



Corso di
SISTEMI TELEMATICI
a.a. 2009-2010

Livello di Trasporto: esercizi

Slow start: Esempio 1

Quesito

Si consideri la procedura di Slow Start in una connessione TCP caratterizzata dai seguenti parametri: i) maximum segment size = 1000 byte; ii) RTT = 15 ms; iii) ampiezza finestra di ricezione comunicata dal ricevitore = 12000 byte.

Si chiede di determinare (1) l'intervallo di tempo necessario affinché lato sorgente sia raggiunta l'ampiezza massima della finestra di emissione a seguito dell'applicazione della procedura di slow start all'inizio della connessione; (2) il numero di byte trasmessi dopo 75 ms

Si considerino le seguenti ipotesi: a) nessun segmento è perso, b) i riscontri sono emessi immediatamente, c) i tempi di trasmissione dei segmenti e dei riscontri sono trascurabili.

Slow start: Esempio 1

- (1) La finestra comunicata dal ricevitore RW:

$$RW = m * MSS = 12 * MSS \text{ byte} = 12000 \text{ byte}$$

- CW raggiunga il valore di regime dopo $n * RTT$, con $n > \log_2 m$, cioè dopo $4 RTT = 60 \text{ ms}$

- (2) $75 \text{ ms} = 5 * RTT$, quindi include sia la fase di slow start che di regime

- Definiamo $n^\#$ il numero intero immediatamente inferiore a $\log_2 m$:

$$n^\# = \lfloor \log_2 m \rfloor = \lfloor \log_2 12 \rfloor = 3$$

- Il numero di byte trasmessi nella fase di slow start è:

$$\begin{aligned} MSS * (2^{n^\# + 1} - 1) &= MSS * (2^{\lfloor \log_2 m \rfloor + 1} - 1) = \\ &= 1000 * (2^4 - 1) = 15000 \text{ byte} \end{aligned}$$

Slow start: Esempio 1

- Il numero di byte trasmessi nella fase di regime è:

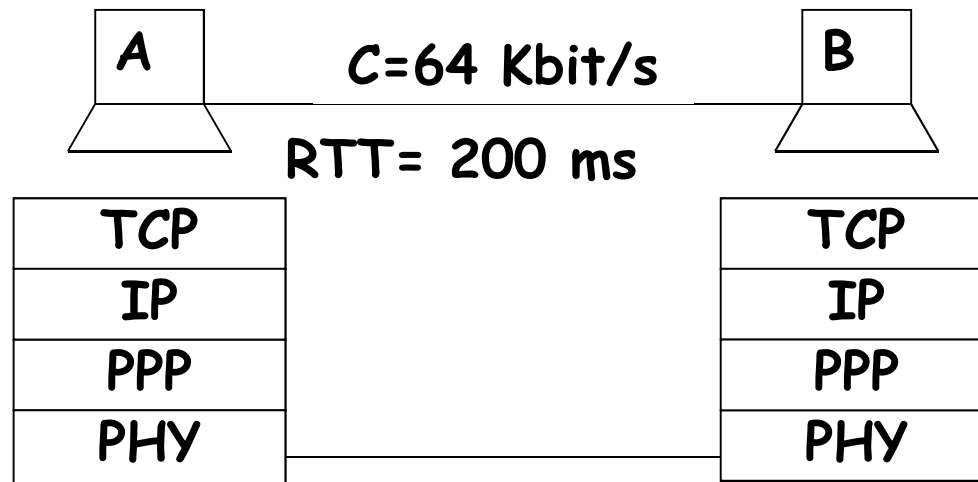
$$(5-n^{\#}) * m * MSS = (5-3) * 12 * 1000 = 24000 \text{ byte}$$

- Quindi in totale in 75 ms vengono trasmessi $15000 + 24000 \text{ byte} = 39000 \text{ byte}$

$$MSS * (2^{\lfloor \log_2 m \rfloor + 1} - 1) + (5-n^{\#}) * m * MSS$$

TCP: Esempio 1

- La stazione A trasferisce byte verso la stazione B su un link di capacità $C=64$ Kbit/sec, con $RTT=200$ msec



- MSS è 512 byte. Per trasmettere un segmento TCP da 512 byte bisogna aggiungere un header TCP di 20 byte, un header IP di 20 byte (se non si hanno opzioni) e un header PPP di 8 byte, per cui al livello fisico è inviata una trama di 560 byte; a causa dell'overhead si ha un fattore di merito pari a:

$$\eta = 512/560 = 0.914$$

TCP: Esempio 1

- Il tempo di trasmissione di un segmento t_f è uguale a:

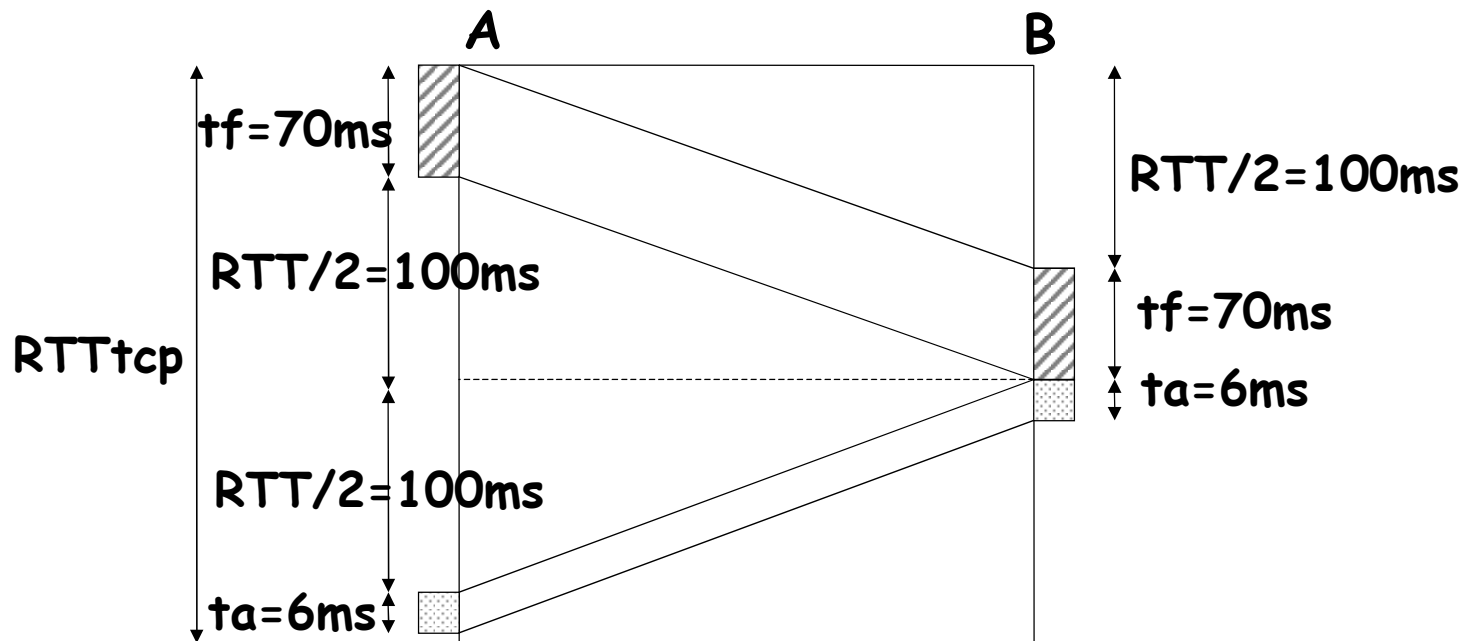
$$t_f = (MSS/h)/C = (560*8) \text{ [bit]} / 64.000 \text{ [bit/s]} = 70 \text{ ms}$$

