

# Acceso al medio

Area de Ingeniería Telemática  
<http://www.tlm.unavarra.es>

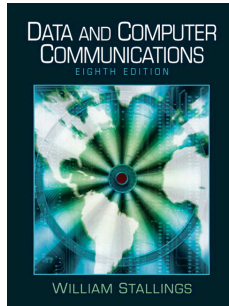
Arquitectura de Redes, Sistemas y Servicios  
3º Ingeniería de Telecomunicación

# Temario

- Introducción
- Arquitecturas, protocolos y estándares
- Conmutación de paquetes
- Conmutación de circuitos
- Tecnologías
- Control de acceso al medio en redes de área local
- Servicios de Internet

# Temario

- Introducción
- Arquitecturas, protocolos y estándares
- Conmutación de paquetes
- Conmutación de circuitos
- Tecnologías
- **Control de acceso al medio en redes de área local**
- Servicios de Internet

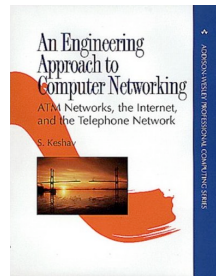


# Material

Capítulo 16 de

**W. Stallings,**

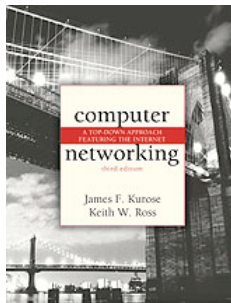
**Data and Computer Communications**



Capítulo 7 de

**S. Keshav**

**An Engineering Approach to  
Computer Networks**



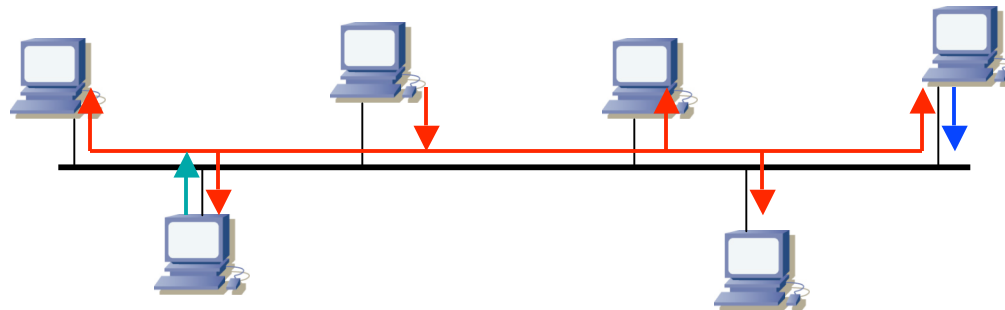
Capitulo 5 de

**J.F. Kurose & K.W. Ross**

**Computer Networking. A top-down  
approach featuring the Internet**

# Acceso al medio

- Red basada en un medio compartido (de broadcast)
  - Todos oyen lo que envío
  - Para enviar una trama a un nodo concreto indico su dirección (nivel de enlace/ethernet)



- Pero... ¿y si hay varios intentando enviar a la vez?
- Problema del **Acceso al Medio**  
Mecanismos de esperas, colisiones, turnos... (ya se ha comentado que el de Ethernet se llama CSMA/CD)... pero empecemos por el principio...

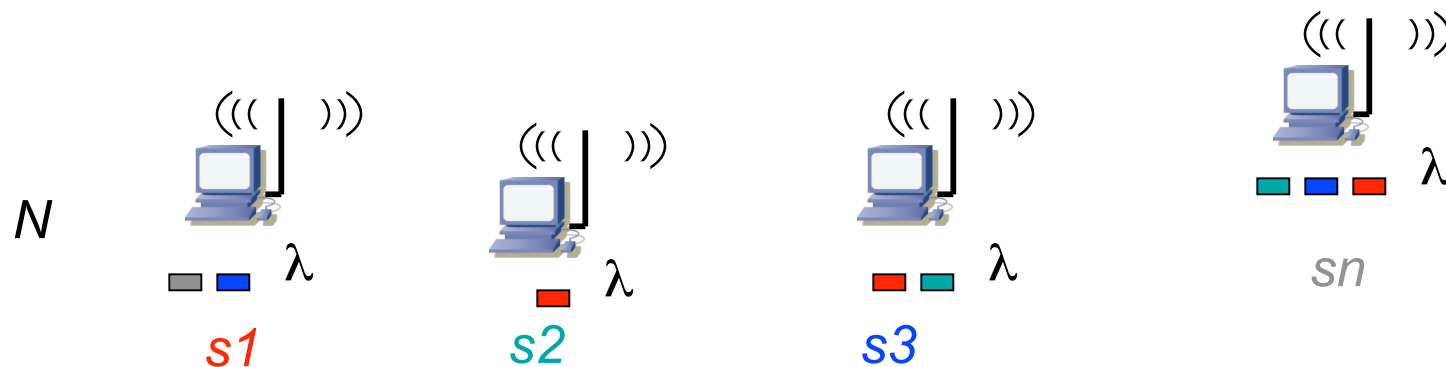
# Problema más simple

- Se planteo originalmente en redes por radio
- $N$  estaciones que pueden enviar y recibir tramas por radio
- Las estaciones tienen mensajes para otras estaciones

Nivel de enlace

Generan paquetes a enviar de forma aleatoria... caracterizadas por una variable aleatoria y con tasa media de  $\lambda$  tramas por unidad de tiempo

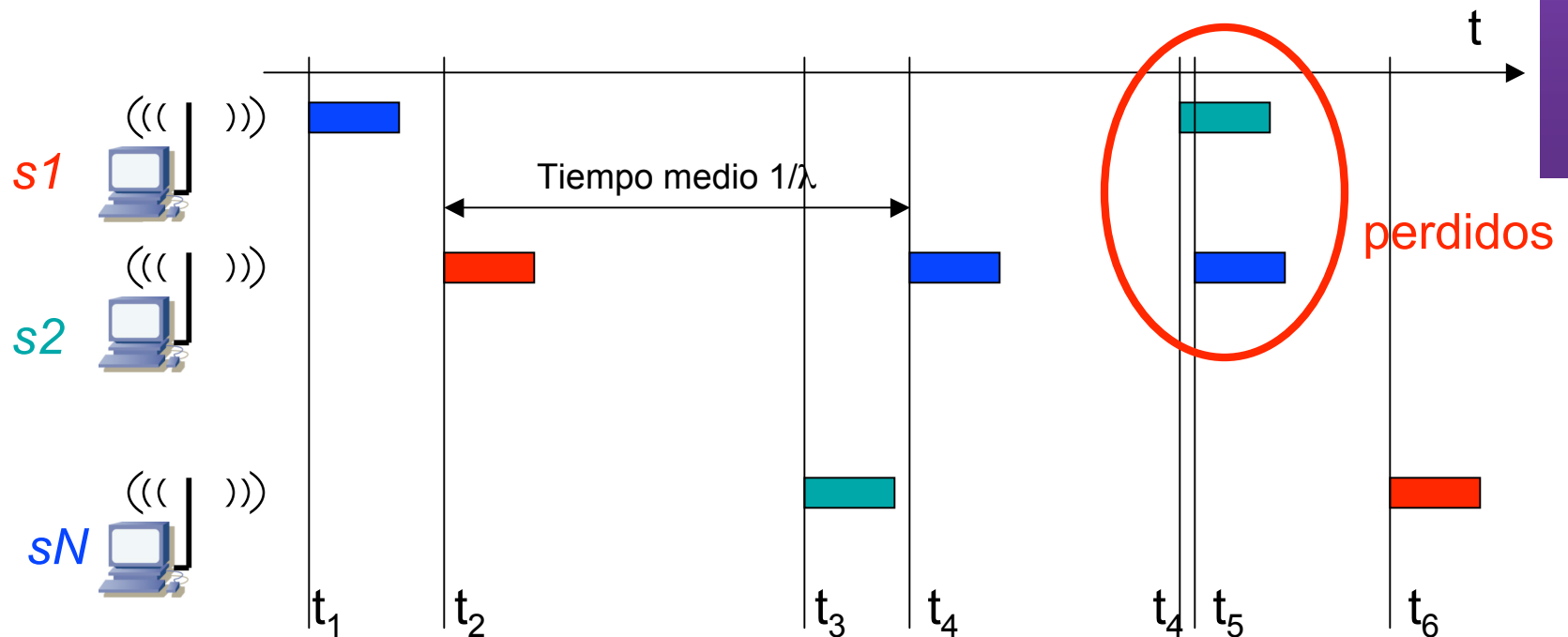
- ¿Cuántas tramas podemos conseguir que lleguen a su destino?



