

## • Redes de computadores - Capítulos II: Capa aplicación

Aplicación de red: programa que, desde un sist. terminal, se comunica por la red.

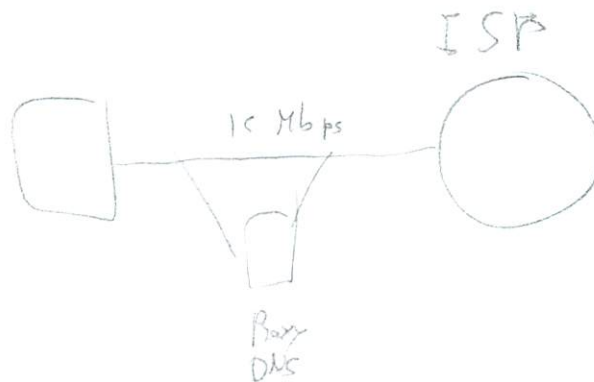
### 1.- Principios de las apps. en red:

#### 1.1.- Arg. de las aplicaciones en red:

- Cliente - Servidor:
  - Host siempre activo, Se comunican con el servidor
  - IP permanente ↔ Gestión intermitente
  - Clusters. IP's dinámicas
  - Comunicación ↑ No se comunican entre sí

#### • Arquitectura P2P (pura)

- No hay un servidor siempre activo.
- Sist. terminales se comunican directamente entre sí.
- Los pares se conectan intermitentemente y cambian IP.
- Muy escalable, difícil de gestionar.



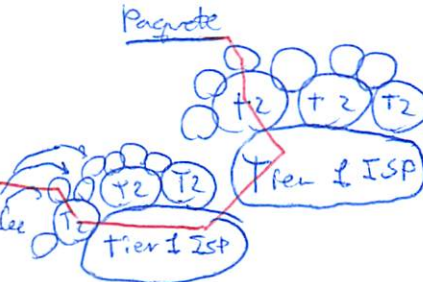
$$d_{\text{nodo}} = d_{\text{procesos}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{prop}}$$

$$\leftarrow < 1 \text{ ms} \quad \frac{L_c}{R} \quad \frac{L}{\text{Ancho banda}} \quad \frac{1000}{\text{vel prop}}$$

## Redes de computadores - Capítulo 1. - P4H

### 3.3.- Estructura de internet: red de redes locales

- Grossa nodo es jerárquica.
- Pequeño nº de redes bien conectadas.
- TIER 1: ISPs comerciales, y grandes distribuidores de contenido (Nacionales e internacionales)
- TIER 2: ISPs más pequeñas (regionales). Se conectan a los tier 1. Redes de clientes
- TIER 3: ISPs locales. Redes de acceso.



### 4.- Retardos, pérdidas y tasa de transferencia en las redes de comunicación de paquetes.

Retardo de nodo general.

•  $d_{proc}$ : proceso nodal. Comprueba enlce y determina salida. Normalmente:  $< 1 \mu s$ .

•  $d_{cola}$ : retardo de cola: tiempo salida para transmisión.

•  $d_{trans}$ : retardo transmisión =  $\frac{\text{Longitud paquete (bits)}}{\text{ancho de banda (bps)}}$

•  $d_{prop}$ : retardo propagación =  $\frac{\text{Longitud enlace físico}}{\text{vel. prop. en el medio}} \quad (\sim 2 \cdot 10^8 \text{ m/s})$

•  $d_{cola} = \frac{L_c}{R}$  }  $\left. \begin{array}{l} L_c = \text{long. paquete (bits)} \\ a = \text{tasa de llegada paquetes} \\ R = \text{ancho banda} \end{array} \right\} \begin{array}{l} \sim 0 \text{ pequeño retardo} \\ \sim 1 \text{ grande retardo} \\ > 1 \text{ más trabajo del que se puede atender.} \end{array}$

4.1.- Pérdida de paquetes: si la cola del enlace se llena, se ignorarán los paquetes que lleguen. Pueden volver a enviarse desde el nodo anterior o perderse para siempre.

4.2.- Tasa de transferencia: velocidad a la que se transfieren bits entre emisor y receptor.

• Instantánea: vel. en un instante.

• Media: vel. sobre un periodo de tiempo.

### 5.- Capas de protocolo y modelos de servicios:

Dentro de	Aplicación	→ Da servicios a las apps de red. (FTP, SMTP...)
	Presentación	→ Permite interpretación de datos. (encabezados, compresión...)
	Sesión	→ sincronización, organización, recuperación de datos
	Transporte	→ Transf. de datos de proceso a proceso (TCP, UDP)
	Red	→ Enrutamiento de paquetes de origen a destino. (IP)
	Enlace	→ Transf. datos entre elementos vecinos. (Ethernet)
	Física	→ Bits en el cable.

Retardos:

$$N = \lambda \cdot T \quad \left\{ \begin{array}{l} N = \text{promedio de paquetes en router} \\ \lambda = \text{tasa promedio de llegada} \\ T = \text{tiempo promedio en el router} \end{array} \right.$$

$$\frac{L_a}{n} = \frac{L}{R} \cdot \lambda = \frac{\lambda}{\mu} = \rho \quad \left\{ \begin{array}{l} \mu = \text{tasa de salida} \\ \lambda = \text{tasa prom de llegada} \\ \rho > 1 \Rightarrow \text{Atasco seguro} \\ \rho < 1 \Rightarrow \text{Posible atasco} \\ \rho = \text{paquetes por seg dentro del router} \end{array} \right.$$

$$\rho = \frac{\lambda}{\mu}; N = \frac{\rho}{(1-\rho)}$$

$$d_{\text{cola}}(W) = \frac{\rho}{(\mu - \lambda)}$$

Poisson

De un promedio  $\lambda$ ,  $P(x)$

Binomial

De  $N$  veces, pase la situación  $x$  con  $p(x) = 0.5$

SDM  + TDM 

$$d_{\text{total}} = d_{\text{proc}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{trans m}}$$

$$\frac{L(x) + \text{llegada}}{R} \quad \frac{L}{R} \quad \frac{m}{v}$$



# Redes de computadores - Capítulo 1

## 1.- Internet

### 1.1.- Componentes de la red:

- Dispositivos terminales: ejecutan aplicaciones de red
- Enlaces de comunicación: fibra, cobre, satélite... determinan el ancho de banda
- Routers: envían paquetes.
- Protocolos: controlan envío y recepción mensajes
- Internet: red de redes
  - RFC
  - IETF

### 1.2.- Servicios: infraestructura de comunicación. Permite apps. distribuidas Web, juegos, email...

↳ Proporcionados a las aplicaciones: { Envío fiable  
Envío no fiable: lo mejor posible.

### 1.3.- Protocolos: definen la interacción entre aplicaciones o computadores:

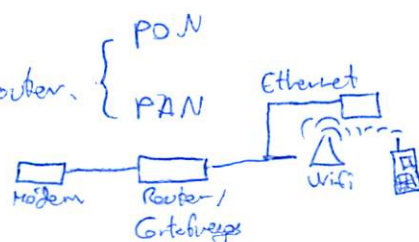
- Controlan formato
- Orden de los mensajes
- Acciones a realizar al recibir o transmitir mensajes.

## 2.- Frontera de la red: aplicaciones y sistemas terminales.

- Se accede mediante redes de acceso: cableada o wireless.
- Núcleo de la red: routers interconectados.

### 2.1.- Acceso: se conecta el sistema terminal al router de frontera. { Ancho de banda Compartida o dedicada

- Acceso telefónico: 56 kbps. No permite usar Hf. y red a la vez.
- (A)DSL: infraestructura telefónica. Línea física a centralita. Acceso dedicado.
- HFC: Híbrido fibra-coaxial. Asimétrico.
- Red fibra óptica: domicilios comparten acceso router.
- Wifi (Wireless): inalámbrica y compartida.



### 2.2.- Medio físico: medio mediante el cual se da la conexión.

- bit: se propaga entre emisor/receptor.
- enlace físico: G que hay entre emisor y receptor.
- Medios { guiados: señales propagadas por medio sólido. Ej. fibra óptica
- no guiados: señales propagadas libremente. Ej. radio.

