



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Escuela Técnica Superior de Ingenieros  
de Telecomunicación

PRÁCTICA 3  
**Estudio de acceso al medio con  
CSMA/CD**

*Redes de Ordenadores*

*5 de diciembre de 2005*

**Himar Alonso Díaz**

# Índice

<b>1. Introducción</b>	<b>4</b>
1.1. ¿Qué es CSMA/CD?	4
1.2. Parámetros	5
1.2.1. Parámetros de estación	5
1.2.2. Parámetros de la red	5
1.3. Colisiones	6
1.4. Software utilizado	6
<b>2. Simulaciones manuales</b>	<b>8</b>
2.1. Ejemplo básico	8
2.2. Prioridad fija	9
2.3. Velocidad de simulación	10
<b>3. Simulaciones automáticas</b>	<b>11</b>
3.1. Ejemplo básico	11
3.2. Prioridad fija	12
3.3. Tiempo de simulación y carga en la red	13
<b>4. Conclusiones</b>	<b>14</b>

# Contenido

# 1. Introducción

## 1.1. ¿Qué es CSMA/CD?

El CSMA/CD – *Carrier Sense Multiple Access with Collision Detection* (en español, Acceso Múltiple con Escucha de Portadora y Detección de Colisiones) es un *protocolo* de acceso a un medio físico compartido. Se utiliza en redes *Ethernet* principalmente, y su objetivo es mejorar las prestaciones de este tipo de redes. Anteriormente a esta técnica se usaron las de *Aloha* y *Aloha ranurado*, pero ambas presentaban muy bajas prestaciones. Por eso apareció primeramente la técnica CSMA, que fue posteriormente mejorada con la aparición de CSMA/CD [1].

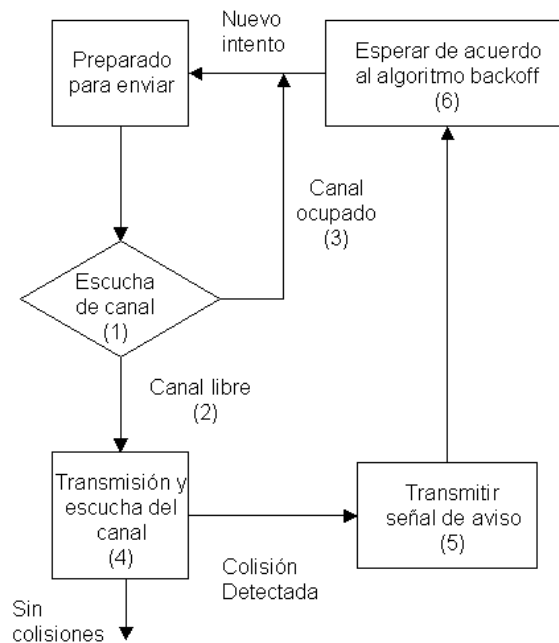


Figura 1: Diagrama de flujo del algoritmo CSMA/CD [2]

A continuación se describe el funcionamiento del algoritmo CSMA/CD, mediante un símil humano: [3]

- Supóngase una conversación alrededor de una mesa en un cuarto oscuro.
- Antes de hablar, cualquier participante debe escuchar durante unos segundos para comprobar que nadie está hablando.
- Cuando esto ocurre –nadie habla–, cualquiera tiene oportunidad de hablar.
- Si dos personas comienzan a hablar en el mismo momento, se darán cuenta y dejarán de hablar.

Traducido al mundo *Ethernet*, podemos entender el funcionamiento de este protocolo, siguiendo el diagrama de bloques de la Figura 1.

## 1.2. Parámetros

A lo largo de todas las simulaciones analizaremos principalmente los siguientes tipos de parámetros. Es por ello que dedico este apartado para hacer un breve resumen del significado de cada uno de ellos. [4]

### 1.2.1. Parámetros de estación

**Prioridad** Cuando se produce una colisión (esto se estudiará en el siguiente apartado), las estaciones involucradas esperan un tiempo determinado (*backoff*) antes de intentar transmitir de nuevo. Si en el modo de simulación, se elige *con prioridad*, el calculo de este *backoff* se efectuará en orden a las prioridades establecidas, de forma que la estación con una prioridad numéricamente más baja intentará antes la re-transmisión. Las prioridades están comprendidas entre 1 y 10, y son excluyentes, de forma que dos estaciones no podrán tener la misma prioridad. Este es un parámetro válido en simulación manual y automática.

**Carga** Es la cantidad de tráfico que asignamos a cada estación. Si a una estación le asignamos una carga del 30 %, esta estación intentará estar transmitiendo durante un 30 % del tiempo programado para la simulación, es decir, ocupará el bus durante dicho porcentaje de tiempo. Este es un parámetro válido en simulación automática.

