

Exercicios Extra

- Supoñamos unha mensaxe de 10^7 bits que se quere enviar entre dous hosts entre os que hai 2 routers intermedios. Supoñamos que os enlaces son de 2 Mbps (millóns de bits por segundo). Obter o tempo de transmisión nos dous casos seguintes:

- A mensaxe sen segmentar
- Segmentada en 2500 paquetes de 4000 bits cada unha

Considerar que os routers son dispositivos de almacenamento e reenvío. Utilizar diagramas de tempo.

Vamos ter en conta so o tempo de almacenamento e reenvío. Sabendo que temos enlaces de 2 Mbps, os tempos necesarios para almacenar e reenviar a mensaxe completa e un dos paquetes son, respectivamente,

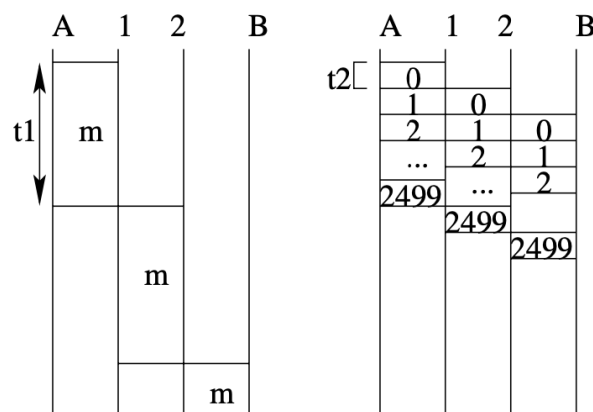
$$t_1 = \frac{10^7 \text{ bits}}{2 \text{ Mbps}} = 5 \text{ sg} \quad (1)$$

$$t_2 = \frac{4000 \text{ bits}}{2 \text{ Mbps}} = 2 \text{ msg} \quad (2)$$

Tendo en conta o esquema da seguinte figura, os retardos de transmisión nos dous casos, son:

$$t_{\text{mensaxe}} = 3t_1 = 15 \text{ sg} \quad (3)$$

$$t_{\text{paquetes}} = 2500t_2 + 2t_2 = 5 + 0,004 = 5,004 \text{ sg} \quad (4)$$



Se supoñemos que aos paquetes se lles engade unha cabeceira de 40 bytes (a cabeceira IP é de 20 bytes de tamaño mínimo e 60 bytes de tamaño máximo), cada paquete terá 4320 bits e o resultado será:

$$t_2 = \frac{4320 \text{ bits}}{2 \text{ Mbps}} = 2,16 \text{ msg} \quad (5)$$

$$t_{\text{paquetes}} = 2500t_2 + 2t_2 = 5,400 + 0,00432 = 5,40432 \text{ sg} \quad (6)$$

2. Tes que diseñar un protocolo de capa de transporte que usa fiestra deslizante de forma similar a como o fai TCP. Este protocolo vai ser usado sobre unha rede a 100 Mbps. O retardo da rede é de 100 ms, e o tempo de vida máximo dun segmento (MSL) é de 60 segundos. Se queremos aproveitar a capacidade da rede e que non se superpoñan os números de secuencia, cantos bits, como mínimo, deberías utilizar para os campos *Número de secuencia* (SN) e *Fiestra otorgada* (AW) da cabeceira do teu protocolo?

Solución para o SN:

Para calcular o tamaño do campo SN, temos que ter en conta que os números de secuencia non deben dar a volta en menos do MSL (*Maximum Segment Lifetime*), que é 60 s. En 60 s, a 100 Mbps transmitense 6.000 Mbits, que son $750 \cdot 10^6$ bytes. Para representar ese número de bytes, necesitamos polo menos 30 bits, xa que

$$2^{29} = 536.870.912 < 750 \cdot 10^6 < 2^{30} = 1.073.741.824, \implies \text{SN de 30 bits.}$$

Solución para a AW:

Se queremos aproveitar ao máximo a capacidade da rede, a fiestra otorgada debe se-lo suficientemente grande como para poder enviar información mentras non chegue o primeiro ACK. Desta forma, durante un RTT poderíamos enviar os bytes:

$$N\text{Bytes} = RTT \times BW = 2 \cdot 100 \text{ ms} \cdot 100 \text{ Mbps} = 20 \text{ Mbits} = 2.500.000 \text{ bytes.}$$

Para indicar unha fiestra dese tamaño necesitamos 22 bits, xa que

$$2^{21} = 2.097.152 < 2.500.000 < 2^{22} = 4.194.304, \implies \text{AW de 22 bits.}$$

Hai que verificar que o rango de números de secuencia sexa o dobre do tamaño da fiestra, que neste caso se cumpre sobradamente.

Outra solución:

A fiestra otorgada debe se-lo suficientemente grande como para poder encher a capacidade da rede. O produto retardo \times ancho de banda é:

$$R \times BW = 100 \text{ ms} \cdot 100 \text{ Mbps} = 10 \text{ Mbits} = 1.250.000 \text{ bytes.}$$

Para indicar unha fiestra dese tamaño necesitamos 21 bits, xa que

$$2^{20} = 1.048.576 < 1.250.000 < 2^{21} = 2.097.152, \implies \text{AW de 21 bits.}$$

Hai que verificar que o rango de números de secuencia sexa o dobre do tamaño da fiestra, que neste caso se cumpre sobradamente.

3. Consideremos unha única conexión TCP (Reno, con recuperación rápida) que emprega un enlace a 10 Mbps, que é o único entre os hosts emisor e receptor. Supoñamos que se quere enviar un arquivo de gran tamaño e que o buffer de recepción é moito maior que a xanela de conxestión. O tamaño de segmento TCP é de 1.500 bytes, o RTT é de 100 milisegundos e nunca se perde un segmento por expiración do temporizador.

