección de errores	Corrección de errores	Retransmisión de tramas
PPP					
ATM					
Ethernet					
Frame relay					

4. ¿Por qué no se halla el control de congestión entre las funcionalidades del nivel de enlace?

Solución ejercicio 4

Si el control de flujo intenta no saturar la memoria intermedia (*buffer*) del nodo receptor, el objetivo del control de congestión es no saturar las memorias intermedias. Es evidente que en un enlace no hay nodos intermedios. Trata la comunicación directamente con el vecino y, por tanto, el control de congestión no tiene sentido.

5. Indicad las ventajas y desventajas que puede haber en la corrección de errores respecto a la retransmisión de tramas erróneas. ¿En qué situaciones es preferible la corrección de errores?

Solución ejercicio 5

La corrección de errores evita el retraso que implica solicitar la retransmisión de las tramas. Por el contrario, los códigos correctores de errores necesitan añadir mucha redundancia (bits extras) y, en consecuencia, la transmisión es menos eficiente y se aumenta el sobrecoste (la relación entre los bits de información y de control disminuye).

Un caso típico en el que es preferible un código corrector de errores a un código detector acompañado de retransmisiones es en comunicaciones vía satélite. Conviene más pagar en ineficiencia por el incremento de bits redundantes que en tiempo de retransmisiones porque la distancia que hay que recorrer es muy grande.

2. Gestión de tramas

2.1. Entramado

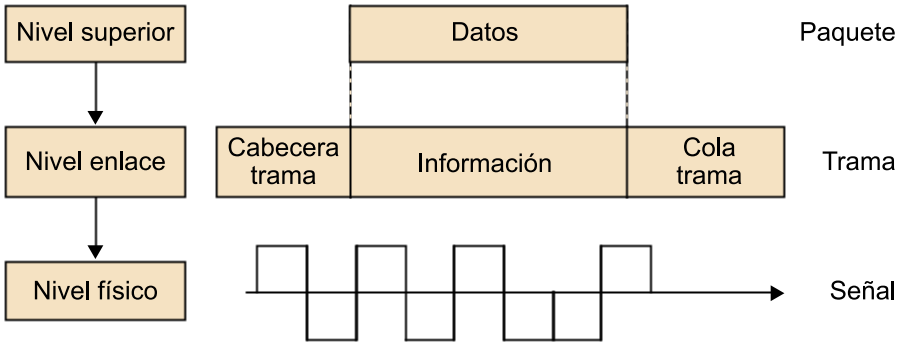
En el nivel de enlace, el control de la transmisión de datos entre nodos vecinos se vertebra sobre el proceso de creación y envío de la trama.

En la capa física el envío de información se realiza con bits sueltos de manera no fiable; la capa de enlace actúa de un modo distinto: construye con los bits estructuras ordenadas, conocidas como tramas[14], que son las que se envían por el enlace.

(14)En inglés, *frames*.

La gran mayoría de los protocolos de la capa de enlace encapsulan los datagramas de la capa de red dentro de una trama antes de viajar por el enlace. Una buena parte de las tareas de la capa de enlace está relacionada con la construcción e identificación de las tramas. Por ejemplo, una ventaja de la utilización de tramas es que permite simplificar el proceso de detección y corrección de errores.

Figura 4



Las tramas se organizan en campos, de manera que hay campos con bits de información y campos con bits de control. Aunque la estructura de una trama depende de cada protocolo específico del nivel de enlace, generalmente la podemos dividir en las siguientes partes:

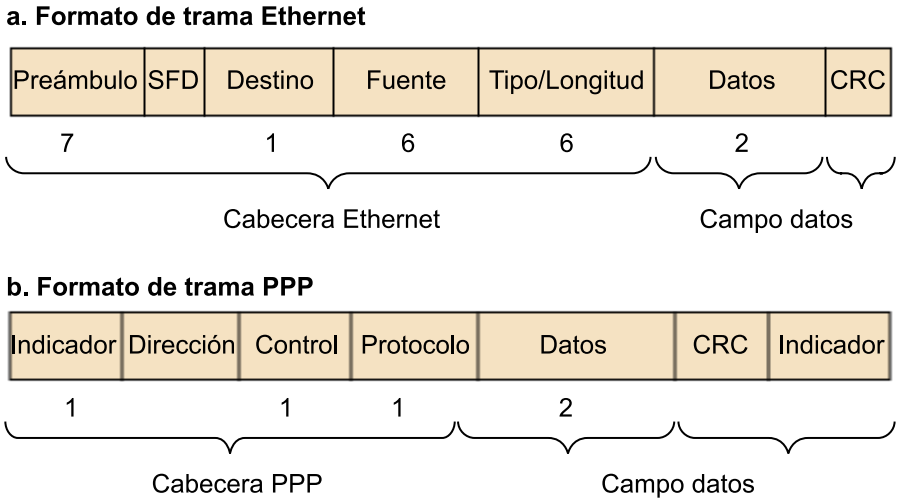
- Una cabecera, compuesta por campos de bits de control de la trama: dirección física, longitud de la trama, tipo de datos que transporta, etc.

- Un campo de datos, al que van los bits de información correspondientes a datagramas de la capa de red.

- Una cola que cierra la trama, esto es, un campo de control necesario para controlar los errores.

Podemos comprobar esta división observando la estructura de trama de los dos protocolos más importantes de nivel 2: Ethernet y PPP.

Figura 5. Formatos de trama Ethernet y PPP



2.2. Sincronización a nivel de trama

El sincronismo de trama[15] es el mecanismo que utiliza el nivel de enlace para determinar el inicio y el final de una trama dentro del flujo de bits o caracteres que llega del nivel físico.

(15)En inglés, *framing*.

Hasta ahora hemos hablado siempre del nivel físico como un medio capaz de transportar un flujo de bits. Sin embargo, existen algunos medios físicos que tienen como unidad de transmisión el carácter, que se define como un bloque fijo de bits. Este caso se conoce como transmisión orientada a carácter.

De hecho, se habla de dos tipos de protocolos, uno para cada uno de los tipos de transmisión mencionados:

- **Protocolos orientados a bit**: protocolos de nivel de enlace diseñados para ir sobre una transmisión orientada a bit. En este caso, el nivel físico tiene como unidad de transmisión el bit. Un ejemplo típico de este grupo es el protocolo HDLC.

- **Protocolos orientados a carácter**: protocolos de nivel de enlace diseñados para ir sobre una transmisión orientada a carácter. El medio físico tiene como unidad de transmisión el carácter. Un ejemplo típico de este tipo es el protocolo BSC[16] de IBM.

(16)BSC es la sigla de *binary synchronous control*.

Para describir los mecanismos de sincronización en el nivel de trama, pondremos como ejemplos los que utilizan los protocolos HDLC y BSC, por ser muy representativos.

2.2.1. Detección del inicio de trama

Depende del tipo de transmisión:

- En transmisión orientada a carácter, el inicio de trama se indica con un carácter especial denominado carácter de inicio de trama, por ejemplo, STX[17]. STX está definido en ASCII y en EBCDIC (IBM). Se utiliza en terminales IBM cuyo protocolo es BSC.

- En transmisión orientada a bit, se indica el inicio de trama con una combinación especial de bits denominada indicador[18] de inicio de trama. En HDLC el patrón de bits que identifica el inicio de trama es 01111110.

[17] STX es la sigla de *start of text*.

[18] En inglés, *flag*.

2.2.2. Detección de final de trama

Se puede implementar utilizando dos métodos:

- Utilizando un carácter especial denominado carácter de final de trama (en transmisión orientada a carácter) o una combinación especial de bits llamada indicador de final de trama (en transmisión orientada a bit).

- Utilizando un campo de longitud que indica el tamaño de la trama.

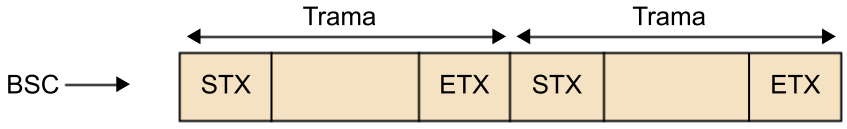
Algunos protocolos utilizan las dos técnicas conjuntamente para llevar a cabo el control de errores. De este modo, si el carácter de final de trama o el indicador, según el caso, no llega al final de la trama indicado por el campo de longitud, se detecta un error de delimitación de trama, o error de *framing* (por ejemplo, Ethernet>).

Ejemplos de sincronización de trama

1) Sincronización de trama en una transmisión orientada a carácter:

Tanto el código ASCII como el código EBCDIC tienen los caracteres de control STX y ETX[19]. Algunos de los protocolos orientados a carácter más extendidos utilizan estos caracteres en el sincronismo de trama. La figura 6 muestra cómo sería el sincronismo de trama con estos dos caracteres:

Figura 6



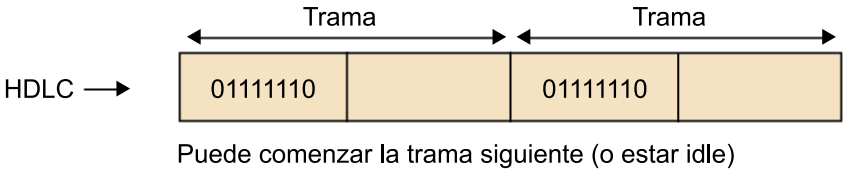
Uno de los protocolos orientados a carácter más conocidos, BSC, utiliza los caracteres STX y ETX en la sincronización de las tramas.

[19] ETX es la sigla de *end of text*.

2) Sincronización de trama en una transmisión orientada a bit:

Como ejemplo de sincronización de trama consideraremos la figura 7, que muestra los indicadores que utiliza el protocolo HDLC:

Figura 7



En el protocolo HDLC se define el indicador de final de trama con el mismo conjunto de bits que el de inicio de trama. El protocolo permite que si existen dos tramas consecutivas, el indicador de final de trama sea también el de inicio de trama de la siguiente, y así se ahorra la transmisión de este indicador.

2.3. Mecanismo de transparencia

Los datos de información que transporta la trama son totalmente arbitrarios. Si no se utiliza el campo de longitud, puede darse el caso de que en la detección de final de trama, tanto en transmisiones orientadas a carácter como orientadas a bit, un carácter o un conjunto de bits de datos se pueda confundir con el indicador de final de trama. Esto puede provocar situaciones erróneas, en las que se interpretaría como erróneos finales de trama cuando no lo son. Para evitarlo, se utiliza un mecanismo de transparencia que permite que el uso del protocolo no afecte de ningún modo al mensaje transmitido.

Ejemplos de mecanismos de transparencia

1) Mecanismo de transparencia en una transmisión orientada a carácter:

Para conseguir la transparencia, el protocolo BSC utiliza un tercer carácter, el DLE[20]. Éste permite hacer transparentes ("escapar") caracteres de control que pueden aparecer fortuitamente dentro del mensaje (ya que éste podría estar compuesto por cualquier carácter del alfabeto del código) y que en caso de interpretarse afectaría de manera muy negativa al proceso de la transmisión.

La técnica para conseguir la transparencia se conoce como relleno de caracteres[21]. El funcionamiento es el siguiente:

a) En transmisión:

- Los caracteres de control STX y ETX de inicio y final de trama van precedidos de un DLE.
- Cuando encuentra un DLE entre los datos de información, inserta otro DLE independientemente del carácter que siga. Por ejemplo, si los caracteres DLE-STX o DLE-ETX se encuentran mezclados en los datos, la capa de enlace inserta un DLE justo antes de cada carácter DEL y finalmente envía DLE-DLE-ETX o DLE-DLE-STX.

b) En recepción:

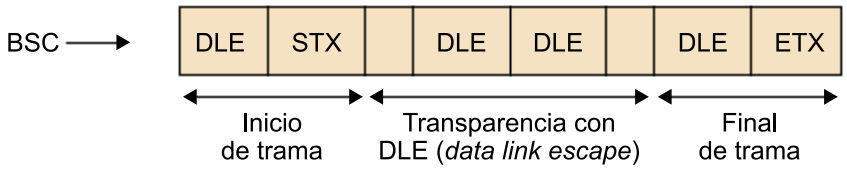
- Si recibe los caracteres DLE-STX, detecta inicio de trama.
- Si recibe los caracteres DLE-DLE, elimina uno de los caracteres DLE y no se interpreta el siguiente carácter como carácter de control.
- Si recibe DLE-ETX, lo interpreta como final de trama.

[20] DLE es la sigla de *data link escape*.

[21] En inglés, *character stuffing*.

[22] En inglés, *bit stuffing*.

Figura 8



2) Mecanismo de transparencia en una transmisión orientada a bit:

Hemos visto que en el protocolo HDLC el indicador de delimitación de trama es 01111110. Se debe evitar que esta secuencia de bits se encuentre dentro del mensaje de datos, porque ello provocaría interpretaciones erróneas. El mecanismo de transparencia que evita esto recibe se conoce como relleno de bit[22] en protocolos orientados a bit.

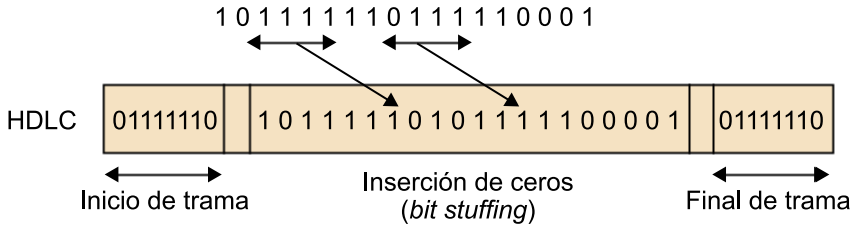
a) En TX:

Si hay cinco unos seguidos en el campo del mensaje, se inserta un 0, independientemente del bit que siga a continuación. Si se produce la coincidencia de que dentro de los bits de información estuviera el indicador 01111110, la inserción del bit 0 provocaría que el receptor no lo interpretara como final de trama. De este modo, la secuencia de delimitación de trama (seis unos seguidos) es transmitida de forma única por el canal.

b) En RX:

El receptor elimina todos los ceros "extras" insertados por el transmisor. Si llegan cinco unos seguidos de un 0, se elimina el 0 y no se interpreta la secuencia como un posible indicador.

Figura 9



2.4. Numeración y secuenciación

Hemos visto cómo los protocolos de retransmisión ARQ[23] necesitaban numerar tanto las tramas de información como las tramas de confirmación para poder relacionar unas con otras y garantizar de esta manera el correcto funcionamiento de la retransmisiones. Estas tramas incluyen un número de secuencia en uno de los campos de la cabecera que añade el protocolo de nivel de enlace, junto con el campo de control que sirve para detectar posibles errores.

[23]ARQ es la sigla de *automatic repeat request*.

Ved también

Podéis ver la numeración de las tramas en el módulo "La capa de transporte de datos" de esta asignatura.

Vemos que la numeración de tramas es consecuencia directa de un protocolo que realiza recuperación automática de errores (retransmisiones). En el resto de los casos no sería estrictamente necesario realizar una numeración de tramas.

En el apartado "Gestión del enlace", veremos que normalmente las tramas de señalización y control del enlace (las que no son de información) no suelen llevar número de secuencia. Estas tramas reciben el nombre de tramas no numeradas[24].

[24]En inglés, *unnumbered frames*.

2.5. Multiplexación en el nivel de enlace

El concepto de multiplexación ya se ha introducido en otros módulos de la asignatura. Esta técnica se puede utilizar en cualquier nivel de la arquitectura de comunicaciones. Veremos que también se puede encontrar en un protocolo de nivel de enlace.

La idea de la multiplexación consiste en utilizar un único medio para la transmisión de diferentes flujos de información.

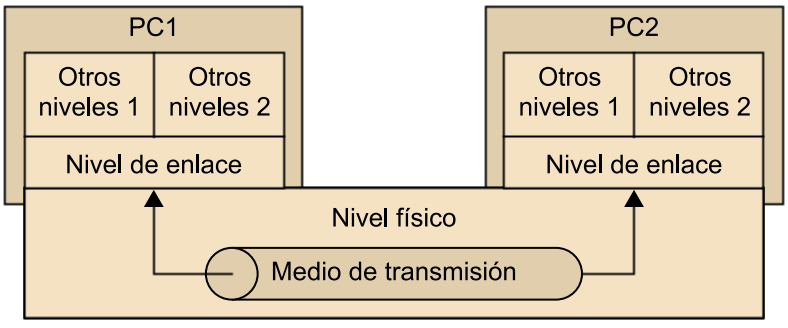
Obviamente, hay que definir un mecanismo que permita separar los distintos flujos con respecto a la recepción.

Veamos dos ejemplos en los que puede ser útil tener multiplexación en el nivel de enlace:

- Por ejemplo, en un enlace multipunto. El hecho de tener el medio compartido se puede interpretar como si hubiera un único enlace en el que se multiplexan las tramas de las distintas estaciones. El mecanismo de distinción de las diferentes estaciones se puede interpretar como un mecanismo de multiplexación. En este caso, la multiplexación se conseguiría utilizando una dirección de nivel de enlace, dentro de uno de los campos de control de la trama, que identifica cada estación.

- Otro ejemplo de multiplexación en el nivel de enlace es el que muestra la figura 10. En este ejemplo hay un nivel físico y de enlace común y, por encima, dos arquitecturas de comunicaciones (conjunto de protocolos) diferentes. En este caso el nivel de enlace lleva las unidades de información de la arquitectura 1 o 2 a su arquitectura par 1 o 2, respectivamente.

Figura 10. Ejemplo de multiplexación en el nivel de enlace



Este ejemplo no es extraño en la práctica, ya que existen numerosos protocolos de comunicaciones que puede interesar que convivan dentro de un mismo entorno y que compartan el mismo enlace; por ejemplo, en una red de área local.

Para poder distinguir la arquitectura que transmiten las tramas y, por lo tanto, la arquitectura a la que hay que entregarlas, también se utilizan direcciones de nivel de enlace.

2.6. Direccionamiento

El direccionamiento en el nivel de enlace depende del tipo de enlace existente:

- En los enlaces de comunicación punto a punto, el campo de direccionamiento deja de tener sentido al haber dos entidades participantes en el enlace, y por lo tanto siendo de sobra conocido el otro extremo del enlace. Por ejemplo, el campo dirección de una trama PPP normalmente lleva siempre la misma dirección.

- En los enlaces *broadcast*, sí que es necesario el campo dirección de la cabecera de la trama, al existir más de un posible destinatario del mensaje. Por ejemplo, Ethernet utiliza los campos de dirección MAC destino y origen con tal fin.

3. Gestión del enlace

Por gestión del enlace entendemos el modo en el que los nodos administran y establecen el enlace, es decir, si estructuran la transmisión por fases (como inicialización, mantenimiento y finalización), o si realizan transmisiones sin establecer una conexión previa.

El nivel de enlace reconoce dos maneras de establecer un enlace entre dos entidades:

1) Protocolos orientados a la conexión[25], como el protocolo PPP.

2) Protocolos no orientados a la conexión[26], como el protocolo Ethernet.

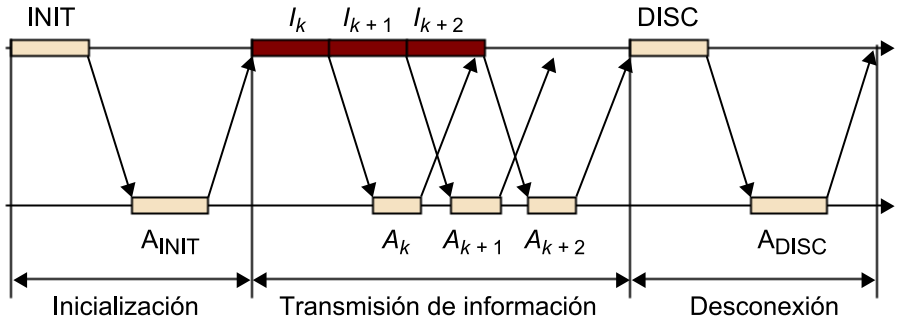
[25] En inglés, *connection oriented*.

[26] En inglés, *connectionless*.

Un protocolo orientado a la conexión es aquel que requiere una fase de inicialización previa a la fase de transmisión de tramas de información, en la que normalmente se negocian parámetros necesarios para la transmisión. También necesita una fase de desconexión, en la que se acuerda la finalización del enlace. Esta fase permite liberar los recursos que se le dedican, como las memorias intermedias para almacenar las listas de transmisión y recepción.

La figura 11 muestra las diferentes fases que pueden haber en una comunicación orientada a conexión:

Figura 11. Fases de inicialización y desconexión en un protocolo orientado a la conexión



La trama INIT solicita la inicialización, que es confirmada por la trama AINIT. Las tramas DISC y ADISC funcionan de manera análoga para la desconexión. Estas tramas son de control, necesarias para el establecimiento del enlace, pero no llevan información útil de niveles superiores. En las tramas hay un campo de control que indica el tipo de trama y que permite diferenciarlas.

Las tramas de control no suelen estar numeradas. La numeración de tramas de información es necesaria en protocolos de recuperación automática de errores, que generalmente son orientados a la conexión.

En cambio, en un protocolo de enlace no orientado a la conexión las entidades empiezan a intercambiar tramas de información sin previo aviso por el enlace. Habitualmente, son protocolos que no utilizan recuperación de errores en el nivel de enlace y que, por lo tanto, no necesitan numeración. En este caso, tendríamos un nivel de enlace no orientado a la conexión que podría realizar detección de errores (descartando las tramas erróneas) pero no pedir la retransmisión de tramas recibidas incorrectamente.

Hay motivos que pueden justificar un nivel de enlace no orientado a la conexión: en aplicaciones en tiempo real en las que se transmite voz o vídeo, por ejemplo, es posible que el retraso que se necesite para poder hacer la recuperación de errores no sea aceptable.

De modo más concreto, la capa de enlace puede suministrar uno de los siguientes tipos de servicio a la capa de red (son los tipos de servicio que suministra el protocolo HDLC):

1) Servicio no orientado a conexión y sin acuse de recibo

El envío se realiza sin esperar ninguna indicación del receptor sobre el éxito o fracaso de la operación. Tampoco se establece o libera una conexión. Este tipo de servicio es apropiado cuando la tasa de error es muy baja (redes locales o fibra óptica) y se deja la misión de comprobar la corrección de la transmisión a las capas superiores (nivel de red o de transporte). También se usa el servicio no confirmado cuando se quiere transmitir información en tiempo real (normalmente, voz o datos) y no se quiere sufrir el retraso que impondría un servicio más sofisticado en la capa de enlace (se supone que este tipo de información puede sufrir una pequeña tasa de error sin efecto apreciable).

2) Servicio no orientado a conexión con acuse de recibo

Se produce un acuse de recibo, para cada trama enviada, aunque todavía no hay establecimiento de conexión. De esta manera el emisor puede estar seguro de que ha llegado.

3) Servicio orientado a conexión con acuse de recibo

Es el más seguro y sofisticado. El emisor y el receptor establecen una conexión explícita de antemano, las tramas que enviar se enumeran y se asegura de que son recibidas todas correctamente en destino y transmitidas a la capa de red.

En el servicio orientado a conexión se pueden distinguir tres fases: establecimiento de la conexión, envío de los datos y terminación de la conexión. En la primera se disponen los contadores y las memorias temporales necesarias para la transmisión; en la segunda se envían los datos, y en la tercera se libera la memoria ocupada con datos temporales y variables.

4. Control de errores

En caso de que se reciba una trama con errores, el nivel de enlace puede adoptar una de las soluciones siguientes:

1) Detección de errores y descarte de la trama

Se trata de un servicio muy común en los protocolos de la capa de enlace, que generalmente se implementa en hardware. Es un mecanismo que permite detectar si algún bit de la trama original ha cambiado debido a efectos indeseables del canal (atenuación, ruido, etc.). En el caso de que la comprobación diera positivo, se rechazaría la trama o se tomarían otras acciones. Las capas de transporte y de red también proporcionan una forma limitada de detección de errores. La detección sólo es factible en aplicaciones que toleren un cierto grado de error en la información recibida.

2) Corrección de errores (si se utiliza un código corrector adecuado)

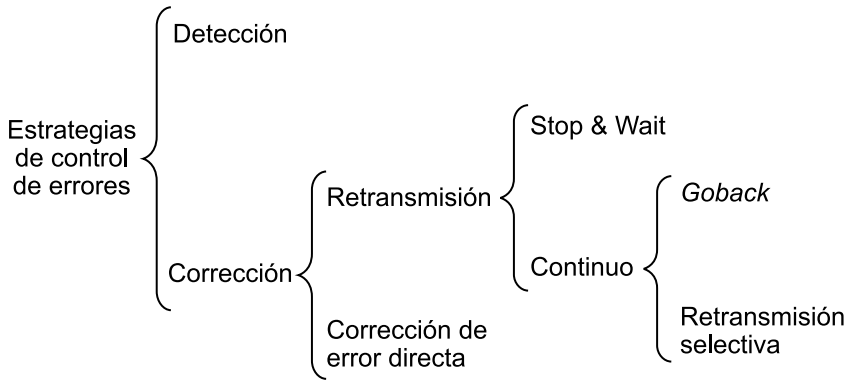
La corrección de errores es similar a la detección, pero en este caso el receptor no sólo se limita a detectar errores en los bits de la trama, sino que además intenta determinar dónde se han producido dichos errores (y, por tanto, corregirlos). Algunos protocolos (como ATM) proporcionan corrección de errores sólo para la cabecera del paquete, pero no es un servicio muy común.

3) Solicitar la retransmisión para una entrega fiable

El servicio de entrega fiable de la capa de enlace se realiza mediante reconocimientos y retransmisiones. Se trata de un servicio poco común en los protocolos de nivel de enlace, ya que tradicionalmente esta función la realiza el nivel de transporte de la torre TCP/IP. En general, su utilización adquiere sentido en enlaces propensos a tasas de error altas, como los enlaces sin cable, con el objetivo de corregir un error localmente (en el enlace cuando ocurre el error), en lugar de forzar una retransmisión extremo a extremo de los datos por un protocolo de nivel superior. Sin embargo, la entrega fiable de la capa de enlace se puede considerar como un gasto innecesario para enlaces de errores en pocos bits, como los de fibra óptica, los de cable coaxial y muchos pares trenzados de cobre. Por esta razón, muchos protocolos de la capa de enlace no proporcionan un servicio de entrega fiable.

Estas estrategias en la lucha contra los errores aparecen resumidas en la figura 12.

Figura 12



4.1. Estrategias de detección de errores

Supongamos que el nivel de enlace es capaz de delimitar perfectamente las tramas del flujo de bits que recibe del nivel físico. Ahora nos queda el problema de detectar cuáles de estas tramas tienen uno o más bits erróneos. Fundamentalmente, en este apartado veremos algunos conceptos básicos de la detección de errores y algunos de la corrección de errores.

Las técnicas de detección y corrección de errores son dos técnicas de control de errores que garantizan la integridad de las tramas enviadas mediante un canal con errores, combatiendo los efectos indeseables que introduce como la atenuación, las interferencias, el ruido, etc.

4.1.1. El ruido y sus efectos

El ruido es la componente que se incorpora al mensaje en algún momento de la transmisión y que no sólo no eleva el nivel de información, sino que incluso puede hacerlo disminuir por debajo del inicial. Se clasifica en extrínseco, o ajeno al circuito de datos, e intrínseco, que tiene su origen en algún elemento de este circuito. Sería ideal que las transmisiones se llevasen a cabo sin ruido, pero esto no es posible.

El ruido en los sistemas de transmisión produce errores. Denominamos tasa de error en los bits a la relación existente entre el número de bits recibidos erróneamente en un intervalo de tiempo y el número de bits enviados en ese tiempo. La medida de esta magnitud debe efectuarse en un intervalo lo suficientemente largo para proporcionar un promedio. La ITU-T recomienda un mínimo de 15 minutos. En los sistemas comerciales más usuales, esta tasa de error suele fluctuar entre 10-4 y 10-12 minutos, en función de las líneas, la velocidad de transmisión, etc.

$$t_{Error} = \frac{Bits\ erróneos}{Total\ bits}$$

Definimos como tasa de error residual la relación entre el número de bits erróneamente recibidos y no detectados o corregidos en un período de tiempo por el sistema de protección antierror que se está aplicando (si es que se aplica alguno) y el número total de bits enviados. Esta tasa es la que permite apreciar la seguridad teleinformática de un sistema.

$$t_{Error\ residual} = \frac{Bits\ erróneos\ no\ detectados}{Total\ bits}$$

4.1.2. Métodos de lucha pasiva

Un primer nivel de disminución de errores se consigue disminuyendo las causas que los producen. Se trata de aplicar métodos de lucha pasiva, tal y como podemos ver en la siguiente tabla:

Diferentes ruidos y métodos de lucha pasiva aplicados		
Tipo de ruido	**Causa**	**Sistema de lucha pasiva**
Eco	Malas conexiones. Mal estado de las líneas.	Supresor de eco.
Ruido blanco	Agitación térmica de la materia a temperatura por encima del 0 absoluto. Otras.	Filtraje y ampliación. Utilización de buenos conductores (superconductores) o fibra óptica.
Ruido impulsivo	Interferencias electromagnéticas y descargas de cualquier tipo sobre la línea o su entorno.	Blindaje de la línea. Utilización de líneas de fibra óptica.
Distorsión de fase	Características físicas de la línea utilizada.	Ecualizador (amplificador selectivo).
Distorsión de atenuación	Características físicas de la línea utilizada.	Ecualizador (amplificador selectivo).
Diafonía	Inducción electromagnética en conductores adyacentes.	Trenzado de pares. Blindaje de los pares. Uso de coaxial o fibra óptica.
Dispersión intermodal	Contrafases en los haces multimodo de índice escalonado en fibra óptica.	Uso de fibra monomodo o multimodo de índice gradual.
Fallos en conexiones, equipos y otros	Equipos de transmisión defectuosos y/o con tecnología obsoleta.	Instalación y mantenimiento adecuados. Mejoras técnicas.

No obstante, el ruido no puede ser eliminado totalmente del sistema; por ello, en el 100% de los supuestos la probabilidad de error es grande y ha de tenerse en cuenta.

4.1.3. Aspectos básicos de la detección de errores: codificación para la protección

Ante la imposibilidad de eliminar los errores, y si se desea evitar sus desagradables consecuencias en transmisión de datos, surge la necesidad de detectarlos una vez generados para lograr que el mensaje emitido se pueda reconstruir en el extremo receptor con la máxima fidelidad.

Toda trama puede tener una combinación de bits arbitraria. Si hay error en un bit de la trama o más, la nueva combinación es otra posible trama. De este modo, no basta con mirar los bits de la trama para averiguar si se ha producido algún error.

En casi todos los casos, los sistemas utilizados para la protección pasan por la codificación. Esta técnica consiste en añadir bits extras a la trama realmente enviada, de manera que en recepción se permita la detección de errores. Estos bits extras se calculan a partir de los bits que hay que proteger.