\* Par trenzado: dos cables de cobre aislados. Ethernet.

\* Coaxial: dos conductores de cobre concéntricos. Banda ancha: varios canales.  
Banda base: canal único.

\* Fibra óptica: cable de fibra que lleva pulsos de luz. Cada uno es un bit.

↳ Alta velocidad

↳ Poca pérdida de energía: ni ruido ni electro magnético.

2.3.- Radio: señal transmitida a través del espectro electromagnético.

- Bidireccional, sin cables
- Efectos del entorno: reflexión, obstáculos por objetos, interferencias...

Tips:

- \* *Microcodus tenestres*

- \* LAN (WIFI) :

- \* Área extensa (móvil).: p. ej. 3G

- \* Subsite: canales de hasta 45 Mbps.

↳ Rebendas 270 ms

→ Geostrophie: beginn latitud.

3.- Núcleo de la red: malla de routers, interconectados.

3.1.- Conmutación de circuitos: recursos de la red divididos en "parcelas."

- Porción asignada a llamada: no se usa si no se la llama

- División del ancho de banda por  $\underbrace{\text{frecuencia}}_{\text{FDM}}$  ó por  $\underbrace{\text{tiempo}}_{\text{TDM}}$ .

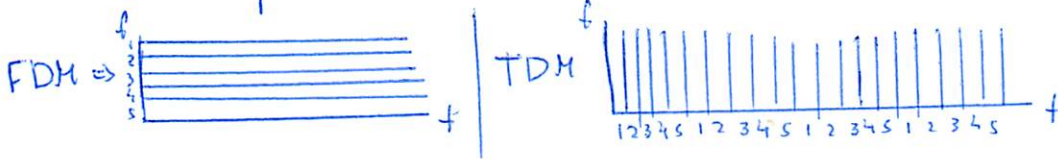
Ej. TDM: enviar de A a B 640 000 b. a 1536 Mbps,

con 24 particiones y tardando 500 ms. en establecer circuitos.

Solution:

$$6400006 / ((1536 \text{ kbps} / 24 \text{ particiones}) \cdot 10^6) = 0.01 \text{ s} = 10 \text{ ms}$$

500 ms para conec. + 10 ms para transf. = 510 ms totales.



### 3.2.- Conmutación de paquetes

Usuario A  $\rightarrow$ 

A	A	A
---	---	---

 $\rightarrow$ 

A	A	B	A	B
---	---	---	---	---

 $\rightarrow$ 

A	A	A	B	A	B	A
---	---	---	---	---	---	---

 ...

- Cada flujo se divide en paquetes.
- Los paquetes comparten los recursos de la red.
- Cada paquete usa todo el ancho de banda.
- Los recursos se ocupan a demanda.

$$\text{Retardo} = \frac{3. \text{ Mbits por paquete}}{(\text{vel}) \text{ Mbits por seg.}}$$

- Puede surgir congestión.
- Pueden perderse paquetes si se supera el almacenamiento.
- ↳ Se debe recibir un paquete completo antes de retransmitirlo.



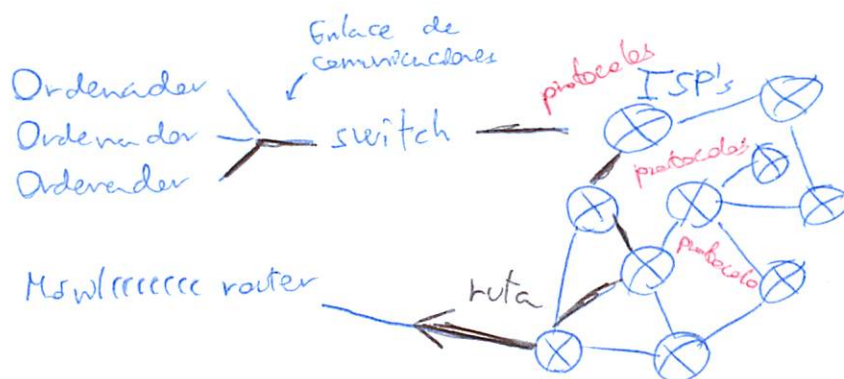
## • Arquitectura de redes

### • Capítulo 1.

• ¿Qué es internet?: Red de comunicación entre dispositivos.

Componentes:

- Sistemas terminales: dispositivos conectados a la red, usados por usuarios finales, conectados mediante enlaces de comunicaciones y conmutadores de paquetes. Ejemplo: móviles, pc's, portátiles...
- Enlaces de comunicaciones: medios físicos que unen sistemas terminales. Tienen una velocidad de transmisión concreta. Ejemplo: fibra óptica, cable coaxial...
- Gestor de paquetes: un sistema terminal envía datos a otro, segmentándolos: paquetes. Una vez los recibe el otro S.T., los ensambla y obtiene los datos originales. Un conmutador de paquetes recibe paquetes y los reenvía hacia su destino. La secuencia de enlaces que siguen los paquetes del S.T. → S.T. se le llama ruta. Convencionalmente, los conmutadores de paquetes son routers y switch's.



ISP: Internet service provider: red de conmutadores de paquetes que proporcionan acceso a la red a los S.T., y están interconectados entre sí.

Protocolos: controlan el envío y la recepción de información dentro del internet.

Estándares de internet: es importante que todo el mundo esté de acuerdo en utilizar los mismos protocolos para que se creen sistemas capaces de interoperar. Esto se recoge en el documento RFC, que contiene contenido técnico sobre estos protocolos.

## • Servicios

- Internet: infraestructura que proporciona servicios a las aplicaciones
- Aplicaciones distribuidas: implican varios S.T. intercambiando datos entre si.

↳ Estas apps. tienen un interfaz de sockets, mediante el cual ~~envían~~ envían datos de una app. dist. en un s.t. a otra en un s.t.

## • PROTOSCOLOS SEGMENTADOS:

### • GBN: Go back N

El receptor rechazará todo paquete que esté en desorden.  
Los paquetes deben llegar correctos y en orden.

Cada vez que llega un paquete, el receptor envía ACK rechamando al siguiente.

N: tamaño de la ventana. (Paquetes que puede lanzar el emisor antes de recibir ACK)

$$NS_{\min} - NS_{\max} \leq N \quad \text{y} \quad NS_{\min} \leq NR \leq NS_{\max}$$

Temporizador: cuando llega ACK, se pone a 0. Si adquiere un valor muy alto, se volverá a enviar todo el conjunto de paquetes.

### • SR: Selective Repeat

El receptor tiene un buffer donde recibe paquetes.

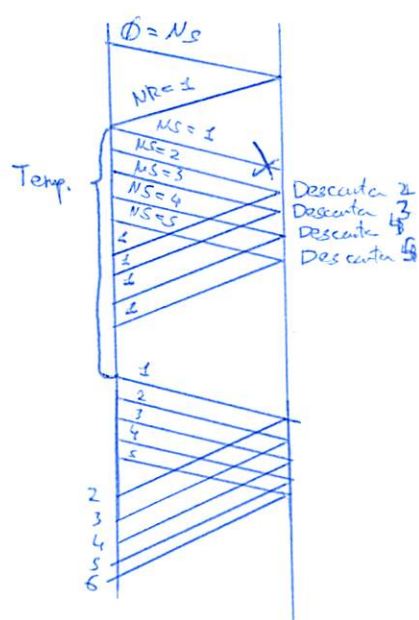
En cuanto tenga varios paquetes continuos, desplaza el buffer.

Temporizador individualizado para cada paquete.