- a) Cal é o tamaño máximo de xanela (en segmentos) que esta conexión TCP pode alcanzar?

Sexa W o tamaño máximo de fiestra en segmentos. Entón,

$$\frac{W \times \text{MSS}}{\text{RTT}} = 10 \text{ Mbps}$$

xa que se descartarán paquetes se a taxa máxima de envío supera a capacidade do enlace. Entón, temos que

$$\frac{W \times 1500 \times 8}{0,1} = 10^7 \text{ bps} \implies W = 83,3 \Rightarrow 84 \text{ segmentos}$$

- b) Cales son o tamaño medio de xanela (en segmentos) e a taxa de transferencia media (en bps) desta conexión?

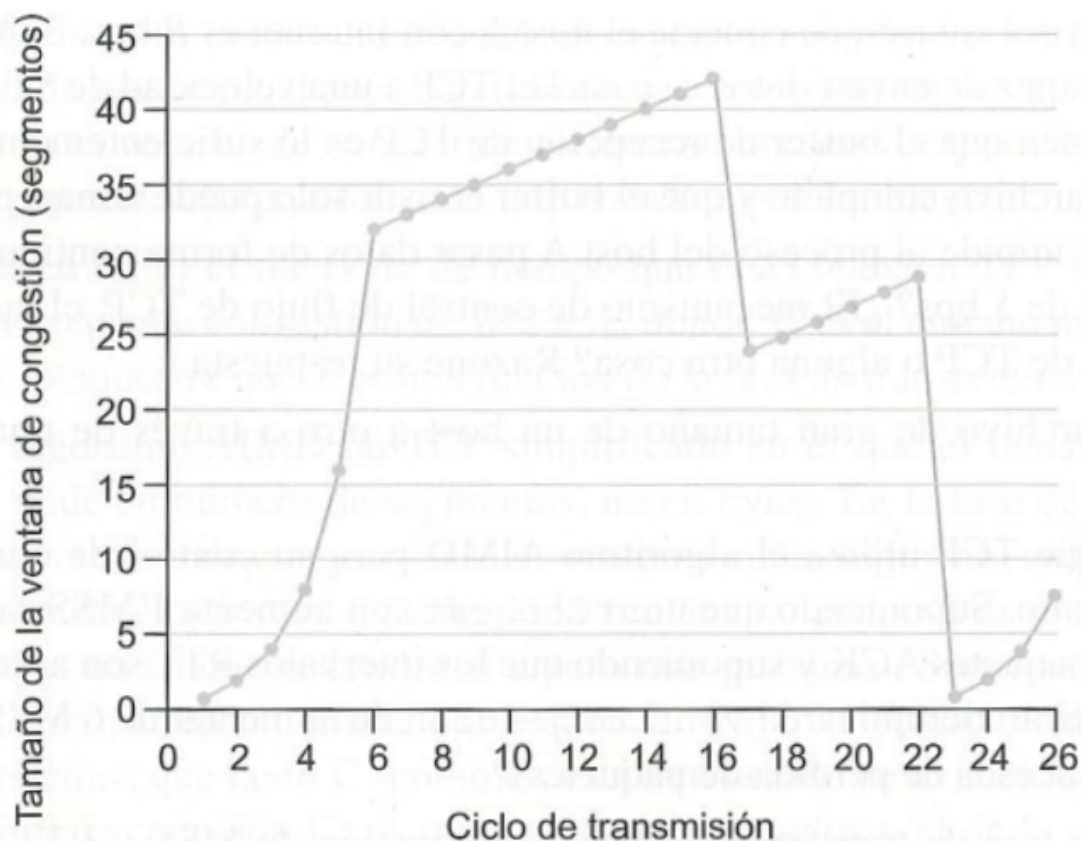
Como a fiestra de conexión W varía desde $W/2$ a W , o tamaño medio da xanela de conexión é $0,75W = 63$ segmentos. A taxa de transferencia media é

$$\frac{63 \times 1500 \times 8}{0,1} = 7,56 \text{ Mbps}$$

- c) Asumendo que a xanela de conexión está no seu valor máximo, canto tarda esta conexión TCP en alcanzar de novo o seu tamaño de xanela máximo despois de recuperarse dunha perda de paquete?

Cando se perde un paquete (3 ACKs duplicados) a ventana pasa a $W/2 + 3$. Para chegar ata W , a razón de 1 segmento por RTT, necesítanse $(W/2 - 3)$ RTTs. Polo tanto,

$$\left(W - \left(\frac{W}{2} + 3\right)\right) \cdot \text{RTT} = \left(\frac{W}{2} - 3\right) \cdot \text{RTT} = \left(\frac{84}{2} - 3\right) \times 0,1 = 3,9 \text{ segundos}$$



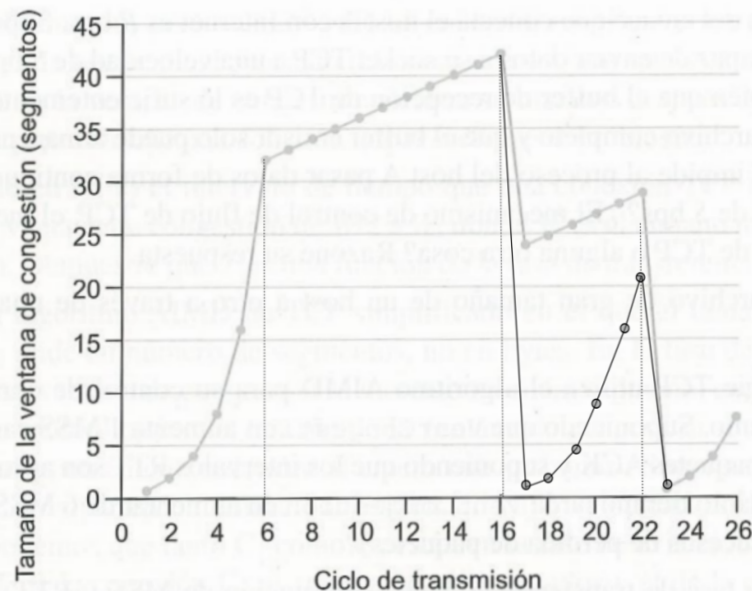
4. Na figura mostrase o comportamento de TCP Reno. Responder ás seguintes preguntas, razoando brevemente as respostas.