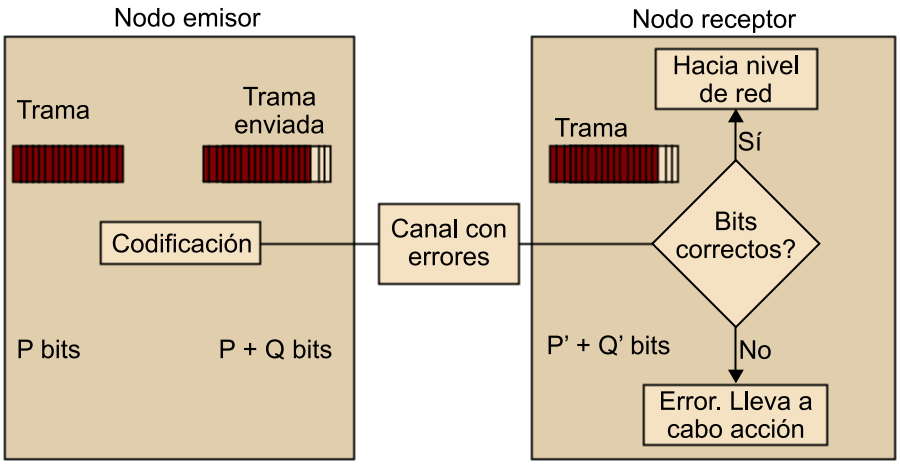
Al añadir estos bits estamos utilizando más bits de los estrictamente necesarios para transmitir la trama. Por este motivo se dice que los códigos detectores de errores añaden redundancia a los bits de datos que se quieren proteger. Se define el concepto de redundancia de un código como la diferencia entre la información máxima que podría proporcionar dicho código utilizando su alfabeto y la que proporciona realmente:

$$\%_{Redund} = \frac{Bits\ de\ control}{Bits\ totales} \times 100$$

Proceso de la codificación

En la figura 13 podemos observar el funcionamiento de la operación de codificación de una trama de *P* bits:

Figura 13



Supongamos que el tamaño de la trama de información es de *P* bits, y que añadimos *Q* bits para la detección y corrección de errores de los *P* bits anteriores. En ese caso, transformamos el conjunto de los *P* bits que queremos proteger en una nueva combinación de *P* + *Q* bits. Nos referiremos a esta nueva com-

Palabra código

En este apartado nos referiremos a las tramas como palabras código, que es el término que se utiliza en el contexto de la detección de errores.

binación como palabra código del nuevo código creado. Esta transformación es biunívoca, es decir, a cada combinación determinada de los P bits que hay que proteger le corresponde una sola combinación de $P + Q$ bits, y viceversa.

Se puede proteger tanto el datagrama encapsulado dentro de la trama como su cabecera, en la que reside la información de direccionamiento a nivel de enlace, los números de secuencia, etc.

Por el canal son enviados al nodo receptor todos los bits juntos en una trama de nivel enlace. El nodo receptor recibe la nueva trama de $P' + Q'$ bits, lo que significa que puede ser diferente de la secuencia original, tanto en el mensaje como en los bits de control de error. El reto del receptor es determinar si la secuencia P' es igual o no que la original P, dado que sólo ha recibido P' y Q'.

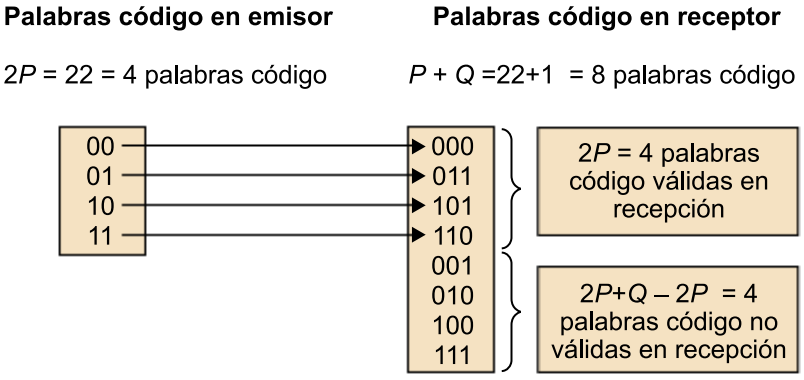
Esta cuestión se resuelve en términos deterministas, pero veremos que existirán ciertas probabilidades de no detectar secuencias erróneas. En efecto, observamos que el número de palabras código válidas es igual al número de combinaciones posibles de los bits de los datos que queremos proteger ($2P$). En cambio, el número posible de palabras código en recepción es de $2P + Q$. Las $2P + Q - 2Q$ combinaciones restantes son palabras código no válidas y no se utilizarán nunca en transmisión (porque no pueden resultar nunca de la transformación que aplica el código a los P bits que deben protegerse).

Detectamos que hay un error cuando los bits erróneos transforman una palabra código válida en una no válida. Si como consecuencia de los bits erróneos resulta otra palabra código válida, no detectaremos el error.

Ejemplo de detección de error

Observemos el siguiente ejemplo, en el que cada palabra código original de $P = 2$ bits se ha protegido con $Q = 1$ bit de redundancia. La protección utilizada para confeccionar la lógica del bit redundante ha sido la OR exclusiva de los bits originales.

Figura 14



Podemos obtener las siguientes situaciones en la transmisión de la secuencia "11":

Palabra código recibida	Acción	Decisión correcta
110	Palabra válida	SÍ

Palabra código recibida	Acción	Decisión correcta
111	Palabra no válida	SÍ
101	Palabra válida	NO: situación que hay que evitar

Debíamos haber recibido "110", pero en el último caso se ha recibido la secuencia "101", al producirse dos errores durante la transmisión. Al observar la tabla de correspondencia de palabras código, de un modo determinista deducimos que la transmisión ha sido correcta. Pero, en realidad, se ha producido un error no detectado.

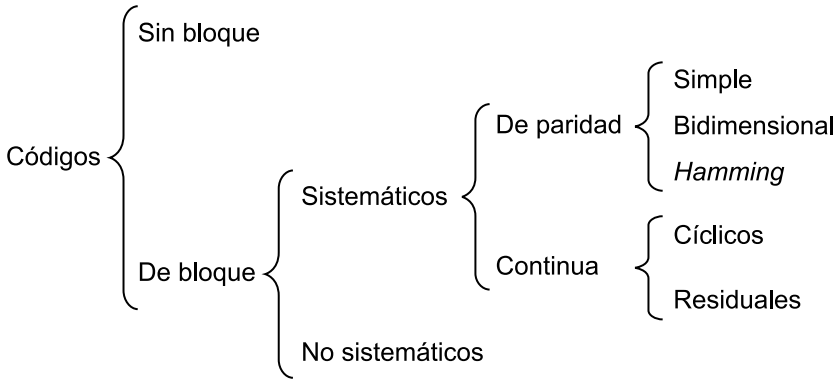
Implícitamente, observamos que existe un límite en la detección de errores dada una determinada codificación. En concreto, la codificación anterior sólo permite detectar en un bit. Falla cuando hay errores en más de un bit.

4.1.4. Clasificación de los códigos detectores/correctores de errores

La estructura de los códigos varía según el tipo de error que deben detectar o corregir. Normalmente, los codificadores y descodificadores se implementan en autómatas lineales. Por lo tanto, surge la necesidad de conseguir un equilibrio entre la capacidad de tratamiento del código, la velocidad de codificación/descodificación y la complejidad y el coste de los circuitos asociados.

En el gráfico de la figura 15 se propone una clasificación de los códigos de transmisión de datos:

Figura 15



- Los **códigos de bloque** se caracterizan porque el número de elementos que componen las palabras es constante. En ellos resulta de gran importancia el concepto de distancia de Hamming que veremos más adelante.

- Los **códigos no sistemáticos** son aquellos que forman sus palabras aleatoriamente a partir de un conjunto de dos posibles palabras. Este conjunto debe existir también en la memoria de trabajo del receptor para que éste pueda buscar la palabra recibida y así comprobar que la transmisión es correcta. Por ejemplo, si se establece un código para transmitir fechas en el que los meses se sustituyen por un signo del Zodiaco, será necesario que

emisor y receptor tengan la tabla de signos en memoria junto a la correspondencia meses-signos.

- Los **códigos sistemáticos** utilizan un algoritmo reversible que permite al receptor recuperar o comprobar la palabra original mediante la aplicación de dicho algoritmo en sentido inverso al emisor. Por ejemplo, contamos con el mismo código para transmitir fechas, pero, en este caso, codificamos el mes mediante el siguiente algoritmo: nuevoMes = (13 – mes) + (añoBisiesto?1:0).

Ahora ya no es necesario que emisor y receptor guarden una tabla de símbolos, sino que sean capaces de invertir el algoritmo anterior: mes = 13 + añoBisiesto?1:0) – nuevoMes.

En este módulo estudiaremos los códigos de paridad simple, bidimensional y de Hamming y los códigos continuos cíclicos (CRC[27]) y sumas de comprobación[28]. Todos son códigos detectores de errores y algunos, bajo determinadas circunstancias, también permiten la corrección de errores.

[27]CRC es la sigla de *cyclic redundancy check*.

[28]En inglés, *checksum*.

Veremos que:

- Las comprobaciones de paridad se utilizan muy poco en la práctica.
- Las sumas de comprobación son utilizadas en la capa red y de transporte.
- Las comprobaciones de redundancia cíclica son utilizadas en la capa de enlace.

4.1.5. Robustez de un código detector de errores

No siempre se consiguen detectar todos los errores de bits que provoca el canal de transmisión. Hemos visto que existe la posibilidad de que el nivel de enlace no detecte secuencias de bits que contienen errores, de modo que el receptor puede entregar un datagrama adulterado a la capa de red.

Una medida ideal para comparar los diferentes códigos detectores de error sería determinar la probabilidad de que se produjera un error no detectado en la transmisión de una trama. Desafortunadamente, esta probabilidad depende de las características del medio de transmisión y del código detector de errores, por lo que resulta difícil de determinar.

Para medir empíricamente la probabilidad de una trama errónea no detectada deberíamos contabilizar todas las tramas erróneas no detectadas y todas las tramas erróneas que se producen en el enlace. La relación entre estos dos valores sería la probabilidad buscada.

A causa de las dificultades que implica determinar la probabilidad de una trama errónea no detectada, consideraremos los tres parámetros para medir la **robustez de un código detector de errores** que se explican a continuación:

1) La distancia mínima del código (distancia de Hamming del código).

2) La capacidad de detección de ráfagas de error[29].

(29)En inglés, *burst detecting capability*.

3) La probabilidad de que una combinación arbitraria de bits sea aceptada como palabra válida.

Como veremos a continuación, para tomar estas medidas no debemos tener en cuenta el tipo de errores que introduce el medio de transmisión. Es decir, estas medidas dan idea de la facilidad que tiene un código para determinar ciertos tipos de errores. A la hora de elegir un código u otro habrá que tener en cuenta qué tipo de error introduce el medio de transmisión para elegir el código más adecuado, esto es, el que minimice la probabilidad de tener una trama errónea no detectada.

Por tanto, queremos elegir un esquema de detección de errores en el que la probabilidad de dichas ocurrencias sea pequeña. Generalmente, las técnicas de detección y corrección de errores más sofisticadas (aquellas que tienen una probabilidad menor de permitir errores de bits no detectados) incurren en un coste mayor (se precisa más computación para computar y transmitir un grado mayor de detección y corrección de errores de bits).

Probabilidades de error

Es importante no confundir la probabilidad de error en una trama con la probabilidad de una trama errónea no detectada. La probabilidad de error en una trama depende exclusivamente del medio. La probabilidad de una trama errónea no detectada es mucho más difícil de calcular porque depende, además, del código detector de errores. El código detector ideal detectaría todas las tramas erróneas.

Distancia de Hamming

Para definir la distancia de Hamming de un código, en primer lugar es necesario introducir el concepto de distancia de Hamming entre dos palabras código.

La distancia de Hamming entre dos palabras código se define como el número de bits diferentes que existe entre estas palabras. La distancia mínima de un código, o distancia de Hamming de un código, se define como la menor distancia entre dos palabras válidas cualesquiera del código.

Ejercicio

6. Calculad la distancia de Hamming entre estas dos palabras código: 100100101 y 000100001.

Solución ejercicio 6

Entre las siguientes palabras código, hay 2 bits de diferencia; por lo tanto, su distancia vale 2:

100100**1**01
000100**0**01

De la definición de la distancia de Hamming de un código deducimos también que un método exhaustivo para calcularla sería considerar todas las parejas posibles de palabras válidas, observar cuántos bits diferentes hay y elegir el mínimo. Generalmente, en la práctica no se aplica este método, sino que el cálculo se realiza a partir de las propiedades del código.

Cuanto mayor es la distancia de Hamming, más bits erróneos debe haber para que se produzca un error no detectado y, por lo tanto, el código detector de errores será mejor.

De esta definición se deduce que si la distancia de Hamming de un código vale *DH*, cualquier combinación de *n* bits erróneos se detectará con probabilidad 1, si cumple que:

$$n < D_H$$

Capacidad de detección de una ráfaga de error

Muchas veces, los errores no se producen en bits aislados, sino que son originados por chispas (interferencias) que afectan a distintos bits consecutivos. Sin embargo, una chispa no suele introducir errores en todos los bits que coinciden con su duración. Según las variaciones eléctricas de la intensidad de la chispa, algunos bits cambian, con lo que se produce un error, y otros no.

En una trama se define la ráfaga de error como el número de bits que existe entre el primer bit erróneo y el último, ambos incluidos.

La capacidad de detección de una ráfaga de error se define como el entero mayor, llamémosle B, tal que el código es capaz de detectar todas las ráfagas de error menores o iguales que B.

Ejemplo de ráfaga de error

En la trama siguiente, los bits erróneos son los que están marcados. Dado que entre el primer bit erróneo y el último (ambos incluidos) hay 7 bits, decimos que la ráfaga de error vale 7:

```
101000000000000
```

Evidentemente, cuanto mayor sea la capacidad de detección de ráfagas de error, mejor será el código detector de errores.

La capacidad de detección de ráfagas de error es especialmente importante cuando el medio de transmisión tiene tendencia a introducir los errores a modo de ráfagas. En este caso, cuanto mayor sea la capacidad de detección de ráfagas, menor será la probabilidad de tener una trama errónea no detectada.

Probabilidad de que una combinación arbitraria de bits sea aceptada como palabra válida

Hemos visto que si el número de bits erróneos de una trama no excede la distancia de Hamming ni la capacidad de detección de ráfagas, la trama errónea se detectará con probabilidad 1. En caso contrario, hay dos posibilidades:

a) La palabra código correspondiente a la trama errónea coincide con otra palabra código válida y, por lo tanto, no se detecta el error.

b) La palabra código resultante es una palabra no válida y se detecta el error.

El cálculo exacto de la probabilidad de que la trama errónea no sea detectada no es obvio. No obstante, podemos deducir de manera intuitiva un valor aproximado, mediante la siguiente suposición: que la palabra código correspondiente a la trama errónea pasa a ser, con la misma probabilidad, cualquier otra palabra código. Esto equivale a suponer que se elige una combinación arbitraria de bits. Si esta combinación es una palabra código válida, no se detectará el error; si no lo es, el error se detectará.

Dado que una palabra código tiene una medida de $P + Q$ bits, cada una de las combinaciones arbitrarias posibles se puede recibir con una probabilidad de $1/2^{P+Q}$. Dado que hay 2^P palabras código válidas, la probabilidad de que una combinación arbitraria de bits sea aceptada como una palabra válida será $2^P / 2^{P+Q} = 2^{-Q}$.

La probabilidad de que una combinación arbitraria de bits sea aceptada como una palabra válida es 2^{-Q}, donde Q es el número de bits que añade el código detector de errores.

Cuanto mayor sea *Q*, menor será esta probabilidad y mejor será el código. Esto demuestra que, cuantos más bits añade el código detector de errores, más difícil es que se produzca un error no detectado.

4.1.6. Comprobaciones de paridad

En la práctica se utilizan poco, dado que son poco robustos. No obstante, son útiles para proporcionar comprensión de las técnicas de corrección de errores.

Paridad simple (bit de paridad)

Es el código detector de errores más sencillo que existe. El control de paridad consiste en añadir un solo bit (denominado bit de paridad) al bloque de bits que se quiere proteger. Si la información que se ha de enviar tiene *P* bits, el emisor simplemente incluye un bit adicional de paridad, y transmite *P* + 1 bits.

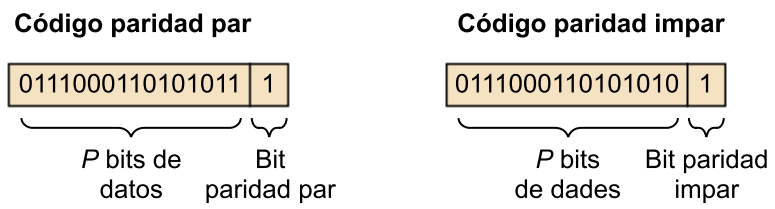
El valor del bit de paridad codifica el número total de unos de la secuencia de *P* + 1 bits (la información original más el bit de paridad). Existen dos esquemas según cómo se codifique el bit de paridad:

- Esquema paridad par: bit de paridad a 1, si el número de unos de la secuencia *P* + 1 es par.

- Esquema paridad impar: bit de paridad a 1, si el número de unos de la secuencia *P* + 1 es impar.

La figura 16 muestra un esquema de paridad par y otro impar, con el bit de paridad simple almacenado en un campo separado:

Figura 16



La operación del receptor es también sencilla con un bit de paridad simple. El receptor sólo necesita contar el número de unos en los *P* + 1 bits recibidos.

Si en la transmisión de la palabra código se produce un único error (un 1 pasa a valer 0 o un 0 pasa a valer 1), la paridad de la palabra código cambiará y no coincidirá con la del bit de paridad. Por lo tanto, se detectará el error. Pero si se

produce un número par de errores de bits, la paridad será la misma y el error no se detectará. Por lo tanto, deducimos que con el bit de paridad el código permite detectar un número impar de bits erróneos.

Cálculo del bit paridad

Para la generación del bit de paridad, los sistemas informáticos utilizan el cálculo de la operación binaria XOR de los bits que se quieren proteger:

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

- Para la paridad par, el bit de paridad se calcula:

$$P = b1 \oplus b2 \oplus ... \oplus bn$$

- Para la paridad impar, el bit de paridad se calcula:

$$P = NOT(b1 \bigoplus b2 \bigoplus \ldots \bigoplus bn)$$

Utilización de la técnica de bit paridad

Cuando se utiliza la paridad como código detector de errores, no debemos imaginar que toda la trama está protegida con un solo bit de paridad. La paridad sólo se utiliza en transmisiones orientadas a carácter asíncronas de bajo rendimiento, en las que las tramas de éstas se hallan formadas por más de un carácter, cada uno con su bit de paridad.

La paridad simple es el código detector de errores que más se utiliza en transmisiones orientadas a carácter. Por ejemplo, la transmisión por el puerto serie de un PC es orientada a carácter. El dispositivo que controla el puerto serie (denominado UART[30]) añade automáticamente un bit de paridad a cada carácter transmitido. Respecto a la recepción, el UART también controla automáticamente que el bit de paridad sea correcto; en caso contrario, se produce una condición de error.

[30]UART es la sigla de *universal asynchronous receiver transmitter*.

Robustez del código de paridad simple

Deduciremos el valor de los tres parámetros, introducidos antes, que definen la robustez del código paridad simple.

a) Para cualquier palabra válida, si se cambia un bit, se obtiene una palabra no válida, y si se cambian dos, se obtiene otra palabra válida. Deducimos, pues, que la diferencia mínima entre dos palabras código válidas es de dos bits y, por lo tanto, que la distancia de Hamming vale 2. En consecuencia, el código es capaz de detectar con probabilidad 1 todas las combinaciones de bits erróneos inferiores a 2 (es decir, el código detecta un bit erróneo, como ya habíamos visto).

b) Dado que el código no detecta una ráfaga de error igual a 2 (dos bits consecutivos erróneos), la capacidad de detección de ráfagas vale 1. Es incapaz de detectar un número par de errores y tampoco permite determinar la posición del bit erróneo.

c) Por último, la probabilidad de que una combinación arbitraria de bits sea aceptada como palabra válida es: $2^{-Q} = 2^{-1} = 0{,}5$, es decir, de los caracteres que tengan muchos bits erróneos, sólo se detectará la mitad.

Por lo tanto, comprobamos que en condiciones de error racheado, la probabilidad de errores no detectados en una trama protegida por una paridad de bit simple es muy alta (se puede aproximar al 50%).

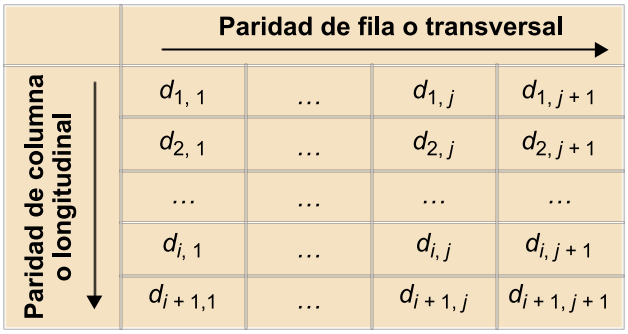
Códigos de paridad bidimensional

Una manera de mejorar la robustez del código detector de errores por medio del bit de paridad consiste en organizar los *P* bits que hay que proteger en una matriz de *i* filas y *j* columnas. Se calcula un valor de paridad para cada fila y para cada columna.

La paridad de las filas se denomina **paridad transversal** (u horizontal), y la de las columnas, **paridad longitudinal** (o vertical). Los $i + j + 1$ bits de paridad resultantes comprenden los bits de detección de error de la trama de enlace de datos.

De este modo, la información se transmite organizada en bloques, con su respectivas paridad longitudinal y transversal. La transmisión del bloque se realiza por filas, de manera que los últimos bits transmitidos son los bits de la paridad longitudinal. La figura 17 muestra una generalización de dos dimensiones del esquema de paridad de bit único.

Figura 17



En la figura:

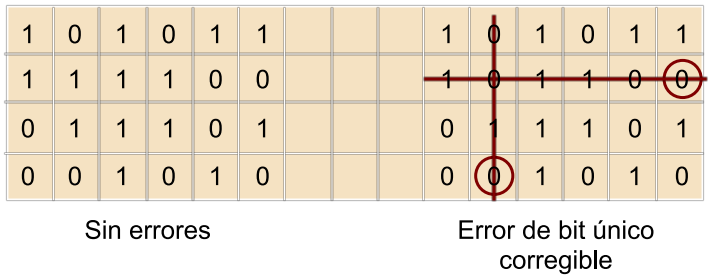
- $d_{i+1,x}$ para $x \in [1,j]$ son paridades longitudinales.
- $d_{x,j+1}$ para $x \in [1,i]$ son paridades transversales.
- $d_{i+1,j+1}$ es la paridad de las paridades transversales resultantes, que coincide con la paridad de las paridades longitudinales. Se denomina bit de cuadre.

Funcionamiento en presencia de errores

Supongamos ahora que ocurre un error de bit único en los P bits originales de información. Con este esquema de paridad de dos dimensiones, la paridad de la fila y la columna que contienen el bit cambiado dará un error. El receptor no sólo podrá detectar el error de un bit simple, sino también utilizar los índices de la fila y de la columna con errores de paridades para identificar, de hecho, el bit que se ha modificado y corregir ese error.

La figura 18 muestra un ejemplo de un bit con valor 1 en la posición (2, 2) que se ha modificado y se ha cambiado a 0.

Figura 18

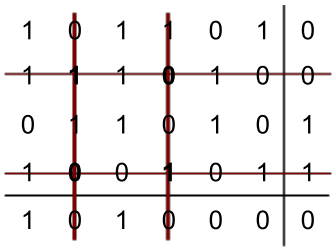


Una vez comprobado que el bit de las paridades transversales coincide con las paridades longitudinales y transversales de la matriz, se puede aislar el valor de la matriz (*i*, *j*) erróneo mediante las paridades filas y columnas que resulten erróneas. Detectaremos el error en los datos originales y lo podremos corregir.

Asimismo, se puede detectar y corregir un error no sólo en los bits originales de información, también en los propios bits de paridad. Eso sí, una combinación de dos errores en un paquete puede ser detectada, pero ya no corregida.

A continuación, investigaremos los casos en los que una combinación de errores no sería detectada. De la definición de paridad deducimos que este código detectará todas las combinaciones de bits erróneos que tengan un número impar de errores en alguna fila o columna. Es decir, no se detectarán las combinaciones de bits erróneos que tengan un número par de errores en todas las filas y columnas simultáneamente. El caso más sencillo es el que muestra la figura 19.

Figura 19. Combinación de errores que no sería detectada



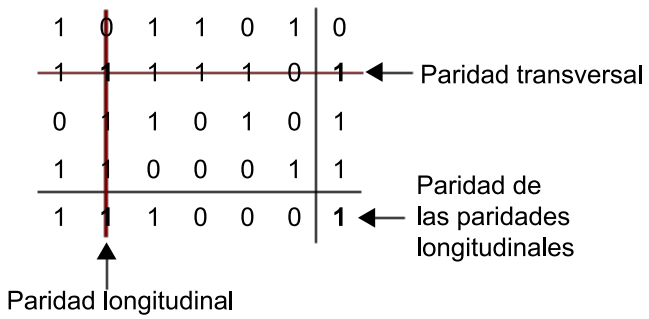
En este caso, el sistema de codificación fallará y se tomará como válida una secuencia de datos con errores.

Robustez del código de paridad bidimensional

Deduciremos el valor de los parámetros que miden la robustez del código paridad bidimensional:

a) Si en un bloque cambia uno de los bits que hay que proteger, cambiarán, además, las paridades transversal, longitudinal y la paridad de las longitudinales: cambian 4 bits; por lo tanto, $D_H = 4$.

Figura 20. Bits que cambian entre dos palabras código válidas consecutivas



b) Para determinar la capacidad de detección de ráfagas debemos encontrar la ráfaga mínima no detectada. A partir de la figura 19 es fácil deducir que la ráfaga mínima no detectada se produce cuando los cuatro bits erróneos son adyacentes, y su tamaño es igual a la longitud de una fila más dos. Así, la capacidad de detección de ráfagas es la longitud de una fila más uno.

c) La probabilidad de que una combinación arbitraria de bits sea aceptada como palabra válida es $1/2^{\text{Longitud fila + Longitud columna} - 1}$.

Utilización del código de paridad bidimensional

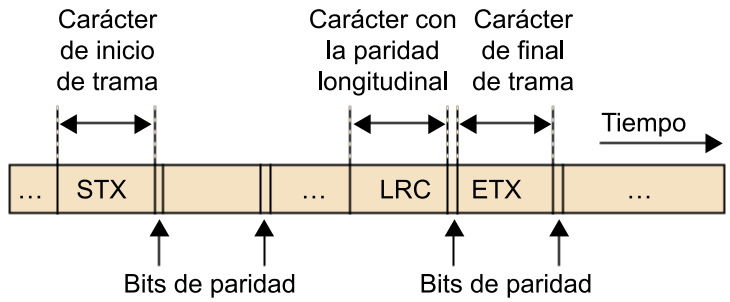
El código de paridad longitudinal y transversal se suele utilizar en transmisiones asíncronas orientadas a carácter. El dispositivo transmisor añade automáticamente un bit de paridad a cada carácter; de esta manera la trama queda formada por un conjunto de caracteres a los que se añade un carácter con la paridad longitudinal.

La figura 21 nos muestra la disposición que acabamos de explicar de los bits de paridad dentro de la trama. A causa de este carácter extra, el código detector de errores se conoce también como LRC[31] o BCC[32].

(31)LRC es la sigla de *longitudinal redundancy check*.

(32)BCC es la sigla de *block check character*.

Figura 21. Transmisión de una trama con los bits de paridad y el carácter de paridad longitudinal LRC



El carácter LRC se suele calcular mediante la operación XOR (paridad par) de los caracteres que se quieren proteger

Es mucho menos habitual que el anterior, a causa, entre otras razones, de su gran ocupación de canal, que en el caso de bloques de 8 × 8 supone una redundancia del 22,2%, si no contamos el bit de cuadre y del 23,4%, incluido.

Métodos de comprobación de sumas

En la técnica de comprobación de sumas, los *P* bits de la secuencia enviada son tratados como una secuencia de enteros de *k* bits. Un método sencillo de comprobación de sumas consiste simplemente en sumar estos enteros de *k* bits y utilizar la suma resultante como bits de detección de errores.

El RFC 1071 discute el algoritmo de comprobación de suma de Internet con detalle. Se implementa para comprobar la integridad y detectar errores en el datagrama de Internet. Pero en su cálculo sólo tiene en cuenta los bytes de la cabecera IP (sólo protege los campos de la cabecera, como la dirección IP origen y destino).

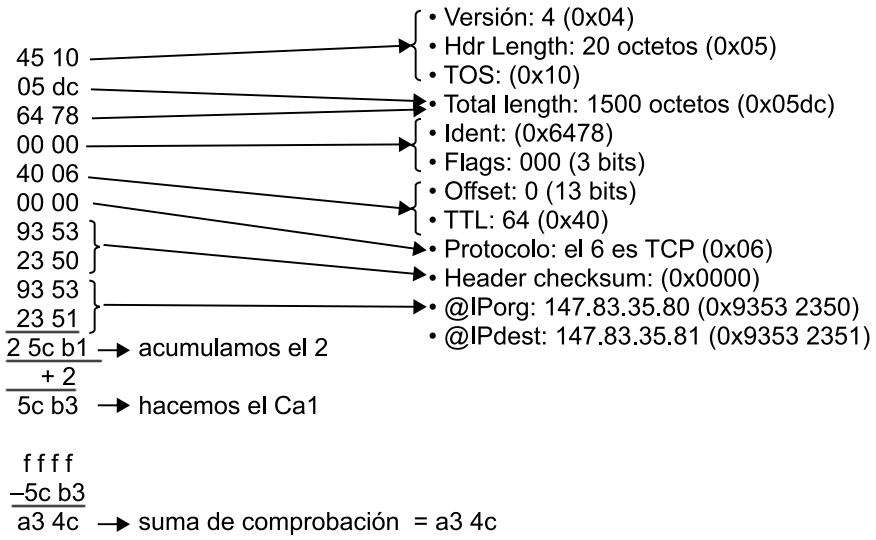
Proceso de cálculo de la suma de comprobación

- Los bytes de la cabecera del datagrama son alineados como palabras de 16 bits.
- Se inicializa la suma de comprobación (resultado de la suma) a 0.
- Se suman las palabras de la cabecera con acarreo.
- Se acumula el acarreo final junto al resultado de la suma.
- Se calcula el complemento a 1 del resultado final (con el acarreo acumulado). El complemento a 1 consiste en cambiar los unos por los ceros y viceversa.

En emisión, se rellena el campo de suma de comprobación de la cabecera del datagrama con el valor obtenido de la suma. El valor de la suma de comprobación se debe recalcular cada vez que se atraviesa un encaminador (ya que hay campos de la cabecera que son mutables, por ejemplo el campo TTL).

En recepción, se realiza la suma teniendo en cuenta el campo de comprobación de suma generado en el emisor. Si el resultado de la suma tiene todos los bits a 1, el datagrama es correcto. Si alguno de los bits está a 0, se indica que ha habido un error.

Figura 22. Ejemplo de cálculo de la suma de comprobación



Códigos de redundancia cíclica

Como hemos comentado, los códigos detectores con bit de paridad están indicados para transmisiones orientadas a carácter. Para transmisiones orientadas a bit no son útiles porque las tiras de bits en las que se podría aplicar la paridad son mucho más largas y perderían efectividad. En lugar del bit de paridad se utilizan los denominados códigos CRC. Se trata de una técnica de detección de errores ampliamente utilizada en las redes de ordenadores actuales.