# Primera aproximación

- Si tengo un paquete para transmitir... lo envío
  - Si tengo suerte llegara
  - Si otro transmite a la vez no llegara ninguno de los dos
  - Si en media el tiempo que tarda en llegar el siguiente paquete a alguien es menor que el tiempo que tardo en transmitirlo bastantes llegaran a sus destinos

Supongamos que el tiempo de propagación es muy pequeño...



# ALOHA

- Desarrollado para redes de paquetes por radio
- En la universidad de Hawaii
- Cuando la estación tiene una trama para transmitir... la envía
  - La trama incluye la identificación del destino que debe recibirla
  - La trama incluye un código de detección de errores (Frame Check Sequence)
- Si recibo una trama correctamente (FCS=ok) envío una trama de confirmación (ACK) al emisor
  - La trama de ACK es una trama pequeña (poco más que el destino y una indicación de que es un ACK)
- Después escucha durante un tiempo (un poco más que el máximo RTT)
  - Si recibe un ACK la considera transmitida (y pasa a transmitir la siguiente que le llegue)
  - Si no recibe un ACK vuelve a enviar la misma
  - Si lleva  $n$  intentos de retransmisión sin recibir ACK la da por perdida (y pasa a transmitir la siguiente que le llegue)
- La trama se puede corromper por ruido o por mezclarse con otra trama enviada por otra estación (colisión)
- Cualquier solapamiento de dos tramas causa una colisión



# Prestaciones ALOHA

- $N$  estaciones
- Mensajes de tamaño fijo  $s$  bytes
- Capacidad del canal  $C$  bytes/segundo
- Cada estación genera mensajes siguiendo una proceso de Poisson con parámetro  $\lambda$  (en media  $\lambda$  mensajes por segundo)

- Los mensajes ocupan el canal un tiempo fijo  $m=s/C$
- La carga ofrecida (intensidad de tráfico) al medio compartido será

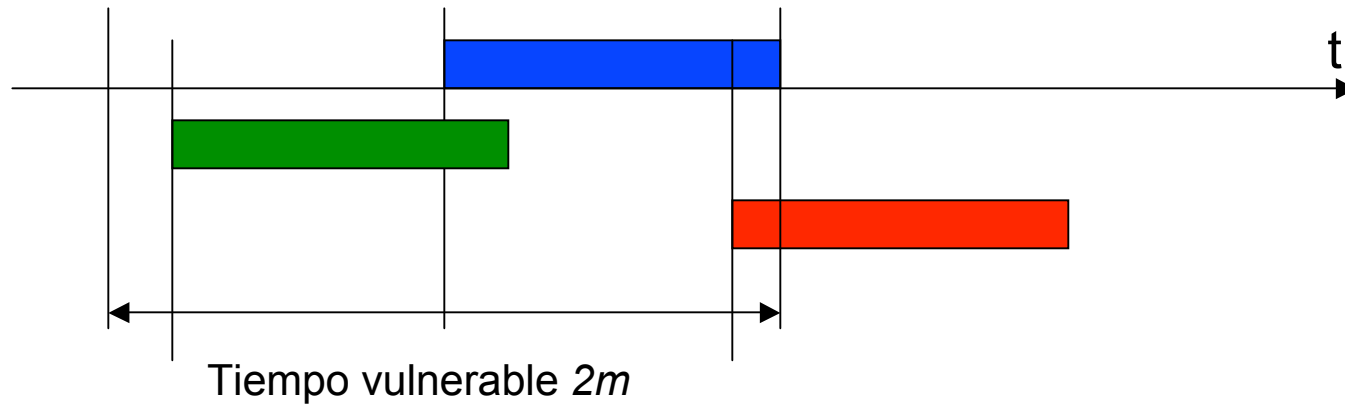
Una estación:  $\rho_i = m \lambda$

Todas las estaciones:  $\rho = N m \lambda$

- Ej: Mensajes de  $200B$  enviados a  $1Mbps$   $m=1.6ms$   
 $N=20$   $\lambda=5$  mensajes/s  $\rho = 0.16$  16% del tiempo ocupado  
 $N=20$   $\lambda=20$  mensajes/s  $\rho = 0.64$  64% ¿muchas perdidas?  
 $N=40$   $\lambda=20$  mensajes/s  $\rho = 1.28$  128% No caben

# Prestaciones ALOHA

- ¿Cual es la probabilidad de que un mensaje llegue al otro extremo?  
 $P[\text{llegar}] = P[\text{no haya otro paquete "cerca"}]$



- Un paquete desde  $m$  segundos antes hasta  $m$  segundos despues nos estropea el envío
- Numero de paquetes que llegan en  $\Delta t = 2m$  es un v.a. de Poisson con parámetro  $2mN\lambda$