- d) Al final do ciclo 22, detectase perda de segmento mediante 3 ACKs duplicados ou mediante un fin de temporización? Un fin de temporización e por iso a CW pasa a 1 e segue con inicio lento.
- e) Cal é o valor inicial de **umbral**? É 32, xa que termina a fase de inicio lento e pasase a AIMD.
- f) Cal é o valor de **umbral** transcurridos 18 ciclos? O **umbral** fíxase é metade da xanela de conxestión cando se detecta a perda do paquete. No ciclo 16 a CW é 42, polo tanto, no ciclo 18 **umbral** é 21 (no 17 pasa a ser 21). A CW é 24 (21 + 3).
- g) Cal é o valor de **umbral** transcurridos 24 ciclos? O **umbral** fíxase á metade da xanela de conxestión cando se detecta a perda do paquete. No ciclo 22 a CW é 29, polo tanto, no ciclo 24 **umbral** é 14 (no 23 pasa a ser 14).
- h) Durante que ciclo envíase o segmento 100? Durante o primeiro ciclo envíase o segmento 1; no segundo os segmentos 2 e 3; no terceiro os segmentos 4-7; no cuarto os segmentos 8-15; no quinto os segmentos 16-31; no sexto os segmentos 32-63; no sétimo envíanse os segmentos 64-96 (so 33, xa que está en AIMD); no octavo envíanse 34 segmentos, do 97 en diante. Polo tanto, o segmento 100 envíasea durante o ciclo 8.
- i) Supoñendo que se detecta unha perda de paquete despois do ciclo 26 a causa da recepción de 3 ACKs duplicados, cales serán os valores da xanela de conxestión e **umbral**? O **umbral** fíxase á metade da xanela de conxestión cando se detecta a perda do paquete. A xanela de conxestión fíxase a ese novo **umbral** + 3 MSS. No

ciclo 26 a CW é 8, polo tanto, no ciclo 26 o **umbral** pasa a ser 4 e a CW 7 e segue con AIMD.

- j) Supón que se utiliza TCP Tahoe (non ten recuperación rápida) e que se recibiron 3 ACKs duplicados no ciclo 16. Cales serán os valores da xanela de conxestión e **umbral** no ciclo 19? Como non tne recuperación rápida, no ciclo 17 a CW pasa a valer 1 e o **umbral** pasa á metade da CW cando se detecta a pérdida do paquete, que é 21. A partires de ahí pasase á fase de inicio lento, polo tanto, no ciclo 19 CW vale 4 e o **umbral** 21.
- k) Supón que se utiliza TCP Tahoe e que se produce un fin de temporización no ciclo 22. Cantos segmentos foron enviados desde o ciclo 17 ao 22, ambos incluídos? Como non ten recuperación rápida, no ciclo 17 a CW pasa a valer 1, polo tanto envíase 1 segmento no ciclo 17; 2 segmentos no 18; 4 segmentos no 19; 8 segmentos no 20; 16 segmentos no 21 e 21 segmentos no ciclo 22 (**umbral** vale 21 e despois pasárase a AIMD se non houberse pérdidas). En total 52 segmentos.

A figura mostra como sería con TCP Tahoe.



5. Mostra e calcula, utilizando diagramas de liñas de tempo, o tempo de transferencia dunha páxina web con dúas imaxes (documento base e dous obxectos) nos diferentes tipos de conexións que permite o protocolo HTTP. Supón que esta páxina é primeira que se solicita ao servidor web.