- Il tempo di trasmissione t_a di un Ack, costituito solo dall'header TCP di 20 byte (più header IP e PPP), cioè una trama di 48 byte a livello fisico, è uguale a:

$$t_a = 48*8 \text{ bit} / (64.000 \text{ bit/s}) = 6 \text{ ms}$$

- Se si trascura il tempo di processamento e si suppone che gli ACK vengano inviati per ogni trama, che la rete sia scarica e il collegamento dedicato, si ha il diagramma temporale mostrato in figura

TCP: Esempio 1

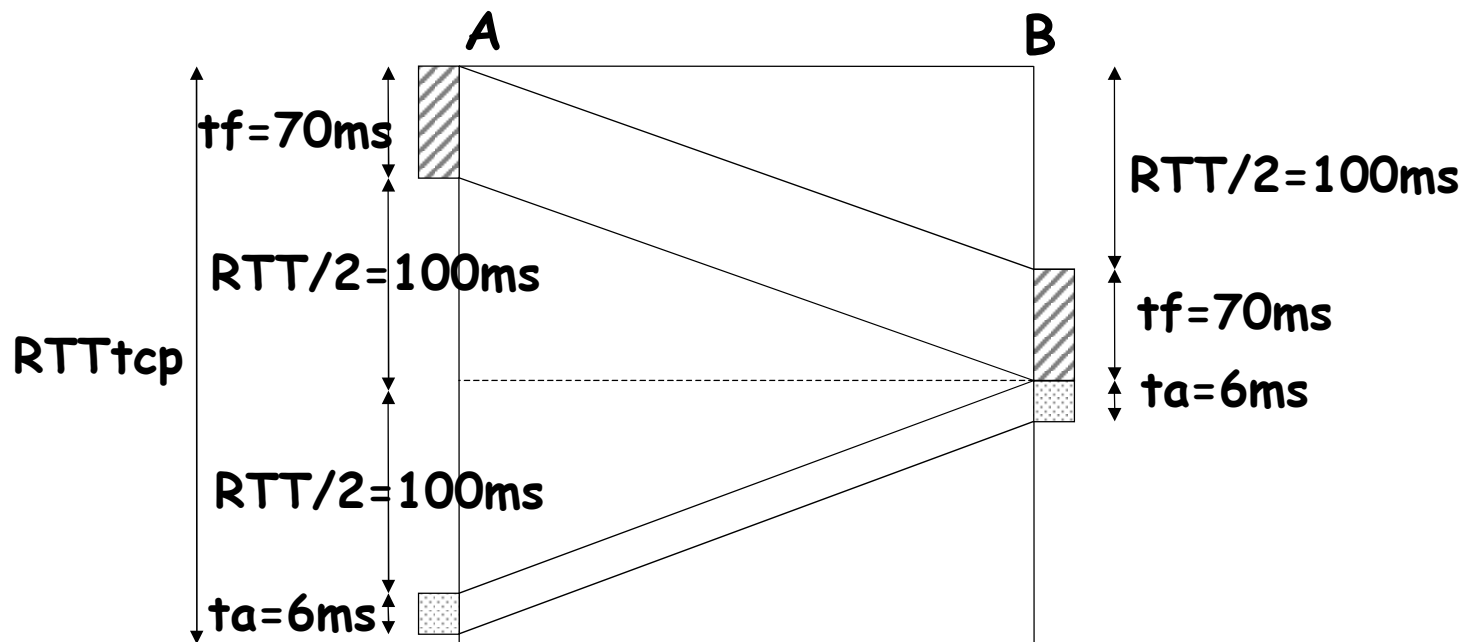


- RTT per il TCP è il tempo che intercorre dall'invio del 1° bit del segmento alla ricezione dell'ultimo bit dell'ACK:

$$RTT_{TCP} = RTT + t_f + t_a = 276 \text{ msec}$$

- $RTT_{TCP} > RTT$ tiene conto dei tempi di propagazione e di trasmissione (RTT_{TCP} deve tener conto anche dei tempi di elaborazione negli host e dei ritardi di accodamento nella rete)