**Tamaño de trama** Indica el tiempo aproximado que durará cada transmisión de un equipo. El programa facilita dos tamaños de trama, media y larga, con el objetivo de dar al usuario mayores posibilidades de simulación. Este es un parámetro válido en simulación manual y automática.

### 1.2.2. Parámetros de la red

**Velocidad de la simulación** El programa proporciona tres velocidades distintas de ejecución. Se trata de la velocidad con la que sucederán los distintos eventos que conforman la simulación, excepto los tiempos de retardo. Este es un parámetro válido en simulación automática.

**Tamaño de la ventana de colisión** La ventana de colisión es el intervalo de tiempo que va desde que una estación inicia una transmisión hasta que esta es detectada por todos los equipos de la red. Es por tanto el intervalo de tiempo para cada transmisión en el que pueden ocurrir colisiones. Este es un parámetro válido en simulación manual y automática.

**Tiempo de simulación** Es la duración en segundos que se asigna a una simulación automática. Este tiempo no se cumplirá siempre, ya que lo aleatorio de la simulación puede provocar que cuando este tiempo se agote todavía queden estaciones resolviendo una situación de colisión, en cuyo caso se muestra un mensaje indicando que el tiempo de simulación ha terminado, pero que la simulación continúa hasta que se hayan enviado los *frames* pendientes. Este es un parámetro válido en simulación automática.

### 1.3. Colisiones

Según el protocolo descrito, antes de iniciar una transmisión, los equipos escucharán el canal para comprobar que no esté siendo utilizado. Aun tomando esta precaución puede darse el caso de que dos o más estaciones comiencen a transmitir una trama al mismo tiempo. Entonces decimos que se ha producido una *colisión*. Cuando esto sucede, la estación que haya detectado la colisión enviará un mensaje de *jam* (de 32 bits) para notificárselo a las demás estaciones.

Cuando todos los equipos de la red tienen conocimiento de ello, automáticamente se paran todas las transmisiones y se ejecuta un algoritmo especial, denominado *Truncated binary exponential backoff*, que consiste en esperar un intervalo de tiempo aleatorio (llamado *backoff*), múltiplo de 512 tiempos de bit antes de volver a intentar la transmisión. De esta forma se evita (aunque no siempre) que todos vuelvan a acceder en el mismo instante, provocando una colisión permanente.

Este algoritmo funciona de la siguiente manera:

- Se estima un tiempo de espera:  $Backoff\ Time = Número\ Aleatorio \times Slot\ Time$  donde:

**Slot Time** es el tiempo para propagar 512 bits (51, 2 $\mu$ s en 10Mbps ó 5, 12 $\mu$ s en 100 Mbps. *Gigabit Ethernet* utiliza un Slot time de 4096 bits –512 bytes–).

**Número aleatorio** es un número entero mayor o igual a cero y menor que  $2^n$ .

$n$  es el número de intentos de retransmisión para las primeras 10 veces ó  $n = 10$  para los intentos número 11 . . . 16.

- Después de 16 intentos –fallidos– el algoritmo notificará un error a las capas superiores. [3]

### 1.4. Software utilizado

Para el estudio de este protocolo utilizaremos un simulador virtual<sup>1</sup> que nos permitirá analizar qué ocurre cuando tenemos varios ordenadores conectados a una misma red y dos o más de ellos intentan transmitir al mismo tiempo. Veremos cómo se detectan *colisiones* de datos y cómo se organizan los equipos para evitar futuras colisiones, mediante la aplicación del protocolo CSMA/CD.

El programa nos proporciona una *configuración típica* de red, esto es, una disposición de equipos preestablecida (véase la Figura 2) que nos servirá para introducir su funcionamiento. Cada equipo estará caracterizado, como ya se ha dicho, por una *dirección*, una *prioridad* y un porcentaje de *carga* en la red. La figura asociada a cada terminal es indiferente para nuestro estudio, de manera que para nosotros no habrá diferencia entre un servidor, una estación, una impresora. . .

En cada equipo disponemos de un monitor de red sencillo, con tres luces que nos indicará cuándo el equipo asociado está transmitiendo información (TX), recibiendo información (RX) y cuándo ha detectado una colisión de datos (CO).

---

<sup>1</sup>Software desarrollado por Jaime Picó Rueda.