- a) Conexións non persistentes

- Serie
- Paralelo

- b) Conexións persistentes

- Sen entubamento
- Con entubamento

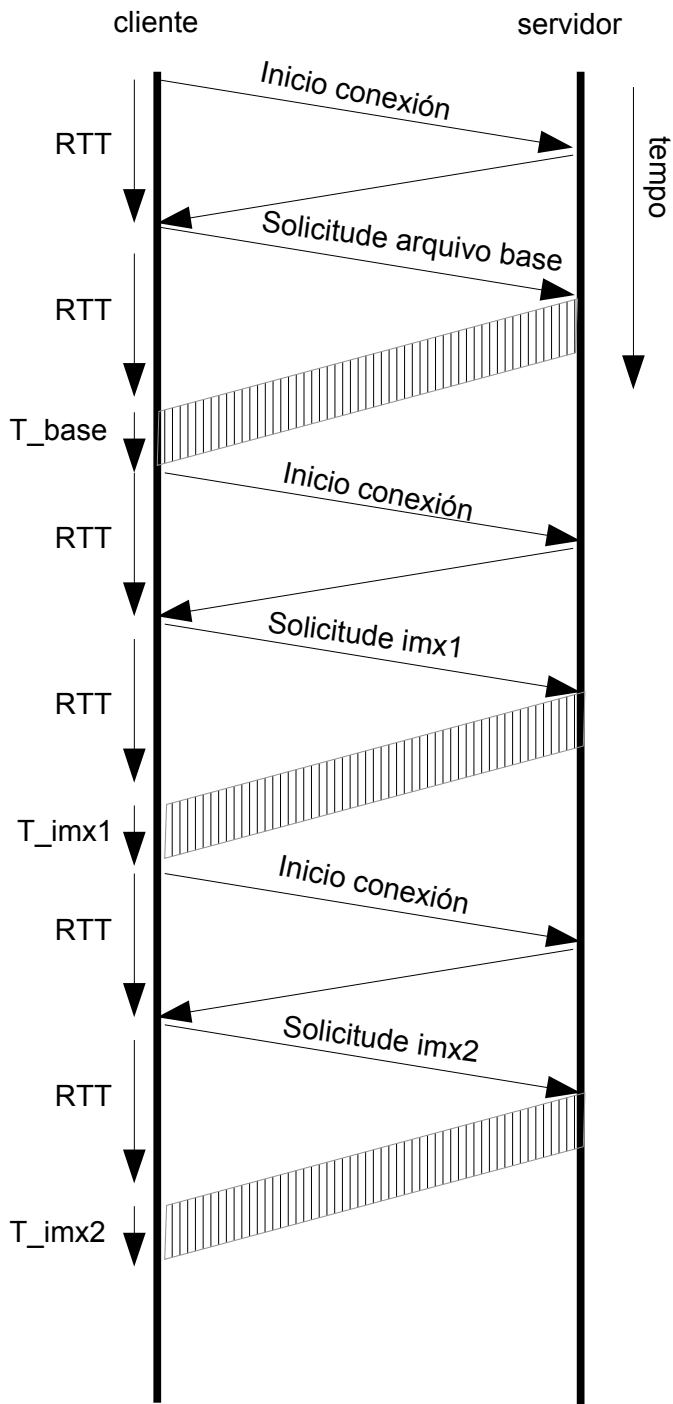
Sobre o tempo de desconexión: Desde o punto de vista do protocolo HTTP pódese considerar o seguinte nas desconexións: cando o servidor remata de enviar todos os arquivos (ou 1 arquivo, en conexións non persistentes) mada a mensaxe de peche de conexión inmediatamente. Cando o cliente recibe o arquivo ao final recibe a mensaxe de desconexión e pecha a conexión. Desta forma a fase de desconexión queda oculta dentro do tempo

de transferencia do arquivo e non se ten en conta. Esta é a forma en que está explicado no libro é moitos exercicios deste. Desde o punto de vista do protocolo TCP non é tan sinxelo. Pódese considerar que ao recibir o cliente o último segmento do arquivo envíase un ACK ao servidor para confirmar, e é entón o servidor é cando desconecta e envía esa mensaxe de desconexión (non o pode facer antes de ter todos os ACK). Despois o cliente responde a esa mensaxe co seu ACK de desconexión e inmediatamente (polo que non hai retardo, o tempo é despreziable), porque non precisa confirmación, pode abrir unha conexión nova (son dúas mensaxe pero se fan case ao mesmo tempo).

A diferenza entre as dúas opcións é considerar o protocolo TCP que vai por debaixo explícitamente ou non. No primeiro caso o tempo de desconexión se incúe no tempo de transferencia, no segundo é explícito. Como en realidade no segundo caso non se teñen en conta cantos segmento hai e os seus ACK, senón que se mete todo como tempo de transferencia ten sentido meter aí tamén a desconexión. Na práctica todo isto está moi optimizado e se superpoñen moitas cousas, polo que non ten moitísima importancia.

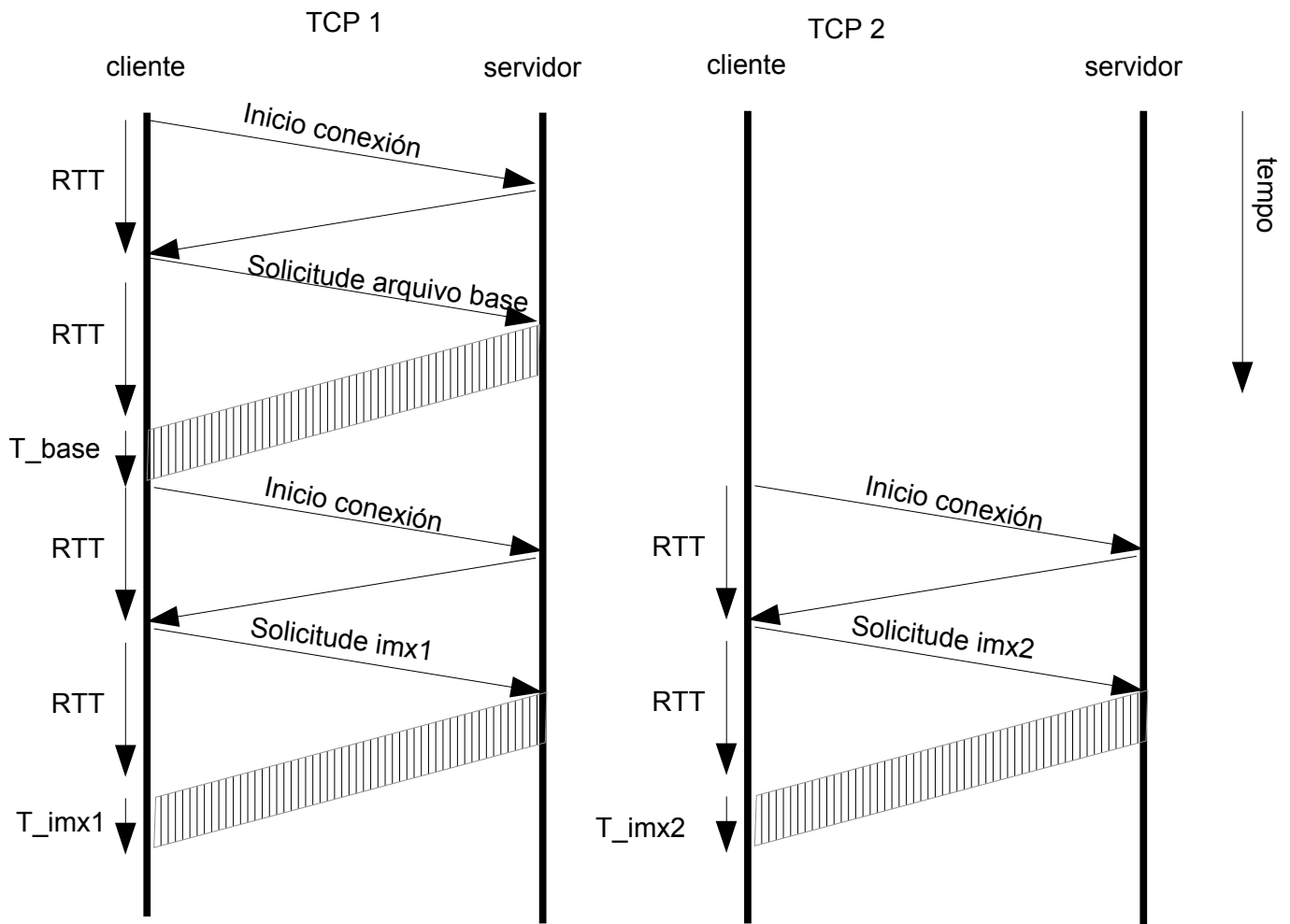
No exame e exercicios podedes usar o que queirades, sen dar explicacións.