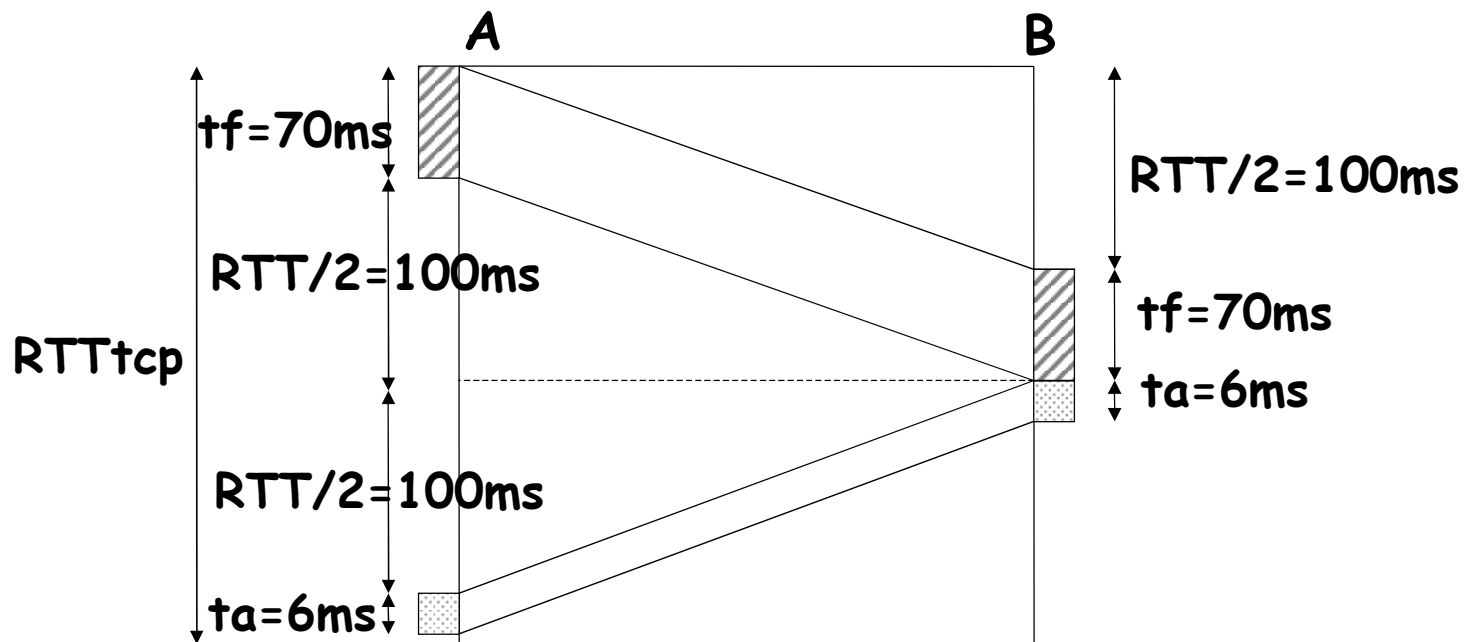
TCP: Esempio 1



- Qual è la dimensione minima della finestra del TCP per sfruttare tutta la capacità del link?
- La minima finestra di trasmissione W è data dal prodotto BANDA-RITARDO:

$$W_{\text{MIN}} = C \cdot RTT_{\text{TCP}} = 64.000 \cdot 0,276 \text{ s} = 17.644 \text{ bit} = 2.208 \text{ byte}$$

TCP: Esempio 1



- Sovradimensionare W non crea problemi, invece, se $W=1024$ byte, tutta la finestra è trasmessa in un tempo RTT_{TCP} e:

$$\text{Velocità eff.} = W * RTT_{TCP} = 1.024 * 8 \text{ bit} / 0,276 \text{ s} = 29.681 \text{ bit/s}$$

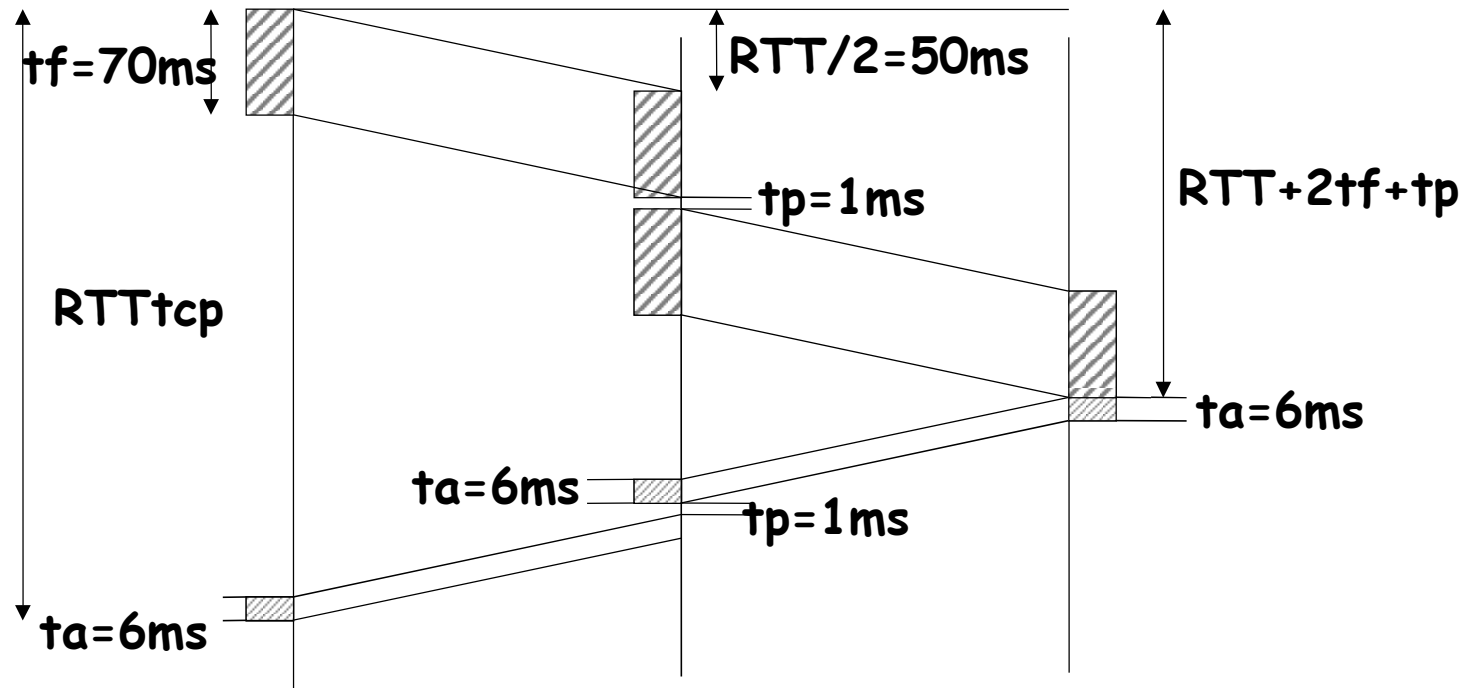
TCP: Esempio 2



- Se fra gli host A e B c'è un router, la situazione cambia
- I router possono essere di due tipi:
 - Store & Forward
 - Cut-Through
- Il router Store & Forward spedisce un pacchetto, dopo averlo interamente ricevuto ed elaborato (supponiamo un tempo di processamento t_p di 1 ms)
- Il router Cut-Through elabora soltanto gli header IP e PPP (supponiamo un tempo di processamento t_p 1 ms)