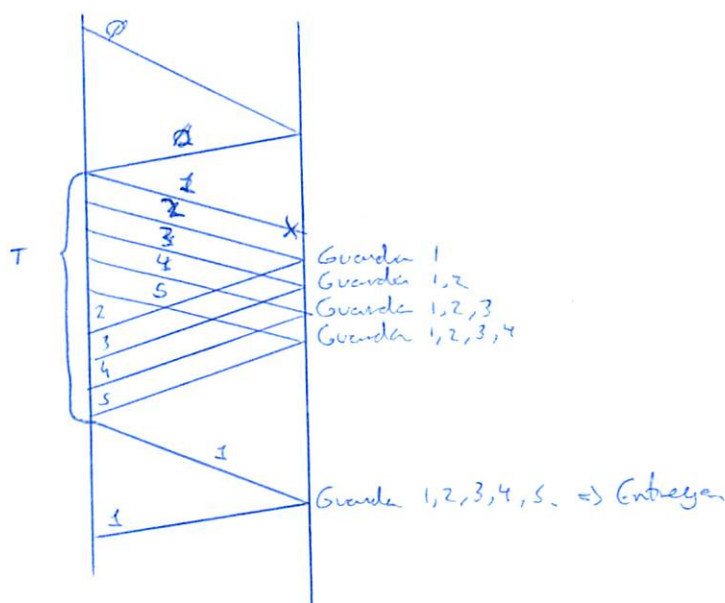
El ACK reconoce un paquete recibido, NO solicita el siguiente.

↳ Si llega un paquete duplicado, devuelve su n° reconocimiento.

#### • GBN

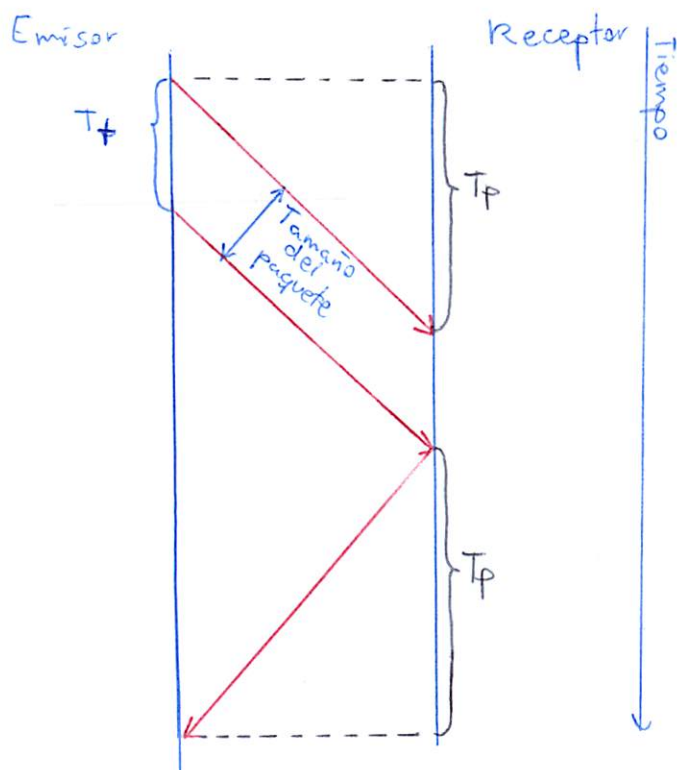


#### • SR





Arg. Redes - T3



- $T_p$  = tiempo de propagación  
 $\hookrightarrow$  tiempo que se tarda en enviar 1 bit del emisor al receptor.
- $T_t$  = tiempo de transmisión  
 $\hookrightarrow$  vel. a la que es capaz de sacar los bits del ordenador y meterlos en el paquete de red.
- RTT: tiempo de ida y vuelta.  
 $L \Rightarrow$  Tamaño (bytes) del paquete.  
 $R \Rightarrow$  Vel. del emisor.

$$\text{Velocidad} = \frac{1}{\frac{L}{R} + RTT}$$

$$RTT = 2T_p$$

Protocolo ARQ (ACK, NS, NR) (Parada y espera)

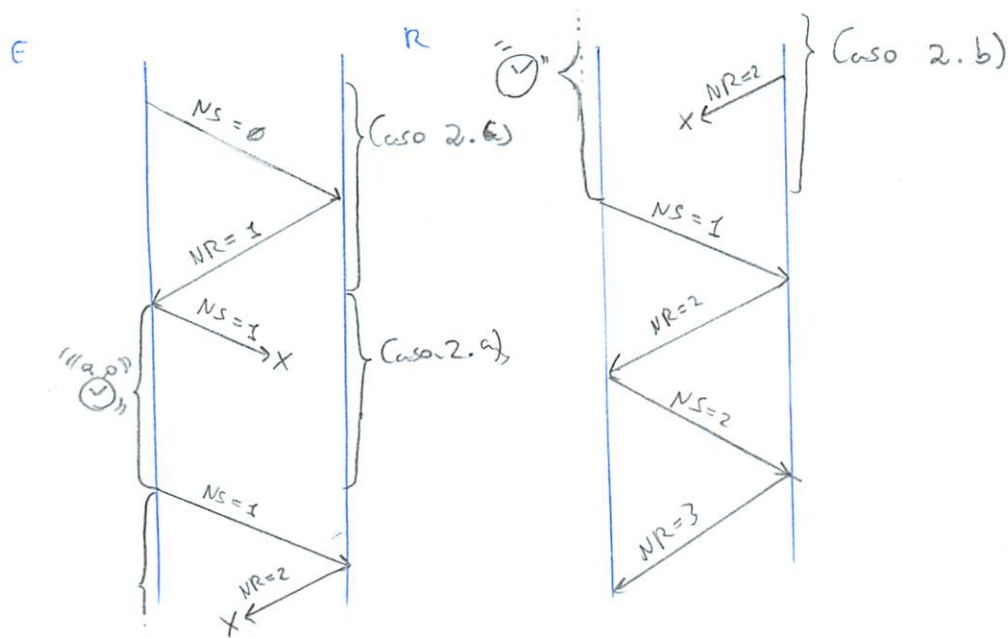
"si el paquete llega mal o no llega, vuélvelo a mandar"

NS = n° de seguimiento

NR = n° de reconocimiento

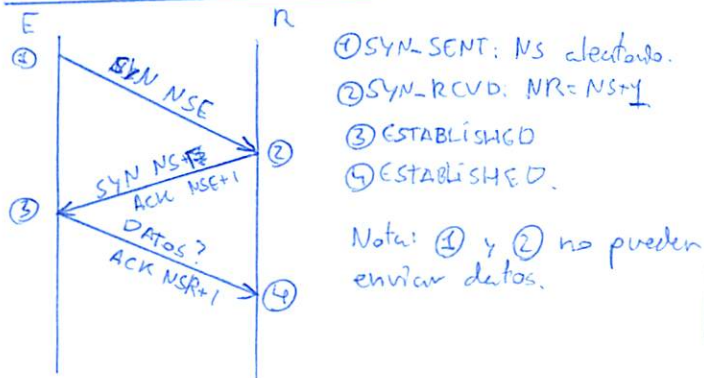
① Emisor envía paquete con NS. (N° de seguimiento)

- ② 3 opciones
- (2.a) No llega el paquete:  
 $\hookrightarrow$  Pasado el temporizador, el emisor vuelve a enviar el paquete perdido.
  - (2.b) Llega el paquete pero se pierde el ACK:  
 $\hookrightarrow$  Pasado el temporizador, el emisor vuelve a enviar el mismo paquete.
  - (2.c) El paquete y el ACK llegan correctamente:  
 $\hookrightarrow$  El ACK contiene un NR  $\leftarrow$  NS + 1. Cuando el receptor lo recibe, envía el paquete con NS  $\leftarrow$  NR.



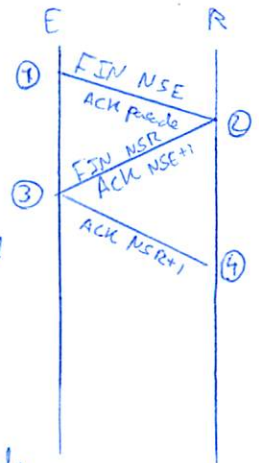
## • Gestión de la conexión

### - Establecer conexión



### - Finalizar conexión

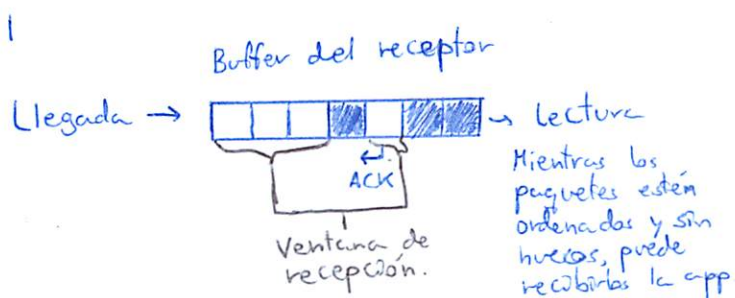
- ① FIN, WAIT-1: solicitud de cierre.
- ② CLOSE-WAIT: envía sol. LAST-ACK de cliente y confirmación del cliente del emisor.
- ③ TIME-WAIT: envía ACK de confirmación de sol. de cliente del receptor.
- ④ CLOSED: conexión cerrada.