Non Persistente - Serie



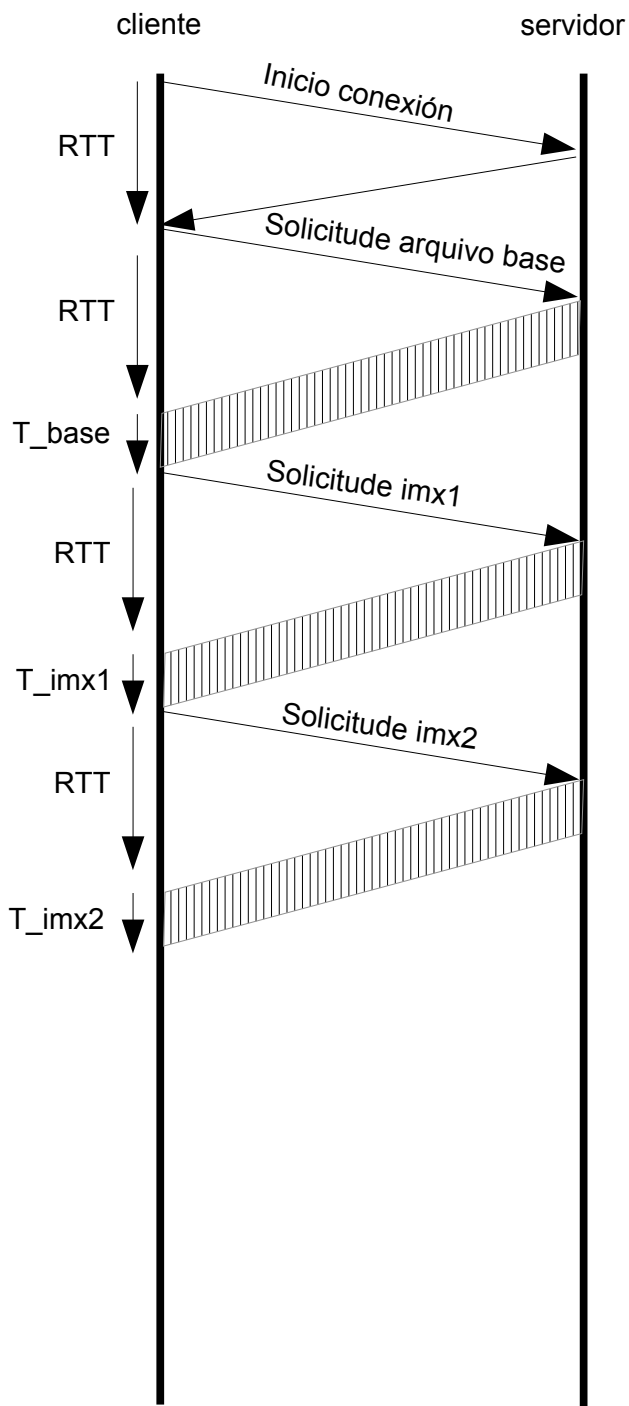
$$\text{Tempo total} = 6 \text{ RTT} + T_{base} + T_{imx1} + T_{imx2}$$