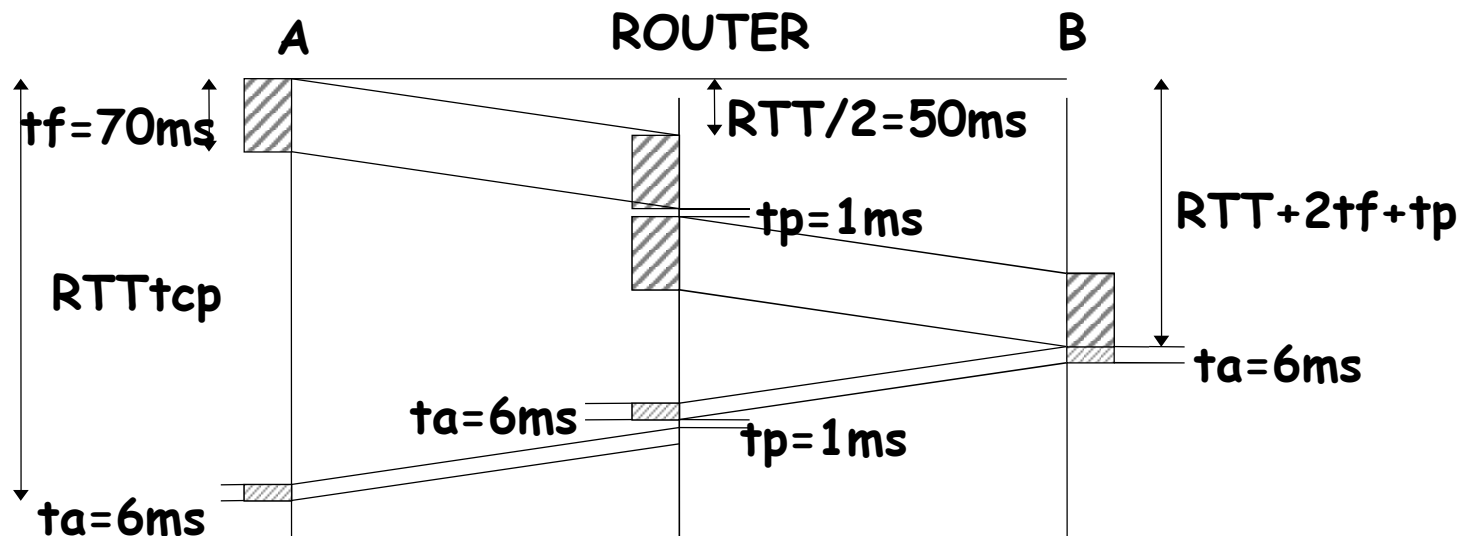
TCP: Esempio 2

A ROUTER STORE&FORWARD B



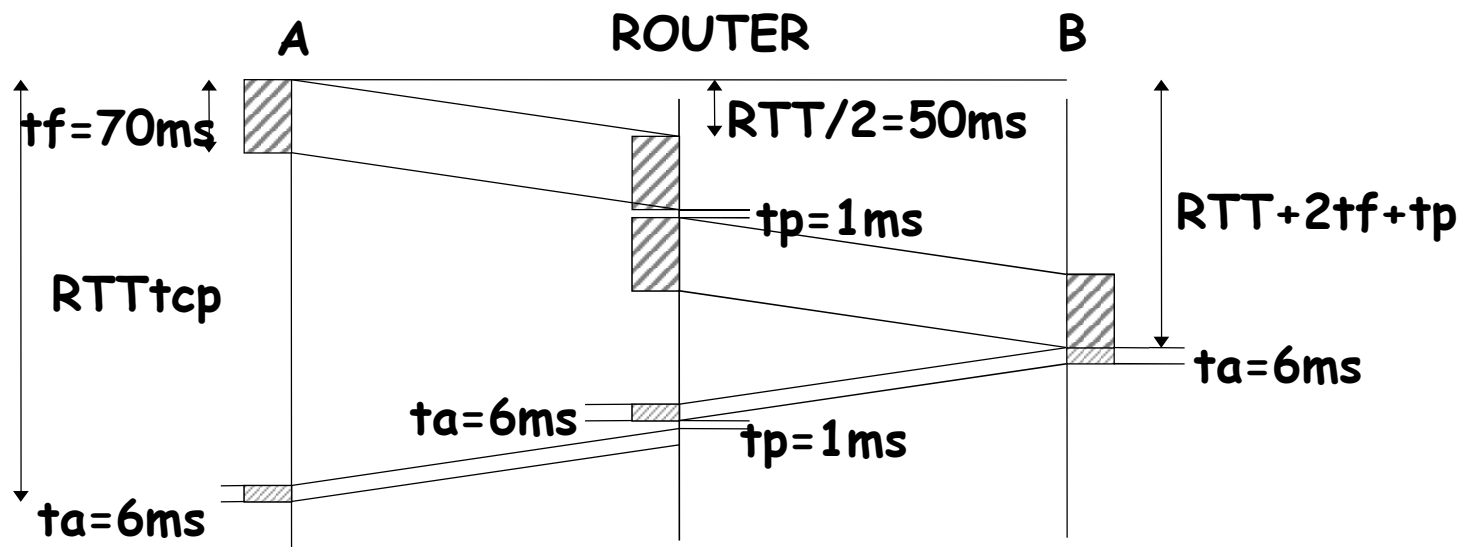
$$RTT_{TCP} = (70 + 50 + 1 + 70 + 50 + 6 + 50 + 1 + 50 + 6)\text{ms} = 354\text{ ms}$$