Los códigos CRC son conocidos como códigos polinómicos puesto que posibilitan ver la secuencia de bits enviados como un polinomio cuyos coeficientes son los valores 0 y 1 en la cadena de bits.

Sea S una secuencia de P bits s_{K-1}, s_{K-2}, ..., s_0, definimos la representación polinomial $S(x)$ de la secuencia S de la siguiente manera:

$$S(x) = s_{p-1}x^{p-1} + s_{p-2}x^{p-2} + s_1x + s_0$$

El objetivo de las potencias x^j es distinguir el peso del bit s_j dentro de la secuencia. Por ejemplo, la representación polinomial de la secuencia 1001001 es: $x^6 + x^3 + 1$.

Los códigos detectores de errores polinomiales se basan en el cálculo de un número binario, conocido como CRC, resultado de una cierta operación matemática efectuada con los bits que se deben proteger. Este número se pone en el campo de control de errores de la trama. En recepción se repite el cálculo y se interpreta que hay o no hay error, en función de si coincide o no con el CRC recibido.

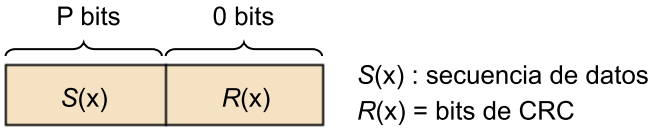
Codificación en emisión

Consideramos que la secuencia inicial S es la formada por los P bits de la trama que queremos proteger, que denominaremos $S(x)$ en su expresión polinómica. Inicialmente, el emisor y el receptor se deben poner de acuerdo, en primer lugar, en un patrón de $Q + 1$ bits, conocido como polinomio generador, representado como $G(x)$. A los P bits de la trama original el emisor le añade una secuencia de Q bits conocidos como el CRC de la trama y representado como $R(x)$. Estos bits son obtenidos como el resto de la siguiente división polinómica en módulo 2:

$$R(x) = resto\left(\frac{S(x)\cdot x^Q}{G(x)}\right)$$

En módulo 2, el resto es igual a la suma, que a la vez es la operación XOR binaria. La trama transmitida estará formada por los bits *P* iniciales y el CRC de *Q* bits.

Figura 23



La expresión polinomial de la trama transmitida será:

$$S'(x) = S(x) \times x^Q + R(x)$$

Ejemplo de cálculo del CRC

Supongamos que la secuencia de bits que hay que proteger es 11001, con un CRC de tres bits, y que el polinomio generador es $G(x) = x^3 + 1$. Tenemos que $S(x) = x^4 + x^3 + 1$; por lo tanto, $S(x) \cdot x^3 = x^7 + x^6 + x^3$. Mientras que la división en módulo 2 de $S(x) \cdot x^Q$ / $G(x)$ es:

$$\begin{array}{l|l} x^7 + x^6 + x^3 & x^3 + 1 \\ \hline x^7 + x^4 & x^4 + x^3 + x \\ \hline x^6 + x^4 + x^3 & \\ x^6 + x^3 & \\ \hline x^4 & \\ x^4 + x & \\ \hline x & \end{array}$$

Se obtiene que $R(x) = x$. Por lo tanto, el CRC que habría que añadir sería 010.

Comprobación en recepción

El patrón de bits resultante *P* + *Q* (interpretado como un número binario) es exactamente divisible por *G*, utilizando aritmética de módulo 2 (considerando que las sumas y las restas que se realizan son sin acarreos ni adeudos). Esto se utilizará en recepción para comprobar la integridad de los datos.

El receptor sólo debe dividir los *P'* + *Q'* bits recibidos entre *G*(*x*). Si el resto no es cero, el receptor sabe que ha ocurrido un error; en el caso contrario, se acepta que es correcto.

Robustez de los códigos detectores CRC

Las propiedades del código CRC dependen del polinomio generador. Sin embargo, generalmente se puede demostrar que si elegimos un polinomio generador adecuado de grado *Q* (es decir, con un CRC de *Q* bits):

a) La distancia de Hamming del código es mayor o igual a 4.

b) La capacidad de detección de ráfagas de error es menor o igual a *Q* (es decir, que se pueden detectar todos los bits de error consecutivos de Q bits o menos).

c) La probabilidad de que una combinación arbitraria de bits sea aceptada como palabra válida vale 2^{-Q}. Asimismo, cada uno de los estándares de CRC puede detectar cualquier número impar de errores en bits.

Polinomios generadores estandarizados

El grado del polinomio generador $G(x)$ no es arbitrario, sino que está determinado por el número de bits que se desea que tenga el CRC. El residuo de la división por un polinomio de grado P es un polinomio de grado menor o igual a $P - 1$. Si queremos que el CRC tenga Q bits, es decir, que su representación polinomial tenga un grado menor o igual a $Q - 1$, deberemos elegir un polinomio generador de grado Q. En otras palabras, el grado del generador debe ser igual al número de bits del CRC.

La eficacia del sistema depende del polinomio generador elegido. Hay polinomios generadores muy utilizados que han sido estandarizados internacionalmente, de 8, 12, 16 y 32 bits. Por ejemplo, se utiliza un CRC de 8 bits para proteger la cabecera de 5 bytes en las celdas ATM. A continuación, podemos ver algunos polinomios generadores CRC estandarizados:

- $CRC-12=x^{12}+x^{11}+x^3+x^2+x^1+1$
- $CRC-16=x^{16}+x^{15}+x^2+1$
- $V41-ITU-T=x^{16}+x^{12}+x^5+1$
- $CRC-32=x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x^1+1$

El estándar de 32 bits, CRC-32, que se ha adoptado en un cierto número de protocolos IEEE[33] de nivel de enlace, utiliza el generador de *G*(CRC-32) = 100000100110000010001110110110111.

(33)IEEE es la sigla de Institute of Electrical and Electronic Engineers.

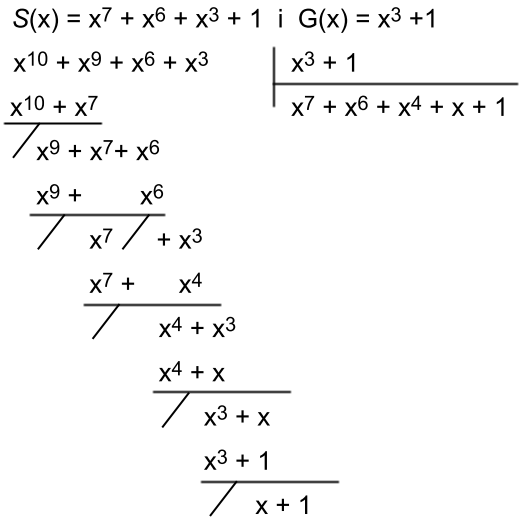
Los códigos de 16 bits capturan todos los errores simples y dobles, todos los errores en los que el número de bits afectados es impar, todos los errores en ráfaga con tamaño de ráfaga menor o igual a 16, el 99,997% de los errores de ráfaga de 17 bits y el 99,998% de los de 18 bits o mayores.

Ejemplo de uso de polinomio

Se desea transmitir el mensaje 11001001 protegiéndolo de errores mediante el uso del polinomio CRC: $x^3 + 1$.

a) Determinar el mensaje que hay que transmitir por el nodo emisor.

Realizamos la siguiente división polinomial $S(x)\cdot x^3 / G(x)$, donde:



La división $S(x)\cdot x^3 / G(x)$ da como cociente $C(x) = x^7 + x^6 + x^4 + x + 1$ y el resto $R(x) = x + 1$.

Por tanto, se enviará la secuencia: 11001001 + 011 (se añadirán 3 bits de CRC).

Podemos comprobar que la secuencia $S(x) \cdot x^3 + R(x)$ es divisible entre $G(x)$.

b) Si se recibe el mensaje 01001001, debido a que se invierte el bit más significativo, ¿cuál sería el resultado del cálculo de CRC en recepción? ¿Cómo se sabe en recepción que ha ocurrido un error?

Se debe realizar la división del polinomio representado por la cadena de bits recibida (01001001 + 011 = $x^9 + x^6 + x^3 + x + 1$) entre el polinomio generador. Debería salir un resto 0 para detectar que todo es correcto. $x^9 + x^6 + x^3 + x + 1 : x^3 + 1$ da de resto x, por lo que se ha producido un error.

Ejercicio

7. ¿Por qué se utiliza CRC en nivel de enlace y suma de comprobación en nivel de red y transporte?

Solución ejercicio 7

La detección de errores en la capa de enlace está implementada en el hardware dedicado de los adaptadores, que pueden realizar rápidamente las más complejas operaciones de CRC.

La capa de red y de transporte está implementada en software en un equipo final (*host*), como parte del sistema operativo del equipo final. La suma de comprobación es fácil de implementar en software, ya que es un esquema de detección de errores simple y rápido. Sin embargo, proporcionan una protección relativamente débil contra los errores si se comparan con CRC.

4.2. Estrategias de corrección de errores

Existe la posibilidad de instaurar un tipo de código con la suficiente redundancia que, además de detectar errores, permita corregir algunos bits errados en el receptor, sin necesidad de solicitar una repetición de la transmisión. A este tipo de códigos se les denomina autocorrectores y son eficaces siempre y cuando los errores no se presenten en ráfagas de tamaño superior a un máximo admisible.

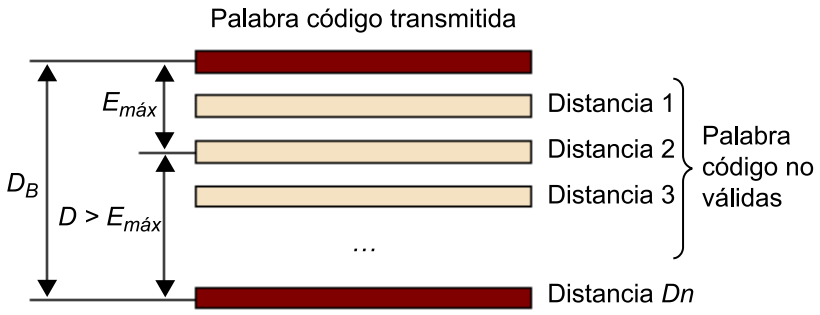
La técnica de utilizar en el receptor un código corrector de errores para detectar y recuperar errores (en lugar de solicitar la retransmisión de la trama) se conoce como técnica de corrección de errores hacia adelante (FEC[34]).

[34]FEC es la sigla de *forward error correction*.

Esta técnica se utiliza habitualmente en el almacenamiento de audio y en dispositivos de reproducción, como los CD de audio. En la inicialización de la red se pueden utilizar técnicas FEC por sí solas, o junto con las técnicas ARQ que hemos examinado en el módulo 3. Las técnicas FEC son valiosas porque pueden disminuir el número de retransmisiones del emisor requeridas. Y, lo que es quizá más importante, permiten la corrección inmediata de errores en el receptor. Esto evita esperar el retraso de propagación de ida y vuelta necesario para que el emisor reciba un paquete NAK y para propagar hacia atrás el paquete retransmitido al receptor (una ventaja potencialmente valiosa para aplicaciones de tiempo real).

Para realizar un análisis más formal de los códigos correctores se utiliza el concepto de distancia de Hamming, que ya hemos introducido. En caso de error, la corrección consiste en suponer que la palabra código transmitida es la palabra código válida más próxima a la palabra recibida, según el concepto de distancia (criterio de la distancia mínima). Por lo tanto, será la que tenga menos bits de diferencia.

Figura 24. Código corrector según el criterio de la distancia mínima



La figura 24 es una representación gráfica de la idea que acabamos de exponer. En esta figura podemos ver una posible palabra código transmitida y, agrupadas, todas las palabras con el número de bits de diferencia (o sea, que distan): 1, 2... hasta las palabras válidas más próximas a D_H bits de distancia. Si se recibe una de las tramas que se encuentra a una distancia 1, 2,..., $E_{máx}$, es decir, una de las palabras que no tienen otra palabra válida más próxima que la transmitida, el código corregirá el error.

Para saber cuántos bits es capaz de corregir el código con probabilidad 1, supongamos que, en la figura, D_H es la distancia mínima entre dos palabras válidas (la distancia de Hamming del código).

Sea $E_{máx}$ el número de bits erróneos, de la figura deducimos que el criterio de distancia mínima corregirá el error si $E_{máx} < D$, donde $D = D_H - E_{máx}$.

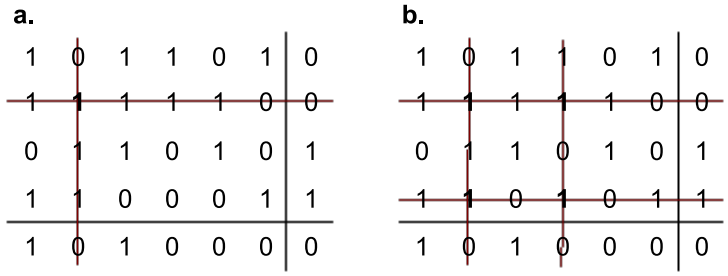
En definitiva, si la distancia de Hamming de un código es D_H, utilizando el **criterio de la distancia mínima** se puede corregir cualquier combinación de $E_{máx}$ bits erróneos que cumpla:

$$E_{máx} < D_H / 2$$

4.2.1. Corrección de errores en códigos de paridad bidimensional

Hemos visto que los códigos de paridad bidimensional permiten corregir cualquier error de un solo bit buscando la fila y la columna con la paridad cambiada (tal como muestra el esquema *a* de la figura 25). Sin embargo, si el error se produce en dos bits (esquema *b* de la figura), el código ya no es capaz de corregir el error.

Figura 25. Corrección de bits con un código con paridad transversal y longitudinal



En esta figura, los bits erróneos están marcados en negro. Si el error se produjera en los bits marcados en blanco, se tendría el mismo error longitudinal y transversal. Así, el código de paridad longitudinal y transversal no es capaz de averiguar cuál de las dos posibilidades debería corregirse.

La distancia de Hamming de los códigos con paridad transversal y longitudinal es igual a 4. Por lo tanto, se pueden corregir $E_{máx} < 2$, es decir, 1 bit, tal como habíamos deducido antes. Un código con $D_H = 5$ puede corregir $E_{máx} < 2{,}5$; es decir, 2 bits.

4.2.2. Códigos de Hamming

Son un tipo de códigos de control de paridad en los que los dígitos de paridad se intercalan en la palabra, de manera que pueden identificar los posibles bits erróneos. Normalmente, presentan distancia mínima 3 (corrigen un error) y pueden utilizar la paridad par o impar. A continuación, presentaremos las reglas de composición para palabras de código Hamming de distancia 3 y paridad par:

- Si la palabra original de datos tiene m bits, se necesitarán *h* bits de paridad. Se ha de cumplir que $2^h \geq m + h + 1$.

- Los bits se numerarán de izquierda a derecha, comenzando por 1.

- En las posiciones que son potencias de dos (1, 2, 4,..., 2*P*), se intercalarán los bits de paridad y se dejará el resto para bits de datos.

- Cada bit de paridad par se calcula a partir de una serie de bits de datos, pero no a partir de ningún otro de paridad.

- Como norma general, un bit de datos *bn* es comprobado por los bits de paridad b_i, b_j,..., b_k de modo que $n = i + j + ... + k$. Dicho de otro modo, un bit de datos es comprobado por aquellos bits de paridad cuyas posiciones son la descomposición en potencias de dos distintas, de la posición del bit de datos.

Por ejemplo:

- b_{18} será comprobado por b_{16} y b_2, ya que $18 = 16 + 2 = 2^4 + 2^1$
- b_{22} será comprobado por b_{16}, b_4 y b_2, ya que $22 = 16 + 4 + 2 = 2^4 + 2^2 + 2^1$
- b_{32} es bit de paridad, ya que su posición es una potencia de 2, $32 = 2^5$

El emisor envía la palabra de código al receptor (compuesta de datos y paridad), éste comprueba las ecuaciones de paridad sobre los datos recibidos y, en el hipotético caso de que un bit haya sufrido un cambio, puede detectar su posición si restaura su valor inicial.

A continuación, desarrollaremos las ecuaciones de paridad para 4 bits de datos y 3 de paridad ($2^3 = 8 = 4 + 3 + 1$). En cada ecuación aparece un único bit de paridad junto a los bits de datos que controla.

Ecuaciones de paridad para 4 bits de datos

a) $0 = b_1 \oplus b_3 \oplus b_6 \oplus b_7$
b) $0 = b_2 \oplus b_3 \oplus b_6 \oplus b_7$
c) $0 = b_4 \oplus b_5 \oplus b_6 \oplus b_7$

Para comprender el funcionamiento del método utilizaremos un ejemplo.

Ejemplo

Generar la palabra de código para transmitir los datos 1001: hay que intercalar los bits de paridad en las posiciones 1, 2 y 4, de lo que resulta $b_1\ b_2\ 1\ b_4\ 0\ 0\ 1$. Para que cumplan las ecuaciones los bits de paridad serán $b_1 = 0$, $b_2 = 0$, $b_4 = 1$, y la palabra completa 0011001.

En el caso de 3 bits de paridad y 4 de datos, la redundancia será:

$$R = (3/7) \times 100 = 42{,}86\%$$

Ésta es una redundancia alta, pero se suele admitir en ciertos casos, a cambio del poder corrector de los códigos Hamming.

Supongamos ahora que el emisor envía la palabra calculada al receptor y que a éste le llega 0011000. Como se puede apreciar, se ha producido un error en un bit. Ahora bien, ¿es posible conocer dónde se ha generado ese error a partir del dato recibido? Para ello el receptor ha de comprobar las ecuaciones con los valores recibidos:

$0 \oplus 1 \oplus 0 \oplus 0 = 1$: no se cumple *a*
$0 \oplus 1 \oplus 0 \oplus 0 = 1$: no se cumple *b*
$1 \oplus 0 \oplus 0 \oplus 0 = 1$: no se cumple *c*

Por lo tanto, el fallo estará en el bit que aparece en *a*, *b* y *c*, es decir, en el 7.°.

Como es fácil comprobar, si se leen los resultados de las ecuaciones en orden *c*, *b*, *a* y se interpreta el valor resultante como un número codificado en binario, éste nos indica la posición del bit erróneo. En este caso, del ejemplo: $c = 1$, $b = 1$, $a = 1$, luego $cba = 111 = 7_{10}$.

Con frecuencia, los códigos Hamming se utilizan sobre 7 bits de datos (1 carácter ASCII). Para ello es necesaria la utilización de 4 bits de paridad. Las ecuaciones correspondientes serán:

a) $0 = b_1 \oplus b_3 \oplus b_5 \oplus b_7 \oplus b_9 \oplus b_{11}$
b) $0 = b_2 \oplus b_3 \oplus b_6 \oplus b_7 \oplus b_{10} \oplus b_{11}$
c) $0 = b_4 \oplus b_5 \oplus b_6 \oplus b_7$
d) $0 = b_8 \oplus b_9 \oplus b_{10} \oplus b_{11}$

Cálculo del tamaño de los códigos Hamming

¿Cuál es la razón de que si la palabra original de datos tiene *m* bits se necesiten *h* bits de paridad, de manera que se deba cumplir que $2^h \geq m + h + 1$?

La respuesta es que para codificar la posición de 1 bit erróneo en $n = m + h$ bits de palabra de código, las ecuaciones de control deberán poder detectar $n + 1$ resultados distintos. La suma de una unidad corresponde al resultado ausencia de error (no se ha producido error en ningún bit).

Al codificarse en base dos, el número de bits de paridad (ecuaciones de paridad) necesario será el que cumpla que: $2^h = n + 1 = m + h + 1$.

Por lo tanto, al aplicar la definición de logaritmo a la anterior ecuación se cumplirá que $h = \log_2(n + 1)$ y, dado que resultados decimales no tienen sentido (no es posible tomar medio bit de paridad), se tomará siempre el siguiente entero por exceso. Esto se podría representar como:

$$h = \lceil \log_2(n + 1) \rceil$$

A los códigos Hamming que cumplen la relación $2^h = n+1 = m+h+1$ se les denomina óptimos, en contraste con aquellos otros que utilizan menos bits de datos de los que podrían usar con los bits de control disponibles. Así, el código que utiliza 7 bits de datos y 4 de paridad no es óptimo, pues con 4 bits de paridad se podrían controlar $2^4 = 16$ resultados, correspondientes a 15 bits distintos, es decir, se podrían utilizar hasta 15 – 4 = 11 bits de datos.

Ejercicios

8. ¿Qué tamaño mínimo debe tener una palabra de código Hamming con 15 bits de datos capaz de corregir un error?

Solución ejercicio 8

$m = 15$; por tanto, se va probando hasta encontrar el valor más pequeño de h, que hace que $2^h \geq 15 + h + 1$. Dicho valor es 5, luego $n = 15 + 5 = 20$.

9. ¿Es óptimo el código anterior? En caso de que no lo sea, indicad qué habría que hacer para convertirlo en óptimo.

Solución ejercicio 9

No es óptimo. Para que fuera óptimo debería cumplir:

$$2^h = n + 1 = m + h + 1: 2^5 = 32 \neq m + b + 1 = 15 + 5 + 1 = 21$$

Para que fuera óptimo se debería verificar que:

$32 = m + 5 + 1$
$m = 32 – 5 – 1 = 26$ bits de datos

10. Explicad la configuración y las ecuaciones de paridad de un código óptimo con 4 bits de paridad.

Solución ejercicio 10

$b = 4$, $2^h = 2^4 = 16 = m + b + 1$

$m = 16 – h – 1 = 16 – 4 – 1 = 11$, es decir, la configuración debe ser 11 bits de datos y 4 de paridad, que hacen un total de 15 bits. Las ecuaciones de paridad serán:

a) $0 = b_1 \oplus b_3 \oplus b_5 \oplus b_7 \oplus b_9 \oplus b_{11} \oplus b_{13} \oplus b_{15}$
b) $0 = b_2 \oplus b_3 \oplus b_6 \oplus b_7 \oplus b_{10} \oplus b_{11} \oplus b_{13} \oplus b_{15}$
c) $0 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 \oplus b_{12} \oplus b_{13} \oplus b_{14} \oplus b_{15}$
d) $0 = b_8 \oplus b_9 \oplus b_{10} \oplus b_{11} \oplus b_{12} \oplus b_{13} \oplus b_{14} \oplus b_{15}$

11. ¿Cuál es el nivel de redundancia de un código Hamming con 7 bits de datos y 4 de paridad?

Solución ejercicio 11

$$\%_{Redund} = \frac{Bits\ de\ control}{Bits\ totales} \times 100 = \frac{4}{11} \times 100 = 36{,}36\%$$

Como vemos, el porcentaje de redundancia es bastante elevado. Éste es uno de los motivos de que tales códigos se utilicen únicamente en circuitos en los que

la implantación de sistemas de reenvío supondría un coste tiempo/canal muy elevado.

4.3. Estrategias de retransmisión de tramas

Ved también

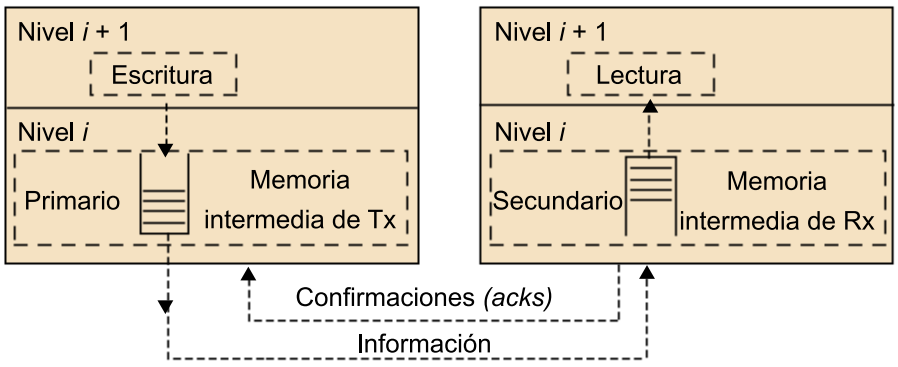
Podéis ver el funcionamiento de los protocolos ARQ en el módulo "La capa de transporte de datos" de esta asignatura.

El nivel de enlace puede implementar técnicas de retransmisión de tramas basadas en los protocolos ARQ. De hecho, estas técnicas pueden implementarse tanto en el nivel de enlace (por ejemplo, los protocolos XMODEM, YMODEM y ZMODEM) como en el nivel de transporte (protocolo TCP).

Hemos visto que el principal objetivo de las técnicas ARQ es que la información transmitida llegue sin errores, sin duplicaciones y en el mismo orden en el que se envía. Básicamente, un protocolo ARQ retransmite la información que no llega, o que llega con errores al nodo receptor. El receptor envía tramas de confirmación al emisor para informarle de que ha recibido correctamente las tramas de información.

4.3.1. Elementos de un protocolo ARQ

Figura 26



- Canal bidireccional. Para un sistema de retransmisiones es necesario que la comunicación sea bidireccional (semidúplex o dúplex).

- Primario: entidad que transmite la información (emisor).

- Secundario: entidad que recibe/consume la información (receptor). Envía los mensajes de confirmación[35]. En la práctica, las dos entidades se pueden comportar como primaria o como secundaria (*piggybacking*).

- Memoria intermedia de transmisión: donde se guarda la información que se debe enviar o que se ha enviado y que todavía no ha sido confirmada por el secundario.

- Memoria intermedia de recepción: memoria en el secundario donde se guarda la información recibida hasta que la lee el nivel superior.

(35)En inglés, *acks*; abreviatura de *acknowledgements*.

4.3.2. Funcionamiento básico de un protocolo ARQ

- El primario o transmisor envía tramas de información y las va guardando en una memoria intermedia de transmisión.

- Si la memoria de transmisión se llena, el primario bloquea la escritura del nivel superior hasta que reciba confirmaciones de tramas de información.

- A medida que llegan confirmaciones del secundario, el primario borra la información confirmada de la memoria intermedia de transmisión y deja espacio para que el nivel superior pueda escribir más información.

- En caso de error, el primario puede retransmitir la información, dado que la tiene almacenada en su memoria intermedia de transmisión.

4.3.3. Algoritmos de retransmisión ARQ

Existen 3 técnicas ARQ:

1) *Stop & wait* (técnica Idle RQ; se usa en transmisiones orientadas a carácter).

2) *Go-back-N*.

3) Retransmisión selectiva (estas dos últimas técnicas Continuous Rq se usan sobre todo en transmisiones orientadas a bit).

4.3.4. Eficiencia de los protocolos ARQ

Los protocolos ARQ se evalúan mediante el concepto de eficiencia:

$$E = \frac{V_{ef}}{V_t} = \frac{Duraciónde\ transmisiónde\ información}{Tiempodetransmisión} = \frac{T_{Trama}}{T_{Ciclo}}$$

En la siguiente tabla podemos ver las formulas de eficiencia de los protocolos ARQ en presencia y ausencia de errores:

Eficiencia	Sin errores	Con errores
Stop & wait	$\frac{1}{1+2a}$	$\frac{1}{N_t(1+2a)}$
Go-back-N	100%	$\frac{1}{N_t(1+2a)-2a}$
Retransmisión selectiva	100%	$\frac{1}{N_t}$

N_t es el número medio de transmisiones necesarias para la transmisión con éxito de una trama y a es la relación entre tiempo de propagación y tiempo de trama:

$$N_t = \frac{1}{(1-P_T)} = \frac{1}{P_{Trama\ sin\ errores}} = \frac{1}{(1-P_{Bit})^L} \qquad a = \frac{T_{Prop}}{T_{Trama}}$$

4.3.5. *Piggybacking*

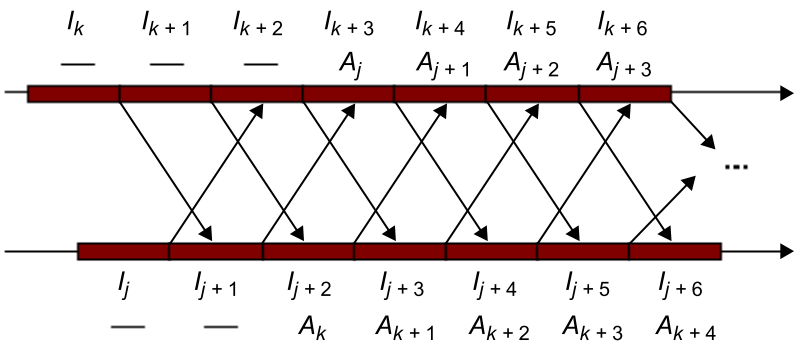
Se trata de una técnica que podemos encontrar en un protocolo de nivel de enlace. Hasta este momento, siempre hemos considerado que había una estación que transmitía tramas de información (el primario) y otra que las confirmaba (el secundario). Sin embargo, se dan casos en los que las dos estaciones intercambian tramas de información recíprocamente y, por lo tanto, actúan como primario y secundario al mismo tiempo y deberán alternar las tramas de información con las confirmaciones.

Las tramas de confirmación suelen tener un tamaño pequeño, ya que la única información relevante es la del identificador de la trama que confirman. La mayor parte de los bits de estas tramas la ocupan los campos de control (indicadores de sincronización, CRC, etc.).

Si las dos estaciones envían información, nos interesará conseguir que la eficiencia en los dos sentidos sea tan alta como se pueda. Una manera de aumentar la eficiencia en esta situación es incorporar las confirmaciones a las tramas de información (de este modo, nos ahorramos la transmisión de los otros campos de control de las confirmaciones).

La manera de hacerlo consiste en destinar un campo de la trama de información al identificador de la trama que se quiere confirmar del primario contrario. Esta técnica se conoce como ***piggybacking***. La figura 27 muestra el diagrama de tiempo que se obtendría si se emplea la técnica del *piggybacking*.

Figura 27. Confirmación de las tramas con paridad *piggybacking* en un protocolo de transmisión continua



5. Control de flujo

El objetivo del control de flujo es la adaptación de la velocidad de transmisión eficaz entre el transmisor, o primario, y el receptor, o secundario, de modo que siempre haya recursos disponibles y no se produzca pérdida de información.

Generalmente, el receptor establece una zona de almacenamiento temporal o memoria intermedia en la que va acumulando las tramas recibidas por el enlace, ya que necesita un cierto tiempo para su procesado (para comprobar errores, ordenar por número de secuencia, desencapsular tramas, enviar al nivel superior, que puede estar ocupado en ese momento, etc).

Si no existiesen procedimientos para el control de flujo y un nodo recibiera tramas a una tasa superior a lo que puede procesarlas, la memoria intermedia temporal del receptor se desbordaría y se perderían tramas. Un protocolo de la capa de enlace con control del flujo evitaría que el nodo emisor sature la memoria intermedia del nodo receptor y se pierda información.

A continuación se tratan diferentes mecanismos de control de flujo implementados en el nivel de enlace.

5.1. Mecanismo de control de flujo X-ON / X-OFF

Se utiliza en algunas transmisiones entre dispositivos informáticos (ordenadores, impresoras, etc.) orientadas a carácter. Básicamente, este protocolo utiliza dos caracteres para controlar el flujo:

- Carácter XON, código 17 ASCII.
- Carácter XOFF, código 19 ASCII.

Cuando el receptor del mensaje desea que el emisor detenga el flujo de datos, manda un carácter XOFF (carácter de pausa) al emisor en el que le indica que su memoria intermedia no admite más caracteres. Cuando el transmisor recibe un carácter XOFF, se bloquea y queda a la espera de recibir el carácter de activación XON para reanudar la transmisión. Este carácter lo manda el receptor cuando tiene suficiente espacio en su memoria intermedia de recepción.