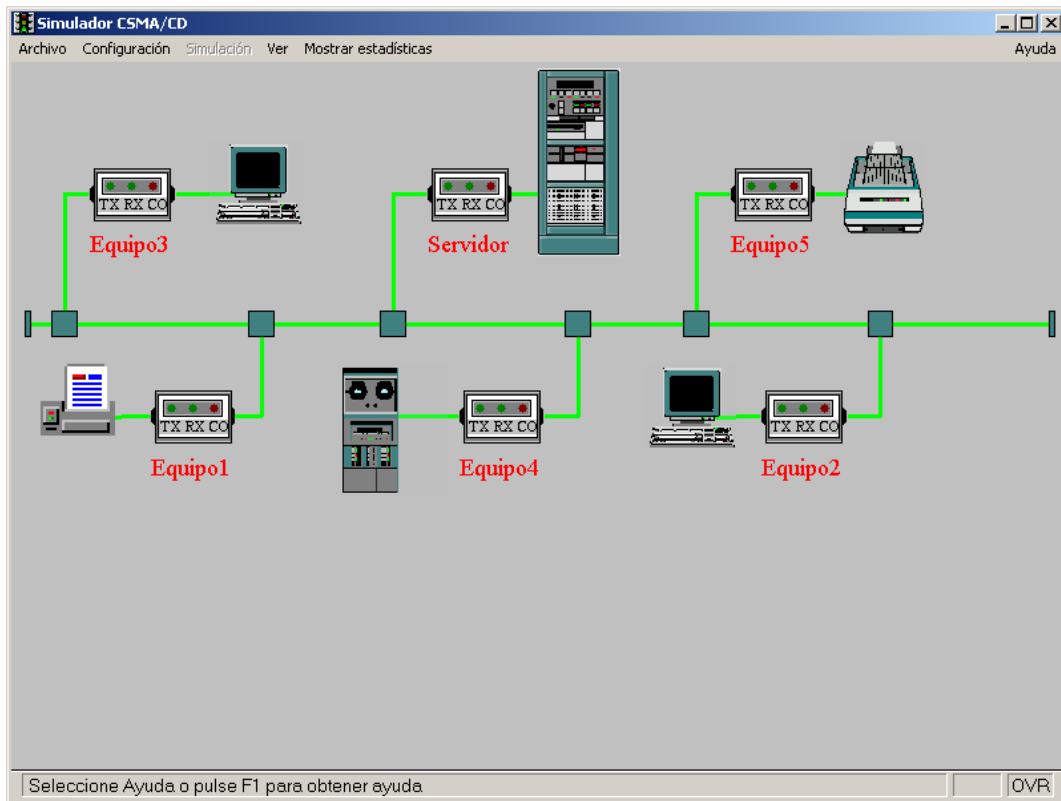


Figura 2: Ventana con la configuración típica del programa

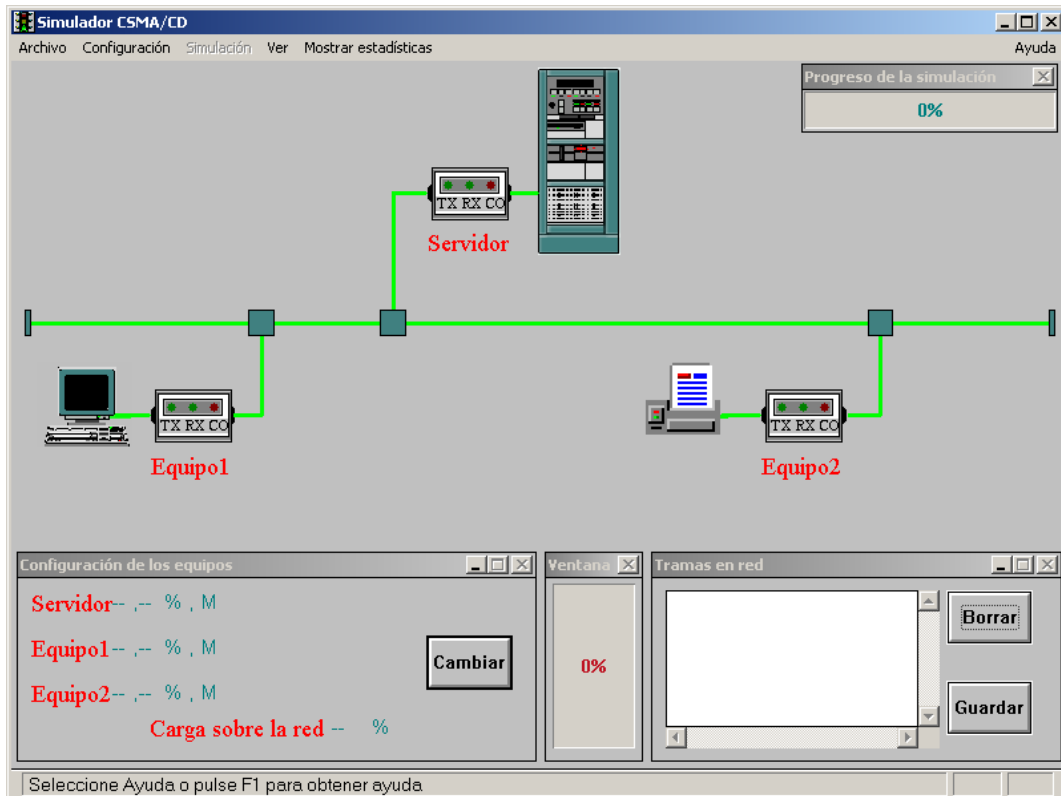
## 2. Simulaciones manuales

### 2.1. Ejemplo básico

PARÁMETROS:

Modelo de simulación: Manual	Velocidad: Media
Número de estaciones: 3	Prioridad: Aleatoria
Tamaño de la ventana de colisión: Media	

CONFIGURACIÓN UTILIZADA:



COMENTARIOS Y ESTADÍSTICAS:

En este primer ejemplo, haremos un *broadcast* con el Servidor, y al mismo tiempo intentaremos transmitir desde el Equipo1 hasta el Equipo2. En el siguiente *log* vemos cómo ambos equipos intentan transmitir al mismo tiempo y se producen varias colisiones, finalmente el Servidor espera –aleatoriamente– un *backoff* que permite al Equipo1 enviar su trama.

```
Mon Dec 05 13:46:44 2005
Servidor--COLISION
Mon Dec 05 13:46:45 2005
Equipo1--COLISION
Mon Dec 05 13:46:51 2005
Servidor--COLISION
Mon Dec 05 13:46:51 2005
Equipo1--COLISION
Mon Dec 05 13:46:56 2005
Equipo2-Equipo1
Mon Dec 05 13:47:14 2005
Broadcast-Servidor
```

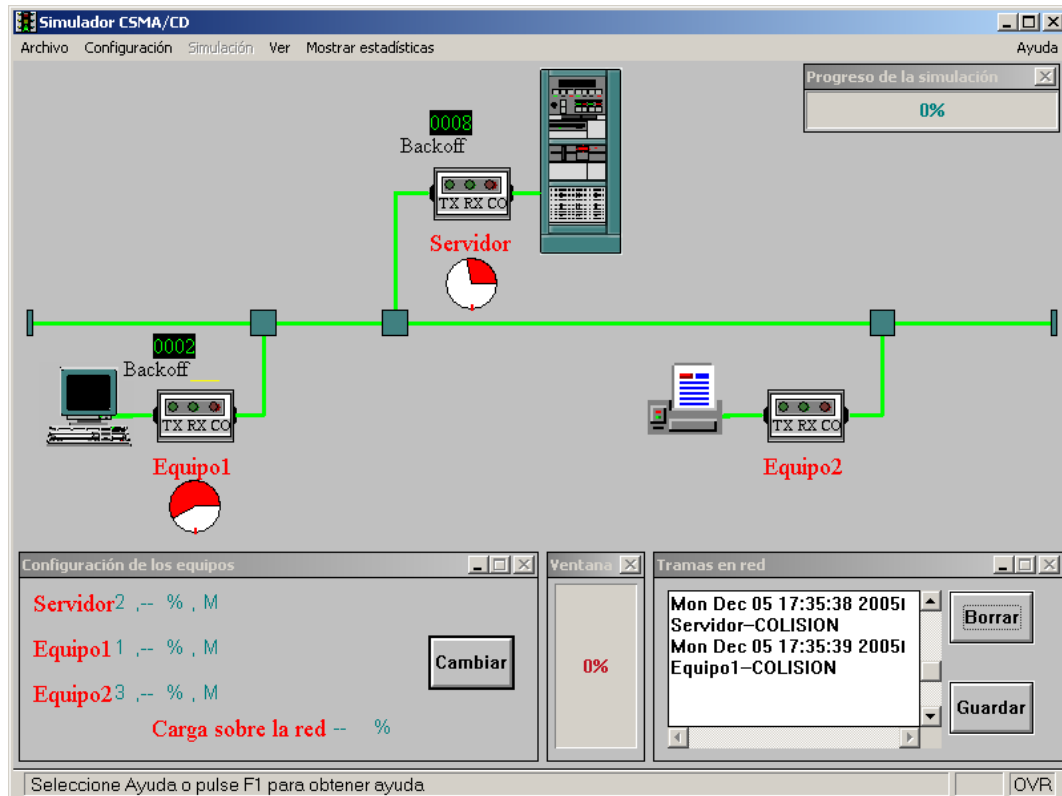
## 2.2. Prioridad fija

PARÁMETROS:

Modelo de simulación: Manual  
Número de estaciones: 3  
Tamaño de la ventana de colisión: Grande

Velocidad: Media  
Prioridad: Fija

CONFIGURACIÓN UTILIZADA:



COMENTARIOS Y ESTADÍSTICAS:

En esta simulación, haremos también dos transmisiones simultáneas y veremos qué ocurre con las colisiones. Al tener una prioridad fija, y como es mayor la prioridad del Equipo1 que la del Servidor (véase en la captura), cuando sucede una colisión, el Servidor tendrá que esperar más tiempo que Equipo1 para volver a transmitir. Si Equipo2 también hubiera producido una colisión al mismo tiempo, el intervalo de *backoff* que tendría que esperar sería aún mayor. Además hemos seleccionado una ventana de colisión *Grande*. Por tanto el canal estará ocupado más tiempo y el tiempo de transmisión es mayor.