TCP: Esempio 2



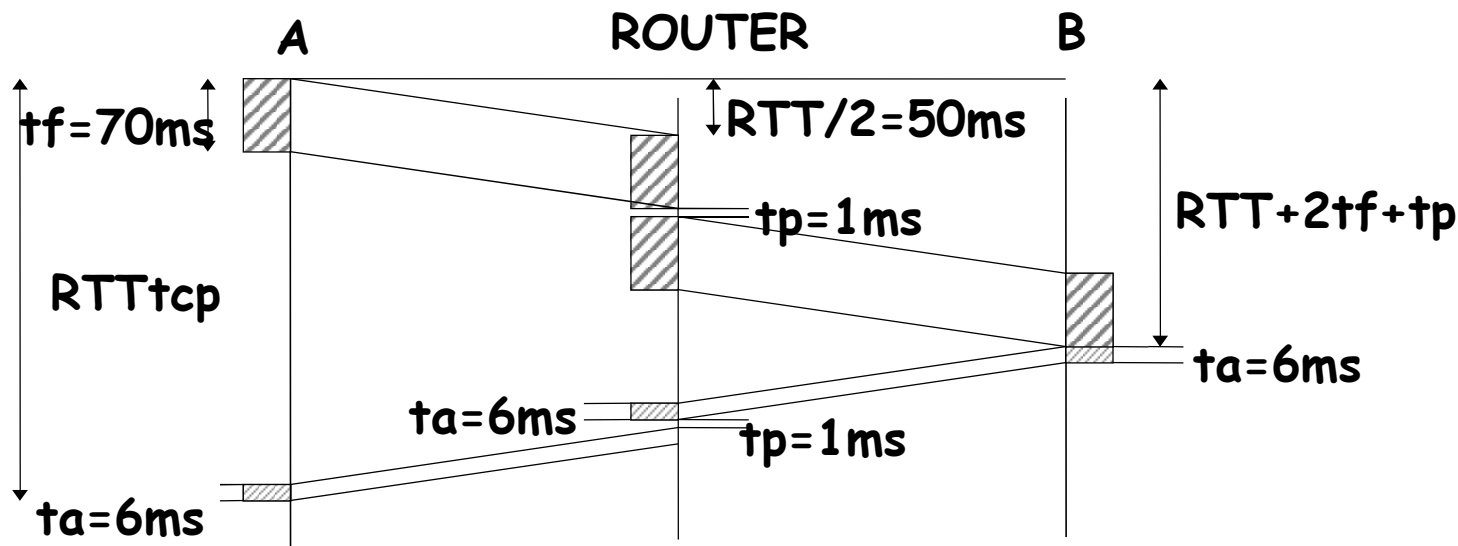
1. $t=0$ ms \rightarrow A inizia la trasmissione della trama
2. $t=50$ ms \rightarrow Il router riceve il 1° bit della trama
3. $t=70$ ms \rightarrow A trasmette l'ultimo bit della trama
4. $t=120$ ms \rightarrow Il router riceve l'ultimo bit della trama
5. $t=121$ ms \rightarrow Il router, finito il processamento, invia il 1° bit della trama a B
6. $t=171$ ms \rightarrow B riceve il 1° bit della trama
7. $t=191$ ms \rightarrow Il router invia a B l'ultimo bit della trama
8. $t=241$ ms \rightarrow B riceve l'ultimo bit della trama e invia l'ACK ($t_{p,ACK}=0$)

TCP: Esempio 2



9. $t=247\text{ ms} \rightarrow$ B ha trasmesso l'ultimo bit dell'ACK al router
10. $t=291\text{ ms} \rightarrow$ Il router riceve il 1° bit dell'ACK
11. $t=297\text{ ms} \rightarrow$ Il router riceve l'ultimo bit dell'ACK
12. $t=298\text{ ms} \rightarrow$ Dopo $t_{PACK}=1\text{ms}$ il router inizia a mandare l'ACK ad A
13. $t=304\text{ ms} \rightarrow$ Il router invia l'ultimo bit dell'ACK ad A
14. $t=348\text{ ms} \rightarrow$ A riceve il 1° bit dell'ACK
15. $t=354\text{ ms} \rightarrow$ A riceve l'ultimo bit dell'ACK

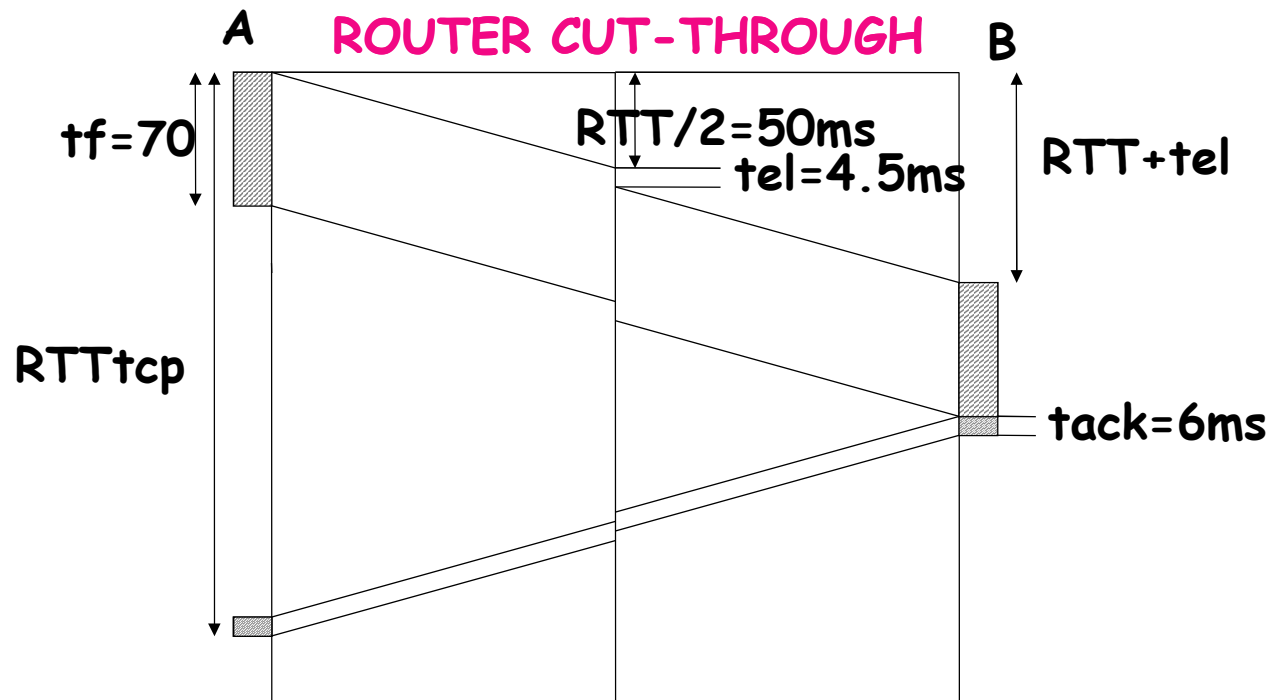
TCP: Esempio 2



- Il router Store & Forward ha quindi rallentato la trasmissione (354ms contro 276ms) peggiorando le prestazioni
- La minima dimensione della finestra che permette di sfruttare la capacità di trasmissione è:

$$W = C * RTT_{TCP} = 64.000 \text{ bit/s} * 0,354 * 1/8 \text{ byte/bit} = 2832 \text{ byte}$$