$$P_{2mN\lambda}[k] = \frac{(2mN\lambda)^k}{k!} e^{-2mN\lambda}$$

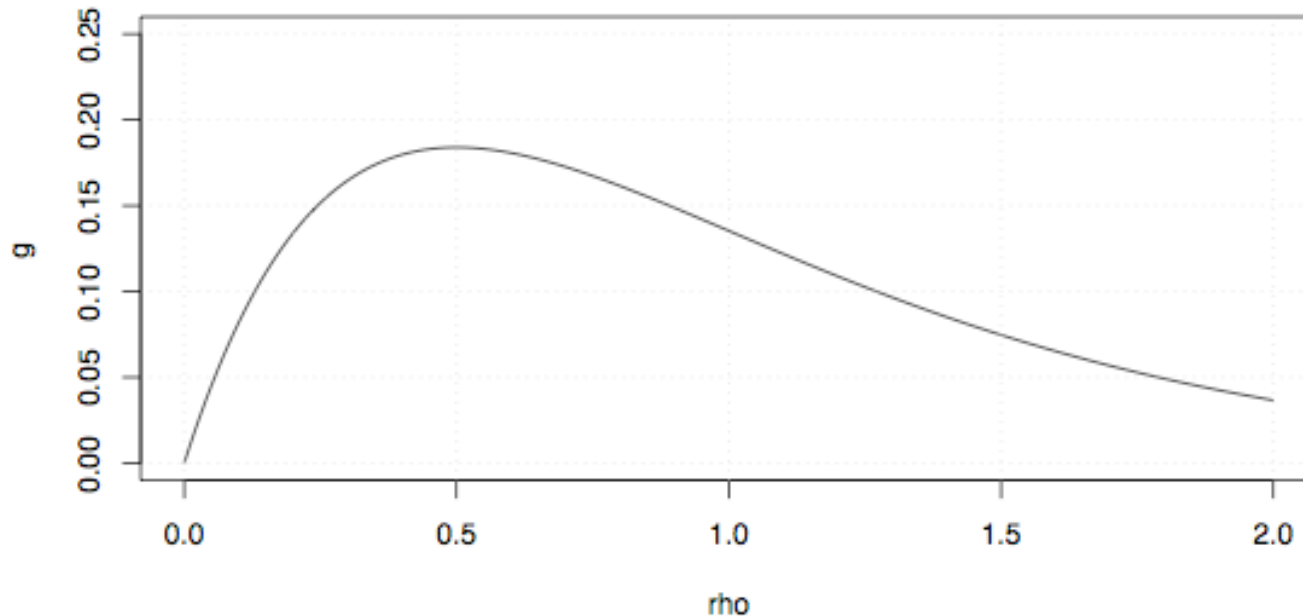
- Probabilidad de 0 llegadas en  $\Delta t = 2m$   $P_{2mN\lambda}[0] = e^{-2mN\lambda}$

# Prestaciones ALOHA

- Tráfico ofrecido  $\rho = Nm\lambda$
- Mensajes que llegan  $\lambda' = \lambda P[\text{llegar}] = \lambda e^{-2mN\lambda}$
- Tráfico aprovechado (cursado, goodput)

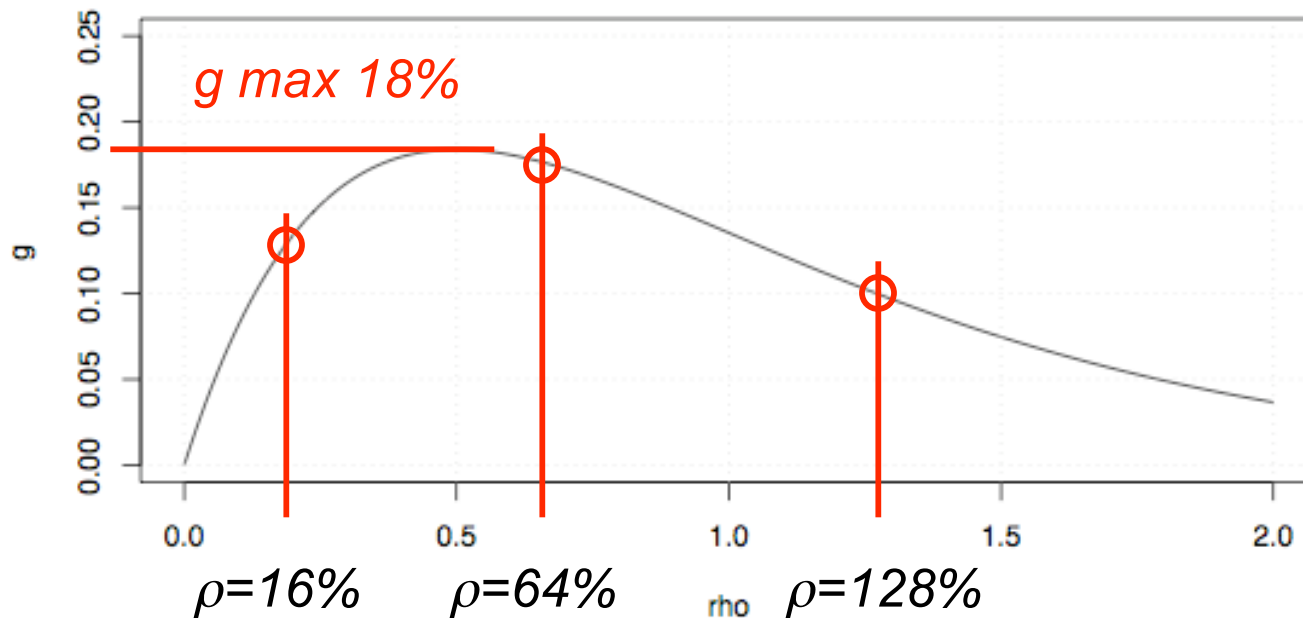
$$g = Nm\lambda' = Nm\lambda e^{-2mN\lambda} = \rho e^{-2mN\lambda} = \rho e^{-2\rho}$$

$$g = f(\rho)$$



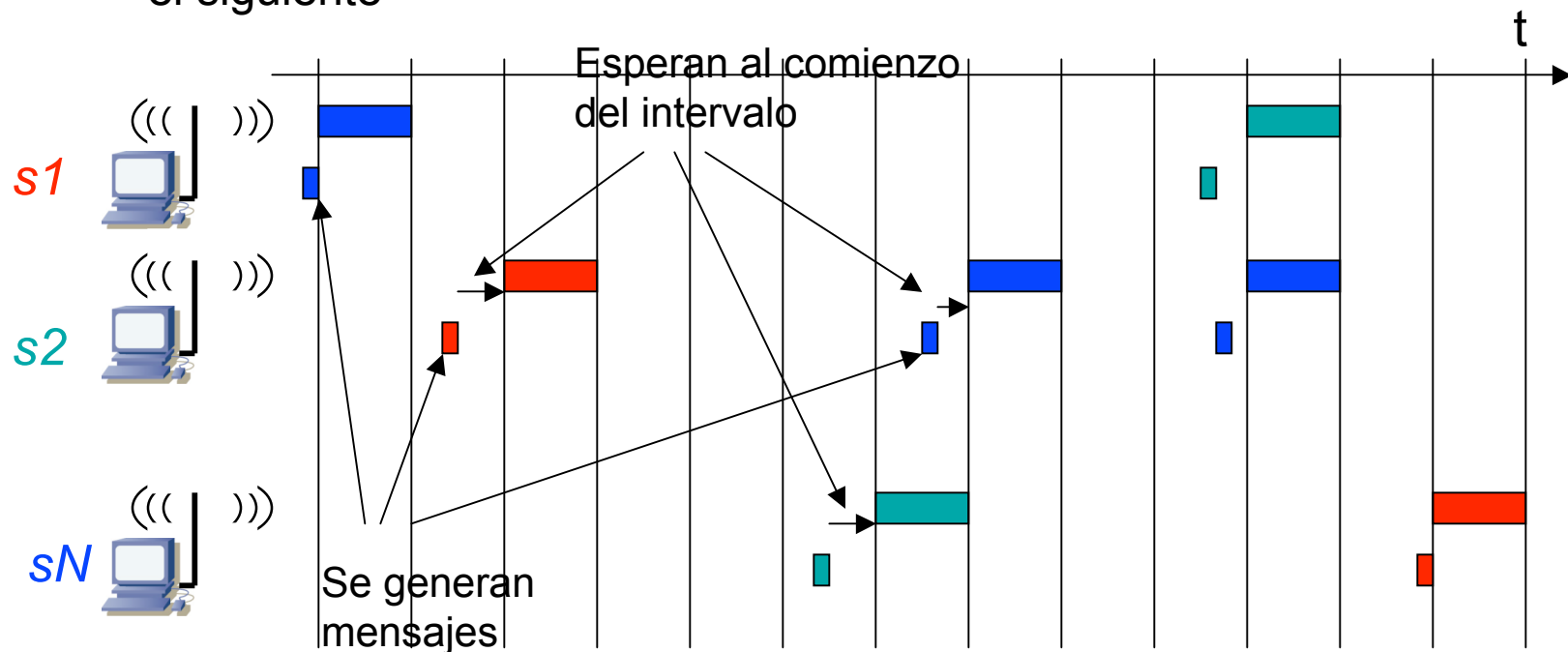
# Prestaciones ALOHA

- Máximo goodput  $\sim 18\%$  (para  $\rho=50\%$ )
- No parece muy eficiente
- Si transmitimos al canal 1Mbps como mucho tendremos 180Kbps para repartir entre todos los que transmitan...
- Por otra parte si la velocidad es aceptable ALOHA resuelve el problema del acceso al medio