Non Persistente - Paralelo

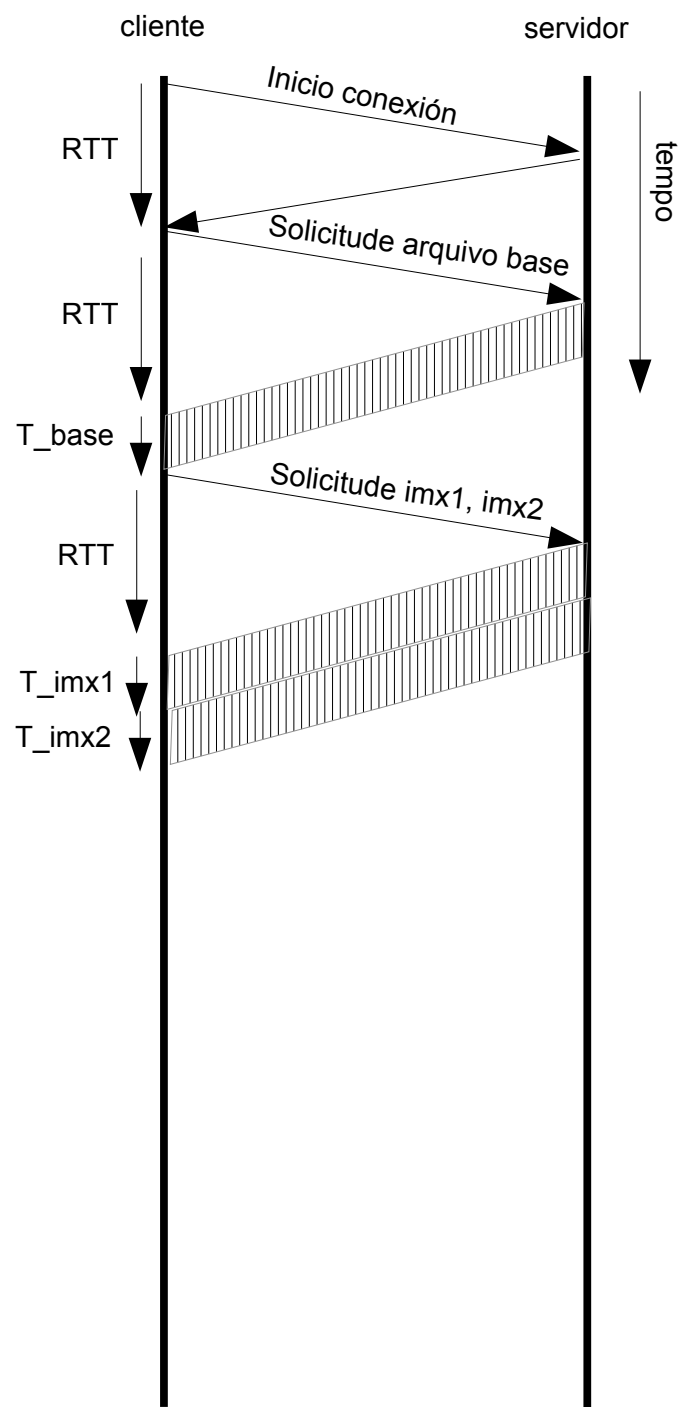


$$\text{Tempo total} = 4 \text{ RTT} + T_{\text{base}} + \max(T_{\text{imx1}} + T_{\text{imx2}})$$

Persistente – Sen entubamento



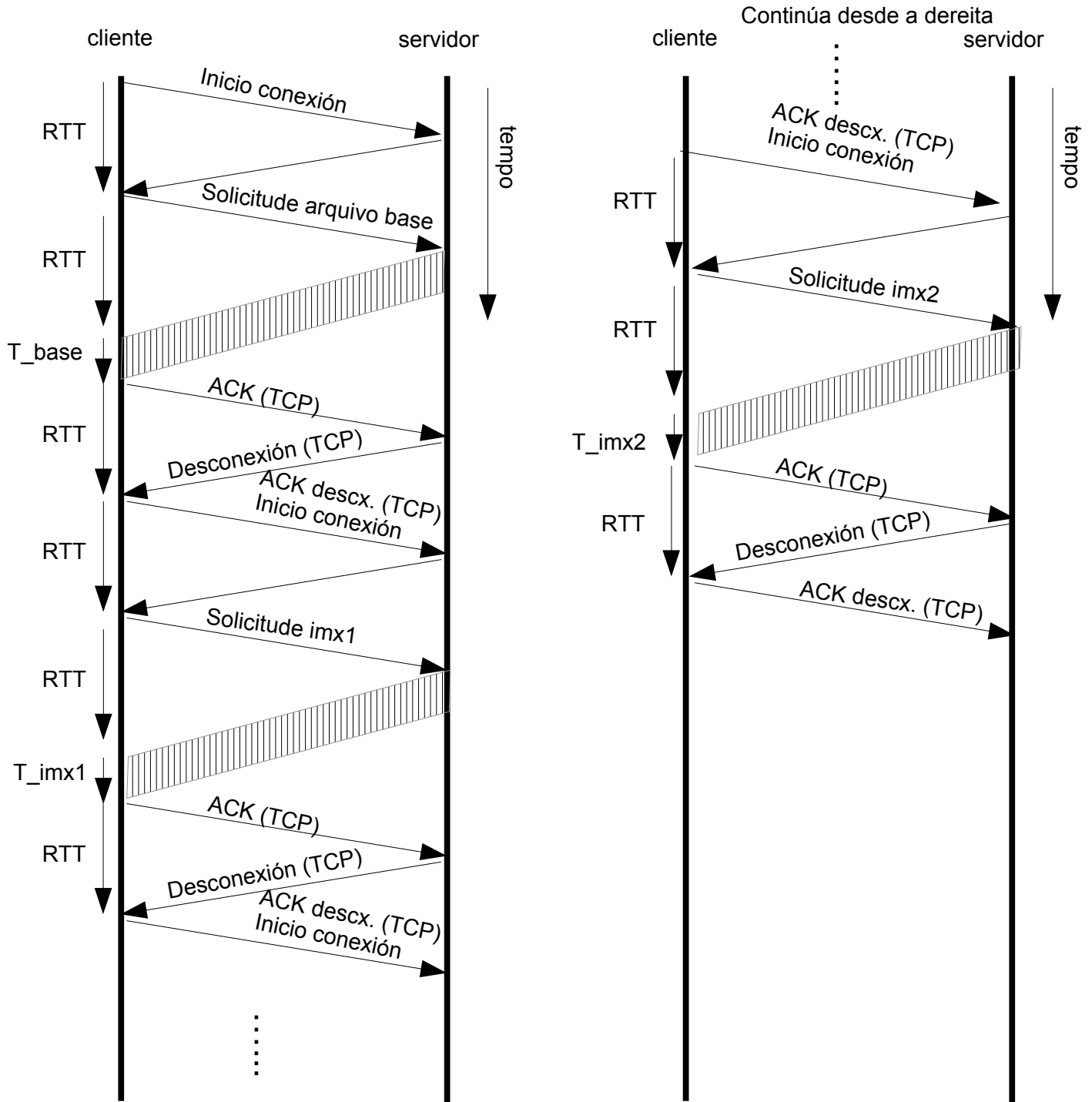
Persistente – Con entubamento



Tempo total (sen ent.) = $4 \text{ RTT} + T_{\text{base}} + T_{\text{imx1}} + T_{\text{imx2}}$