TCP: Esempio 2

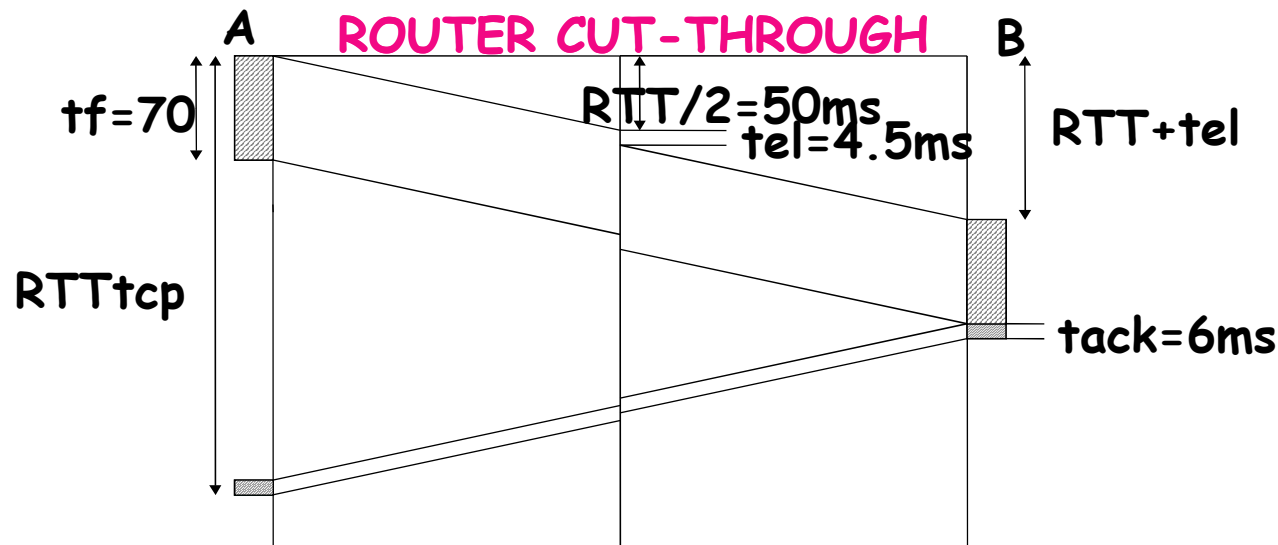


- Il router Cut-Through elabora gli header IP e PPP, e dopo un tempo di processamento t_p di 1 ms, inoltra il pacchetto; poiché esamina 28 byte, il tempo totale di processamento è:

$$t_{el} = t_p + 28 \cdot 8 \text{ bit} / 64.000 \text{ bit/s} = 4,5 \text{ ms}$$

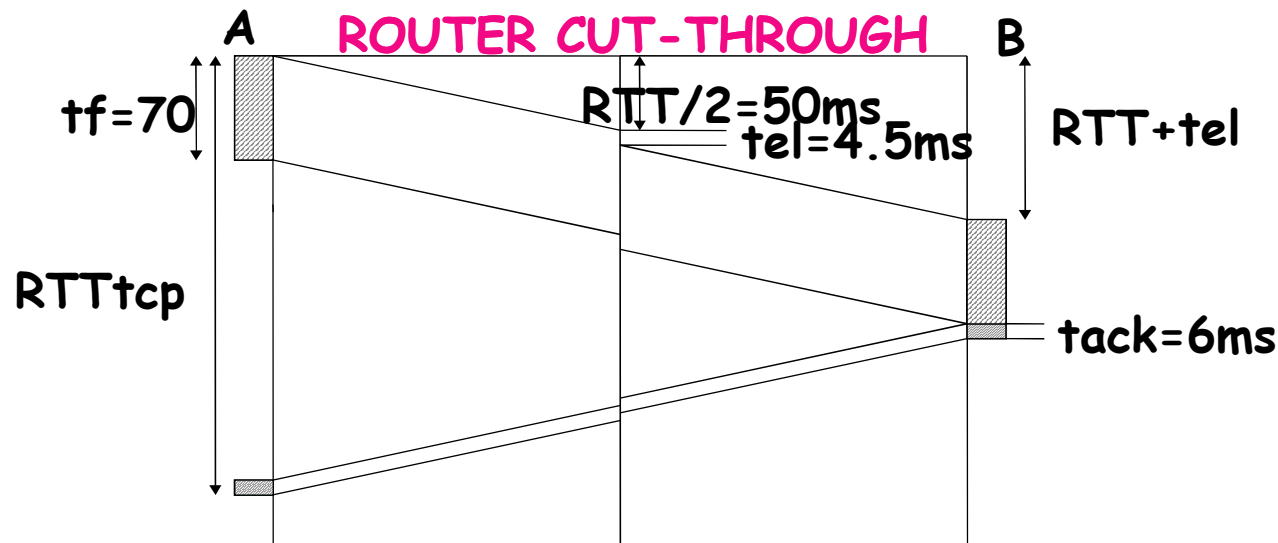
- Si consideri $t_f = 70$ ms, $t_{ACK} = 6$ ms e $RTT = 100$ ms

TCP: Esempio 2



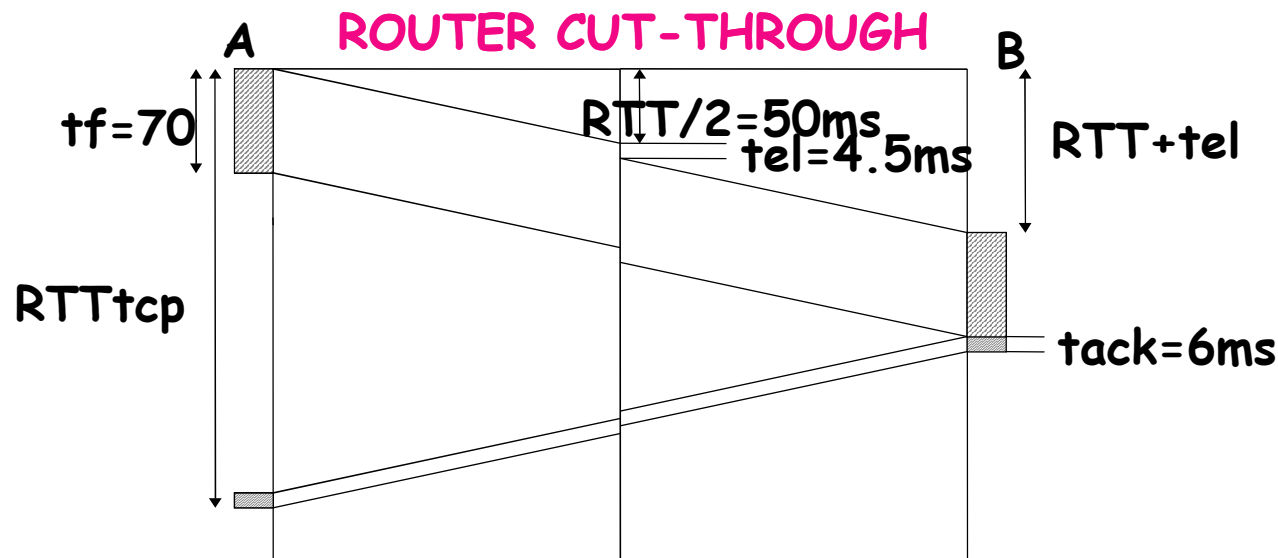
1. $t = 0 \text{ ms}$ → A trasmette il 1° bit della trama al router
2. $t = 50 \text{ ms}$ → Il router riceve il 1° bit spedito da A
3. $t = 54,5 \text{ ms}$ → Il router trasmette il 1° bit a B
4. $t = 70 \text{ ms}$ → A trasmette l'ultimo bit della trama al router
5. $t = 104,5 \text{ ms}$ → B riceve il 1° bit della trama spedita dal router
6. $t = 120 \text{ ms}$ → Il router riceve l'ultimo bit della trama spedita da A