# Mejorando ALOHA

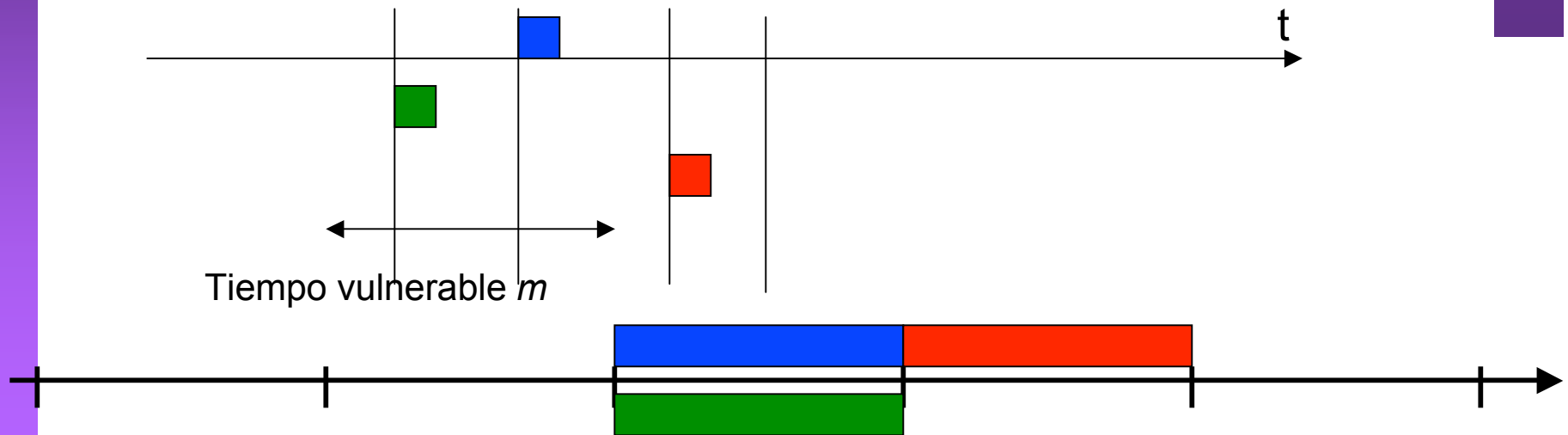
- Mejora típica **ALOHA ranurado**
- Dispositivos sincronizados en el tiempo. Existen intervalos temporales (slots) conocidos por todas las estaciones  
Slots de la duración de la trama
- Algoritmo: Igual que ALOHA
  - Salvo que sólo se puede empezar a transmitir al principio del slot.
  - Si un paquete se genera en un slot deberá esperar a que comience el siguiente



# ALOHA ranurado

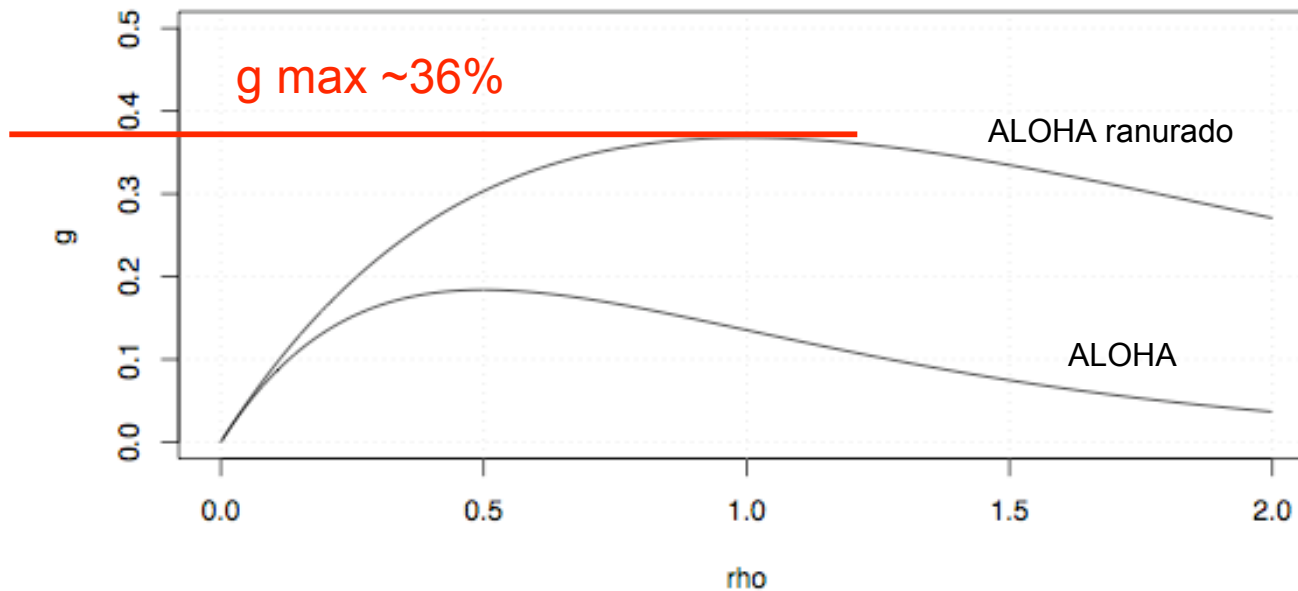
- La trama colisionará con los paquetes que llegan en el mismo slot
- Si llegan cerca pero en el siguiente slot esperarán
- El tiempo vulnerable es ahora  $m$
- La probabilidad de no colisionar es ahora la probabilidad de que se produzcan 0 llegadas en un tiempo  $m$