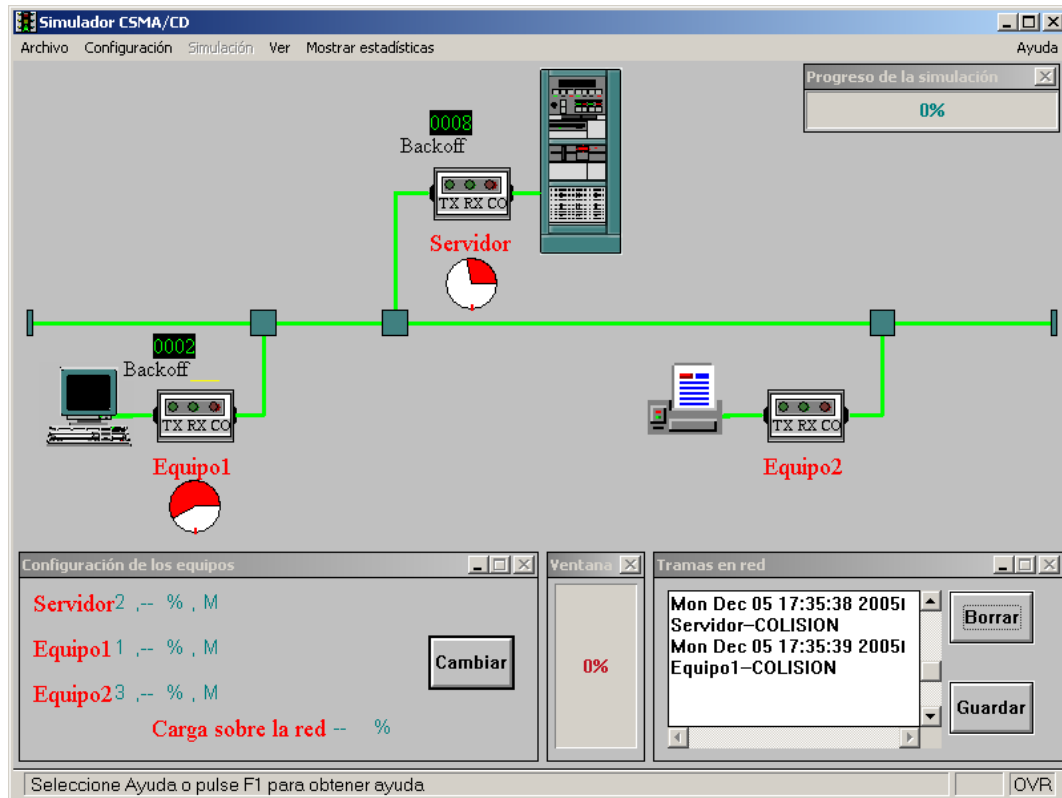
```
Mon Dec 05 17:32:04 2005
Equipo1--COLISION
Mon Dec 05 17:32:06 2005
Servidor--COLISION
Mon Dec 05 17:32:19 2005
Servidor-Equipo1
Mon Dec 05 17:32:41 2005
Equipo2-Servidor
Mon Dec 05 17:35:38 2005
Servidor--COLISION
Mon Dec 05 17:35:39 2005
Equipo1--COLISION
Mon Dec 05 17:35:53 2005
Servidor-Equipo1
Mon Dec 05 17:36:15 2005
Equipo2-Servidor
```

## 2.3. Velocidad de simulación

PARÁMETROS:

Modelo de simulación: Manual	Velocidad: Alta
Número de estaciones: 3	Prioridad: Aleatoria
Tamaño de la ventana de colisión: Media	

CONFIGURACIÓN UTILIZADA:



COMENTARIOS Y ESTADÍSTICAS:

Esta simulación es muy similar a la primera que hicimos. Hemos aumentado la velocidad de simulación. Lógicamente este parámetro no influye en el funcionamiento de la red. Lo que ocurre es que las transmisiones han sido más rápidas. La primera tardó 30s y ésta 22s. Modificar la velocidad de simulación (sea ésta manual o automática) nos permite comparar redes que funcionen a distintas velocidades (por ejemplo *Ethernet* a 10 Mbps y a 100 Mbps)

```
Mon Dec 05 18:07:47 2005
Equipo2--COLISION
Mon Dec 05 18:07:48 2005
Servidor--COLISION
Mon Dec 05 18:07:50 2005
Equipo2--COLISION
Mon Dec 05 18:07:51 2005
Servidor--COLISION
Mon Dec 05 18:07:58 2005
Equipo2--COLISION
Mon Dec 05 18:07:58 2005
Servidor--COLISION
Mon Dec 05 18:08:00 2005
Broadcast-Equipo2
Mon Dec 05 18:08:09 2005
Broadcast-Servidor
```