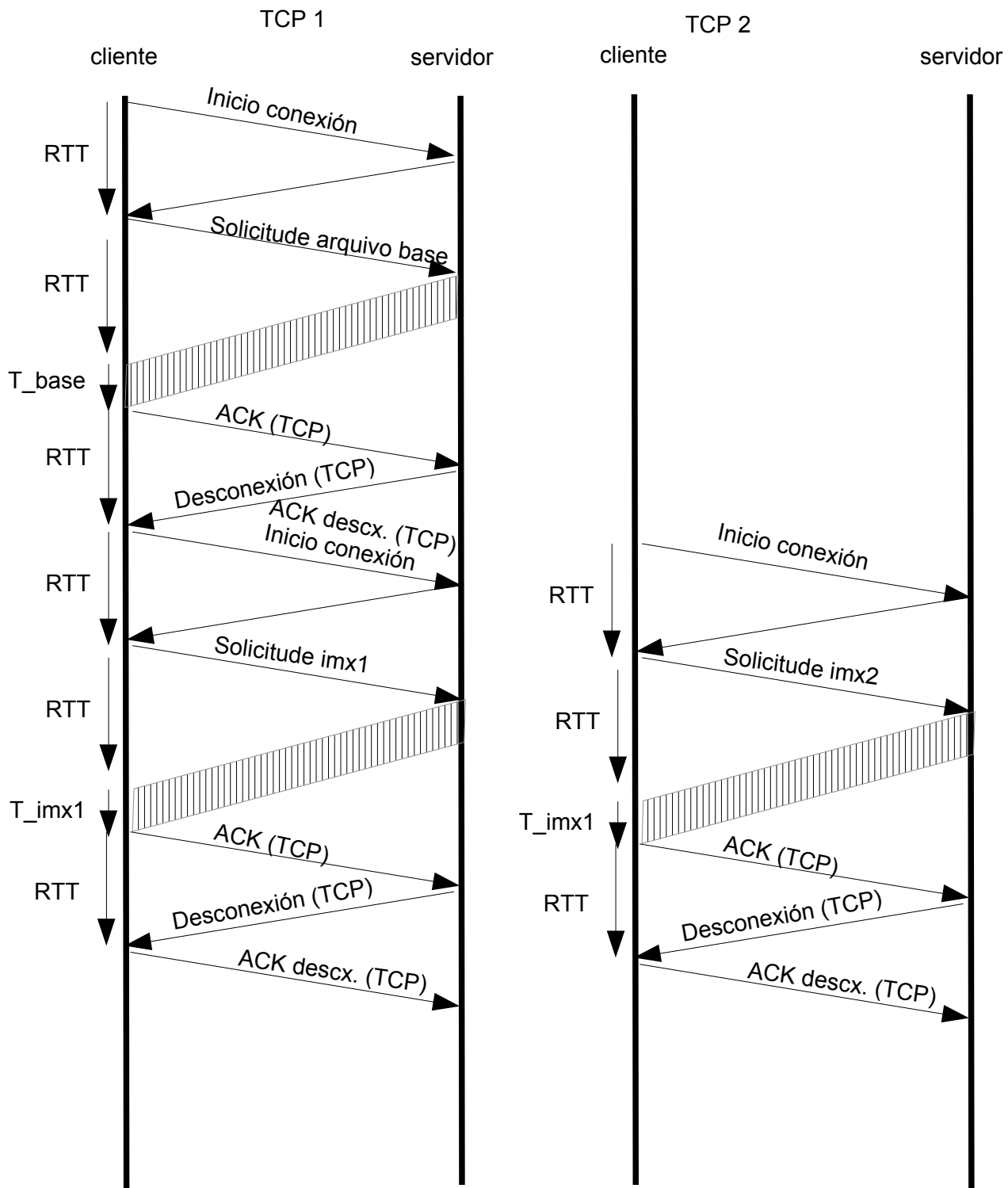
Tempo total (con ent.) = $3 \text{ RTT} + T_{\text{base}} + T_{\text{imx1}} + T_{\text{imx2}}$

Non Persistente – Serie (tendo en conta desconexión TCP explícita)