TCP: Esempio 2



7. $t = 124,5$ ms \rightarrow Il router trasmette l'ultimo bit della trama a B
8. $t = 174,5$ ms \rightarrow B riceve l'ultimo bit dal router e spedisce il 1° bit dell'ACK
9. $t = 180,5$ ms \rightarrow B trasmette l'ultimo bit dell'ACK
10. $t = 224,5$ ms \rightarrow Il router riceve il 1° bit dell'ACK
11. $t = 229$ ms \rightarrow Il router invia il 1° bit dell'ACK ad A
12. $t = 230,5$ ms \rightarrow Il router riceve l'ultimo bit dell'ACK da B
13. $t = 235$ ms \rightarrow Il router invia l'ultimo bit dell'ACK ad A

TCP: Esempio 2



14. $t=279$ ms \rightarrow A riceve il 1° bit dell'ACK dal router

15. $t=285$ ms \rightarrow A riceve l'ultimo bit dell'ACK dal router

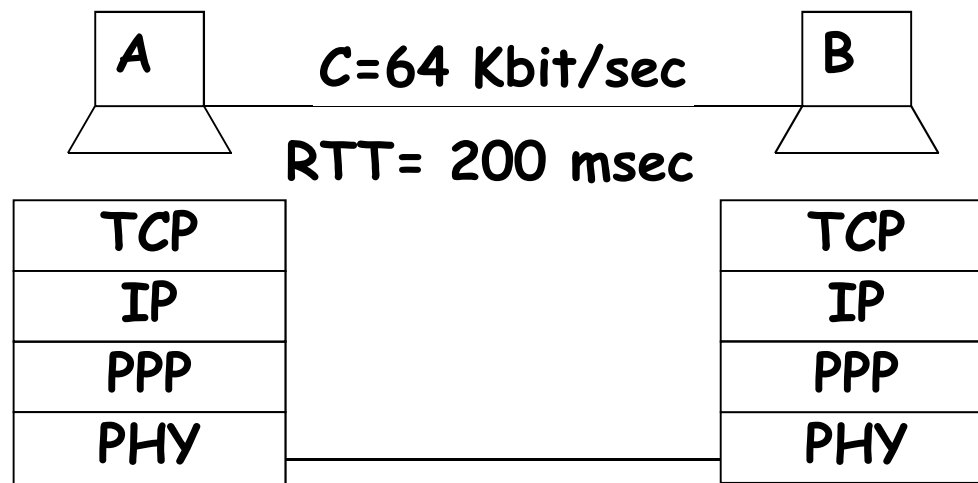
• L'RTT visto dal TCP risulta essere:

$$RTT_{TCP} = (70 + 200 + 6 + 4.5 + 4.5) \text{ msec} = 285 \text{ ms}$$

• Quindi un router Cut-Through ritarda meno la trasmissione rispetto ad un router Store & Forward, tuttavia Store & Forward è il più usato perché è più semplice da implementare

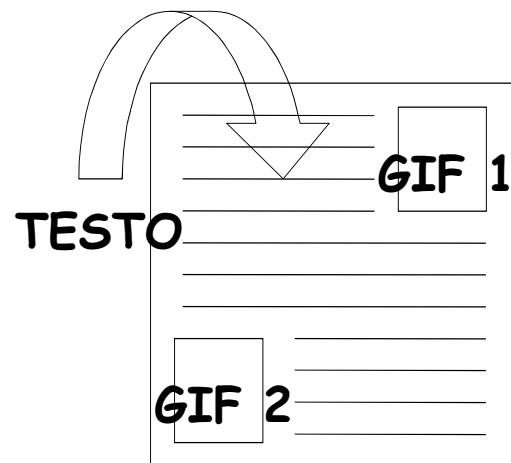
TCP: Esempio 3

- La stazione A trasferisce byte verso la stazione B su un link di capacità $C=64$ Kbit/sec, con $RTT=200$ msec



- Si supponga che A richieda a B il trasferimento di una pagina HTML costituita da testo e immagini
- Viene stabilita una finestra di trasmissione (e di ricezione) pari a $W = 5 \cdot MSS = 5 \cdot 512 = 2560$ byte

TCP: Esempio 3



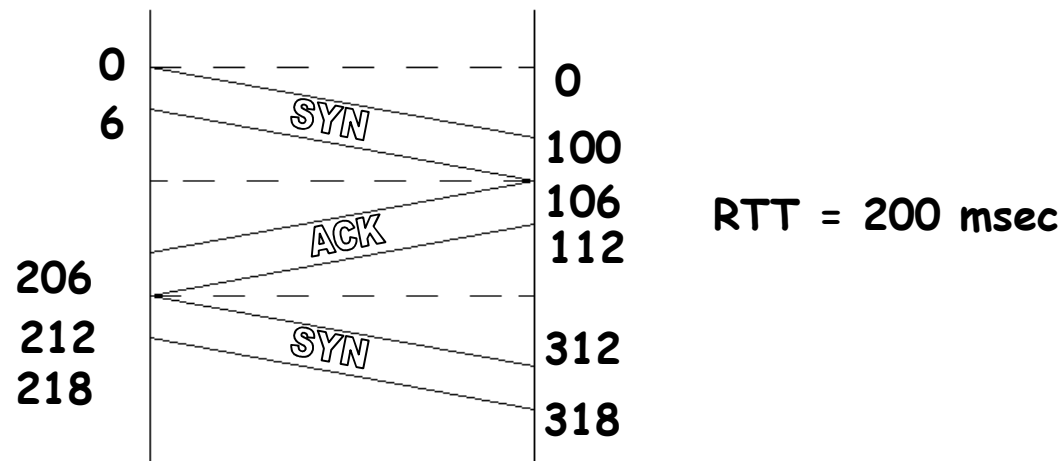
Il testo è costituito da 3584 byte

GIF 1 è un'immagine da 1024 byte

GIF 2 è un'immagine da 1536 byte

- L'UPPER LAYER PROTOCOL sia HTTP 1.0; in questo caso si aprono (e chiudono) 3 connessioni TCP, una per ogni porzione delle pagina (testo, Gif 1 e Gif2)
- Si assume che B invii un Ack per ogni segmento; dopo aver instaurato la connessione la trasmissione dei segmenti avviene in regime di Slow Start