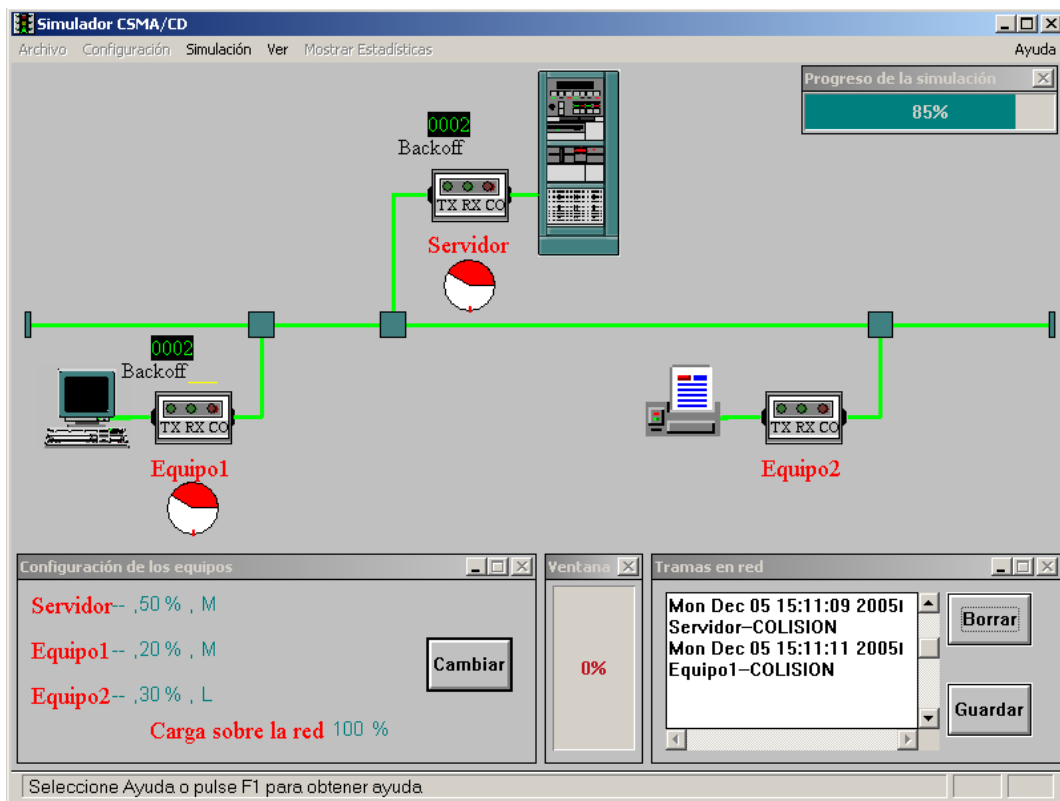
### 3. Simulaciones automáticas

#### 3.1. Ejemplo básico

PARÁMETROS:

Modelo de simulación: Automática	Tiempo de simulación: 60s
Número de estaciones: 3	Velocidad: Media
Tamaño de la ventana de colisión: Media	Prioridad: Aleatoria

CONFIGURACIÓN UTILIZADA:



ESTADÍSTICAS:

- Por equipos:

Dirección	TX	RX	CO	T. Espera	Máx. Espera	Carga efectiva
Servidor	2	1	2	24s	18s	63 %
Equipo1	0	1	2	6s	5s	13 %
Equipo2	1	0	0	12s	12s	30 %

- Globales:

Carga de la red: 100 %	Transmisiones: 3	Recepciones: 2	Colisiones: 2
------------------------	------------------	----------------	---------------