## • TCP: Control de flujo o cómo no sobrecargar al receptor

- MSS: máx. n° de bytes que puedo meter en el segmento de datos.
- Ventana de recepción: tamaño del espacio libre en el buffer del receptor.  

$$V. \text{ Recepción} = \text{Buffer de recepción} - \text{N° de bytes no leídos}$$



El control de flujo se encuentra gobernado por el receptor.

**Algoritmo de Karn:**  
 Duplicar el temporizador si se pierde un paquete.  
 (Es más probable que esta vez el paquete haya llegado).

Caso: ventana de recepción = 0.

Emisor:

- ① Alg. Nagle: intentará llegar a un MSS.
- ② Se establece un temporizador persistente. (Muy grande).
- ③ Se establece un temporizador persistente. (Muy grande).
- ④ Emisor envía 1 byte ("Dime algo").
- ⑤ Emisor envía 1 byte de que tiene sitio si cabe un MSS.

Receptor: solo avisa de que tiene sitio si cabe un MSS.

Abir ventana si:  $\begin{cases} \text{tamaño(ventana)} = 1 \text{ MSS} \\ \text{tamaño(ventana)} > \frac{1}{2} \text{ Buffer.} \end{cases}$

## • Establecer Timeout: debe ser mayor que RTT pero no muy grande.

$RTT_E$  = RTT estimado.

$RTT_M$  = RTT promedio calculado a base de muestras empíricas.

$RTT_D$  = Posible desviación del RTT.  
 + tiempo.

TCP estándar:

$\alpha = 1/8$

$\beta = 1/4$

$$RTT_E(t+1) = (1 - \alpha) \cdot RTT_E(t) + \alpha \cdot RTT_M(t+1)$$

$$RTT_D(t+1) = (1 - \beta) \cdot RTT_D(t) + \beta \cdot (RTT_M(t+1) - RTT_E(t+1))$$

$$V. \text{ Timeout} = RTT_E(t+1) + 4 \cdot RTT_D(t+1)$$

Nota:  $\begin{cases} \alpha \\ \beta \end{cases}$  Sirven para dar más o menos peso a las estimaciones previas.  
 $0 \leq \alpha \leq 1$ ;  $0 \leq \beta \leq 1$

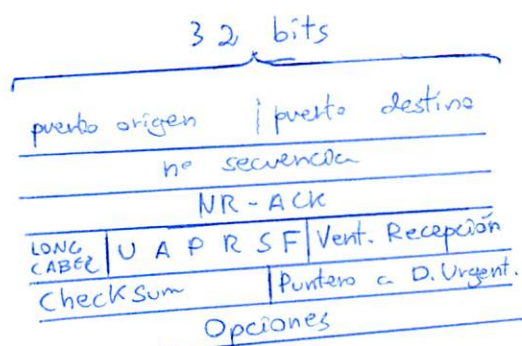


## • TCP: Protocolo de Control de Transmisión.

Proporciona un transporte seguro sobre un canal no seguro.

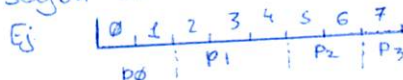
- Orientado a flujo: no hay límites de mensaje.
- Segmentado en buffers de emisión y recepción.
- Control de flujo: el emisor no satura al receptor.  
↳ El receptor le dice al emisor el tam. de la ventana. (Ventana de tam. variable)
- Flujo de datos bidireccional.
- Antes de iniciar un transporte, se intercambian seg. de control que establecen el acuerdo previo a la conexión.

## • Estructura de segmento TCP



- La ventana se mide en nº bytes.

• El NS. no aumenta de uno en uno, sino según el nº bytes de datos.



NSP0 = 0	NRP0 = 2
NSP1 = 2	NRP1 = 5
NSP2 = 5	NRP2 = 7
NSP3 = 7	NRP3 = 8

$$NR(i) = NS(i+1) = NS(i) + Datos(i)$$

Emisor y receptor envían datos simultáneamente.

A ⇒ Indica Ack válido.

R, S, F ⇒ Control Conex.  $\begin{cases} R = \text{reset} \\ S = \text{start connect} \\ F = \text{finish} \end{cases}$

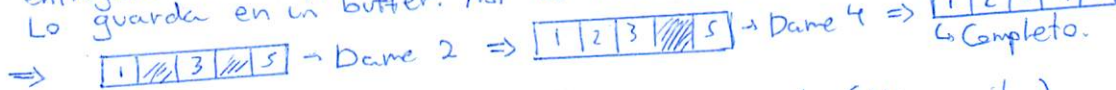
U ⇒ Datos urgentes.

P ⇒ Push. Enviar YA.

## • Eventos de emisión TCP (Mej. Opciones)

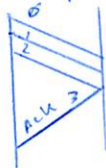
- Salvaguarda: el receptor guardará paquetes desordenados, pero no se lo entregará a la app. hasta que no los tenga todos en orden. Lo guarda en un buffer. Así es más eficiente.

\*timeout\*



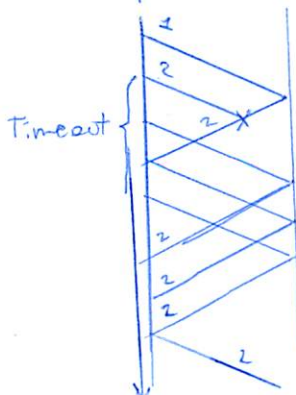
- Time out: enviará sólo 1 paquete si se produce un time out. (Ver arriba)

- Reconocimiento retardado: el receptor no envía inmediatamente el ACK, sino que espera un poquito a ver si llega un segundo paquete y ahorrar un Ack.



- Triple recon. repetido (Retransm. rápida):

Si recibe el emisor 3 ACK's solicitando un paquete que ya ha enviado, lo envía de nuevo inmediatamente a pesar de que no se pase el timeout.





- Control de congestión - TCP

Lo dividiremos en 3 fases:

4 Detección de la congestión: 2 opciones.

- ① Congestión dura: expiran los temporizadores muy seguidamente. RTT muy grande. Esto es debido a que la ventana de recepción está saturada y al no caber los nuevos paquetes, son rechazados y se pierden.
- ② Congestión blanda: <sup>emisor</sup> está perdiendo (deshechando) algún paquete, pero dejando pasar los demás. Hay un poco de congestión, algunos paquetes vuelan y otros no. Por lo tanto, llegan ACK's repetidas.

2) Medidas: tendremos una ventana de congestión y ventana de recepción. la iremos incrementando poco a poco hasta ver a donde llegamos sin que se produzcan síntomas de congestión. Si se detectan estos síntomas, explicados en ① y ②, disminuimos esta ventana de congestión, y luego la aumentamos poco a poco.

### 3 Implementación (mecanismo):

- Supondremos que la ventana de recepción es muy grande. (Así trabajaremos con la ventana de congestión).
- Mediremos en M.S.S.

Tendremos 2 fases:

① Arrange lento:

- ranque lento:
- 1.- Se comienza aumentando exponencialmente la ventana de congestión hasta que se llega a un valor umbral.
  - 2.- Pasado ese valor umbral, el tamaño de la VC aumenta linealmente.
  - 3.- Fase evitación de congestión.

2. Evitación de congestión:

- 1.- Aumentará linealmente el tam. de ventana.
- 2.- Si se detectan síntomas de congestión:

- ### 2.1.- Congestión blanda:

- 1ro)  $VC \leftarrow VC / 2$ ; Se reduce el tamaño de VC.