Este mecanismo funciona muy bien cuando se trata de transmitir ficheros de texto, ya que los caracteres XON y XOFF no forman parte de los caracteres usados normalmente en este tipo de ficheros.

5.2. Mecanismo de control de flujo entre un PC y un módem conectado al puerto serie

El puerto serie utiliza el protocolo de nivel físico fuera de banda RS232. Este protocolo tiene dos líneas que sirven para controlar el flujo de datos: RTS[36] y CTS[37].

(36)RTS es la sigla de *request to send*.

(37)CTS es la sigla de *clear to send*.

Normalmente, el puerto serie se configura con una velocidad de transmisión mayor de la que puede conseguir el módem a través de la línea telefónica. El módem tiene una memoria intermedia de transmisión donde se guarda la información que transmite a través de la línea telefónica.

Cuando el PC tiene datos preparados para transmitir al módem, activa la línea RTS. Si el módem activa la línea CTS, el PC le envía información a una velocidad mayor de la que puede enviar el módem a la línea telefónica. Por lo tanto, la memoria intermedia de transmisión del módem se llena. Cuando llega a un cierto umbral, el módem desactiva la línea CTS y la vuelve a activar cuando la memoria intermedia se vacía. De esta manera, el módem siempre tiene información lista para transmitir a través de la línea telefónica y puede aprovechar al máximo su velocidad de transmisión.

5.3. Mecanismo de control del protocolo ARQ *stop & wait*

Es un mecanismo de control de flujo inherente a su funcionamiento. En el envío de cada trama existe una adaptación implícita en las velocidades del emisor y del receptor, que no se puede sobrepasar por el modo de trabajar del protocolo.

El primario no puede enviar otra nueva trama si no recibe la confirmación de la anterior. Por lo tanto, para conseguir disminuir la velocidad de transmisión del primario, el secundario sólo debe retrasar el envío de las confirmaciones. Recordad que este protocolo sólo mantiene en vuelo una única trama sin confirmar.

5.4. Mecanismo de control de los protocolos ARQ de transmisión continua

Los protocolos de transmisión continua ARQ no sólo se utilizan para la recuperación de errores, sino también para el control de flujo. Para ello, utilizan el concepto de ventana deslizante[38]. Los protocolos que utilizan este mecanismo reciben el nombre de protocolos de ventana.

(38)En inglés, *sliding window*.

Ved también

Podéis ver el funcionamiento de los protocolos ARQ en el módulo "La capa de transporte de datos" de esta asignatura.

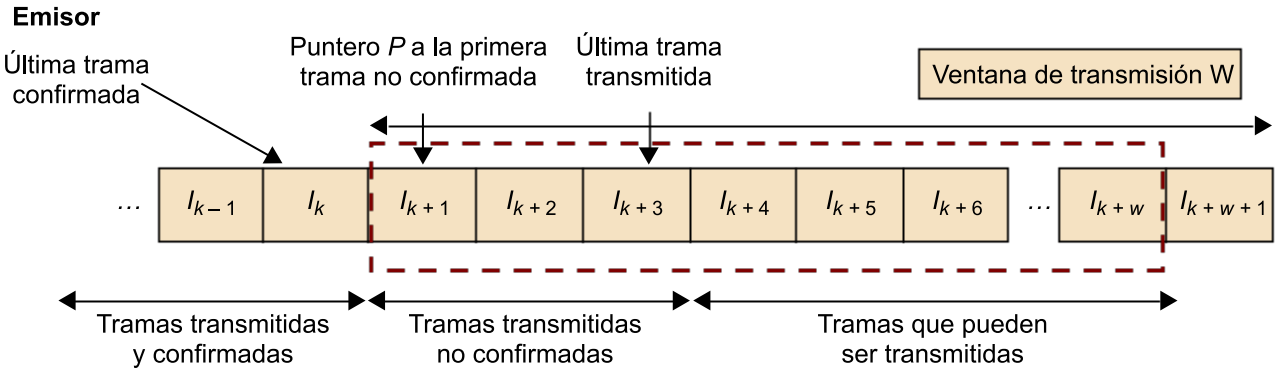
No se debe confundir el concepto de ventana deslizante con el de memoria intermedia de transmisión (o recepción), aunque se utilice la ventana para definir el tamaño de dicha memoria intermedia. El mecanismo de la ventana deslizante se monta sobre la memoria intermedia de transmisión y se desplaza sobre ella a medida que llegan las confirmaciones de las tramas.

A continuación, refrescamos los conceptos de ventana de transmisión y ventana de recepción.

Ventana de transmisión

La ventana de transmisión marca el número de tramas que se pueden transmitir sin confirmar en el primario. La figura 28 ilustra el funcionamiento de un protocolo de ventana en transmisión.

Figura 28. Funcionamiento de un protocolo de ventana en transmisión



En esta figura suponemos que las confirmaciones son acumulativas, es decir, quedan confirmadas todas las tramas con número de secuencia menor o igual a la última confirmada.

El primario puede enviar hasta W tramas de información sin confirmar, que quedan almacenadas en la memoria intermedia de transmisión. El parámetro W es el tamaño de la ventana de transmisión. Si I_k es la última trama confirmada, el emisor sólo puede transmitir hasta la trama I_{k+w}.

Básicamente, el primario mantiene un puntero P en la primera trama no confirmada. Antes de cada transmisión (siempre que lo permita el nivel inferior), evalúa la siguiente diferencia:

- Si (*Número de secuencia de trama que transmitir – Número de secuencia de trama P no confirmada*) < W = TRANSMITE.
- Si (*Número de secuencia de trama que transmitir – Número de secuencia de trama P no confirmada*) ≥ W = SE PARA.

En esta situación, para hacer control de flujo en un protocolo de ventana, basta que el secundario deje de enviar confirmaciones. En el ejemplo anterior, si no llegan más confirmaciones después de transmitir la trama I_{k+w}, el primario agotará la ventana y se parará.

Cuando llegan confirmaciones de nuevas tramas, el índice P que apunta a la primera trama no confirmada (y, por lo tanto, la ventana de tramas que se pueden transmitir) se actualiza y avanza, lo que permite la transmisión de nuevas tramas.

Ventana de recepción

La ventana de transmisión no sólo permite dimensionar el tamaño de la memoria intermedia de transmisión, sino también la memoria intermedia de recepción.

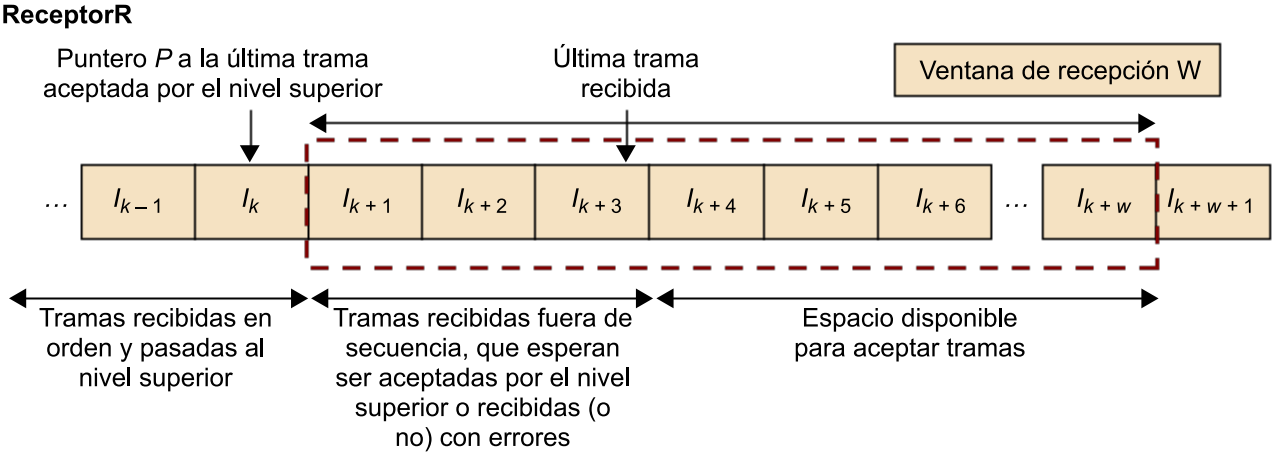
Por analogía con la ventana de transmisión, la ventana de recepción se define como el número máximo de tramas que debe almacenar el secundario. Asumiendo que el nivel de enlace saca de la ventana de recepción todas las tramas con número de secuencia

anterior, que han sido recibidas correctamente, en orden y aceptadas por el nivel superior, podemos encontrar los siguientes tipos de tramas en la ventana de recepción:

- Tramas recibidas sin errores y en orden que no pueden ser aceptadas momentáneamente por el nivel superior.
- Tramas recibidas sin errores, pero fuera de orden, que se deben reordenar.
- Tramas recibidas con errores y, por lo tanto, descartadas.
- Tramas no recibidas porque se han perdido.

En la figura 29 podemos ver un ejemplo de ventana de recepción.

Figura 29



La trama I_k y todas las tramas anteriores han sido recibidas sin errores y en orden por el secundario y han sido aceptadas por el nivel superior. Por lo tanto, el nivel de enlace las ha borrado de la ventana de recepción. Las únicas tramas que puede tener que confirmar y ordenar el secundario son las tramas que van desde la I_{k+1} hasta la I_{k+w}, las únicas que el primario ha sido autorizado a transmitir.

Observamos que el valor máximo de la ventana de recepción será W. Éste es el tamaño de la ventana de recepción en un protocolo ARQ con retransmisión selectiva, que coincide con el tamaño de la ventana de la ventana de transmisión. El problema de la reordenación sólo tiene sentido en el caso del protocolo ARQ de retransmisión selectiva. En los protocolos *stop & wait* y *go-back-N* basta con que la ventana de recepción sea igual a 1. En la siguiente tabla podemos ver un resumen del tamaño de las ventanas en los tres protocolos:

Tamaño de las ventanas de transmisión y recepción		
Protocolo	**Ventana de transmisión**	**Ventana de recepción**
Stop & wait	1	1
Go-back-N	*W*	1
Retransmisión selectiva	*W*	*W*

5.5. Ventana óptima

Hemos visto que, en ausencia de errores, la eficiencia de los protocolos de transmisión continua es del 100%, gracias al hecho de que el primario no permanece nunca parado transmitiendo y esperando confirmaciones. En un protocolo de ventana, esta condición puede no darse si el tamaño de la ventana

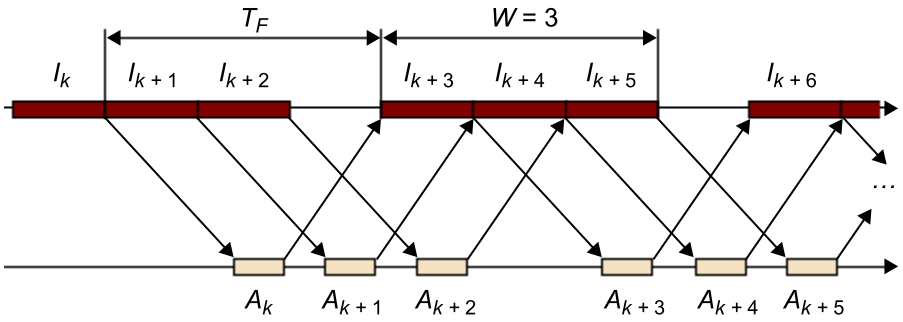
no es lo suficientemente grande, ya que el transmisor puede llegar a bloquearse en espera de recibir las confirmaciones que liberen su ventana, lo que se traduce en una bajada del rendimiento o la eficiencia del protocolo.

Por otro lado, si la ventana del protocolo es muy grande, puede ser un inconveniente, ya que las memorias intermedias de transmisión y recepción han de dimensionarse para poder almacenar un número de tramas igual al de la ventana. Además, no se experimenta ninguna mejoría en rendimiento por el hecho de que sea muy grande.

La **ventana óptima** (W_{opt}) se define como la ventana deslizante mínima que permite una eficiencia del protocolo del 100%.

Podemos estudiar el esquema de la figura 30.

Figura 30. Parada del primario en caso de ventana pequeña



Muestra un ejemplo en el que la ventana de transmisión vale $W = 3$, por lo tanto, se pueden transmitir tres tramas sin confirmar. Como $T_{ciclo} > T_w$, el primario permanece un cierto tiempo bloqueado, esperando la llegada de confirmaciones que permita avanzar la ventana de transmisión. Si tenemos ocupado el primario transmitiendo un tiempo igual a T_{ciclo}, obtenemos la máxima eficiencia de la transmisión. Es decir, si $T_{ciclo} < T_w$, el primario no se pararía nunca y en ausencia de errores se obtendría el 100% de eficiencia de transmisión.

Definimos la ventana óptima como:

$$E = \frac{T_w}{T_{Ciclo}} = 1 \text{ donde } T_w = W_{Trama}$$

$$T_w = T_{Ciclo} = W_{Trama} \rightarrow W = \frac{T_{Trama}}{T_{Ciclo}}$$

- Si $W < W_{óptima}$, la velocidad efectiva será inferior a la que podríamos conseguir con una ventana más grande.

- Si $W > W_{óptima}$, no aumentaremos la velocidad efectiva más allá de la conseguida con la ventana óptima.

6. Importancia del nivel de enlace según el contexto

Hasta este momento, hemos explicado las funciones que podemos encontrar en el nivel de enlace. No obstante, es importante entender que el nivel de enlace no siempre efectúa todas las funciones explicadas. Dependiendo del contexto en el que trabaje un protocolo de nivel de enlace, éste puede realizar unas funciones y no implementar otras.

Por ejemplo, en Internet (torre TCP/IP), el nivel de enlace no lleva a cabo ninguna función de recuperación de errores, simplemente descarta las tramas erróneas. La recuperación de errores la realizan los niveles superiores, normalmente el nivel de transporte.

Podemos encontrar el nivel de enlace en diferentes situaciones:

a) Comunicación punto a punto entre dos computadores locales; por ejemplo, la comunicación por el puerto serie entre dos PC para poder hacer una transferencia de ficheros. A causa del reducido número de elementos que intervienen en este caso, toda la arquitectura de comunicaciones estará normalmente integrada en el mismo programa. Por supuesto, aquí no habrá nivel de red y el nivel de enlace será responsable de la recuperación de errores.

b) Entorno de acceso a WAN[39] (Internet). Actualmente, es uno de los más habituales. Millones de usuarios lo utilizan para acceder a Internet. En este entorno el protocolo de nivel de enlace se establece entre el computador del usuario y el computador del proveedor de Internet. Aquí, el usuario se conecta al proveedor mediante un módem. El proveedor dispone de una batería de módems para que múltiples usuarios puedan conectarse simultáneamente.

[39]WAN es la sigla de *wide area networks*.

c) Red de área local (LAN[40]). Una característica de estos tipos de redes es que está formada por una comunidad de computadores que comparten un único medio de transmisión. Son redes multipunto o de difusión. En las redes de área local podemos distinguir dos modos de organizar la comunicación entre los ordenadores sin interferir entre sí:

[40]LAN es la sigla de *local area networks*.

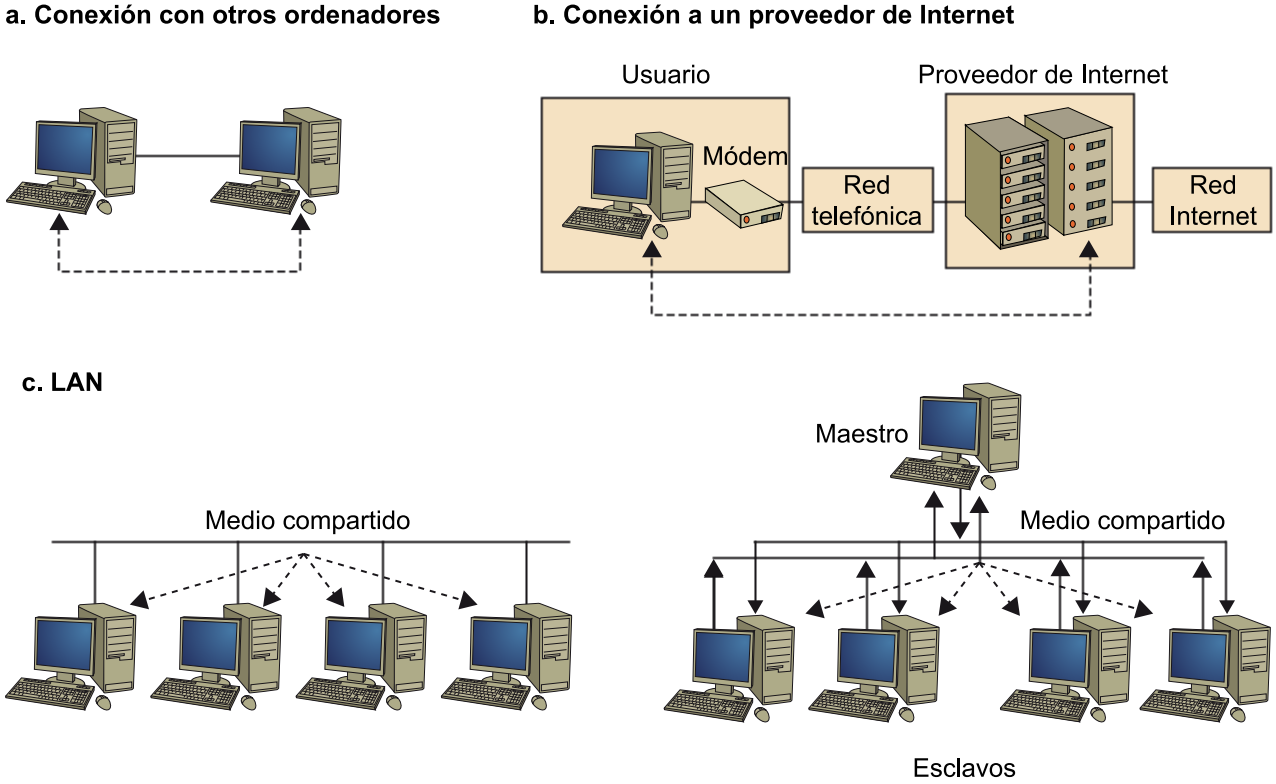
- Un computador **maestro** se encarga de arbitrar todas las comunicaciones. Las comunicaciones se producen entre el maestro y otro de los ordenadores (los **esclavos**), o viceversa. En este entorno podemos interpretar que hay un enlace punto a punto entre el maestro y cada uno de los esclavos, y que el maestro selecciona alternativamente uno de los enlaces posibles, según algún algoritmo de arbitraje.

- No existe un árbitro que seleccione uno de los enlaces posibles. El algoritmo de acceso está distribuido entre los ordenadores que acceden al medio.

d) Redes troncales de área extensa (WAN[41]). Se corresponden con el nivel de enlace existente en las redes troncales de los proveedores de acceso a Internet o empresas de telecomunicaciones. Estas tecnologías sirven para comunicar ordenadores separados por distancias muy grandes.

[41]WAN es la sigla de *wide area networks*.

Figura 31. Ejemplos de contextos en los que se puede encontrar el nivel de enlace



En esta figura, las líneas discontinuas conectan los dispositivos entre los que se establecen el protocolo de nivel de enlace.

La clasificación de los protocolos de nivel de enlace sobre el contexto de utilización guiará la estructuración de los siguientes puntos del módulo de enlace:

- El nivel de enlace entre dos computadores locales. Estudiaremos protocolos utilizados en la comunicación entre ordenadores locales o un ordenador y un dispositivo de entrada/salida mediante el puerto serie o el puerto paralelo: RS232 o BSC.

- El nivel de enlace en las redes de área local (LAN). Estudiaremos el problema de las redes de difusión que se han utilizado históricamente en las redes de área local. Centraremos el estudio en las tecnologías más utilizadas en las redes de área local, tanto en medios cableados (Ethernet) como en medios inalámbricos (WIFI 802.11). Aquí se establecerá la clasificación de las tecnologías inalámbricas según su extensión y se incluirá el estudio de Wimax (802.16).

- El nivel de enlace en las redes de acceso a WAN. Estudiaremos los protocolos básicos de nivel de enlace, HDLC y PPP, sobre los que se fundamenta buena parte de las tecnologías de acceso a las redes de área extendida mediante un operador de telecomunicaciones. Para complementar este apartado, veremos las principales tecnologías o sistemas de acceso a WAN que se han utilizado en los últimos años hasta hoy: RTC/RTB, RDSI (T1/E1), ADSL y HFC.

- El nivel de enlace en las redes de transporte WAN. Estudiaremos las tecnologías principales utilizadas por las operadoras de telecomunicaciones en el nivel de enlace en las redes de transporte WAN: X.25, Frame relay, ATM y MPLS.

7. El nivel de enlace en las redes de área local

La capa del nivel de enlace[42] puede gestionar dos tipos de enlaces: los enlaces punto a punto[43] y los enlaces de difusión[44].

(42)En inglés, *link layer*.

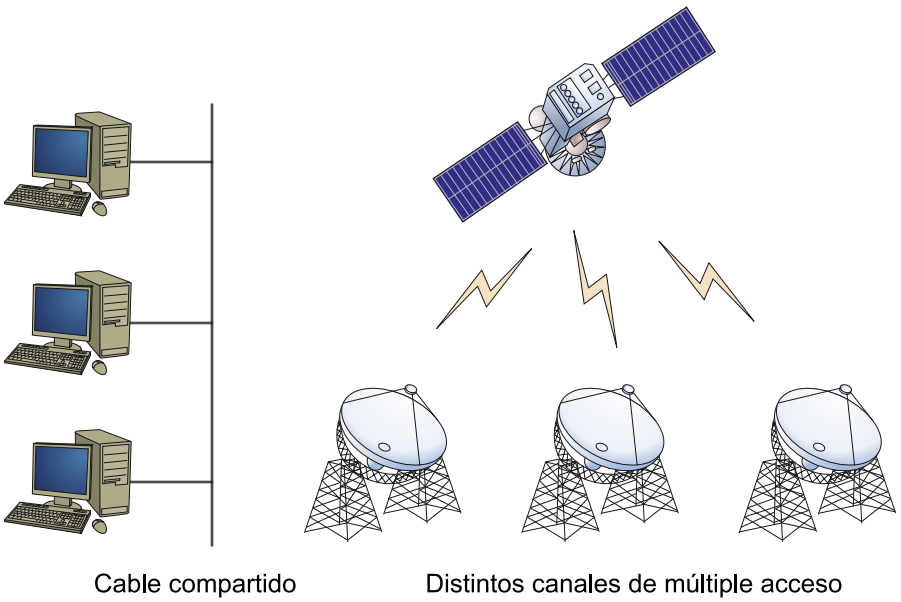
(43)En inglés, *point to point links*.

(44)En inglés, *broadcast links*.

Un enlace punto a punto consiste en un solo emisor y un solo receptor conectados por un solo cable. Existen determinados protocolos de comunicaciones que funcionan sobre enlaces punto a punto, como el PPP o el HDLC. Coordinar el acceso en estos tipos de enlaces resulta trivial.

En los enlaces por difusión existen múltiples nodos (o estaciones) emisores y receptores conectados al mismo cable (o canal); comparten el canal para enviar y recibir información. Se habla de concepto de difusión porque cuando un nodo transmite una trama de información el canal difunde una copia de la trama en cada estación conectada al cable. La noción de difusión nos puede resultar familiar en el caso de transmisión de señales de televisión. La televisión consiste en un nodo fijo (antena o repetidor) que transmite a todos los nodos (aparatos o receptores de televisión). Las redes de área local Ethernet o las redes WiFi son otros ejemplos de redes locales en las que se aplican estos conceptos.

Figura 32



7.1. MAC

Los protocolos de acceso múltiple (MAC[45]) son mucho más beneficiosos cuando las comunicaciones punto a punto se vuelven ineficientes: cuando no es posible tener una línea punto a punto entre cada par de estaciones de la red, cuando la utilización de las líneas punto a punto son muy bajas, por razones económicas, etc. También son muy útiles en el caso de que exista un número elevado de nodos transmitiendo información de una manera descoordinada. Por ejemplo, cuando cada estación decide independientemente de las otras lo que quiere transmitir, a quién desea enviarlo y en qué momento lo quiere hacer. En resumen, la necesidad de un protocolo de acceso múltiple aparece cuando existe la necesidad de comunicaciones entre nodos independientes en una red interconectada.

(45)MAC es la sigla de *media access control*.

Las dos características importantes de estos tipos de redes son:

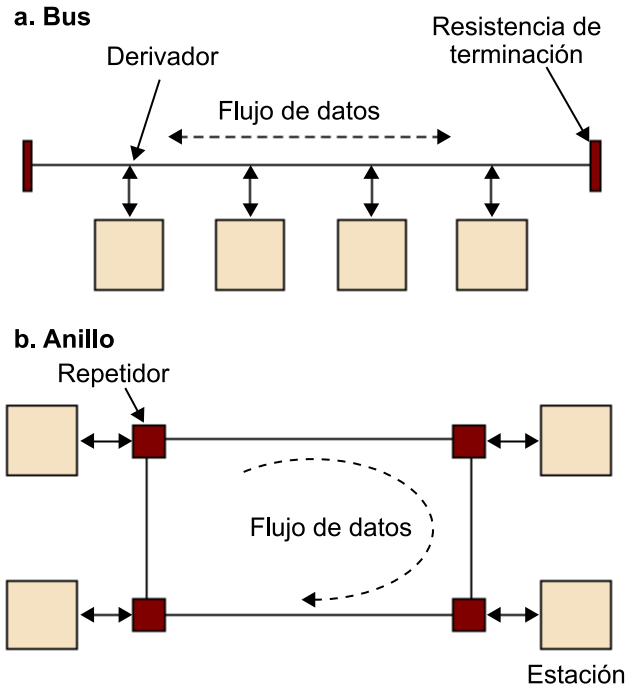
1) La red contiene nodos independientes intentando comunicarse a través de un único canal de comunicación compartido. Un nodo que desea transmitir información necesita revisar el estado del canal por si está libre o no para ello.

2) En un determinado instante, el número de estaciones de la red que desean transmitir información es desconocido y cambia dinámicamente con el tiempo.

Definir los protocolos de acceso al medio consiste en una serie de reglas que cada nodo o estación de la red debe seguir para compartir un recurso, en nuestro caso, un canal compartido (cable, el aire, etc.). La elección de los protocolos de acceso depende mucho de la naturaleza del tipo de tráfico y del rendimiento que demandarán las estaciones de la red.

Supongamos que tenemos un conjunto de estaciones o nodos que están interconectadas de alguna manera entre ellas y que comparten el medio o canal de comunicación para enviar y recibir información entre ellas. Una red funciona a modo de difusión cuando la información transmitida desde una estación origen hacia una estación destino puede ser escuchada por el resto de las estaciones de la red, a pesar de que la información no vaya destinada explícitamente a ellas.

Figura 33. Ejemplo de red de difusión



Las estaciones están interconectadas con una topología a modo de bus y de anillo.

Una estación estará en estado activo si tiene información para transmitir. Pueden existir estaciones conectadas a una red que no se hallen en estado activo.

Durante los últimos años se han desarrollado muchos protocolos de acceso múltiple al medio; los podemos clasificar de la siguiente manera:

a) MAC estáticos (TDMA, FDMA, CDMA)

b) MAC dinámicos:

- Acceso dinámico por control centralizado.
- Acceso dinámico por control distribuido (paso de testigo)

c) MAC aleatorios (Aloha, Aloha segmentado, CSMA[46], CSMA/CD[47]).

[46]CSMA es la sigla de *carrier sense multiple access*.

[47]CSMA/CD es la sigla de *CSMA with collision detection*.

Las principales ventajas de los protocolos estáticos son que cada nodo tiene garantizado un ancho de banda determinado y que cada transmisión no suele interferir en la de otro. Las desventajas principales son que el ancho de banda del canal es asignado tanto a las estaciones o nodos que quieren transmitir como a los que no quieren transmitir y que el ancho de banda desperdiciado (inutilizado) no se puede traspasar de un nodo a otro. En general, los estáticos tienen un rendimiento bastante aceptable ante altas cargas de tráfico y su tiempo de respuesta para iniciar la transmisión suele ser bajo. Los protocolos de acceso dinámicos son muy atractivos, ya que ofrecen retrasos de respuesta bajos ante un tráfico bajo.

7.1.1. TDM

La técnica de TDM[48] sirve para repartir el ancho de banda del canal entre todos los nodos que comparten el canal de comunicaciones en el dominio temporal.

(48)TDM es la sigla de *time division multiplexing*.

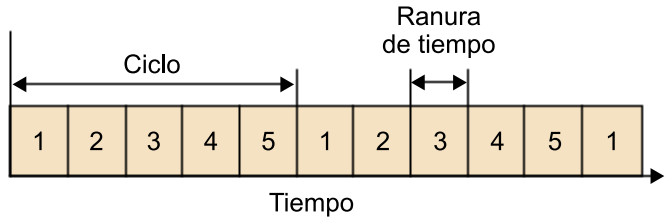
Si tenemos un canal que quiere soportar la transmisión de *N* nodos, TDM divide la línea temporal en *N* particiones temporales[49]. Cada partición temporal es asignada a un único de los *N* nodos para que durante la partición este nodo pueda transmitir información. En general, la duración de la partición temporal se elige de manera que una trama entera se pueda transmitir durante la duración de la partición. Por lo tanto, cada estación o nodo tiene derecho a transmitir durante un período fijo de tiempo. Este derecho pasa de nodo a nodo correlativamente, hasta que todos los nodos han tenido el derecho a transmitir. Una vez que se llega al último nodo, este ciclo empieza de nuevo, lo que le vuelve a dar derecho de transmisión al primer nodo.

(49)En inglés, *time slots*.

Si tenemos un canal compartido con un ancho de banda de *R* bps, con *N* nodos, un nodo dispone de media de un ancho de banda dedicado sólo para él de *R*/*N* bps.

Este sistema tiene dos inconvenientes: el primero es que, si de los *N* nodos sólo hay $M < N$ que desean transmitir información, el ancho de banda total del canal R no se aprovecha totalmente, ya que cada nodo sólo utiliza un ancho de banda de *R/N* y, en conjunto, sólo se aprovecha un ancho de banda de $R \cdot M/N < R$. El segundo inconveniente es que cuando un nodo quiere transmitir dos tramas seguidas, después de haber transmitido el primero, debe esperar un turno completo para que pueda volver a transmitir el segundo paquete.

Figura 34. Ejemplo de un acceso al canal en TDM



7.1.2. FDM

La técnica FDM[50] divide el ancho de banda del canal entre diferentes frecuencias y asigna un ancho de banda (en frecuencia) a cada uno de los nodos.

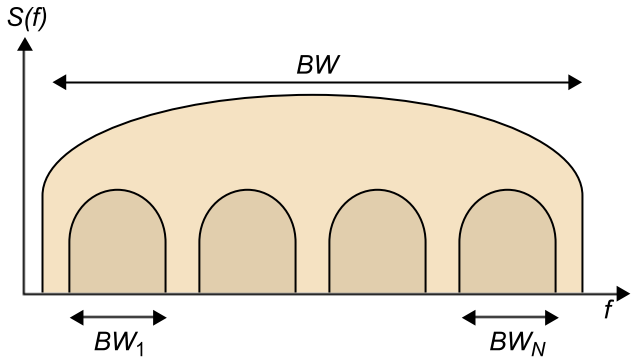
(50)FDM es la sigla de *frequency division multiplexing*.