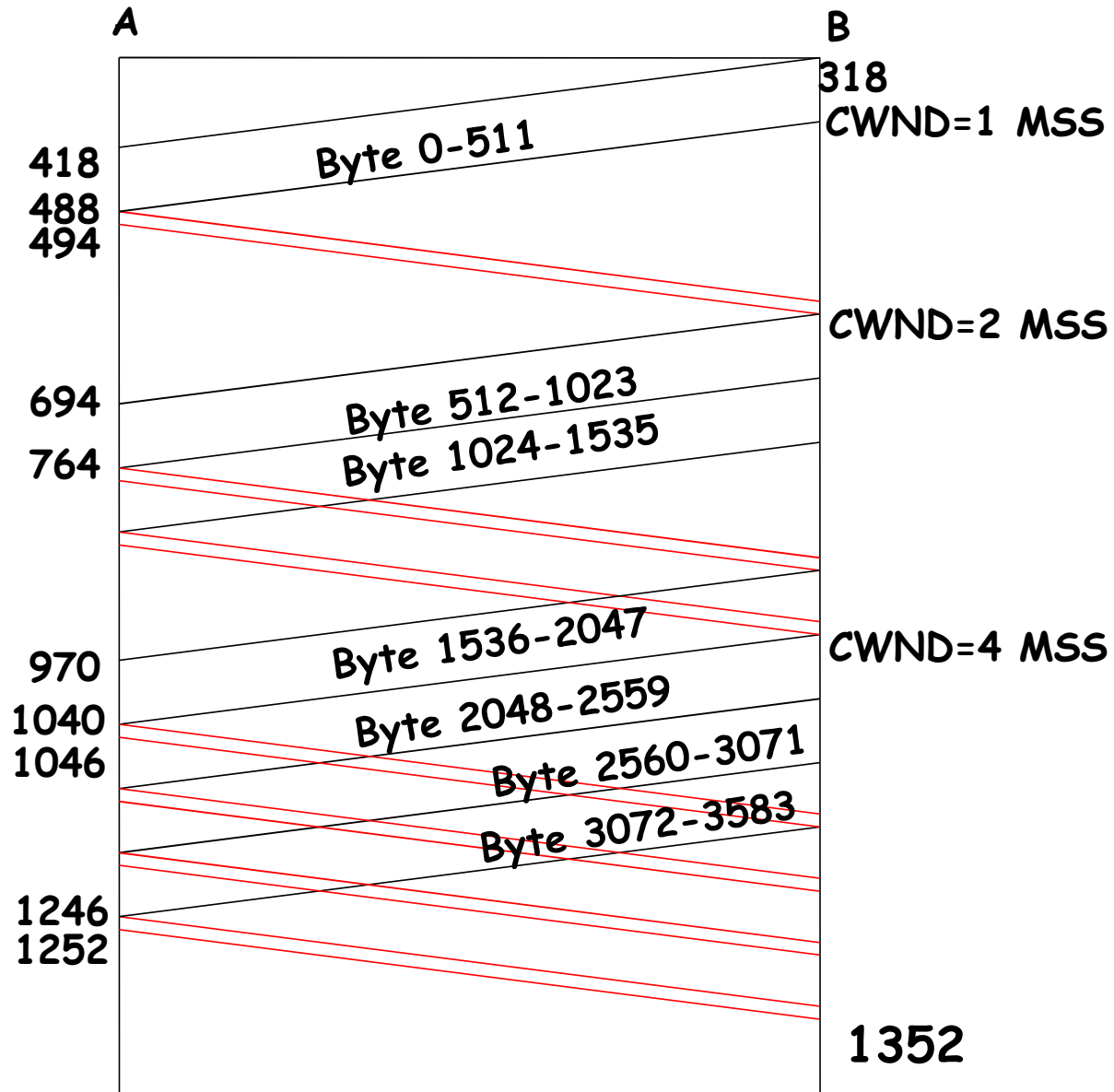
TCP: Esempio 3



- Per semplicità si separa la parte relativa all'instaurazione della connessione; i segmenti di tipo SYN e ACK sono costituiti dal solo header TCP di 20 byte, per cui:

$$t_{\text{ack}} = t_{\text{syn}} = 6 \text{ ms}$$
- Per instaurare la connessione in modalità *Three Way Handshake* servono 318 ms (trascurando i tempi di processamento)
- A questo punto, si invia la parte di testo iniziando la trasmissione in regime di Slow Start, con una finestra di congestione iniziale di 1 MSS (MSS=512 byte e $t_f=70 \text{ ms}$); ogni volta che B riceve un ACK aumenta CWND di 1MSS

TCP: Esempio 3



TCP: Esempio 3

- Dopo aver trasmesso il testo in 1,352s, viene chiusa la connessione (si trascuri il tempo di chiusura)
- Per trasmettere la prima immagine “Gif1” da 1024 byte viene aperta una nuova connessione in 0,318s e vengono trasmessi 1024 byte (corrispondenti a 2 MSS) in un tempo complessivo di 0,870s
- Lo stesso vale per la seconda immagine “Gif2” da 1536 byte (a cui corrispondono 3 MSS) che viene trasmessa in 0,94s
- Il tempo totale per trasferire l'intera pagina HTML è:

$$t_{\text{TOTALE}} = (1,352 + 0,87 + 0,94) \text{ s} = 3,162 \text{ s}$$

- A riceve completamente l'intera pagina HTML (senza considerare il tempo di ritorno dell'Ack) dopo un tempo globale di ricezione:

$$(1,25 + 0,764 + 0,834) \text{ sec} = 2,848 \text{ sec}$$

TCP: Esempio 3

Con la versione 2.0 di HTTP che instaura una sola connessione sia per il testo che per le immagini, la trasmissione dell'intera pagina HTML richiede il trasferimento di un numero di byte complessivo N_B :

$$N_B = (3584 + 1024 + 1536) \text{ byte} = 6144 \text{ byte}$$

- per trasmettere questi byte servono 12 segmenti e, considerando l'overhead per ogni segmento, si avranno 6722 byte; con una capacità trasmissiva di 64000 bit/sec. Se avessimo un unico Ack cumulativo sarebbero necessari 840,2 msec per trasmettere la pagina. In realtà anche in questo caso RTT_{TCP} è sicuramente maggiore dato il carico di pacchetti a cui è sottoposta la rete Internet.
- Se si suppone la stessa politica di Ack vista nell'esempio precedente, si trova un tempo globale di ricezione della pagina da parte di A pari a 1,706 s. Si vede, quindi, come HTTP 2.0 permetta una trasmissione più veloce rispetto ad HTTP 1.0.