$$P_{mN\lambda}[0] = e^{-mN\lambda}$$



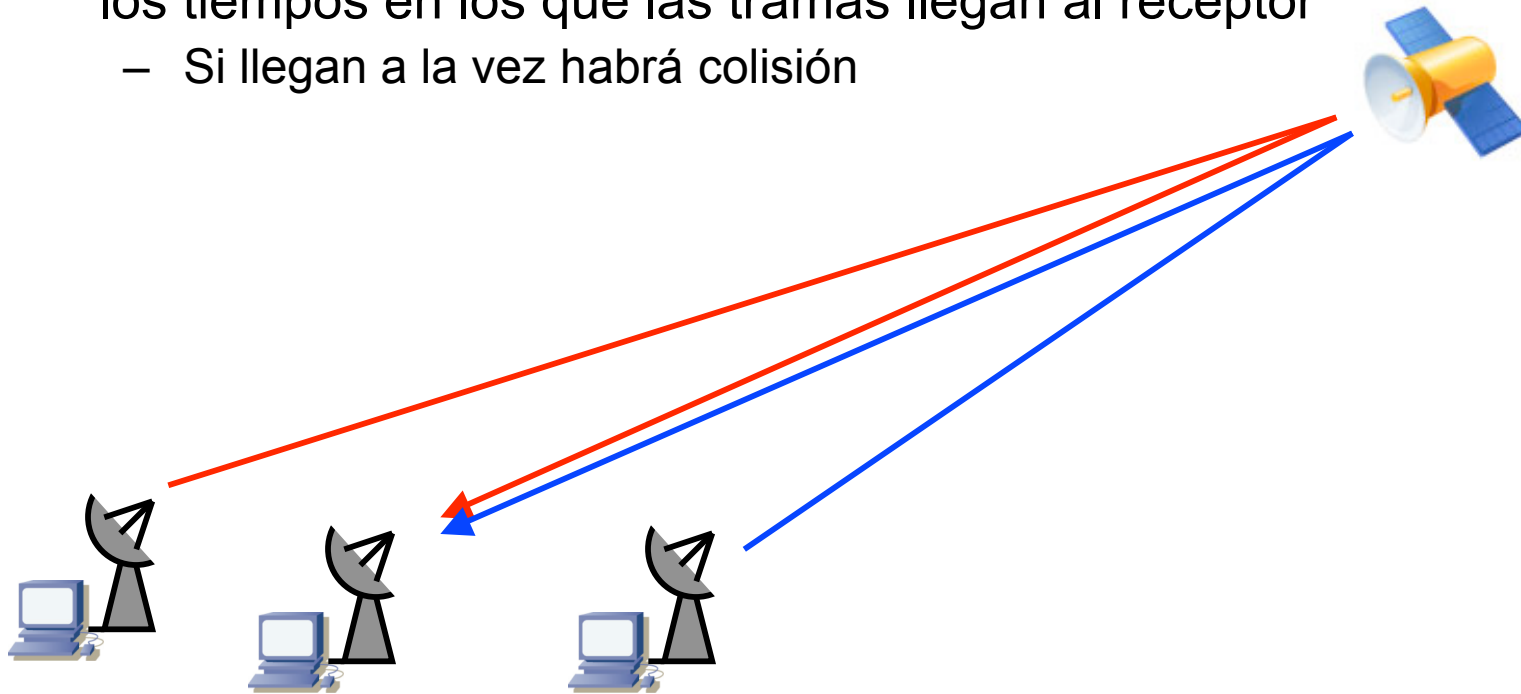
# ALOHA ranurado

- El goodput en función de la carga ahora sera  $g = \rho e^{-\rho}$
- Mejora el de ALOHA
- El máximo alcanzable es ahora ~36%
- Desventaja: es más complejo, tener sincronización en los nodos no siempre es facil



# Tiempo de propagación

- Estabamos suponiendo en los dibujos que el tiempo de propagación era pequeño
- ¿Que pasa si no lo es?  
Nada. Es más difícil de dibujar. Pero ALOHA funciona igual
- De hecho protocolos de tipo ALOHA se usan en escenarios de comunicación por satélite
- El análisis de las colisiones es equivalente, lo que importa son los tiempos en los que las tramas llegan al receptor
  - Si llegan a la vez habrá colisión





# ALOHAAs resumiendo

- Resuelven el problema de acceso al medio de forma simple
- No son demasiado eficientes (máximos de 18%-36%)
- Funcionan independientemente de que el tiempo de propagación sea grande o pequeño comparado con el de transmisión
- ¿Podemos mejorar esto?

# Mejorando ALOHA

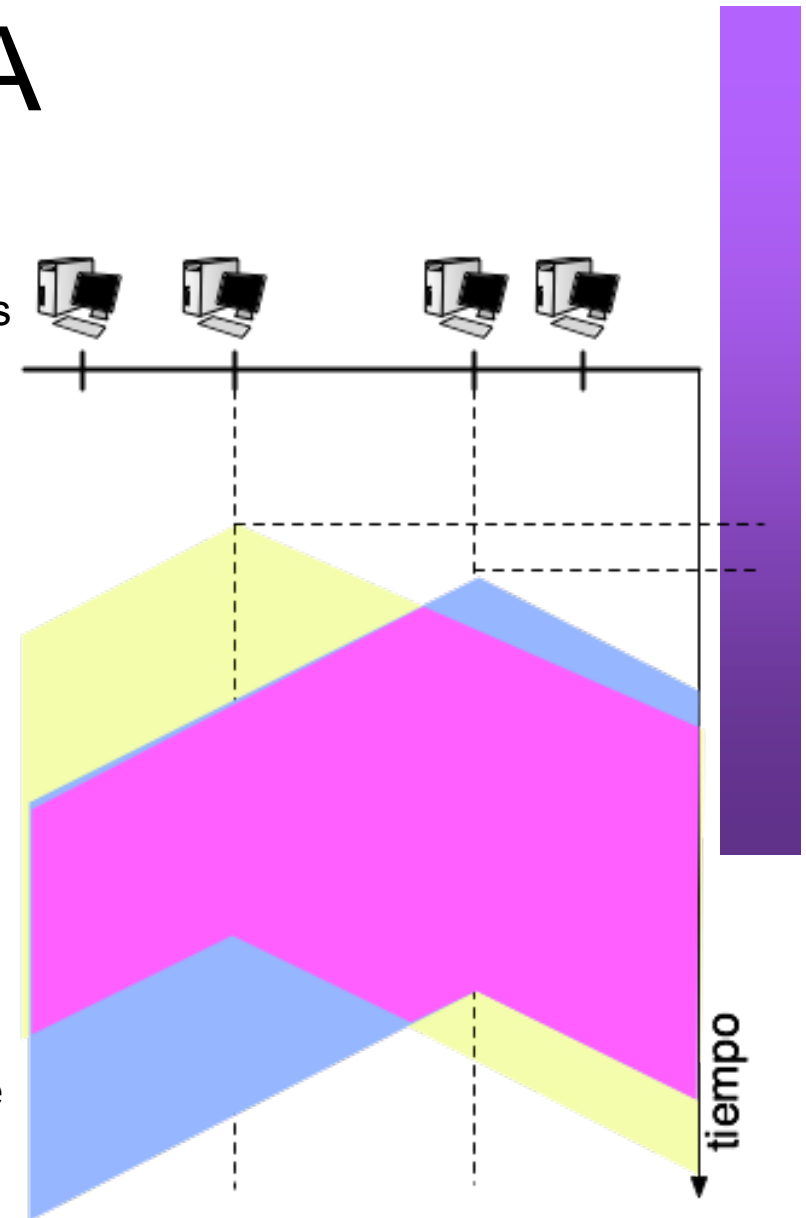
- ¿Podemos mejorar el protocolo si el tiempo de propagación es pequeño comparado con el de transmisión?
- Hay una mejora obvia...

## Mirar antes de enviar

- Solo tiene sentido si la transmisión se mantendrá más tiempo que el que tardo en decidir si hay una transmisión es decir (propagación  $\ll$  transmisión)
- Y qué hago si el medio esta ocupado?
  - Espero a que este libre y envío entonces?
  - Espero un tiempo aleatorio?