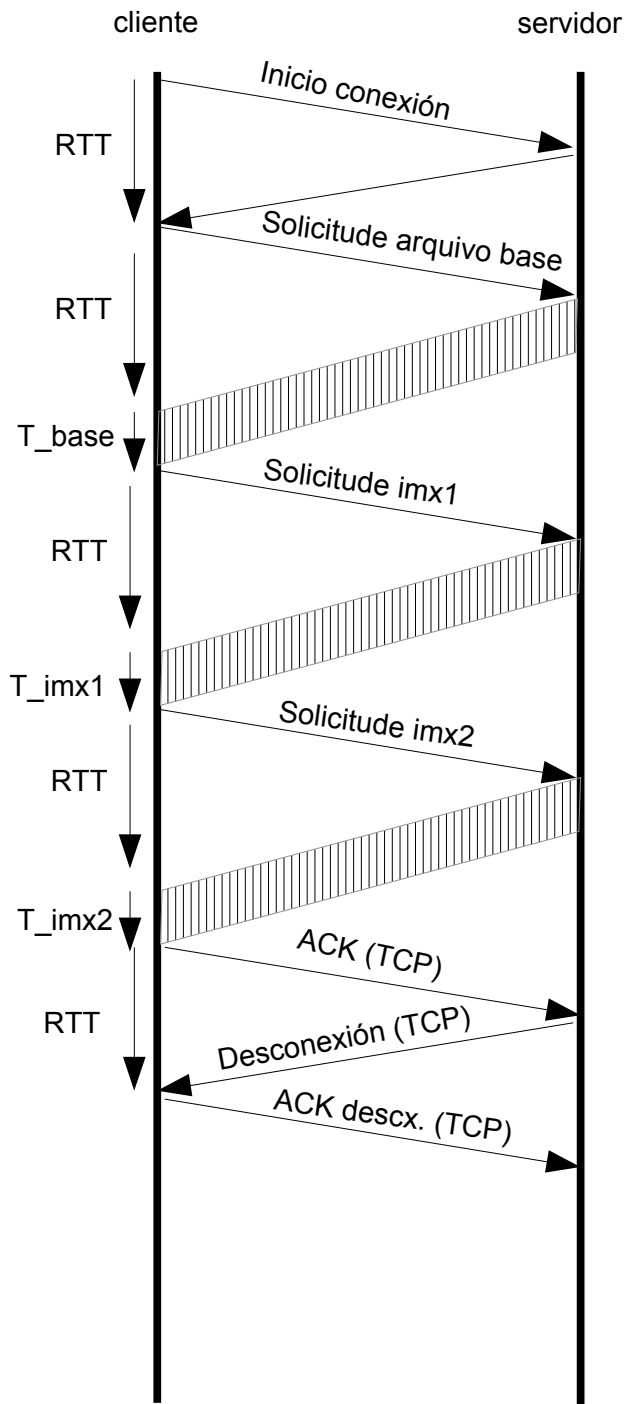
$$\text{Tempo total} = 9 \text{ RTT} + T_{base} + T_{imx1} + T_{imx2}$$

Non Persistente – Paralelo (tendo en conta desconexión TCP explícita)

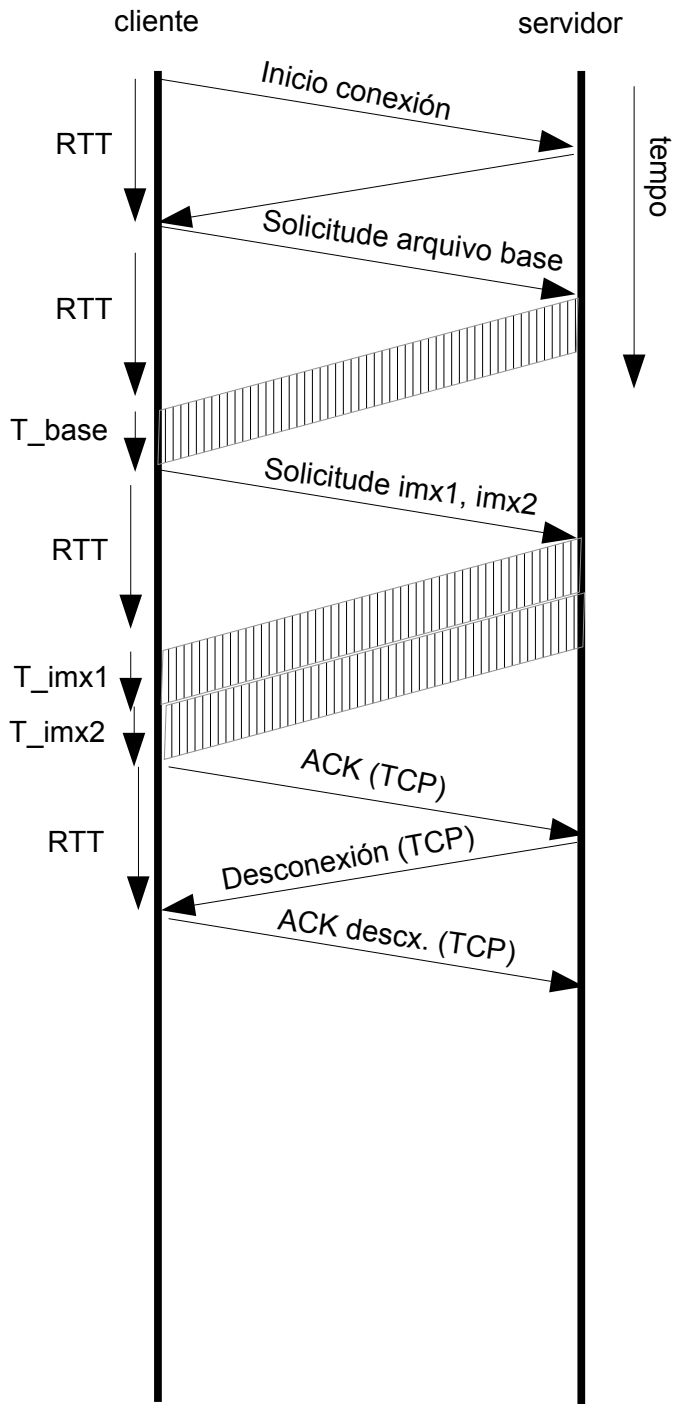


$$\text{Tempo total} = 6 \text{ RTT} + T_{\text{base}} + \max(T_{\text{imx1}} + T_{\text{imx2}})$$

Persistente – Sen entubamento
(tendo en conta desconexión TCP explícita)



Persistente – Con entubamiento
(tendo en conta desconexión TCP explícita)



Tempo total (sen ent.) = 5 RTT + T_base + T_imx1 + T_imx2

Tempo total (con ent.) = 4 RTT + T_base + T_imx1 + T_imx2