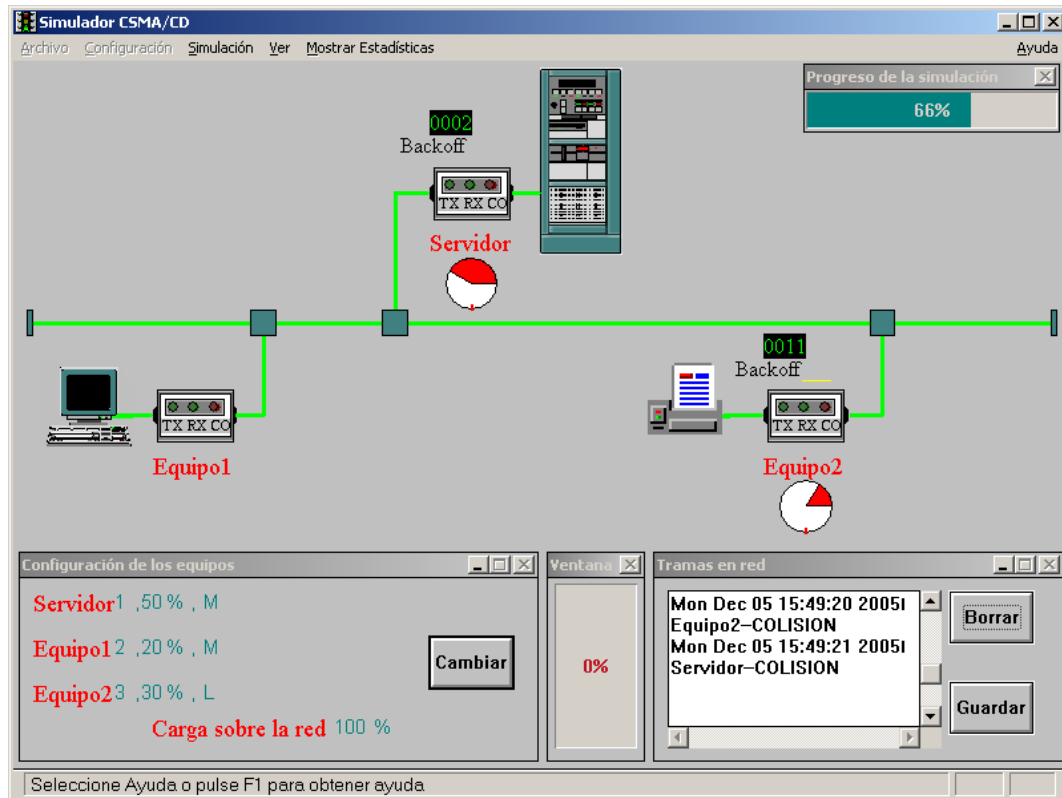
COMENTARIOS: En esta primera simulación automática hemos puesto a funcionar la red con parámetros estándares y hemos observado lo que sucede. Hemos hecho la captura de pantalla mientras se ejecutaba la simulación. En este instante se observa que el Servidor y el Equipo1 han detectado una colisión y están ejecutando el algoritmo *truncated binary exponential backoff*. El reloj indica el tiempo de *backoff* (en segundos). Aunque no puede apreciarse en la captura, un dato interesante ha sido que la trama enviada por Equipo2, al ser más grande, el tiempo de transmisión ha sido mayor.

### 3.2. Prioridad fija

PARÁMETROS:

Modelo de simulación: Automática	Tiempo de simulación: 60s
Número de estaciones: 3	Velocidad: Media
Tamaño de la ventana de colisión: Media	Prioridad: Fija

CONFIGURACIÓN UTILIZADA:



ESTADÍSTICAS:

- Por equipos:

Dirección	TX	RX	CO	T. Espera	Máx. Espera	Carga efectiva
Servidor	1	1	3	27s	17s	45 %
Equipo1	1	0	1	10,5s	10,5s	26,7 %
Equipo2	0	1	2	40s	23s	13,3 %

- Globales:

Carga de la red: 100 %	Transmisiones: 2	Recepciones: 3	Colisiones: 3
------------------------	------------------	----------------	---------------

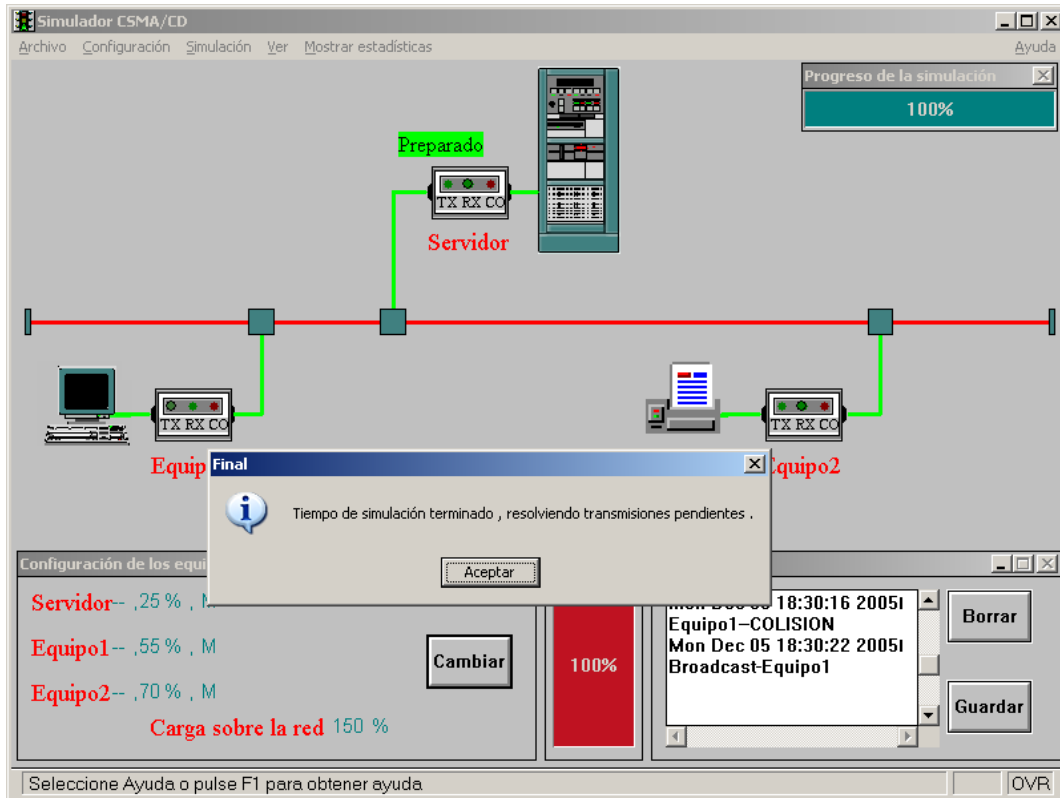
COMENTARIOS: Cuando asignamos prioridades fijas, el algoritmo de *backoff* hace que los equipos que tengan mayor prioridad esperen menos. En este caso el Servidor es el equipo con mayor prioridad, luego el Equipo1 y finalmente el Equipo2. Por eso, como se puede ver en esta captura, ante una colisión de señales entre el Servidor y Equipo2, el cronómetro del Servidor avanza más rápidamente.