# CSMA

- CSMA (Carrier sense multiple access)  
Acceso múltiple con detección de portadora
- Tiempo de propagación corto: Las estaciones saben rápido si otra estación está transmitiendo
- Primero escucha y espera a que el medio este libre (Carrier Sense)
- Si el medio esta libre -> transmite  
Si no espera
- Aun así hay puede haber colisiones porque el tiempo de propagación no es cero  
Si dos estaciones empiezan a transmitir al mismo tiempo (aproximadamente por el tiempo de propagación)  
= COLISION
- La utilización máxima depende del tiempo de propagación (longitud del medio) y de la longitud de la trama



# Variaciones de CSMA

Surgen de contestar a estas preguntas

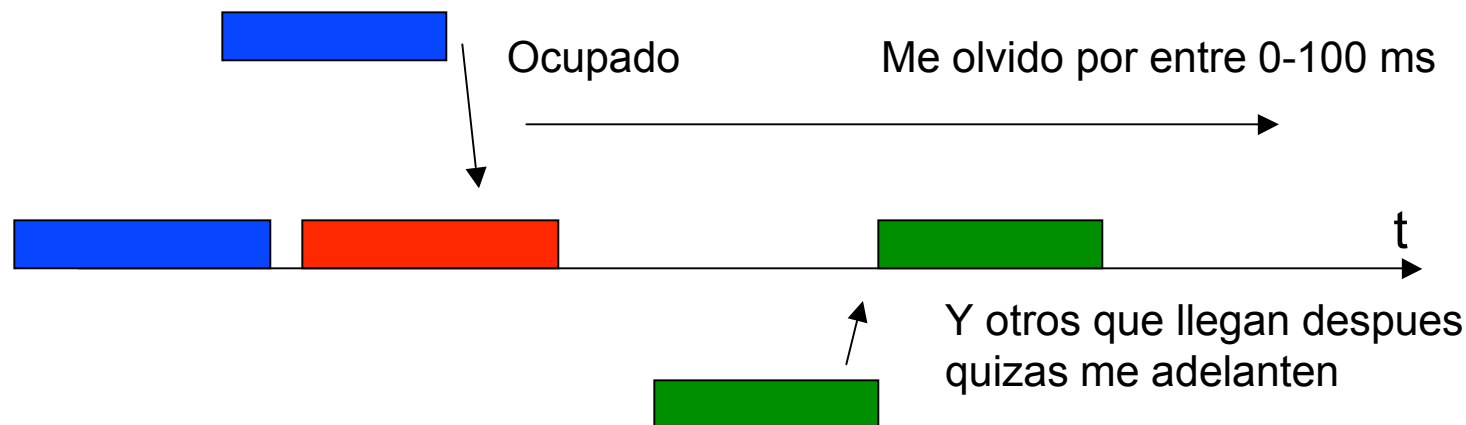
- ¿Qué hago si el canal esta libre?
- ¿Qué hago si se produce a pesar de todo una colision?

# CSMA No persistente

- **Reglas:**
  - Si el medio esta libre, transmitir
  - Si el medio esta ocupado, espera un tiempo generado con una distribucion de probabilidad (retardo de retransmisión) e intentalo de nuevo
- **Ventaja**

El retardo aleatorio reduce la probabilidad de colisión
- **Pero...**

Se desperdicia capacidad porque el medio se queda libre despues de una transmision
- Las estaciones no persistentes son respetuosas

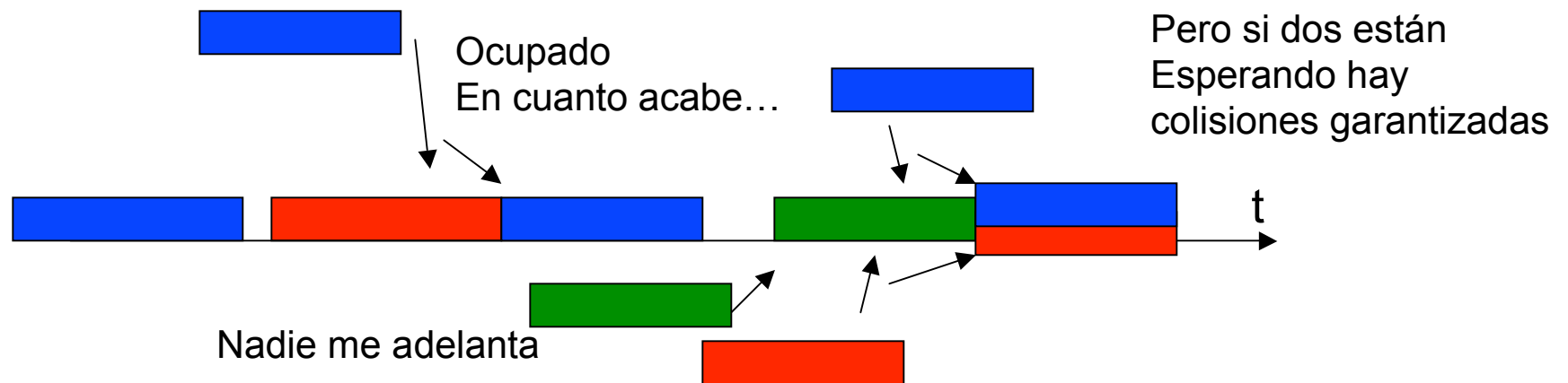


# CSMA 1-persistente

- Evita que el canal quede libre
- **Reglas**
  - Si el medio esta libre transmitir
  - Si el medio esta ocupado espera hasta que este libre; entonces transmite inmediatamente
- **Ventaja:**

Aumenta la utilización del canal (no se desperdicia tiempo)
- **Pero...**

Si hay 2 o mas estaciones esperando la colision esta garantizada
- Las estaciones 1-persistentes son egoistas



# CSMA p-persistente

- Compromiso para intentar reducir las colisiones y que el canal no quede vacío
- **Reglas:**
  - Si el medio está libre transmite con probabilidad  $p$  y espera una unidad de tiempo con probabilidad  $(1-p)$
  - Si el medio está ocupado espera hasta que este libre y repite el paso 1
  - Si la transmisión se retrasa una unidad de tiempo repite el paso 1
- Pendiente: elegir el valor efectivo de  $p$  para evitar inestabilidad en alta carga

# Valor de $p$ ?

- $n$  estaciones esperando transmitir
- Al final de la transmisión el número esperado de estaciones será  $np$ 
  - Si  $np > 1$  en media habrá una colisión
- Si hay una colisión esas  $n$  estaciones se sumarán a las que lleguen nuevas y aumenta la probabilidad de colisión
- Eventualmente todas las estaciones intentan retransmitir y el throughput tiende a cero
- Así que debe cumplirse que  $np < 1$  para los picos esperados de  $n$ 
  - Si esperamos mucha carga,  $p$  debe ser pequeño
  - Pero  $p$  pequeño quiere decir que las estaciones esperan mucho para transmitir