- 2da) Umbral  $\leftarrow VC$  ; Después, se setea el umbral.

- ## 2.2.- Congestión dura:

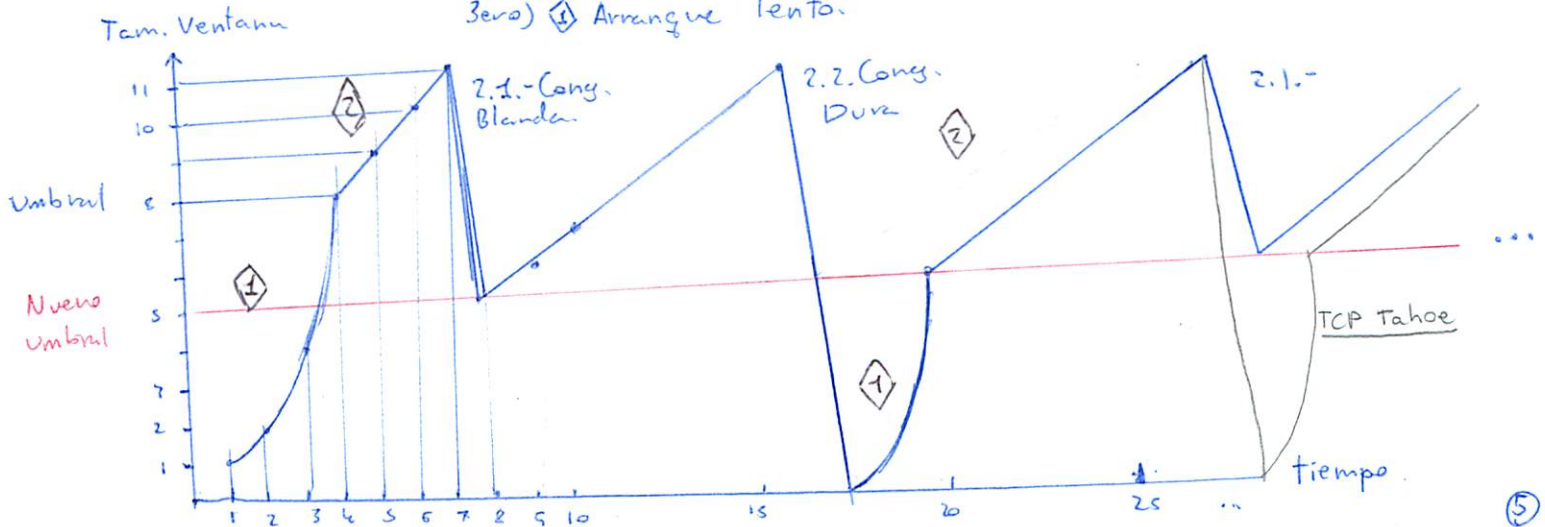
- (ro) Umbral  $\leftarrow VC/2$ ; Nuevo umbral.

- 2do)  $VC \leftarrow 1$  ; Se resetea la ventana de congestión.

- 3ero)  $\boxed{I}$  Arrangue lento.

② TCP Tahoe:  
al detectar ① o ②,  
salta a arrumgue  
lento

- TCP realiza funciones como se describe en **3**



• Control de Congestión TCP - Pt 2.

$w$  = Ventral.

Este  $w$  es el tamaño actual de la ventana

• Vel. Inst. =  $\frac{\text{Nº pkt's que puedo enviar} \cdot \text{tam paquete}}{s}$  =  $\frac{w \cdot MSS}{RTT}$

Tiempo que espero hasta poder volver a enviar

• Vinst. media =  $\bar{V} = \frac{w + (w+1) + (w+2) + \dots + (2w)}{w \cdot RTT} \cdot MSS = \frac{3w^2}{2} \cdot \frac{MSS}{w \cdot RTT} = \frac{3}{2} w \cdot \frac{MSS}{RTT}$

• Tasa pérdidas:

$\frac{\text{Pérdidas}}{\text{Totales}}$

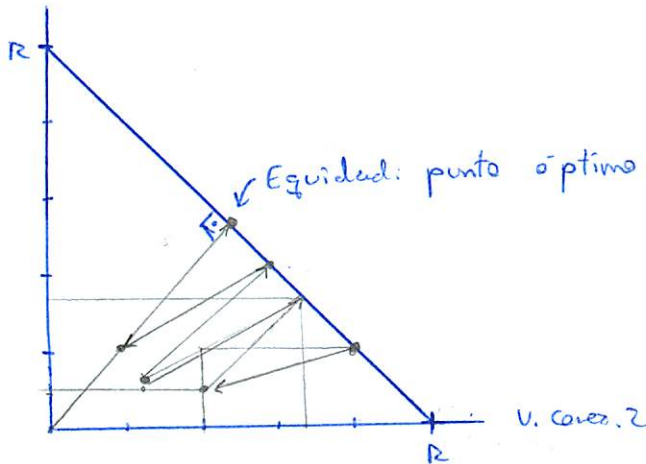
$\bar{L} = \frac{1}{3wc} \Rightarrow w = \sqrt{\frac{2}{3L}}$

$\bar{V} = \frac{3}{2} w \cdot \frac{MSS}{RTT} = \left( \sqrt{\frac{2}{3L}} \right) \cdot \frac{3}{2} \cdot \frac{MSS}{RTT} = \sqrt{\frac{3}{2}} \cdot \frac{MSS}{\sqrt{L} \cdot RTT}$

• Egüidad en TCP

Si existen varias conexiones simultáneas utilizando el mismo ancho de banda, estas tenderán a tener una velocidad de conexión equitativa.

v. conex. 1



## Ejercicios Redes - T3

1 a)  $a = 01011100$   
 $b = 01010110$   


---

 $10110010;$   
 $CAI = 01001101 = \text{Checksum}$   
 $\begin{array}{cc} 4 & 0 \\ \hline 77 \end{array}$

b)  $a = 11011010$   
 $b = 00110110$   


---

 $100010000;$   
 $CAI = 11011111 = \text{Checksum}$

## 2 Protocolo de parada y espera

$T_t$  = tiempo transmisión

$T_f$  = tiempo procesamiento

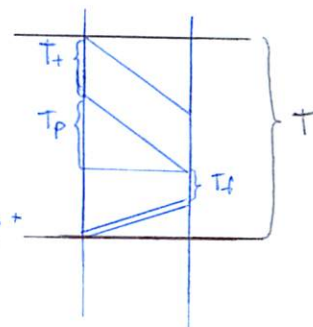
$T_p$  = tiempo de propagación

a) Tiempo esperando  $T$ ?

Sol.:

$$T = 1T_t + 1T_p + 1T_f + 1T_p + 1T_t$$

$$RTT = 2T_p$$



b)  $p_t$  = prob. error PKT datos  
 $p_r$  = prob. error Ack reciv.  
 $q$  = prob. éxito en ambas?  $\left. \begin{array}{l} \\ \end{array} \right\} q = (1-p_t) \cdot (1-p_r)$

c) Demostrar  $E = \frac{1}{q}$

Sol.:

$$E = \underbrace{q \cdot (1-q)^1}_{\text{He tenido éxito tras 1 fracaso}} + \underbrace{q \cdot (1-q)^2}_{\text{He tenido éxito tras 2 fracasos}} + \dots + \underbrace{q \cdot (1-q)^n}_{\text{He tenido éxito tras n fracasos}} = \frac{1}{q}$$

d)  $\left. \begin{array}{l} T_t = 1 \\ T_f = 0,1 \\ T_p = 0,1 \\ p_t, p_r = 10^{-3} \end{array} \right\} \begin{array}{l} q = (1-p_t) \cdot (1-p_r) = 0,998; \\ E = \frac{1}{q} = 1,001002003 \dots \\ V_{\text{real}} = \frac{L \cdot q}{\frac{L}{R} + RTT} = V \cdot q = \frac{L}{T} \cdot 0,998 \end{array}$



③ Se transmiten  $L$  bytes de A a B.  $MSS = 536$  bytes.  $NS = 4$  bytes

a) Valor mín  $L$  que provoca rep. de NS's?