FDM crea *N* pequeños subcanales, cada uno a una frecuencia diferente de los otros, con la particularidad de que todos los nodos pueden transmitir a la vez, pero a diferente frecuencia. Con este sistema, el ancho de banda del canal R se divide entre cada uno de los *N* nodos, con un ancho de banda de *R*/*N* por nodo.

Este sistema, por ejemplo, es utilizado en la retransmisión de la televisión convencional. Diferentes cadenas de televisión transmiten sus programas mediante ondas de radio y de televisión por el aire, todas al mismo tiempo, pero cada cadena de televisión en un canal diferente (a una frecuencia diferente), de manera que las emisiones no se superponen en el espacio frecuencial. Por medio de un filtro (selector del canal), el receptor (en nuestro caso, el aparato de televisión) decide qué frecuencias quiere aceptar (y, por lo tanto, rechazar las ondas de otros canales) para ver un determinado canal de televisión.

Como principal inconveniente, este sistema reduce el ancho de banda de un nodo a *R/N* bps, a pesar de que sea el único nodo de la red que desee transmitir información.

Figura 35. Ejemplo de división de la frecuencia del canal *BW* en subcanales BW_i



7.1.3. CDMA

CDMA[51] se ha utilizado en aplicaciones militares y actualmente en canales de acceso múltiple inalámbricos.

(51)CDMA es la sigla de *code division multiple access*.

CDMA permite que varios nodos transmitan simultáneamente. Este sistema asigna un código de unas características especiales a cada nodo. Cada nodo utiliza su único código para codificar los bits que transmite, y los respectivos nodos receptores de la información conocen el código del emisor.

Si no se producen interferencias por varias transmisiones simultáneas de diferentes emisores, el receptor, a partir de unas operaciones matemáticas y del código del emisor, es capaz de recuperar el mensaje original transmitido. En cambio, cuando varios emisores transmiten a la vez, el receptor recibe una se-

ñal formada por la suma de las diferentes emisiones y, mediante un proceso de codificación/descodificación matemática con el código de un emisor, detecta que la información es incorrecta.

Por analogía real, CDMA sería como si tuviéramos un grupo de personas hablando en varias lenguas diferentes: los humanos somos capaces de captar muy bien los mensajes en la lengua que entendemos y no tenemos en cuenta los mensajes en las lenguas que desconocemos.

7.1.4. Protocolos de acceso dinámicos

Las dos características ideales que debe tener un protocolo de acceso múltiple son las siguientes:

1) Cuando existe un único nodo activo, aquel que quiere transmitir información, el ancho de banda del canal *R* debe estar disponible para ese nodo.

2) Cuando hay *N* nodos que utilizan el canal, el ancho de banda disponible (o rendimiento[52]) para cada nodo ha de estar lo más aproximado posible a *R/N*. Esta condición no la cumplen los protocolos aleatorios o de contención que se explican en otro apartado.

[52]En inglés, *throughput*.

Ved también

Podéis ver esta condición en el apartado 8 de este módulo didáctico.

Por este motivo, los investigadores desarrollaron una nueva clase de protocolos denominados protocolos de acceso por rotación circular[53]. Lo más importante de esta clase de protocolos es el control centralizado[54]: requiere que uno de los nodos sea el denominado máster (o central) y controle el canal. Este nodo autoriza[55] al primero a transmitir varias tramas mediante el envío de un mensaje. Una vez que este nodo ha acabado de transmitir, el nodo máster comunica al segundo nodo la autorización para transmitir. Y el proceso se va repitiendo de manera cíclica[56]. Además, el nodo máster controla cuándo un nodo ha acabado de transmitir sus tramas observando el estado de la señal del canal. Este sistema elimina las particiones vacías y ello aumenta la eficiencia global del sistema. El principal inconveniente es que introduce el tiempo que se tarda en comunicar a un nodo la indicación que puede empezar a transmitir la información (*delay polling*). El segundo inconveniente es que si el nodo máster falla, la red se vuelve inoperativa.

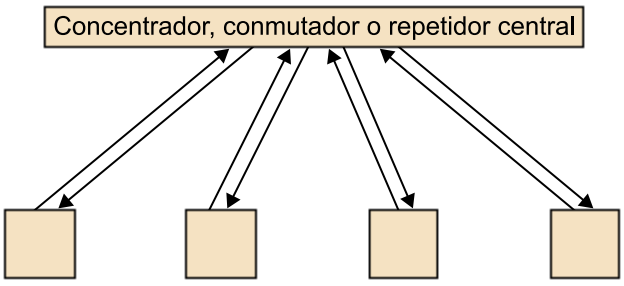
[53]En inglés, *taking turns protocols*.

[54]En inglés, *polling control*.

[55]En inglés, *poll*.

[56]En inglés, *round-robin*.

Figura 36. *Polling* con topología en estrella

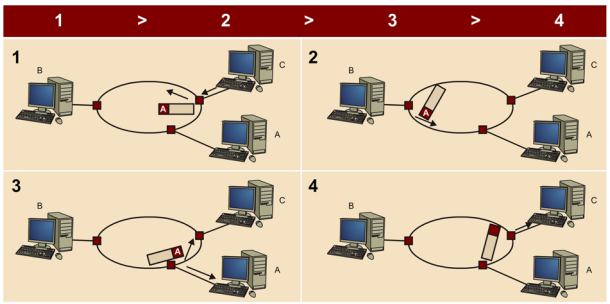


Otros protocolos de estos tipos son los protocolos basados en el control distribuido[57]. En este caso, no existe ningún nodo máster que regule el acceso al canal. El acceso se basa en una pequeña trama, denominada testigo[58], cuyo propósito especial circula por los nodos dispuestos en una topología en anillo. El nodo 1 transmite el testigo al nodo 2, éste al 3 y, finalmente, el nodo *N* al nodo 1, para volver a empezar otro ciclo rotatorio. Cuando una estación ha recibido el testigo, tiene el derecho a transmitir una trama de información, que también circula por el anillo (de estación en estación), hasta que la estación destino la captura. Si una estación recibe el testigo y no quiere transmitir ninguna trama de información, envía el testigo a la siguiente estación. El paso del testigo se realiza de una manera descentralizada y aumenta mucho el rendimiento de la red. El inconveniente principal es que si un nodo falla, puede fallar todo el anillo que forman los nodos.

(57)En inglés, *token passing protocol.*

(58)En inglés, *token.*

Figura 37. Control distribuido por el paso del testigo



1. C transmite una trama a A.
2. La trama no va dirigida a B. La estación B la ignora.
3. La estación A recopila la trama y la retransmite a C.
4. La estación C absorbe la trama.

7.1.5. Protocolos de acceso aleatorio o de contención

Las técnicas de contención (o aleatorias) ofrecen un acceso fácil al canal de la red cuando la carga global sobre la red es baja. En general, se utilizan en las redes de difusión que comparten un mismo canal con un gran número y variable de nodos con tráfico a ráfagas[59].

(59)En inglés, *bursty.*

Muchas estaciones no tienen información que transmitir durante la gran parte del tiempo y en un instante sólo una pequeña parte de estas estaciones desea enviar información y lo hacen a ráfagas.

En los protocolos de contención es posible tener varias transmisiones superpuestas en el tiempo para varias estaciones. Una superposición en la transmisión de una trama produce una colisión, lo que provoca la destrucción de todas las tramas involucradas en aquel momento. Si transmite un único nodo, la trama es recibida por el destinatario sin ningún problema. Es importante distinguir la diferencia entre un error de transmisión (errores provocados por el bullicio del canal) y una colisión de tramas (provocada por la superposición de dos tramas sobre el mismo canal de comunicación).

En los protocolos de acceso aleatorio, cuando un nodo transmite lo hace a la velocidad que le permite el ancho de banda del canal *R*. Cuando se produce una colisión, los nodos involucrados en la colisión retransmiten las tramas hasta conseguir transmitir la trama sin colisión. Después de detectar una colisión, el nodo espera un tiempo aleatorio para volver a intentar la retransmisión de la trama y cada nodo involucrado en esta colisión elige un tiempo aleatorio diferente e independiente del otro nodo, lo que disminuye la probabilidad de que se vuelva a producir una colisión.

Las colisiones y las consecuentes retransmisiones son el precio que se ha de pagar por la descoordinación entre estaciones y por el acceso aleatorio sobre el mismo canal. Las colisiones limitan la cantidad de información que se puede transmitir sobre el canal, proporcionan un orden aleatorio para iniciar la transmisión e introducen un retraso variable[60] entre los paquetes. La gran desventaja que tienen es que grandes fluctuaciones estadísticas de las características del tráfico pueden provocar que el canal tenga un rendimiento prácticamente cero, ya que el canal se inunda de colisiones continuamente.

[60]En inglés, *delay jitter*.

Se define el tiempo de propagación de una señal (t_p) como el máximo tiempo que la señal se propaga entre cualquier par de transmisores y receptores de la red. En general, el período de detección de la colisión se calcula aproximadamente como el tiempo de propagación de la señal. Este valor afecta al rendimiento de los protocolos.

Se define el tiempo de vulnerabilidad (T_v) o ventana de colisiones como el tiempo durante el que una trama es susceptible de experimentar colisiones.

Aloha puro

El primer protocolo de la familia Aloha, del año 1970, es un protocolo descentralizado denominado Aloha puro: cuando llega una trama del nivel de enlace, el nodo la transmite inmediatamente sobre el canal. Si la trama transmitida experimenta una colisión con una o más tramas transmitidas por otros nodos, el nodo, después de acabar de transmitir su trama colisionada, retransmitirá enseguida la trama con probabilidad *p*. En caso contrario, el nodo esperará un tiempo, que es la duración de la transmisión de una trama completa. Después de esta espera, transmitirá la trama con probabilidad *p* o esperará otro período de transmisión de trama completa con probabilidad 1 – *p*.

Ejemplo de Ahola puro

Supongamos que la estación *A* transmite una trama en el instante t_0. La transmisión de una trama tiene una duración t_{Trama}. La estación *A* no sabe si alguna estación ha transmitido una trama antes de t_0 o si la transmitirá después de t_0. Esto significa que, si en el intervalo $[t_0 - t_{Trama}, t_0]$ hay alguna estación que ha empezado a transmitir una trama, o si alguna estación empieza a transmitir una trama en el intervalo $[t_0, t_0 + t_{Trama}]$, se producirá una colisión. Por lo tanto, el tiempo de vulnerabilidad valdrá:

$$T_V = (t_0 + t_{Trama}) - (t_0 - t_{Trama}) = 2t_{Trama})$$

Para calcular la duración de una colisión, si una estación *A* transmite una trama en el instante t_0:

- Si una estación *B* transmite la trama en el instante t_0, la duración de la colisión será $t_{col} = t_{Trama}$.
- Si la estación B transmite en el intervalo $[t_0 - t_{Trama}, t_0]$ o $[t_0, t_0 + t_{Trama}]$, la duración de la colisión será:

$$t_{Trama} \leq t_{col}\ 2t_{Trama}$$

Para analizar el rendimiento de un protocolo Aloha puro, definimos las siguientes variables:

- *S*: rendimiento del canal. El número medio de transmisiones con éxito por tiempo de transmisión de trama t_{Trama}.
- *G*: carga ofrecida. El número medio de intentos de transmisiones por tiempo de transmisión de trama t_{Trama}.
- *E*: número medio de retransmisiones.
- P_0: probabilidad de que durante el tiempo de vulnerabilidad $T_v = 2 \cdot t_{Trama}$ ninguna estación genere trama alguna que transmitir y, por lo tanto, no se generen colisiones.

Para modelar matemáticamente el comportamiento, consideramos un conjunto de infinitos nodos, en el que cada uno genera tramas de longitud fija, según un proceso de Poisson. Supongamos que el proceso de llegadas de nuevas tramas y de tramas retransmitidas sigue un proceso de Poisson. Esto nos da que:

$$P_0 = e - \frac{G \cdot 2 \cdot T_{Trama}}{T_{Trama}} = e^{-2G}$$
$$S = G \cdot P_0 = G \cdot e^{-2G}$$
$$\frac{dS}{dG} = e^{-2G} - 2 \cdot G \cdot e^{-2G}$$
$$\frac{dS}{dG} = 0 \Rightarrow G = 0{,}5$$
$$S_{\max} = \frac{1}{2e} = 0{,}184$$

Aloha segmentado

Cada trama tiene una longitud fija de L bits. El tiempo se divide en segmentos o *slots* de L/R segundos (una partición tiene una duración igual al tiempo para transmitir una trama), donde R es la velocidad de transmisión. En este caso, el sistema pasa de ser continuo (Aloha) a discreto.

Los nodos sólo pueden transmitir las tramas al inicio de cada partición. Los nodos están sincronizados de tal manera que saben en qué instante empieza una partición. Cuando dos o más tramas colisionan en una partición, todos los nodos detectan la colisión antes que la partición temporal acabe.

Una partición temporal donde hay un único nodo que transmite se considera una partición satisfactoria: no se produce colisión y la transmisión es satisfactoria.

Para calcular el tiempo de colisión en este caso, sólo dos estaciones pueden transmitir al principio de los segmentos, de manera que:

$$t_{col} = t_{Trama}$$

Para calcular el tiempo de vulnerabilidad, sólo se pueden producir colisiones en el tiempo del segmento, de manera que la vulnerabilidad valdrá:

$$t_v = t_{Trama}$$

Y dado que $t_v = t_{Trama}$, las ecuaciones del rendimiento de Aloha segmentado son:

$$P_0 = e^{-\frac{G \cdot T_{Trama}}{T_{Trama}}} = e^{-G}$$
$$S = G \cdot P_0 = G \cdot e^{-G}$$
$$\frac{dS}{dG} = e^{-G} - \cdot G \cdot e^{-G}$$
$$\frac{dS}{dG} = 0 \Rightarrow G = 1$$
$$S_{\max} = \frac{1}{e} = 0{,}368$$

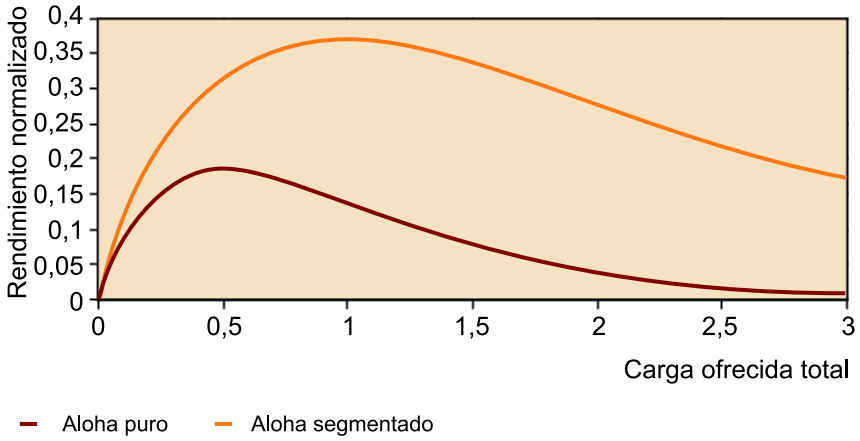
Comparación del rendimiento entre Aloha puro y Aloha segmentado

La figura 38 nos muestra el rendimiento de ambos protocolos. El rendimiento de Aloha segmentado es máximo cuando la carga ofrecida (las tramas nuevas más las tramas retransmitidas) es 1 (por ejemplo, una trama por tiempo de transmisión de una trama). Como el rendimiento máximo de Aloha segmentado es $1/e$, esto significa que, de media, cada trama se debe transmitir e ve-

ces (2,718) o, aproximadamente, tres veces. Menos del 40% de las tramas por tiempo de transmisión de una trama se pueden transmitir correctamente en canales Aloha.

Se ha comprobado que el rendimiento máximo al que se puede llegar en un canal Aloha puro debe ser menor que $1/2e$, menor que en Aloha segmentado. Aloha segmentado tiene la ventaja de una mayor eficiencia de rendimiento y los inconvenientes de una sincronización necesaria (para conocer el inicio de cada *slot*) y del incremento del sobrecoste (relación entre el número de bits de información de una trama y el número de bits total de una trama) de las cabeceras, cuando las tramas de mayor tamaño están segmentadas en tramas más cortas para hacerlas caber en la duración de las particiones.

Figura 38. Rendimiento por Aloha puro y segmentado



Con respecto al número medio de retransmisiones, la ratio G/S mide el retraso medio que transcurre, ya que representa el número de transmisiones antes de que una trama sea transmitida con éxito. Por Aloha segmentado, la probabilidad de tener $K-1$ intentos, seguido de un intento con éxito de transmisión de una trama, es:

$$P_K = e^{-G}(1-e^{-G})^{K-1}$$

Y el número medio de transmisiones es:

$$Q = \sum_{K=1}^{\infty} K \cdot P_K = \sum_{K=1}^{\infty} K \cdot e^{-G}(1-e^{-G})^{K-1} = e^{G}$$

$$S = \frac{G}{e^{G}} = \frac{\ln Q}{Q}$$

Por último, el número medio de retransmisiones es:

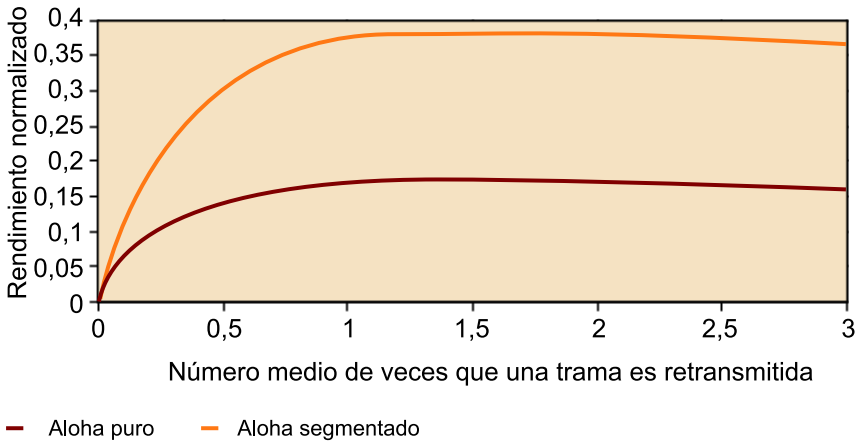
$$E = Q - 1 = e^{G-1}$$

Por Aloha puro, tenemos que:

$$E = e^{2G-1}$$

A partir de estas ecuaciones se puede deducir la figura 39, que nos muestra el número de retransmisiones en función del rendimiento del canal (*S*):

Figura 39. Número de retransmisiones por Aloha puro y segmentado



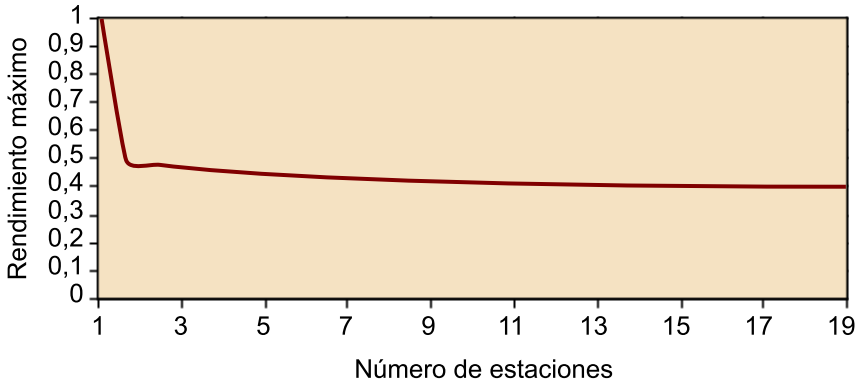
Rendimiento de Aloha segmentado en función del número de estaciones

Ahora, supongamos que tenemos *N* nodos en una red Aloha segmentado. Cada nodo transmite una trama con una probabilidad *p* y decide no transmitir con probabilidad 1 – *p*:

$$S = \binom{N}{1} \cdot p \cdot (1-p)^{N-1} = N \cdot p \cdot (1-p)^{N-1}$$
$$\frac{dS}{dp} = N \cdot (1-p)^{N-1} - N \cdot (N-1) \Rightarrow p \cdot (1-p)^{N-2}$$
$$\frac{dS}{dp} = 0 \Rightarrow p = \frac{1}{N}$$
$$S_{max} = (1-\frac{1}{N})^{N-1} \rightarrow \frac{1}{e} \text{ cuando } N \rightarrow \infty$$

La figura 40 nos muestra el rendimiento máximo en función del número de usuarios o nodos reales. Para un número de nodos bajo, la probabilidad del éxito es alta. Cuando el número de nodos se incrementa, el rendimiento degenera asintóticamente a 1/*e*, tal como habíamos calculado en el anterior modelo.

Figura 40. Transmisión simétrica en Aloha puro

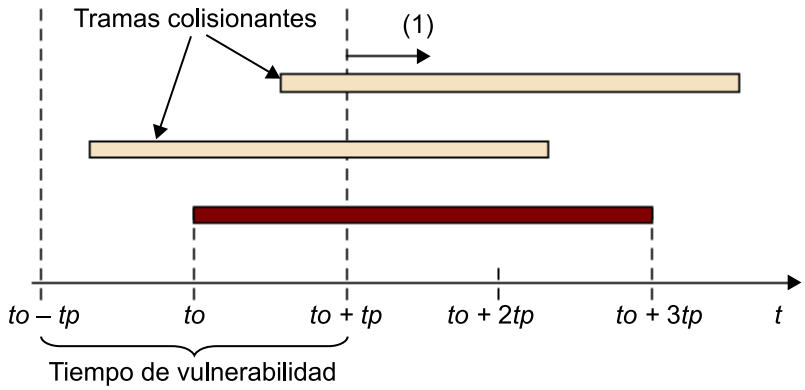


CSMA

Para canales con un tiempo de propagación bajo, en comparación con el tiempo de transmisión de una trama, las colisiones se pueden reducir significativamente exigiendo que cada nodo activo escuche el canal por si alguna trama se está transmitiendo por el canal antes de iniciar su propia transmisión (escuchar antes de hablar). En este caso, cuando una estación o nodo está transmitiendo un paquete, todas las otras estaciones de la red detienen su transmisión durante el tiempo en el que se transmite la trama. En suma, durante la transmisión de una trama las otras estaciones permanecen en silencio.

En CSMA el tiempo de vulnerabilidad se elige como el tiempo máximo de propagación $T_v = 2 \cdot t_p$, siendo t_p = Distancia del bus/V_p. El tiempo se divide en *slots* de tiempos muy pequeños (de duración t_p). Es como si utilizáramos Aloha segmentado con particiones muy cortas, lo que provoca la reducción del número de colisiones.

Figura 41



En CSMA tenemos varias estrategias de funcionamiento:

- No persistente. La estación activa escucha el canal y opera de la siguiente manera: 1) Si la estación detecta que el canal está libre, la trama se transmite inmediatamente. 2) Si la estación detecta que el canal está ocupado,

la estación espera un tiempo aleatorio antes de volver a intentar testear el estado del canal.

- P-persistente. Sólo es utilizado en los canales segmentados. En este caso, la estación opera de la siguiente manera: 1) Si la estación detecta que el canal está libre, la estación transmite con probabilidad p (o difiere en la siguiente partición de tiempo con probabilidad $1 – p$). 2) Si la estación detecta que el canal está ocupado, la estación continúa escuchando hasta que el canal se detecta libre y, entonces, transmite la trama con probabilidad p (o difiere en la siguiente partición de tiempo con probabilidad $1 – p$).

Un caso especial del caso p-persistente es el caso 1-persistente, que permite transmitir una trama inmediatamente cuando se detecta que el canal está libre. El caso p-persistente está pensado para ser utilizado cuando todas las estaciones de un canal tienen tramas que transmitir (y, por lo tanto, difícilmente se halla en un estado libre), con el fin de obtener un rendimiento elevado.

Es posible que una estación detecte que el canal está libre cuando, precisamente, otra estación haya iniciado la transmisión de su trama. Esto provocaría una colisión.

Para calcular el tiempo de colisión se asume que la estación A transmite una trama en el instante t_0.

- Si la estación B transmite su trama en el instante t_0, la duración de la colisión: $t_{col} = t_{Trama}$

- Si la estación B transmite en el instante $[t_0 – t_p, t_0]$ o $[t_0, t_0 + t_p]$:

$$t_{Trama} \leq t_{col} \leq + t_{Trama} \approx 2t_p$$

Para calcular el rendimiento del CSMA, supongamos que t_p es el tiempo de propagación máximo del canal y T_{Trama} el de transmisión del paquete, y definimos $a = t_p/T_{Trama}$. El rendimiento máximo del canal CSMA se puede obtener en función del ancho de banda de Aloha segmentado S_{SA}:

$$S_{CSMA} = \frac{S_{SA}}{2a + S_{SA}(1+a)}$$

Para entornos abiertos (por ejemplo, redes inalámbricas, típicamente $a = 0{,}001$ hasta $0{,}1$), cuando el canal está ligeramente cargado, el retraso para que una estación pueda acceder y transmitir al canal es relativamente corto. Este retraso es independiente del número de estaciones. En particular, cuando sólo hay

una sola estación, este retraso es cero. El rendimiento del canal para CSMA decae cuando la carga introducida en la red crece como todos los algoritmos de contención.

A modo de resumen, la siguiente tabla nos muestra el tiempo de vulnerabilidad y el tiempo de duración de una colisión con los tres algoritmos mencionados:

	Tiempo de vulnerabilidad	**Duración de una colisión**
Aloha puro	$2 \cdot t_{Trama}$	$t_{Trama} \leq t_{col} \leq 2 \cdot t_{Trama}$
Aloha segmentado	t_{Trama}	t_{Trama}
CSMA	$2 \cdot t_{prop}$	$t_{Trama} + t_{prop}$

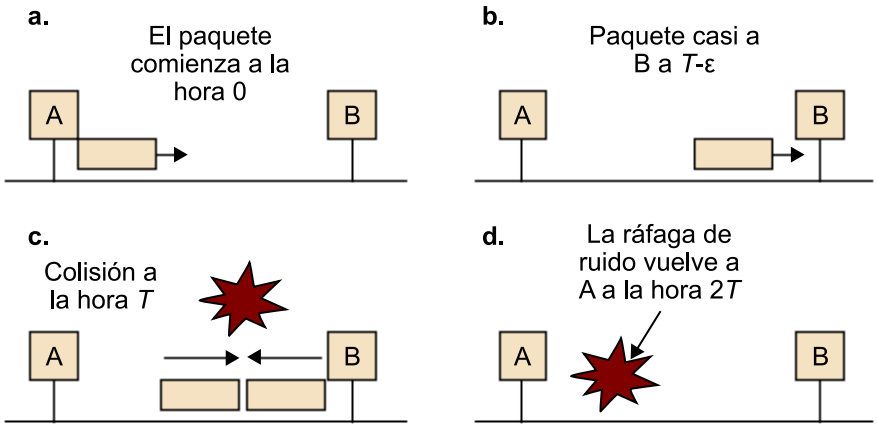
CSMA/CD

El rendimiento de la escucha de la portadora (escucha del estado del canal) se puede mejorar si se permite que las estaciones involucradas en una colisión puedan abortar su transmisión, una vez hayan detectado la colisión, sin que sea necesario finalizar toda la transmisión de la trama. Las tramas que han sufrido una colisión son retransmitidas después de un retraso aleatorio y en cada colisión que afecte al mismo paquete se va duplicando el retraso de la retransmisión. La razón de esto es que los períodos de colisión se hacen más cortos y las colisiones no continuarán durante toda la transmisión de la trama. Los mecanismos típicos para detectar la colisión consisten en comparar las señales emitida y recibida por el canal.

Este mecanismo de funcionamiento "Si hay alguien más que habla al mismo tiempo, deja de hablar" se denomina detección de la colisión[61] (CD).

[61] En inglés, *colision detection*.

Figura 42. Tiempo máximo de detección de una colisión (*t* = Tiempo de propagación)



7.1.6. Direccionamiento en el nivel MAC

Tal como se ha explicado, las estaciones en una red de área local envían tramas a otras estaciones sobre un canal compartido. Esto significa que cuando un nodo envía una trama, el resto de los nodos de la red recibe esta trama. En el caso de que un nodo de LAN no quiera transmitir la trama a todas las otras estaciones, sino sólo a una estación concreta, cada nodo o estación de la red debe disponer de una dirección propia que le permita dirigir una trama a una estación concreta. Por ello, dentro de la trama que se transmite suele haber un campo que contiene la dirección a quien va destinada esta trama. De este modo, cuando un nodo o estación recibe una trama puede determinar si ésta va dirigida a ella o no.

Si la dirección destino de la trama coincide con la dirección propia de la estación, la estación extrae el datagrama de nivel de red de la trama del nivel de enlace y pasa el datagrama al nivel superior de la pila de protocolos. Si la dirección destino no coincide con la dirección de la estación que la escucha, la estación descarta esta trama.

Direccionamiento en una LAN

Cada dirección de nivel LAN, también conocida como dirección física, dirige MAC o dirección Ethernet si emplea la tecnología Ethernet. En Ethernet (y otras tecnologías) las direcciones están constituidas por 6 bytes, que proporcionan 2^{48} posibles direcciones diferentes.

En general, las direcciones se expresan en formato hexadecimal, separadas por el símbolo '-', como 1A-23-F9-CD-06-9B. Los adaptadores de red LAN contienen una memoria ROM con su dirección de fábrica, que es permanente. Dos adaptadores de red nunca tienen la misma dirección LAN. El organismo IEEE gestiona el espacio de direcciones físicas de todo el mundo. IEEE fija o determina los primeros 24 bits de la dirección para cada fabricante y da permiso al fabricante de los adaptadores para crear una combinación única para los últimos 24 bits de la dirección.

Las direcciones LAN no tienen una estructura jerárquica, ya que ésta es fijada por el fabricante en el momento de su fabricación, al contrario que las direcciones IP en Internet. Por ejemplo, cuando un ordenador se desplaza o cambia de una red IP a otra, hay que cambiar la dirección IP (y también la máscara y la pasarela[62]), a pesar de que la dirección física sea la misma.

(62)En inglés, *gateway*.

Los adaptadores LAN interpretan una dirección especial denominada dirección de difusión[63]. Para Ethernet es FF-FF-FF-FF-FF-FF (48 bits en unos consecutivos). Sirve para enviar una trama de nivel LAN a todas las estaciones poniendo en el campo de dirección destino esta dirección *broadcast*. A diferencia

(63)En inglés, *broadcast address*.

de una dirección que no es *broadcast*, que varias estaciones pueden recibir pero no la procesan, si no va destinada a una estación en concreto, con la dirección *broadcast* enviamos la trama para que todas las estaciones que la reciban la procesen.

Para asociar las direcciones IP con las direcciones de nivel LAN, existe un protocolo denominado ARP[64] que mantiene dentro de una tabla ARP los siguientes datos: dirección IP, dirección física y TTL[65]. El campo TTL sirve para indicar si una entrada de la tabla se debe borrar o no, si ha expirado o no temporalmente su validez (cada 20 minutos, en general). Por ello, cuando una estación quiere asociar una dirección IP con una dirección física desconocida envía a una trama de nivel LAN de difusión (a todas las estaciones de la red local con dirección FF-FF-FF-FF-FF-FF) un paquete especial denominado paquete ARP, que se enmarca dentro de una trama MAC con la dirección IP por la que se pregunta. La estación que realmente tiene asignada la dirección IP solicitada responde al paquete ARP y lo envía a la estación que ha pedido su dirección MAC física con una trama estándar (no difusión). Finalmente, la última estación que lo ha pedido puede actualizar su tabla ARP.