### 3.3. Tiempo de simulación y carga en la red

PARÁMETROS:

Modelo de simulación: Automática	Tiempo de simulación: 50s
Número de estaciones: 3	Velocidad: Media
Tamaño de la ventana de colisión: Media	Prioridad: Fija

CONFIGURACIÓN UTILIZADA:



ESTADÍSTICAS:

- Por equipos:

Dirección	TX	RX	CO	T. Espera	Máx. Espera	Carga efectiva
Servidor	0	1	0	1s	1s	0 %
Equipo1	2	1	1	12s	9s	63 %
Equipo2	1	1	1	24s	14s	31 %

- Globales:

<b>Carga de la red: 150 %</b>	<b>Transmisiones: 3</b>	<b>Recepciones: 3</b>	<b>Colisiones: 1</b>
-------------------------------	-------------------------	-----------------------	----------------------

COMENTARIOS: Hemos hecho estos cambios para forzar la aparición de este cuadro de aviso. Efectivamente hemos disminuido el tiempo de simulación, pero al mismo tiempo hemos aumentado la carga de los Equipos 1 y 2. Por tanto éstos intentarán transmitir datos con mayor frecuencia. Esto ha provocado que el servidor no haya transmitido ninguna trama y que tiempo de simulación acabara, pero aún quedaran pendientes por solucionar algunas transmisiones. Esto llevó a tener que esperar un total de 93s (43s más de lo previsto).

## 4. Conclusiones

A la vista de los datos que aportan las simulaciones podemos resumir la influencia de cada uno de los parámetros:

**Prioridad** Entra en juego cuando se producen colisiones. Si la prioridad es aleatoria, se ejecuta un algoritmo que obligará a la máquina a esperar un tiempo de *backoff* aleatorio. Si hay una prioridad establecida, esperarán más las máquinas con menor prioridad.

**Carga** Nos indicará con qué frecuencia accede una estación al medio para transmitir una trama. Un porcentaje de carga muy elevado provocará congestión en la red y mayor número de colisiones.

**Tamaño de trama** De forma análoga a la carga, una trama muy grande congestionará el canal, pero no por las colisiones, sino porque necesitará mayor tiempo para transmitirse.

**Velocidad de simulación** No altera el funcionamiento de la red, sólo la velocidad con la que se realizan las transmisiones. Como dije anteriormente, puede ser útil para comparar dos redes que funcionen a velocidades distintas.

**Tamaño de la ventana de colisión** También tendrá que ver con la congestión del canal, con las colisiones, y con el tiempo necesario para realizar una transmisión. Una ventana de colisión muy grande se traduce en el medio físico como una red muy amplia, tanto como para que la señal tarde un tiempo considerable en propagarse. Durante ese tiempo, los demás equipos deben evitar realizar transmisiones porque podría producirse una colisión.

**Tiempo de simulación** Este parámetro sólo ha tenido relevancia en las simulaciones automáticas. Un caso real podría considerarse como *el tiempo máximo permitido para realizar una transmisión*. Si se superase este tiempo, no sería posible enviar las tramas que se desean transmitir.

## Referencias

- [1] Wikipedia (Enciclopedia Libre):  
<http://es.wikipedia.org/wiki/CSMA/CD>
- [2] Web ETS Informática Aplicada - Universidad Politécnica de Valencia:  
<http://www.eui.upv.es/ineit-mucon/Applets/TutorialEthernet/gestion.htm>
- [3] Arcesio.net  
<http://www.arcesio.net>
- [4] Manual del simulador  
Autor: *Jaime Picó Rueda*.