R:

$$\frac{L}{MSS} > 2^{32}; L > 2^{32} \cdot MSS \Rightarrow L > 2^{32} \cdot 536 \cdot 10^3 \approx 2^{31} \cdot 10^{12}$$

b) Header: 66 bytes.

¿t mínima para transmitir todo a través de 155 Mb/s? Despreciando ACK's.

$RTT = 2T_p$ ;  $R = \text{bytes/s}$

$N = n^\circ$  paquetes necesarios =  $\frac{L}{MSS} + 1$ .  
 Resto de paquetes completos:  $\rightarrow$  ult. paquete, quizás no completo.

$$T = (N-1) \cdot \left[ RTT + \frac{MSS + H}{R} \right] + 1 \cdot \left[ RTT + \frac{L \% MSS + H}{R} \right]$$

$$V = \frac{L}{T} = \frac{L}{(N-1) \cdot \left[ RTT + \frac{MSS + H}{R} \right] + \left[ RTT + \frac{L \% MSS + H}{R} \right]}$$

$$V = \frac{L}{T}; T = L / V = 2^{31} \cdot 10^{12} + \left( \frac{2^{31} \cdot 10^{12}}{536} \cdot 66 \right) \parallel 155 \cdot 2^{20} = 15962191.5$$

④ Protocolo GBN.

Tiempo  $t \Rightarrow NS = K$

NS ventana?

$$N \begin{cases} NS_{p_1} = K \\ NS_{p_2} = K+1 \\ NS_{p_3} = K+2 \\ \vdots \\ NS_{p_n} = K+N \end{cases}$$

⑤ GBN.

$N = t.$  ventana

$R = \text{rango.}$

$N < R$

En tiempo  $t \Rightarrow$  Receptor espera  $NS = S$ ;

a) Pos. rangos valores

$$\text{Ventana} \begin{cases} NS_{p_1} = S = R - N \\ NS_{p_2} = S+1 = R - N + 1 \\ \vdots \\ NS_{p_N} = S+N = R \end{cases}$$

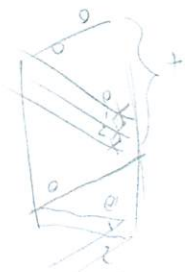
b) Rango pos. de los ACK's. =  $R + 1$ .

Es decir, si el rango del emisor era de 0 a 5  $\Rightarrow [0, 1, 2, 3, 4, 5]$

Ahora se sumará 1 a cada valor, por ser un reconocimiento que recibe el siguiente.  $\Rightarrow [1, 2, 3, 4, 5, 6]$

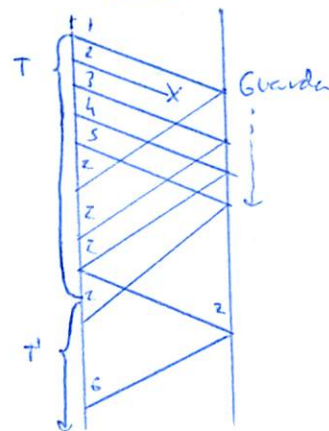
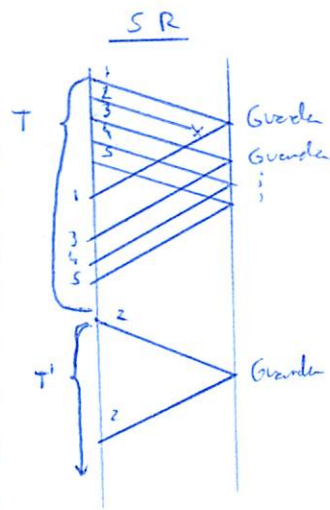
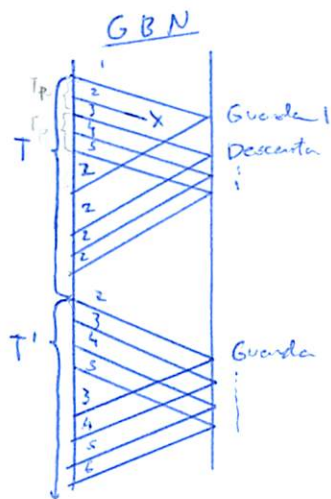
⑥ Suponiendo  $NS$  en mod  $N$ .

Si: Rango  $(N) = 3$ , Posibles  $NS = 0, 1, 2$ .



• Ejercicios Redes - Cap. 3 - Pt II

7) A → B 5 segmentos. El 2do se pierde. Comparar GBN, SR, TCP/IP. (Supongo triple confirm. rep.)

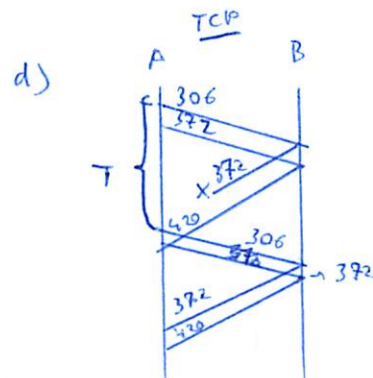
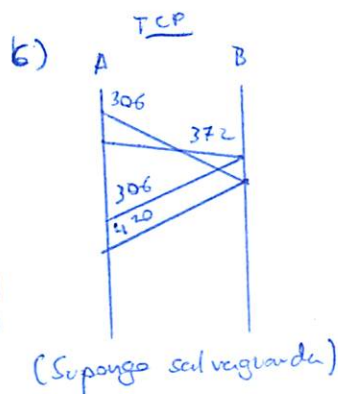
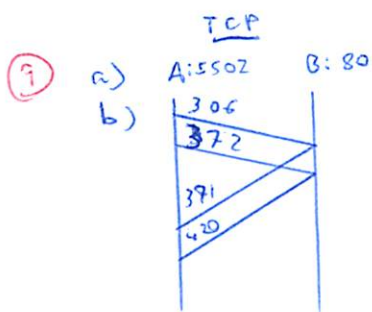


8)  $P = n^{\circ}$  paquetes neces. a enviar =  $\left\lceil \frac{F}{MSS} \right\rceil$ ;  $RTT = 2Tr.$

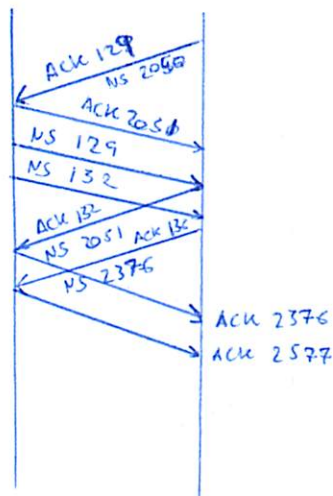
$$T = \left( \frac{P \cdot MSS}{N} \right) \cdot RTT + \frac{MSS}{R/N}; \quad T = \frac{PKT's \text{ necesarios}}{\text{Tam Ventana}} \cdot \left[ RTT \text{ de segmento} + \left( \frac{\text{Tam. PKT}}{\text{Ancho de banda}} \right) \cdot \frac{\text{Tam Ventana}}{\text{Tam Ventana}} \right]$$

$$T = \left( \frac{P}{N} \right) \cdot \left( RTT + \frac{MSS}{R/N} \right)$$

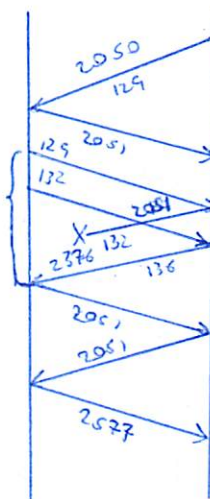
$$Vel. = \frac{\text{T. Fichero}}{\text{Temp rec.}} = \frac{P}{N} \cdot \left( RTT + \frac{MSS}{R/N} \right)$$



10 a)



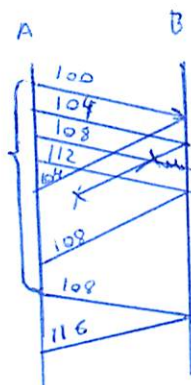
b)