# Qué algoritmo de persistencia?

- IEEE 802.3 (Ethernet) usa 1-persistente
- No persistente y p-persistente tienen problemas de eficiencia
- 1-persistent parece más inestable que p-persistente
  - Porque las estaciones son egoistas
  - Pero el tiempo desperdiciado en las colisiones es pequeño
- Se puede solucionar utilizando un tiempo de backoff (tiempo sin volver a intentarlo)
  - Con un tiempo de backoff aleatorio es poco probable que vuelvan a colisionar de nuevo en el siguiente intento
- También se puede recuperar el tiempo de la colisión:

## **Detección de colisiones**

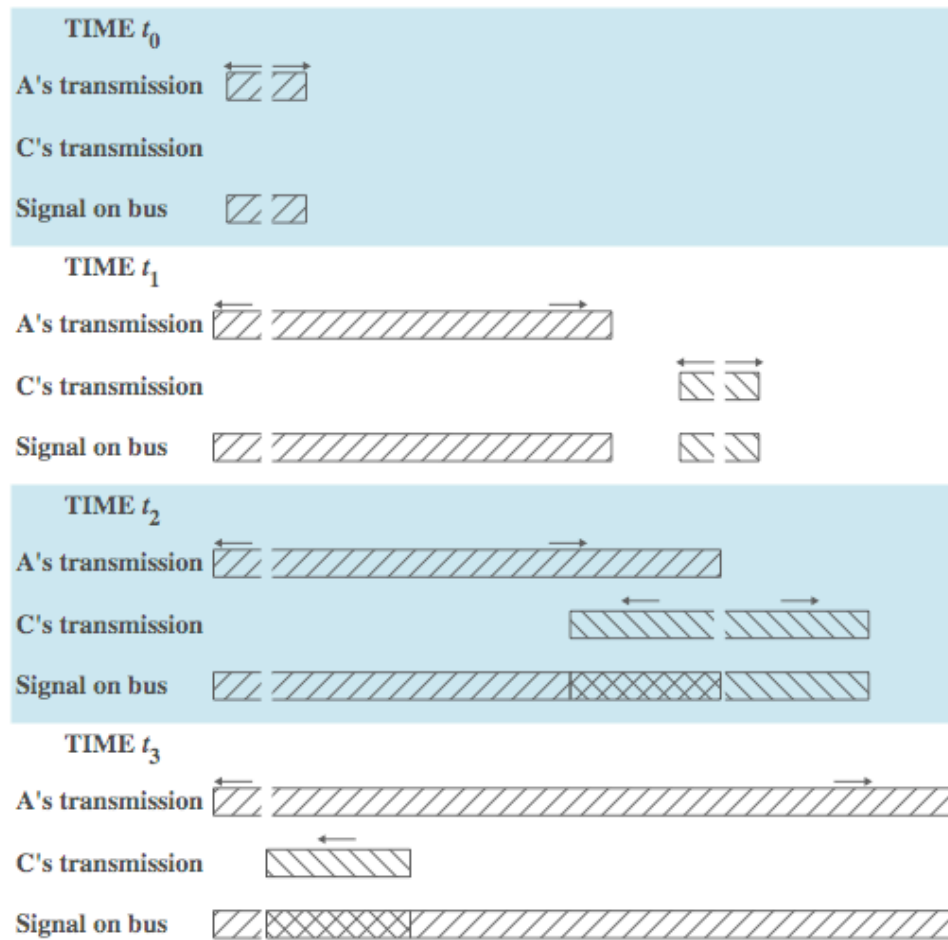
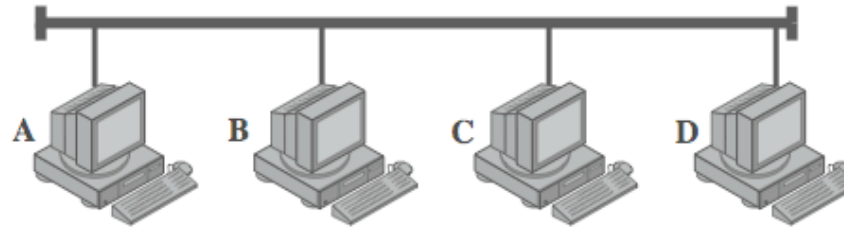
# CSMA/CD

- Con CSMA, la colisión ocupa el medio durante el tiempo de transmisión
- Se puede mejorar si las estaciones son capaces de recibir a la vez que transmiten
  - No siempre es posible
  - Determinado hardware por ejemplo antenas o receptores no permiten a la vez enviar y escuchar el medio
- CSMA/CD reglas:
  - Si el medio está libre transmitir
  - Si está ocupado esperar a que este libre y transmitir
  - **Si veo una colisión dejar de transmitir**
  - Después esperar un tiempo aleatorio y retransmitir
- CD = Detección de colisión (collision detection)

# Detección de colisión

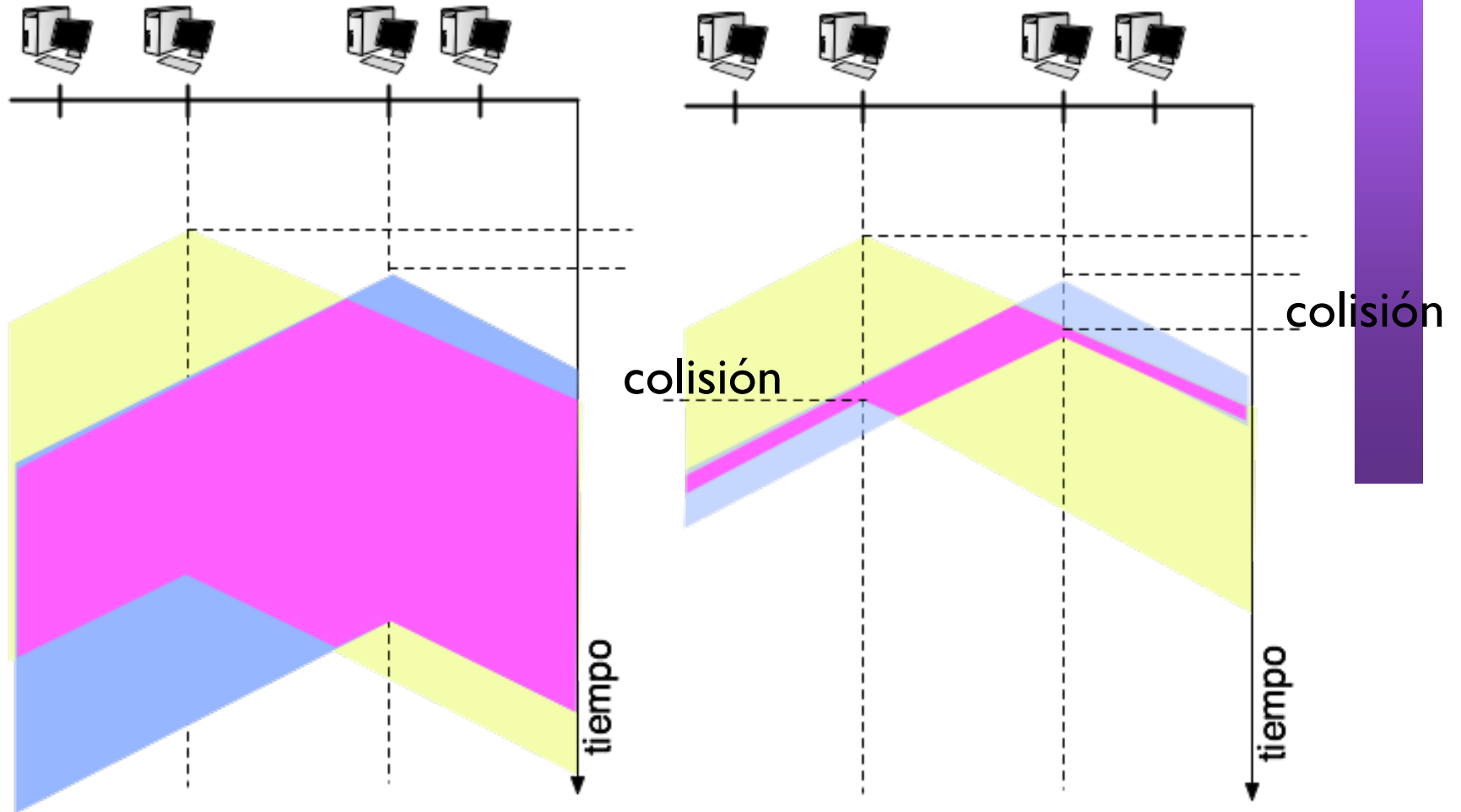
- Bus en banda base
  - La colisión produce mayor voltaje
  - Si la señal del cable es mayor que la que está generando la estación detecta colisión
  - La señal se atenúa con la distancia
  - Límite de 500m (10Base5) o 200m (10Base2)
- En par trenzado (topología en estrella)
  - Actividad en más de un puerto es una colisión
  - Se usa una señal especial para indicar colisión