(64)ARP es la sigla de *address resolution protocol*.

(65)TTL es la sigla de *time to live*.

7.2. Ethernet

En los años setenta, Bob Metcalfe diseñó un protocolo para conectar los ordenadores de la empresa Xerox. Este protocolo estaba basado en el protocolo Aloha y le puso el nombre de Ethernet.

Durante los años ochenta, un grupo formado por las empresas Digital, Intel y Xerox, conocido como DIX, fue el primero en implementar Ethernet DIX. Y se creó e implementó la primera especificación de LAN Ethernet. A mediados de los años ochenta, el instituto IEEE utilizó la base de Ethernet DIX para publicar la especificación 802.3 Ethernet.

Entre los años ochenta y noventa existían en el mercado comercial dos tipos de redes de área local: las basadas en el protocolo Ethernet (estandarizadas en IEEE 802.3) y las basadas en protocolos de acceso de turno rotatorio (*token ring* IEEE 802.5 y FDDI).

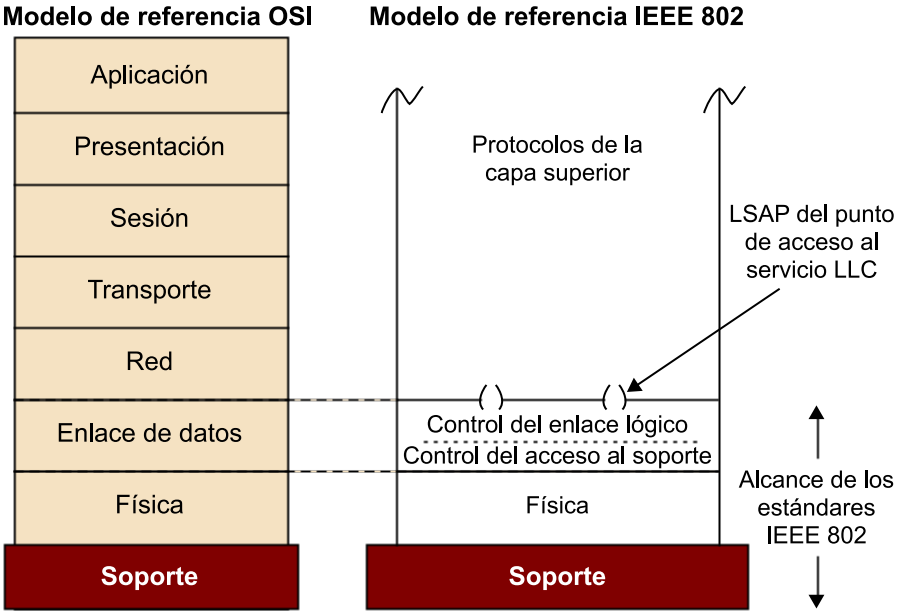
Poco a poco, aunque las prestaciones de rendimiento no fueran muy óptimas, el estándar Ethernet fue ganando terreno a los protocolos basados en testigo y desarrolló nuevas tecnologías basadas en el estándar básico, incrementando la velocidad de transmisión y adaptándose a nuevos tipos de cableado.

Actualmente, Ethernet se ha convertido en el estándar de facto de las redes de área local y es la tecnología LAN de uso más frecuente.

Cuando se diseña una LAN del comité IEEE 802 se deben definir los niveles más bajos del modelo OSI[66]. Hay dos subcapas: el nivel físico y el nivel de enlace, que a su vez se subdivide en dos niveles: LLC y MAC. IEEE definió el siguiente modelo de referencia:

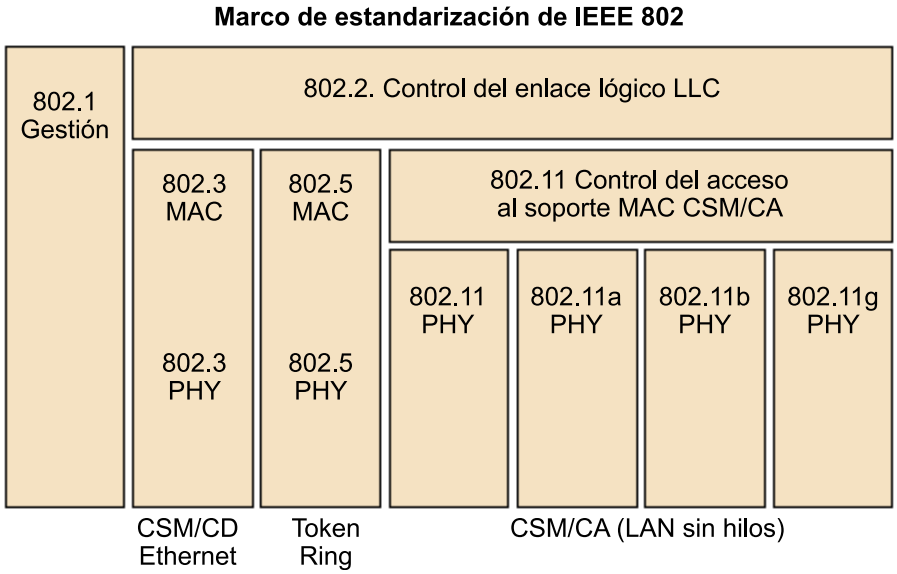
[66]OSI es la sigla de *open systems interconnection*. En castellano, *interconexión de sistemas abiertos*.

Figura 43



El subnivel LLC (IEEE 802.2), basado en el protocolo HDLC, se definió como una interfaz común con los niveles superiores para todos sus estándares de LAN (802.3 Ethernet, 802.4 Token-Bus, 802.5 Token-Ring, 802.11 Wifi, etc.), ocultando la complejidad de los diferentes sistemas de acceso al medio y del formato de las tramas.

Figura 44. Esquema de estandarización IEEE 802.2



7.2.1. Formato de las tramas Ethernet

En la práctica, existen dos versiones de trama Ethernet: el formato Ethernet II o DIX y el formato IEEE 802.3 Los dos formatos son compatibles y se pueden utilizar simultáneamente.

En un principio, el formato Ethernet DIX fue desarrollado por el consorcio Digital, Intel y Xerox (DIX), con la particularidad de que no utilizaba la capa LLC.

Figura 45. Ethernet DIX versión II

Preámbulo	SFD	Destino	Origen	Tipo	Datos	CRC
7	1	6	6	2	46 < datos < 1500	4

Posteriormente, IEEE publicó el estándar 802.3, con el mismo protocolo de acceso CSMA/CD, pero con un pequeño cambio en el formato de las tramas para hacerlo coherente con el estándar IEEE 802.2 (LLC) en el RFC 1042.

Figura 46. IEEE 802.3

Preámbulo	SFD	Destino	Origen	Longitud	Datos y relleno	CRC
7	1	6	6	2	46 < datos < 1500	4

Formato de la trama IEEE 802.3

La descripción de los campos es la siguiente:

- **Preámbulo**: servicios para sincronizar las tarjetas en la recepción de la trama. 7 bytes de ceros y de unos alternados.

- **SFD**[67]: responsable de que las estaciones receptoras sincronicen sus relojes con el mensaje entrante con la finalidad de que no se produzcan errores al leerlo.

- **Direcciones destino y fuente**: identifican la estación transmisora y receptora. Cada NIC tiene un número de identificación de 6 bytes, que es único y está en el hardware de la tarjeta. El organismo de estandarización IEEE suministra bloques de direcciones a las empresas que fabrican tarjetas para garantizar que sean únicas.

- **Tipo** (utilizado en Ethernet DIX): indica el tipo de protocolo de nivel superior (IP, ARP, etc.) que está ocupando el formato de paquete Ethernet DIX versión II. Cuando una trama llega a un ordenador, se necesita saber su tipo para identificar el módulo del software que lo debe utilizar para

(67)SFD es la sigla de *start frame delimiter*.

procesarla. Los valores asignados por el IEEE al RFC1700 con valores superiores a x005DC (1.500 decimal) son:

Ether Type	Protocolo
0800	Datagrama IP
0806	ARP request/reply
8053	RARP
8137	Netware IPX

- ***Length*** (Ethernet IEEE 802.3): define la longitud del campo de datos. No se tienen en cuenta los bytes adicionales. Posee los valores extremos de (46 ≤ *payload* ≤ 1.500 bytes).

- ***Payload***: campo de información. Puede poseer entre 46 y 1.500 bytes. Este campo debe tener un tamaño mínimo para poder detectar las colisiones. Si el número de bytes de información es inferior a 46, Ethernet le añade bytes adicionales hasta completar 46. Debe existir un mecanismo que permita descubrir los bytes que se han añadido. Por ejemplo, en el caso de llevar un datagrama IP, se puede deducir que será a partir del campo *header length* de la cabecera IP. El mínimo tamaño de la trama es: 6 + 6 + 2 + 46 + 4 = 64 bytes (sin preámbulo). El tamaño máximo de la trama es: 6 + 6 + 2 + 1.500 + 4 = 1.518 bytes (sin preámbulo).

- **Condiciones de error**: es *jabber* cuando la longitud de trama mayor de 1.518 bytes (trama larga[68]), y *runt* cuando la longitud de trama es menor de 64 bytes (trama corta[69]). En ese caso se produce un error, a pesar de que el CRC sea correcto. Son frecuentes en una red Ethernet debido a las colisiones. No obstante, las tramas que colisionan tendrán el CRC incorrecto y se descartarán.

- **CRC**: sirve para la detección de errores. El remitente realiza un control CRC (*cyclical redundancy*) para efectuar una revisión de integridad.

[68] En inglés, *long frame*.

[69] En inglés, *short frame*.

La única diferencia entre las tramas Ethernet DIX e IEEE 802.3 es la sustitución del campo *type* por el campo *length*. El campo *length* no tiene en cuenta los bytes adicionales para llegar a los 46; por lo tanto, no es necesario un mecanismo adicional por poder descubrir los bytes que ha añadido el MAC para llegar a la trama mínima.

Los dos formatos de trama se diferencian por:

- *Type* (tramas DIX) < 1.500 (0x05DC) (oficialmente; en la práctica empieza desde 0x0600 o 1.536).

- *Length* (tramas 802.3) > 1.514 máximo.

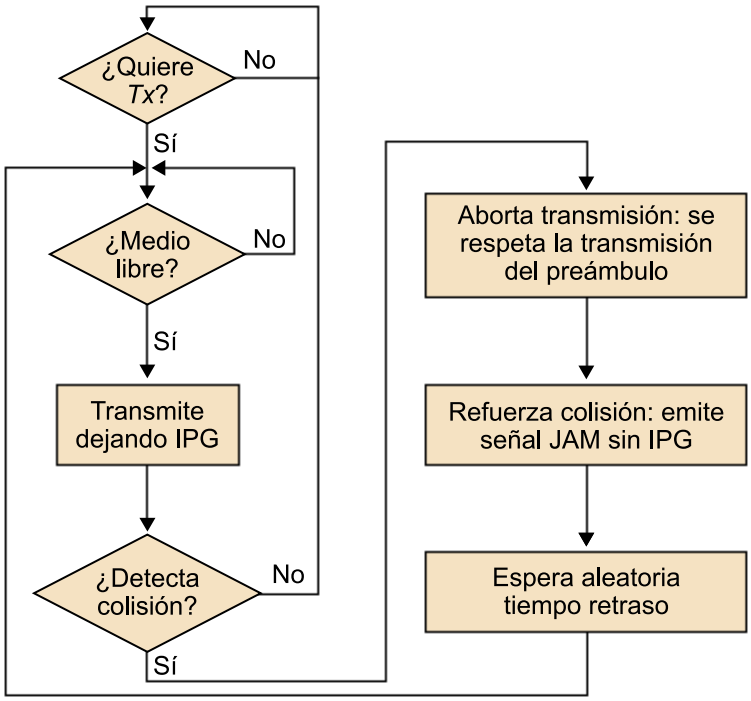
- Permite que las versiones de Ethernet no se confundan y puedan ser utilizadas al mismo tiempo en la misma red LAN.

7.2.2. Funcionamiento del protocolo: CSMA/CD

El protocolo utilizado en Ethernet es el CSMA/CD, una variante del CSMA.

CSMA es más eficiente que Aloha puro y Aloha segmentado, pero cuando las dos tramas colisionan, el canal se vuelve inutilizable mientras dure la transmisión de las tramas que colisionan. Si el tamaño de las tramas es elevado en comparación al tiempo de propagación, se desperdicia una gran cantidad de tiempo. CSMA/CD intenta reducir el tiempo de transmisión de las colisiones.

Figura 47



CSMA/CD se comporta como CSMA 1-persistente. Antes de transmitir escucha el medio y, si está libre, transmite la trama inmediatamente (con probabilidad 1 si el canal está libre).

Si el canal está ocupado, continúa escuchando hasta que queda libre y entonces transmite la trama inmediatamente.

CSMA/CD antes de transmitir debe dejar un tiempo, entre trama y trama, mayor o igual que el IGP[70]. Tiene un valor de 96 bits. Sirve para dar tiempo a las estaciones a detectar si el medio está libre y detectar el final de la recepción de la trama:

[70]IGP es la sigla de *inter packet gap*.

- Tiempo de espera entre tramas consecutivas enviadas por una misma estación.

- Tiempo de espera desde el último bit recibido.

Mientras se transmite la trama, la estación continúa escuchando el canal. Si no se detecta colisión durante la transmisión de la trama, se asume que no hay colisión. Por lo tanto, no es necesario que la estación receptora envíe una confirmación.

Si se detecta una colisión durante la transmisión:

- Se deja de transmitir inmediatamente.

- Se transmite una pequeña señal de interferencia denominada JAM de 32 bits. Ésta sirve para que ninguna estación pueda detectar una trama que ha colisionado con una trama correcta. De esta manera, todas las tarjetas Ethernet descartan la trama.

- Después de transmitir la señal de interferencia JAM, se espera un tiempo aleatorio denominado *back-off* y se intenta transmitir de nuevo la trama.

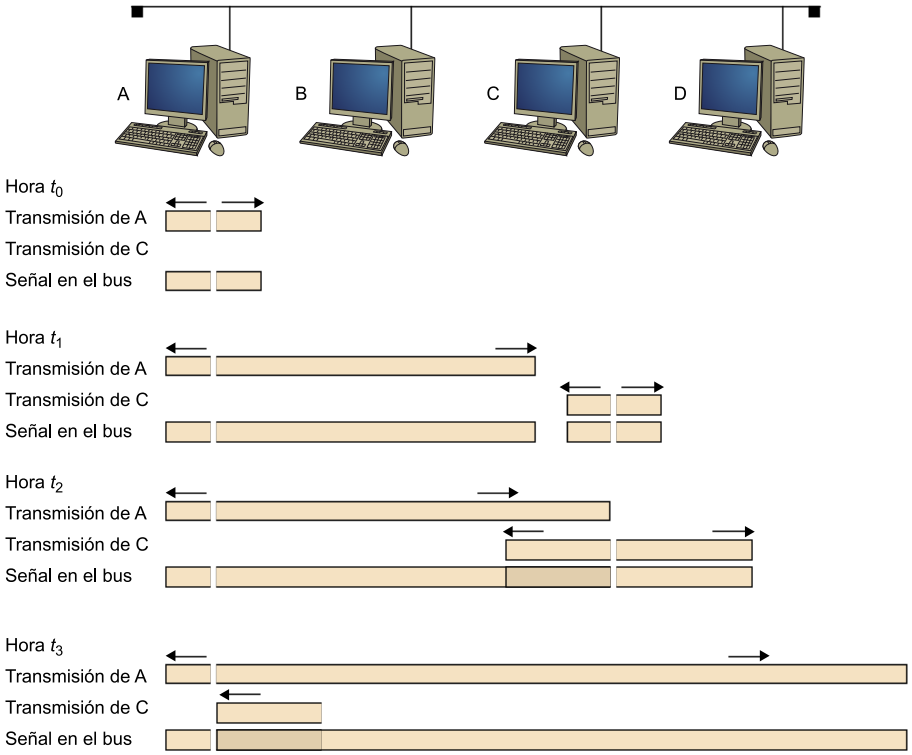
El algoritmo de *back-off* genera un número aleatorio de media que se multiplica por dos cada vez que se retransmite la misma trama. De esta manera se intenta eliminar el problema de la inestabilidad que puede haber en los MAC aleatorios:

$$T_{backoff} = n \cdot T_{t(512)}$$

donde $T_{t(512)}$es el tiempo de transmisión de 512 bits (por ejemplo, 51,2 µs a 10 Mbps). Se denomina tiempo de ranura. n es un número entero aleatorio distribuido uniformemente en $\{0, 2^{mín\{N, 10\}} - 1\}$. $N \geq 1$ es el número de retransmisiones de la trama.

Por ejemplo, para 10 Mbps: *back-off* para $N = 1$ (1.ª retransmisión) = {0, 51,2 µs}, *back-off* para $N = 2$ (2.ª retransmisión) = {0, 51,2, 102,4, 153,6 µs}. Este algoritmo se repite un máximo de 16 veces. Si en la retransmisión número 16 se vuelve a colisionar, la trama se descarta.

Figura 48. Superposición de varias señales sobre el mismo canal



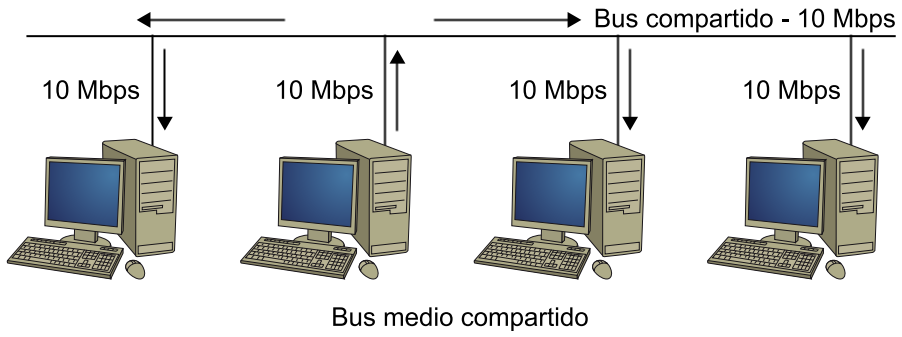
7.2.3. Dominios de colisión y dominio de difusión

Un dominio de colisión es el conjunto de segmentos en el que las estaciones conectadas comparten el mismo medio de transmisión y pueden colisionar directamente entre ellas. Las estaciones Ethernet interconectadas con dispositivos de nivel 1 y 2 (coaxial, concentradores[71]) forman un único dominio de difusión.

[71] En inglés, *hubs*.

Un dominio de difusión define un conjunto de segmentos por los que se envían tramas de difusión. Las tramas de difusión tienen como objetivo llegar a todas las estaciones de la red y a nivel Ethernet se transmiten con la dirección MAC de destino FF:FF:FF:FF:FF:FF. Existen numerosos protocolos que envían tramas de este tipo: ARP, DHCP, DNS, RIP, etc.

Figura 49

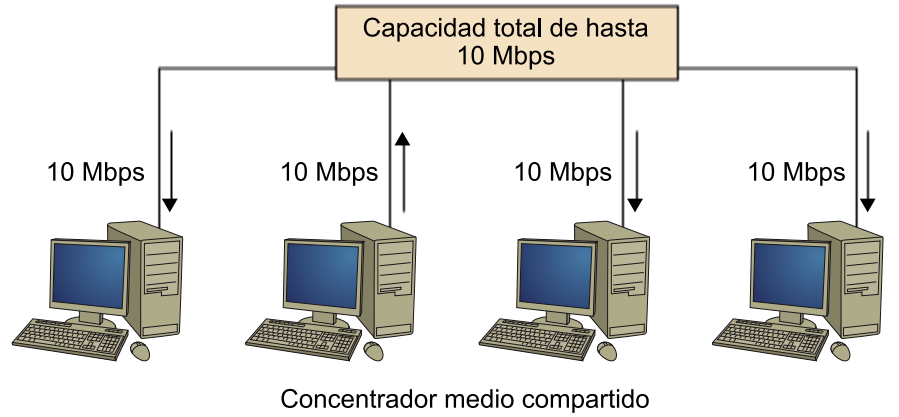


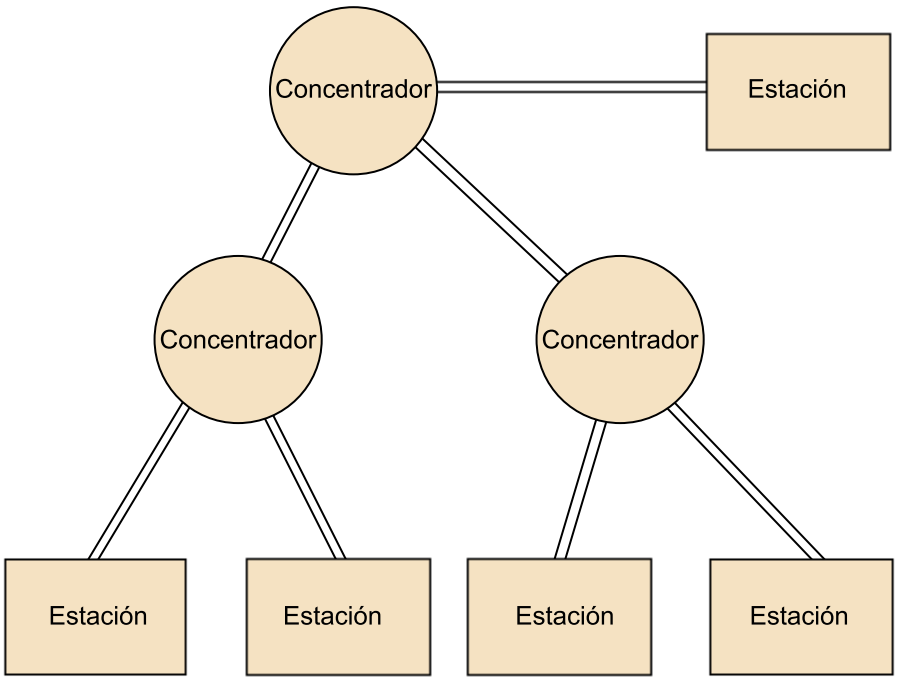
Un repetidor consta sólo de 2 puertos. Un concentrador es un repetidor multipuerto (más de 2 puertos).

Un concentrador es un dispositivo que opera a nivel 1 del modelo OSI (mueven bits entre dispositivos). Regenera y sincroniza la señal. Son dispositivos de red muy económicos. No realiza ninguna función de conmutación. No espera tener toda la trama para empezar a reenviarla al resto de los puertos. No entiende de trama, sólo de bits.

Un concentrador amplía el dominio de colisiones: la red a ambos lados del repetidor es un mismo dominio de colisión. Esto provoca una degradación del rendimiento de la red que depende del número de terminales conectados. Los terminales conectados a un concentrador comparten el ancho de banda. Los concentradores de forma inherente son elementos semidúplex e incrementan el tamaño de un dominio de difusión.

Figura 50





Un puente[72] consta de 2 puertos y tiene el funcionamiento basado principalmente en software. Cada puerto de un puente es un dominio de colisión distinto.

(72)En inglés, *bridge*.

Figura 51. Funcionamiento de un puente

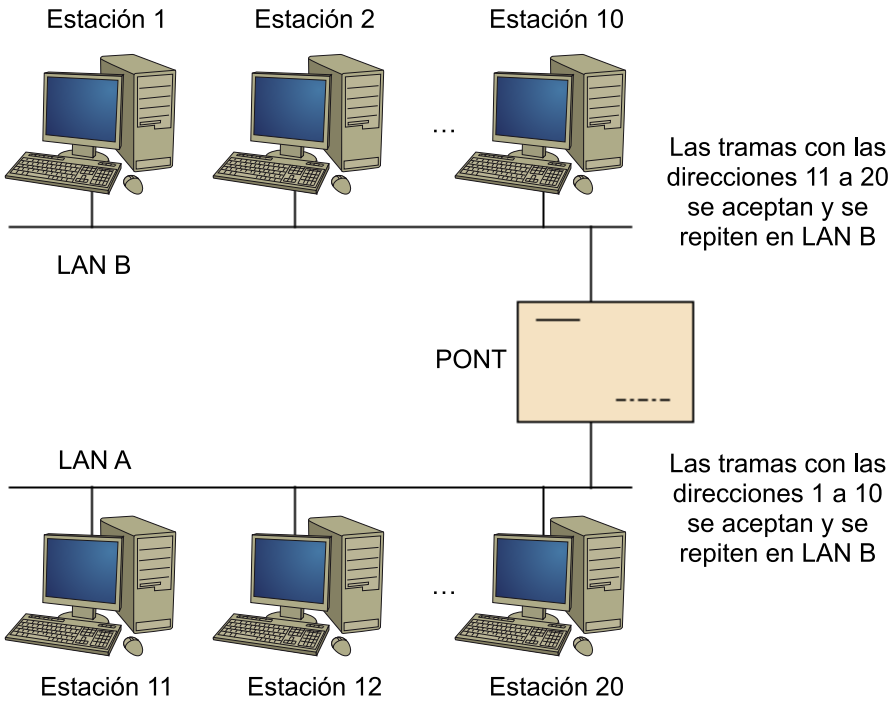
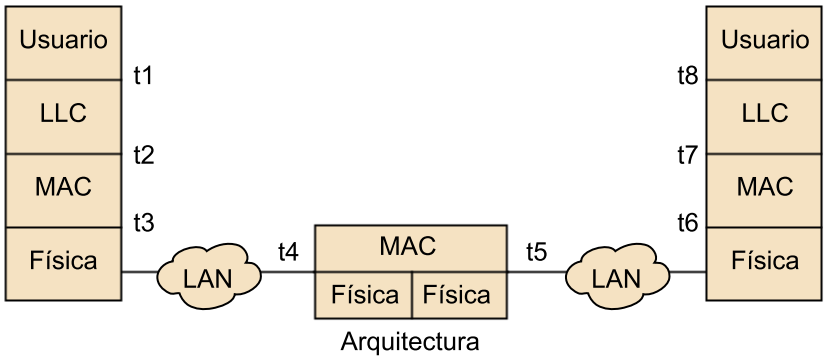


Figura 52. Arquitectura de un puente



Un conmutador[73] consta de más puertos y de mayor capacidad de conmutación que un puente. El funcionamiento de un conmutador está basado sobre todo en hardware. Toman decisiones de encaminamiento basadas en direcciones MAC. Operan a nivel 2 de la torre OSI. Ambos disponen de una tabla MAC con la dupla (dirección MAC, número de puerto) e indican las direcciones MAC conocidas que cuelgan de cada puerto. Las entradas en la tabla MAC de un conmutador deben ser actualizadas dentro de un determinado temporizador[74] (como en ARP):

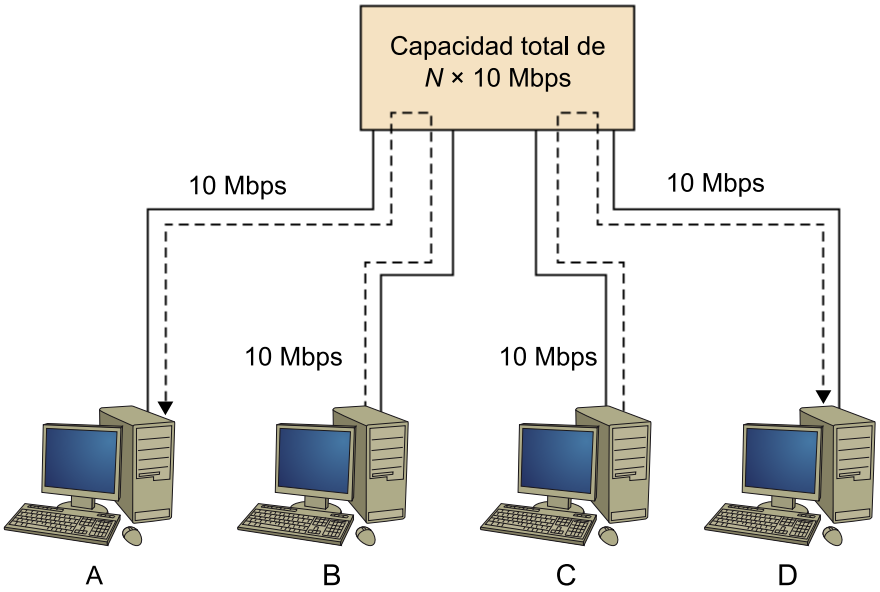
(73)En inglés, *switch*.

(74)En inglés, *time-out*.

- Cada vez que una entrada se utiliza, el temporizador se refresca.
- Si desaparece el temporizador, la entrada se borra de la tabla.

Como los conmutadores son dispositivos *store & forward* (conmutan tramas a nivel 2), no propagan las colisiones y, por lo tanto, segmentan el dominio de colisiones de una red Ethernet (disminuye el tamaño de los dominios de colisión). Aumenta el ancho de banda disponible por usuario, ya que minimiza el tráfico de colisiones. Crea un circuito virtual (camino dedicado) entre dos dispositivos que quieren comunicarse. En principio, ni aumenta ni disminuye el dominio *broadcast*. Los conmutadores con VLAN sí que pueden segmentar el dominio de difusión. Los conmutadores transmiten a mayor velocidad que los encaminadores (nivel 3) y son más económicos.

Figura 53. *Commutadoring hub*



7.2.4. Ethernet conmutada

Los conmutadores son los dispositivos que han permitido evolucionar las antiguas redes Ethernet compartidas multiacceso (construidas mediante un bus o mediante concentradores) a las redes Ethernet conmutadas en las que cada usuario tiene un ancho de banda del 100%.

Fundamentalmente, un conmutador está compuesto por los siguientes elementos: procesador, puertos, detectores de colisión, memoria intermedia, tablas de direcciones MAC y la matriz de conexiones.

La utilización del procesador se debe a que se necesita una gran velocidad del procesamiento de la información (tramas).

Los puertos son las entradas y salidas de los ordenadores u otras redes que pueden tener distintas velocidades (10, 100, 1.000, etc.) y de diferentes tecnologías (Ethernet, *token ring*, etc.).

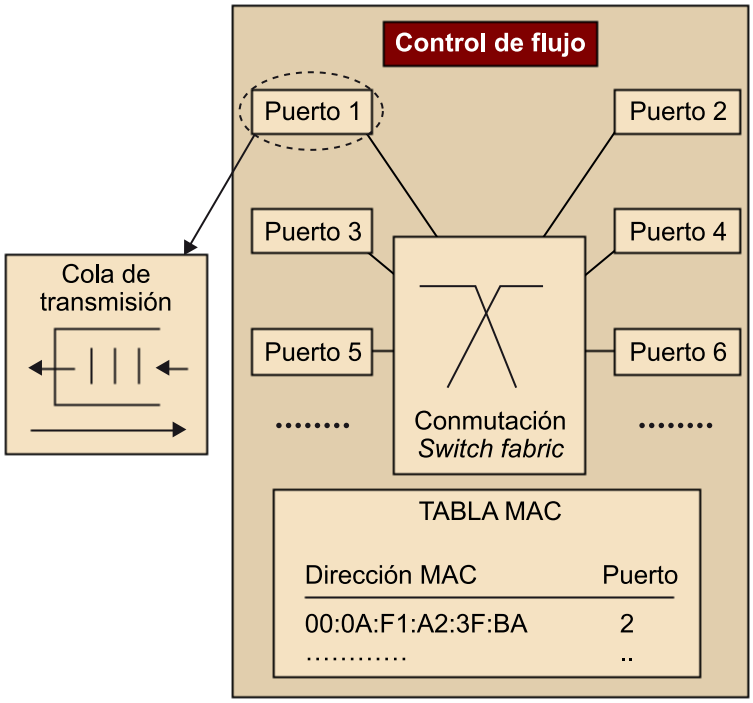
Los detectores de colisión son necesarios cuando utilizamos una transmisión semidúplex en lugar de dúplex. Utilizamos las memorias intermedias para no tener que efectuar repeticiones desde las estaciones. La tabla de direcciones MAC es la que indica dónde está conectado un equipo final o una red. La matriz de conexiones efectúa físicamente las conexiones entre los puertos.