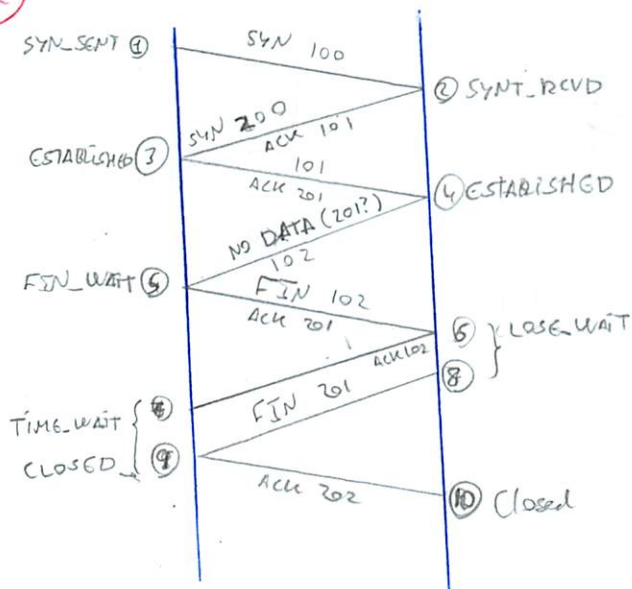
\*  
NS  
ACK  
NS  
ACK

c) Si RTT = 2 ms, ¿Vel. trnsf. A? ¿y B?

11



12



- ① Solicitud de conexión.
- ② Sol. recibida. Sol. conexión.
- ③ y ④ Establecida.
- ⑤ Solicitud de cierre.
- ⑥ Solic. recibida. Sol. cierre y last ACK.
- ⑦ Solicitud recb. ⑧ last ACK.
- ⑨ Conexión cerrada.



## Ejercicios Redes - Capítulo 3 - Parte III

- (13)  $V = R$  bytes/s  
 Datos = 15 MSS  
 $\Rightarrow$  Arranque lento.

a)  $4 \frac{\text{MSS}}{R} \gg \frac{\text{MSS}}{R} + \text{RTT} > 2 \frac{\text{MSS}}{R}$

$$\left( \frac{1 \text{ MSS}}{R} + \text{RTT} \right) + \left( 2 \frac{\text{MSS}}{R} + \text{RTT} \right) + \left( 4 \frac{\text{MSS}}{R} + \text{RTT} \right) + \left( 8 \frac{\text{MSS}}{R} + \text{RTT} \right)$$

- (14) Expresar  $\text{RTT}_E$  en función de  $\text{RTT}_H$ .

$$\text{RTT}_E(t+1) = \frac{1-\alpha}{N} \cdot \text{RTT}_E(t) + \alpha \cdot \text{RTT}_H(t)$$

$$\hookrightarrow \frac{\sum \text{RTT}_E(0 \dots t+1)}{N+1}$$

- (16)  $A \rightarrow B, \Rightarrow \text{TCP}$ . Evitación de congestión.  $2W = 60 \text{ MSS};$   
 $\text{MSS} = 1460 \text{ bytes}$   
 $\text{RTT} = 50 \text{ ms}$ .

a) Vel. transf. prom. =  $\frac{3}{2} W \cdot \frac{\text{MSS}}{\text{RTT}} = \frac{3}{2} \cdot \frac{60}{2} \cdot \frac{1460}{50 \cdot 10^{-3}} = 1314 \cdot 10^3 \text{ B/s}$

b)  $\bar{t}_{\text{uso}} = \frac{\left( \frac{\bar{W} \cdot \text{tamaño seg.}}{\text{velocidad}} \right)}{\text{RTT}} = \frac{\left( \frac{\frac{3}{2} W \cdot (\text{MSS} + H)}{R_t} \right)}{\text{RTT}} = \frac{\left( \frac{3/2 W \cdot (1460 + 66)}{10^8} \right)}{50 \cdot 10^{-3}}$

$\hookrightarrow = 0'013734; \text{ ó } \bar{t}_{\text{uso}} = \frac{\text{vel real media}}{\text{apto banda}} = \frac{3/2 W \cdot \frac{\text{MSS} + 66}{\text{RTT}}}{100 \cdot 2^{20}} = 0'0139 \dots$

16) a) 3 MSS  $\rightarrow$  4 MSS  $\rightarrow$  5 MSS  $\dots \rightarrow$  24 MSS, (cada RTT incrementa en 1.  
Tardará  $24 - 3 = 21$  RTT. seg.

b)  $V_{tr. 24 MSS} - V_{tr. 3 MSS} = \frac{3}{2} W \cdot \frac{MSS}{RTT} \Rightarrow \left[ \frac{3}{2} \cdot 24 \cdot \frac{MSS}{RTT} \right] - \left[ \frac{3}{2} \cdot 3 \cdot \frac{MSS}{RTT} \right] =$

$$\boxed{\frac{3}{2} \cdot 21 \cdot \frac{MSS}{RTT}}$$

17) A  $\xrightarrow{2'5 s.}$  B 100 Mb; RTT = 3ms. MTU = 1500 bytes.  
 $\hookrightarrow MTU = \underbrace{MSS}_{1462} + \underbrace{Cabeceras}_{38}$

a)  $V_{media} = B/s; 100 Mb / 2'5 s \Rightarrow 41'94 \cdot 10^6 B/s$

$$41'94 \cdot 10^6 = \frac{3}{2} W \cdot \frac{MSS}{RTT} = \frac{3}{2} W \cdot \frac{1462 + 38}{3 \cdot 10^{-3}} = \frac{1500000}{2} W;$$

b)  $\left\{ \begin{array}{l} W \approx 55'92 \approx 60 \text{ MSS de } \text{umbra} \text{ m/mo.} \\ \text{El m\u00e1ximo ha sido } 120 \text{ MSS.} \\ \text{La media es de } 80 \text{ MSS.} \end{array} \right.$

c) Cada 60 RTT's  $= 60 \cdot 3 \cdot 10^{-3} = 0'18 \text{ seg.}$

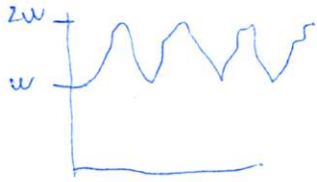
d) ~~2'5 seg.~~  $60 + 61 + 62 \dots 120 = 180 \cdot \frac{(120 - 60)}{2} = 5400 \text{ PKT's} + 90 = 5490 \text{ PKT's}$

(19) TCP Tahoe: siempre salta a arranque lento:

a) Exp. vel.

$$v = \frac{2W}{2RTT}$$

$$v = \frac{W}{2RTT}$$



$$\text{Three times: media} = \frac{0.75 W}{RTT} = \frac{3}{4} \frac{W}{RTT} \cdot MSS$$

$$\bar{V} = \frac{3}{4} W \cdot \frac{MSS}{RTT}$$

$$L = \frac{1}{\frac{3W^2}{4}} = \frac{4}{3W^2} = \frac{1}{L} ; \sqrt{\frac{4}{3L}} = W = \sqrt{\frac{4}{3L}}$$

$$\bar{V} = \frac{3}{4} W \cdot \frac{MSS}{RTT} = \sqrt{\frac{4}{3L}} \cdot \frac{3}{4} \cdot \frac{MSS}{RTT}$$