# CSMA/CD



# CSMA/CD

- Cada dispositivo para al detectar la colisión

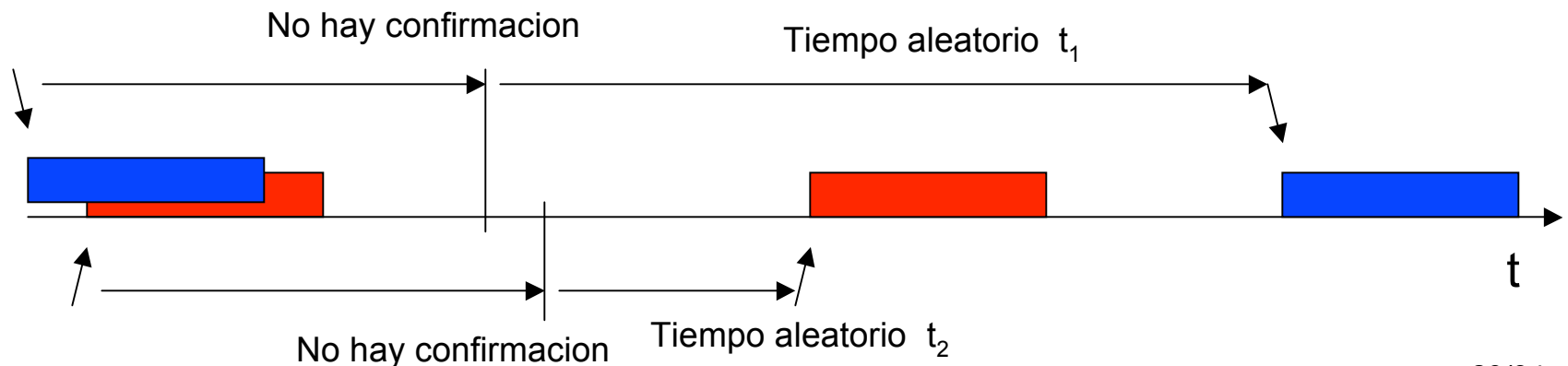


# Backoff

- Antes de volver a transmitir en muchos protocolos se espera un tiempo aleatorio
  - Normalmente para evitar la coincidencia de varias estaciones que puedan querer transmitir a la vez

## Le llamaremos **backoff**

- Por ejemplo en ALOHA se usa despues de una trama errónea para no volver a colisionar
- Consigue efectos parecidos al CSMA p-persistente
- Lo más simple es elegir un número aleatorio uniforme en un rango conocido
- En CSMA/CD se complica un poco



# Binary Exponential Backoff

- IEEE 802.3 y Ethernet usan binary exponential backoff
- Las estaciones reintentan el envío de las tramas que colisionan
  - En los 10 primeros intentos el tiempo medio de espera se dobla
  - En los 6 siguientes el tiempo medio se mantiene constante
  - Después de 16 colisiones la estación desiste y da error para esa trama
- El algoritmo 1-persistente con binary exponential es eficiente para un amplio rango de cargas
  - Poca carga, ocupa el canal inmediatamente
  - Mucha carga, espera más tiempo y hay menos colisiones
- Problema: el backoff tiene un efecto last-in, first-out
  - Las estaciones con tramas nuevas tienen preferencia sobre las que ya llevan tiempo reintentando un envío
  - Captura del canal

# Control de acceso: clasificación

- Esto no ha sido una clasificación exhaustiva  
Hemos visto los protocolos más básicos/usados
- Diferentes tipos para diferentes redes de área local
- Según dónde se controle
  - **Centralizado** (un dispositivo decide quien transmite)
    - Más control, un único punto de fallo
  - **Distribuido** (se resuelve el derecho a transmitir sin dispositivo central (colisiones))
    - Más complejo pero mas robusto
- Según cómo se controle
  - **Síncrono** (modo circuito)
    - capacidad dedicada (por conexión)
    - No óptimo
    - Usado en GSM y en ciertos tipos de protocolos para satelites
  - **Asíncrono** (modo paquete)
    - En respuesta a la demanda, tengo un mensaje y reservo o compito con los demas para transmitirlo



# Control de acceso al medio asíncrono

- Varias filosofías
- Round robin
  - A cada estación se le da el turno para transmitir
- Reserva
  - Divide el tiempo en slots
  - Petición y concesión de slots
  - Bueno para tráfico continuo
  - Reserva no implica centralizado
  - Hay técnicas de resolver reservas distribuidas basadas en ALOHA (R-ALOHA) tiempo destinado a pedirse el canal, el que consigue transmitir la reserva con ALOHA tiene derecho al slot de transmisión que viene después
- Contención
  - Las estaciones compiten por el tiempo
  - Bueno para tráfico a ráfagas
  - Simple de implementar
  - Tiende a colapsarse con mucha carga
  - ALOHA, S-ALOHA, CSMA, CSMA/CD, CSMA/CA son de estos

# Ejemplos

Donde	Cuando	Protocolo	Usos
Centralizados	Sincronos/circuitos	GSM	Telefonía celular
	Asincronos/paquetes	Polling/probing	Redes de cable
		Basados en reserva: FPODA, PDAMA	Satelite
Distribuidos	Asincronos/paquetes	Polling/probing	Redes de cable
		CSMA/CD	Redes de cable
		CSMA/CA	Redes inalambricas
		BTMA: MACA, MACAW	Redes inalambricas
		Token ring, FDDI	Redes de cable
		ALOHA, S-ALOHA, R-ALOHA	Satelite

# Conclusiones

- Los protocolos de acceso al medio permiten a varios dispositivos compartir un mismo medio de transmisión de forma coordinada
- Protocolos basicos
  - ALOHA y slotted-ALOHA
  - CSMA, 1-persistente, p-persistente, no-persistente
  - CDMA/CD y veremos CSMA/CA
- Son de tipo distribuido y asíncrono. Tambien existen centralizados y síncronos. Aunque no se usan tanto en LANs