El funcionamiento de un conmutador se basa en el procedimiento de *store & forward*. Cuando llega una trama por un puerto, ésta la guarda para estudiar por qué puerto debe retransmitirla:

- Mira su dirección MAC origen: si la dirección no se encuentra en la tabla MAC o está en un puerto diferente, apunta esta dirección junto al puerto por el que ha llegado. La tabla MAC se construye automáticamente: aprende de las entradas de la tabla MAC observando las direcciones fuente de las tramas recibidas, junto a puerto por el que han llegado.

- Mira su dirección MAC destino y comprueba que se encuentre en la tabla MAC para saber a qué cola de transmisión debe enviar la trama: si la dirección no se encuentra en la tabla MAC (o si es una dirección de difusión), envía la trama a las colas de transmisión de todos los puertos (excepto por el que han llegado). Si la dirección destino se encuentra en la tabla, la trama sólo se pone en la cola de transmisión del puerto que indica la tabla.

Figura 54. Arquitectura genérica de un conmutador Ethernet



Cuando un conmutador recibe una trama, en primer lugar la guarda en una memoria intermedia para después enviarla a los puertos concretos de salida. El conmutador envía la trama desde el puerto origen al puerto destino y, si se produce una colisión, se retransmite la trama desde la memoria intermedia.

(75)FIFO es la sigla de *first in/first out*.

Con este proceso se evita que las estaciones de origen deban volver a reenviar las tramas en caso de colisión. No existe contención, ya que se envían las tramas en el mismo orden en el que llegan, como una cola FIFO[75].

El conmutador puede gestionar la memoria intermedia de dos maneras diferentes: una cola vinculada a cada puerto específico o una memoria compartida para todos los puertos del conmutador.

Existen dos tipos de conmutación:

1) ***Store & forward***: el conmutador guarda toda la trama completa antes de retransmitirla. Ofrece la máxima latencia (retraso) para la comprobación de errores. Con este método, se comprueban todos los campos de la trama con el CRC y, si el valor es correcto, la trama se reenvía. Suele ser el método predeterminado en la gran mayoría de los conmutadores.

2) ***Cut-through***: consiste en enviar una trama tan pronto como se reciba la cabecera de la trama, sin esperar a que se haya recibido completamente. Si se producen errores en la trama, éstos se envían con ella, lo que provoca que las retransmisiones perjudiquen al rendimiento de la red. También tiene una opción (*fragment free*) para filtrar los fragmentos de colisión antes de realizar la conmutación.

7.2.5. STP/RSTP

Las topologías redundantes ofrecen protección ante la caída de un determinado enlace, puerto o dispositivo. Sería deseable que hubiera bucles para tener varios caminos alternativos, así, si uno dejase de funcionar, habría otro camino alternativo. A pesar de ello, las topologías conmutadas presentan ciertos problemas como en las tormentas de difusión, transmisiones de múltiples copias de tramas e inestabilidad en las tablas MAC de los conmutadores. El algoritmo STP[76] (IEEE 802.1d) o el RSTP[77] permiten crear topologías libres de lazos a partir de topologías físicas con lazos redundantes. Los ordenadores que utilizan STP/RSTP intercambian un conjunto de mensajes BPDU para dejar una topología libre de bucles (topología en árbol o estrella). Por ello, puede llegar a desconectar/bloquear un puerto si existe otro camino en el mismo segmento. Los puertos bloqueados descartan todas las tramas de datos que reciben y sólo capturan los mensajes RSTP. De este modo, pueden pasar al estado normal de funcionamiento si hay un cambio en la topología que lo requiera.

(76)STP es la sigla de *spanning tree protocol*.

(77)RSTP es la sigla de *rapid spanning tree protocol*.

7.2.6. Ethernet semidúplex

En general, las tarjetas dúplex tienen un mecanismo de autonegociación que permite detectar si es posible activarlo. En Ethernet existen dos tipos de comunicaciones:

1) Semidúplex: un único dispositivo puede enviar y recibir información a la vez. Si varios dispositivos quieren comunicarse al mismo tiempo, se producirán colisiones. Cuando se produce una colisión la estación deja de transmitir. Una conexión de este tipo es la que se produce cuando varios *hosts* están conectados a un concentrador; por su modo de funcionar, si la señal se recibe por un puerto de entrada, debe enviarla a los otros puertos, y si recibe otra señal por un puerto diferente, no podrá enviar simultáneamente estas señales. El modo de funcionamiento semidúplex se implementa con el protocolo CSMA/CD.

2) Dúplex: las comunicaciones permiten que dos dispositivos se comuniquen entre sí de manera simultánea. Cuando se conecta una tarjeta semidúplex a un conmutador con UTP, se activa el modo dúplex y se desactiva el mecanismo CSMA/CD. Cuando la red funciona como dúplex se duplica la capacidad del enlace y no se producen colisiones, ya que los dos dispositivos forman un único dominio de colisión. El cable UTP tiene 4 pares de cables que ofrecen muchas posibilidades: 10BaseT y 100BaseTX utilizan 2 pares, uno para la transmisión y otro para la recepción, simultáneamente. La colisión se detecta porque se recibe la señal por el par de recepción, mientras se transmite por el par de transmisión. 100BaseT4 utiliza un par para transmitir, otro para recibir y dos más para transmitir/recibir al mismo tiempo.

Asimismo, los puertos 10/100 Mbps suelen tener un mecanismo de autonegociación que permite detectar la velocidad de transmisión del dispositivo que se conecta. Lo realizan mediante un control de flujo con tramas especiales *jabber* (modo semidúplex) o *pause* (modo dúplex).

7.2.7. LAN virtuales

Una LAN virtual (VLAN) es un agrupamiento lógico de dispositivos de red o de estaciones que no están sujetos a un agrupamiento físico. Permiten agrupar dispositivos por funciones, equipos, departamentos o aplicaciones en la organización empresarial.

El conmutador tiene una tabla MAC diferente para cada VLAN. Cada conmutador aísla los puertos que pertenecen a VLAN diferentes. Por ello, si llega una trama de difusión por un puerto, el conmutador sólo la reenvía por los otros puertos que pertenecen a la misma VLAN. Una VLAN es un único dominio de difusión creado por uno o más conmutadores, que no está sujeta a ningún segmento físico y es tratada como una subred. La comunicación entre diferentes VLAN se realiza mediante un dispositivo de nivel 3 (por ejemplo, un encaminador).

La creación de las VLAN mejora el rendimiento, la seguridad de la red conmutada y se controla la propagación de la difusión. Proporciona segmentación y flexibilidad organizativa. Permite agrupar a usuarios por funciones lógicas y no por ubicación física. Admite servidores y ordenadores relacionados entre diferentes dominios de difusión, cada uno identificado con una red. Simplifica la tarea de agregar y mover recursos por una subred.

Existen dos tipos de VLAN:

1) VLAN estática: basada en puertos. Cada puerto de un conmutador se asocia estáticamente a una VLAN. Todas las estaciones vinculadas a este puerto pertenecen a la VLAN que tiene asociada. Es el método más utilizado. Es segura y fácil de configurar y controlar.

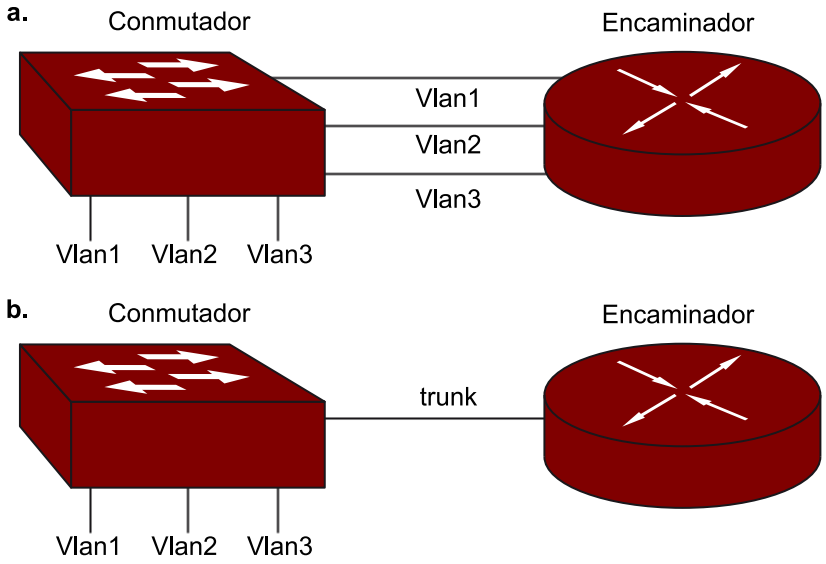
2) VLAN dinámica: basada en direcciones lógicas (IP) o en direcciones físicas (MAC). En un mismo puerto puede haber ordenadores de diferentes VLAN. Cada conmutador detecta que una nueva estación se conecta a un puerto y consulta una base de datos para saber a qué VLAN pertenece. En este caso, la VLAN a la que pertenece el ordenador se identifica por dirección MAC del ordenador. Cuando una estación se conecta, el puerto se configura automáticamente con la configuración de la VLAN correspondiente mediante su dirección MAC.

El *VLAN trunking protocol* se desarrolló para gestionar la transferencia de tramas de diferentes VLAN a través de una sola línea física[78]. Sirve para evitar enlazar un cable por cada VLAN entre dos conmutadores o entre un encaminador y un conmutador. Un enlace de línea física agrupa múltiples enlaces virtuales sobre un único enlace físico, añadiendo unas etiquetas especiales a las tramas e identificar así a qué VLAN pertenece la trama.

[78] En inglés, *trunk*.

En la figura 55 se muestran dos alternativas para interconectar VLAN mediante un encaminador: sin o con línea física.

Figura 55. Interconexión de VLAN mediante un encaminador: sin o con línea física



a. Utiliza muchos puertos. No es escalable. **b.** Ahorra puertos cableados. Permite añadir VLAN sin coste. Requiere el protocolo para etiquetar las tramas ISL o IEEE 802.1Q.

Los dos métodos, ISL[79] e IEEE 802.1Q, consisten en añadir un identificador[80] a la cabecera de la trama cuando está encaminada por el conmutador. El identificador identifica la VLAN a la que pertenece la trama. Cuando la trama se envía por un puerto que no tiene *trunking*, el conmutador elimina al identificador antes de enviarlo a la estación destino.

[79] ISL es la sigla de *inter switch link*.

[80] En inglés, *tag*.

7.2.8. Tecnologías Ethernet

El comité IEEE ha definido diferentes configuraciones físicas alternativas que ha tenido esta tecnología, lo que proporciona una gran variedad de opciones.

Nombre comercial	Estándar	Denominación	Cable	Pares UTP	Dúplex	Conector	Codificación	Distancia segmento
Ethernet	802.3	10Base5	Coaxial Thick		No	AUI	Manchester	500 m
	802.3a	10Base2	Coaxial Thin		No	BNC	Manchester	185 m
	802.3i	10BaseT	UTP cat.3	2	Sí	RJ45	Manchester	100 m
Fast Ethernet	802.3u	100BaseTX	UTP cat.5	2	Sí	RJ45	4 B/5 B	100 m
	802.3u	100BaseT4	UTP cat.3	4	No	RJ45	8 B/6 T	100 m
Gigabit Ethernet	802.3ab	100BaseT	UTP cat.5	4	Sí	RJ45	8 B/10 B	100 m

La nomenclatura de Ethernet utilizada es **XBaseY**, donde:

- X es la velocidad de transmisión en Mbps.
- **Base** es la codificación en banda base.

- **Y** puede tener varios significados: si es un número, hace referencia a la distancia máxima (aproximada) del segmento en centenares de metros. Puede hacer referencia al tipo de medio de transmisión (T: par trenzado, F: fibra óptica) y puede tener alguna otra característica (4: utiliza los 4 pares trenzados, X: dúplex).

1) Especificaciones IEEE 802. 3 10 Mbps (Ethernet)

a) 10BaseT

- Utiliza el cableado que se ha convertido en el más económico: UTP (par trenzado no apantallado) con los conectores RJ45.

- Topología en estrella. La señalización es Manchester Digital.

- La longitud máxima del cable es 100 metros.

- Utiliza cable UTP y conectores RJ45 en la NIC y el repetidor.

- En 10BaseT las estaciones se conectan mediante un concentrador. El concentrador regenera y amplifica la señal que recibe por un puerto y la transmite al resto de los puertos con un retraso de pocos bits.

- Sin embargo, en la actualidad los cableados con coaxial (10Base5 y 10Base2) han quedado obsoletos en favor del cableado con UTP. Si se necesita cubrir distancias mayores de las que permite UTP, se utiliza fibra óptica.

- El modo de funcionamiento puede ser semidúplex o dúplex, en función del dispositivo conectado.

Especificaciones 10BaseX				
	10Base5	**10Base2**	**10BaseT**	**10Base-FP**
Transmission medium	*Coaxial cable* (50 ohm)	*Coaxial cable* (50 ohm)	*Unshielded twister pair*	850-nm *optical fiber pair*
Signaling Technique	*Baseband* (Manchester)	*Baseband* (Manchester)	*Baseband* (Manchester)	Manchester/*On-Off*
Topology	*Bus*	*Bus*	*Star*	*Star*
Maximum segment length (m)	500	185	100	500
Nodos por segmento	100	30	----	33
Cable diameter (mm)	10	5	0.4 *to* 0.6	62.5/125 µm

2) Especificaciones IEEE 802. 3 100 Mbps (*fast Ethernet*)

Es el conjunto de especificaciones desarrolladas por el comité IEEE 802.3 para proporcionar mayor velocidad a las LAN.

a) 100Base-TX

- También conocido como 100Base-X. Se permite la utilización tanto de STP como de UTP de cat5.
- Utiliza dos pares de cable de par trenzado: un par para transmisión y otro para recepción.
- La longitud máxima es de 100 metros.
- Utiliza la señalización 4B/5B-NRZI. Utiliza grupos de 5 bits para enviar 4 bits de datos.

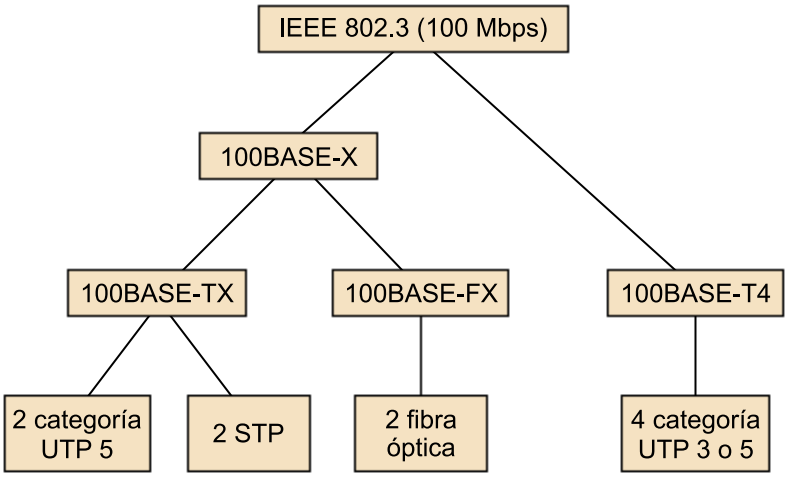
b) 100Base-FX

- Utiliza dos fibras ópticas, una para transmisión y otra para recepción, modo dúplex.
- Es necesario un convertidor optoelectrónico que convierta la secuencia de grupos del código 4B/5B-NRZI en señales ópticas.
- La longitud máxima es 100 metros.

c) 100Base-T4

- Se utilizan los 4 pares trenzados: tres se utilizan para la transmisión con una velocidad efectiva de 33,3 Mbps y el otro, junto con los dos empleados en la transmisión, se utilizan para la recepción. Hay dos pares que se deben configurar para una transmisión bidireccional.
- La longitud máxima es 100 metros. Utiliza la señalización 8B/6T.
- Es utilizado por redes que necesitan baja calidad de pares trenzados en una red de 100 Mbps Ethernet.

Figura 56. Diagrama 100BaseX



3) Especificaciones IEEE 802.3 1000 Mbps (*gigabit Ethernet*)

a) 1000Base-CX

Estándar *gigabit Ethernet* sobre cable de cobre que ha sido reemplazado por 1000Base-T.

b) *Gigabit Ethernet* 1000Base-T (802.3z/802.3ab)

- Utiliza los 4 pares UTPcat5 para enviar/recibir simultáneamente.
- Codificación 8B/10B.
- Distancia típica de 1.000 metros.

c) 1000Base-SX

- Estándar *gigabit Ethernet* sobre fibra óptica que opera sobre fibra multimodo.

- Típica distancia hasta 550 metros.

d) 1000Base-LX

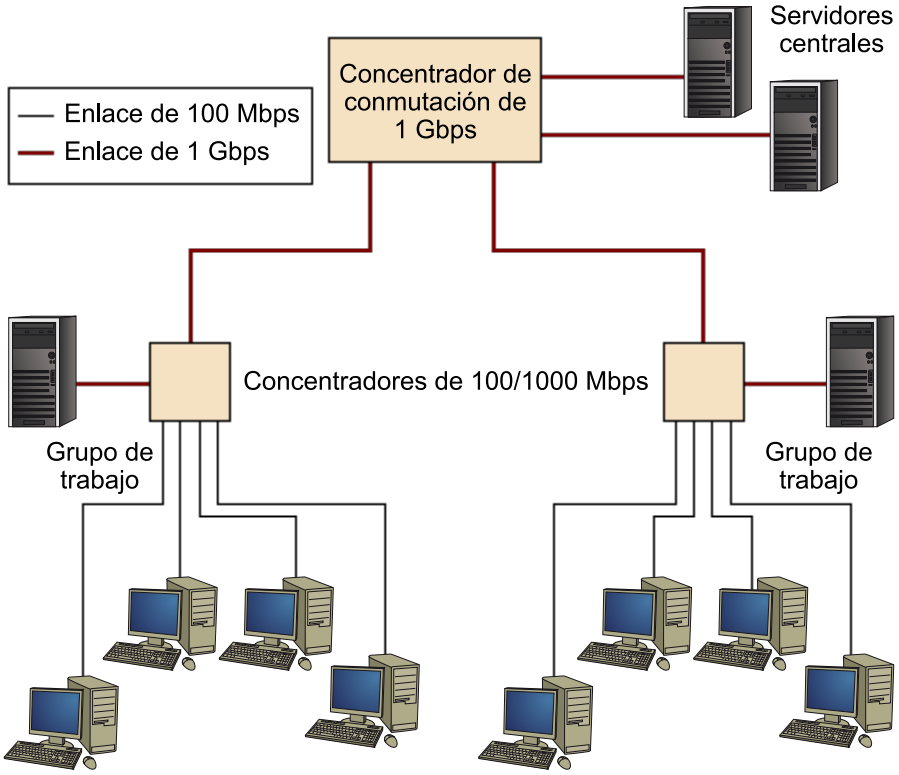
- Estándar *gigabit Ethernet* sobre fibra óptica que opera sobre fibra monomodo.

- Típica distancia de 550 metros a 5 kilómetros.

e) 10 *gigabit Ethernet* (802.3ae)

Opera sólo en modo dúplex y con fibra óptica.

Figura 57. Los enlaces 1000BaseX se pueden utilizar para interconectar redes LAN de menos velocidad



Cuadro resumen de varias especificaciones 10xzBaseY

	10Base 2	10Base 5	10Base -T	100Base -TX	1000Base -FX	1000Base -CX	1000Base -T	1000Base -SX	1000Base -LX
Medios	Coaxial de 50 ohms (Thinnte)	Coaxial de 50 ohms (Thinteh)	UTP categoría 3,4,5 EIA/TIA, dos pares	UTP categoría 5 EIA/TIA dos pares	Fibra multimodo 62,5/125	STP	UTP categoría 5 EIA/TIA cuatro pares	Fibra micro multimodo 62,5/50	Fibra micro multimodo 62.5/50; fibra monomode de 9 micrones
Longitud de segmento máximo	185 m	500 m	100 m	100 m	400 m	25 m	100 m	275 m por microfibra 62,5;550 m por microfibra de 50	440 m por microfibra 62.5;550 m por microfibra de 50; de 3 a 10 km por fibra monomode
Topología	Bus	Bus	Estrella	Estrella	Estrella	Estrella	Estrella	Estrella	Estrella
Conector	BNC	AUI (Interfaz de unidad de conexión)	RJ-45 ISO 8877	RJ-45 ISO 8877		RJ-45 ISO 8877	RJ-45 ISO 8877		

7.3. Redes inalámbricas

Las ventajas de las redes sin hilo son la movilidad y su flexibilidad, la facilidad de instalación, la escalabilidad y el dinamismo en los cambios de la topología, así como su capacidad para llegar adonde no puede llegar el cable. Su elevado coste inicial y su seguridad son los principales inconvenientes.

Su ámbito de aplicación es muy amplio. Son de gran utilidad en edificios históricos, en entornos cambiantes, en los que hay usuarios en movimiento (hospitales, oficinas, fábricas, etc.), en grupos de trabajo eventuales, en ambientes industriales donde las condiciones medioambientales son problemáticas, en usos domésticos, etc.

7.3.1. Características de las redes inalámbricas

Las diferentes tecnologías inalámbricas se suelen agrupar basándose en el radio de acción (el alcance) de cada una de ellas:

- Redes personales inalámbricas (WPAN[81]): este concepto se aplica cuando la distancia que se quiere cubrir es del orden de unos cuantos metros. Las familias de estándares más representativas son 802.15.1 (Bluetooth), 802.15.3a (UWB) y 802.15.4 (Zigbee).

(81)WPAN es la sigla de *wireless personal area network*.

- Redes locales inalámbricas (WLAN[82]): permiten dar servicios a distancias del orden de un centenar de metros (un piso, la planta de un edificio, una nave industrial, unas calles, etc.). El estándar más destacado en este campo es 802.11 (Wi-Fi).

(82)WLAN es la sigla de *wireless local area network*.

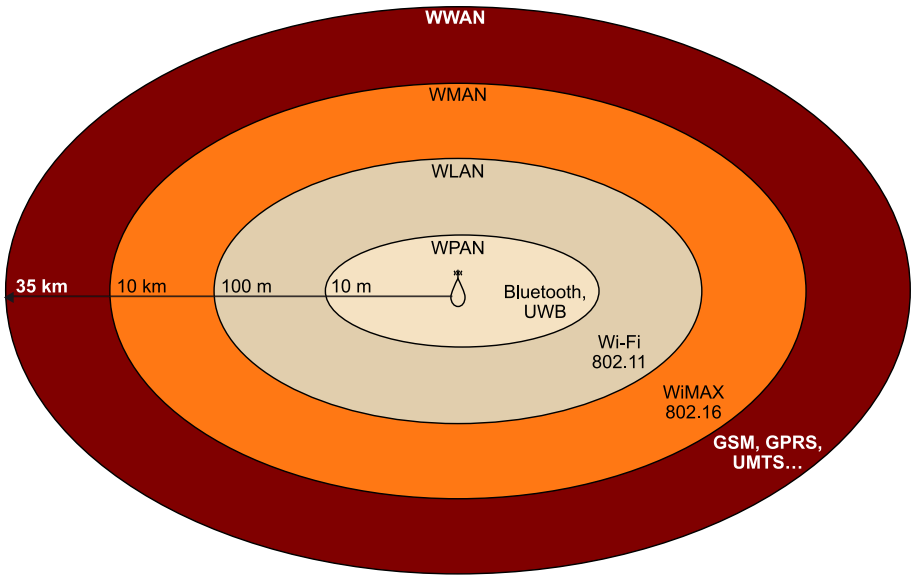
- Redes metropolitanas inalámbricas (WMAN[83]): permiten dar servicios a distancia del orden de unos cuantos kilómetros (un barrio, un pueblo, una urbanización, etc.). El estándar más destacado en este campo es 802.16 (WiMAX).

(83)WMAN es la sigla de *wireless metropolitan area network*.

- Redes de gran alcance inalámbricas (WWAN[84]): tienen una cobertura más amplia. Las familias de estándares más representativos son GSM, GPRS y UMTS.

(84)WWAN es la sigla de *wireless wide area network*.

Figura 58. Clasificación de las tecnologías inalámbricas



7.3.2. Wi-Fi-IEEE 802.11

El estándar IEEE 802.11, también denominado *wireless Ethernet*, fue aprobado en 1997. Se ideó para desarrollar LAN dentro de la banda de frecuencias ISM: banda de 2,4 GHz, pensada para usos industriales, científicos, médicos y no comerciales sin autorización administrativa de ningún gobierno. Se utiliza dentro de zonas geográficas muy limitadas. Este estándar especifica una interfaz aérea entre un cliente sin hilo y una estación base o entre dos clientes sin hilo.

El término *Wi-Fi*[85] hace referencia al conjunto de estándares para redes sin hilo basado en las especificaciones IEEE 802.11x. Fue creado por la Wi-Fi Alliance. Todo producto que ha sido testado y aprobado por la Wi-Fi Alliance lleva el texto "Wi-Fi Certified", lo que garantiza la interoperatibilidad. Las principales características de las diferentes especificaciones IEEE 802.11x se detallan en el siguiente cuadro:

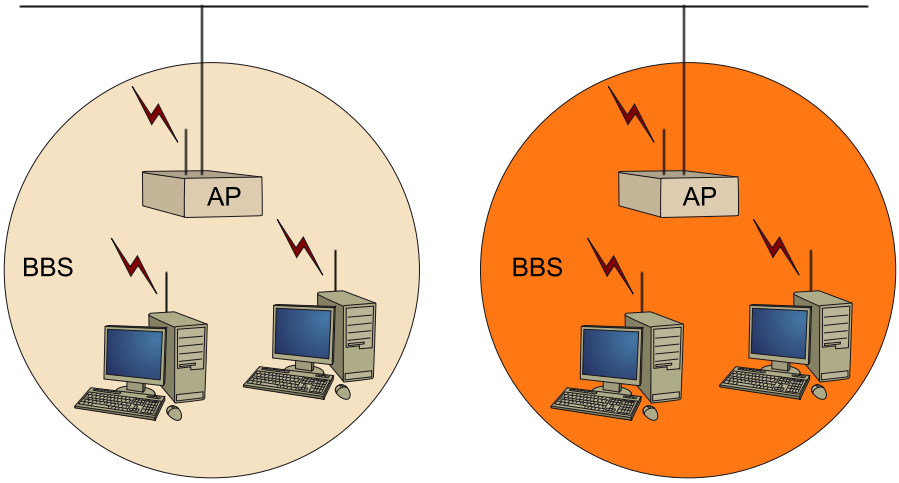
(85)En inglés, *wireless fidelity*.

Protocolo	Año	Frecuencia operación	Esquema de modulación	Velocidad máxima	Rendimiento	Seguridad
802.11	1997	2,4-2,5 GHz	FHSS o DSSS	2 Mbps		WEP y WPA/WPA2
802.11a	1999	5,15-5,35/ 5,47-5,725/ 5,725-5,875 GHz	OFDM	54 Mbps	25 Mbps	WEP y WPA/WPA2
802.11b	1999	2,4-2,5 GHz	DSSS amb CKK	11 Mbps	6 Mbps	WEP y WPA/WPA2
802.11g	2003	2,4-2,5 GHz	OFDM sobre 20 Mpbs, DSSS con CKK sobre 20 Mbps	54 Mbps	22 Mbps	WEP y WPA/WPA2
802.11n	2008	2,4 GHz o 5 GHz bandas		540 Mbps		WEP y WPA/WPA2

Arquitectura de red

La arquitectura básica de una LAN inalámbrica es la siguiente:

Figura 59



Se denomina BSS[86] y normalmente contiene una o más estaciones inalámbricas y una estación base central, conocida como AP[87]. Las estaciones inalámbricas pueden estar fijas o ser móviles, y se comunican con la estación base central con el protocolo MAC IEEE 802.11. Se conoce como el modo de infraestructura. Esta topología utiliza el concepto de *celda*, que es el área en la que una señal radioeléctrica es efectiva.

(86)BSS es la sigla de *basic service set*.

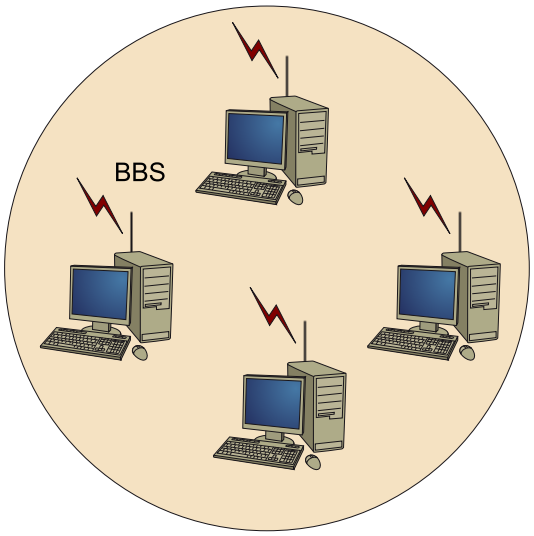
(87)AP es la sigla de *access point*.

Varios AP se pueden conectar entre sí (por ejemplo, utilizando una Ethernet cableada u otro canal inalámbrico) y formar lo que se denomina un sistema de distribución (DS). En este caso, el sistema DS aparece en las capas superiores (por ejemplo, en el nivel IP) como una red 802.

También las estaciones pueden formar una BSS o un grupo de estaciones entre ellas, de igual a igual[88], sin control central, y se dice que funciona a modo de red *ad hoc*. En general, se utiliza con estaciones que casualmente encuentran y desean comunicarse. Sólo es necesario dos equipos con su correspondiente adaptador inalámbrico.

(88)En inglés, *peer to peer*.

Figura 60



Capa física

El estándar IEEE 802.11 define la capa física y la capa de acceso al medio (MAC). La capa MAC del IEEE 802.11 asume funciones que en general son asumidas en otros protocolos por las capas superiores, como la fragmentación, la recuperación de errores, el control de la movilidad y la conservación de la potencia.

La capa física utiliza un espectro extendido por secuencia directa (DSSS[89]), que codifica cada bit en una cadena de bits, denominado código. Esta técnica es muy similar a la utilizada en CDMA, excepto en que todas las estaciones móviles (o estaciones base) utilizan el mismo código. Debido a la utilización del mismo código por parte de todas las estaciones que configuran la red, DSSS no es un protocolo de acceso múltiple: se trata de un mecanismo de la capa física que a partir de una señal emite una energía sobre un rango de frecuencias concreto, lo que provoca que el receptor pueda recuperar la señal original.