$$\sqrt{\frac{4}{3L}} \cdot \sqrt{\frac{9}{16}} \cdot \frac{MSS}{RTT}$$

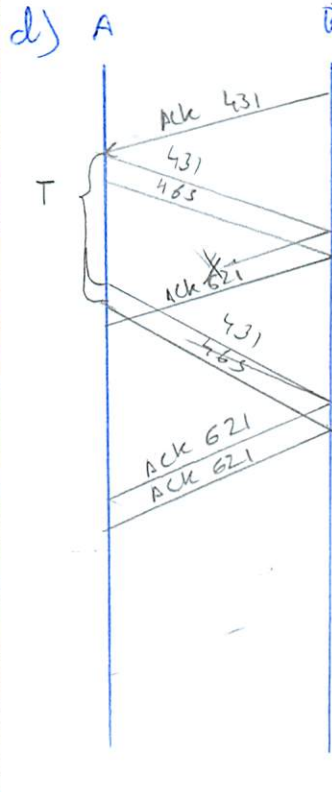
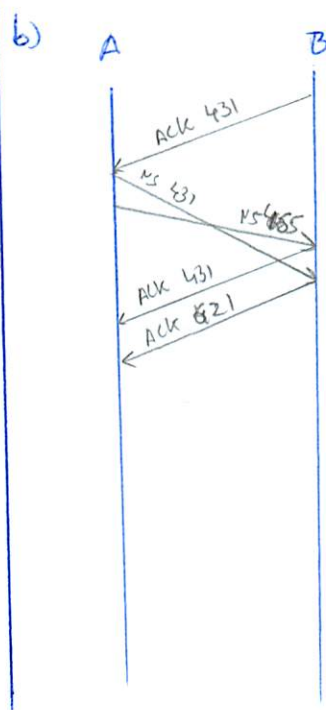
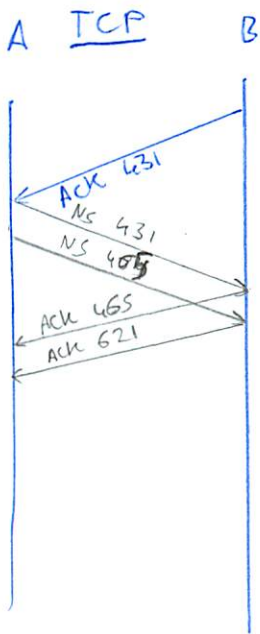
$$\sqrt{\frac{36}{48L}} \cdot \frac{MSS}{RTT} = \frac{\sqrt{3}}{2} \cdot \frac{MSS}{\sqrt{L} \cdot RTT}$$

$$\sqrt{\frac{2}{3}} \cdot \sqrt{\frac{1}{2}} = \sqrt{\frac{2}{6}}$$



• G. Redes Pt IV

(2) Rec



(18)

$$\frac{w}{2} + \frac{w}{2^2} + \dots + w = \frac{w^2}{4} + \frac{w}{2} + \frac{w^2}{8} + \frac{w}{4} = \frac{3}{8} w^2 + \frac{3}{4} w$$

$$L = \frac{1}{2w^3} = \frac{1}{\frac{3}{8} w^2 + \frac{3}{4} w} = L \quad \frac{1}{4} x^2 + \frac{1}{8} x^2 + \frac{1}{2} x + \frac{1}{4} x$$

$$V = \frac{1}{2} w \quad w = \sqrt{\frac{8}{3L}}$$

$$V = \frac{3}{4} w \quad \frac{MSS}{RTT} \Rightarrow \frac{3}{4} \cdot \sqrt{\frac{8}{3L}} \cdot \frac{MSS}{RTT} = \frac{\sqrt{6}}{2} \cdot \frac{MSS}{\sqrt{3LRTT}}$$

$$\frac{w}{2} + \frac{w}{2} + \dots + w = \frac{1}{8} w^2 + \frac{1}{4} w^2 + \frac{1}{2} w + \frac{1}{2} w$$

$$\bar{V} = \frac{3}{4}$$

## Preguntas a Raúl

⇒ ¿Los puertos cambian alguna vez en TCP? No cambian

⑥ Tam Ventana ns, ms?

NS para lo error?

NS rango NS = (0,  $\text{TVentana} + 1$ )

⑦ Checksum a veces + Suma a veces

⑧ ¿Estos bien?

⑩ c)

13



TCP Reno vs Tahoe ??

Reno: tripleAck rep.

Tahoe: siempre saltar a ventana 1

A continuación figuran una serie de ejercicios propuestos, con su calificación correspondiente. Se proponen un ejercicio de 1 punto, dos de 2 puntos y uno de 3 puntos, de entre los cuales el alumno elegirá los que prefiera, para optar al máximo de 4 puntos.

Para cada uno de los ejercicios se exige cumplir las especificaciones completamente. Para aquellos detalles del ejercicio que no se especifiquen en el enunciado, se admite la decisión justificada del alumno. En general se deberán utilizar los espacios y alimentación de línea necesarios para facilitar la lectura tanto de datos como de resultados.

Para todos los ejercicios, y salvo que se especifique lo contrario en el propio enunciado, utilizar la entrada estándar para obtener los datos, y la salida estándar para mostrar los resultados.

1. **(1 punto)** Crear un programa que obtenga del usuario una matriz de enteros dimensión arbitraria (pero inferior a 10 filas y 10 columnas), y a continuación obtenga del usuario un entero positivo. El programa mostrará por pantalla el resultado de eliminar de la matriz la fila y la columna correspondientes al entero que se obtuvo del usuario.

Ejemplo:

Matriz original:

```
1 3 2 5
2 2 3 0
7 2 4 1
4 1 4 0
```

Fila y columna a eliminar: 2

Matriz resultado:

```
1 3 5
2 2 0
4 1 0
```

**Notas:**

a) Recuérdese que filas y columnas comienzan a contar en 0.

2. **(2 puntos)** Escribir un programa que informe de la proporción de aparición de cada letra del alfabeto con respecto al total de letras que se introduzcan. El programa leerá de teclado una cadena de caracteres arbitrariamente larga que finalice con el carácter fin de fichero, EOF (en Linux, Control-D si la entrada es por pantalla). Solo se mostrará la proporción de las letras presentes.

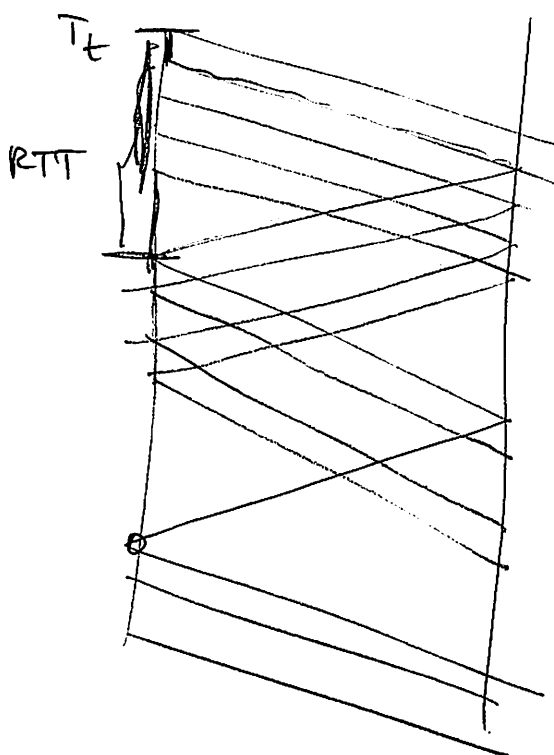
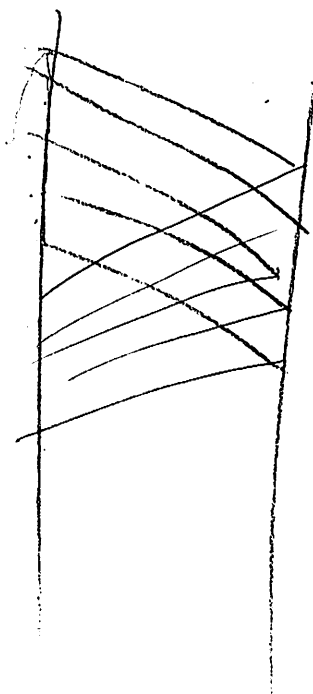
En el informe, las letras pueden ir en orden cualquiera. El aspecto del informe puede ser, por ejemplo, así:

```
e: 32 veces, 23%
d: 29 veces, 21%
...
```

**Notas:**

- a) No hay que distinguir entre mayúsculas y minúsculas: por ejemplo, si aparece una 'A' y una 'a', debe contarse como que la 'a' aparece dos veces.
- b) El programa debe considerar que la cadena de entrada puede ser arbitrariamente larga y finaliza cuando el usuario proporciona el carácter de fin de fichero, EOF (si la entrada es por teclado, hay que pulsar Control-D).





$$\bar{V} = \frac{\textcircled{N} \text{ MSS}}{T_t + \text{RTT}}$$