(89)DSSS es la sigla de *direct sequence spread spectrum*.

Existen dos tipos de medios para la instalación de redes inalámbricas: por ondas de radiofrecuencia y por señales ópticas de infrarrojos.

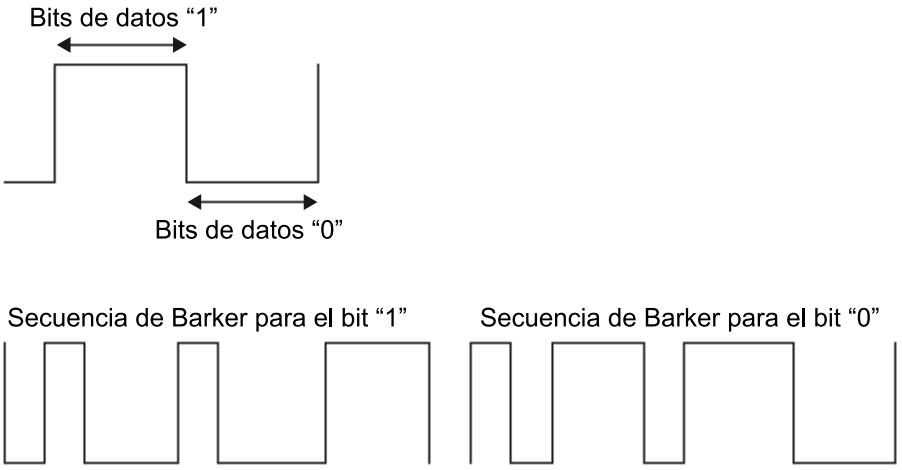
La capa IEEE 802.11 define tres posibles esquemas de la capa física: DSSS, FHSS[90] (espectro extendido por orden de frecuencia) y luz infrarroja en banda base (sin modular).

(90)FHSS es la sigla de *frequency hopping spread spectrum*.

Por radiofrecuencia utiliza las bandas de 2,4 GHz y 5,7 GHz. No tienen problemas para propagarse mediante los obstáculos. El DSSS genera un patrón de bits pseudoaleatorio (señal de chip) por cada uno de los bits que configuran la señal. Cuanto mayor sea el patrón, más resistentes son los datos a posibles interferencias (de 10 a 100 bits). Esta secuencia es conocida como secuencia de Barker (o código de dispersión). La secuencia está equilibrada, es decir, aproxi-

madamente existe la misma cantidad de ceros que de unos. Todas las estaciones conocen la secuencia utilizada. Esta secuencia proporciona una ganancia de procesamiento (por 10 bits, se obtiene una G = 10 dB, por 100 bits, G = 20 dB). La ganancia debe ser mayor o igual que el SNR (relación señal/ruido).

Figura 61. Codificación de Barker



En DSSS, la modulación en la frecuencia 2,4 GHz utiliza variaciones en fase de una sola portadora en amplitud constante: DBPSK[91] y DQPSK[92]. En la banda de 5,7 GHz se utilizan variaciones de frecuencia de múltiples portadoras: OFDM[93].

(91)DBPSK es la sigla de *differential binary phase shift keying*.

(92)DQPSK es la sigla de *differential quadrature phase shift keying*.

(93)OFDM es la sigla de *ortogonal frecuency division multiplexing*.

En canales de Europa y Estados Unidos, DSSS utiliza un rango de frecuencias de 2,400 GHz-2,4835 GHz. Esto da un ancho de banda de 83,5 MHz. Se subdivide en canales de 5 MHz cada uno, lo que nos proporciona un total de 14 canales independientes. Cada estado está autorizado a emplear un subconjunto de estos canales. En España, se usan los canales 10 y 11, correspondientes a frecuencias centrales de 2,457 GHz y 2,462 GHz, respectivamente. Los identificadores de canales, frecuencias centrales y dominios reguladores para cada canal usado por IEEE 802.11b e IEEE 802.11g son los siguientes:

Canal	Frecuencia en MHz	Dominios reguladores				
		América (-A)	EMEA (-E)	Israel (-I)	China (-C)	Japón (-J)
1	2.412	X	X	–	X	X
2	2.417	X	X	–	X	X
3	2.422	X	X	X	X	X
4	2.427	X	X	X	X	X
5	2.432	X	X	X	X	X
6	2.437	X	X	X	X	X
7	2.442	X	X	X	X	X
8	2.447	X	X	X	X	X
9	2.452	X	X	X	X	X

Canal	Frecuencia en MHz	Dominios reguladores				
		América (-A)	EMEA (-E)	Israel (-I)	China (-C)	Japón (-J)
10	2.457	X	X	–	X	X
11	2.462	X	X	–	X	X
12	2.467	–	X	–	–	X
13	2.472	–	X	–	–	X
14	2.484	–	-	–	–	X

En la capa física que utiliza FHSS, la modulación en la banda de 2,4 GHz utilizada es la FSK (modulación en frecuencia), según el siguiente cuadro:

Rango frecuencias centrales con FHSS			
Límite inferior	Límite superior	Rango regulatorio	Área geográfica
2,402 GHz	2,480 GHz	2,400-2,4835 GHz	Norteamérica
2,402 GHz	2,480 GHz	2,400-2,4835 GHz	Europa
2,473 GHz	2,495 GHz	2,471-2,497 GHz	Japón
2,447 GHz	2,473 GHz	2,445-2,475 GHz	España
2,448 GHz	2,482 GHz	2,4465-2,4835 GHz	Francia

La banda de frecuencias en FHSS asignada se divide en subbandas de menor frecuencia, denominadas canales, con el mismo ancho de banda. Cada tramo de información se transmitirá a una frecuencia distinta durante un intervalo de tiempo muy corto[94] (menor que 400 ms), saltando a continuación a una frecuencia diferente. El patrón de uso del canal es pseudoaleatorio. La secuencia de orden se guarda en tablas, que las conocen tanto el emisor como el receptor. Así, la banda de 2,4 GHz se organiza en 79 canales, con un ancho de banda de 1 MHz cada uno. El número de órdenes por segundo se regula en cada país. Por ejemplo, en Estados Unidos es de 2,5 órdenes/segundo.

(94)En inglés, *dwell*.

La capa física por infrarrojos se utiliza en entornos muy localizados, en una sola área o habitación. Emplea unas frecuencias de emisión entre $3{,}10 \cdot 10^{14}$ y $3{,}52 \cdot 10^{14}$ Hz. El comportamiento es similar al de la luz. Los inconvenientes de este sistema son que no atraviesa los objetos sólidos, que tiene poca capacidad de difusión y que es demasiado sensible a objetos móviles, a la luz solar directa y a las lámparas. Las restricciones de potencia de emisión limitan la cobertura a decenas de metros. Se produce dispersión y rebotes, que provocan interferencias y limitan la velocidad de transmisión.

7.3.3. CSMA/CA

El protocolo utilizado en IEEE 802.11 es el CSMA/CA[95]. Lo primero que observa el protocolo CSMA es el canal, para determinar si está ocupado o no por otra estación que esté transmitiendo una trama. En una especificación inalámbrica, la capa física monitoriza el nivel de energía de las ondas de radio en una determinada frecuencia para determinar si una estación ocupa o no el canal (el aire), y envía esta información a su capa MAC. Si se observa que el canal está libre por un tiempo igual o superior al DIFS[96], la estación está autorizada a transmitir. Esta trama será recibida por la estación receptora si ninguna estación ha interferido con la transmisión de esta trama.

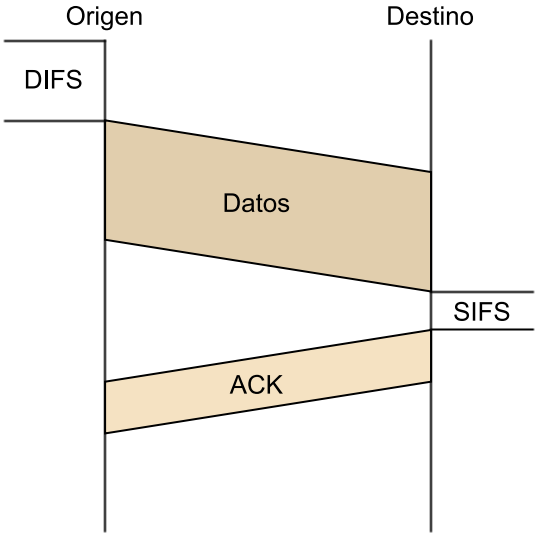
(95)CSMA/CA es la sigla de *CSMA with collision avoidance.*

(96)DIFS es la sigla de *distributed inter frame space.*

Cuando la estación ha recibido correcta y completamente una trama, espera un espacio corto de tiempo, denominado SIFS[97], y envía una confirmación explícita hacia el emisor indicándole que ha recibido correctamente la trama. Esta trama se debe enviar, ya que, en un entorno abierto como es el aire, el emisor, por sí sólo, no puede determinar si se ha producido una colisión o no.

(97)SIFS es la sigla de *short inter frame spacing.*

Figura 62

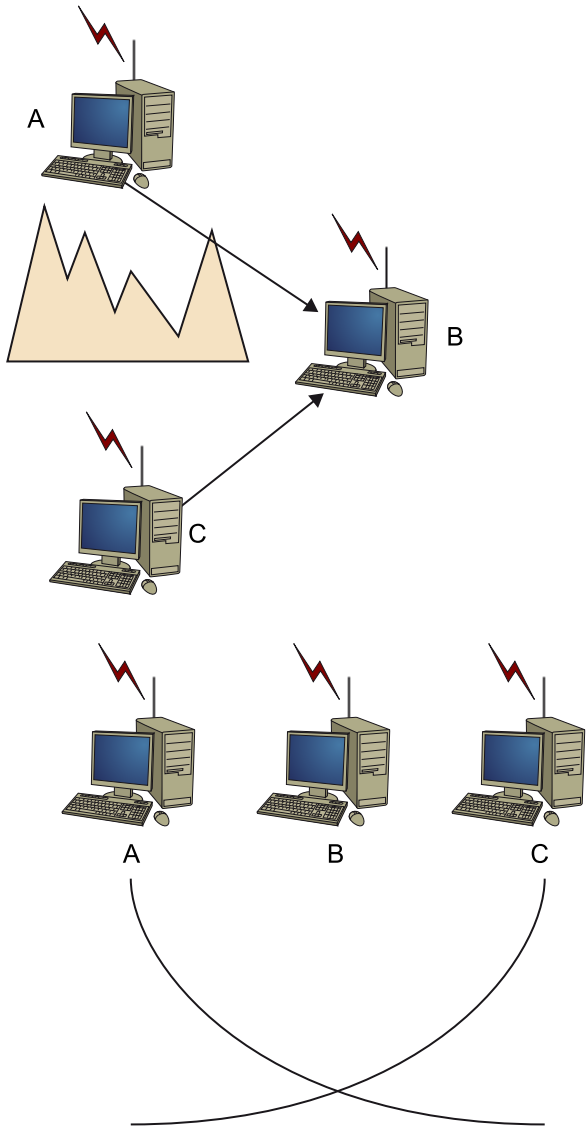


Cuando el emisor detecta que el canal está ocupado, ejecuta un algoritmo de *back-off* similar al que emplea Ethernet: cuando vuelve a detectar que el canal está libre, espera un tiempo DIFS y después la estación calcula un tiempo adicional aleatorio de *back-off* y empieza a contarlo con un contador hacia atrás mientras el canal se encuentra libre. Cuando el temporizador del *back-off* aleatorio llega a cero, la estación transmite la trama. El intervalo de tiempo sobre el que el temporizador de *back-off* calcula el tiempo aleatorio se va doblando cada vez que una trama transmitida experimenta una colisión.

Una situación que se puede dar es el denominado problema del terminal oculto. Supongamos que la estación A transmite a la estación B y que la estación C también transmite a la estación B. Las obstrucciones físicas en el entorno (una montaña, por ejemplo) pueden provocar que A y C no se puedan comu-

nicar entre sí, ya que sus transmisiones sólo llegan a la estación B. Un segundo escenario es aquél en el que el receptor no puede detectar las colisiones por *fading* (pérdida) de la señal cuando se propaga por el medio sin hilos. La figura 63 muestra el caso en el que A y C están colocadas de tal manera que sus señales no son suficientemente potentes para que entre ellas se detecten transmisiones y sólo sirven para comunicarse con la estación que está en medio, la estación B.

Figura 63



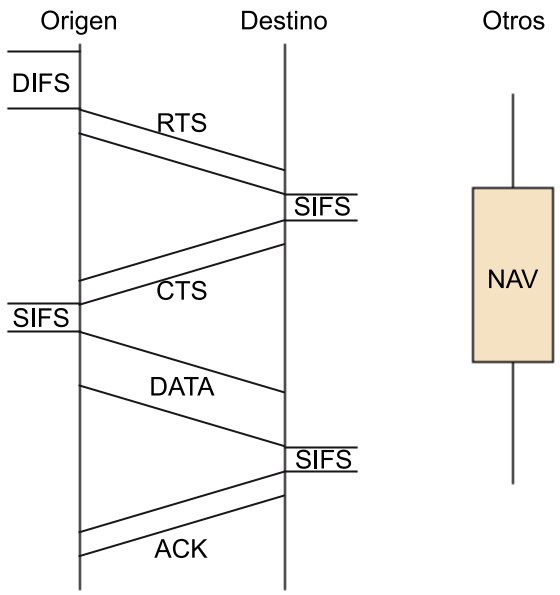
El protocolo IEEE 802.11 no implementa el mecanismo de detección de colisión (CD) como lo hace Ethernet (CSMA/CD). Esto se debe a que la capacidad de detectar colisiones requiere la capacidad de enviar y recibir al mismo tiempo. Debido a las dificultades para detectar las colisiones en un entorno inalámbrico, los ingenieros de IEEE 802.11 desarrollaron este acceso al medio con la idea de prevenir las colisiones, en lugar de detectar y recuperar las colisiones. En primer lugar, una trama IEEE 802.11 contiene un campo de duración en el que la estación emisora indica explícitamente la cantidad de tiempo

(98)NAV es la sigla de *network allocation vector*.

durante el que la trama se estará transmitiendo. Estos valores permiten a las otras estaciones determinar el tiempo mínimo (NAV[98]) que deben esperar para acceder al medio.

Asimismo, el protocolo puede utilizar una pequeña trama de control denominada RTS y otra que se conoce como CTS para reservar el acceso al canal. Cuando un emisor quiere enviar una trama, primero envía una RTS al receptor, indicándole la duración de la trama de datos en el paquete RTS. El receptor, cuando recibe una trama RTS, le responde con un paquete CTS, dando al emisor permiso explícito para empezar a transmitir. Todas las otras estaciones que escuchan las tramas RTS y CTS saben que deben esperar sus respectivas transmisiones para no interferir con esta transmisión. Un emisor y un receptor pueden operar de esta manera o sin utilizar las tramas RTS y CTS. El uso de RTS y CTS nos proporciona dos ayudas: la primera, dado que la trama CTS será escuchada por todas las estaciones dentro del radio de acción de la estación receptora, es la trama CTS, que ayuda a resolver el problema de los terminales ocultos. La segunda, dado que las tramas RTS y CTS son cortas, es que una colisión sólo se producirá durante la comunicación de RTS y CTS, es decir, durante la transmisión de RTS y de CTS (debemos señalar que cuando las tramas RTS y CTS son correctamente enviadas, no se produce ninguna colisión durante la transmisión de la trama de datos y de la trama de ACK).

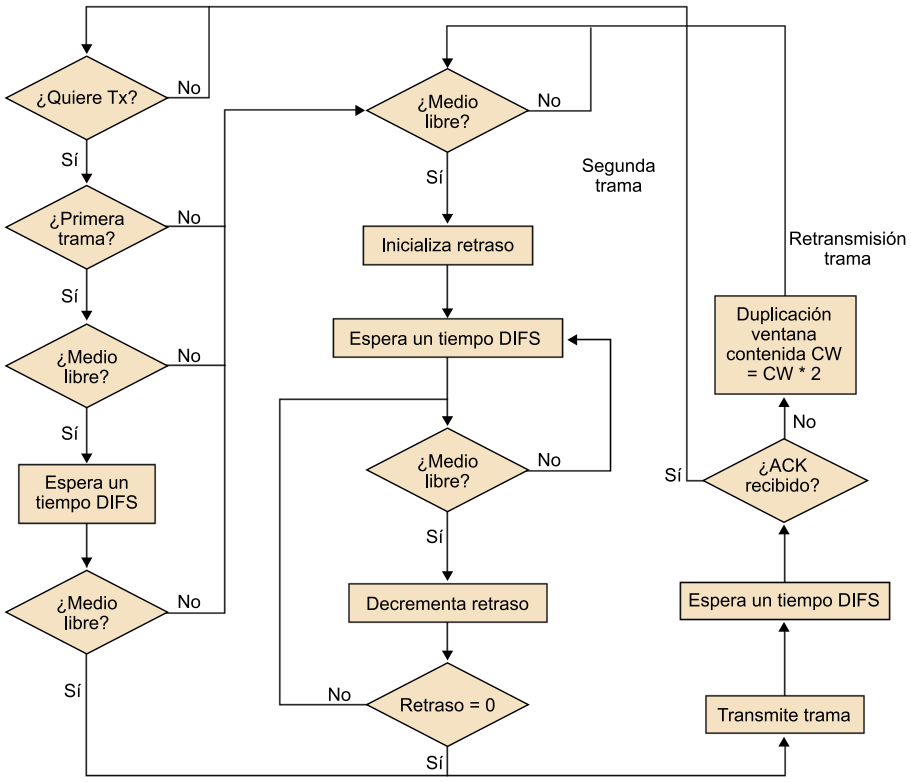
Figura 64



El estándar IEEE 802.11 también describe otras características, como la sincronización temporal, la gestión de la potencia, la unión y desunión de las estaciones en la red o los mecanismos de seguridad, encriptación, etc.

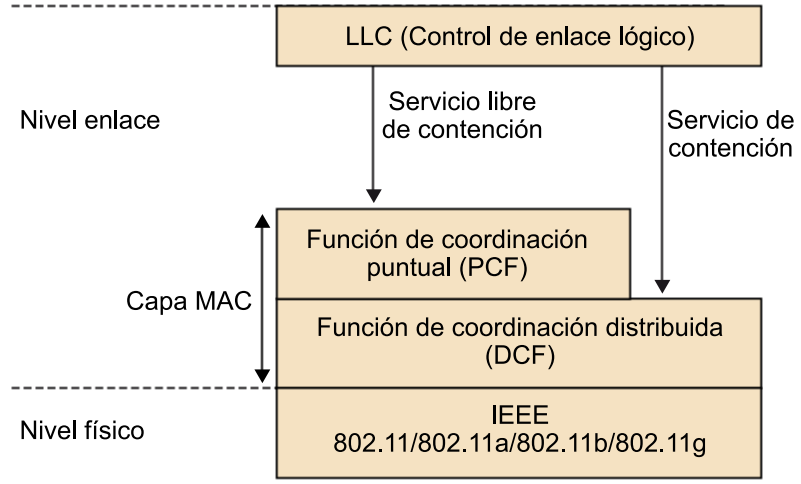
En resumen, el algoritmo de funcionamiento del CSMA/CA es el siguiente:

Figura 65



Definimos la función de coordinación como aquella que determina, dentro de un conjunto básico de servicios BSS, cuándo una estación puede transmitir o recibir unidades de datos de nivel MAC por un medio inalámbrico. La capa MAC se compone de dos funciones básicas: función de coordinación puntual (PCF) y función de coordinación distribuida (DFC). La gran mayoría de las tarjetas comerciales sólo implementan DCF y no el modo opcional PCF:

Figura 66



La DFC:

- Se puede utilizar tanto en modo de infraestructura como en modo *ad hoc*.

- Empleo de CSMA/CD con RTS/CTS, que se conoce como MACA.
- Reconocimiento de ACK, que provoca retransmisiones si no se recibe.
- Utilización del campo duración/ID, que contiene el tiempo de reserva para la transmisión y ACK.
- Implementación de fragmentación de datos.
- Priorización las tramas con el uso del espaciado entre tramas IFS.
- Soporta difusión y multidifusión sin ACK.

7.3.4. Tramas IEEE 802.11

Las tramas MAC contienen los siguientes componentes básicos:

- Una cabecera MAC, que contiene campos de control, duración, direccionamiento y control de secuencia.
- Un cuerpo de trama de longitud variable, que contiene información específica del tipo de trama.
- Una secuencia de suma de comprobación (FCS), que contiene un código de redundancia CRC de 32 bits.

Las tramas MAC se pueden clasificar en tres tipos:

1) Tramas de datos.

2) Tramas de control: ACK, RTC y CTS, y tramas libres de contienda.

3) Tramas de gestión: servicio de asociación, tramas de Beacon o portadora y tramas TIM o de tráfico pendiente en el punto de acceso.

El formato de una trama MC genérica tiene la siguiente estructura:

Figura 67

2 octetos	2 octetos	6 octetos	6 octetos	6 octetos	2 octetos	6 octetos	0-2.312 octetos	4 octetos
Control de trama	Duración /ID	Dirección 1	Dirección 2	Dirección 3	Control de secuencia	Dirección 4	Cuerpo de trama	FCS

Los campos que componen la trama son:

- **Duración**: tramas *power save* por dispositivos con limitaciones de potencia. Contienen el identificador o AID de estación. En el resto, se utiliza para indicar la duración del período que ha reservado una estación.

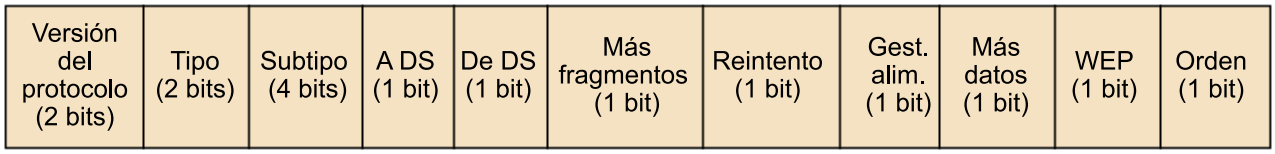
- **Address 1-4**: contiene las direcciones de 48 bits en las que se incluirán la estación emisora, la que recibe, la del punto de acceso origen y la del punto de acceso destino.

- **Cuerpo de la trama**: varía según el tipo de trama que se quiere enviar.

- **FCS**: contiene la suma de comprobación.

Los campos de control tienen la siguiente estructura:

Figura 68



- **Versión**.

- ***Type*/*subtype***: *type* indica si la trama es de datos, control o gestión; el campo *subtype* identifica los diferentes tipos de trama de cada uno de ellos.

- ***To DS*/*From DS***: identifica si la trama se envía o se recibe al/del sistema de distribución. En redes *ad hoc*, *To DS* y *From DS* están a cero. El caso más complejo contempla el envío entre dos estaciones mediante un sistema de distribución, y estos dos bits están a 1.

- **Más fragmentos**: se activa si se usa una fragmentación.

- ***Retry***: se activa si la trama es de retransmisión.

- ***Power management***: se activa si la estación utiliza el modo de economía de potencia.

- ***More data***: se activa si la estación tiene tramas pendientes en un punto de acceso.

- **WEP**: se activa si se utiliza el mecanismo de autoidentificación y encriptado.

- ***Order***: se utiliza con el servicio de ordenamiento estricto.

7.3.5. WiMax-IEEE 802.16

La tecnología WiMax[99] (interoperabilidad mundial para el acceso por microondas) supone una evolución con respecto a la Wi-Fi.

[99]WiMAX es la sigla de *worldwide interoperability microwave access*.

Está basada en los estándares 802.16 (WMAN), desarrollados por IEEE I para HiperMAN del ETSI. Permite la conectividad entre puntos fijos, nómadas y móviles, y eventualmente la conectividad móvil sin la necesidad de tener una línea punto a punto con una estación base.

La norma IEEE 802.16, publicada en diciembre del 2001, sirvió para fomentar la operatividad entre los sistemas LMDS[100]. En un principio, el rango de frecuencias se situaba entre 10 y 66 GHz, con necesidad de visión directa. A principios del 2003, con la aparición del 802.16a, se amplió el rango de frecuencias campo en las bandas de 2 a 11 GHz. En el año 2004, aparece el estándar 802.16-2004, también conocido como WiMAX.

[100]LMDS es la sigla de *local multipoint distribution system*.

El WiMAX Forum es una agrupación de más de 350 compañías. Se encarga de promover la interoperabilidad de dispositivos 802.16 y la unificación de los estándares a nivel mundial.

	802.16	802.16a	802.16e
Espectro	10-66 GHz	< 11 GHz	< 6 GHz
Funcionamiento	Solamente con visión directa	Sin visión directa (NLOS)	Sin visión directa (NLOS)
Ancho de banda	32-134 Mbps con canales de 28 MHz	Hasta 75 MHz con canales de 20 MHz	Hasta 15 Mbps con canales de 5 MHz
Modulación	QPSK, 16 QAM y 64 QAM	OFDM con 256 subportadoras QPSK, 16 QAM, 64 QAM	Lo mismo que 802.16a
Movilidad	Sistema fijo	Sistema fijo	Movilidad pedestre
Ancho del espectro	20, 25 y 28 MHz	Selección entre 1,25 y 20 MHz	El mismo que 802.16a con los canales de subida para ahorrar potencia
Distancia	2-5 km aproximadamente	5-50 km aproximadamente	2-5 km aproximadamente

Sus principales características se resumen a continuación:

- Modulación adaptativa: se eligen dinámicamente, en función de las condiciones del enlace. Si éste tiene un buen comportamiento (pocas pérdidas), se utiliza una modulación que lleva más bits y, por lo tanto, la velocidad aumenta.

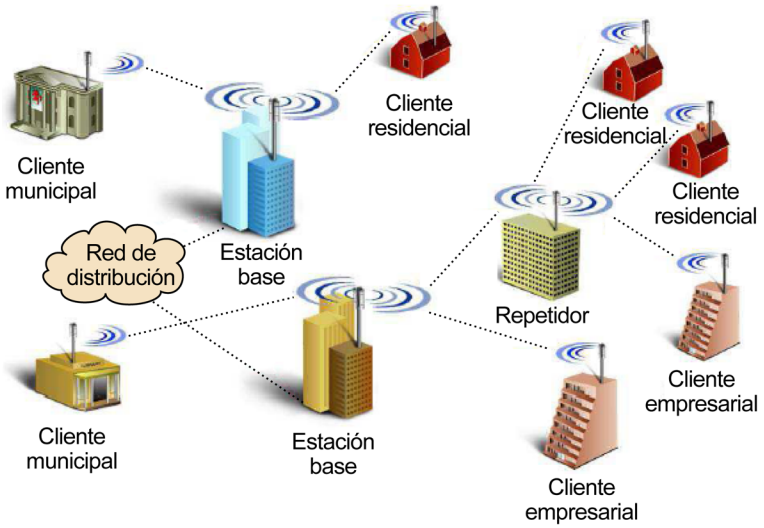
- Banda frecuencial: se puede trabajar con banda libre de 5,4 GHz, pero con poca potencia y con visión directa. También existe banda liberada de 3,5 GHz, en la que no es imprescindible la visión directa.

- Elementos: existen dos tipos de componentes, la estación base (unidades de acceso AVE) y las unidades de abonado (SU).

- Perfiles: permiten enlaces punto a punto (con visión directa) y punto multipunto (sin necesidad de visión directa).

- Permite calidad de servicio (QoS): gracias a que WiMAX está orientado a la conexión.

Figura 69. Topología red WiMAX



Pre-WiMAX

Cabe señalar la aparición del equipamiento denominado **pre-WiMAX**. Muchos fabricantes no esperaron a la aprobación definitiva del estándar 802.16 y sacaron al mercado (y todavía existen) equipos que implementaban un protocolo propietario basado en los desarrollos realizados por la tecnología WiMAX. Estos dispositivos, a pesar de proporcionar altas prestaciones, no permiten interoperabilidad con los otros fabricantes. Por el contrario, estos equipos trabajan en bandas de frecuencia libre (sin licencia), de manera que han acabado siendo una buena opción (y muy utilizada) para despliegues en este tipo de entornos.

Respecto a las velocidades, hay que diferenciar entre la velocidad de transmisión en el aire y la velocidad real (conocida como *throughput*). En el caso concreto de WiMAX y pre-WiMAX, la velocidad de los equipos es ligeramente diferente:

Velocidades pre-WiMAX/WiMAX		
Tecnología	**Velocidad máxima aire**	**Velocidad máxima real**
pre-WiMAX	54 Mbps	~30 Mbps
WiMAX	70 Mbps	~40 Mbps

Resumen

En este módulo didáctico se han abordado las características y funcionalidades principales del nivel de enlace, como la gestión de tramas, gestión del enlace, control de flujo y control de errores. Se trata de una serie de funciones que transforman un medio físico no perfecto y con errores en un medio que ofrece un servicio fiable a los protocolos de nivel de red. Hemos visto que normalmente estas funcionales son realizadas por unos dispositivos de enlace, denominados tarjetas de red.

También hemos presentado los diferentes contextos en los que se podía encontrar el nivel de enlace: comunicación punto a punto entre dos ordenadores locales, entorno a acceso a redes WAN, contexto de red de área local (LAN) y redes de transporte de área extensa (WAN).

Precisamente en el siguiente apartado se describe el contexto de red de área local asociado al nivel de enlace: los ámbitos en los que se instalan y las características que definen su funcionamiento, así como los medios de transmisión que emplean, las topologías y los protocolos de acceso al medio. Desde el punto de vista del medio de transmisión, hemos distinguido entre LAN cableadas, si el medio es guiado (cable o fibra óptica), y LAN sin hilo, cuando el medio es el aire.

Hemos visto que las topologías tienen un papel importante en el diseño y la instalación de una LAN: la estrella, el bus y el anillo son las más habituales. Últimamente, han aparecido los buses y los anillos en estrella, es decir, redes que presentan una topología física en estrella y se comportan como si fueran buses o anillos (la topología lógica).

Como las LAN son un medio compartido, se ha estudiado la necesidad de establecer protocolos de acceso para decidir qué estación puede transmitir tramas de información en cada momento. Son mecanismos flexibles, justos y fáciles de implementar. De los muchos que se han propuesto, CSMA/CD (en las redes Ethernet), el paso de testigo (en los anillos) y el CSMA/CA (en las redes sin hilo) son los más utilizados.

Finalmente, se han descrito los estándares aprobados que describen las tecnologías más utilizadas en las redes de área local:

- Ethernet IEEE 802.3, tecnología dominante en medios cableados.
- Wireless LAN IEEE 802.11, estándar en medios inalámbricos.
- Wimax IEEE 802.16, tecnología inalámbrica para redes MAN.

Bibliografía

Bertsekas, D.; Gallager, R. (1992). *Data networks* (2.ª ed.). Englewood Cliffs: Prentice Hall.

Halsall, F. (1998). *Comunicaciones de datos, redes de computadoras y sistemas abiertos* (4.ª ed.). Addison-Wesley.

Kurose, James F.; Ross, Keith W. (2005). *Computer networking: a top-down approach featuring the Internet*. Addison-Wesley.

Stallings, William (2000). *Comunicaciones de datos y redes de computadores 6*. Prentice-Hall.

Tanenbaum, Andrew S. (2003). *Redes de computadores* (4.ª ed.). Pearson.